

High-order Nodal Space-time Flux Reconstruction Methods for Hyperbolic Conservation Laws on Curvilinear Moving Grids

Meilin Yu^{a,*}

^a*Department of Mechanical Engineering,
University of Maryland, Baltimore County (UMBC), Baltimore, MD 21250*

Abstract

High-order nodal space-time flux reconstruction (STFR) methods have been developed to solve hyperbolic conservation laws on curvilinear moving grids. Unlike the method-of-lines approach for moving domain simulation, the grid velocity is implicitly embedded into the curvilinear geometric representation of space-time elements. Several key issues in moving domain simulation, including the discrete geometric conservation law (GCL), solution and flux approximation, and aliasing error control, are discussed in the context of the nodal STFR framework. Conditions and the corresponding numerical strategies to reduce aliasing errors due to the curvilinear space-time representation of moving domain problems, including the discrete GCL errors (i.e. one type of aliasing errors in the space-time framework), are then explained and examined. Since a space-time tensor product is used to construct the FR formulation in this study, all space-time schemes show the temporal superconvergence property, similar to that presented by the implicit Runge–Kutta discontinuous Galerkin (IRK-DG) schemes, in moving domain simulation. Specifically, a nominal k th order scheme can achieve a $(2k - 1)$ th order superconvergence rate when solutions on k Gauss–Legendre points are used to construct polynomials in the time dimension. The robustness of temporal superconvergence in the existence of aliasing errors induced by the curvilinear space-time representation, and upon de-aliasing operations based on polynomial filtering, has been examined with numerical experiments.

Keywords: Space-time Method, Flux Reconstruction, Moving Domain, Curvilinear Space-time Grid, Superconvergence, Projection-based Polynomial Filtering

1. Introduction

High-order computational fluid dynamics (CFD) methods [1] have achieved remarkable progress in recent years, enabling accurate simulation of multi-scale and multi-physics flow phenomena across a wide range of applications, including atmospheric dynamics, turbomachinery, plasma flows, and bio-inspired aerodynamics. Numerical simulation of fluid flows over moving domains presents significant challenges in the development of high-order CFD methods based on the arbitrary Lagrangian-Eulerian (ALE) formulation [2]. Extensive research has been conducted to improve high-order CFD methods for moving domain simulations [3–10]. A central focus in these efforts is to enforce the discrete geometric conservation law (GCL) in mapping from the time-dependent domain to a fixed reference domain in the ALE method. Therein, high-order accuracy is typically limited to spatial discretization, and the method-of-lines approach is then adopted for time integration.

The space-time formulation provides a consistent numerical treatment of both spatial and temporal discretizations, and can resolve GCL automatically to the level of the numerical resolution of the schemes used in the simulation. Although some research has been devoted to advancing space-time methods, such as the space-time discontinuous Galerkin (DG) method [11–17] and the space-time streamline-upwind/Petrov-Galerkin (SUPG) method [18–20], further work is required to fully exploit the numerical potential of space-

*Corresponding author
Email address: mlyu@umbc.edu (Meilin Yu)

time methods for moving-domain simulations. As is well known, the efficiency of explicit Runge–Kutta time-discretization schemes suffers from Butcher barriers (see **Theorem 324A–C** in [21]) when the order of accuracy exceeds four. Even some popular implicit Runge–Kutta (IRK) schemes, such as the explicit first stage, single diagonally implicit Runge–Kutta (ESDIRK) method [22], need to use more stages than the order in their design. Moreover, non-trivial efforts are needed to preserve their formal order of accuracy when extending time integration methods established on stationary grids to moving domain simulation [3]. Thus, understanding the convergence rate property of space-time methods, especially in the time dimension for moving domain simulation with large grid deformation, has practical importance. Another interesting feature of the space-time method for moving domain simulation is that the grid velocity, which is needed in the ALE formulation, does not explicitly show up in space-time schemes. Instead, it becomes the slope of the space-time element along its time dimension. Thus, the moving domain simulation problem becomes that of space-time simulation using general curvilinear elements, and discrete GCL errors become aliasing errors associated with the curvilinear space-time geometric representation. Therefore, we need to understand how the general curvilinear space-time element representation of moving grids affects simulation accuracy, and how to effectively control the associated numerical errors, especially aliasing errors. Some previous works [4, 23–25] on high-order simulation with curvilinear grids in the context of the method of lines have provided insights on possible solutions.

In this study, a high-order nodal tensor-product space-time flux reconstruction (STFR) method is developed for moving domain simulation. We mention that an implicit space-time method with right Radau points in the time dimension has been developed by Huynh [26] for conservation laws. Recently, Huynh [27] showed the equivalence between the DG-type discretization methods of ordinary differential equations (ODEs) and several IRK methods, including Radau IA, Radau IIA [28], and DG-Gauss [29, 30] with the assistance of the flux reconstruction (FR) concept. Note that FR was originally developed by Huynh [31, 32] for compact spatial discretization of partial differential equations (PDEs), and is a generalization of various types of discontinuous finite/spectral element methods, such as DG [33–35], spectral difference [36, 37], and spectral volume [38]. The FR method for spatial discretization has been well established; see some fundamental contributions in [39, 40] and a review [41] that summarized FR-related numerical algorithm developments and applications by 2014. In [42], we developed a nodal STFR method based on Gauss–Legendre points to solve hyperbolic conservation laws on stationary grids. Temporal superconvergence with a convergence rate of $2k - 1$ for a degree $k - 1$ (i.e. a nominal k th local order of accuracy) polynomial construction in the time dimension was observed. According to [27], in the time dimension the STFR scheme based on Gauss–Legendre points is equivalent to the IRK DG-Gauss scheme, which also has a superconvergence rate of $2k - 1$ for a k stage construction. In [43], we extended the STFR method for moving domain simulation with linear space-time grids, and discussed its connection with the dynamic grid formulation based on the method of lines. In this work, we developed a rigorous mathematical foundation for the nodal tensor-product STFR method for moving domain simulation with general curvilinear space-time grids.

The remainder of the paper is organized as follows. In **Sect. 2**, the basic idea of the nodal tensor-product STFR method on general curvilinear moving grids is explained. In **Sect. 3**, the connection between the general curvilinear space-time element representation of moving/deformable grids and GCL is discussed. It reveals the space-time numerical resolution requirement to enforce GCL automatically. Following that, requirements on the solution and flux approximation for curvilinear space-time grids are discussed in **Sect. 4**. Therein, different solution and flux approximation methods are discussed, and a new prospective is presented on how to design numerical schemes that satisfy the discrete GCL on general curvilinear space-time grids. Space-time projection-based polynomial filtering is then introduced to control aliasing errors associated with the STFR schemes. In **Sect. 5**, numerical experiments are conducted to examine key numerical features of the nodal tensor-product STFR method for moving domain simulation with curvilinear space-time grids. Finally, **Sect. 6** concludes this study, and points out potential future research directions.

2. Numerical Methods

Consider the general hyperbolic conservation law in conservation form,

$$\frac{\partial \mathbf{Q}}{\partial t} + \nabla \cdot \mathbf{F}(\mathbf{Q}) = 0, \quad (1)$$

defined on $\Omega \times [0, T)$ with the spatial domain Ω bounded by $\partial\Omega$, and T as the physical time covered by numerical simulation. Herein, $\mathbf{Q}$ is the vector of conserved variables, $\mathbf{F}$ is the spatial flux tensor, and ∇ is the gradient operator in the physical domain. For a d -dimensional spatial system with $d = 1, 2$, or 3 , $\mathbf{Q}$ is represented as $\mathbf{Q} = (Q_1, \dots, Q_{N_v})$, where N_v is the number of solution variables (i.e. conserved variables Q_i , $i = 1, \dots, N_v$, in this study). The flux tensor is written as $\mathbf{F} = [\mathbf{F}_1 \dots \mathbf{F}_d]$, where $\mathbf{F}_i$, $i = 1, \dots, d$, is the spatial flux in the i -th dimension. Let $\mathbf{x} = (x_1, \dots, x_d)$ be the spatial coordinates. Now we introduce the space-time domain $\Omega^{\text{st}} = \Omega \times [0, T)$ with the boundary $\partial\Omega^{\text{st}}$, and the space-time gradient operator $\nabla^{\text{st}} = (\partial_t, \partial_{x_1}, \dots, \partial_{x_d})$. Eq. (1) can then be written in the space-time domain as

$$\nabla^{\text{st}} \cdot \mathbf{F}^{\text{st}}(\mathbf{Q}) = 0, \quad (2)$$

where $\mathbf{F}^{\text{st}}$ is the space-time flux tensor, defined as $\mathbf{F}^{\text{st}} = [\mathbf{Q} \ \mathbf{F}_1 \dots \mathbf{F}_d]$. Note that as a naming convention, bold non-italic symbols, such as $\mathbf{F}^{\text{st}}$, denote tensors of dimension greater than one, bold italic symbols, such as $\mathbf{Q}$, denote vectors, and italic non-bold symbols, such as Q_i , denote scalars.

To numerically solve Eq. (2), the space-time domain Ω^{st} can be divided into N_e^{st} non-overlapping space-time elements Ω_e^{st} , $e = 1, \dots, N_e^{\text{st}}$, with the element boundary denoted as $\partial\Omega_e^{\text{st}}$. Two approaches, namely the so-called physical domain approach and reference domain approach (see a complete discussion in **Sect. 4.1**), can then be used to implement Eq. (2). Next we use the reference domain approach to further explain the FR formulation for three-dimensional (3D) tensor-product space-time elements with two dimensions in space and one dimension in time.

2.1. Tensor-product space-time FR

Consider the hyperbolic conservation law in 2D Cartesian space with $\mathbf{F}_1 = \mathbf{F}$ and $\mathbf{F}_2 = \mathbf{G}$,

$$\frac{\partial \mathbf{Q}}{\partial t} + \frac{\partial \mathbf{F}}{\partial x} + \frac{\partial \mathbf{G}}{\partial y} = 0, \quad (3)$$

and transform the governing equation from the physical domain (t, x, y) to the reference (computational) domain (τ, ξ, η) with $\tau, \xi, \eta \in [-1, 1]$ as

$$\frac{\partial \tilde{\mathbf{Q}}}{\partial \tau} + \frac{\partial \tilde{\mathbf{F}}}{\partial \xi} + \frac{\partial \tilde{\mathbf{G}}}{\partial \eta} = 0, \quad (4)$$

with

$$\begin{cases} \tilde{\mathbf{Q}} = |J| \tau_t \mathbf{Q} + |J| \tau_x \mathbf{F} + |J| \tau_y \mathbf{G} \\ \tilde{\mathbf{F}} = |J| \xi_t \mathbf{Q} + |J| \xi_x \mathbf{F} + |J| \xi_y \mathbf{G} \\ \tilde{\mathbf{G}} = |J| \eta_t \mathbf{Q} + |J| \eta_x \mathbf{F} + |J| \eta_y \mathbf{G} \end{cases}. \quad (5)$$

Herein, J is the Jacobian matrix for the space-time coordinate transformation, which takes the form:

$$J = \frac{\partial(t, x, y)}{\partial(\tau, \xi, \eta)} = \begin{pmatrix} t_\tau & t_\xi & t_\eta \\ x_\tau & x_\xi & x_\eta \\ y_\tau & y_\xi & y_\eta \end{pmatrix}. \quad (6)$$

The so-called Jacobian $|J|$ is the determinant of the Jacobian matrix J . The inverse transformation of Eq. (6) must exist for a non-singular transformation, which can be related to the Jacobian matrix J as:

$$J^{-1} = \frac{\partial(\tau, \xi, \eta)}{\partial(t, x, y)} = \begin{pmatrix} \tau_t & \tau_x & \tau_y \\ \xi_t & \xi_x & \xi_y \\ \eta_t & \eta_x & \eta_y \end{pmatrix}. \quad (7)$$

The metrics $t_\tau, t_\xi, t_\eta, x_\tau, x_\xi, x_\eta, y_\tau, y_\xi, y_\eta$ within each space-time element Ω_e^{st} can be calculated when a specific coordinate transformation $t(\tau, \xi, \eta)$, $x(\tau, \xi, \eta)$, and $y(\tau, \xi, \eta)$ is given, and $\tau_t, \tau_x, \tau_y, \xi_t, \xi_x, \xi_y, \eta_t, \eta_x, \eta_y$ can then be determined by evaluating J^{-1} correspondingly. Two examples of high-order curvilinear tensor-product space-time elements for grids with a general motion and the associated coordinate transformations are presented in Figure 1. Note that the curvilinear space-time elements are represented with polynomials along each dimension. To facilitate implementation, it is required that the polynomial degrees along each spatial dimension are the same, and they can be different from the polynomial degree along the temporal dimension. Since the geometric features of the space-time elements are always known (e.g. grid locations at any time are known when a prescribed motion is used or grid motion needs to be specified and iterated when two-way coupled fluid-structure interaction is performed), Gauss-Lobatto points along each dimension are used to describe the geometry of the high-order curvilinear space-time elements.

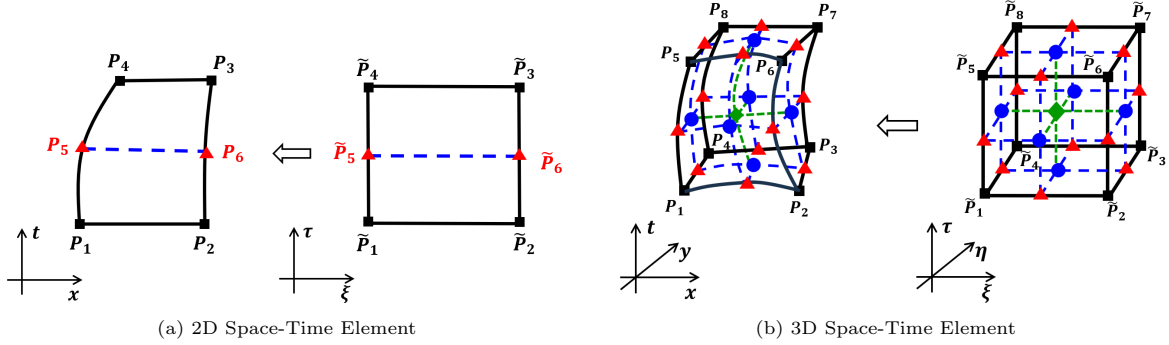


Figure 1: High-order curvilinear (a) 2D and (b) 3D tensor-product space-time elements. For the 2D space-time element in (a), the P^1 polynomial is used to construct the 1D spatial element, and the P^2 polynomial is used to construct the curve along the time dimension. For the 3D space-time element in (b), P^2 polynomials are used to construct the spatial and temporal curves along each dimension. Note that the edge, surface, and volume points for the 3D space-time element are not numbered in (b).

In space-time FR, each flux vector within a space-time element Ω_e^{st} is decomposed into a local component $(*)^{\text{loc}}$, constructed from the solution inside the element, and a correction component $(*)^{\text{cor}}$, formed using information from the element interfaces. The correction fluxes account for the discrepancies between the local fluxes and the common (numerical) fluxes imposed on the element boundaries. The total space-time flux is thus given by

$$\begin{cases} \tilde{Q}(\tau, \xi, \eta) = \tilde{Q}^{\text{loc}}(\tau, \xi, \eta) + \tilde{Q}^{\text{cor}}(\tau, \xi, \eta) \\ \tilde{F}(\tau, \xi, \eta) = \tilde{F}^{\text{loc}}(\tau, \xi, \eta) + \tilde{F}^{\text{cor}}(\tau, \xi, \eta) \\ \tilde{G}(\tau, \xi, \eta) = \tilde{G}^{\text{loc}}(\tau, \xi, \eta) + \tilde{G}^{\text{cor}}(\tau, \xi, \eta) \end{cases}. \quad (8)$$

For a tensor-product space consisting of (τ, ξ, η) , we have

$$\begin{aligned} \tilde{Q}^{\text{cor}}(\tau, \xi, \eta) &= \left(\tilde{Q}_L^{\text{num}}(\xi, \eta) - \tilde{Q}^{\text{loc}}(-1, \xi, \eta) \right) g_L(\tau) + \left(\tilde{Q}_R^{\text{num}}(\xi, \eta) - \tilde{Q}^{\text{loc}}(1, \xi, \eta) \right) g_R(\tau), \\ \tilde{F}^{\text{cor}}(\tau, \xi, \eta) &= \left(\tilde{F}_L^{\text{num}}(\tau, \eta) - \tilde{F}^{\text{loc}}(\tau, -1, \eta) \right) g_L(\xi) + \left(\tilde{F}_R^{\text{num}}(\tau, \eta) - \tilde{F}^{\text{loc}}(\tau, 1, \eta) \right) g_R(\xi), \\ \tilde{G}^{\text{cor}}(\tau, \xi, \eta) &= \left(\tilde{G}_L^{\text{num}}(\tau, \xi) - \tilde{G}^{\text{loc}}(\tau, \xi, -1) \right) g_L(\eta) + \left(\tilde{G}_R^{\text{num}}(\tau, \xi) - \tilde{G}^{\text{loc}}(\tau, \xi, 1) \right) g_R(\eta). \end{aligned} \quad (9)$$

Herein, $g_{L/R}(\cdot)$ are the correction functions, which are required to satisfy the following conditions: $g_L(-1) = g_R(1) = 1$ and $g_L(1) = g_R(-1) = 0$. It is clear that the correction functions map the differences between the numerical fluxes $(*)^{\text{num}}$ and local fluxes $(*)^{\text{loc}}$ on the space-time element boundaries to the entire element.

Different correction functions can recover different numerical schemes [31]. For example, the right and left Radau polynomials are used as correction functions to recover the DG scheme in this study. In Eq. (9), $\tilde{\mathbf{Q}}_{L/R}^{\text{num}}$, $\tilde{\mathbf{F}}_{L/R}^{\text{num}}$, and $\tilde{\mathbf{G}}_{L/R}^{\text{num}}$ are the numerical fluxes on the left or right boundaries of the standard space-time element in the τ -, ξ - and η -direction, respectively. They are evaluated from the common normal fluxes $\mathbf{F}_n^{\text{st,com}}$ on the corresponding element surfaces in the physical domain as follows:

$$\begin{cases} \tilde{\mathbf{Q}}^{\text{num}} = |J| \|\nabla^{\text{st}} \tau\|_2 \text{sign}(\mathbf{n}^{\text{st}} \cdot \nabla^{\text{st}} \tau) \mathbf{F}_n^{\text{st,com}} \\ \tilde{\mathbf{F}}^{\text{num}} = |J| \|\nabla^{\text{st}} \xi\|_2 \text{sign}(\mathbf{n}^{\text{st}} \cdot \nabla^{\text{st}} \xi) \mathbf{F}_n^{\text{st,com}} \\ \tilde{\mathbf{G}}^{\text{num}} = |J| \|\nabla^{\text{st}} \eta\|_2 \text{sign}(\mathbf{n}^{\text{st}} \cdot \nabla^{\text{st}} \eta) \mathbf{F}_n^{\text{st,com}} \end{cases}, \quad (10)$$

where the function $\text{sign}(\cdot)$ takes the sign of the input, and $\|\cdot\|_2$ takes the ℓ^2 -norm of a vector input. Note that the normal space-time flux vector $\mathbf{F}_n^{\text{st}}$ is defined as $\mathbf{F}_n^{\text{st}} = \mathbf{n}^{\text{st}} \cdot \mathbf{F}^{\text{st}}$, where $\mathbf{n}^{\text{st}}$ is the outward-going unit normal vector of the space-time boundary $\partial\Omega_e^{\text{st}}$ for any space-time element Ω_e^{st} . On any surface of the space-time element, there exist two sets of values of $\mathbf{F}_n^{\text{st}}$ constructed separately from the two elements (including ghost elements over the domain boundaries) sharing the surface. Therefore, common normal fluxes $\mathbf{F}_n^{\text{st,com}}$ are calculated from the two sets of values of $\mathbf{F}_n^{\text{st}}$ to enforce the mass, momentum, and energy conservation laws at the element boundaries. In this study, the common inviscid fluxes at the element interfaces in the physical domain are calculated using the local Lax-Friedrichs (or Rusanov) approximate Riemann solver [44].

After substituting Eq. (8) into Eq. (4), the governing equations become

$$\frac{\partial \tilde{\mathbf{Q}}^{\text{loc}}}{\partial \tau} + \frac{\partial \tilde{\mathbf{F}}^{\text{loc}}}{\partial \xi} + \frac{\partial \tilde{\mathbf{G}}^{\text{loc}}}{\partial \eta} + \tilde{\boldsymbol{\delta}}^{\text{cor,st}} = 0, \quad (11)$$

where

$$\tilde{\boldsymbol{\delta}}^{\text{cor,st}} = \frac{\partial \tilde{\mathbf{Q}}^{\text{cor}}}{\partial \tau} + \frac{\partial \tilde{\mathbf{F}}^{\text{cor}}}{\partial \xi} + \frac{\partial \tilde{\mathbf{G}}^{\text{cor}}}{\partial \eta}. \quad (12)$$

Note that $\tilde{\boldsymbol{\delta}}^{\text{cor}}$ is called the correction field in standard FR, and with the superscript ‘st’, it is the space-time correction field. $\tilde{\boldsymbol{\delta}}^{\text{cor,st}}$ can be directly evaluated with the derivative of the correction functions, that is, $g'_{L/R}(\cdot)$, due to the special structure of the correction fluxes $(\cdot)^{\text{cor}}$ shown in Eq. (9).

Typical distributions of solution and flux points in a standard space-time element used to formulate STFR are shown in Figure 2. Therein, Gauss-Legendre points are used as solution points in each dimension, and flux points are defined as the intersections of the element boundaries with the coordinate-aligned lines drawn from each solution point. This should be distinguished from the Gauss-Lobatto points used to represent the curvilinear geometry of the space-time element displayed in Figure 1. We mention that the accuracy features of different treatments of solutions and fluxes can vary much, especially for curvilinear elements. A formal discussion of the polynomial approximation of solutions and fluxes, and their order of accuracy measure will be presented in **Sect. 4**.

To solve the STFR formulation (11), we augment it with a pseudo-time derivative term $\partial \tilde{\mathbf{Q}} / \partial \tilde{t}$, where $\tilde{t}$ is the pseudo-time. Define the residual vector $\tilde{\mathbf{R}}(\tilde{\mathbf{Q}})$ as the opposite of the left-hand side of Eq. (11). Thus, Eq. (11) can be written as

$$\frac{\partial \tilde{\mathbf{Q}}}{\partial \tilde{t}} = \tilde{\mathbf{R}}(\tilde{\mathbf{Q}}). \quad (13)$$

Both explicit and implicit pseudo-time marching methods can be used to solve Eq. (13). In this study, local pseudo-time stepping with the explicit two-stage, second-order strong stability preserving Runge-Kutta (SSPRK) method [45] is used to accelerate convergence. We mention that implicit pseudo-time marching methods [46], such as the backward Euler and second-order backward differentiation formula (BDF2), can also be used to solve Eq. (13) when the physical time step is sufficiently large. However, this is beyond the scope of the current work.

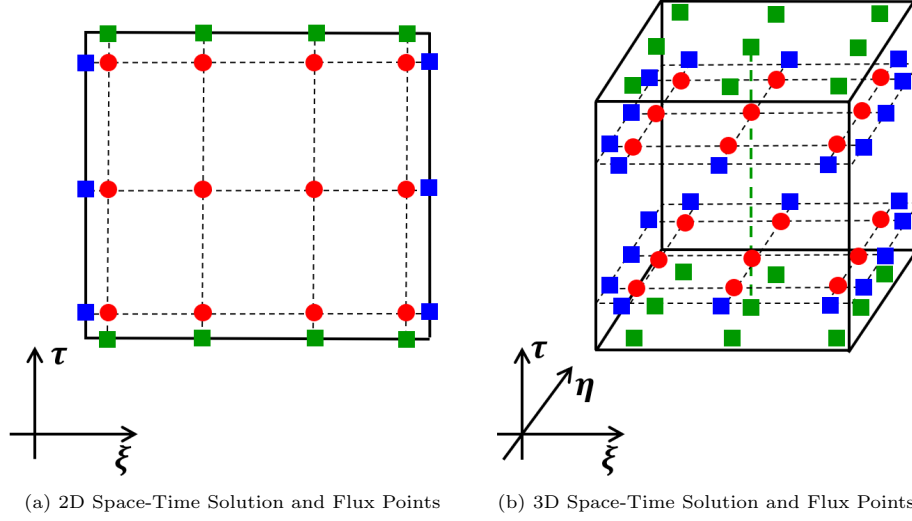


Figure 2: Distribution of solution points (circle) and flux points (square) in (a) 2D and (b) 3D tensor-product space-time elements. Gauss-Legendre points are used in each dimension to construct solution and flux polynomials. For the 2D space-time element in (a), the P^3 polynomial is used to construct the solution along the spatial dimension, and the P^2 polynomial is used to construct the solution along the time dimension. For the 3D space-time element in (b), P^2 polynomials are used to construct the solution along each (i.e., ξ - and η -) spatial dimension, and the P^1 polynomial is used to construct the solution along the time dimension.

3. Discussions of Moving Grid Simulations

In the space-time coordinate transformation from Eq. (3) to Eq. (4), the following geometric conservation laws are used:

$$\frac{\partial}{\partial \tau} \begin{pmatrix} |J|\tau_t \\ |J|\tau_x \\ |J|\tau_y \end{pmatrix} + \frac{\partial}{\partial \xi} \begin{pmatrix} |J|\xi_t \\ |J|\xi_x \\ |J|\xi_y \end{pmatrix} + \frac{\partial}{\partial \eta} \begin{pmatrix} |J|\eta_t \\ |J|\eta_x \\ |J|\eta_y \end{pmatrix} = 0. \quad (14)$$

3.1. Method of lines

When the method of lines is used for moving grid simulation, the metrics τ_x and τ_y are always zeros, i.e. $\tau(t)$ is only a function of t . Therefore, only the first equation of the GCL (14) is time-dependent. Moreover, the grid velocity $\mathbf{v}_g \equiv d\mathbf{x}/dt$ can be expressed with the metrics τ_t , x_τ , and y_τ through $(v_{g,x}, v_{g,y}) = (x_\tau \tau_t, y_\tau \tau_t)$, where $v_{g,x}$ and $v_{g,y}$ are the x - and y -component of $\mathbf{v}_g$, respectively. Therefore, the metrics ξ_t and η_t in Eq. (14) can be calculated from the grid velocity as:

$$\begin{cases} \xi_t = -\mathbf{v}_g \cdot \nabla \xi \\ \eta_t = -\mathbf{v}_g \cdot \nabla \eta \end{cases}. \quad (15)$$

Since the temporal discretization of $|J|\tau_t$ is usually not consistent with the spatial discretization of $|J|\xi_t$ and $|J|\eta_t$ in the first equation of the GCL (14), the so-called discrete GCL errors can contaminate moving grid simulation. In general, there are two common approaches to control GCL errors in high-order numerical simulations with the method of lines. In the work by Persson *et al.* [4], the first GCL formula in Eq. (14) was solved using the same numerical time integration scheme as that for the physical laws (3) or (4). The analytical $|J|$ in $\tilde{Q}$ is then substituted with the numerical $|J|$ calculated from the first GCL to cancel the GCL error. Another approach to eliminate the GCL error is to replace the term $\partial(|J|\tau_t)/\partial \tau$ in Eq. (4) directly with $-(\partial(|J|\xi_t)/\partial \xi + \partial(|J|\eta_t)/\partial \eta)$ in the scheme implementation. As a result, all GCL-related terms only have errors related to spatial discretization, which is high-order and consistent between relevant terms. A local source term $-\mathbf{v}_g \cdot \nabla Q$ is thus added to the governing equation to cancel the GCL error [6, 9].

To summarize, when the method of lines is used for moving grid simulation, the grid velocity needs to be calculated, and the time-dependent GCL needs to be enforced to ensure accuracy.

3.2. Space-time method

In the space-time method, the GCL (14) is automatically enforced, to the level of the numerical resolution, by the geometry of the space-time elements as shown in Figure 1. As will be discussed in Sect. 4, GCL can be either enforced by using special metric construction strategies [24, 47] or satisfied by embedding them exactly into the numerical schemes (see Sect. 4.2).

A special attention should be paid to the metrics t_ξ and t_η when using the tensor-product space-time curvilinear elements. In general, they are function of space and time; this indicates that the spatial grid points are allowed to move freely in the time dimension (see the space-time unstructured grid setup in [48–50]). Here we use the concept of space-time slabs $[t_j, t_j + \Delta t_j]$ with piece-wise constant time steps Δt_j , and all spatial grid points are only allowed to march with the same distance in the temporal direction within any space-time curvilinear element. As a result, the physical time t is only a (nonlinear) function of the reference time τ . The proof is straightforward as for a full set of Lagrange polynomials $\phi_i(\xi)$, $i = 1, \dots, i_{max}$, we have $\sum_{i=1}^{i_{max}} \phi_i(\xi) \equiv 1$. Therefore, $\tilde{\mathbf{Q}}$ in Eq. (5) is expressed as $\tilde{\mathbf{Q}} = |J|\tau_t \mathbf{Q}$. The inverse of the Jacobian matrix J^{-1} (2D space plus time) in Eq. (7) can then be calculated from the so-called non-conservative formulation:

$$|J|J^{-1} = |J| \begin{pmatrix} \tau_t & \tau_x & \tau_y \\ \xi_t & \xi_x & \xi_y \\ \eta_t & \eta_x & \eta_y \end{pmatrix} = \begin{pmatrix} x_\xi y_\eta - x_\eta y_\xi & 0 & 0 \\ -x_\tau y_\eta + y_\tau x_\eta & t_\tau y_\eta & -t_\tau x_\eta \\ x_\tau y_\xi - y_\tau x_\xi & -t_\tau y_\xi & t_\tau x_\xi \end{pmatrix}. \quad (16)$$

We then have the following conclusion:

Lemma 1. *The GCL (14) is automatically satisfied numerically when the geometric features of a curvilinear element can be completely represented by polynomials in a numerical scheme.*

Proof. Take the second formula, i.e.

$$\frac{\partial}{\partial \tau} (|J|\tau_x) + \frac{\partial}{\partial \xi} (|J|\xi_x) + \frac{\partial}{\partial \eta} (|J|\eta_x) = 0$$

as an example. Similar approaches can be used to prove other formulas in Eq. (14).

In the space-time slab setup, we have $\tau_x \equiv 0$. The coordinates $x(\xi, \eta, \tau)$, $y(\xi, \eta, \tau)$, and $t(\xi, \eta, \tau) \equiv t(\tau)$ of a curvilinear element can, in general, be written as

$$\begin{cases} x(\xi, \eta, \tau) = \sum_{i_1} \sum_{i_2} \sum_{i_3} L_{i_1, i_2, i_3}(\xi, \eta, \tau) x_{i_1, i_2, i_3} = \sum_{i_1} \sum_{i_2} \sum_{i_3} \phi_{i_1}(\xi) \phi_{i_2}(\eta) \phi_{i_3}(\tau) x_{i_1, i_2, i_3} \\ y(\xi, \eta, \tau) = \sum_{i_1} \sum_{i_2} \sum_{i_3} L_{i_1, i_2, i_3}(\xi, \eta, \tau) y_{i_1, i_2, i_3} = \sum_{i_1} \sum_{i_2} \sum_{i_3} \phi_{i_1}(\xi) \phi_{i_2}(\eta) \phi_{i_3}(\tau) y_{i_1, i_2, i_3} \\ t(\tau) = \sum_{i_3} \phi_{i_3}(\tau) t_{i_3} \end{cases},$$

where $\phi_i(*)$, $*$ = ξ, η , or $\tau \in [-1, 1]$ are 1D Lagrange polynomials, and $L_{i_1, i_2, i_3}(\xi, \eta, \tau)$ is their tensor product.

Since the coordinates are fully represented by polynomials, the term $\partial(|J|\xi_x)/\partial \xi$ can be calculated analytically from

$$\frac{\partial |J|\xi_x}{\partial \xi} \Big|_{t_{\xi*} \equiv 0} t_\tau y_{\xi\eta} = \left(\sum_{i_3} \frac{d\phi_{i_3}(\tau)}{d\tau} t_{i_3} \right) \left(\sum_{i_1} \sum_{i_2} \sum_{i_3} \frac{d\phi_{i_1}(\xi)}{d\xi} \frac{d\phi_{i_2}(\eta)}{d\eta} \phi_{i_3}(\tau) y_{i_1, i_2, i_3} \right).$$

Following the same procedure, it can be shown that $\partial(|J|\eta_x)/\partial \eta$ is exactly $-\partial(|J|\xi_x)/\partial \xi$. This ensures that the second formula in Eq. (14) always holds numerically. $\square$

4. Discussions of Solution and Flux Approximation

In this section, we first introduce two popular ways to construct solution and flux polynomials for curvilinear elements in static domains, and discuss their numerical features. To facilitate discussion, only the spatial dimensions are considered. Based on that understanding, we then develop a new approach to approximate solutions and fluxes as polynomials over general curvilinear space-time elements.

4.1. Two popular solution and flux approximation approaches

4.1.1. Physical domain approach

Divide the spatial domain Ω into N_e non-overlapping elements Ω_e , $e = 1, \dots, N_e$, with the element boundary denoted as $\partial\Omega_e$. In many DG-based constructions [51], each component Q_i , $i = 1, \dots, N_v$, of the solution vector $\mathbf{Q}$ is approximated as a polynomial $Q_{h,i}(\mathbf{x}, t)$, which sits in the degree k polynomial space $P^k(\Omega_e)$ based on the physical element Ω_e . Let $w_j(\mathbf{x})$, $j = 1, \dots, N(k; \Omega_e)$, denote the orthogonal bases of the polynomial space. We have

$$Q_{h,i}(\mathbf{x}, t) = \sum_j^{N(k)} q_{h,i,j}(t) w_j(\mathbf{x}),$$

where $N(k)$ depends on the element type and the dimension of the polynomial space. For example, for a tetrahedral element, $N(k) = (k+1)(k+2)(k+3)/6$; and for a d -dimensional tensor-product element, $N(k) = (k+1)^d$. The strong form of Eq. (1) for each spatial element Ω_e bounded by $\partial\Omega_e$ finally becomes

$$\int_{\Omega_e} \frac{\partial \mathbf{Q}_h}{\partial t} \otimes \mathbf{w} dV + \int_{\Omega_e} \nabla \cdot \mathbf{F}^{\text{loc}} \otimes \mathbf{w} dV + \int_{\partial\Omega_e} (\mathbf{F}_n^{\text{com}} - \mathbf{F}_n^{\text{loc}}) \otimes \mathbf{w} dS = 0. \quad (17)$$

Herein, $\mathbf{Q}_h$ is the solution vector consisting of all $Q_{h,i}(\mathbf{x}, t)$, $\mathbf{w}$ is a vector consisting of all bases $w_j(\mathbf{x})$, $\mathbf{F}_n^{\text{loc}}$ and $\mathbf{F}_n^{\text{com}}$ are the local and common normal fluxes with $\mathbf{F}_n = \mathbf{n} \cdot \mathbf{F}$, where $\mathbf{n}$ is the outward-going unit normal vector of the physical element boundary $\partial\Omega_e$, and $\otimes$ is the tensor product operator. Numerical quadrature can then be performed to calculate each component of the mass matrix, and to carry out the volumetric and surface integration for flux-related terms. Over-integration can be used to enhance the integration accuracy by reducing aliasing errors from the flux-related terms.

To facilitate implementation, Q_i can be approximated as a polynomial $Q_{h,i}^r(\boldsymbol{\xi}, t) \in P^k(\Omega^r)$, where Ω^r is the standard element in the reference domain, and $\boldsymbol{\xi} = (\xi_1, \dots, \xi_d)$, e.g. $\boldsymbol{\xi} = (\xi, \eta)$ for the two-dimensional governing equations. Since the mapping between the reference and physical domains is one-to-one within any physical element Ω_e (see Figure 1), the inverse map exists and $\boldsymbol{\xi} = \boldsymbol{\xi}(\mathbf{x})$. Analogous to the nomenclature used for the physical element, we can define the solution vector $\mathbf{Q}_h^r(\boldsymbol{\xi})$ for the reference domain, and the vector $\mathbf{w}^r(\boldsymbol{\xi})$ consisting of all bases of the polynomial space $P^k(\Omega^r)$. Thus, the integration of Eq. (1) in the physical element Ω_e can be calculated from the reference element Ω^r as

$$\begin{aligned} & \int_{\Omega_e} \left(\frac{\partial \mathbf{Q}_h^r(\boldsymbol{\xi}(\mathbf{x}), t)}{\partial t} + \nabla \cdot \mathbf{F}^r(\boldsymbol{\xi}(\mathbf{x}), t) \right) \otimes \mathbf{w}^r(\boldsymbol{\xi}(\mathbf{x})) dV = \\ & \int_{\Omega^r} \left(\frac{\partial \mathbf{Q}_h^r(\boldsymbol{\xi}, t)}{\partial t} + \sum_{i=1}^d \nabla \xi_i \cdot \frac{\partial \mathbf{F}^r(\boldsymbol{\xi}, t)}{\partial \xi_i} \right) \otimes \mathbf{w}^r(\boldsymbol{\xi}) |J|(\boldsymbol{\xi}; \Omega_e) dV^r. \end{aligned} \quad (18)$$

Note that although the original integration concept (i.e. the left-hand side of the equation) is based on the physical domain, the implementation (i.e. the right-hand side of the equation), including the approximation of solutions and fluxes, Jacobian and metric calculation, and numerical integration, is based on the reference domain. Therefore, this approach can be termed as a hybrid reference-physics domain approach, which will be further discussed and developed in following sections.

Remark 4.1. In the analytical governing equations, the value of $\mathbf{Q}^r(\boldsymbol{\xi}, t)$ is equal to that of $\mathbf{Q}(\mathbf{x}, t)$ at the point pair $\boldsymbol{\xi}$ and $\mathbf{x}$. However, the polynomial approximations $\mathbf{Q}_h(\mathbf{x}, t)$ in the physical domain do not equal to the polynomial approximations $\mathbf{Q}_h^r(\boldsymbol{\xi}, t)$ in the reference domain at the point pair $\boldsymbol{\xi}$ and $\mathbf{x}$ when the transformation between $\mathbf{x}$ and $\boldsymbol{\xi}$ is nonlinear.

Remark 4.2. Over-integration is usually needed to enhance the integration accuracy of the flux-related terms in Eq. (18) for curvilinear elements.

4.1.2. Reference domain approach

Without loss of generality, we use the 2D governing equations to explain solution and flux approximations here. When the method of lines is used to solve Eq. (3) in a stationary domain, the solution and flux vectors

take the following forms in the reference domain:

$$\begin{cases} \tilde{\mathbf{Q}} = |J|\mathbf{Q} \\ \tilde{\mathbf{F}} = |J|\xi_x\mathbf{F} + |J|\xi_y\mathbf{G} \\ \tilde{\mathbf{G}} = |J|\eta_x\mathbf{F} + |J|\eta_y\mathbf{G} \end{cases}, \quad (19)$$

with $\tau \equiv t$, and the governing equations read like Eq. (4) but with different definitions of $\tilde{\mathbf{Q}}$, $\tilde{\mathbf{F}}$, and $\tilde{\mathbf{G}}$.

In the reference domain approach, each component $\tilde{Q}_i$ of the solution vector $\tilde{\mathbf{Q}}$ is approximated as a polynomial $\tilde{Q}_{h,i}(\boldsymbol{\xi}, t)$, which sits in the degree k polynomial space $P^k(\Omega^r)$ based on the standard element Ω^r . Let $w_j^r(\boldsymbol{\xi})$, $j = 1, \dots, N(k; \Omega^r)$, denote the orthogonal bases of the polynomial space $P^k(\Omega^r)$. As a result, $\tilde{Q}_{h,i}(\boldsymbol{\xi}, t)$ can be expressed as

$$\tilde{Q}_{h,i}(\boldsymbol{\xi}, t) = \sum_j^{N(k)} \tilde{q}_{h,i,j}(t) w_j^r(\boldsymbol{\xi}).$$

Let $\tilde{\mathbf{Q}}_h$ be the solution vector consisting of all $\tilde{Q}_{h,i}(\boldsymbol{\xi}, t)$, and $\mathbf{w}^r$ be the vector consisting of all bases $w_j^r(\boldsymbol{\xi})$. The DG scheme in the strong form for the governing equation on a stationary grid in the reference domain Ω^r bounded by $\partial\Omega^r$ can then be written as

$$\int_{\Omega^r} \frac{\partial \tilde{\mathbf{Q}}_h}{\partial t} \otimes \mathbf{w}^r dV^r + \int_{\Omega^r} \nabla^r \cdot \tilde{\mathbf{F}}^{\text{loc}} \otimes \mathbf{w}^r dV^r + \int_{\partial\Omega^r} \left(\tilde{\mathbf{F}}_{n^r}^{\text{num}} - \tilde{\mathbf{F}}_{n^r}^{\text{loc}} \right) \otimes \mathbf{w}^r dS^r = 0. \quad (20)$$

Herein, ∇^r is the gradient operator in the reference domain, $\tilde{\mathbf{F}} = \begin{pmatrix} \tilde{\mathbf{F}} \\ \tilde{\mathbf{G}} \end{pmatrix}$ in 2D, and $\tilde{\mathbf{F}}_{n^r} = \mathbf{n}^r \cdot \tilde{\mathbf{F}}$, where n^r is the outward-going unit normal vector of the standard element Ω^r .

The accuracy of the reference domain approach can suffer on curvilinear elements due to the associated geometric nonlinearity. Moreover, the freestream flow may not be preserved by the reference domain approach due to the violation of the GCL (14) in its discrete form. To restore the accuracy of the reference domain approach for curvilinear elements, several conservative metric construction methods have been developed; see [24, 47]. If the non-conservative metric calculation (see Eq. (16) as an example) is used, the coordinate transformation concept presented in Eq. (18) can be used to enhance the accuracy of the implementation of Eq. (20) on curvilinear elements. With the correction procedure via reconstruction (CPR) concept [39], Eq. (18) can be recast into the differential format for any physical element Ω_e as

$$\frac{\partial \mathbf{Q}_h^r(\boldsymbol{\xi}, t)}{\partial t} + \nabla \cdot \mathbf{F}^{r, \text{loc}}(\boldsymbol{\xi}, t) + \boldsymbol{\delta}^{r, \text{cor}}(\boldsymbol{\xi}, t) = 0, \quad (21)$$

with

$$\int_{\Omega^r} \boldsymbol{\delta}^{r, \text{cor}}(\boldsymbol{\xi}, t) \otimes \mathbf{w}^r(\boldsymbol{\xi}) |J|(\boldsymbol{\xi}; \Omega_e) dV^r = \int_{\partial\Omega^r} \left(\tilde{\mathbf{F}}_{n^r}^{\text{num}}(\boldsymbol{\xi}, t) - \tilde{\mathbf{F}}_{n^r}^{\text{loc}}(\boldsymbol{\xi}, t) \right) \otimes \mathbf{w}^r(\boldsymbol{\xi}) dS^r.$$

Herein, $|J|(\boldsymbol{\xi}; \Omega_e) \boldsymbol{\delta}^{r, \text{cor}}$ becomes the spatial correction field in the FR formula (11) in a stationary domain.

Note that the numerical implementation of Eq. (21) can preserve the freestream flow on curvilinear meshes [23], but does not necessarily hold the discrete GCL. As a result, although Eq. (20) and Eq. (21) are analytically identical to each other, they are numerically different.

4.2. A new perspective

As discussed in Sect. 4.1.2, when $\tilde{\mathbf{Q}}$ is approximated as a polynomial, the accuracy of the solution $\mathbf{Q}$ can suffer due to the geometric nonlinearity of the curvilinear elements. The fix via CPR through Eq. (21) does not necessarily hold discrete GCL, which is not desirable. Here we introduce a new perspective to approximate $\tilde{\mathbf{Q}}$ as a polynomial by approximating $\mathbf{Q}$ and metrics as polynomials separately.

4.2.1. Stationary curvilinear elements

Similar to **Sect. 4.1.2**, only 2D governing equations on stationary grids and solved with the method of lines are used to explain the concept in this subsection. Moreover, the tensor product polynomial space $\mathbb{Q}(\Omega^r)$ defined on the standard element Ω^r will be considered in this study. Thus, we can explicitly write $\mathbb{Q}^k(\Omega^r)$ with the polynomial degree at most k in each variable defined on the 2D standard quadrilateral element Ω^r as $\mathbb{Q}^k(\xi, \eta) = P^k(\xi) \otimes P^k(\eta)$.

Let $Q_{h,i}(\xi; t) \in \mathbb{Q}^k(\xi, \eta)$ at a fixed time t be the polynomial approximation of each component Q_i of the solution vector $\mathbf{Q}$. For curvilinear elements, the physical domain coordinates (x, y) can be represented as polynomials, i.e. $x, y \in \mathbb{Q}^l(\xi, \eta)$. For 2D coordinate transformation between the stationary physical domain (x, y) and the reference domain (ξ, η) , we summarize the polynomial space where the Jacobian and metrics sit in Table 1. Therefore, $\tilde{Q}_{h,i}(\xi) = |J|Q_{h,i} \in \mathbb{Q}^{k+2l-1}(\xi, \eta)$.

Remark 4.3. Note that the coefficients of the polynomials used to represent the Jacobian and metrics are known (see **Sect. 3.2**). Therefore, in the degree $k + 2l - 1$ polynomial approximation of $\tilde{Q}_{h,i}$, only $k + 1$ coefficients in each dimension are independent.

	$ J $	$ J \xi_x, J \xi_y$	$ J \eta_x, J \eta_y$
PS	$\mathbb{Q}^{2l-1}(\xi, \eta)$	$P^l(\xi) \otimes P^{l-1}(\eta)$	$P^{l-1}(\xi) \otimes P^l(\eta)$

Table 1: A list of polynomial space occupied by the Jacobian and metrics of a stationary curvilinear element with $x, y \in \mathbb{Q}^l(\xi, \eta)$. ‘PS’ stands for polynomial space. Note that $l \geq 1$. When $l = 1$, there exists a special case when the element is a proper rectangle. Therein, the Jacobian J and all metrics are constant.

Based on Eq. (19), to match the polynomial degree of $\tilde{Q}_{h,i}$ in the differential form of the governing equation, each component of the flux vector $\tilde{\mathbf{F}}$ in the reference domain can be approximated as a polynomial sitting in the space $P^{k+2l}(\xi) \otimes P^{k+2l-1}(\eta)$, and each component of $\tilde{\mathbf{G}}$ can be approximated as a polynomial sitting in the space $P^{k+2l-1}(\xi) \otimes P^{k+2l}(\eta)$. Thus, we have $\partial \tilde{F}_i / \partial \xi \in \mathbb{Q}^{k+2l-1}(\xi, \eta)$, and $\partial \tilde{G}_i / \partial \eta \in \mathbb{Q}^{k+2l-1}(\xi, \eta)$, where $\tilde{F}_i$ and $\tilde{G}_i$ are the i -th component of $\tilde{\mathbf{F}}$ and $\tilde{\mathbf{G}}$, respectively. Following the FR formulation (8), we can split $\tilde{\mathbf{F}}$ and $\tilde{\mathbf{G}}$ into local fluxes $\tilde{\mathbf{F}}^{\text{loc}}(\xi, \eta)$, $\tilde{\mathbf{G}}^{\text{loc}}(\xi, \eta)$, and correction fluxes $\tilde{\mathbf{F}}^{\text{cor}}(\xi, \eta)$, $\tilde{\mathbf{G}}^{\text{cor}}(\xi, \eta)$. Similar to Eq. (9), the correction fluxes can be approximated with high-order correction functions as

$$\begin{cases} \tilde{\mathbf{F}}^{\text{cor}}(\xi, \eta) = \left(\tilde{\mathbf{F}}_L^{\text{num}}(\eta) - \tilde{\mathbf{F}}^{\text{loc}}(-1, \eta) \right) g_L(\xi) + \left(\tilde{\mathbf{F}}_R^{\text{num}}(\eta) - \tilde{\mathbf{F}}^{\text{loc}}(1, \eta) \right) g_R(\xi), \\ \tilde{\mathbf{G}}^{\text{cor}}(\xi, \eta) = \left(\tilde{\mathbf{G}}_L^{\text{num}}(\xi) - \tilde{\mathbf{G}}^{\text{loc}}(\xi, -1) \right) g_L(\eta) + \left(\tilde{\mathbf{G}}_R^{\text{num}}(\xi) - \tilde{\mathbf{G}}^{\text{loc}}(\xi, 1) \right) g_R(\eta). \end{cases} \quad (22)$$

Herein, $g_{L/R}(\xi) \in P^{k+2l}(\xi)$, and $g_{L/R}(\eta) \in P^{k+2l}(\eta)$.

Now we examine polynomial space that can be used to approximate local fluxes $\tilde{\mathbf{F}}^{\text{loc}}(\xi, \eta)$ and $\tilde{\mathbf{G}}^{\text{loc}}(\xi, \eta)$. Note that the piece-wise continuous total fluxes $\tilde{\mathbf{F}}$ and $\tilde{\mathbf{G}}$ sit in the same polynomial space as that of $\tilde{\mathbf{F}}^{\text{cor}}$ and $\tilde{\mathbf{G}}^{\text{cor}}$. According to the structure of Eq. (22), $\tilde{F}_i^{\text{loc}}(*, \eta) \in P^{k+2l-1}(\eta)$ along the η direction, and $\tilde{G}_i^{\text{loc}}(\xi, *) \in P^{k+2l-1}(\xi)$ along the ξ direction, where $\tilde{F}_i^{\text{loc}}$ and $\tilde{G}_i^{\text{loc}}$ are the i -th component of $\tilde{\mathbf{F}}^{\text{loc}}$ and $\tilde{\mathbf{G}}^{\text{loc}}$, respectively. This gives flexibility to approximate $\tilde{\mathbf{F}}^{\text{loc}}$ along the ξ direction, and $\tilde{\mathbf{G}}^{\text{loc}}$ along the η direction. We study this issue by checking the DG formulation (20) in the integral form.

Let $\mathbf{w}^r = (w_1^r, \dots, w_f^r, \dots, w_{(k+2l) \times (k+2l)}^r)$, and $w_j^r(\xi, \eta) \in \mathbb{Q}^{k+2l-1}(\xi, \eta)$, and $\nabla^r = (\partial/\partial\xi, \partial/\partial\eta)$ in 2D. When the Gauss-Legendre quadrature rule with $k + 2l$ quadrature points in each dimension is used to carry out integration, it can integrate the degree $2(k + 2l) - 1$ polynomial exactly. Therefore, to ensure that all integrals in Eq. (20) can be integrated exactly, one choice is to approximate $\tilde{\mathbf{F}}^{\text{loc}} = (\tilde{\mathbf{F}}^{\text{loc}}, \tilde{\mathbf{G}}^{\text{loc}})$ as $\tilde{F}_i^{\text{loc}} \in P^{k+2l}(\xi) \otimes P^{k+2l-1}(\eta)$, and $\tilde{G}_i^{\text{loc}} \in P^{k+2l-1}(\xi) \otimes P^{k+2l}(\eta)$. Note that this indicates that $F_i^{\text{loc}}, G_i^{\text{loc}} \in \mathbb{Q}^{k+l}(\xi, \eta)$. We can further define the correction field $\tilde{\delta}^{\text{cor}} = |J|\delta^{\text{cor}}$ with each of its component $\tilde{\delta}_i^{\text{cor}} \in \mathbb{Q}^{k+2l-1}(\xi, \eta)$ or equivalently $\delta_i^{\text{cor}} \in \mathbb{Q}^k(\xi, \eta)$ as

$$\int_{\Omega^r} \tilde{\delta}^{\text{cor}} \otimes \mathbf{w}^r dV^r = \int_{\partial\Omega^r} \left(\tilde{\mathbf{F}}_{n^r}^{\text{num}} - \tilde{\mathbf{F}}_{n^r}^{\text{loc}} \right) \otimes \mathbf{w}^r dS^r. \quad (23)$$

According to [52], $\tilde{\delta}^{\text{cor}}$ defined in Eq. (23) is equivalent to the following definition with $\tilde{\mathbf{F}}^{\text{cor}}$ and $\tilde{\mathbf{G}}^{\text{cor}}$ in 2D:

$$\tilde{\delta}^{\text{cor}} = \frac{\partial \tilde{\mathbf{F}}^{\text{cor}}}{\partial \xi} + \frac{\partial \tilde{\mathbf{G}}^{\text{cor}}}{\partial \eta}. \quad (24)$$

As a result, we can exactly convert the DG formulation (20) in the integral form into the FR formulation in the differential form numerically:

$$\frac{\partial \tilde{\mathbf{Q}}_h}{\partial t} + \frac{\partial \tilde{\mathbf{F}}^{\text{loc}}}{\partial \xi} + \frac{\partial \tilde{\mathbf{G}}^{\text{loc}}}{\partial \eta} + \tilde{\delta}^{\text{cor}} = 0. \quad (25)$$

Based on the aforementioned analyses, we summarize the polynomial degrees of solutions and fluxes used in the above analysis in Table 2.

	$\tilde{\mathbf{Q}}$	$\mathbf{Q}$	$\mathbf{F}^{\text{loc}}$	$\mathbf{G}^{\text{loc}}$
PS	$\mathbb{Q}^{k+2l-1}(\xi, \eta)$	$\mathbb{Q}^k(\xi, \eta)$	$\mathbb{Q}^{k+l}(\xi, \eta)$	$\mathbb{Q}^{k+l}(\xi, \eta)$
	$\tilde{\delta}^{\text{cor}}$	δ^{cor}	$\tilde{\mathbf{F}}^{\text{cor}}, \tilde{\mathbf{F}}^{\text{loc}}$	$\tilde{\mathbf{G}}^{\text{cor}}, \tilde{\mathbf{G}}^{\text{loc}}$
PS	$\mathbb{Q}^{k+2l-1}(\xi, \eta)$	$\mathbb{Q}^k(\xi, \eta)$	$P^{k+2l}(\xi) \otimes P^{k+2l-1}(\eta)$	$P^{k+2l-1}(\xi) \otimes P^{k+2l}(\eta)$

Table 2: A list of polynomial space where the FR solution, flux, and correction field variables in the physical domain and reference domain sit at any specific time.

Theorem 2. *The FR scheme (25) with solution, flux and correction field polynomial degrees following those listed in Table 2 numerically satisfies the following GCL (i.e. the GCL (14) in the stationary domain)*

$$\frac{\partial}{\partial \xi} \begin{pmatrix} |J|\xi_x \\ |J|\xi_y \end{pmatrix} + \frac{\partial}{\partial \eta} \begin{pmatrix} |J|\eta_x \\ |J|\eta_y \end{pmatrix} = 0, \quad (26)$$

no matter the metrics are calculated with the non-conservative or conservative approaches [24].

Proof. Consider the freestream with constant $\mathbf{Q}$. The fluxes $\mathbf{F}$ and $\mathbf{G}$ are also constant. Note that in a stationary domain, the Jacobian $|J|$ is not a function of time. Therefore, when $\mathbf{Q}$ is constant, the term $\partial \tilde{\mathbf{Q}}_h / \partial t$ in Eq. (25) is zero. In numerical discretization, values that are analytically zero can only be represented approximately, appearing as machine zeros due to finite precision. However, these values are typically several tens of orders of magnitude smaller than non-trivial numerical errors, such as those arising from GCL violations (see examples in Ref. [6]). Therefore, machine zeros are neglected in the following discussions to facilitate a cleaner and more transparent proof. We then have from Eq. (25) that

$$\begin{aligned} & \mathbf{F}^{\text{loc}} \left(\sum_j (|J|\xi_x)_j^{\text{SP}} \frac{\partial w_j^r(\xi)}{\partial \xi} + \sum_j (|J|\eta_x)_j^{\text{SP}} \frac{\partial w_j^r(\xi)}{\partial \eta} \right) \\ & + \mathbf{G}^{\text{loc}} \left(\sum_j (|J|\xi_y)_j^{\text{SP}} \frac{\partial w_j^r(\xi)}{\partial \xi} + \sum_j (|J|\eta_y)_j^{\text{SP}} \frac{\partial w_j^r(\xi)}{\partial \eta} \right) + \tilde{\delta}^{\text{cor}} = 0, \end{aligned} \quad (27)$$

with $w_j^r(\xi) = w_j^r(\xi, \eta) \in \mathbb{Q}^{k+2l-1}(\xi, \eta)$, which is the basis function constructed via tensor product of 1D Lagrange polynomials rooted on solution points, and

$$\tilde{\mathbf{F}}^{\text{loc}}(\xi) = \sum_j (\tilde{\mathbf{F}}^{\text{loc}})_j^{\text{SP}} w_j^r(\xi), \quad \tilde{\mathbf{G}}^{\text{loc}}(\xi) = \sum_j (\tilde{\mathbf{G}}^{\text{loc}})_j^{\text{SP}} w_j^r(\xi).$$

The superscript ‘SP’ in Eq. (27) indicates that the function is calculated at solution points.

We first prove that $\tilde{\delta}^{\text{cor}} = 0$. Since $\mathbf{F}$ and $\mathbf{G}$ are constant, we have $\mathbf{F}_n^{\text{com}} = \mathbf{F}_n^{\text{loc}}$. Without loss of

generality, check the term $\tilde{\mathbf{F}}_L^{\text{num}}(\eta) - \tilde{\mathbf{F}}^{\text{loc}}(-1, \eta)$ in $\tilde{\mathbf{F}}^{\text{cor}}(\xi, \eta)$ from Eq. (22). It can be written as

$$\begin{aligned} \tilde{\mathbf{F}}_L^{\text{num}}(\eta) - \tilde{\mathbf{F}}^{\text{loc}}(-1, \eta) &= \mathbf{F}^{\text{loc}}(-1, \eta) [(|J|\xi_x)^{\text{FP}}(-1, \eta) - (|J|\xi_x)^{\text{SP,I}}(-1, \eta)] \\ &\quad + \mathbf{G}^{\text{loc}}(-1, \eta) [(|J|\xi_y)^{\text{FP}}(-1, \eta) - (|J|\xi_y)^{\text{SP,I}}(-1, \eta)]. \end{aligned}$$

Herein, the superscript ‘FP’ indicates that the function is calculated at flux points, and ‘SP,I’ indicates that the function is first calculated at solution points and then interpolated from solution points to the corresponding flux points.

Based on Table 1, we have $|J|\xi_x, |J|\xi_y \in P^l(\xi) \otimes P^{l-1}(\eta)$. To construct the FR scheme (25), the polynomial space $\mathbb{Q}^{k+2l-1}(\xi, \eta)$ with $k+2l$ Gauss-Legendre quadrature points (i.e. solution points in FR) in each dimension is used. Therefore, $|J|\xi_x$ and $|J|\xi_y$ can be exactly evaluated at any point within the curvilinear element when they are constructed as polynomials analytically. As a result, we have $(|J|\xi_{x/y})^{\text{FP}} = (|J|\xi_{x/y})^{\text{SP,I}}$, and $\tilde{\mathbf{F}}_L^{\text{num}}(\eta) - \tilde{\mathbf{F}}^{\text{loc}}(-1, \eta) = 0$. Furthermore, we have $\tilde{\mathbf{F}}^{\text{cor}} = 0$, $\tilde{\mathbf{G}}^{\text{cor}} = 0$, and thus, $\tilde{\boldsymbol{\delta}}^{\text{cor}} = 0$.

Now Eq. (27) is reduced to

$$\begin{aligned} &\mathbf{F}^{\text{loc}} \left(\sum_j (|J|\xi_x)_j^{\text{SP}} \frac{\partial w_j(\boldsymbol{\xi})}{\partial \xi} + \sum_j (|J|\eta_x)_j^{\text{SP}} \frac{\partial w_j(\boldsymbol{\xi})}{\partial \eta} \right) \\ &+ \mathbf{G}^{\text{loc}} \left(\sum_j (|J|\xi_y)_j^{\text{SP}} \frac{\partial w_j(\boldsymbol{\xi})}{\partial \xi} + \sum_j (|J|\eta_y)_j^{\text{SP}} \frac{\partial w_j(\boldsymbol{\xi})}{\partial \eta} \right) = 0. \end{aligned}$$

Since $|J|\xi_x, |J|\xi_y \in P^l(\xi) \otimes P^{l-1}(\eta)$ and $|J|\eta_x, |J|\eta_y \in P^{l-1}(\xi) \otimes P^l(\eta)$, they can be fully represented with $w_j^r(\boldsymbol{\xi})$. Therefore, taking $|J|\xi_x$ as an example, we have

$$(|J|\xi_x)(\boldsymbol{\xi}) = \sum_j (|J|\xi_x)_j^{\text{SP}} w_j^r(\boldsymbol{\xi}), \quad (28)$$

and

$$\frac{\partial (|J|\xi_x)(\boldsymbol{\xi})}{\partial \xi} = \sum_j (|J|\xi_x)_j^{\text{SP}} \frac{\partial w_j^r(\boldsymbol{\xi})}{\partial \xi}. \quad (29)$$

Note that $(|J|\xi_x)_j^{\text{SP}}$ used in Eq. (28) and (29) can be calculated either with the non-conservative approach (see Eq. (16)) or with the conservative approach (see Ref. [24]). Similar results apply to $|J|\xi_y$, $|J|\eta_x$, and $|J|\eta_y$.

On plugging Eq. (28) and (29) in Eq. (27), we have

$$\mathbf{F}^{\text{loc}} \left(\frac{\partial |J|\xi_x}{\partial \xi} + \frac{\partial |J|\eta_x}{\partial \eta} \right) + \mathbf{G}^{\text{loc}} \left(\frac{\partial |J|\xi_y}{\partial \xi} + \frac{\partial |J|\eta_y}{\partial \eta} \right) = 0. \quad (30)$$

Since the geometric features of the curvilinear element are fully represented by polynomials in the FR scheme, based on **Lemma 1**, we have $\partial |J|\xi_x / \partial \xi + \partial |J|\eta_x / \partial \eta = 0$, and $\partial |J|\xi_y / \partial \xi + \partial |J|\eta_y / \partial \eta = 0$.

This proves that for freestream the GCL (26) held numerically. $\square$

With **Theorem 2**, the FR scheme (25) can be numerically converted back to its physical domain format as

$$\frac{\partial \mathbf{Q}_h}{\partial t} + \frac{\partial \mathbf{F}^{\text{loc}}}{\partial x} + \frac{\partial \mathbf{G}^{\text{loc}}}{\partial y} + \boldsymbol{\delta}^{\text{cor}} = 0. \quad (31)$$

Key procedures to implement Eq. (31) are summarized in **Key Procedures 1**. Note that the FR scheme (31) is different from Eq. (21) discussed in **Sect. 4.1.2** as in Eq. (31) variables describing flow physics and those describing geometry are approximated by separate polynomials with $\tilde{\mathbf{Q}}$ served as the hidden working variable.

Key Procedures 1 A Hybrid Reference-Physics Domain Approach

- 1: The terms $\partial \mathbf{F}^{\text{loc}}/\partial x$ and $\partial \mathbf{G}^{\text{loc}}/\partial y$ on solution points in each dimension can be calculated from

$$\frac{\partial \mathbf{F}^{\text{loc}}}{\partial x} = \frac{\partial \mathbf{F}^{\text{loc}}}{\partial \xi} \xi_x + \frac{\partial \mathbf{F}^{\text{loc}}}{\partial \eta} \eta_x, \quad \frac{\partial \mathbf{G}^{\text{loc}}}{\partial y} = \frac{\partial \mathbf{G}^{\text{loc}}}{\partial \xi} \xi_y + \frac{\partial \mathbf{G}^{\text{loc}}}{\partial \eta} \eta_y.$$

▷ Although $\partial \xi_i/\partial x_j$ are not polynomials, $|J|\partial \xi_i/\partial x_j$ are polynomials sitting either in $P^l(\xi) \otimes P^{l-1}(\eta)$ or in $P^{l-1}(\xi) \otimes P^l(\eta)$. This numerically ensures that

$$|J| \left(\frac{\partial \mathbf{F}^{\text{loc}}}{\partial x} + \frac{\partial \mathbf{G}^{\text{loc}}}{\partial y} \right) = \frac{\partial \tilde{\mathbf{F}}^{\text{loc}}}{\partial \xi} + \frac{\partial \tilde{\mathbf{G}}^{\text{loc}}}{\partial \eta}$$

holds on solution points in each dimension.

- 2: Use the univariate correction functions $g_{L/R}(\cdot)$ for $\tilde{\mathbf{F}}^{\text{cor}}$ and $\tilde{\mathbf{G}}^{\text{cor}}$ in Eq. (22), and evaluate $\tilde{\delta}^{\text{cor}}$ from Eq. (24). Then δ^{cor} can be calculated from $\delta^{\text{cor}} = \tilde{\delta}^{\text{cor}}/|J|$ on the solution points in each dimension.
- ▷ Only the values of $g'_{L/R}(\cdot)$ are needed on 1D solution points. These values can be stored in pre-processing.
-

Before ending this section, we briefly compare the consistent symmetric conservative metric (CSC) construction approach in [24] and the new approach developed above. Both approaches have the global conservation and freestream preservation properties. The differences are summarized as follows:

- Different from the CSC approach, metrics in the new FR approach can be calculated with either the non-conservative formulation or the conservative one.
- The metrics in the CSC approach are reconstructed from their exact values to match the order of the accuracy of physical variables. However, the metrics in the new FR approach here are exact, and their polynomial order is independent from that of physical variables.

4.2.2. Space-time moving curvilinear elements

For space-time curvilinear elements over a moving domain, let $x, y \in \mathbb{Q}^l(\xi, \eta) \otimes P^n(\tau)$, $t \in P^n(\tau)$. Thus, the Jacobian $|J| = t_\tau(x_\xi y_\eta - x_\eta y_\xi) \in \mathbb{Q}^{2l-1}(\xi, \eta) \otimes P^{3n-1}(\tau)$. The polynomial space where the Jacobian and metrics sit for a tensor-product curvilinear space-time element are summarized in Table 3.

	$ J $	$ J \xi_x, J \xi_y$	$ J \eta_x, J \eta_y$
PS	$\mathbb{Q}^{2l-1}(\xi, \eta) \otimes P^{3n-1}(\tau)$	$P^l(\xi) \otimes P^{l-1}(\eta) \otimes P^{2n-1}(\tau)$	$P^{l-1}(\xi) \otimes P^l(\eta) \otimes P^{2n-1}(\tau)$
	$ J \tau_t$	$ J \xi_t$	$ J \eta_t$
PS	$\mathbb{Q}^{2l-1}(\xi, \eta) \otimes P^{2n}(\tau)$	$P^{2l}(\xi) \otimes P^{2l-1}(\eta) \otimes P^{2n-1}(\tau)$	$P^{2l-1}(\xi) \otimes P^{2l}(\eta) \otimes P^{2n-1}(\tau)$

Table 3: A list of polynomial space occupied by the Jacobian and metrics of a space-time moving curvilinear element with $x, y \in \mathbb{Q}^l(\xi, \eta) \otimes P^n(\tau)$ and $t \in P^n(\tau)$. Note that $l \geq 1$ and $n \geq 1$.

To numerically solve Eq. (4), a pseudo-time term $\partial |J| \mathbf{Q}_h / \partial \tilde{t}$ is augmented to the governing equation. As a result, the STFR scheme can be written as

$$\frac{\partial |J| \mathbf{Q}_h}{\partial \tilde{t}} + \frac{\partial \tilde{\mathbf{Q}}^{\text{loc}}}{\partial \tau} + \frac{\partial \tilde{\mathbf{F}}^{\text{loc}}}{\partial \xi} + \frac{\partial \tilde{\mathbf{G}}^{\text{loc}}}{\partial \eta} + \tilde{\delta}^{\text{cor, st}} = 0, \quad (32)$$

with

$$\tilde{\delta}^{\text{cor, st}} = \frac{\partial \tilde{\mathbf{Q}}^{\text{cor}}}{\partial \tau} + \frac{\partial \tilde{\mathbf{F}}^{\text{cor}}}{\partial \xi} + \frac{\partial \tilde{\mathbf{G}}^{\text{cor}}}{\partial \eta}. \quad (33)$$

Since t is only a function of τ , we have from Eq. (5) that $\tilde{\mathbf{Q}}^{\text{loc}} = |J| \tau_t \mathbf{Q}^{\text{loc}}$. Define $Q_{h,i}(\xi, \eta, \tau) \in \mathbb{Q}^k(\xi, \eta) \otimes P^m(\tau)$. Then following the procedures explained in Sect. 4.2.1, the STFR scheme (32) can be converted

back to its physical domain format as

$$\frac{\partial \mathbf{Q}_h}{\partial \tilde{t}} + \frac{\partial \mathbf{Q}^{\text{loc}}}{\partial t} + \frac{\partial \mathbf{F}^{\text{loc}}}{\partial x} + \frac{\partial \mathbf{G}^{\text{loc}}}{\partial y} + \boldsymbol{\delta}^{\text{cor,st}} = 0. \quad (34)$$

Similar to Table 2, the polynomial degrees of solutions, fluxes and correction field used in the implementation of Eq. (34) are summarized in Table 4.

Table 4: A list of polynomial space occupied by the Jacobian and metrics of a space-time moving curvilinear element with $x, y \in \mathbb{Q}^l(\xi, \eta) \otimes P^n(\tau)$ and $t \in P^n(\tau)$.

	$\mathbf{Q}_h$	$ J \mathbf{Q}_h$
PS	$\mathbb{Q}^k(\xi, \eta) \otimes P^m(\tau)$	$\mathbb{Q}^{k+2l-1}(\xi, \eta) \otimes P^{m+3n-1}(\tau)$
	$\boldsymbol{\delta}^{\text{cor,st}}$	$\tilde{\boldsymbol{\delta}}^{\text{cor,st}}$
PS	$\mathbb{Q}^k(\xi, \eta) \otimes P^m(\tau)$	$\mathbb{Q}^{k+2l-1}(\xi, \eta) \otimes P^{m+3n-1}(\tau)$
	$\mathbf{Q}^{\text{loc}}$	$\tilde{\mathbf{Q}}^{\text{loc}}, \tilde{\mathbf{Q}}^{\text{cor}}$
PS	$\mathbb{Q}^k(\xi, \eta) \otimes P^{m+n}(\tau)^*$	$\mathbb{Q}^{k+2l-1}(\xi, \eta) \otimes P^{m+3n}(\tau)$
	$\mathbf{F}^{\text{loc}}$	$\tilde{\mathbf{F}}^{\text{loc}}, \tilde{\mathbf{F}}^{\text{cor}}$
PS	$\mathbb{Q}^{k+l}(\xi, \eta) \otimes P^{m+n}(\tau)$	$P^{k+2l}(\xi) \otimes P^{k+2l-1}(\eta) \otimes P^{m+3n-1}(\tau)$
	$\mathbf{G}^{\text{loc}}$	$\tilde{\mathbf{G}}^{\text{loc}}, \tilde{\mathbf{G}}^{\text{cor}}$
PS	$\mathbb{Q}^{k+l}(\xi, \eta) \otimes P^{m+n}(\tau)$	$P^{k+2l-1}(\xi) \otimes P^{k+2l}(\eta) \otimes P^{m+3n-1}(\tau)$

* This is used to assist understanding the numerical effect of the temporal correction field.

Moreover, analogous to **Theorem 2**, we have the following conclusion:

Theorem 3. *The FR scheme (32) with solution, flux and correction field polynomial degrees following those listed in Table 4, and with a spatial order higher than two (i.e. $k \geq 1$), numerically satisfies the GCL (14), no matter the metrics are calculated with the non-conservative or conservative approaches [24]. Moreover, under the conditions listed above, Eq. (32) is numerically equivalent to Eq. (34).*

Proof. The proof is analogous to that for **Theorem 2** by adding the numerical representation of $\partial \tilde{\mathbf{Q}}^{\text{loc}} / \partial \tau$, and that of the temporal flux (i.e. solution) $\mathbf{Q}$ -related terms in $\tilde{\mathbf{F}}^{\text{loc}}$ and $\tilde{\mathbf{G}}^{\text{loc}}$. Again, consider the freestream with constant $\mathbf{Q}$. Thus, Eq. (27) is augmented with the following term

$$\mathbf{Q}^{\text{loc}} \left(\sum_j (|J|\tau_t)_j^{\text{SP}} \frac{\partial w_j^r(\xi, \tau)}{\partial \tau} + \sum_j (|J|\xi_t)_j^{\text{SP}} \frac{\partial w_j^r(\xi, \tau)}{\partial \xi} + \sum_j (|J|\eta_t)_j^{\text{SP}} \frac{\partial w_j^r(\xi, \tau)}{\partial \eta} \right). \quad (35)$$

Different from the GCL (26) used in **Theorem 2**, the first formula in the GCL (14) needs the following two equations

$$k + 2l - 1 \geq 2l, \text{ and } m + 3n - 1 \geq 2n \quad (36)$$

to hold to ensure that $|J|\tau_t$, $|J|\xi_t$, and $|J|\eta_t$ can be exactly evaluated at any point within the curvilinear space-time element when they are constructed as polynomials analytically. Since $n \geq 1$ for any space-time element (i.e. τ is at least a linear function of t), we have $n = 3n - 2n \geq 1 \geq 1 - m$, when $m \geq 0$. This means that the condition $m + 3n - 1 \geq 2n$ always holds. Therefore, the only condition that needs to meet is $k \geq 1$. $\square$

Compared to the stationary grid case, special attention is needed when evaluating $\mathbf{Q}^{\text{loc}}$ and $\tilde{\mathbf{Q}}^{\text{cor}}$ from $\mathbf{Q}_h$. Note that there are $m + 3n$ solution points along the time dimension in the present space-time framework. As shown in Table 4, to ensure that the polynomial approximations of $\tilde{\mathbf{Q}}^{\text{loc}}$ and $\tilde{\mathbf{Q}}^{\text{cor}}$ are consistent with those of $\tilde{\mathbf{F}}^{\text{loc/cor}}$ and $\tilde{\mathbf{G}}^{\text{loc/cor}}$ to ease analysis and implementation, $\mathbf{Q}^{\text{loc}}$ can always be treated as degree $m + n$ polynomials along the time dimension. Thus, the degree P^{m+3n} correction function $g_{L/R}(\tau)$ is used

for $\tilde{Q}^{\text{cor}}$ to completely eliminate the aliasing errors caused by the Jacobian $|J|$. As a result, we have the following observation regarding Q^{loc} and Q_h .

Remark 4.4. *Since $g_{L/R}(\tau) \in P^{m+3n}(\tau)$, this indicates that $Q^{\text{loc}} \in \mathbb{Q}^k(\xi, \eta) \otimes P^{m+n}(\tau)$ essentially. This high-order information has been propagated to Q_h through the correction field to enhance its accuracy.*

4.2.3. Discussions and aliasing error control

According to Table 4, the STFR formulation for moving curvilinear elements can suffer from the so-called “curse of dimensionality”, especially in the time dimension. There are two ways to fix this issue. As the first one, the CSC approach in [24] may provide a solution by matching the polynomial degrees of symmetric conservative metrics to that of physical variables. As the other, we can find solutions by re-examining Theorem 2 and 3.

Note that when considering the GCL (14), the solution Q , and fluxes F and G are treated as constants. For moving grid simulations, based on Table 3, the most demanding GCL formulation is

$$\frac{\partial |J| \tau_t}{\partial \tau} + \frac{\partial |J| \xi_t}{\partial \xi} + \frac{\partial |J| \eta_t}{\partial \eta} = 0.$$

We can use the basis function $w_j^r(\xi, \eta, \tau) \in \mathbb{Q}^{2l}(\xi, \eta) \otimes P^{2n}(\tau)$, with $x, y \in \mathbb{Q}^l(\xi, \eta) \otimes P^n(\tau)$ and $t \in P^n(\tau)$, in the discrete GCL when constructing metrics. Since $|J| \tau_t$, $|J| \xi_t$, and $|J| \eta_t$ are at most of degree $2l$ in ξ and η directions, and at most of degree $2n$ in the τ direction, they can be exactly evaluated at any point within the curvilinear space-time element when they are constructed as polynomials analytically. As a result, derivations related to the discrete GCL in Theorem 2 and 3 hold. This gives the following conclusion:

Proposition 4. *To numerically satisfy the GCL (14) by using exact metrics of curvilinear moving grids with a nominal solution approximation $Q_{h,i}(\xi, \eta, \tau) \in \mathbb{Q}^k(\xi, \eta) \otimes P^m(\tau)$, the basis function $w_j^r(\xi, \eta, \tau)$ should satisfy the following requirement:*

$$w_j^r(\xi, \eta, \tau) \in \mathbb{Q}^\alpha(\xi, \eta) \otimes P^\beta(\tau), \text{ where } \alpha = \max(2l, k) \text{ and } \beta = \max(2n, m). \quad (37)$$

This condition should not be confused with Eq. (36) used in Theorem 3. Note that $\tilde{Q} = |J|Q$ is used as the hidden working variable when implementing Eq. (31) or (34) strictly following the polynomial degree requirements listed in Table 2 or 4. Therein, $\alpha = k + 2l - 1 \geq \max(2l, k)$ when $k \geq 1$, and $\beta = m + 3n - 1 \geq \max(2n, m)$, considering the fact that $l \geq 1$ and $n \geq 1$. We also mention that Proposition 4 gives concrete numerical conditions that satisfy Lemma 1.

Note that the requirement (37) is similar to those required to satisfy freestream preservation conditions for curvilinear stationary grids in the work by Abe *et al.* [24]. However, we have the following conclusion regarding the requirement (37).

Remark 4.5. *Although satisfying the requirement (37) ensures that the GCL (14) is held numerically (or the discrete GCL holds), it cannot guarantee that Eq. (32) is numerically equivalent to Eq. (34). We emphasize that only when the conditions in Theorem 3 are satisfied, Eq. (32) is numerically equivalent to Eq. (34).*

Recall the discussions in Sect. 4.1.2, explicitly using $\tilde{Q}$ as the working variable may cause accuracy deterioration of the solution Q . Therefore, it is desirable to implement Eq. (34) with respect to the reference domain (see Key Procedures 1 in Sect. 4.2.1). Since it is very challenging, if not impossible, to exactly measure how the geometric nonlinearity affects the solution accuracy, we develop projection-based polynomial filtering procedures to control the aliasing errors caused by the nonlinear interaction between physical quantities of the flow and the curvilinear geometric representation of space-time elements.

Assume that any solution variable $Q^H(\xi) = Q^H(\xi, \eta, \tau)$ from the solution vector Q is approximated by the polynomial space $\mathbb{Q}^{k^H}(\xi, \eta) \otimes P^{m^H}(\tau)$, and the basis functions are the multidimensional Lagrange polynomials $L_{i_1, i_2, i_3}^H(\xi) = \phi_{i_1}^H(\xi) \phi_{i_2}^H(\eta) \phi_{i_3}^H(\tau)$, $i_1, i_2 = 1, \dots, k^H + 1$ and $i_3 = 1, \dots, m^H + 1$, consisting of 1D Lagrange polynomials $\phi_i^H(*)$, where ‘ $*$ ’ stands for $\xi \in [-1, 1]$, $\eta \in [-1, 1]$, or $\tau \in [-1, 1]$. $\phi_i^H(*)$ ’s are constructed based on the $k^H + 1$ Gauss–Legendre quadrature points (i.e. solution points in FR), denoted

as ξ_j^H or η_j^H , in each spatial dimension, or on the $m^H + 1$ Gauss–Legendre quadrature points, denoted as τ_j^H , in the time dimension. The corresponding quadrature weights associated with spatial quadrature points are denoted as $\omega_{s,j}^H$, and those with temporal quadrature points as $\omega_{t,j}^H$. The projection space sits in $\mathbb{Q}^{k^L}(\xi, \eta) \otimes P^{m^L}(\tau)$, and all quantities associated with this space are marked with a superscript ‘L’, such as the projected solution $Q^L(\xi) = Q^L(\xi, \eta, \tau)$ and its basis function $L_{i_1, i_2, i_3}^L(\xi) = \phi_{i_1}^L(\xi)\phi_{i_2}^L(\eta)\phi_{i_3}^L(\tau)$, $i_1, i_2 = 1, \dots, k^L + 1$ and $i_3 = 1, \dots, m^L + 1$. We can then express $Q^H(\xi)$ and $Q^L(\xi)$ as

$$\begin{aligned} Q^H(\xi) &= \sum_{i_1=1}^{k^H+1} \sum_{i_2=1}^{k^H+1} \sum_{i_3=1}^{m^H+1} Q^H(\xi_{i_1, i_2, i_3}^H) L_{i_1, i_2, i_3}^H(\xi) \\ &= \sum_{i_1=1}^{k^H+1} \sum_{i_2=1}^{k^H+1} \sum_{i_3=1}^{m^H+1} Q_{i_1, i_2, i_3}^H \phi_{i_1}^H(\xi) \phi_{i_2}^H(\eta) \phi_{i_3}^H(\tau) \\ Q^L(\xi) &= \sum_{i_1=1}^{k^L+1} \sum_{i_2=1}^{k^L+1} \sum_{i_3=1}^{m^L+1} Q^L(\xi_{i_1, i_2, i_3}^L) L_{i_1, i_2, i_3}^L(\xi) \\ &= \sum_{i_1=1}^{k^L+1} \sum_{i_2=1}^{k^L+1} \sum_{i_3=1}^{m^L+1} Q_{i_1, i_2, i_3}^L \phi_{i_1}^L(\xi) \phi_{i_2}^L(\eta) \phi_{i_3}^L(\tau) \end{aligned} \quad (38)$$

and the projection operation can be written as

$$\begin{aligned} &\iiint \sum_{i_1=1}^{k^L+1} \sum_{i_2=1}^{k^L+1} \sum_{i_3=1}^{m^L+1} Q_{i_1, i_2, i_3}^L L_{i_1, i_2, i_3}^L(\xi) L_{j_1, j_2, j_3}^L(\xi) d\xi \\ &= \iiint \sum_{i_1=1}^{k^H+1} \sum_{i_2=1}^{k^H+1} \sum_{i_3=1}^{m^H+1} Q_{i_1, i_2, i_3}^H L_{i_1, i_2, i_3}^H(\xi) L_{j_1, j_2, j_3}^L(\xi) d\xi \end{aligned} \quad (39)$$

for any test function $L_{j_1, j_2, j_3}^L(\xi)$, $j_1, j_2 = 1, \dots, k^L + 1$ and $j_3 = 1, \dots, m^L + 1$. Note that Q_{i_1, i_2, i_3}^H ’s are known on solution points, and Q_{j_1, j_2, j_3}^L ’s are calculated from Eq. (39) as

$$Q_{j_1, j_2, j_3}^L = \frac{1}{\omega_{s, j_1}^L \omega_{s, j_2}^L \omega_{s, j_3}^L} \sum_{i_1=1}^{k^H+1} \sum_{i_2=1}^{k^H+1} \sum_{i_3=1}^{m^H+1} Q_{i_1, i_2, i_3}^H \phi_{j_1}^L(\xi_{i_1}^H) \phi_{j_2}^L(\eta_{i_2}^H) \phi_{j_3}^L(\tau_{i_3}^H) \omega_{s, i_1}^H \omega_{s, i_2}^H \omega_{s, i_3}^H. \quad (40)$$

A proof of Eq. (40) is given in [Appendix A](#).

Then with $Q^L(\xi)$ defined in Eq. (38), we can calculate $Q^L(\xi_{j_1, j_2, j_3}^H)$, $j_1, j_2 = 1, \dots, k^H + 1$ and $j_3 = 1, \dots, m^H + 1$. Since $k^L < k^H$ and $m^L < m^H$, $Q^L(\xi)$ can also be defined as

$$Q^L(\xi) = \sum_{i_1=1}^{k^H+1} \sum_{i_2=1}^{k^H+1} \sum_{i_3=1}^{m^H+1} Q^L(\xi_{i_1, i_2, i_3}^H) \phi_{i_1}^H(\xi) \phi_{i_2}^H(\eta) \phi_{i_3}^H(\tau). \quad (41)$$

A proof of the equivalency between $Q^L(\xi)$ defined in Eq. (38) and that in Eq. (41) is given in [Appendix B](#).

On defining a parameter $\theta \in [0, 1]$, and $\Delta f_{i_1, i_2, i_3}^H = Q^H(\xi_{i_1, i_2, i_3}^H) - Q^L(\xi_{i_1, i_2, i_3}^H)$, we can reconstruct $\bar{Q}^H(\xi)$, the filtered $Q^H(\xi)$, as

$$\bar{Q}^H(\xi) = \sum_{i_1=1}^{k^H+1} \sum_{i_2=1}^{k^H+1} \sum_{i_3=1}^{m^H+1} (Q^L(\xi_{i_1, i_2, i_3}^H) + \theta \Delta f_{i_1, i_2, i_3}^H) \phi_{i_1}^H(\xi) \phi_{i_2}^H(\eta) \phi_{i_3}^H(\tau). \quad (42)$$

It is clear that when $\theta = 1$, $\bar{Q}^H(\xi) = Q^H(\xi)$; and when $\theta = 0$, $\bar{Q}^H(\xi) = Q^L(\xi)$. On defining $\Delta f(\xi) = Q^H(\xi) - Q^L(\xi)$, and its energy $E_g = \iiint \Delta f^2(\xi) d\xi$, we have the following conclusion:

- For any $\theta \in [0, 1]$, a $(1 - \theta^2)$ portion of E_g is exactly filtered from $Q^H(\boldsymbol{\xi})$ in the numerical reconstruction of the solution by Eq. (42).

This conclusion is also proved in [Appendix B](#).

To summarize, key procedures of projection-based polynomial filtering for aliasing error control of curvilinear space-time moving elements after each physical time step when implementing Eq. (34) are given below:

Key Procedures 2 Projection-based Filtering for Aliasing Error Control

- 1: Perform the solution projection from the polynomial space $\mathbb{Q}^{k^H}(\xi, \eta) \otimes P^{m^H}(\tau)$ to $\mathbb{Q}^{k^L}(\xi, \eta) \otimes P^{m^L}(\tau)$ following Eq. (39) and (40).
 - 2: Reconstruct the filtered solution polynomial using Eq. (42).
-

5. Numerical Experiments

In this section, we use numerical experiments to study the spatial and temporal accuracy of the nodal tensor-product STFR method on curvilinear moving grids. Both 1D and 2D hyperbolic conservation laws are used to conduct the numerical tests. For example, when the 1D linear wave propagation (advection) problem is studied, Q becomes a scalar, and the flux $F = cQ$, where c is a constant. When the 2D nonlinear Euler vortex propagation problem is studied, we have $\mathbf{Q} = (\rho, \rho u, \rho v, E)$, $\mathbf{F} = (\rho u, \rho u^2 + p, \rho uv, (E + p)u)$, and $\mathbf{G} = (\rho v, \rho uv, \rho v^2 + p, (E + p)v)$, where ρ is the fluid density, u and v are the velocities in the x - and y -directions, p is the pressure, and E is the total energy per unit volume. The Euler system is closed by the perfect gas law. Periodic boundary conditions are enforced in space in all tests presented in this section.

To measure the accuracy, the ℓ_2 error of any quantity q within the spatial domain Ω at a specific time, e.g. the time t_n of the simulation, is used. It is defined as

$$\|\Delta q_n\|_{\ell_2; \Omega} = \left(\frac{\int_{\Omega} (q_{h,n}^I - q_{a,n})^2 dV}{V} \right)^{\frac{1}{2}}. \quad (43)$$

Herein, q_h is the space-time approximation of the quantity q , q_a is the analytical (exact) value of q , and the superscript ‘ I ’ and subscript ‘ n ’ indicate that the value is an interpolated one at the time t_n . Specifically, $q_{h,n}^I$ is calculated from the space-time slab $\Omega \times [t_n - \Delta t, t_n]$ as

$$q_{h,n}^I = \sum_{i=1}^N \phi_i(1) q_{h,i}, \quad \text{with } \phi_i(\tau) = \prod_{j=1, j \neq i}^N \frac{\tau - \tau_j}{\tau_i - \tau_j}, \quad (44)$$

where $\tau_i \in [-1, 1]$, $i = 1, \dots, N$, are the temporal solution points in the standard space-time element.

Note that when conducting spatial accuracy measurement (i.e. convergence tests via spatial grid refinement), very high-order accurate temporal discretization, such as the 14th-order temporal scheme in this study, is used to ensure that the spatial errors dominate in numerical experiments. Similarly, when conducting temporal accuracy measurement (i.e. convergence tests via time step refinement), very high-order accurate spatial discretization, such as the 14th-order spatial scheme in this study, is used to ensure that the temporal errors dominate.

We first summarize several key features observed from numerical tests of the STFR method on stationary grids [42]. This sets up the cornerstone when explaining numerical experiment results from STFR simulation on curvilinear moving/deformable grids.

5.1. Stationary domain results summary

This section presents some key convergence test results for simulations of the 1D linear wave propagation and 2D Euler vortex propagation on stationary grids. We mention that on stationary grids tessellated with

regular uniform quadrilateral elements, where the Jacobian and metrics are constant, all STFR schemes with Gauss–Legendre solution points satisfy **Theorem 2**. In Figure 3, the spatial convergence rates using P^1 to P^5 spatial constructions, and the temporal convergence rates using P^1 to P^4 temporal constructions are presented. We observe that the spatial convergence rate reaches its optimal value, i.e. the $(k + 1)$ th order is achieved for a degree k polynomial construction; see Figure 3a. As shown in Figure 3b, superconvergence with the $(2k + 1)$ th order for a degree k temporal polynomial construction shows up in temporal convergence tests. We mention again that a space-time tensor product operation is used to construct the FR formulation, and the Gauss–Legendre quadrature points are used as solution points both in space and time (see Figure 2). As has been proved by Huynh [27], in the temporal direction, the FR scheme with the Gauss–Legendre solution points is equivalent to the so-called IRK DG-Gauss scheme when the quadrature rule based on the solution points (i.e. quadrature points used in DG) is sufficiently accurate to integrate the space-time curvilinear elements. Since the space-time discretization in the time dimension with a degree k polynomial construction is equivalent to the IRK DG-Gauss scheme with $(k + 1)$ stages, the order of accuracy of which is $(2k + 1)$, the temporal superconvergence rate of the STFR scheme agrees with its theoretical value; see Section 4.4 in [27].

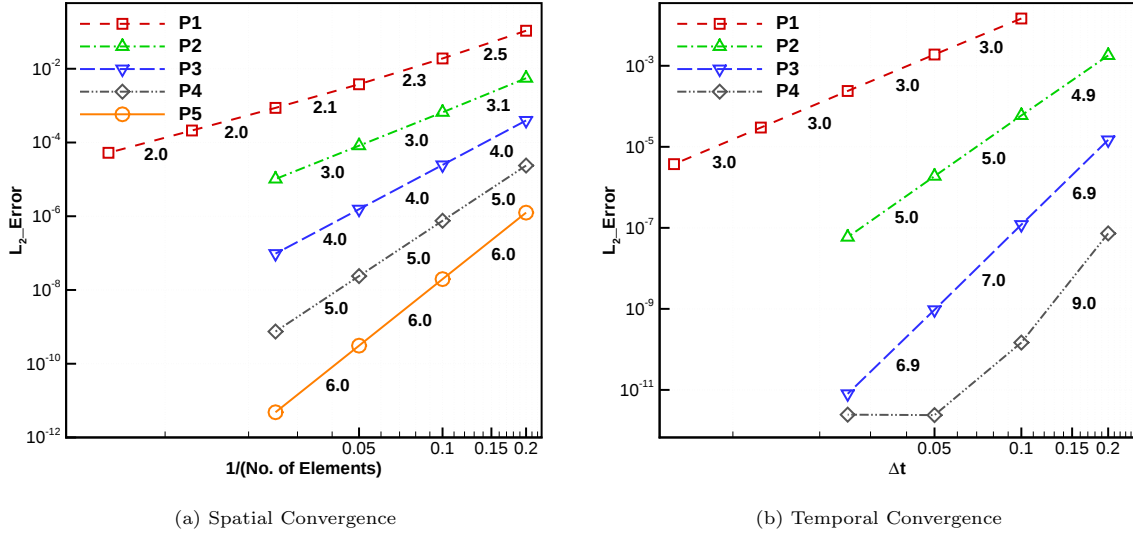


Figure 3: (a) Spatial convergence rates from P^1 to P^5 spatial constructions; and (b) temporal convergence rates from P^1 to P^4 temporal constructions for the 1D wave propagation problem on a stationary grid at $t = 1$ (i.e. one wave propagation period).

The spatial and temporal convergence rates for simulations of the Euler vortex propagation (see the inviscid vortex setup in [42]) were also studied with Q^2 to Q^5 spatial constructions and P^1 to P^3 temporal constructions. Results are presented in Figure 4. Similar to the 1D linear wave propagation tests, we observe that the spatial convergence rates match their optimal values, i.e. the $(k + 1)$ th order for a degree k polynomial construction, and the temporal convergence rates show the theoretical superconvergent features, i.e. the $(2k + 1)$ th order for a degree k polynomial construction. This demonstrates that spatial nonlinearity from the governing PDEs does not affect the temporal superconvergence of the STFR schemes. Therefore, we will use 2D linear wave propagation to test the curvilinear space-time effects of moving domains on the convergence of STFR schemes in following sections.

Key Observation 5.1. *With Gauss–Legendre quadrature points serving as spatiotemporal solution points, the nodal STFR formulation shows a $(2k + 1)$ th order superconvergence rate for a degree k polynomial construction (i.e. with $k + 1$ solution points) in the time dimension on stationary grids, no matter for linear or nonlinear hyperbolic conservation laws with adequate smooth features.*

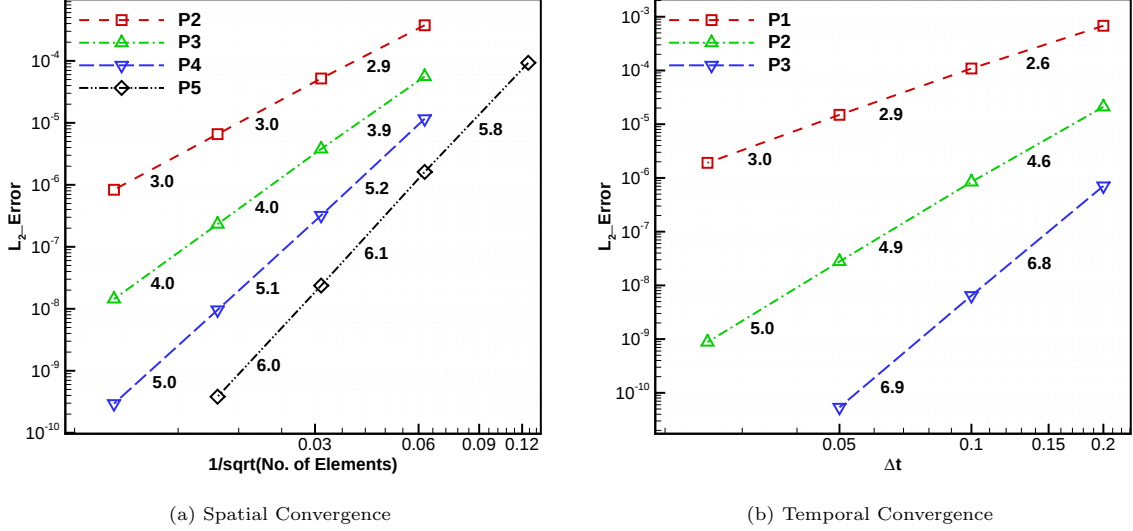


Figure 4: (a) Spatial convergence rates from Q^2 to Q^5 spatial constructions for the 2D Euler vortex propagation problem on a stationary grid at the $1/8$ wave propagation period; and (b) temporal convergence rates from the P^1 to P^3 temporal constructions evaluated at one wave propagation period.

5.2. Tensor-product space-time elements with linear spatial and temporal geometric representations in a moving domain

Starting from this section, numerical schemes will be termed with three labels, i.e. ‘S’ representing those satisfying **Theorem 3**, ‘P’ representing those satisfying the **Proposition 4** but violating **Theorem 3**, and ‘V’ representing those violating the **Proposition 4** (and **Theorem 3**). We mention that when Eq. (34) is implemented with **Key Procedures 1**, a scheme labeled ‘V’ that violates **Proposition 4** can still preserve the freestream.

Now we examine which accuracy label, i.e. ‘S’, ‘P’, or ‘V’, the numerical schemes with different nominal spatial approximation degree k and temporal approximation degree m should have when linear spatial and temporal representations are used to describe the grid element. According to the polynomial space used in Table 4, we have $l = 1$ and $n = 1$ for linear space-time elements. Thus, for $k = 2, 3$, and 4 (i.e. nominal 3^{rd} -, 4^{th} -, and 5^{th} -order spatial constructions), and $m = 1$ and 2 (i.e. nominal 2^{nd} - and 3^{rd} -order temporal constructions with the corresponding 3^{rd} - and 5^{th} -order superconvergence, respectively), the accuracy labels of the STFR schemes are summarized in Table 5. Since $\bar{Q}$ is used as the hidden working variable in the STFR schemes for general curvilinear grids, different numbers of solution points can be used to construct the physical quantity Q , depending on the desired accuracy of the hidden working variable $\bar{Q}$. Note that as a rule of thumb, the order of accuracy of a numerical scheme is measured by the number of solution points used to construct the explicit working variable, i.e. the solution Q here. However, depending on the nominal order of accuracy, which depends on the expected spatial approximation degree k , we can label the scheme with ‘S’, ‘P’, or ‘V’. Take “# of SPs” equal to 4 in Table 5a as an example. If **Theorem 3** is strictly satisfied (i.e. with label ‘S’), then the nominal order of accuracy for the solution Q should be 3; if only **Proposition 4** needs to be satisfied, then the nominal order of accuracy for the solution Q can be up to 4. In practical numerical tests, the convergence rate can be between 3 and 4 when **Key Procedures 1** is used to implement STFR. The filtering operation in **Key Procedures 2** can then be used to stabilize the convergence rate to 3; see numerical experimental results presented in **Sect. 5.2.1**.

This section shows the convergence test results for simulations of the 2D linear wave propagation on moving/deformable grids with the linear space-time element representation. Two types of problems are considered: (1) deformable grids with spatial and temporal symmetry; and (2) moving and deformable grids without spatial and temporal symmetry. Starting from this section, we use the number of solution points (SPs) employed in STFR schemes for solution construction as the measure of desired order of accuracy. Note that the desired order of accuracy is always equal to or greater than the nominal one.

Table 5: Scheme labels based on the nominal order of accuracy when linear space-time elements, i.e. $l = 1$ and $n = 1$, are used to represent the moving/deformable grids. Herein, “OoA” stands for “Order of Accuracy”, “SPs” stands for “Solution Points”, and “#” indicates “number”.

(a) Spatial Discretizations				(b) Temporal Discretizations			
Desired OoA = # of SPs				Desired OoA with superconvergence = $2 \times (\# \text{ of SPs}) - 1$			
k	Nominal OoA	# of SPs	Label	m	Nominal superconvergence	# of SPs	Label
2	3	3	‘P’	1	3	2	‘V’
		4	‘S’			3	‘P’
3	4	4	‘P’			4	‘S’
		5	‘S’			5	‘P’
4	5	5	‘P’	2	5	3	‘P’
		6	‘S’			4	‘P’
						5	‘S’

5.2.1. Deformable grids with spatial and temporal symmetry

In this subsection, numerical tests are performed with linear spatial and temporal representations of deformable grids with spatial and temporal symmetry. The grid deformation strategy is given as follows [53]:

$$\begin{cases} dx(\mathbf{x}, t) = A_x L_x \sin(\omega_t t) \sin(\omega_x x) \sin(\omega_y y) dt / t_{max}, \\ dy(\mathbf{x}, t) = A_y L_y \sin(\omega_t t) \sin(\omega_x x) \sin(\omega_y y) dt / t_{max}. \end{cases} \quad (45)$$

Herein, dx and dy are the grid displacement at the time t in the x - and y -directions, dt is the temporal step size, L_x , L_y and t_{max} are reference values in the x -, y - and t -directions, and A_x and A_y are scaling factors to control the amplitudes of grid displacement in the x - and y -directions. The temporal and spatial angular frequencies are defined as

$$\omega_t = (n_t \pi) / t_{max}, \quad \omega_x = (n_x \pi) / L_x, \quad \omega_y = (n_y \pi) / L_y.$$

In this study, the parameters are set as $A_x = A_y = 0.1$, $L_x = L_y = 1.0$, $n_t = 0.5$, $n_x = n_y = 4.0$, and $t_{max} = 0.2$. The computational domain is set as $[0, 1] \times [0, 1]$, and all simulations are run until $t = t_{max} = T/4$. The wave field on top of the corresponding deformed grid tessellated with 64×64 elements for the 2D linear wave propagation problem at $t = t_{max}$ is displayed in Figure 5a for visualization. Since bilinear polynomials are used to represent grid elements, as shown in Figure 5b, where a grid with 16×16 elements is plotted on top of that with 8×8 elements, the two sets of grids used in the spatial convergence tests do not match each other. This issue also exists in the time dimension, and can affect the spatiotemporal convergence rates as will be shown shortly.

We first present the spatial convergence rates for spatial constructions using 3 to 6 solution points in each spatial dimension in Figure 6a. We observe that the order of accuracy shows severe deterioration at the coarser grid side, e.g. grids with 8×8 and 16×16 elements, and gradually recovers towards the optimal value (i.e. the number of solution points in each spatial dimension) when the grid becomes finer. The convergence rate deterioration over coarser grids is expected to be fixed when grid representation errors are reduced when curvilinear elements are adopted. This will be further discussed in **Sect. 5.3**.

The temporal convergence rates are shown in Figure 6b. We observe that the schemes with 2 and 3 solution points in the time dimension show the same superconvergence rate as that on stationary grids; specifically, with j temporal solution points, a $(2j - 1)$ th order superconvergence shows up. Note that based on Table 5, the scheme with 2 temporal solution points violates the **Proposition 4** as well as **Theorem 3**. However, this does not cause convergence rate deterioration. As will be reported in following sections, similar phenomena have been observed in cases with stronger nonlinearity. This contributes to the good numerical property of the STFR schemes when Eq. (34) is implemented with **Key Procedures 1**; see similar observations reported in [23]. For schemes with 4 and 5 temporal solution points, the superconvergence rates

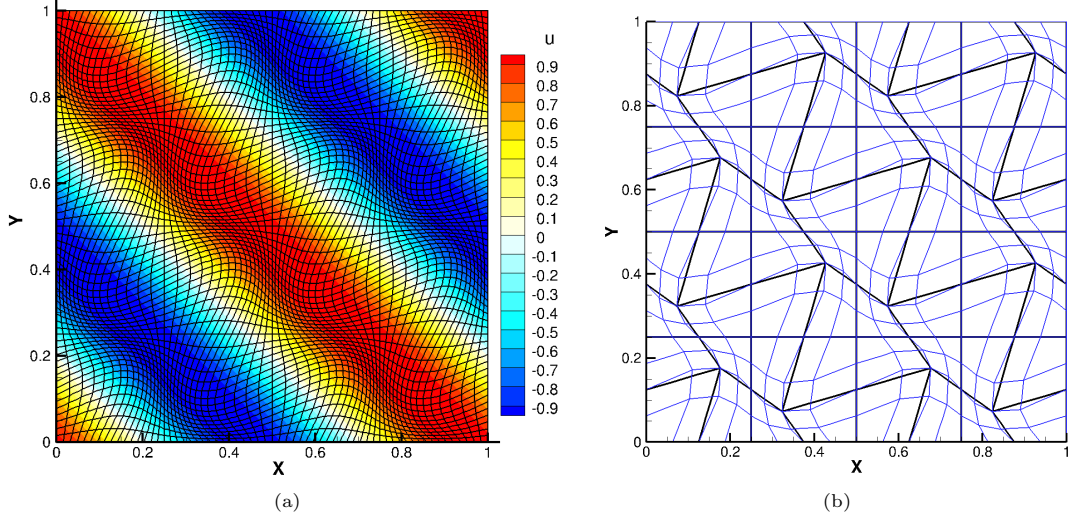


Figure 5: (a) The wave field and the corresponding deformed grid at $t = t_{max} = 0.2$ for the 2D linear wave equation. (b) The grid with 16×16 bilinear elements plotted on top of that with 8×8 bilinear elements.

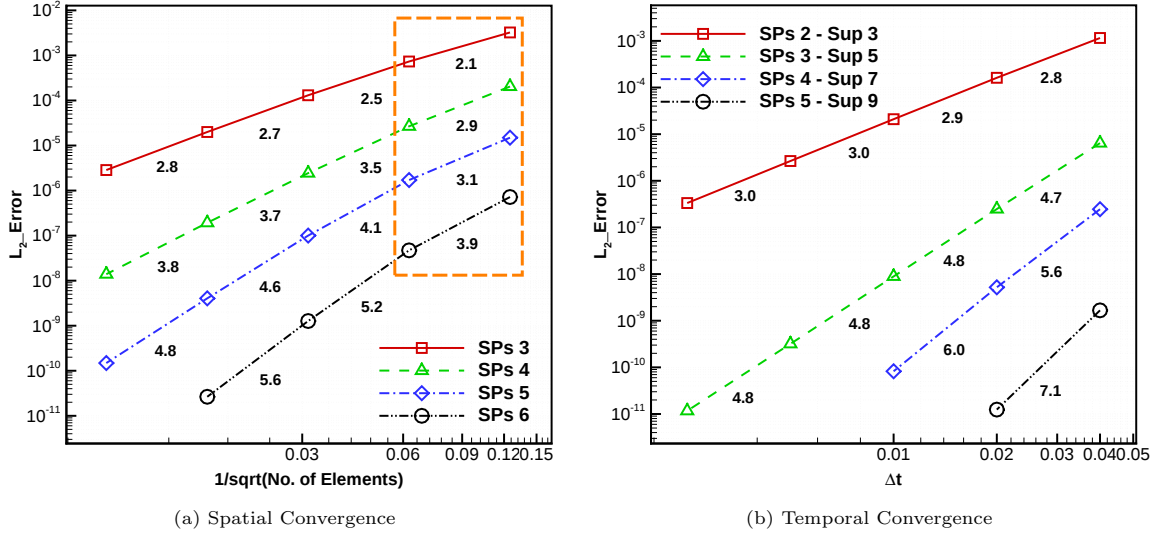


Figure 6: (a) Spatial convergence rates for spatial constructions using 3 to 6 solution points in each spatial dimension; and (b) temporal convergence rates for temporal constructions using 2 to 5 solution points in the time dimension for the 2D linear wave equation on a deforming square domain tessellated with linear space-time elements.

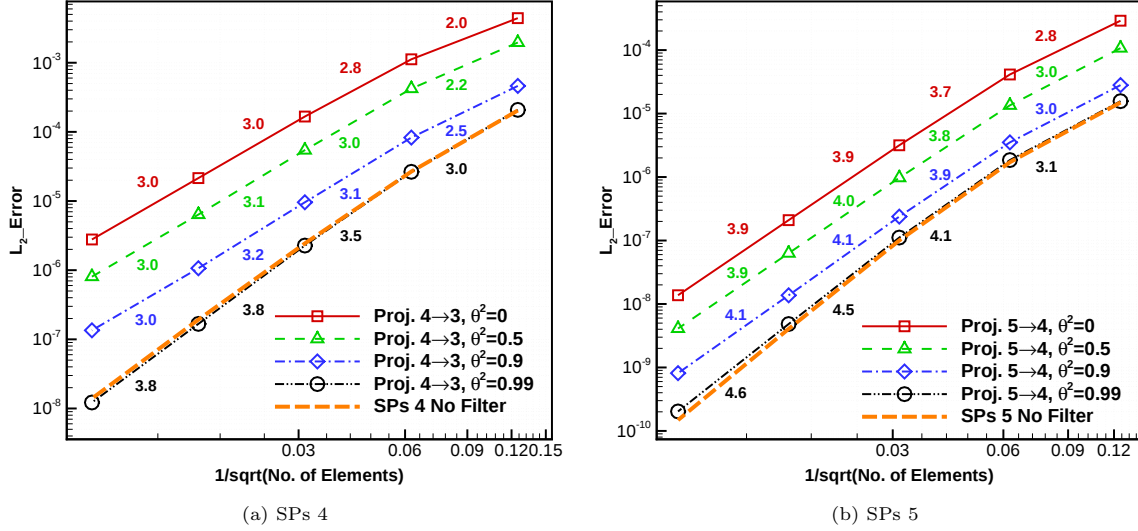


Figure 7: Spatial convergence rates for spatial constructions using (a) 4 and (b) 5 solution points in each spatial dimension with different filtering strengths.

do not reach the desired optimal values. Due to the very small absolute errors (i.e. smaller than 10^{-9}) possessed by this deforming grid problem, it is hard to probe the exact numerical issues. But as will be seen from the deforming grid cases without temporal symmetry presented in **Sect. 5.2.2** and **5.3**, the superconvergence rate deterioration can be due to the grid representation error. Evidence will be provided later when numerical errors are sufficiently large so that a group of superconvergence rates can be compared in a meaningful manner.

Note that all schemes presented in Figure 6 do not explicitly apply the aliasing error control. As discussed in **Sect. 4.2.3**, we can use projection-based polynomial filtering to control aliasing errors caused by the nonlinear interaction between flow quantities and the curvilinear geometric representation of space-time elements, thus stabilizing the convergence rates. We first present results of spatial polynomial filtering with different filtering strengths in Figure 7. Recall that with the parameter θ , a $(1 - \theta^2)$ portion of the energy in the solution between the original and projection spaces is filtered out. From Figure 7, when 1% (i.e. $\theta^2 = 0.99$) energy is filtered out, the solution is not significantly affected; when a nontrivial portion, such as more than 10%, of energy is filtered out, the convergence rate of the scheme can be stabilized at the value possessed by the projection space. This further explains the practical meaning of the scheme labels presented in Table 5a. As explained early in **Sect. 5.2**, and still taking “# of SPs” of 4 as an example, when the expected nominal order of accuracy is 4, the scheme is labeled with ‘P’. As shown in Figure 7, its actual convergence rate without filtering is 3.8, close to 4. However, when the expected nominal order of accuracy is 3, the scheme is labeled with ‘S’. Apparently, the actual convergence rate of the solution is greater than 3 (or in another way, at least 3, which meets the order of accuracy expectation according to the conditions to meet **Theorem 3**), and with polynomial filtering of considerable strength, the convergence rate can be stabilized at 3.

When filtering is applied in the time dimension, we find that the superconvergence property of the temporal schemes is lost. As shown in Figure 8, when a degree $j - 1$ temporal polynomial construction with j solution points is projected onto the space sit by degree $j - 2$ polynomials, the local convergence rate is $j - 1$; due to the error accumulation in time marching (i.e. when the time step size is halved, the simulation step doubles), the global convergence rate of the filtered scheme becomes $j - 2$. It is clear that all schemes tested with different filtering strengths show the theoretical global convergence rate. Since the convergence rate of the filtered scheme drops from the superconvergent one $2j - 1$ to the regular global value $j - 2$ when j temporal solution points are used, and the absolute error of the filtered scheme can be significantly larger than that of the original scheme, temporal polynomial filtering is not recommended even if a scheme cannot

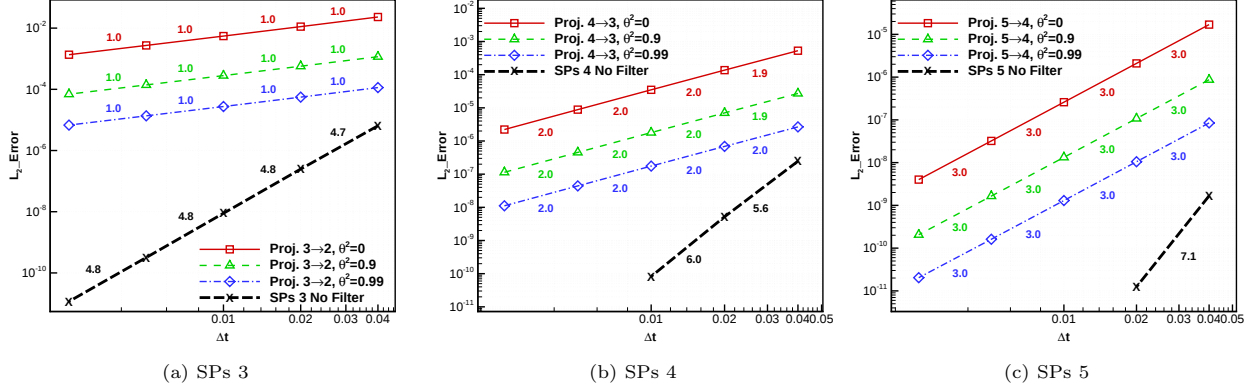


Figure 8: Global temporal convergence rates for temporal constructions using (a) 3, (b) 4, and (c) 5 solution points in the time dimension with different filtering strengths. The convergence rate of the filtered scheme drops from the superconvergent one $2j - 1$ to the regular global value $j - 2$, where j is the number of temporal solution points.

reach its theoretical superconvergence rate, such as the schemes with 4 or 5 temporal solution points.

5.2.2. Deformable grids without spatial and temporal symmetry

To further study the convergence property of the moving grid STFR method, a moving and deforming circular domain case designed in [54] is studied in this subsection. Since the purpose of this study is to test whether the time-varying Jacobian and metrics of a moving grid with spatiotemporal symmetry-breaking large deformation can disturb the flow fields, only the 2D linear wave propagation problem with analytical boundary conditions is used for testing purposes. The grid motion within a circular domain of a radius 0.5 centered at $(0, 0)$ is given as follows:

$$\begin{pmatrix} x(r_0, \theta_0, t) \\ y(r_0, \theta_0, t) \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & \alpha(t) \end{pmatrix} \begin{pmatrix} \cos(A_\theta \alpha(t)) & -\sin(A_\theta \alpha(t)) & 0 \\ \sin(A_\theta \alpha(t)) & \cos(A_\theta \alpha(t)) & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \psi(t) & 0 & 0 \\ 0 & 1/\psi(t) & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} r_0 \cos(\theta_g(r_0, \theta_0, t)) \\ r_0 \sin(\theta_g(r_0, \theta_0, t)) \\ 1 \end{pmatrix}. \quad (46)$$

Herein, (r_0, θ_0) is the relative position of the grid point to the origin $(0, 0)$ at $t = 0$ measured in the polar coordinate system. The functions $\alpha(t)$, $\psi(t)$, and $\theta_g(r_0, \theta_0, t)$ are defined as follows:

$$\begin{cases} \alpha(t) = t^3(8 - 3t)/16 \\ \psi(t) = 1 + (A_a - 1)\alpha(t) \\ \theta_g(r_0, \theta_0, t) = \theta_0 + A_g f_g(r_0, \theta_0, t) \end{cases}, \quad (47)$$

where the function f_g is defined as

$$f_g(r_0, \theta_0, t) = \frac{t^6}{t^6 + 0.01} (16r_0^4 + \eta(t, 10, 0.7)(\cos(32\pi r_0^4) - 1)) \eta(\theta_0, 1, 0.7) \quad (48)$$

with the symmetry-breaking perturbation η defined as

$$\eta(\lambda, \omega, \tau) = \sin(\omega\lambda + \tau(1 - \cos(\omega\lambda))).$$

Note that the definition of the function f_g in Eq. (48) is slightly different to that defined in [54] to ensure that there is no motion jump at $t = 0$. In the above equations, A_θ is a constant rotation amplitude, A_a is a constant amplification factor for the deformation of a circle into an ellipse, A_g is a constant volume deformation amplitude. Following [54], they are set as $A_\theta = \pi$, $A_a = 1.5$, and $A_g = 0.15$. In the function

η , the variable λ represents the independent variables, such as t and θ_0 , and the variables ω and τ are the angular frequency and phase-lag factor, respectively. The physical domain at $t = 0$ and the deformed one at $t = 1$, together with the corresponding linear wave fields, are shown in Figure 9. We observe that the wave field is not disturbed by the grid motion and deformation.

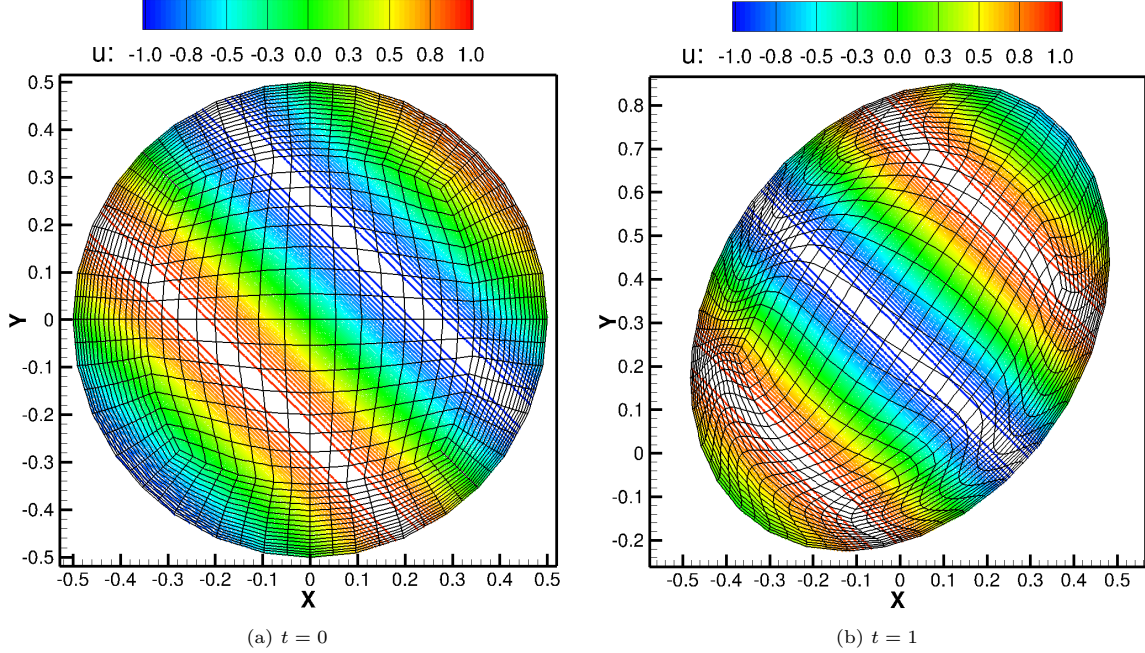


Figure 9: The wave field and the corresponding grid at (a) $t = 0$ and (b) $t = 1$ for the 2D linear wave equation on a moving and deforming circular domain.

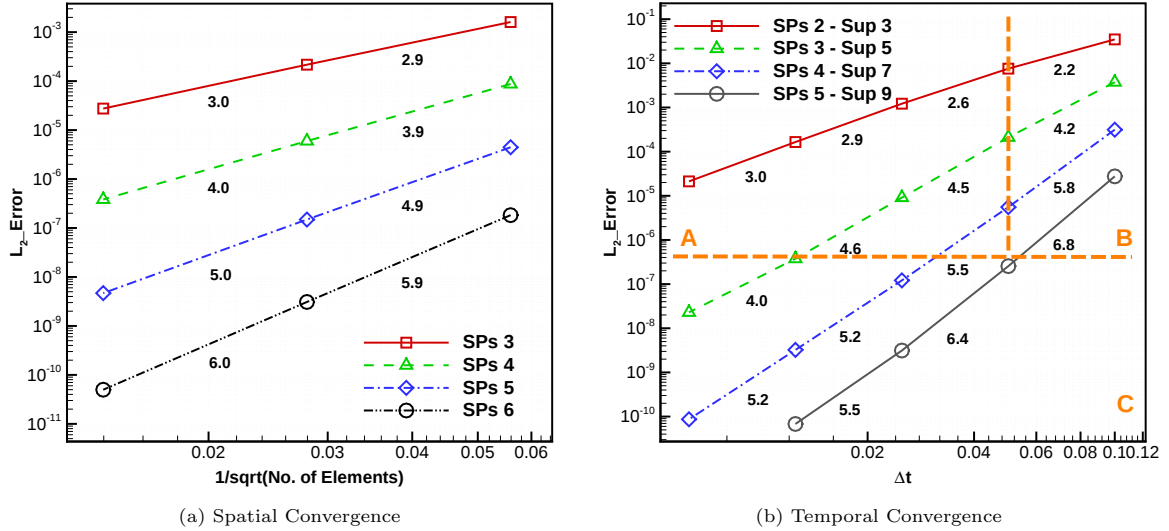


Figure 10: (a) Spatial convergence rates for spatial constructions using 3 to 6 solution points in each spatial dimension; and (b) temporal convergence rates for temporal constructions using 2 to 5 solution points in the time dimension for the 2D linear wave equation on a moving and deforming circular domain tessellated with linear space-time elements.

We now present results from the space-time order of accuracy studies in Figure 10. The spatial convergence rates for all STFR schemes reach their optimal rates. The temporal convergence rates show some interesting trends, which becomes more apparent in Figure 16b when curvilinear space-time elements are

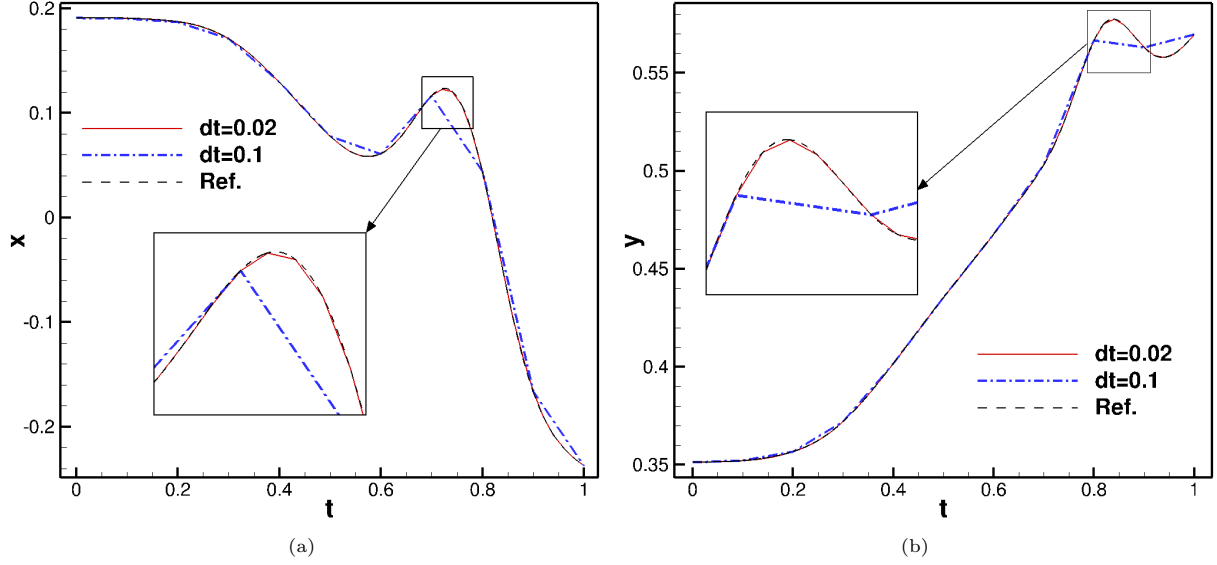


Figure 11: The trajectories (a) x and (b) y of the grid point originally at $(r_0, \theta_0) = (0.4, 1.07274)$ under different Δt conditions when the grid motion is approximated as a linear polynomial.

used to represent the moving/deforming grid. When 2 temporal solution points are used, the scheme's order of accuracy converges toward the superconvergent one, i.e. 3; when 3 temporal solution points are used, the scheme's order of accuracy first converges toward the superconvergent one, i.e. 5, and then starts to decrease when the time step size is further refined; and when 4 or 5 temporal solution points are used, the schemes show superconvergence but not at the optimal rate, and this rate decreases in the time step refinement study.

To explain this phenomenon, we divide the Δt -Error diagram into three regions, as shown in Figure 10b. In Region "A", the physical errors dominate. This is why the order of accuracy of the schemes converge towards the optimal superconvergence rate. In Region "B", significant space-time element representation errors exist. In Figure 11, the trajectories x and y of the grid point originally at $(r_0, \theta_0) = (0.4, 1.07274)$ within the time interval $[0, 1]$ under different Δt conditions are displayed. It shows that although at the end time the x and y are the same for all cases, their trajectories are different. This results in the uncertain convergence rate for all schemes tested in that region. In Region "C", the space-time element representation errors interact with the physical errors, causing convergence rate deterioration. This will be further confirmed in Figure 16b when curvilinear space-time elements are used to reduce the space-time element representation errors.

Key Observation 5.2. *STFR schemes show superconvergence in the temporal direction on deformable grids with large deformation, although they may not reach the optimal rate for very high-order setups due to insufficient grid motion trajectory resolution compared to the resolution of flow variables. The same observation also applies to the curvilinear space-time element cases tested in Sect. 5.3.*

Key Observation 5.3. *STFR schemes with the label 'S' (i.e. satisfying Theorem 3) can at least achieve the nominal order of accuracy when no filtering operation is applied. The same observation also applies to the curvilinear space-time element cases tested in Sect. 5.3.*

Key Observation 5.4. *Spatiotemporal projection-based polynomial filtering can stabilize the convergence rate of the STFR scheme. Specifically, spatial filtering can be an effective way to control the aliasing errors when implementing Eq. (34) with Key Procedures 1. Temporal filtering can result in the loss of superconvergence.*

5.3. Curvilinear tensor-product space-time elements in a moving domain

In this section, curvilinear space-time grids are used to represent moving/deforming domains. The same two grid deformation strategies, i.e. Eq. (45) and (46), used in Sect. 5.2, are employed here. Quadratic

Table 6: Scheme labels based on the nominal order of accuracy when curvilinear space-time elements with $l = 2$ and $n = 2$ are used to represent the moving/deforming grids.

(a) Spatial Discretizations				(b) Temporal Discretizations			
Desired OoA = # of SPs				Desired OoA with superconvergence = $2 \times (\text{\# of SPs}) - 1$			
k	Nominal OoA	# of SPs	Label	m	Nominal superconvergence	# of SPs	Label
2	3	3	‘V’	1	3	2	‘V’
		4	‘V’			3	‘V’
		5	‘P’			4	‘V’
		6	‘S’			5	‘P’
3	4	4	‘V’			6	‘P’
		5	‘P’			7	‘S’
		6	‘P’			3	‘V’
		7	‘S’			4	‘V’
4	5	5	‘P’	2	5	5	‘P’
		6	‘P’			6	‘P’
		7	‘P’			7	‘P’
		8	‘S’			8	‘S’

space-time elements with $l = 2$ and $n = 2$ are used to better capture the shape and motion of moving and deforming grids. In Figure 12a, the grid tessellated with 16×16 curvilinear elements is plotted on top of that with 8×8 curvilinear elements for comparison. Although they do not strictly match each other, the errors caused by imperfect geometric representation of deforming grids during the convergence rate study are better controlled compared to the linear grid cases used in Sect. 5.2 (see Figure 5b). Similarly, as shown in Figure 12b and 12c, quadratic representations of the motion can better capture the grid point trajectories compared to linear representations in Figure 11.

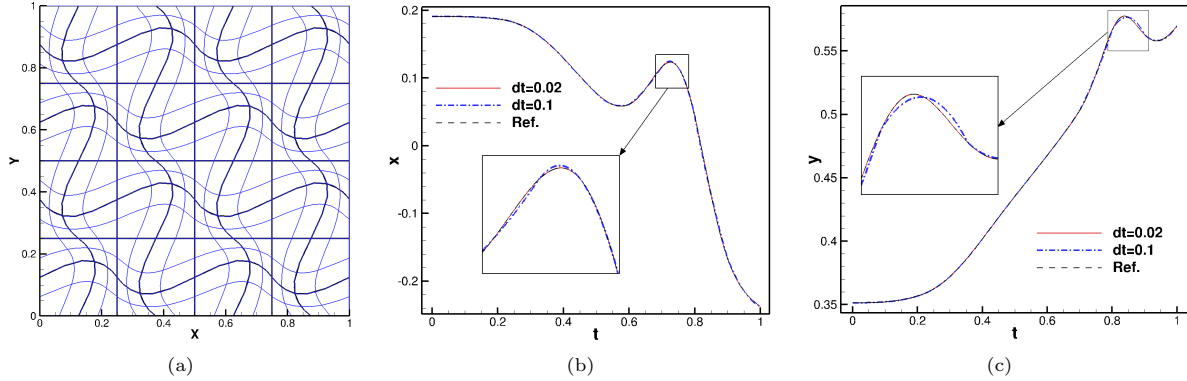


Figure 12: (a) The grid with 16×16 curvilinear elements plotted on top of that with 8×8 curvilinear elements with the motion described by Eq. (45). The trajectories (b) x and (c) y of the grid point originally at $(r_0, \theta_0) = (0.4, 1.07274)$ under different Δt conditions when the grid motion described by Eq. (46) is approximated as a quadratic polynomial.

Similar to Sect. 5.2, we list the accuracy label, i.e. ‘S’, ‘P’, or ‘V’, of the STFR schemes with different nominal spatial approximation degree k and temporal approximation degree m for curvilinear spatial and temporal representations of grid elements with $l = 2$ and $n = 2$ in Table 6. Compared to the linear space-time grid cases with $l = 1$ and $n = 1$, it becomes harder for the STFR schemes to meet the conditions required by Theorem 3 when quadratic space-time elements are used to represent moving/deforming grids. As a result, numerical simulations are more prone to being affected by aliasing errors due to the nonlinear interaction between flow quantities and the geometric representation of space-time elements.

We first present results in convergence studies of the deformable grid case with the motion described by

Eq. (45). The results of the spatial and temporal convergence rates are shown in Figure 13. Compared to the linear spatial grid case presented in Figure 6a, the spatial convergence rate at the coarser grid side has recovered to the optimal value due to the better geometric representation. Similar to the linear motion representation case presented in Figure 6b, the schemes with 2 and 3 temporal solution points show the optimal superconvergence rate. The superconvergent feature of the scheme with 4 temporal solution points improves due to the reduced motion trajectory representation errors. The deterioration of the superconvergence rate of the scheme with 5 temporal solution points is due to accuracy limitation (i.e. with an absolute error around $\mathcal{O}(10^{-12})$) of the numerical schemes used in this study.

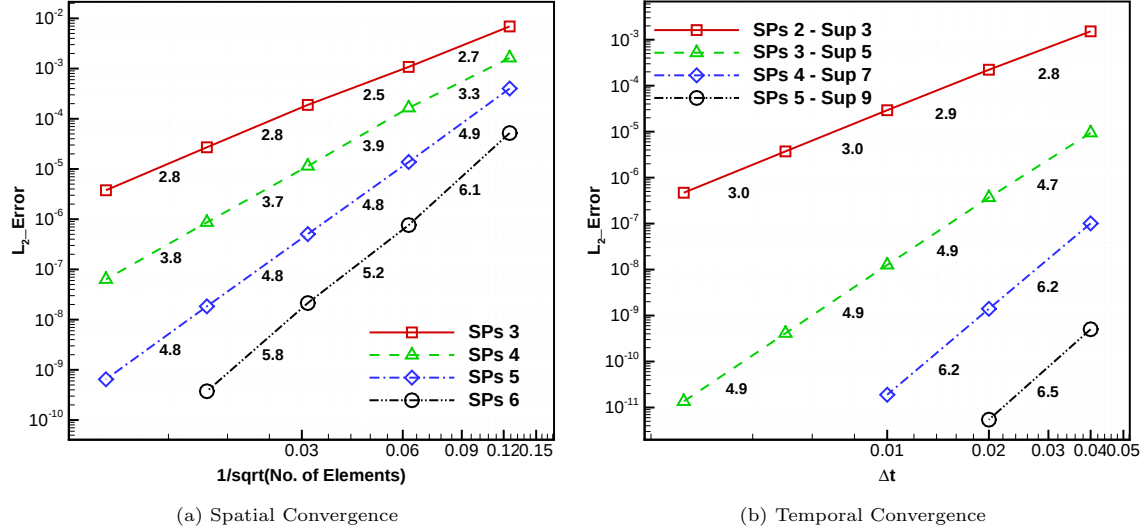


Figure 13: (a) Spatial convergence rates for spatial constructions using 3 to 6 solution points in each spatial dimension; and (b) temporal convergence rates for temporal constructions using 2 to 5 solution points in the time dimension for the 2D linear wave equation on a deforming square domain tessellated with quadratic space-time elements.

Spatial and temporal projection-based polynomial filtering is then applied to the schemes tested in Figure 13, and results are presented in Figure 14 and 15. As expected, we have similar observations as those obtained from the linear space-time grids; see **Key Observation 5.4**. According to Table 6a, the scheme with 6 solution points in each spatial dimension has the label ‘S’, ‘P’, and ‘P’ when the nominal order of accuracy is expected to 3, 4, and 5. Therefore, the projection-based polynomial filtering to $\mathbb{Q}^2$ (i.e. the tensor-product space, on which the scheme with 3 solution points in each spatial dimension is constructed), $\mathbb{Q}^3$, and $\mathbb{Q}^4$ spaces with the filtering parameter $\theta^2 = 0.9$ is applied to the scheme with 6 solution points in each spatial dimension. We observe from Figure 14b that all filtered schemes achieve the expected convergence rates. Therein, the corresponding unfiltered schemes with the same convergence rates are also plotted for comparison. We find that the absolute errors of the filtered schemes are about two orders of magnitude smaller than those of their unfiltered counterparts.

Finally, we present results in the order of accuracy studies of the moving/deforming circular case with the motion described by Eq. (46) in Figure 16. Similar to the linear grid case (see Figure 10a), the spatial convergence rates for all STFR schemes reach their optimal rates. As shown in Figure 16b, we divide the Δt -Error diagram into three regions. Compared to Figure 10b, the area of Region “A” in Figure 16b is enlarged, where the superconvergence rates of the schemes with 2, 3, and 4 temporal solution points are converging towards their optimal values. This is due to the reduced motion trajectory representation error when curvilinear space-time elements are used. Nevertheless, compared to the physical flow field error, the motion trajectory representation error becomes relatively large again in Region “C”, and the superconvergence rates start to degrade in this region.

Key Observation 5.5. *Curvilinear space-time elements can better capture the geometric features and motion of moving/deforming grids compared to their linear counterpart, with the trade-off of accumulated alias-*

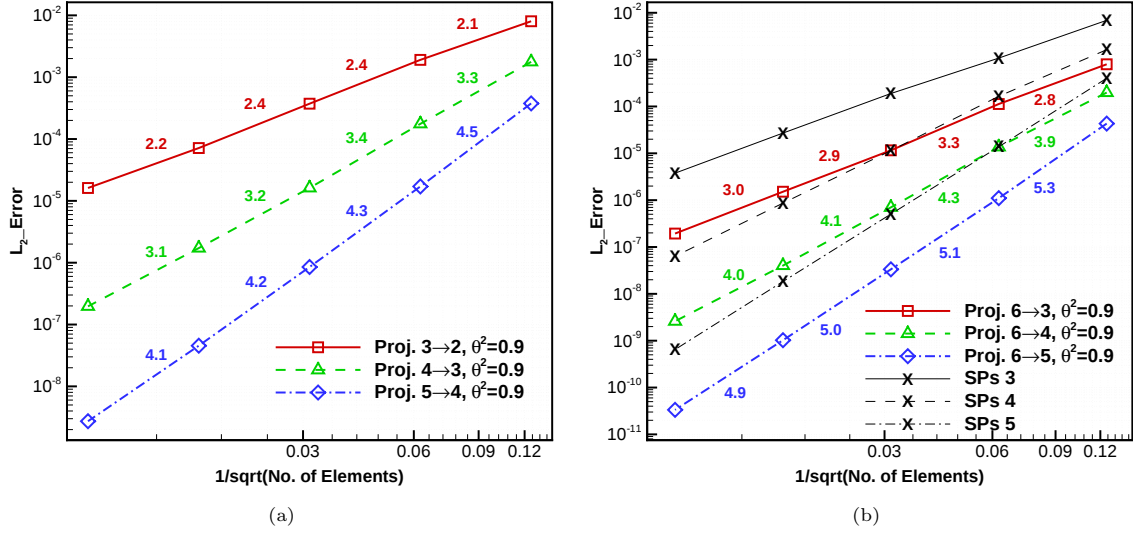


Figure 14: (a) Spatial convergence rates for spatial constructions using 3 to 5 solution points in each spatial dimension with the filtering parameter $\theta^2 = 0.9$. (b) Spatial convergence rates for the spatial construction using 6 solution points in each spatial dimension with the project-based polynomial filtering to $\mathbb{Q}^2$, $\mathbb{Q}^3$, and $\mathbb{Q}^4$ spaces and with the filtering parameter $\theta^2 = 0.9$.

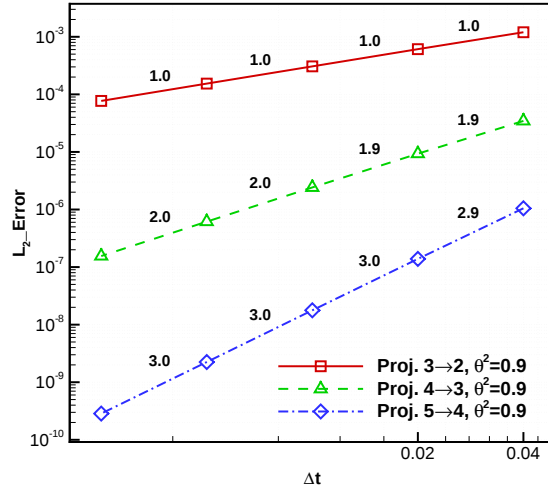


Figure 15: Global temporal convergence rates for temporal constructions using 3, 4, and 5 solution points in the time dimension with the filtering parameter $\theta^2 = 0.9$.

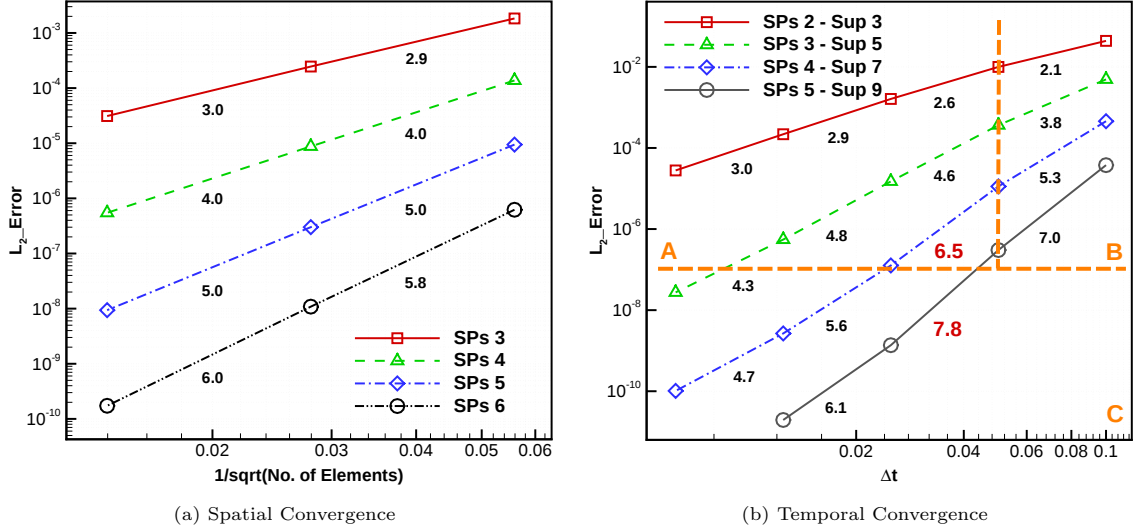


Figure 16: (a) Spatial convergence rates for spatial constructions using 3 to 6 solution points in each spatial dimension; and (b) temporal convergence rates for temporal constructions using 2 to 5 solution points in the time dimension for the 2D linear wave equation on a moving and deforming circular domain tessellated with quadratic space-time elements.

ing errors. When aliasing errors due to insufficient spatial resolution dominate, they can be effectively controlled by spatial projection-based polynomial filtering. Aliasing errors due to the insufficient temporal resolution seem not to cause serious accuracy deterioration. Similarly to the linear space-time element cases, temporal filtering can make the filtered schemes fully lose temporal superconvergence, and present a regular global convergence rate.

5.4. Discussions

From the numerical tests presented in **Sect. 5.2** and **5.3**, we observed that schemes violating **Theorem 3** and **Proposition 4** do not substantially destroy the flow simulation accuracy, and when the scheme's order of accuracy improves, their accuracy is similar to that of the schemes which satisfy **Theorem 3** or **Proposition 4**. This is *de facto* the numerical accuracy enhancement feature of the STFR schemes when Eq. (34) is implemented with **Key Procedures 1**. Similar phenomena have been reported in [23] (see Section 5.3 in that work). We restate that when a scheme violates **Theorem 3**, Eq. (34) and (32) are not numerically identical to each other. Violating **Proposition 4** (i.e. the discrete GCL) can cause serious accuracy deterioration when Eq. (32) is implemented with $|J|Q_h$ explicitly approximated as polynomials; see examples from [10, 23, 24]. As discussed in **Sect. 4.2.3**, aliasing errors still need to be controlled when implementing Eq. (34) with Q_h explicitly approximated as polynomials in the reference domain. Based on the tests presented in this section, polynomial filtering can be an effective way to steer the spatial convergence rate of numerical schemes towards the desired value. Temporal filtering is not recommended considering that it can cause the numerical schemes to fully lose the temporal superconvergence property.

6. Conclusion and Future Work

In this study, high-order nodal tensor-product STFR methods based on Gauss-Legendre points have been developed to solve hyperbolic conservation laws on curvilinear moving grids. The complex grid motion and deformation are represented with general curvilinear space-time elements, and numerical resolution requirements to resolve curvilinear geometric features are studied. A key finding of this study is that the STFR schemes consistently exhibit superconvergence in the time dimension, irrespective of whether the governing equations are linear or nonlinear, or whether the computations are performed on stationary or highly deforming moving grids. The superconvergence rate, however, can be influenced by errors associated with the grid motion representation. The use of curvilinear space-time elements, even when the grid and

its motion are represented quadratically, substantially enhances the accuracy of the space-time geometric representation. Nevertheless, a finer numerical resolution than that used for linear space-time grids is required to adequately resolve both the flow field and the geometric features of the curvilinear elements.

A STFR scheme implementation procedure based on a hybrid reference-physics domain approach, i.e. **Key Procedures 1**, is developed to enhance the accuracy of the numerical schemes. Note that this procedure can preserve the freestream even when the discrete GCL is violated. It has been proved that this procedure is numerically equivalent to the conservative form in the reference domain when **Theorem 3** is satisfied. Space-time projection-based polynomial filtering has been developed to control aliasing errors of the STFR schemes implemented with the hybrid reference-physics domain approach when their resolution can either not satisfy that required by the discrete GCL (i.e. violating **Proposition 4**) or not guarantee the equivalence between the hybrid form and its corresponding conservative form (i.e. violating **Theorem 3**). A key observation here is that the polynomial filtering procedure can be used to stabilize the spatial order of accuracy of the STFR scheme to any value less than the desired one determined by the spatial solution point setup. However, the temporal superconvergence is totally lost when temporal polynomial filtering is used; instead, a global temporal convergence rate is observed.

Several directions for future work naturally extend from the present study. First, in practical fluid flow simulations, four-dimensional (4D), i.e. 3D space plus 1D time, space-time grids need to be used. It is not clear whether the temporal superconvergence, an essential feature for numerical efficiency, still holds for general 4D polytopes [48, 50]. Second, much research is still needed to bridge the method of lines and space-time discretization for moving domain simulation with curvilinear grids. Based on some preliminary tests presented in [43], very small time steps need to be used when the method of lines is adopted for time integration to match the accuracy obtained by the space-time method. Therefore, another promising future research direction can be identifying equivalent method of lines schemes to space-time ones, such as the DG-type methods for ordinary differential equations reported in [27]. As a result, the knowledge created from the space-time methods can be directly transferred to the development of the method of lines. Last but not least, curvilinear space-time (or its equivalent method-of-lines form) moving grid simulation of Navier–Stokes equations can be pursued. The numerical performance between the tensor-product-based schemes and those for irregular 4D polytopes needs to be compared.

Acknowledgments

The early stage of this work was sponsored by the Office of Naval Research (ONR) through the award N00014-16-1-2735, and this work was also partially sponsored by the Army Research Office (ARO) through the Cooperative Agreement W911NF2420183. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the U.S. Government. Part of the hardware used in the computational studies is from the UMBC High Performance Computing Facility (HPCF). The facility is supported by the U.S. National Science Foundation through the MRI program (grant nos. CNS-0821258, CNS-1228778, OAC-1726023, and CNS-1920079) and the SCREMS program (grant no. DMS-0821311), with additional substantial support from the University of Maryland, Baltimore County (UMBC).

Data Availability

Data sets generated during the current study are available from the corresponding author on reasonable request.

Declarations

Competing interest The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Appendix A. Proof of Eq. (40)

The same nomenclature as that adopted in **Sect. 4.2.3** is used here as well as in [Appendix B](#). Since Gauss–Legendre quadrature points are used as solution points in STFR, and 1D Lagrange polynomials are generated based on those points, we have the following two fundamental properties:

- With $N + 1$ Gauss–Legendre quadrature points, a degree $2N + 1$ polynomial can be integrated exactly.
- The set of Lagrange polynomials based on $N + 1$ Gauss–Legendre quadrature points form a complete basis of the polynomial space P^N , and the bases are orthogonal to each other.

Take $\phi_{i_1}^H(\xi)$, $i_1 = 1, \dots, k^H + 1$, a set of degree k^H Lagrange polynomials, as an example. They are based on $k^H + 1$ Gauss–Legendre quadrature points $\xi_{j_1}^H$, and those points can integrate a degree $2k^H + 1$ polynomial exactly. Therefore, the product of any two bases $\phi_{i_1}^H(\xi)\phi_{j_1}^H(\xi)$, a degree $2k^H$ polynomial, can be integrated exactly with the $k^H + 1$ Gauss–Legendre quadrature points. Moreover, $\phi_{i_1}^H(\xi_{j_1}^H)$ equals to 1 when $i_1 = j_1$, and equals to 0 when $i_1 \neq j_1$. For any two bases $\phi_{i_1}^H(\xi)$ and $\phi_{j_1}^H(\xi)$, we then have their inner product calculated as

$$\begin{aligned} \int_{-1}^1 \phi_{i_1}^H(\xi)\phi_{j_1}^H(\xi)d\xi &= \sum_{k_1=1}^{k^H+1} \phi_{i_1}^H(\xi_{k_1}^H)\phi_{j_1}^H(\xi_{k_1}^H)\omega_{s,k_1}^H \\ &= \phi_{i_1}^H(\xi_{j_1}^H)\omega_{s,j_1}^H \\ &= \begin{cases} \omega_{s,i_1}^H & \text{if } i_1 = j_1 \\ 0 & \text{if } i_1 \neq j_1 \end{cases}. \end{aligned} \quad (\text{A.1})$$

The same procedures apply to $\phi_{i_1}^L(\xi)$, $i_1 = 1, \dots, k^L + 1$, a set of degree k^L Lagrange polynomials. Now we check the product of $\phi_{i_1}^H(\xi)\phi_{j_1}^L(\xi)$. It is a degree $(k^H + k^L)$ polynomial, and can be integrated exactly with the $k^H + 1$ Gauss–Legendre quadrature points $\xi_{j_1}^H$. Therefore, we have the following formula

$$\int_{-1}^1 \phi_{i_1}^H(\xi)\phi_{j_1}^L(\xi)d\xi = \sum_{k_1=1}^{k^H+1} \phi_{i_1}^H(\xi_{k_1}^H)\phi_{j_1}^L(\xi_{k_1}^H)\omega_{s,k_1}^H = \phi_{j_1}^L(\xi_{i_1}^H)\omega_{s,i_1}^H. \quad (\text{A.2})$$

The left side of Eq. (40) can then be written as

$$\begin{aligned} &\iiint \sum_{i_1=1}^{k^L+1} \sum_{i_2=1}^{k^L+1} \sum_{i_3=1}^{m^L+1} Q_{i_1,i_2,i_3}^L L_{i_1,i_2,i_3}^L(\boldsymbol{\xi}) L_{j_1,j_2,j_3}^L(\boldsymbol{\xi}) d\boldsymbol{\xi} \\ &= \sum_{i_1=1}^{k^L+1} \sum_{i_2=1}^{k^L+1} \sum_{i_3=1}^{m^L+1} Q_{i_1,i_2,i_3}^L \int_{-1}^1 \phi_{i_1}^L(\xi)\phi_{j_1}^L(\xi)d\xi \int_{-1}^1 \phi_{i_2}^L(\eta)\phi_{j_2}^L(\eta)d\eta \int_{-1}^1 \phi_{i_3}^L(\tau)\phi_{j_3}^L(\tau)d\tau \\ &= \sum_{i_1=1}^{k^L+1} \sum_{i_2=1}^{k^L+1} \sum_{i_3=1}^{m^L+1} Q_{i_1,i_2,i_3}^L \phi_{i_1}^L(\xi_{j_1}^L)\phi_{i_2}^L(\eta_{j_2}^L)\phi_{i_3}^L(\tau_{j_3}^L)\omega_{s,j_1}^L\omega_{s,j_2}^L\omega_{t,j_3}^L \\ &= Q_{j_1,j_2,j_3}^L \omega_{s,j_1}^L \omega_{s,j_2}^L \omega_{t,j_3}^L. \end{aligned} \quad (\text{A.3})$$

The right side of Eq. (40) is calculated from

$$\begin{aligned}
& \iiint \sum_{i_1=1}^{k^H+1} \sum_{i_2=1}^{k^H+1} \sum_{i_3=1}^{m^H+1} Q_{i_1,i_2,i_3}^H L_{i_1,i_2,i_3}^H(\boldsymbol{\xi}) L_{j_1,j_2,j_3}^L(\boldsymbol{\xi}) d\boldsymbol{\xi} \\
&= \sum_{i_1=1}^{k^H+1} \sum_{i_2=1}^{k^H+1} \sum_{i_3=1}^{m^H+1} Q_{i_1,i_2,i_3}^H \int_{-1}^1 \phi_{i_1}^H(\xi) \phi_{j_1}^L(\xi) d\xi \int_{-1}^1 \phi_{i_2}^H(\eta) \phi_{j_2}^L(\eta) d\eta \int_{-1}^1 \phi_{i_3}^H(\tau) \phi_{j_3}^L(\tau) d\tau \\
&= \sum_{i_1=1}^{k^H+1} \sum_{i_2=1}^{k^H+1} \sum_{i_3=1}^{m^H+1} Q_{i_1,i_2,i_3}^H \phi_{j_1}^L(\xi_{i_1}^H) \phi_{j_2}^L(\eta_{i_2}^H) \phi_{j_3}^L(\tau_{i_3}^H) \omega_{s,i_1}^H \omega_{s,i_2}^H \omega_{t,i_3}^H.
\end{aligned} \tag{A.4}$$

On combining Eq. (A.3) and Eq. (A.4), we have Eq. (40).

Appendix B. Proof of Eq. (41) and projection-based filtering strength discussion

According to the basic principle of interpolation with Lagrange polynomials, $Q^L(\boldsymbol{\xi}) \in \mathbb{Q}^{k^L}(\xi, \eta) \otimes P^{m^L}(\tau)$ can be constructed using the basis functions $\phi_{i_1}^H(\xi)$, $\phi_{j_1}^H(\eta)$ and $\phi_{k_1}^H(\tau)$ of the polynomial space $\mathbb{Q}^{k^H}(\xi, \eta) \otimes P^{m^H}(\tau)$ as

$$Q'^L(\boldsymbol{\xi}) = \sum_{i_1=1}^{k^H+1} \sum_{i_2=1}^{k^H+1} \sum_{i_3=1}^{m^H+1} Q^L(\xi_{i_1,i_2,i_3}^H) \phi_{i_1}^H(\xi) \phi_{i_2}^H(\eta) \phi_{i_3}^H(\tau). \tag{B.1}$$

According to Eq. (38), we have

$$Q^L(\xi_{j_1,j_2,j_3}^H) = \sum_{i_1=1}^{k^L+1} \sum_{i_2=1}^{k^L+1} \sum_{i_3=1}^{m^L+1} Q_{i_1,i_2,i_3}^L \phi_{i_1}^L(\xi_{j_1}^H) \phi_{i_2}^L(\eta_{j_2}^H) \phi_{i_3}^L(\tau_{j_3}^H).$$

Therefore, Eq. (B.1) can be written as

$$\begin{aligned}
Q'^L(\boldsymbol{\xi}) &= \sum_{j_1=1}^{k^H+1} \sum_{j_2=1}^{k^H+1} \sum_{j_3=1}^{m^H+1} \sum_{i_1=1}^{k^L+1} \sum_{i_2=1}^{k^L+1} \sum_{i_3=1}^{m^L+1} Q_{i_1,i_2,i_3}^L \phi_{i_1}^L(\xi_{j_1}^H) \phi_{i_2}^L(\eta_{j_2}^H) \phi_{i_3}^L(\tau_{j_3}^H) \phi_{j_1}^H(\xi) \phi_{j_2}^H(\eta) \phi_{j_3}^H(\tau) \\
&= \sum_{j_1=1}^{k^H+1} \sum_{i_1=1}^{k^L+1} \left(\phi_{i_1}^L(\xi_{j_1}^H) \phi_{j_1}^H(\xi) \sum_{j_2=1}^{k^H+1} \sum_{i_2=1}^{k^L+1} \left(\phi_{i_2}^L(\eta_{j_2}^H) \phi_{j_2}^H(\eta) \sum_{j_3=1}^{m^H+1} \sum_{i_3=1}^{m^L+1} Q_{i_1,i_2,i_3}^L \phi_{i_3}^L(\tau_{j_3}^H) \phi_{j_3}^H(\tau) \right) \right).
\end{aligned} \tag{B.2}$$

Since $k^L < k^H$ and $m^L < m^H$, any Lagrange polynomial $\phi_i^L(*)$ can be expressed by the Lagrange polynomial set $\phi_j^H(*)$. Take $\phi_{i_3}^L(\tau)$ and $\phi_{j_3}^H(\tau)$ as an example. We have

$$\phi_{i_3}^L(\tau) = \sum_{j_3=1}^{m^H+1} \phi_{i_3}^L(\tau_{j_3}^H) \phi_{j_3}^H(\tau).$$

Thus, we further have that

$$\begin{aligned}
\sum_{j_3=1}^{m^H+1} \sum_{i_3=1}^{m^L+1} Q_{i_1,i_2,i_3}^L \phi_{i_3}^L(\tau_{j_3}^H) \phi_{j_3}^H(\tau) &= \sum_{i_3=1}^{m^L+1} Q_{i_1,i_2,i_3}^L \left(\sum_{j_3=1}^{m^H+1} \phi_{i_3}^L(\tau_{j_3}^H) \phi_{j_3}^H(\tau) \right) \\
&= \sum_{i_3=1}^{m^L+1} Q_{i_1,i_2,i_3}^L \phi_{i_3}^L(\tau).
\end{aligned}$$

Therefore, Eq. (B.2) can be simplified as follows

$$\begin{aligned}
Q'^L(\xi) &= \sum_{j_1=1}^{k^H+1} \sum_{i_1=1}^{k^L+1} \left(\phi_{i_1}^L(\xi_{j_1}^H) \phi_{j_1}^H(\xi) \sum_{j_2=1}^{k^H+1} \sum_{i_2=1}^{k^L+1} \left(\phi_{i_2}^L(\eta_{j_2}^H) \phi_{j_2}^H(\eta) \sum_{i_3=1}^{m^L+1} Q_{i_1, i_2, i_3}^L \phi_{i_3}^L(\tau) \right) \right) \\
&= \sum_{j_1=1}^{k^H+1} \sum_{i_1=1}^{k^L+1} \left(\phi_{i_1}^L(\xi_{j_1}^H) \phi_{j_1}^H(\xi) \sum_{i_3=1}^{m^L+1} \phi_{i_3}^L(\tau) \left(\sum_{i_2=1}^{k^L+1} Q_{i_1, i_2, i_3}^L \left(\sum_{j_2=1}^{k^H+1} \phi_{i_2}^L(\eta_{j_2}^H) \phi_{j_2}^H(\eta) \right) \right) \right) \\
&= \sum_{j_1=1}^{k^H+1} \sum_{i_1=1}^{k^L+1} \left(\phi_{i_1}^L(\xi_{j_1}^H) \phi_{j_1}^H(\xi) \sum_{i_3=1}^{m^L+1} \sum_{i_2=1}^{k^L+1} Q_{i_1, i_2, i_3}^L \phi_{i_3}^L(\tau) \phi_{i_2}^L(\eta) \right) \\
&= \sum_{i_3=1}^{m^L+1} \sum_{i_2=1}^{k^L+1} \phi_{i_3}^L(\tau) \phi_{i_2}^L(\eta) \left(\sum_{i_1=1}^{k^L+1} Q_{i_1, i_2, i_3}^L \left(\sum_{j_1=1}^{k^H+1} \phi_{i_1}^L(\xi_{j_1}^H) \phi_{j_1}^H(\xi) \right) \right) \\
&= \sum_{i_3=1}^{m^L+1} \sum_{i_2=1}^{k^L+1} \sum_{i_1=1}^{k^L+1} Q_{i_1, i_2, i_3}^L \phi_{i_3}^L(\tau) \phi_{i_2}^L(\eta) \phi_{i_1}^L(\xi) \\
&= Q^L(\xi).
\end{aligned} \tag{B.3}$$

This proves Eq. (41).

Now the difference $\Delta f(\xi) = Q^H(\xi) - Q^L(\xi) \in \mathbb{Q}^{k^H}(\xi, \eta) \otimes P^{m^H}(\tau)$ can be expressed as

$$\begin{aligned}
\Delta f(\xi) &= \sum_{i_1=1}^{k^H+1} \sum_{i_2=1}^{k^H+1} \sum_{i_3=1}^{m^H+1} (Q^H(\xi_{i_1, i_2, i_3}^H) - Q^L(\xi_{i_1, i_2, i_3}^H)) \phi_{i_1}^H(\xi) \phi_{i_2}^H(\eta) \phi_{i_3}^H(\tau) \\
&= \sum_{i_1=1}^{k^H+1} \sum_{i_2=1}^{k^H+1} \sum_{i_3=1}^{m^H+1} \Delta f_{i_1, i_2, i_3}^H \phi_{i_1}^H(\xi) \phi_{i_2}^H(\eta) \phi_{i_3}^H(\tau).
\end{aligned} \tag{B.4}$$

Thus, its energy E_g can be evaluated from

$$E_g = \iiint \Delta f^2(\xi) d\xi = \sum_{i_1=1}^{k^H+1} \sum_{i_2=1}^{k^H+1} \sum_{i_3=1}^{m^H+1} \Delta f_{i_1, i_2, i_3}^{H,2} \omega_{s, i_1}^H \omega_{s, i_2}^H \omega_{t, i_3}^H. \tag{B.5}$$

Similarly, based on Eq. (42), the difference between $\bar{Q}^H(\xi)$ and $Q^L(\xi)$ is

$$\Delta \bar{f}(\xi) = \theta \sum_{i_1=1}^{k^H+1} \sum_{i_2=1}^{k^H+1} \sum_{i_3=1}^{m^H+1} \Delta f_{i_1, i_2, i_3}^H \phi_{i_1}^H(\xi) \phi_{i_2}^H(\eta) \phi_{i_3}^H(\tau). \tag{B.6}$$

Therefore, following Eq. (B.5), its energy $\bar{E}_g$ is

$$\bar{E}_g = \iiint \Delta \bar{f}^2(\xi) d\xi = \theta^2 \sum_{i_1=1}^{k^H+1} \sum_{i_2=1}^{k^H+1} \sum_{i_3=1}^{m^H+1} \Delta f_{i_1, i_2, i_3}^{H,2} \omega_{s, i_1}^H \omega_{s, i_2}^H \omega_{t, i_3}^H = \theta^2 E_g. \tag{B.7}$$

It is clear that a $(1 - \theta^2)$ portion of E_g is exactly filtered from $Q^H(\xi)$ with Eq. (42).

References

- [1] Z. Wang, K. Fidkowski, R. Abgrall, F. Bassi, D. Caraeni, A. Cary, H. Deconinck, R. Hartmann, K. Hillewaert, H. Huynh, N. Kroll, G. May, P.-O. Persson, B. von Leer, M. Visbal, High-order cfd methods: current status and perspective, Int. J. Numer. Meth. Fluids 72 (2013) 811–845.

- [2] J. Donea, S. Giuliani, J. Halleux, [An arbitrary lagrangian-eulerian finite element method for transient dynamic fluid-structure interactions](#), Computer Methods in Applied Mechanics and Engineering 33 (1) (1982) 689–723. doi:[https://doi.org/10.1016/0045-7825\(82\)90128-1](https://doi.org/10.1016/0045-7825(82)90128-1).
URL <https://www.sciencedirect.com/science/article/pii/0045782582901281>
- [3] C. Farhat, P. Geuzaine, [Design and analysis of robust ale time-integrators for the solution of unsteady flow problems on moving grids](#), Computer Methods in Applied Mechanics and Engineering 193 (39) (2004) 4073–4095, the Arbitrary Lagrangian-Eulerian Formulation. doi:<https://doi.org/10.1016/j.cma.2003.09.027>.
URL <https://www.sciencedirect.com/science/article/pii/S0045782504002130>
- [4] P.-O. Persson, J. Bonet, J. Peraire, [Discontinuous galerkin solution of the navier–stokes equations on deformable domains](#), Computer Methods in Applied Mechanics and Engineering 198 (17) (2009) 1585–1595. doi:<https://doi.org/10.1016/j.cma.2009.01.012>.
URL <https://www.sciencedirect.com/science/article/pii/S0045782509000450>
- [5] D. J. Mavriplis, C. R. Nastase, [On the geometric conservation law for high-order discontinuous galerkin discretizations on dynamically deforming meshes](#), Journal of Computational Physics 230 (11) (2011) 4285–4300, special issue High Order Methods for CFD Problems. doi:<https://doi.org/10.1016/j.jcp.2011.01.022>.
URL <https://www.sciencedirect.com/science/article/pii/S0021999111000477>
- [6] M. Yu, Z. Wang, H. Hu, [A high-order spectral difference method for unstructured dynamic grids](#), Computers & Fluids 48 (1) (2011) 84–97. doi:<https://doi.org/10.1016/j.compfluid.2011.03.015>.
URL <https://www.sciencedirect.com/science/article/pii/S0045793011001162>
- [7] C. Liang, K. Miyaji, B. Zhang, [An efficient correction procedure via reconstruction for simulation of viscous flow on moving and deforming domains](#), Journal of Computational Physics 256 (2014) 55–68. doi:<https://doi.org/10.1016/j.jcp.2013.08.046>.
URL <https://www.sciencedirect.com/science/article/pii/S0021999113005901>
- [8] D. A. Kopriva, A. R. Winters, M. Bohm, G. J. Gassner, [A provably stable discontinuous galerkin spectral element approximation for moving hexahedral meshes](#), Computers & Fluids 139 (2016) 148–160, 13th USNCCM International Symposium of High-Order Methods for Computational Fluid Dynamics - A special issue dedicated to the 60th birthday of Professor David Kopriva. doi:<https://doi.org/10.1016/j.compfluid.2016.05.023>.
URL <https://www.sciencedirect.com/science/article/pii/S0045793016301700>
- [9] M. Yu, L. Wang, [A high-order flux reconstruction/correction procedure via reconstruction formulation for unsteady incompressible flow on unstructured moving grids](#), Computers & Fluids 139 (2016) 161–173, 13th USNCCM International Symposium of High-Order Methods for Computational Fluid Dynamics - A special issue dedicated to the 60th birthday of Professor David Kopriva. doi:<https://doi.org/10.1016/j.compfluid.2016.05.028>.
URL <https://www.sciencedirect.com/science/article/pii/S004579301630175X>
- [10] Y. Abe, T. Haga, T. Nonomura, K. Fujii, [Conservative high-order flux-reconstruction schemes on moving and deforming grids](#), Computers & Fluids 139 (2016) 2–16, 13th USNCCM International Symposium of High-Order Methods for Computational Fluid Dynamics - A special issue dedicated to the 60th birthday of Professor David Kopriva. doi:<https://doi.org/10.1016/j.compfluid.2016.03.024>.
URL <https://www.sciencedirect.com/science/article/pii/S0045793016300883>
- [11] J. van der Vegt, H. van der Ven, [Space–time discontinuous galerkin finite element method with dynamic grid motion for inviscid compressible flows: I. general formulation](#), Journal of Computational Physics 182 (2) (2002) 546–585. doi:<https://doi.org/10.1006/jcph.2002.7185>.
URL <https://www.sciencedirect.com/science/article/pii/S0021999102971858>

- [12] C. Klaij, J. van der Vegt, H. van der Ven, [Space-time discontinuous galerkin method for the compressible navier-stokes equations](#), Journal of Computational Physics 217 (2) (2006) 589–611. doi:<https://doi.org/10.1016/j.jcp.2006.01.018>.
URL <https://www.sciencedirect.com/science/article/pii/S0021999106000246>
- [13] S. Petersen, C. Farhat, R. Tezaur, A space-time discontinuous galerkin method for the solution of the wave equation in the time domain, Int. J. Numer. Meth. Engng 78 (2009) 275–295.
- [14] M. Feistauer, J. Česenek, Space-time discontinuous galerkin finite element method for convection-diffusion problems and compressible flow, in: I. Dimov, S. Dimova, N. Kolkovska (Eds.), Numerical Methods and Applications, Springer Berlin Heidelberg, Berlin, Heidelberg, 2011, pp. 1–13.
- [15] L. Wang, P.-O. Persson, [A high-order discontinuous galerkin method with unstructured space-time meshes for two-dimensional compressible flows on domains with large deformations](#), Computers & Fluids 118 (2015) 53–68. doi:<https://doi.org/10.1016/j.compfluid.2015.05.026>.
URL <https://www.sciencedirect.com/science/article/pii/S0045793015001814>
- [16] A. Corrigan, A. D. Kercher, D. A. Kessler, A moving discontinuous galerkin finite element method for flows with interfaces, Int. J. Numer. Methods Fluids 89 (2019) 362–406.
- [17] H. Luo, G. Absillis, R. Nourgaliev, [A moving discontinuous galerkin finite element method with interface condition enforcement for compressible flows](#), Journal of Computational Physics 445 (2021) 110618. doi:<https://doi.org/10.1016/j.jcp.2021.110618>.
URL <https://www.sciencedirect.com/science/article/pii/S0021999121005131>
- [18] T. J. Hughes, G. M. Hulbert, [Space-time finite element methods for elastodynamics: Formulations and error estimates](#), Computer Methods in Applied Mechanics and Engineering 66 (3) (1988) 339–363. doi:[https://doi.org/10.1016/0045-7825\(88\)90006-0](https://doi.org/10.1016/0045-7825(88)90006-0).
URL <https://www.sciencedirect.com/science/article/pii/0045782588900060>
- [19] T. Tezduyar, M. Behr, J. Liou, [A new strategy for finite element computations involving moving boundaries and interfaces—the deforming-spatial-domain/space-time procedure: I. the concept and the preliminary numerical tests](#), Computer Methods in Applied Mechanics and Engineering 94 (3) (1992) 339–351. doi:[https://doi.org/10.1016/0045-7825\(92\)90059-S](https://doi.org/10.1016/0045-7825(92)90059-S).
URL <https://www.sciencedirect.com/science/article/pii/004578259290059S>
- [20] U. Köcher, M. Bause, Variational space-time methods for the wave equation, Journal of Scientific Computing 61 (2014) 424–453.
- [21] J. C. Butcher, Numerical Methods for Ordinary Differential Equations, Wiley, Chichester, 2002.
- [22] H. Bijl, M. H. Carpenter, V. N. Vatsa, C. A. Kennedy, [Implicit time integration schemes for the unsteady compressible navier-stokes equations: Laminar flow](#), Journal of Computational Physics 179 (1) (2002) 313–329. doi:<https://doi.org/10.1006/jcph.2002.7059>.
URL <https://www.sciencedirect.com/science/article/pii/S0021999102970592>
- [23] M. Yu, Z. Wang, Y. Liu, [On the accuracy and efficiency of discontinuous galerkin, spectral difference and correction procedure via reconstruction methods](#), Journal of Computational Physics 259 (2014) 70–95. doi:<https://doi.org/10.1016/j.jcp.2013.11.023>.
URL <https://www.sciencedirect.com/science/article/pii/S0021999113007857>
- [24] Y. Abe, T. Haga, T. Nonomura, K. Fujii, [On the freestream preservation of high-order conservative flux-reconstruction schemes](#), Journal of Computational Physics 281 (2015) 28–54. doi:<https://doi.org/10.1016/j.jcp.2014.10.011>.
URL <https://www.sciencedirect.com/science/article/pii/S0021999114006937>

- [25] A. Cicchino, D. C. Del Rey Fernández, S. Nadarajah, J. Chan, M. H. Carpenter, [Provably stable flux reconstruction high-order methods on curvilinear elements](#), Journal of Computational Physics 463 (2022) 111259. doi:<https://doi.org/10.1016/j.jcp.2022.111259>. URL <https://www.sciencedirect.com/science/article/pii/S0021999122003217>
- [26] H. T. Huynh, High-order space-time methods for conservation laws, in: 21st AIAA Computational Fluid Dynamics Conference, 2013, p. 2432.
- [27] H. T. Huynh, Discontinuous galerkin and related methods for ode, Journal of Scientific Computing 96 (2023) 51.
- [28] B. L. Ehle, A-stable methods and Pade approximations to the exponential, SIAM J.Math. Anal. 4 (1973) 671–680.
- [29] C. L. Bottasso, A new look at finite elements in time: a variational interpretation of runge-kutta methods, Applied Numerical Mathematics 25 (4) (1997) 355–368.
- [30] W. Tang, Y. Sun, Time finite element methods: A unified framework for numerical discretizations of odes, Applied Mathematics and Computation 219 (4) (2012) 2158–2179.
- [31] H. T. Huynh, A flux reconstruction approach to high-order schemes including discontinuous galerkin methods, in: 18th AIAA Computational Fluid Dynamics Conference, 2007, p. 4079.
- [32] H. T. Huynh, A reconstruction approach to high-order schemes including discontinuous galerkin for diffusion, in: 47th AIAA Aerospace Sciences Meeting including The New Horizons Forum and Aerospace Exposition, 2009, p. 403.
- [33] B. Cockburn, C.-W. Shu, TVB Runge–Kutta local projection discontinuous Galerkin finite element method for conservation laws II: general framework, Mathematics of Computation 52 (1989) 411–435.
- [34] F. Bassi, S. Rebay, A high-order accurate discontinuous finite element method for the numerical solution of the compressible navier–stokes equations, Journal of computational physics 131 (2) (1997) 267–279.
- [35] J. S. Hesthaven, T. Warburton, Nodal Discontinuous Galerkin methods: algorithms, analysis, and applications, Springer, New York, 2008.
- [36] D. A. Kopriva, J. H. Kolas, A conservative staggered-grid Chebyshev multidomain method for compressible flows, Journal of Computational Physics 125 (1996) 244–261.
- [37] Y. Liu, M. Vinokur and Z. J. Wang, Spectral difference method for unstructured grids I: Basic formulation, Journal of Computational Physics 216 (2006) 780–801.
- [38] Z. J. Wang, Spectral (finite) volume method for conservation laws on unstructured grids: basic formulation, Journal of Computational Physics 178 (2002) 210–251.
- [39] Z. Wang, H. Gao, A unifying lifting collocation penalty formulation including the discontinuous galerkin, spectral volume/difference methods for conservation laws on mixed grids, Journal of Computational Physics 228 (21) (2009) 8161–8186.
- [40] P. E. Vincent, P. Castonguay, A. Jameson, A new class of high-order energy stable flux reconstruction schemes, Journal of Scientific Computing 47 (1) (2011) 50–72.
- [41] H. T. Huynh, Z. J. Wang, P. E. Vincent, High-order methods for computational fluid dynamics: A brief review of compact differential formulations on unstructured grids, Computers & Fluids 98 (2014) 209–220.
- [42] M. Yu, Nodal space-time flux reconstruction methods for conservation laws, in: the 23rd AIAA Computational Fluid Dynamics Conference, Denver, Colorado, 2017, p. 3095.

- [43] M. Yu, [High-order space-time flux reconstruction methods for moving domain simulation](#), in: the 12th International Conference on Computational Fluid Dynamics (ICCFD12), Kobe, Japan, 2024. doi:<https://doi.org/10.48550/arXiv.2409.14005>. URL <https://arxiv.org/abs/2409.14005>
- [44] V. Rusanov, [The calculation of the interaction of non-stationary shock waves and obstacles](#), USSR Computational Mathematics and Mathematical Physics 1 (2) (1962) 304–320. doi:[https://doi.org/10.1016/0041-5553\(62\)90062-9](https://doi.org/10.1016/0041-5553(62)90062-9). URL <https://www.sciencedirect.com/science/article/pii/0041555362900629>
- [45] S. Gottlieb, C.-W. Shu, E. Tadmor, [Strong stability-preserving high-order time discretization methods](#), SIAM Review 43 (1) (2001) 89–112. arXiv:<https://doi.org/10.1137/S003614450036757X>, doi:<https://doi.org/10.1137/S003614450036757X>. URL <https://doi.org/10.1137/S003614450036757X>
- [46] L. Wang, M. Yu, Comparison of ROW, ESDIRK, and BDF2 for Unsteady Flows with the High-Order Flux Reconstruction Formulation, Journal of Scientific Computing 83 (2020) 1–27.
- [47] D. A. Kopriva, Metric identities and the discontinuous spectral element method on curvilinear meshes, Journal of Scientific Computing 26 (2006) 301–327.
- [48] M. Behr, Simplex space–time meshes in finite element simulations, Int. J. Numer. Methods Fluids 57 (2008) 1421–1434.
- [49] H. Nishikawa, E. Padway, An adaptive space-time edge-based solver for two-dimensional unsteady inviscid flows, in: AIAA AVIATION 2020 FORUM, 2020, p. 3024.
- [50] C. V. Frontin, G. S. Walters, F. D. Witherden, C. W. Lee, D. M. Williams, D. L. Darmofal, [Foundations of space-time finite element methods: Polytopes, interpolation, and integration](#), Applied Numerical Mathematics 166 (2021) 92–113. doi:<https://doi.org/10.1016/j.apnum.2021.03.019>. URL <https://www.sciencedirect.com/science/article/pii/S0168927421000994>
- [51] H. You, J. Kim, C. Kim, [Deneb: An open-source high-performance multi-physical flow solver based on high-order drm-dg method](#), Computer Physics Communications 286 (2023) 108672. doi:<https://doi.org/10.1016/j.cpc.2023.108672>. URL <https://www.sciencedirect.com/science/article/pii/S0010465523000176>
- [52] M. Yu, Z. Wang, On the connection between the correction and weighting functions in the correction procedure via reconstruction method, Journal of Scientific Computing 54 (2013) 227–244.
- [53] D. J. Mavriplis, C. R. Nastase, On the Geometric Conservation Law for High-order Discontinuous Galerkin Discretizations on Dynamically Deforming Meshes, in: 46th AIAA Aerospace Sciences Meeting and Exhibit, 2008, p. 778.
- [54] N. A. Wukie, K. Fidkowski, P.-O. Persson, Z. J. Wang, High-fidelity cfd verification workshop 2024: Mesh motion, in: AIAA SCITECH 2023 Forum, 2023, p. 1243.