

Discontinuous Galerkin spectral element method with hybrid sub-element order reduction for shock-capturing with Navier–Stokes equations

Fengrui Zhang, Yulia T. Peet^{ID *}

School for Engineering of Matter, Transport and Energy, Arizona State University, Tempe, AZ, 85287, USA

ARTICLE INFO

Keywords:

Compressible flows
Shock capturing
High-order methods
Discontinuous Galerkin methods

ABSTRACT

The current study presents a high-order methodology for the simulation of three-dimensional compressible viscous flows with shocks in complex geometries. The method is developed based on the framework of a split-form discontinuous Galerkin spectral element method (DGSEM) with summation-by-parts (SBP) operators. The Bassi and Rebay (Bassi and Rebay, 1997) (BR1) scheme is employed for the discretization of the viscous terms. For shock capturing, a hybrid sub-element order reduction methodology is developed which is based on a mixed functional space that blends high-order polynomial basis functions with piecewise-constant functions supported on sub-element volumes. An amount of blending is determined based on a modified Ducros indicator which has excellent shock detecting capabilities in viscous turbulent flows. The performance of the methodology is demonstrated on the example of eight test cases, featuring 1D, 2D and 3D inviscid and viscous flows with and without shocks.

1. Introduction

In the computational fluid dynamics community, high-fidelity simulation of high-speed turbulent flows, especially in the presence of shocks and discontinuities, remains to be a challenge. For resolving turbulence, high-order discretizations, especially those involving rapidly convergent spectral or polynomial approximation spaces, are superior to low-order schemes, even with the same number of degrees of freedoms (DOFs), which is one of the major reasons for the pursuit of high-order methods [1]. The discontinuous Galerkin (DG) methodology, first introduced by Reed and Hill [2] in 1973, has gained a wide popularity as a robust high-order methodology for solution of compressible flow equations. It can be considered as a variant of a traditional high-order finite element method that rigorously addresses discontinuities on the element surfaces (See, e.g., [3] and references therein). The solution can theoretically reach an arbitrary order of accuracy in the absence of discontinuity by increasing the number of quadrature points within each DG element. Additionally, DG methods enjoy compact stencils for efficient parallelization, a flexibility in handling complex geometries through unstructured body-fitted mesh capabilities, and a possibility of an h/p adaptation [4].

For the discretization of the non-linear convective terms of the Navier–Stokes equations, there are two major challenges employing high-order DG methods. The first one is related to the aliasing-related instability due to an under-resolution. Common techniques to overcome this issue consist of over-integration [5], filtering [6], and a split formulation of nonlinear terms [7]. Among the three, over-integration has been widely used for the incompressible flows [8]; however, for compressible flows, the theoretical foundation does not hold [9], and no evidence shows that over-integration performs well for general compressible flows. For the filtering, although some successful applications of this stabilization technique have been demonstrated [10], there are tunable parameters to be determined; additionally, by the nature of filtering, it inevitably leads to an energy loss. Therefore, in this paper, we choose to apply a split form of the non-linear terms to reduce the aliasing error [11]. At the same time, utilization of the summation-by-parts (SBP) operators in the discretization ensures conservation of the primary variables (conserved variables) [12]. With a proper choice of the split form, the discretization is able to obtain an energy-stable [13] or an entropy-stable [14] property. This framework was first built up in the finite-difference community [15], and further introduced to the DG community [7]. In [7], the authors illustrated that the split-form discretization is equivalent to the corresponding two-point flux. While high-order energy-stable and entropy-stable schemes in the context of DG were originally constructed for nodal bases that produce diagonal mass matrices and diagonal norm SBP operators [7], an extension of entropy-stable techniques to more general dense norm SBP operators within the modal bases was introduced in [16] for the Euler equations and extended in [17,18] to viscous flows and rarefied nonequilibrium gas flows, respectively. The method used in the current paper, however, is based

* Corresponding author.

E-mail addresses: fzhang82@asu.edu (F. Zhang), ypeet@asu.edu (Y.T. Peet).

upon the first class of approaches that utilize nodal DG schemes with split-form diagonal norm SBP operators built on tensor-product elements in the context of the spectral-element DG methods (DGSEM) [7].

Appropriate diagonal norm SBP operators are straightforward to construct on tensor product elements based on a DG-SEM discretization

Besides the aliasing-related instability, the second challenge that high-order discretizations face when applied to nonlinear convection problems is the Gibbs-type instability due to discontinuities. Two main strategies to regularize the solution in the presence of discontinuities include adding artificial dissipation that gives shocks a finite thickness and smoothens the steep gradients [19], and limiting the order of the solution in the vicinity of shocks to prevent the Gibbs oscillations [20]. While artificial dissipation techniques allow for a robust shock localization and a sub-element shock resolution in DG methods, the main drawback of this methodology is attributed to an adverse impact of the added diffusion terms on the time step restriction of explicit schemes, especially if the artificial viscosity coefficient is high, such as in a presence of strong shocks [21].

The second strategy that instead reduces the order of the solution in the vicinity of discontinuities rather than adds an explicit dissipation, is thus appealing. Early works on introducing low-order limiters to DG methods include down-selective p -adaptation [22], or replacing a high-order solution in a troubled element by its low-order counterpart, typically consisting of cell averages, potentially augmented by second-order total variation diminishing (TVD) slope limiters [23], or a higher-order WENO-type moment reconstruction [24]. A deficiency of this method is that the accuracy within an entire element is compromised, even though a shock can potentially be much thinner than the element size. To alleviate this deficiency, the so-called subcell limiting methods have been proposed. In the original methodology, first introduced in [25], a troubled element, where a shock can potentially exist, is divided into subcells (with the number of subcells typically equal [25] or higher [4] than the number of high-order degrees of freedom within the element), and a high-order DG approximation is abandoned in such elements in favor of a more robust low-order finite-volume (FV) scheme, which, due to an accompanied h -refinement, is presumed to compensate for an order reduction within the element. While this approach is powerful, it has a similar limitation of a global order reduction within an entire element containing a shock.

Yet more flexible, *hybrid* approaches, to a sub-element order reduction, can be traced to a development of the residual distribution schemes by Abgrall and Shu [26], where, instead of substituting a high-order flux by an h -refined low-order flux in the volume contribution of a DG element, the high-order and the low-order fluxes are combined, and their proportion in the blend is determined by the so-called blending coefficient $\alpha \in [0, 1]$, related to a shock indicator. This approach has been revisited in [27] who introduced a new shape of basis functions that are able to blend between continuous and discontinuous interpolation, and in [28] who proposed a combined approximation space consisting of polynomials and piecewise-constant functions to allow for a mixed high-/low-order approximation within the elements. Recently, a hybrid DG–subcell finite volume scheme has been developed for shock capturing within the context of Euler [29] and magnetohydrodynamics (MHD) [30] equations. Several other contributions to a development of hybrid subcell limiting techniques in DG methods include positivity preserving limiters in [31], invariant domain and structure preserving methods in [32], and a-posteriori flux corrections in [33].

The main contribution of the present paper is an extension of the hybrid shock-capturing scheme to the Navier–Stokes equations and to simulations of compressible viscous turbulent flows. Previous attempts employing hybrid high-/low-order discretizations focused predominantly on Euler equations [27–29, 31–36] (with the exception of [30, 33, 35] who also considered MHD problems). For the discretization of the viscous terms, we use a framework that introduces a lifting operator to transform the second-order derivative terms to the first-order derivatives, conceptualized in the BR1 scheme [37]. A special care is devoted to enabling the developed methodology for shock-involved viscous flow simulations on curvilinear meshes and attaining its formal high-order accuracy in curved geometries. We note that Gao et al. [38] applied a similar split-form DGSEM with the BR1 scheme and a subcell finite volume shock-capturing to the Navier–Stokes equations to study shock–boundary layer interactions in turbulent flows, but they did not implement a hybrid high-/low-order blending within the troubled elements and they did not consider curvilinear geometries.

A special attention should be devoted to a choice of a shock indicator function, which aims to identify the elements affected by a shock to activate a shock-capturing scheme and to control the amount of high-/low-order blending in hybrid schemes. In a majority of previous studies utilizing subcell limiting, a Persson-type shock indicator [39] is used [4, 27, 29, 30, 35]. Persson's indicator [39] is a modal indicator which computes the amount of energy in the highest resolved polynomial mode with respect to the total energy of the approximated solution within the element. High energy in the highest resolved mode, or a slow modal energy decay, is typically associated with steep solution gradients, which, in case of inviscid flows, points to discontinuities [39]. However, in viscous turbulent flows, high gradients can also form due to local shearing motions and vorticity fluctuations, as well as due to shocklets [40]. This means that Persson's indicator will excessively damp the strong vortical motions and turbulence-induced local structures, and not just the shocks. To alleviate this problem, a Ducros sensor [41] has been used successfully in the compressible turbulence simulations as a shock detector in conjunction with the artificial viscosity methods [42]. In this paper, we propose to employ a Ducros sensor to identify the shocked areas and to determine the amount of high-/low-order blending in our hybrid DG scheme for viscous compressible flow simulations. An original Ducros sensor [41] still has some deficiencies as applied to compressible turbulence, which is related to a high sensitivity of the sensor to weak dilatational motions in the absence of rotation and to an excessive damping of turbulent shocklets, as well as expansion waves (regions of positive dilatation) [40]. Gao et al. [38] pointed to these deficiencies and discussed some alternative solutions (as well as their drawbacks), including Pirozolli's corrections [43], none of which were eventually adopted and, instead, the authors have implemented a global spanwise filtering of the pressure field and used the Jameson's indicator [44] for their subcell finite volume scheme. We remark, as the authors of [38] indicated, that the global spanwise filtering only works for quasi-2D configurations and is not applicable to 3D geometries, such as, e.g., a flow over a sphere, considered in the current study. Instead, we propose to address the problems associated with the original Ducros sensor deficiencies by employing a modified Ducros sensor introduced by [40] in their simulations of compressible turbulence regulated by an artificial dissipation. The modified sensor has only two tunable parameters, whose values were held constant across all test cases considered in the current paper while yielding good results across all the problems. We remark that the problem of a frequent switching between high- and low-order schemes in the turbulence simulations regulated by a Ducros filter as observed in [38] does not occur in the current hybrid scheme, where, instead of a binary switch, a continuous blending between a high- and a low-order flux is ensured. To the authors' knowledge, the present study is the first documentation of the use of the modified Ducros sensor in conjunction with the subcell limiting approaches with the DG methods.

In order to preserve a variational framework on which the spectral element methods theory is built [45], we derive the blended high-/low-order DGSEM discretization methodology in a monolithic way, starting from a weak formulation of the combined procedure, as opposed to an a-posteriori adjustment of residuals between high-order DG and low-order FV schemes, as originally presented in [26] for unstructured DG methods and further in [29] for split-form DGSEM. To this end, we introduce a hybrid functional space which consists of a combination of high-order polynomials and low-order piecewise constant functions on the subcells, reminiscent of the approaches in [27, 28], except that we use the DGSEM metrics to define the subcells and the corresponding low-order functions on the subcells, consistent with [29]. Doing so allows us to reconcile the variational framework of the finite/spectral-element methods and a DG/FV-type blending presented in earlier works [29].

The paper is structured as follows. Section 2 presents the discretization of the Navier–Stokes equations. Starting from a weak formulation of the governing equations, we introduce a hybrid high-/low-order functional space that results in a blended discretization of the resulting terms in the equations. We then present the methods employed, respectively, for high-order convective fluxes, low-order convective fluxes, and viscous fluxes. Section 3 shows the verification and validation of the current method against eight test cases, including 1D, 2D and 3D problems that contain strong shocks and discontinuities, curvilinear geometries, and viscous compressible turbulence. Conclusions are drawn in Section 4.

2. Numerical methodology

2.1. Governing equations

In the current study, the governing equations are the compressible Navier–Stokes equations,

$$\frac{\partial \mathbf{U}}{\partial t} + \nabla \cdot \mathbf{F}(\mathbf{U}) = \nabla \cdot \mathbf{F}_V(\mathbf{U}, \mathbf{q}), \quad (1a)$$

$$\mathbf{q} = \nabla \mathbf{U}, \quad (1b)$$

where

$$\mathbf{U} = \begin{pmatrix} \rho \\ \rho \mathbf{u} \\ \rho e \end{pmatrix}, \quad \mathbf{F} = \begin{pmatrix} \rho \mathbf{u}^T \\ \rho \mathbf{u} \otimes \mathbf{I} p \\ \rho e \mathbf{u}^T + p \mathbf{u}^T \end{pmatrix}, \quad \mathbf{F}_V = \begin{pmatrix} 0 \\ \boldsymbol{\tau} \\ \boldsymbol{\tau} \cdot \mathbf{u} + \kappa \nabla T \end{pmatrix}, \quad (2)$$

where ρ is the density, \mathbf{u} is the velocity vector, T is the temperature, κ is the thermal conductivity, e is the total energy per unit mass, p is the pressure, \mathbf{I} is the identity matrix, \otimes denotes the tensor product operator, and the superscript “ T ” denotes a matrix transpose. To close the system, the ideal gas equation of state, and the relation between temperature T and the internal energy per unit mass $e_{\text{internal}} = e - |\mathbf{u}|^2/2$ are introduced,

$$p = \rho R T, \quad (3a)$$

$$e_{\text{internal}} = C_v T, \quad (3b)$$

where C_v is the specific heat at constant volume. In Eq. (2), \mathbf{F} is the convective flux, \mathbf{F}_V is the diffusive flux of the Navier–Stokes equations with

$$\boldsymbol{\tau} = \mu [\nabla \mathbf{u} + (\nabla \mathbf{u})^T] - \frac{2}{3} \mu (\nabla \cdot \mathbf{u}) \mathbf{I} \quad (4)$$

for a Newtonian fluid, μ is the dynamic viscosity which can be calculated using the Sutherland law,

$$\mu = \mu_{\text{ref}} \left(\frac{T}{T_{\text{ref}}} \right)^{3/2} \left(\frac{T_{\text{ref}} + T_s}{T + T_s} \right), \quad (5)$$

where $\mu_{\text{ref}} = 1.716 \times 10^{-5} \frac{\text{kg}}{\text{m}\cdot\text{s}}$, $T_{\text{ref}} = 273.15\text{K}$, $T_s = 110.4\text{K}$, for all the viscous test cases in the current study. Thermal conductivity is calculated through viscosity as $\kappa = C_p \mu / Pr$, with Pr being the Prandtl number, and C_p being the specific heat at constant pressure.

2.2. Weak form of the equations

In a discontinuous Galerkin spectral-element formulation, we decompose the computational domain Ω into N_E non-overlapping elements as

$$\Omega = \sum_{E=1}^{N_E} \Omega_E. \quad (6)$$

For each element, we transform the coordinates of Eq. (1) from the physical domain $\Omega_E(\mathbf{x})$, $\mathbf{x} = (x, y, z) = (x_1, x_2, x_3)$ into a reference domain $\hat{\Omega}(\xi) = [-1, 1]^3$, $\xi = (\xi, \eta, \zeta) = (\xi^1, \xi^2, \xi^3)$ (See, e.g., Ref. [46] for more details),

$$J \mathbf{U}_t + \nabla_{\xi} \cdot \tilde{\mathbf{F}}(\mathbf{U}) = \nabla_{\xi} \cdot \tilde{\mathbf{F}}_V(\mathbf{U}, \mathbf{q}), \quad (7a)$$

$$J \mathbf{q} = \tilde{\nabla}_{\xi} \mathbf{U}, \quad (7b)$$

where $\mathbf{U}_t = \partial \mathbf{U} / \partial t$, $\tilde{\mathbf{F}}^l = \mathbf{F} \cdot J \mathbf{a}^l$, $\tilde{\mathbf{F}}_V^l = \mathbf{F}_V \cdot J \mathbf{a}^l$, $l \in \{1, 2, 3\}$, are scaled contra-variant convective and diffusive fluxes, respectively, \mathbf{a}^l is the contra-variant basis vector, the notation

$$\tilde{\nabla}_{\xi} = \frac{\partial}{\partial \xi^l} \cdot J \mathbf{a}^l \quad (8)$$

is introduced, and J is the determinant of the Jacobian matrix,

$$J = \begin{vmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial x}{\partial \eta} & \frac{\partial x}{\partial \zeta} \\ \frac{\partial y}{\partial \xi} & \frac{\partial y}{\partial \eta} & \frac{\partial y}{\partial \zeta} \\ \frac{\partial z}{\partial \xi} & \frac{\partial z}{\partial \eta} & \frac{\partial z}{\partial \zeta} \end{vmatrix}. \quad (9)$$

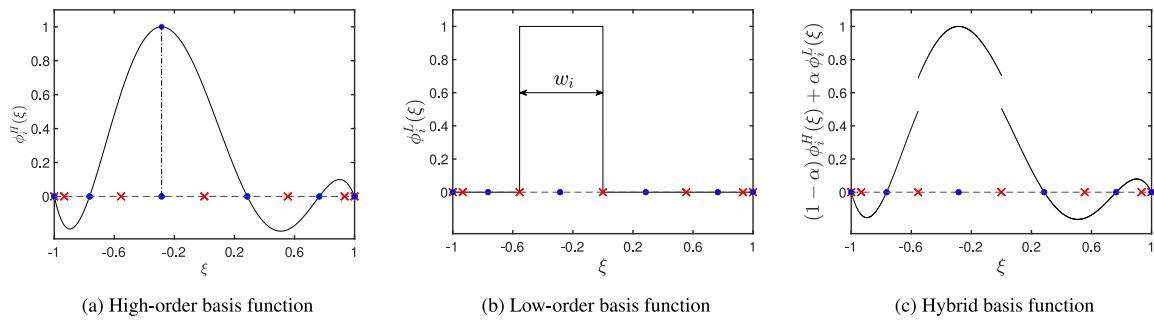


Fig. 1. Example of (a) high-order basis function, (b) low-order basis function, and (c) their combination (hybrid basis function) in 1D for $\alpha = 0.2, i = 3, N = 6$. Circles correspond to GLL points ξ_p , crosses to subcell interfaces $\xi_{(p-1,p)}$.

2.3. Hybrid basis functions

Within each element, we introduce the hybrid basis functions in the form of

$$\phi_{ijk}(\xi, \eta, \zeta) = (1 - \alpha)\phi_{ijk}^H(\xi, \eta, \zeta) + \alpha\phi_{ijk}^L(\xi, \eta, \zeta), \quad (10)$$

where $\phi_{ijk}^H(\xi, \eta, \zeta)$ is a high-order basis function, $\phi_{ijk}^L(\xi, \eta, \zeta)$ is a low-order basis function, and α is a blending coefficient, which will only be non-zero in the vicinity of shocks, so that the low-order approximation will only be activated locally, as required for shock-capturing. The procedure for determining the blending coefficient will be described in Section 2.5.3.

High-order basis function is defined as a tensor product of 1D Lagrange interpolating polynomials, $l_i(\xi)$, $l_j(\eta)$ and $l_k(\zeta)$, such that

$$\phi_{ijk}^H(\xi, \eta, \zeta) = l_i(\xi)l_j(\eta)l_k(\zeta), \quad i, j, k \in \{1, \dots, N\}, \quad (11)$$

N is the number of the collocation points. Here, the polynomials $l_i(\xi)$, $l_j(\eta)$ and $l_k(\zeta)$ are defined on a set of Gauss–Lobatto–Legendre (GLL) points (ξ_p, η_q, ζ_r) , $p, q, r \in \{1, \dots, N\}$, such that, e.g.,

$$l_i(\xi) = \prod_{\substack{1 \leq p \leq N \\ p \neq i}} \frac{\xi - \xi_p}{\xi_i - \xi_p}. \quad (12)$$

From Eq. (12), it is easy to see that the Lagrange interpolating polynomials obey the cardinality property, $l_i(\xi_p) = \delta_{ip}$, and analogously for the other directions.

Low-order basis function is defined as a tensor product of 1D piecewise-constant (step) functions $\chi_i(\xi)$, $\chi_j(\eta)$ and $\chi_k(\zeta)$, such that

$$\phi_{ijk}^L(\xi, \eta, \zeta) = \chi_i(\xi)\chi_j(\eta)\chi_k(\zeta), \quad i, j, k \in \{1, \dots, N\}, \quad (13)$$

where the 1D piecewise-constant functions are defined on the interval $[-1, 1]$ according to the GLL metric as

$$\chi_i(\xi) = \begin{cases} 1, & \xi - (\sum_{p=1}^{i-1} w_p - 1) \in [0, w_i] \\ 0, & \text{otherwise} \end{cases} \quad (14)$$

and analogously for the other directions, where w_p are the quadrature weights associated with the GLL integration, and, with some abuse of notation, $\sum_{p=1}^0 w_p = 0$ by definition. Fig. 1 illustrates an example of a high-order basis function, a low-order basis function, and their combination for some arbitrary values of i and α .

Accordingly, for each piecewise-constant function $\chi_i(\xi)$, we can define its region of compact support, i.e. the interval $\xi \in [\xi_{(i-1,i)}, \xi_{(i,i+1)}]$ within which it assumes a non-zero value. The left and right boundaries of the region of a function support (subcell interfaces) are given by

$$\begin{cases} \xi_{(i-1,i)} = \sum_{p=1}^{i-1} w_p - 1 \\ \xi_{(i,i+1)} = \sum_{p=1}^i w_p - 1, \end{cases} \quad (15)$$

such that $|\xi_{(i,i+1)} - \xi_{(i-1,i)}| = w_i$, with $\xi_{(0,1)} = -1$ and $\xi_{(N,N+1)} = 1$, and analogously for the other directions. Finally, with the tensor product, we can define a subcell $\hat{\Omega}_{\text{sub}(ijk)}$ of the element $\hat{\Omega}$ as a volumetric space where $\phi_{ijk}^L(\xi, \eta, \zeta)$ is non-zero:

$$\hat{\Omega}_{\text{sub}(ijk)} = [\xi_{(i-1,i)}, \xi_{(i,i+1)}] \times [\eta_{(j-1,j)}, \eta_{(j,j+1)}] \times [\zeta_{(k-1,k)}, \zeta_{(k,k+1)}], \quad (16)$$

The volume of this subcell is therefore equal to $|\hat{\Omega}_{\text{sub}(ijk)}| = w_i w_j w_k$ (in the reference space). Fig. 2 illustrates an example of such a subcell in two dimensions.

Taking an inner product of Eq. (7) with the hybrid test function $\phi_{ijk}(\xi, \eta, \zeta)$ of Eq. (10) over each element, the following local weak form is obtained,

$$(J \mathbf{U}_t, \phi_{ijk})_{\hat{\Omega}} + (\nabla_{\xi} \cdot \tilde{\mathbf{F}}, \phi_{ijk})_{\hat{\Omega}} = (\nabla_{\xi} \cdot \tilde{\mathbf{F}}_V, \phi_{ijk})_{\hat{\Omega}}, \quad (17a)$$

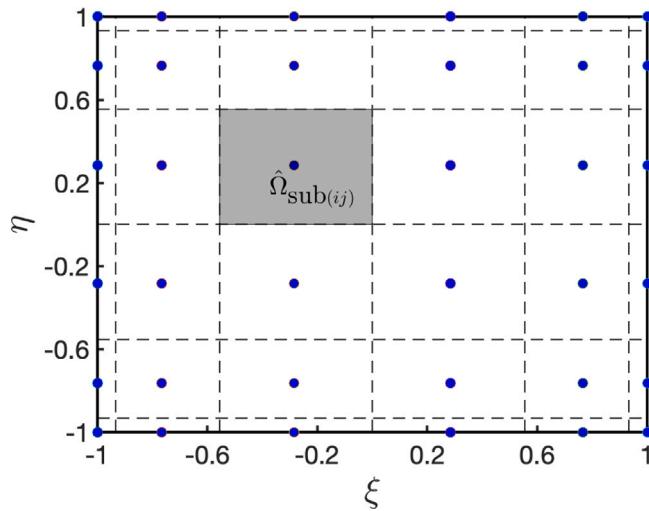


Fig. 2. An example of a subcell (shaded gray rectangle) in 2D for $i = 3, j = 4$. Circles correspond to GLL points (ξ_p, η_q) in 2D, $N = 6$.

$$(J \mathbf{q}, \phi_{ijk})_{\hat{\Omega}} = (\tilde{\nabla}_{\xi} \mathbf{U}, \phi_{ijk})_{\hat{\Omega}}, \quad (17b)$$

where the inner product is defined on a componentwise basis as

$$(\mathbf{f}, \phi_{ijk})_{\hat{\Omega}} = \int_{-1}^1 \int_{-1}^1 \int_{-1}^1 \mathbf{f}(\xi, \eta, \zeta) \phi_{ijk}(\xi, \eta, \zeta) d\xi d\eta d\zeta. \quad (18)$$

2.4. Temporal derivative term

We write the temporal derivative term in Eq. (17a) as

$$(J \mathbf{U}_t, \phi_{ijk})_{\hat{\Omega}} = (1 - \alpha) (J \mathbf{U}_t, \phi_{ijk}^H)_{\hat{\Omega}} + \alpha (J \mathbf{U}_t, \phi_{ijk}^L)_{\hat{\Omega}}. \quad (19)$$

Due to the cardinality property of high-order Lagrangian basis functions, the first inner product in the right-hand side of Eq. (19) reads

$$(J \mathbf{U}_t, \phi_{ijk}^H)_{\hat{\Omega}} = \sum_{p=1}^N \sum_{q=1}^N \sum_{r=1}^N [J \mathbf{U}_t \phi_{ijk}^H](\xi_p, \eta_q, \zeta_r) w_p w_q w_r = J_{ijk} w_i w_j w_k (\mathbf{U}_t)_{ijk}. \quad (20)$$

Similarly, due to a compact support of the low-order basis function $\phi_{ijk}^L(\xi, \eta, \zeta)$ in $\hat{\Omega}_{\text{sub}(ijk)}$, which is a local subcell of the element $\hat{\Omega}$ in the vicinity of the collocation point (ξ_i, η_j, ζ_k) (See Eq. (16)), the second inner product in the right-hand side of Eq. (19) becomes

$$(J \mathbf{U}_t, \phi_{ijk}^L)_{\hat{\Omega}} = \iiint_{\hat{\Omega}_{\text{sub}(ijk)}} J \mathbf{U}_t \phi_{ijk}^L dV = \iiint_{\hat{\Omega}_{\text{sub}(ijk)}} J \mathbf{U}_t dV = J_{ijk} w_i w_j w_k (\mathbf{U}_t)_{ijk}. \quad (21)$$

Here, we take the GLL nodal value f_{ijk} to represent the mean value of a function $f(\xi, \eta, \zeta)$ within the subcell $\hat{\Omega}_{\text{sub}(ijk)}$ in a low-order approximation. Combining Eqs. (20) and (21) as in Eq. (19), we obtain

$$(J \mathbf{U}_t, \phi_{ijk})_{\hat{\Omega}} = J_{ijk} w_i w_j w_k (\mathbf{U}_t)_{ijk}. \quad (22)$$

2.5. Convective term

For the convective term (the second term in Eq. (17a)), we write

$$(\nabla_{\xi} \cdot \tilde{\mathbf{F}}, \phi_{ijk}^H)_{\hat{\Omega}} = (1 - \alpha) (\nabla_{\xi} \cdot \tilde{\mathbf{F}}, \phi_{ijk}^H)_{\hat{\Omega}} + \alpha (\nabla_{\xi} \cdot \tilde{\mathbf{F}}, \phi_{ijk}^L)_{\hat{\Omega}}. \quad (23)$$

For the first term in the right-hand side of Eq. (23) (high-order term), we introduce a split-form discontinuous Galerkin spectral element discretization (DGSEM) in Section 2.5.1, while for the second term, we employ a low-order, finite-volume like discretization procedure in Section 2.5.2.

2.5.1. High-order convective term: Split-form discontinuous Galerkin

For the high-order convective term (the first term in the right-hand side of Eq. (23)), we can introduce a split-form DGSEM discretization [7, 47, 48],

$$\begin{aligned} (\nabla_{\xi} \cdot \tilde{\mathbf{F}}, \phi_{ijk}^H)_{\hat{\Omega}} &= \left(l_i(1) [\tilde{\mathbf{F}}^{1*} - \tilde{\mathbf{F}}^1]_{Njk} - l_i(-1) [\tilde{\mathbf{F}}^{1*} - \tilde{\mathbf{F}}^1]_{1jk} \right) w_j w_k \\ &\quad + \left(l_j(1) [\tilde{\mathbf{F}}^{2*} - \tilde{\mathbf{F}}^2]_{iNk} - l_j(-1) [\tilde{\mathbf{F}}^{2*} - \tilde{\mathbf{F}}^2]_{i1k} \right) w_i w_k \\ &\quad + \left(l_k(1) [\tilde{\mathbf{F}}^{3*} - \tilde{\mathbf{F}}^3]_{ijN} - l_k(-1) [\tilde{\mathbf{F}}^{3*} - \tilde{\mathbf{F}}^3]_{ij1} \right) w_i w_j \\ &\quad + 2w_i w_j w_k \left(\sum_{n=1}^N \tilde{\mathbf{F}}^{1\#}(\mathbf{U}_{njk}, \mathbf{U}_{ijk}) l'_n(\xi_i) + \sum_{n=1}^N \tilde{\mathbf{F}}^{2\#}(\mathbf{U}_{ink}, \mathbf{U}_{ijk}) l'_n(\eta_j) + \sum_{n=1}^N \tilde{\mathbf{F}}^{3\#}(\mathbf{U}_{ijn}, \mathbf{U}_{ijk}) l'_n(\zeta_k) \right), \end{aligned} \quad (24)$$

where $w_{i,j,k}$ are the quadrature weights, $\tilde{\mathbf{F}}^{l*}$ ($l = 1, 2, 3$) is the surface flux computed based on a corresponding Riemann problem solution, and $\tilde{\mathbf{F}}^{l\#}$ ($l = 1, 2, 3$) is a two-point flux operator in the volume contribution [29,31].

For the calculation of fluxes in Eq. (24), we employ a kinetic energy preserving and entropy conserving (KEPEC) scheme [49]. KEPEC flux based on the left and right states ($\mathbf{U}_1, \mathbf{U}_2$) can be defined as [29,31,49]

$$\tilde{\mathbf{F}}^{l,\text{KEPEC}}(\mathbf{U}_1, \mathbf{U}_2) = \begin{pmatrix} \hat{\rho}\{\{\mathbf{u}^T\}\}_{1,2} \\ \hat{\rho}\{\{\mathbf{u}\}\}_{1,2} \otimes \{\{\mathbf{u}\}\}_{1,2} + \mathbf{I}\hat{\rho} \\ \hat{\rho}\{\{\mathbf{u}^T\}\}_{1,2}\hat{h} \end{pmatrix} \cdot \mathbf{N}_A^l, \quad (25)$$

where the arithmetic mean operator

$$\{\{\cdot\}\}_{1,2} = \frac{1}{2}[(\cdot)_1 + (\cdot)_2] \quad (26)$$

is introduced, $\hat{\rho} = (\rho_1 - \rho_2) / (\ln(\rho_1) - \ln(\rho_2))$ is the logarithmic mean of density, $\hat{p} = \{\{\rho\}\}_{1,2} / (2\{\beta\}_{1,2})$ is the averaged pressure, $\beta = \rho/(2p)$ is the temperature inverse, $\hat{\beta} = (\beta_1 - \beta_2) / (\ln(\beta_1) - \ln(\beta_2))$ is the logarithmic mean of the temperature inverse, $\hat{h} = (2\hat{\beta}(\gamma - 1))^{-1} - \{\{\mathbf{u} \cdot \mathbf{u}^T\}\}_{1,2}/2 + \hat{p}/\hat{\rho} + \{\{\mathbf{u}\}\}_{1,2} \cdot \{\{\mathbf{u}\}\}_{1,2}^T$ is the averaged enthalpy, and \mathbf{N}_A^l is an appropriately calculated surface normal scaled with the area. An appropriate calculation of the surface normal is important to preserve a high-order discretization on curvilinear meshes. For the DG fluxes presented in this section, the scaled surface normal is calculated using the arithmetic mean operator,

$$\mathbf{N}_A^l = \{\{J\mathbf{a}^l\}\}_{1,2} = \frac{1}{2}[(J\mathbf{a}^l)_1 + (J\mathbf{a}^l)_2]. \quad (27)$$

The KEPEC flux of Eq. (25) is adopted as a two-point flux formula in the volume contribution, such as, e.g.,

$$\tilde{\mathbf{F}}^{l\#}(\mathbf{U}_{njk}, \mathbf{U}_{ijk}) = \tilde{\mathbf{F}}^{l,\text{KEPEC}}(\mathbf{U}_{njk}, \mathbf{U}_{ijk}), \quad (28)$$

with the normal calculated as

$$\mathbf{N}_A^1 = \{\{J\mathbf{a}^1\}\}_{njk,ijk} = \frac{1}{2}[(J\mathbf{a}^1)_{njk} + (J\mathbf{a}^1)_{ijk}], \quad (29)$$

and similar for the other two directions. For the surface flux, we employ the same KEPEC flux plus a Rusanov-type stabilization term [49],

$$\tilde{\mathbf{F}}^{l*}(\mathbf{U}^+, \mathbf{U}^-) = \tilde{\mathbf{F}}^{l,\text{KEPEC}}(\mathbf{U}^+, \mathbf{U}^-) - \frac{1}{2} \sum_{j=1}^5 \alpha_j |\bar{\lambda}_j| \mathbf{k}^{(j)} |J\mathbf{a}^l|, \quad (30)$$

where

$$\alpha_1 = \frac{\Delta p - \bar{\rho}\bar{c}\Delta u_n}{2\bar{c}^2}, \quad \alpha_2 = \Delta\rho - \frac{\Delta p}{\bar{c}^2}, \quad \alpha_3 = \bar{\rho}, \quad \alpha_4 = 0, \quad \alpha_5 = \frac{\Delta p + \bar{\rho}\bar{c}\Delta u_n}{2\bar{c}^2}, \quad (31)$$

$$\mathbf{k}^{(1)} = \begin{pmatrix} 1 \\ \bar{u} - \bar{c}\hat{n}_1 \\ \bar{v} - \bar{c}\hat{n}_2 \\ \bar{w} - \bar{c}\hat{n}_3 \\ \bar{h} - \bar{c}\bar{u}_n \end{pmatrix}, \quad \mathbf{k}^{(2)} = \begin{pmatrix} 1 \\ \bar{u} \\ \bar{v} \\ \bar{w} \\ \frac{1}{2}|\bar{\mathbf{u}}|^2 \end{pmatrix}, \quad \mathbf{k}^{(3)} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \quad \mathbf{k}^{(4)} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \quad \mathbf{k}^{(5)} = \begin{pmatrix} 1 \\ \bar{u} + \bar{c}\hat{n}_1 \\ \bar{v} + \bar{c}\hat{n}_2 \\ \bar{w} + \bar{c}\hat{n}_3 \\ \bar{h} + \bar{c}\bar{u}_n \end{pmatrix}. \quad (32)$$

In Eqs. (31) and (32), Δ denotes the jump, e.g., $\Delta p = p^+ - p^-$, where $(\cdot)^+$ denotes a boundary gridpoint value taken from the element adjacent to the interface in the increasing ξ^l coordinate direction and $(\cdot)^-$ denotes a boundary gridpoint value taken from the element adjacent to the interface in the decreasing ξ^l coordinate direction, u_n is the component of velocity in the direction of the surface normal, and λ_j are the eigenvalues of the convective flux Jacobian. The overbar denotes the Roe average, e.g.,

$$\bar{\rho} = \sqrt{\rho^- \rho^+}, \quad \bar{\mathbf{u}} = \frac{\sqrt{\rho^-} \mathbf{u}^- + \sqrt{\rho^+} \mathbf{u}^+}{\sqrt{\rho^-} + \sqrt{\rho^+}}, \quad \bar{h} = \frac{\sqrt{\rho^-} h^- + \sqrt{\rho^+} h^+}{\sqrt{\rho^-} + \sqrt{\rho^+}}, \quad \bar{c} = \sqrt{(\gamma - 1)(\bar{h} - \frac{1}{2}|\bar{\mathbf{u}}|^2)}. \quad (33)$$

To preserve kinetic energy stability of the scheme [49], Rusanov condition is used for the stabilization term in Eq. (30), which sets all the eigenvalues equal to $\bar{\lambda}_j = \bar{u}_n + \bar{c}$, $j \in \{1, \dots, 5\}$. Boundary conditions are imposed weakly when calculating the surface fluxes. Further details of the DGSEM discretization employed in the current work, including the derivation of Eq. (24) can be found in Appendix A. As opposed to a standard DGSEM, a split-form DGSEM was found to be able to suppress the aliasing instability caused by the inherent non-linearity of the governing equations, thus improving the stability properties of the method [7,47].

2.5.2. Low-order convective term: Finite-volume inspired approach

For the low-order convective term (the second term in Eq. (23)), a finite-volume like discretization procedure is employed. In particular, due to a compact support of the low-order basis function $\phi_{ijk}^L(\xi, \eta, \zeta)$ in the subcell $\hat{\Omega}_{\text{sub}(ijk)}$ within the element $\hat{\Omega}$ (See Eq. (16)), we can write

$$(\nabla_\xi \cdot \tilde{\mathbf{F}}, \phi_{ijk}^L)_{\hat{\Omega}} = \iiint_{\hat{\Omega}_{\text{sub}(ijk)}} (\nabla_\xi \cdot \tilde{\mathbf{F}}) dV. \quad (34)$$

Upon some manipulation, the low-order flux can be rewritten as

$$(\nabla_\xi \cdot \tilde{\mathbf{F}}, \phi_{ijk}^L)_{\hat{\Omega}} = w_j w_k (\tilde{\mathbf{F}}_{(i,i+1)jk}^{1*,L} - \tilde{\mathbf{F}}_{(i-1,i)jk}^{1*,L}) + w_i w_k (\tilde{\mathbf{F}}_{i(j,j+1)k}^{2*,L} - \tilde{\mathbf{F}}_{i(j-1,j)k}^{2*,L}) + w_i w_j (\tilde{\mathbf{F}}_{ij(k,k+1)}^{3*,L} - \tilde{\mathbf{F}}_{ij(k-1,k)}^{3*,L}), \quad (35)$$

which satisfies the SBP property [29,31]. For the surface fluxes in a low-order discretization, $\tilde{\mathbf{F}}^{l*,L}$ ($l = 1, 2, 3$), we use the KEPEC scheme with the Rusanov-type dissipation term for the Riemann solver [49], the same as in the DG formulas, given by Eq. (30). The only difference is the

calculation of the scaled surface normal in the KEPEC flux in Eq. (25), which needs to account for the subcell finite-volume metrics in the curvilinear coordinates [29,31] as

$$\begin{aligned}\mathbf{N}_{A(i,i+1)jk}^1 &= (\mathbf{J}\mathbf{a}^1)_{1jk} + \sum_{m=1}^i w_m \sum_{n=1}^N l'_n(\xi_m) \left[(\mathbf{J}\mathbf{a}^1)_{mjk} + (\mathbf{J}\mathbf{a}^1)_{njk} \right] \\ \mathbf{N}_{A(i,j+1)k}^1 &= (\mathbf{J}\mathbf{a}^2)_{ijk} + \sum_{m=1}^j w_m \sum_{n=1}^N l'_n(\eta_m) \left[(\mathbf{J}\mathbf{a}^2)_{imk} + (\mathbf{J}\mathbf{a}^2)_{ink} \right] \\ \mathbf{N}_{Aij(k,k+1)}^3 &= (\mathbf{J}\mathbf{a}^3)_{ij1} + \sum_{m=1}^k w_m \sum_{n=1}^N l'_n(\zeta_m) \left[(\mathbf{J}\mathbf{a}^3)_{ijm} + (\mathbf{J}\mathbf{a}^3)_{ijn} \right].\end{aligned}\quad (36)$$

To improve the resolution of the low-order flux, we utilize a reconstruction procedure combined with a total variation diminishing (TVD) slope limiter of the second order [50] on the inner subcells. Following [25], in this work we reconstruct the primitive variables $\{\rho, \mathbf{u}, T\}$ with a linear interpolating polynomial evaluated based on the value of the minmod function, prior to the calculation of the low-order surface fluxes, $\tilde{\mathbf{F}}^{i*,L}$ ($i = 1, 2, 3$). Taking density ρ as an example, a linear interpolating polynomial in the ξ direction, $\rho_{ijk}^{\text{rec}}(\xi)$, is constructed as follows,

$$\rho_{ijk}^{\text{rec}}(\xi) = \rho_{ijk} + (\xi - \xi_i)\psi_{ijk}, \quad (37)$$

with

$$\psi_{ijk} = \begin{cases} \text{Minmod} \left(\frac{\rho_{(i+1)jk} - \rho_{ijk}}{\xi_{(i+1)jk} - \xi_{ijk}}, \frac{\rho_{ijk} - \rho_{(i-1)jk}}{\xi_{ijk} - \xi_{(i-1)jk}} \right), & 1 < i < N \\ 0, & i = 1 \text{ or } N, \end{cases} \quad (38)$$

where

$$\text{Minmod}(\phi_l, \phi_r) = \begin{cases} 0, & \phi_l \phi_r < 0 \\ \phi_r, & \phi_l \phi_r > 0 \text{ and } |\phi_l| > |\phi_r| \\ \phi_l, & \phi_l \phi_r > 0 \text{ and } |\phi_l| < |\phi_r|. \end{cases} \quad (39)$$

To reconstruct the density value at the subcell face normal to the ξ direction, Eq. (37) is evaluated with a corresponding $\xi = \xi_{i\pm1/2}$. Any primitive variable can be reconstructed in any of the coordinate directions accordingly.

2.5.3. Calculation of blending coefficient

To allow for a shock capturing in the vicinity of discontinuities while retaining a high-order accuracy of DG approximation in the shock-free areas, the blending coefficient α is calculated locally at each element. In order to be able to distinguish shocks from regions of high shear created due to local turbulent vorticity fluctuations, a modified Ducros sensor [40] is employed. To this end, we first define the local, pointwise blending coefficient as

$$\alpha_{ijk} = \Phi_{ijk} \times \Xi_{ijk}, \quad (40)$$

where

$$\Phi_{ijk} = \frac{(\nabla \cdot \mathbf{u})^2}{(\nabla \cdot \mathbf{u})^2 + (\nabla \times \mathbf{u})^2 + \epsilon} \quad (41)$$

is the original Ducros sensor [41], and

$$\Xi_{ijk} = \frac{1}{2} \left[1 - \tanh \left(c_1 + c_2 \frac{h_E}{c} \nabla \cdot \mathbf{u} \right) \right] \quad (42)$$

is the activation function introduced by Bhagatwala and Lele [40], where c refers to a local speed of sound, c_1 and c_2 are two adjustable parameters, h_E is the smallest distance between the adjacent vertices of the element in the physical space, and $\epsilon = 10^{-10}$ in Eq. (41) is a small number to prevent from dividing by zero in case all other terms in the denominator are zero. For all the test cases in this paper, c_1 and c_2 are set to 2.0 and 20.0, respectively.

Ducros sensor, Φ_{ijk} , is designed to distinguish the regions with high values of dilatation (shocks) from the regions of high vorticity (potentially strong motions within turbulent shear layers that are accompanied by steep solution gradients but are not shocks). Ducros sensor attains a value close to 1 in the regions of high dilatation while reducing it down to 0 in the regions of high vorticity with no dilatation. As discussed in [38,40], the original Ducros sensor still has limitations, in particular related to spurious identification of some local turbulent dilatational motions (shocklets) as shocks, and inability to distinguish between the regions of negative dilatation (shocks) and positive dilatation (expansion waves), which results in unnecessary damping of expansion waves and turbulent shocklets. To address these deficiencies, an activation function Ξ_{ijk} proposed by [40], is utilized in Eq. (40) to calculate the blending coefficient for the developed hybrid shock capturing scheme. This activation function was found to be effective in eliminating the problems enumerated above in the current method. To determine the elementwise blending coefficient α used in the hybrid scheme, the maximum pointwise value across the element is used,

$$\alpha = \max_{\mathbf{x}_{ijk} \in \Omega} \alpha_{ijk}. \quad (43)$$

2.6. Diffusion term

Incorporation of a low-order flux into a convective term discretization described in Section 2.5 was needed to provide a stable formulation in the presence of shocks. Since diffusive processes are naturally more stable, it was found that incorporation of a finite-volume like discretization for the viscous terms was not required in the current method, even near shocks. Thus, in the current formulation, both “high-order” and “low-order”

diffusive fluxes are discretized in the same way, employing Bassi and Rebay [37] BR1 scheme in curvilinear coordinates. To maintain the rigor of the formulation, we derive the corresponding formulas for the high-order flux, and then let the low-order flux attain the same expressions. Additional details regarding the derivation of the relevant discretizations of the diffusive fluxes are presented in [Appendix B](#).

Separating “high-order” and “low-order” terms in the left-hand side of Eq. [\(17b\)](#) yields

$$(J\mathbf{q}, \phi_{ijk})_{\hat{\Omega}} = (1 - \alpha) (J\mathbf{q}, \phi_{ijk}^H)_{\hat{\Omega}} + \alpha (J\mathbf{q}, \phi_{ijk}^L)_{\hat{\Omega}}. \quad (44)$$

For the high-order term, we have, due to a cardinality property of the Lagrangian basis functions,

$$(1 - \alpha) (J\mathbf{q}, \phi_{ijk}^H)_{\hat{\Omega}} = (1 - \alpha) J_{ijk} \mathbf{q}_{ijk} w_i w_j w_k. \quad (45)$$

Employing the same arguments as in Section [2.4](#), we can show that the low-order term results in a similar formula,

$$\alpha (J\mathbf{q}, \phi_{ijk}^L)_{\hat{\Omega}} = \alpha J_{ijk} \mathbf{q}_{ijk} w_i w_j w_k, \quad (46)$$

and the combination of the two yields the expected,

$$(J\mathbf{q}, \phi_{ijk})_{\hat{\Omega}} = J_{ijk} \mathbf{q}_{ijk} w_i w_j w_k. \quad (47)$$

For the right-hand side of Eq. [\(17b\)](#), we have

$$(\tilde{\nabla}_{\xi} \mathbf{U}, \phi_{ijk})_{\hat{\Omega}} = \sum_{l=1}^3 \left(\frac{\partial J\mathbf{a}^l \mathbf{U}}{\partial \xi^l}, \phi_{ijk} \right)_{\hat{\Omega}}, \quad (48)$$

which can be rewritten as

$$(\tilde{\nabla}_{\xi} \mathbf{U}, \phi_{ijk})_{\hat{\Omega}} = (1 - \alpha) \sum_{l=1}^3 \left(\frac{\partial J\mathbf{a}^l \mathbf{U}}{\partial \xi^l}, \phi_{ijk}^H \right)_{\hat{\Omega}} + \alpha \sum_{l=1}^3 \left(\frac{\partial J\mathbf{a}^l \mathbf{U}}{\partial \xi^l}, \phi_{ijk}^L \right)_{\hat{\Omega}}. \quad (49)$$

Dealing first with the high-order term in the right-hand side of Eq. [\(49\)](#), we take taking $l = 1$ as an example and utilize the BR1 scheme [37] together with the SBP operator [47,51], which yields the following discretization

$$\begin{aligned} \left(\frac{\partial J\mathbf{a}^1 \mathbf{U}}{\partial \xi^1}, \phi_{ijk}^H \right)_{\hat{\Omega}} &= \left[(J\mathbf{a}^1 \mathbf{U})_{Njk}^{**} l_i(1) - (J\mathbf{a}^1 \mathbf{U})_{1jk}^{**} l_i(-1) \right] w_j w_k \\ &\quad + \sum_{n=1}^N (J\mathbf{a}^1 \mathbf{U})_{njk} l'_n(\xi_i) w_i w_j w_k, \end{aligned} \quad (50)$$

where jumps across the interfaces are introduced according to [37] as

$$(J\mathbf{a}^1 \mathbf{U})^{**} = \pm \frac{1}{2} \left((J\mathbf{a}^1 \mathbf{U})^+ - (J\mathbf{a}^1 \mathbf{U})^- \right), \quad (51)$$

where “+” sign is taken for $(J\mathbf{a}^1 \mathbf{U})_{Njk}^{**}$ and “-” sign for $(J\mathbf{a}^1 \mathbf{U})_{1jk}^{**}$.

Similarly, we can discretize the other two components in the sum, to yield

$$\begin{aligned} \sum_{l=1}^3 \left(\frac{\partial J\mathbf{a}^l \mathbf{U}}{\partial \xi^l}, \phi_{ijk}^H \right)_{\hat{\Omega}} &= \left[(J\mathbf{a}^1 \mathbf{U})_{Njk}^{**} l_i(1) - (J\mathbf{a}^1 \mathbf{U})_{1jk}^{**} l_i(-1) \right] w_j w_k \\ &\quad + \left[(J\mathbf{a}^2 \mathbf{U})_{iNk}^{**} l_j(1) - (J\mathbf{a}^2 \mathbf{U})_{i1k}^{**} l_j(-1) \right] w_i w_k \\ &\quad + \left[(J\mathbf{a}^3 \mathbf{U})_{ijN}^{**} l_k(1) - (J\mathbf{a}^3 \mathbf{U})_{ij1}^{**} l_k(-1) \right] w_i w_j \\ &\quad + \sum_{n=1}^N \left[(J\mathbf{a}^1 \mathbf{U})_{njk} l'_n(\xi_i) + (J\mathbf{a}^2 \mathbf{U})_{ink} l'_n(\eta_j) + (J\mathbf{a}^3 \mathbf{U})_{ijn} l'_n(\zeta_k) \right] w_i w_j w_k. \end{aligned} \quad (52)$$

Employing the same discretization as in Eq. [\(52\)](#) for the low-order term, $\sum_{l=1}^3 \left(\partial(J\mathbf{a}^l \mathbf{U}) / \partial \xi^l, \phi_{ijk}^L \right)_{\hat{\Omega}}$, combining the two terms together as in Eq. [\(49\)](#), and utilizing Eq. [\(47\)](#), we can solve for \mathbf{q}_{ijk} as

$$\begin{aligned} \mathbf{q}_{ijk} &= \left[(J\mathbf{a}^1 \mathbf{U})_{Njk}^{**} l_i(1) - (J\mathbf{a}^1 \mathbf{U})_{1jk}^{**} l_i(-1) \right] / (J_{ijk} w_i) \\ &\quad + \left[(J\mathbf{a}^2 \mathbf{U})_{iNk}^{**} l_j(1) - (J\mathbf{a}^2 \mathbf{U})_{i1k}^{**} l_j(-1) \right] / (J_{ijk} w_j) \\ &\quad + \left[(J\mathbf{a}^3 \mathbf{U})_{ijN}^{**} l_k(1) - (J\mathbf{a}^3 \mathbf{U})_{ij1}^{**} l_k(-1) \right] / (J_{ijk} w_k) \\ &\quad + \sum_{n=1}^N \left[(J\mathbf{a}^1 \mathbf{U})_{njk} l'_n(\xi_i) + (J\mathbf{a}^2 \mathbf{U})_{ink} l'_n(\eta_j) + (J\mathbf{a}^3 \mathbf{U})_{ijn} l'_n(\zeta_k) \right] / J_{ijk}. \end{aligned} \quad (53)$$

To discretize the viscous term in Eq. [\(17a\)](#), the similar procedure is employed. First, we write

$$(\nabla_{\xi} \cdot \tilde{\mathbf{F}}_V, \phi_{ijk})_{\hat{\Omega}} = (1 - \alpha) (\nabla_{\xi} \cdot \tilde{\mathbf{F}}_V, \phi_{ijk}^H)_{\hat{\Omega}} + \alpha (\nabla_{\xi} \cdot \tilde{\mathbf{F}}_V, \phi_{ijk}^L)_{\hat{\Omega}}. \quad (54)$$

Taking a high-order term,

$$(\nabla_{\xi} \cdot \tilde{\mathbf{F}}_V, \phi_{ijk}^H)_{\hat{\Omega}} = \sum_{l=1}^3 \left(\frac{\partial \tilde{\mathbf{F}}_V^l}{\partial \xi^l}, \phi_{ijk}^H \right)_{\hat{\Omega}}, \quad (55)$$

and restricting our attention to the first component in the sum above, we obtain the following discretization (refer to Appendix B for additional details),

$$\left(\frac{\partial \tilde{\mathbf{F}}_V^1}{\partial \xi^1}, \phi_{ijk}^H \right)_{\hat{\Omega}} = \left[(\tilde{\mathbf{F}}_V^{1*})_{Njk} l_i(1) - (\tilde{\mathbf{F}}_V^{1*})_{1jk} l_i(-1) \right] w_j w_k - \sum_{n=1}^N \tilde{\mathbf{F}}_V^1 l'_i(\xi_n) w_n w_j w_k, \quad (56)$$

where an arithmetic mean between the two sides of the interface is used to define the viscous surface flux [37],

$$\tilde{\mathbf{F}}_V^{l*} = \{\{\tilde{\mathbf{F}}_V\}\}_{+, -} = \frac{1}{2} \left(\tilde{\mathbf{F}}_V^{l+} + \tilde{\mathbf{F}}_V^{l-} \right), \quad (57)$$

with $(\cdot)^+$ and $(\cdot)^-$ notations the same as in Section 2.5.1. The other two components in the sum on the right-hand side of Eq. (55) are discretized following the same procedure, to yield

$$\begin{aligned} \left(\nabla_{\xi} \cdot \tilde{\mathbf{F}}_V, \phi_{ijk}^H \right)_{\hat{\Omega}} &= \left[(\tilde{\mathbf{F}}_V^{1*})_{Njk} l_i(1) - (\tilde{\mathbf{F}}_V^{1*})_{1jk} l_i(-1) \right] w_j w_k + \left[(\tilde{\mathbf{F}}_V^{2*})_{iNk} l_j(1) - (\tilde{\mathbf{F}}_V^{2*})_{i1k} l_j(-1) \right] w_i w_k \\ &\quad + \left[(\tilde{\mathbf{F}}_V^{3*})_{ijN} l_k(1) - (\tilde{\mathbf{F}}_V^{3*})_{ij1} l_k(-1) \right] w_i w_j - \sum_{n=1}^N \tilde{\mathbf{F}}_V^1 l'_i(\xi_n) w_n w_j w_k - \sum_{n=1}^N \tilde{\mathbf{F}}_V^2 l'_j(\eta_n) w_i w_n w_k \\ &\quad - \sum_{n=1}^N \tilde{\mathbf{F}}_V^3 l'_k(\zeta_n) w_i w_j w_n. \end{aligned} \quad (58)$$

Finally, using the same discretization as in Eq. (58) for the low-order term, $\left(\nabla_{\xi} \cdot \tilde{\mathbf{F}}_V, \phi_{ijk}^L \right)_{\hat{\Omega}}$, and combining the two together as in Eq. (54), yields the final approximation of the viscous term as

$$\begin{aligned} \left(\nabla_{\xi} \cdot \tilde{\mathbf{F}}_V, \phi_{ijk} \right)_{\hat{\Omega}} &= \left[(\tilde{\mathbf{F}}_V^{1*})_{Njk} l_i(1) - (\tilde{\mathbf{F}}_V^{1*})_{1jk} l_i(-1) \right] w_j w_k + \left[(\tilde{\mathbf{F}}_V^{2*})_{iNk} l_j(1) - (\tilde{\mathbf{F}}_V^{2*})_{i1k} l_j(-1) \right] w_i w_k \\ &\quad + \left[(\tilde{\mathbf{F}}_V^{3*})_{ijN} l_k(1) - (\tilde{\mathbf{F}}_V^{3*})_{ij1} l_k(-1) \right] w_i w_j - \sum_{n=1}^N \tilde{\mathbf{F}}_V^1 l'_i(\xi_n) w_n w_j w_k - \sum_{n=1}^N \tilde{\mathbf{F}}_V^2 l'_j(\eta_n) w_i w_n w_k \\ &\quad - \sum_{n=1}^N \tilde{\mathbf{F}}_V^3 l'_k(\zeta_n) w_i w_j w_n. \end{aligned} \quad (59)$$

2.7. Temporal integration

Utilizing Eq. (22) for the temporal term, Eq. (17a) can be rewritten as

$$(\mathbf{U}_t)_{ijk} = \Re(\mathbf{U}_{ijk}, \mathbf{q}_{ijk}), \quad (60)$$

where

$$\Re(\mathbf{U}_{ijk}, \mathbf{q}_{ijk}) = \frac{1}{J_{ijk} w_i w_j w_k} \left[(\nabla_{\xi} \cdot \tilde{\mathbf{F}}_V, \phi_{ijk})_{\hat{\Omega}} - (\nabla_{\xi} \cdot \tilde{\mathbf{F}}, \phi_{ijk})_{\hat{\Omega}} \right], \quad (61)$$

and the discretized expressions for the convective and viscous terms in Eq. (61) can be obtained from Eqs. (23), (24), (35) and (59), respectively.

The third-order strong stability-preserving Runge–Kutta (RK) scheme [52] is applied to integrate Eq. (60) in time from t_n to t_{n+1} ,

$$\begin{aligned} \mathbf{U}_{ijkn}^{(1)} &= \mathbf{U}_{ijkn} + \Delta t \Re(\mathbf{U}_{ijkn}, \mathbf{q}_{ijkn}), \\ \mathbf{U}_{ijkn}^{(2)} &= \frac{3}{4} \mathbf{U}_{ijkn} + \frac{1}{4} \mathbf{U}_{ijkn}^{(1)} + \frac{1}{4} \Delta t \Re(\mathbf{U}_{ijkn}^{(1)}, \mathbf{q}_{ijkn}), \\ \mathbf{U}_{ijk(n+1)} &= \frac{1}{3} \mathbf{U}_{ijkn} + \frac{2}{3} \mathbf{U}_{ijkn}^{(2)} + \frac{2}{3} \Delta t \Re(\mathbf{U}_{ijkn}^{(2)}, \mathbf{q}_{ijkn}), \end{aligned} \quad (62)$$

where $f_{ijkn} = f(\xi_i, \eta_j, \zeta_k, t_n)$. The time step Δt is bounded by the condition numbers of both the convective and the diffusive parts of the equation in the following expressions,

$$\Delta t \leq \min(\Delta t_c, \Delta t_d), \quad (63)$$

with

$$\begin{aligned} \Delta t_c &\leq C_c \min_{\hat{\Omega}_E} \min_{i,j,k} \frac{h_E}{(N-1)^2 \lambda_{ijkn}}, \\ \Delta t_d &\leq C_d \min_{\hat{\Omega}_E} \min_{i,j,k} \frac{h_E^2}{(N-1)^4 \nu_{ijkn}}, \end{aligned} \quad (64)$$

where λ and ν are the maximum eigenvalues of the convective and diffusive Jacobian matrices, $[\partial \mathbf{F} / \partial \mathbf{U}]$, $[\partial \mathbf{F}_V / \partial \mathbf{U}]$, respectively. The coefficients C_c , C_d in Eq. (64) must be less than unity for stability. The CFL number can be subsequently computed as

$$\text{CFL} = \Delta t \max_{\hat{\Omega}_E} \max_{i,j,k} \left(\left| \frac{\lambda \cdot \mathbf{a}^1}{J \Delta \xi_i} \right| + \left| \frac{\lambda \cdot \mathbf{a}^2}{J \Delta \eta_j} \right| + \left| \frac{\lambda \cdot \mathbf{a}^3}{J \Delta \zeta_k} \right| \right) \quad (65)$$

where $\lambda = (|u| + c, |v| + c, |w| + c)$ is the maximum wave speed calculated locally on GLL points.

Table 1
Summary of numerical examples.

Case	Section	CFL	Dimension	Equations	Discontinuity	Curvilinear
1. Isentropic Euler vortex	3.1.1	0.20 – 0.40	2D	Euler	N	Y
2. Sod shocktube	3.1.2	0.40 – 0.50	1D	Euler	Y	N
3. Riemann case 3	3.1.3	0.35 – 0.50	2D	Euler	Y	N
4. Double-Mach reflection	3.1.4	0.32 – 0.45	2D	Euler	Y	N
5. Shock-boundary layer interaction	3.2.1	0.52 – 0.54	2D	N.S.	Y	N
6. Viscous shocktube	3.2.2	0.24 – 0.30	2D	N.S.	Y	N
7. Taylor–Green vortex	3.2.3	0.45 – 0.50	3D	N.S.	N	N
8. Flow over sphere at $Ma = 2$ (DNS)	3.2.4	0.38 – 0.45	3D	N.S.	Y	Y

3. Numerical results

To verify the properties of the developed numerical methodology, we apply it to calculate eight numerical examples summarized in [Table 1](#). The examples test both the inviscid (Euler) and viscous (Navier–Stokes — N.S.) formulations in one, two and three dimensions. Most of the problems (except for problems 1 and 7) include discontinuities, while problems 1 and 7 test smooth solutions to demonstrate a high-order convergence of the methodology in the absence of shocks and its ability to resolve small-scale turbulence, respectively. Problems 1 and 8 specifically test the application of the methodology to curvilinear meshes, wherein Problem 8 performs Direct Numerical Simulations (DNS) of a supersonic turbulent flow past a sphere at $Re_D = 6.5 \times 10^5$ and validates it with the available experimental and computational data. [Table 1](#) also lists the CFL numbers that were used for the test cases. Compared to our previous work [47], where an artificial dissipation method was used for shock capturing, the current hybrid sub-element order reduction shock-capturing scheme permits a use of much larger CFL numbers, hence larger time steps (for example, a 5-fold increase in a CFL number for Riemann case 3, cf. with Table 1 in [47]).

3.1. Test cases for Euler equations

3.1.1. Isentropic Euler vortex

We use the isentropic Euler vortex problem to conduct the convergence study for a smooth case on both Cartesian and curvilinear meshes. The problem is solved on a square domain Ω with a side length $L = 20$, such that $\Omega = [-10, 10]^2$. With specific initial conditions ($t = 0$ in the below equations), the analytical solution can be expressed as

$$\begin{aligned} u(\mathbf{x}, t) &= u_\infty - \frac{b}{2\pi} (y - v_\infty t) e^{1 - ((x - u_\infty t)^2 + (y - v_\infty t)^2)}, \\ v(\mathbf{x}, t) &= v_\infty + \frac{b}{2\pi} (x - u_\infty t) e^{1 - ((x - u_\infty t)^2 + (y - v_\infty t)^2)}, \\ T(\mathbf{x}, t) &= 1 - \frac{(\gamma - 1)b^2}{16\gamma\pi^2} (x - u_\infty t) e^{2(1 - (x - u_\infty t)^2 + (y - v_\infty t)^2)}, \\ \rho(\mathbf{x}, t) &= T(\mathbf{x}, t)^{\frac{1}{\gamma-1}}, \end{aligned} \quad (66)$$

where $u_\infty = 1$, $v_\infty = 0$ and $b = 5$ were used, and $\gamma = C_p/C_v = 1.4$. Tests on both Cartesian and curvilinear meshes are carried out. In the curvilinear case, the domain boundaries are kept as the straight lines, while the interior mesh is distorted by the following mapping function,

$$\begin{aligned} x &= \bar{x} + A_0 \sin(2\pi\bar{x}/\lambda_0) \sin(2\pi\bar{y}/\lambda_0), \\ y &= \bar{y} + A_0 \sin(2\pi\bar{x}/\lambda_0) \sin(2\pi\bar{y}/\lambda_0), \end{aligned} \quad (67)$$

where (\bar{x}, \bar{y}) are the mesh coordinates in the reference (undistorted) configuration, (x, y) are the coordinates of a distorted mesh, $A_0 = 1$ and $\lambda_0 = 20$ for this case. The distorted mesh is shown in [Fig. 3](#).

The errors are calculated at one flow-through time ($t = 20$) for a series of meshes that are progressively refined using h -refinement with the polynomial order $N = 6$, from the coarsest mesh with the element spacing of $L/25$ to the finest mesh with the spacing of $L/400$ (the elements are all square in the original configuration). For the polynomial order $N = 6$, we expect the optimal convergence rate to be 6. The tests are performed with the two values of the CFL number, $CFL = 0.4$ and $CFL = 0.2$, to test the influence of temporal discretization on the convergence rate. The L^2 norm of the errors of all the four conserved variables, together with the convergence rate, are documented in [Tables 2, 3](#) for the Cartesian meshes and in [Tables 4, 5](#) for the curvilinear meshes for both CFL numbers. We measure the convergence rate as $r^{(l)} = \log_2(e^{(l-1)}/e^{(l)})$, where the superscript (l) refers to the refinement level. For a low refinement level, such as $L/50$, the convergence is measured with respect to the coarsest mesh whose solution is unconvolved, which affects convergence rate. As the mesh is refined, the convergence rate reaches its optimal value. It can be seen that the CFL number does not affect convergence for the coarser meshes with the resolution up to $L/100$ in Cartesian and $L/200$ in curvilinear configurations. Decreasing CFL number to 0.2 improves convergence of $L/200$ mesh in Cartesian and $L/400$ mesh in curvilinear configurations. The finest Cartesian mesh with $L/400$ does not achieve optimal convergence with any of the attempted CFL numbers; further reduction in CFL number or a higher-order RK discretization would be needed to subdue the temporal errors at this fine spatial resolution. This behavior can be attributed to the interaction between spatial and temporal errors. When the grid is coarse and the spatial errors dominate, the temporal errors do not interfere with convergence. As the mesh is refined and the spatial errors diminish, the temporal errors may become dominant and halt the convergence; decreasing time step (CFL number) in this situation helps recover an optimal convergence rate. We remark that spatial errors are generally larger on curvilinear meshes, extending the range of refinement levels on which convergence is still dominated by spatial errors.

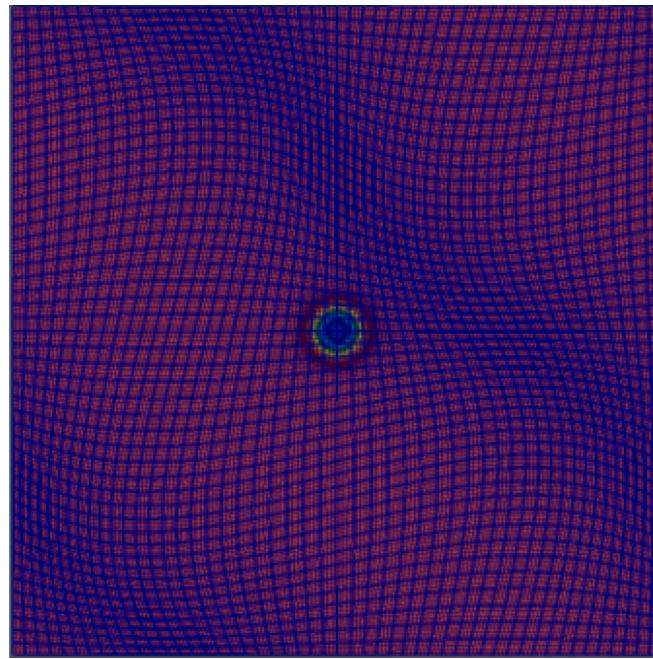


Fig. 3. Density contours corresponding to the initial condition for the isentropic Euler vortex on a distorted mesh.

Table 2

L^2 error of conserved variables for the isentropic Euler vortex case on a Cartesian mesh with CFL = 0.4.

h	$L^2(\epsilon_\rho)$	Rate	$L^2(\epsilon_{\rho u})$	Rate	$L^2(\epsilon_{\rho v})$	Rate	$L^2(\epsilon_{\rho e})$	Rate
$L/25$	3.71E-05	–	1.28E-04	–	6.38E-05	–	1.77E-04	–
$L/50$	2.03E-06	4.20	5.88E-06	4.44	2.74E-06	4.54	8.61E-06	4.36
$L/100$	2.67E-08	6.25	2.17E-08	8.08	2.23E-08	6.94	8.43E-08	6.67
$L/200$	6.40E-10	5.38	1.21E-09	4.16	1.08E-09	4.37	2.62E-09	5.01
$L/400$	7.27E-11	3.14	1.65E-10	2.87	1.32E-10	3.03	3.12E-10	3.07

Table 3

L^2 error of conserved variables for the isentropic Euler vortex case on a Cartesian mesh with CFL = 0.2.

h	$L^2(\epsilon_\rho)$	Rate	$L^2(\epsilon_{\rho u})$	Rate	$L^2(\epsilon_{\rho v})$	Rate	$L^2(\epsilon_{\rho e})$	Rate
$L/25$	3.71E-05	–	1.28E-04	–	6.37E-05	–	1.77E-04	–
$L/50$	2.03E-06	4.19	5.88E-06	4.44	2.75E-06	4.53	8.62E-06	4.36
$L/100$	2.64E-08	6.26	1.97E-08	8.22	2.08E-08	7.05	8.23E-08	6.71
$L/200$	3.08E-10	6.42	3.09E-10	5.99	2.69E-10	6.27	1.02E-09	6.33
$L/400$	2.03E-11	3.92	7.82E-11	1.98	1.69E-11	3.99	7.99E-11	3.67

Table 4

L^2 error of conserved variables for the isentropic Euler vortex case on a curvilinear mesh with CFL = 0.4.

h	$L^2(\epsilon_\rho)$	Rate	$L^2(\epsilon_{\rho u})$	Rate	$L^2(\epsilon_{\rho v})$	Rate	$L^2(\epsilon_{\rho e})$	Rate
$L/25$	9.54E-05	–	5.21E-04	–	2.78E-04	–	6.17E-04	–
$L/50$	2.47E-06	5.27	1.32E-05	5.30	8.19E-06	5.09	1.58E-05	5.29
$L/100$	4.50E-08	5.78	1.67E-07	6.30	1.20E-07	6.09	2.10E-07	6.23
$L/200$	6.44E-10	6.13	3.58E-09	5.54	2.33E-09	5.69	4.31E-09	5.61
$L/400$	4.08E-11	3.98	1.06E-10	5.08	8.32E-11	4.81	1.88E-10	4.52

Table 5

L^2 error of conserved variables for the isentropic Euler vortex case on a curvilinear mesh with CFL = 0.2.

h	$L^2(\epsilon_\rho)$	Rate	$L^2(\epsilon_{\rho u})$	Rate	$L^2(\epsilon_{\rho v})$	Rate	$L^2(\epsilon_{\rho e})$	Rate
$L/25$	9.54E-05	–	5.21E-04	–	2.78E-04	–	6.17E-04	–
$L/50$	2.47E-06	5.27	1.32E-05	5.30	8.19E-06	5.09	1.58E-05	5.29
$L/100$	4.49E-08	5.78	1.67E-07	6.30	1.19E-07	6.10	2.10E-07	6.23
$L/200$	5.62E-10	6.32	3.53E-09	5.56	2.26E-09	5.72	4.09E-09	5.68
$L/400$	1.15E-11	5.61	7.01E-11	5.65	4.05E-11	5.80	8.23E-11	5.64

Table 6
 L^1 error of density in Sod shocktube case.

$N = 4$			$N = 6$			$N = 8$		
DOFs	L^1 error	Rate	DOFs	L^1 error	Rate	DOFs	L^1 error	Rate
100	8.58E-03	–	150	6.61E-03	–	200	5.30E-03	–
200	4.02E-03	1.0939	300	2.20E-03	1.5900	400	1.46E-03	1.8601
400	1.99E-03	1.0147	600	1.15E-03	0.9286	800	7.50E-04	0.9597

Table 7
 L^2 error of density in Sod shocktube case.

$N = 4$			$N = 6$			$N = 8$		
DOFs	L^2 error	Rate	DOFs	L^2 error	Rate	DOFs	L^2 error	Rate
100	1.75E-02	–	150	1.49E-02	–	200	1.21E-02	–
200	1.14E-02	0.6128	300	8.21E-03	0.8640	400	6.85E-03	0.8243
400	7.15E-03	0.6790	600	5.88E-03	0.4825	800	5.11E-03	0.4238

3.1.2. Sod shocktube

One-dimensional Sod shocktube problem [53] is used in the current study to test convergence of the method in the presence of discontinuities. The Sod shocktube problem is initialized as

$$(\rho, u, p) = \begin{cases} (1, 0, 1), & x \leq 0, \\ (0.125, 0, 0.1), & x > 0, \end{cases} \quad (68)$$

on the domain $x \in [-0.5, 0.5]$. Three levels of mesh refinement are investigated: meshes with 25, 50 and 100 elements. The elements are distributed uniformly with 4, 6, or 8 GLL points within each element. All results are obtained at $t = 0.2$. The exact solution is calculated using the method of characteristics [53]. The convergence with both h - and p -refinements is investigated, and the results are shown in Tables 6 and 7. The convergence rates of L^1 and L^2 errors are close to the theoretical values, which are 1 and 0.5, respectively [54].

3.1.3. Riemann problem case 3

The two-dimensional Riemann problem case 3 is presented here to demonstrate the capability of the method to capture two-dimensional shocks. In this problem, domain is divided into four equal parts, and uniform initial conditions are applied in each subdomain. Different initial conditions will develop into different structures [55,56]. The case 3 is initialized on a 2D domain $(x, y) \in [0, 1]^2$ as follows,

$$\begin{aligned} p = 0.3, \quad \rho = 0.5323, \quad \mathbf{u} = (1.206, 0), & \quad \text{in } 0 < x < 0.5, \quad 0.5 < y < 1, \\ p = 0.029, \quad \rho = 0.138, \quad \mathbf{u} = (1.206, 1.206), & \quad \text{in } 0 < x < 0.5, \quad 0 < y < 0.5, \\ p = 0.3, \quad \rho = 0.5323, \quad \mathbf{u} = (0, 1.206), & \quad \text{in } 0.5 < x < 1, \quad 0 < y < 0.5, \\ p = 1.5, \quad \rho = 1.5, \quad \mathbf{u} = (0, 0), & \quad \text{in } 0.5 < x < 1, \quad 0.5 < y < 1. \end{aligned} \quad (69)$$

The computational domain is decomposed into 100×100 square elements. In this problem, shocks are formed at the intersections between each of the two adjacent zones, and this pattern of four shocks moves towards the bottom-left corner of the domain as time progresses. This unsteady motion drives the development of the central structure. The contours of the density ρ and the blending coefficient α are displayed in Fig. 4 for the time $t = 0.3$. By increasing the polynomial order of DG approximation from $N = 4$ to $N = 8$, more details of the central structure are resolved as shown in the density contours. Additionally, with the increase in the polynomial order, the thickness of the shocks decreases as can be deduced from the regions with a high blending coefficient. Consequently, p -refinement, due to a better resolution of the local features, results in an overall smaller contribution of a low-order flux in a final solution. Remarkably, the developed hybrid methodology allowed for a 5-time increase in a stable CFL number as compared to an artificial dissipation method within the same split-form DGSEM-RK formulation (See Table 1 and Table 1 in Ref. [47]). Riemann problem case 3 is selected to test the sensitivity of the method to the two adjustable parameters in the blending coefficient scheme, c_1 and c_2 in Eq. (42), due to its interesting and sharp shock structures. Such sensitivity analysis is presented in Appendix C. It shows that the lower values of c_1 and higher values of c_2 than the selected baseline values of $c_1 = 2, c_2 = 20$ excessively smear out the sharp solution features, while higher values of c_1 and lower values of c_2 under-dissipate discontinuities and may lack robustness in problems with stronger shocks.

3.1.4. Double-Mach reflection

The double-Mach reflection case is a classical test case for hypersonic flows that features a formation of complicated structures related to discontinuities [57,58]. The configuration is identical to the one studied in Bohm et al. [59]. The problem is initialized on a 2D rectangular domain $[-0.25, 3] \times [0, 1]$ as

$$\begin{aligned} p = 116.5, \quad \rho = 8, \quad \mathbf{u} = (8.25 \cos(\pi/6), -8.25 \sin(\pi/6)), & \quad \text{in } y > \sqrt{3}x, \\ p = 1, \quad \rho = 1.4, \quad \mathbf{u} = (0, 0), & \quad \text{in } y \leq \sqrt{3}x. \end{aligned} \quad (70)$$

These initial conditions force an initial Mach 10 oblique shock in the domain. The shock is then driven by the time-varying upper boundary condition, whose values are set to describe the exact motion of the initial Mach 10 shock [57,59]

$$\begin{aligned} p = 116.5, \quad \rho = 8, \quad \mathbf{u} = (8.25 \cos(\pi/6), -8.25 \sin(\pi/6)), & \quad \text{in } x < (1 + 20t)/\sqrt{3}, \quad y = 1, \\ p = 1, \quad \rho = 1.4, \quad \mathbf{u} = (0, 0), & \quad \text{in } x \geq (1 + 20t)/\sqrt{3}, \quad y = 1. \end{aligned} \quad (71)$$

The rest of the boundary conditions is prescribed as follows. The left boundary uses Dirichlet boundary conditions, the right boundary is set to a Neumann boundary condition with zero derivatives. The bottom side is split into two portions: the left portion ($x \leq 0.25$) utilizes Dirichlet boundary conditions, and the right portion ($x > 0.25$) is set to be an inviscid wall. For the Dirichlet boundary conditions, the values from the initial solution

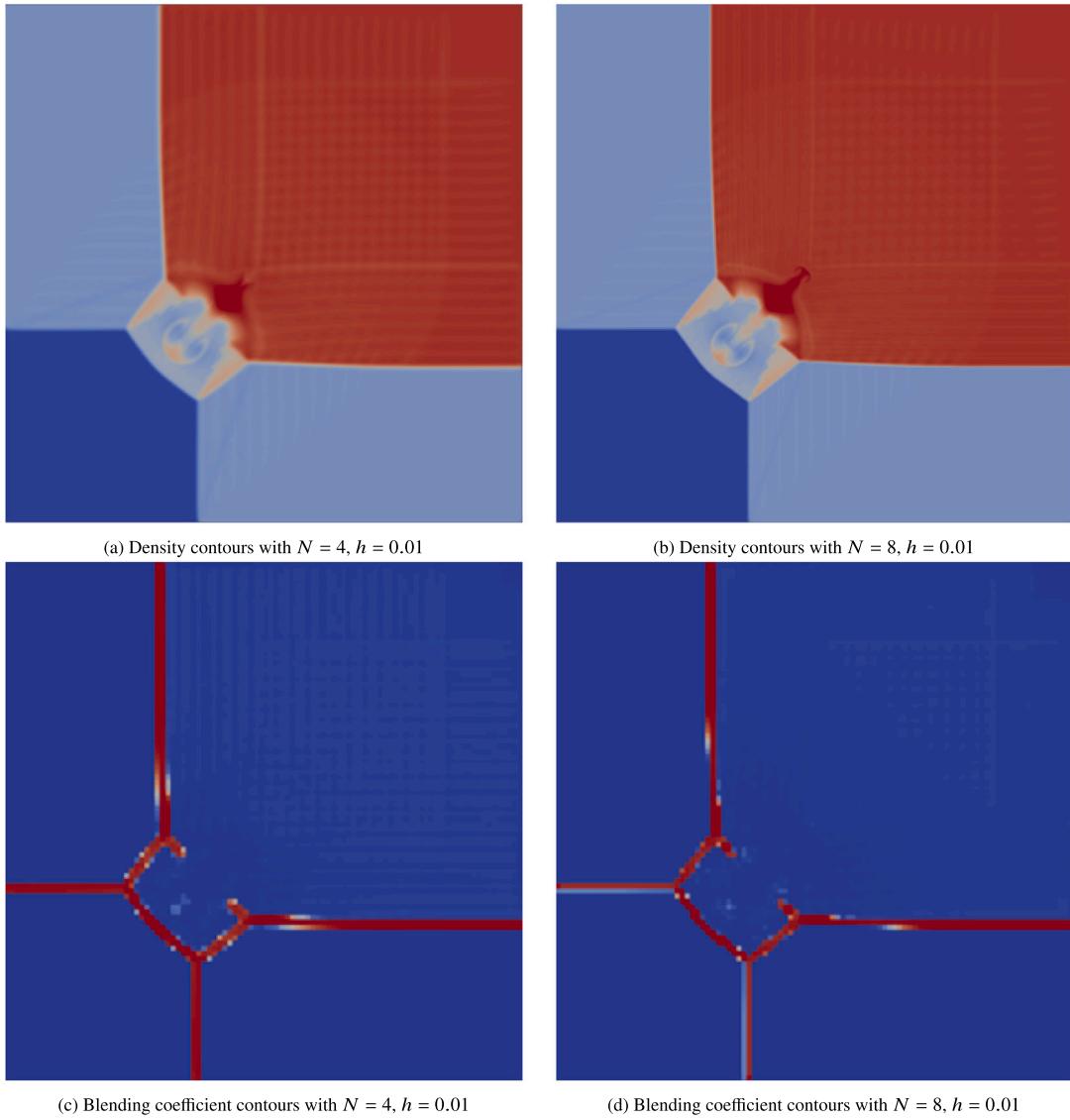


Fig. 4. (a), (b) Density (on a scale of 0.1 to 1.6), and (c), (d) blending coefficient (on a scale of 0 to 1) contours for the 2D Riemann problem case 3 at $t = 0.3$ at different resolutions for the baseline parameters, $c_1 = 2, c_2 = 20$.

(Eq. (70)) are used for both the left and the bottom left boundaries. The density and the blending coefficient contours are illustrated in Fig. 5 for $t = 0.2$. All the major structures, such as the bow shock, the triple points, the primary and the secondary shocks, and the slip line are identified in both $N = 4$ and $N = 8$ cases. As expected, the case with a higher polynomial order, $N = 8$, is able to resolve finer details of the small-scale structures along the slip line and yield thinner shocks as deduced from the contours of the blending coefficient. As in the previous example, increasing the polynomial order of approximation reduces the proportion of a low-order flux in the overall solution. We remark that a stable implementation of the current test case with an artificial viscosity method [47] was not achievable.

3.2. Test cases for Navier–Stokes equations

3.2.1. Interaction of shock wave with a laminar boundary layer

This test case computes an interaction of a shock wave with a laminar boundary layer in a steady setting. It has the flow conditions that are close to the following experimental [60] and numerical [61,62] studies. The domain configuration is illustrated in Fig. 6, where streamwise velocity contours of a steady-state solution are also displayed. An Illingworth solution [63] is prescribed for the velocity and temperature at the inflow. An oblique shock is generated at the inflow boundary by forcing the Rankine–Hugoniot relations. The pre-shock ($\{\cdot\}_1$) freestream has the uniform condition of $\text{Ma}_1 = 2.0$, $T_1 = 160.0$, $Re_{x_0} = 3 \times 10^5$, where x_0 is the distance from the leading edge of a plate. Setting the overall pressure ratio to a value of 1.4 as in [62] produces the corresponding shock angle $\beta_1 = 32.58^\circ$. The post-shock ($\{\cdot\}_2$) values are then computed from the following

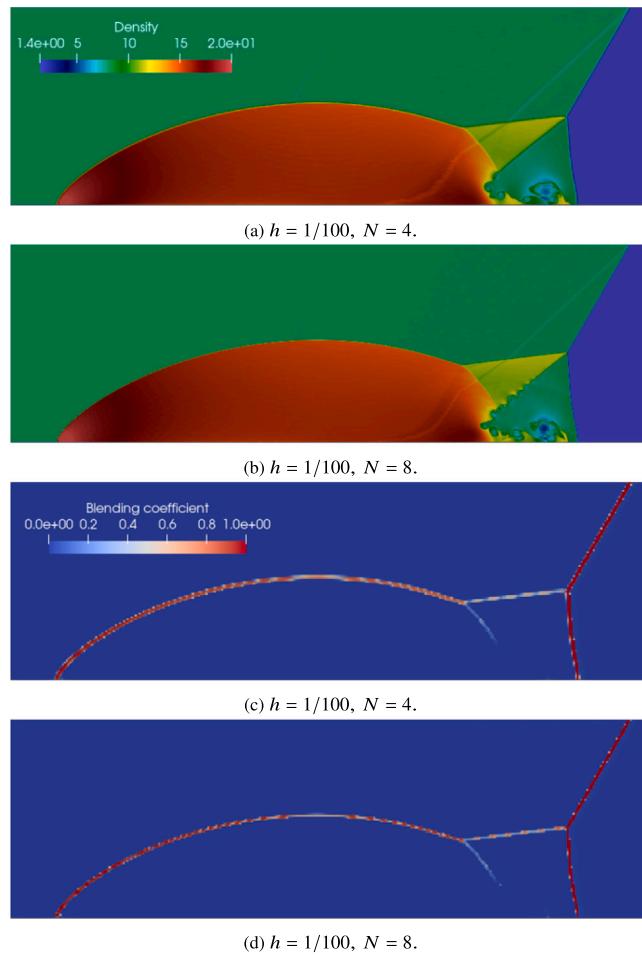


Fig. 5. (a), (b) Density, and (c), (d) blending coefficient contours for the double-Mach reflection case at $t = 0.2$ at different resolutions.

Rankine–Hugoniot relations,

$$\begin{aligned} Ma_2^2 &= \frac{Ma_1^2 + \frac{2}{\gamma-1}}{\frac{2\gamma}{\gamma-1} Ma_1^2 \sin^2 \beta_1 - 1} + \frac{Ma_1^2 \cos^2 \beta_1}{\frac{\gamma-1}{2} Ma_1^2 \sin^2 \beta_1 + 1}, \\ \frac{p_2}{p_1} &= 1 + \frac{2\gamma}{\gamma+1} (Ma_1^2 \sin^2 \beta_1 - 1), \\ \frac{\rho_2}{\rho_1} &= \frac{(\gamma+1)Ma_1^2 \sin^2 \beta_1}{2 + (\gamma-1)Ma_1^2 \sin^2 \beta_1}. \end{aligned} \quad (72)$$

The initial conditions correspond to the Illingworth solution inside the boundary layer [63] and the above oblique shock relations outside of the boundary layer. The bottom wall is considered adiabatic and no-slip. The upper and right boundaries are set to be outflows, and the boundary conditions can switch between subsonic and supersonic outflow conditions automatically, as needed. The distribution of pressure and skin friction coefficients as compared to the available experimental and numerical results [60–62] is plotted in Fig. 7. From Fig. 7(b), it can be noted that the experimental data downstream of the boundary layer separation deviates from all the numerical results for the skin friction coefficient. Overall, a good agreement with the existing data is achieved, especially with the results from Yao et al. [62].

3.2.2. Viscous shocktube at $Re = 1000$

The 2D viscous shocktube problem is an unsteady test case that verifies the capability of the method to resolve complex time-dependent interactions between a shock wave, a density wave and a boundary layer. Uniform mesh is used in this case. For each element, a fifth-order polynomial ($N = 6$) is employed to approximate the solution. To show the convergence trend, four meshes with a progressively increased refinement are tested in the current study. The number of DOFs for the four meshes range from 1500^2 to 2400^2 . The computational domain is a box of the size $[0, 1]^2$. The boundary conditions on all the four boundaries are set as adiabatic no-slip walls. The problem is initialized as follows,

$$\begin{aligned} p &= 120/\gamma, \quad \rho = 120, \quad \mathbf{u} = (0, 0), \quad \text{in } 0 < x < 0.5, \\ p &= 1.2/\gamma, \quad \rho = 1.2, \quad \mathbf{u} = (0, 0), \quad \text{in } 0.5 < x < 1. \end{aligned} \quad (73)$$

The numerical Schlieren contours are shown in Fig. 8 for the four refinement levels at $t = 1$. The solution on each mesh yields the overall similar patterns, especially for the contact discontinuity and the rightmost vortex. The major difference comes from the pattern in the separation

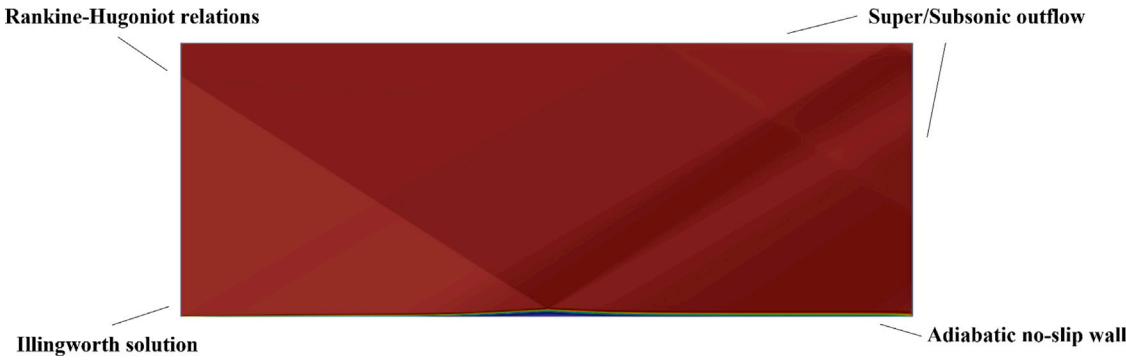


Fig. 6. Domain configuration and the boundary conditions for the shock–laminar boundary interaction problem. Contours of streamwise velocity corresponding to a steady-state solution are displayed.

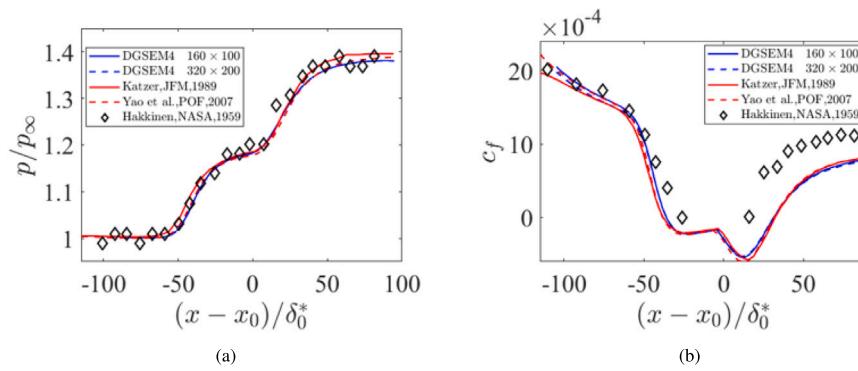


Fig. 7. Distributions of (a) pressure and (b) skin friction coefficient for the shock–laminar boundary interaction problem. δ_0^* denotes the inflow boundary layer thickness.

region. Qualitatively, we can observe the trend of convergence starting from Mesh 3, which can also be confirmed by the quantitative density distributions along the bottom wall in Fig. 9. In this figure, we also notice a good agreement with a numerical data from [64], computed with OSMP7 (seventh-order coupled space–time finite-volume scheme) and RK3-WENO5 (fifth-order WENO with the third-order RK).

3.2.3. 3D Taylor–Green vortex at $Re=1600$

This case is used to test the solver's capability to resolve turbulence. Following [65], the computation is carried out within a 3D box, $[0, 2\pi]^3$, and initialized with,

$$\begin{aligned} u &= U_0 \sin(x/L) \cos(y/L) \cos(z/L), \\ v &= -U_0 \cos(x/L) \sin(y/L) \cos(z/L), \\ w &= 0, \\ p &= p_0 + \rho_0 U_0^2 / 16 [\cos(2x/L) + \cos(2y/L)] [\cos(2z/L) + 2], \end{aligned} \tag{74}$$

where $U_0 = 1$, $L = 1$, $p_0 = 100$, $\rho_0 = 1$. The corresponding reference Mach number is 0.08.

The volume-averaged kinetic energy K and dissipation rate ϵ are computed as follows,

$$\begin{aligned} K(t) &= \frac{1}{2} \int_{\Omega} \rho |\mathbf{u}|^2 dV, \\ \epsilon(t) &= \frac{dK}{dt}. \end{aligned} \tag{75}$$

The normalized kinetic energy is defined as $K(t)/K(0)$. The evolution of the kinetic energy and the dissipation rate is plotted in Fig. 10. In the labels, $\{\}M \times N\epsilon$ gives the information on resolution, meaning M elements with N quadrature points per dimension. We observe that the result of our coarse mesh with 16×4 agrees well with the data obtained by the same resolution in [65] (labeled as "SD", which is a spectral difference method), and the result of our fine mesh with a resolution of 16×8 , which corresponds to 128 degrees of freedom per dimension, displays very close agreement to the reference data from [66] (labeled as "DRP", which is a 13-point dispersion relation preserving finite difference method that utilizes 512 degrees of freedom per dimension).

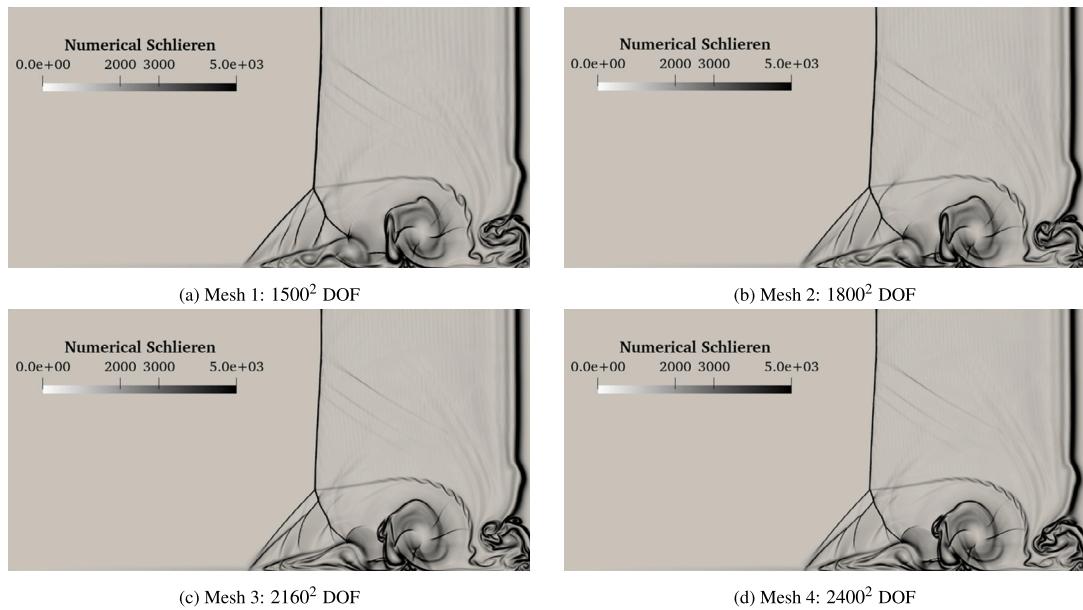


Fig. 8. Numerical Schlieren contours on the four levels of refinement at $t = 1$ for the viscous shocktube problem. Only the bottom half of the domain is displayed due to symmetry.

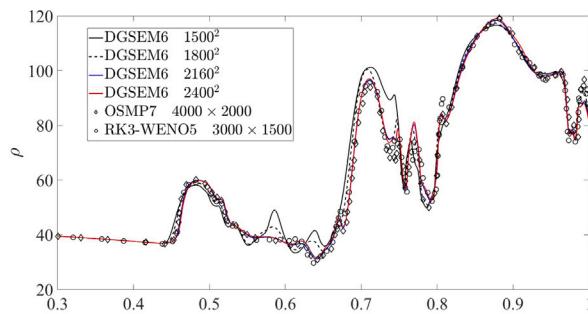


Fig. 9. Distribution of density along the bottom wall at $t = 1$ for the viscous shocktube problem. Lines represent DGSEM with $N = 6$ on four refinement levels. Symbols are from reference data [64] (OSMP7 and WENO5 schemes).

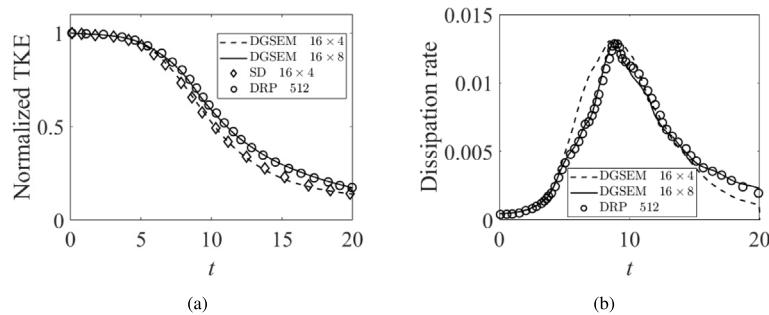


Fig. 10. Time evolution of (a) turbulent kinetic energy and (b) dissipation rate for the 3D Taylor-Green vortex. Lines are from the current DGSEM with varying resolution, symbols are from reference data (SD from [65], DRP from [66]).

3.2.4. Supersonic turbulent flow over a sphere

The final problem involves Direct Numerical Simulations of a supersonic turbulent flow over a sphere and is the apex of the current development. It demonstrates the applicability of the developed methodology to accurately capture compressible turbulence interacting with shocks on three-dimensional curvilinear meshes.

The computational domain is schematically shown in Fig. 11(a). A sphere of a diameter D is placed into the domain with its center located at $x = 0$, which is $5D$ downstream of the inlet boundary, while the outlet is located at $15D$ from the sphere center. The domain is cylindrical, with the outer cylinder radius of $10D$. Four sets of meshes are constructed for this study, with the total number of degrees of freedom summarized in Table 8. O-grid hexahedral mesh is utilized in each case, with a local refinement within the boundary layer of the sphere and in the wake region. The O-grid mesh in the vicinity of the sphere is schematically illustrated in Fig. 11(b) for the coarsest mesh (Mesh 1). The solid boundary

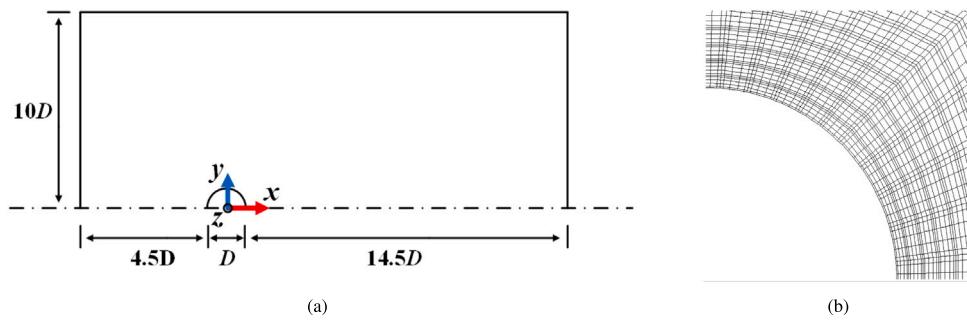


Fig. 11. Computational setup for a DNS simulation of a supersonic flow around a sphere. (a) A schematic of the computational domain (only upper half is shown due to symmetry), (b) A close-up view of the computational grid around the sphere for the coarsest mesh (Mesh 1).

of a sphere is approximated with 5-th order polynomials defined on the GLL points on the surface of the sphere. This accuracy of the surface approximation corresponds to the accuracy of the interior solution, which also utilizes 5-th order polynomials ($N = 6$). The flow parameters are set to $Re_D = 6.5 \times 10^5$, $Ma_\infty = 2.0$, $p_\infty = 716.6264 \text{ Pa}$, which corresponds to the flow conditions investigated in the studies of [67–70] utilized as a reference data, see Table 9. Boundary conditions in the current computations involve a supersonic inflow at the inlet, supersonic outflow at the outlet as well as the lateral cylindrical boundaries of the domain, while no-slip adiabatic boundary conditions are prescribed on the sphere surface. The CFL number employed for the simulations on each mesh is listed in Table 8; it is slightly reduced with the refinement level, but not dramatically, staying above the value of 0.38 for all the cases. All the simulations are advanced to a statistically-steady turbulent state, after which statistics is collected for a time of $\Delta t_{\text{avg}} U_\infty / D = 155.0$ for each mesh. All simulations were run on Sapphire Rapids (SPR) nodes of Stampede 3 supercomputer of the Texas Advanced Computing Center. SPR nodes feature Intel Xeon CPU MAX 9480 processors containing 112 cores on two sockets with 1.9 Hz clock rate and 128 GB memory per node. Each mesh was run utilizing 32 SPR nodes (3384 cores) of Stampede 3. The total number of core hours and the total wall-clock time required to run the simulations (on 3384 cores) is included in Table 8 for the four meshes. As the meshes are refined, the computational time increases, and this increase is faster than the increase in the number of the degrees of freedom (spatial nodes), since the time step is also reduced as the meshes are refined, to keep the CFL number constant. Since the meshes are refined non-uniformly, the same can be said about the time step. For example, Mesh 1 is quite sparse in the neighborhood of the sphere (in the boundary layer and in the near-wake region), allowing for significantly larger time steps. This makes this coarse mesh comparatively inexpensive with respect to the more refined meshes, at the expense of the solution accuracy as shown later. While a direct comparison of computational costs with the other compressible flow solvers (e.g., the ones previously utilized in a DNS of a Mach 2 flow past a sphere — see Table 9) is complicated due to a lack of such data regarding the solvers [68–70], we can compare the costs with the incompressible version of the SEM code in the DNS simulations of the flow past a sphere [71]. In the incompressible SEM, the mesh with approximately 10 mln. DOFs required about 16.6K core hours as compared to a similar time of 18.2K core hours on a 12.3 mln. mesh with the present DGSEM code for the equivalent number of time steps. We note, however, that the Reynolds number in Ref. [71] for the incompressible sphere case was $Re_D = 3.7 \times 10^3$ as opposed to $Re_D = 6.5 \times 10^5$ in the current case. Therefore, the current compressible solver allows for a utilization of similar computational resources to perform DNS of flows with significantly higher Reynolds numbers.

The results of the flow field are presented in Fig. 12, where the temperature and streamwise velocity fields from the current DGSEM simulations on Mesh 3 (Figs. 12(b) and 12(c)) are compared with a schematical representation of the main flow features for this Mach number as summarized in the experimental work of [72] (Fig. 12(a)). It can be seen that all the main structures, such as a bow shock, expansion waves, and a reattachment shock are all captured in the current DNS simulations. A three-dimensional view of the wake structure visualized by the λ_{ci} criterion [73] interacting with the bow shock and the reattachment shock is presented in Fig. 13. The structure of the wake in the current simulations first converges towards the centerline behind the recirculation region and then gradually expands in the radial direction. The wake structure agrees well with the schematic of [72] and with the instantaneous flow structures from the DNS simulations presented in Ref. [69] (Figure 39) and Ref. [70] (Figure 6). We note that the recirculation region computed in [68] (Figure 22) is shorter and wider, perhaps due to the effect of the slip condition on the sphere.

A quantitative comparison of the total drag coefficient on the sphere surface and the shock stand distance with the existing experimental [67] and numerical data [68–70] is shown in Table 9. The table also documents the methods, domain sizes and the number of degrees of freedom employed for the computation of the reference data (whenever applicable). It can be noted that both the shock stand distance and the total drag coefficient agree well with the published data, even on the coarsest mesh. Note that the computational domain for the two IBM methods (without an adaptive mesh refinement) [68,70] is about 25 times smaller than the current domain by volume, which needs to be taken to account while comparing the effective number of degrees of freedom. As an additional remark, the drag coefficient in Ref. [68] is slightly higher and the shock stand distance is lower compared to all the other data, consistent with the shorter extent of the recirculation region computed in the simulations of [68]. The authors of [68] attribute this effect to resolution limitations that affect the size of the low pressure region in the wake behind the sphere and the state of the boundary layer before separation that could affect the transition region, which is potentially not modeled accurately with slip-type boundary conditions.

Figs. 14 and 15 document the results of the grid convergence studies with the current method. In Fig. 14, a temporally averaged streamwise velocity along the domain centerline is plotted for the four mesh levels, and in Fig. 15 the radial profiles of azimuthally and temporally averaged streamwise velocity are compared at different streamwise locations. The results demonstrate a good convergence of the turbulent flow statistics with grid refinement. The presented test problem illustrates the capability of the developed high-order methodology to accurately capture turbulent flow features interacting with shocks and discontinuities on a curvilinear mesh. A more in-depth analysis of the physical properties of the supersonic flow over a sphere and a shock-turbulence interaction in curvilinear geometries is planned for the future work.

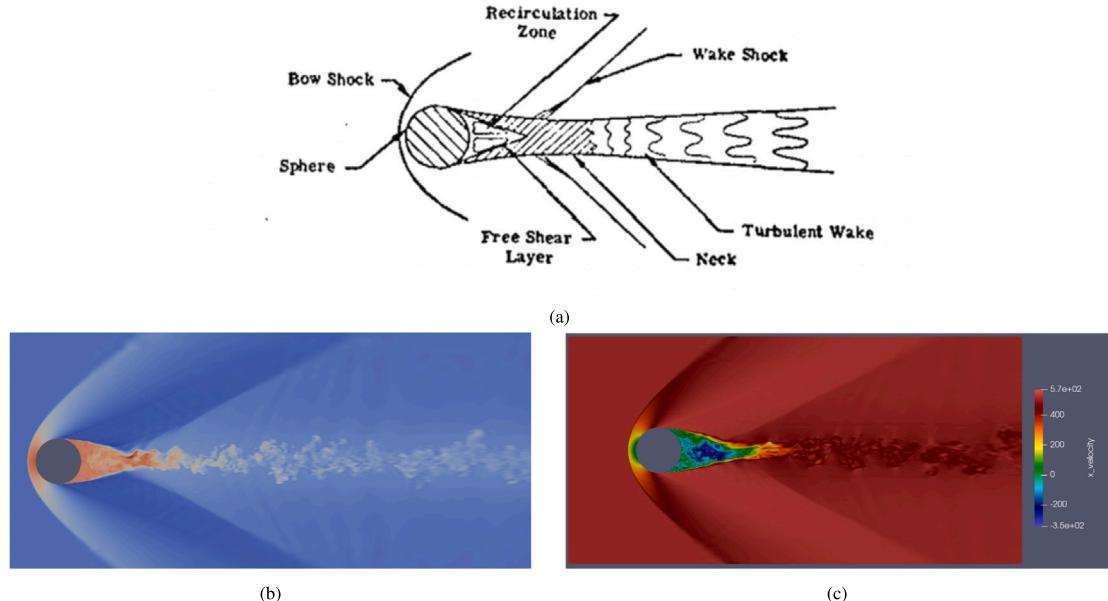


Fig. 12. Main flow structures observed in a supersonic turbulent flow over a sphere at $Ma = 2$. (a) A schematic of the flow as outlined in experiments of [72], (b) Instantaneous temperature field, temperature range T/T_∞ is from 0.81 (blue) to 2.02 (red), and (c) Instantaneous streamwise velocity field (m/s) from the current DGSEM simulations on Mesh 3.

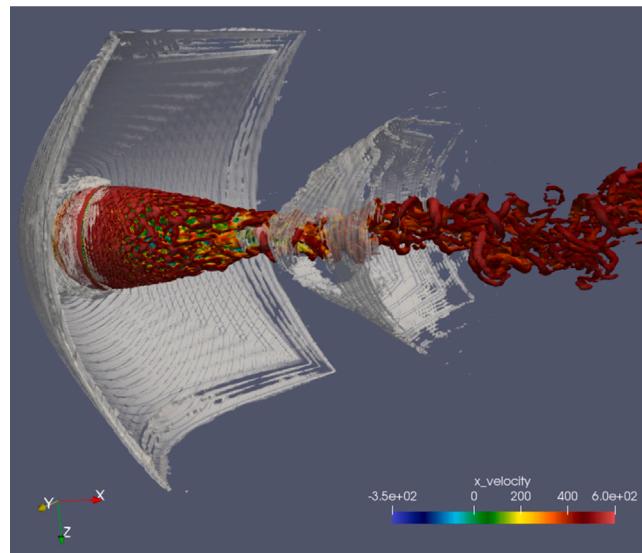


Fig. 13. Flow structures visualized by $\lambda_{ci} = 7.2U_\infty^2/D^2$ (colored by streamwise velocity) and $\|\nabla p\| = 7.0p_\infty/D$ (in white) on Mesh 3.

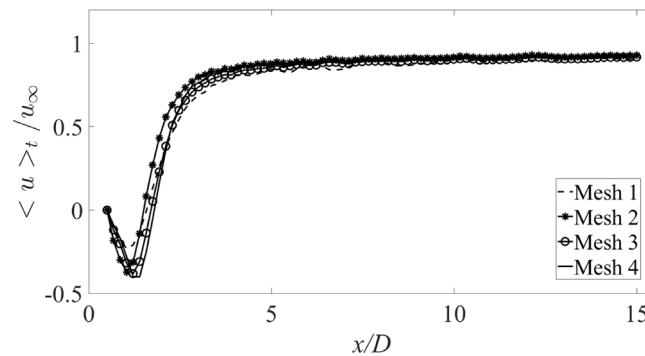


Fig. 14. Temporally averaged streamwise velocity distribution along the domain centerline for the supersonic sphere case on four mesh refinement levels.

Table 8

Computational parameters and computational cost for DNS on different grid refinement levels. Computational time is reported on SPR nodes of Stampede 3 supercomputer. The wall-clock time is calculated using 32 nodes (3384 cores) for each mesh.

Mesh	#DOFs (mln.)	CFL	$\Delta t_{\text{avg}} U_{\infty} / D$	Core hours (K)	Wall-clock time (hours)
Mesh 1	12.3	0.45	155.0	18.228	5
Mesh 2	41.0	0.40	155.0	209.62	60
Mesh 3	53.6	0.38	155.0	416.09	116
Mesh 4	68.7	0.38	155.0	872.91	244

Table 9

Comparison of the shock stand distance (L_{shock}/D) and the total drag coefficient (C_d) between the current DNS and the reference data. “IBM” stands for “Immersed Boundary Method”, “AMR” - “Adaptive Mesh Refinement”.

Cases	Domain size	L_{shock}/D	C_d
Experiment [67]: free-flight tests	-	-	1.00
DNS [68]: IBM slip (33.5 mln. DOFs)	$10D \times 5D \times 5D$	0.180	1.07
DNS [69]: IBM AMR (varying DOFs)	$128D \times 64D \times 64D$	0.183	0.96
DNS [70]: ghost-zone IBM (33.5 mln. DOFs)	$10D \times 5D \times 5D$	0.185	0.96
Current Mesh 1 (12.3 mln. DOFS)	$20D \times 10\pi D \times 10D$	0.181	0.98
Current Mesh 2 (41.0 mln. DOFS)	$20D \times 10\pi D \times 10D$	0.183	0.97
Current Mesh 3 (53.6 mln. DOFS)	$20D \times 10\pi D \times 10D$	0.184	0.97
Current Mesh 4 (68.7 mln. DOFS)	$20D \times 10\pi D \times 10D$	0.184	0.97

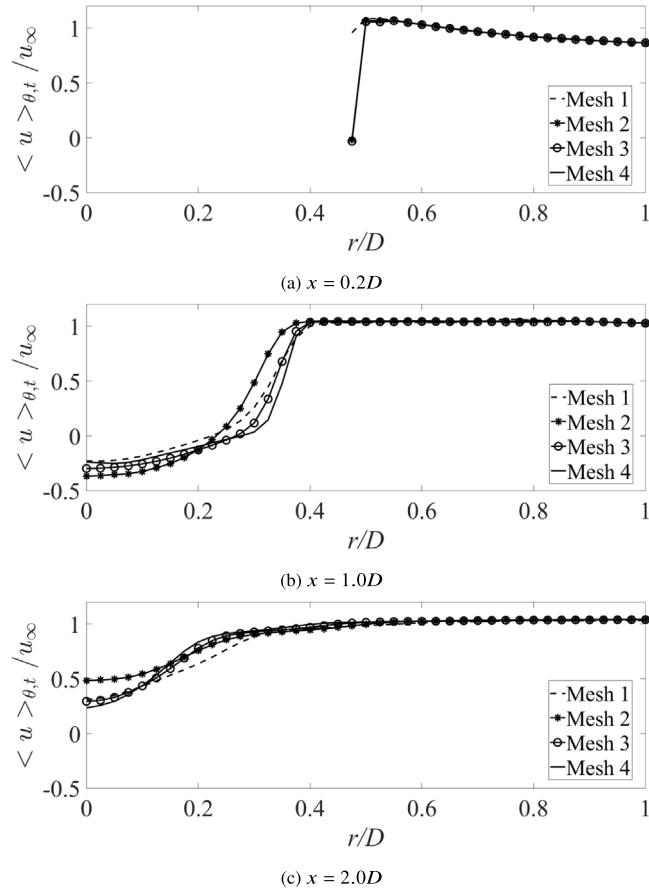


Fig. 15. Radial profiles of azimuthally and temporally averaged streamwise velocity at different streamwise locations: (a) $x = 0.2D$, (b) $x = 1.0D$, and (c) $x = 2.0D$ on four mesh refinement levels.

4. Conclusions

In the current study, a high-order computational methodology for 3D turbulent flows with shocks in complex geometries is developed upon a framework of the split-form DGSEM with SBP operators. For the discretization of the viscous terms, the Bassi and Rebay [37] (BR1) scheme is used. For shock capturing, we implemented an adaptive sub-element order reduction methodology based on a mixed approximation space that contains a blend of high-order Lagrange polynomials and piecewise-constant basis functions with a sub-element support. An amount of blended low-order approximation is determined by a modified Ducros sensor [40], which is able to reliably detect shocks in viscous compressible turbulent flows while not acting upon local high shear spots, dilatational turbulent shocklets and expansion waves. The blending scheme has only two adjustable

parameters and is robust in the sense that all the test cases were successfully run with the same blending coefficient settings, $c_1 = 2.0$ and $c_2 = 20.0$. A sensitivity study with respect to the two adjustable parameters demonstrated that the lower values of c_1 and higher values of c_2 lead to an excessive dissipation of the structures with sharp gradients, while higher values of c_1 and lower values of c_2 result in finer shocks and may cause instabilities, thus justifying the current parameter choice. A high-order flux is discretized with the kinetic energy preserving and entropy conserving (KEPEC) scheme [49], and a low-order sub-element flux (also based on the KEPEC scheme with a Rusanov-type dissipation) is corrected in inner subcells to second-order by utilizing the second-order TVD minmod reconstruction utilizing the values from the neighboring subcells.

The methodology is tested using eight numerical problems. An expected polynomial order of convergence is demonstrated on both Cartesian and curvilinear meshes for the isentropic Euler vortex problem. The effect of CFL number on spatial convergence is demonstrated as well. When a fine resolution is needed, the time step needs to be refined accordingly to achieve an optimal convergence, or a higher-order temporal discretization can also be recommended in such cases. In the 1D discontinuous Sod shocktube test case, results of the convergence study are close to the theoretical values. 2D Riemann problem case 3 demonstrates a highly-accurate solution with 5 times larger allowable time step for stability as compared to the entropy viscosity shock capturing method in our previous work [47]. In a hypersonic double-Mach reflection case ($Ma = 10$), complex shock structures and contact points are all identified and well resolved. For the shock-wave/laminar boundary layer problem, a good agreement with existing data is observed. Time-dependent interactions of a shock wave, a density wave, and a boundary layer are simulated in the viscous shocktube case. The main structures and the density distribution along the bottom wall agree well with the reference results. The developed method demonstrates an excellent capability of resolving turbulence in the 3D Taylor–Green vortex case by showing a good agreement of our results computed using 128 degrees of freedom with the reference data computed by high-order finite differences with 512 degrees of freedom. Finally, we conduct a comprehensive test case that involves Direct Numerical Simulations of a supersonic turbulent flow over a sphere and features complex interactions of turbulence with shocks in a curvilinear geometry. The instantaneous flow structures, as well as the shock stand distance and the drag coefficient values show excellent agreement with existing experimental and numerical data. We also demonstrate the grid convergence for velocity profiles in the wake area on a series of four meshes. Overall, all considered test cases demonstrate high solution accuracy and robust shock capturing capabilities of the hybrid sub-element order reduction DGSEM methodology while significantly alleviating stability constraints on a time step as compared to an artificial viscosity method implemented previously by the authors [47]. The developed methodology is well suited for accurate high-fidelity simulations of compressible viscous laminar and turbulent flows in the presence of strong shocks and discontinuities, hypersonic flow conditions, and complex geometries. The presented methodology is implemented in the open-source spectral-element software Nek5000 [74].

We would like to point out that, while the developed methodology was demonstrated to be robust across the test cases presented, it has potential limitations. One such limitation consists of possible instabilities if the methodology is applied to very high Mach number ($Ma \gtrsim 10$) viscous flows. This can potentially be improved if the high-/low-order blending scheme is also extended to viscous fluxes, which are currently treated uniformly with the high-order BR1 scheme, to be pursued in future work. Another limitation is a choice of an element-wise blending coefficient in Eq. (43) that can potentially limit the local sub-element resolution of the shocks. Some recent approaches that involve designing a sub-element scheme for a blending coefficient in DGSEM that involves solution of a local optimization problem [34] can be further explored in the context of high-speed viscous turbulent flows. Finally, there are potential challenges in extending the current methodology to extremely high Reynolds numbers and resolving fine-scale turbulent structures in highly complex geometries. Due to localized fine-scale solution features expected in these situations, advanced techniques such as adaptive mesh refinement in conjunction with blended high-/low-order shock capturing would prove useful and will be pursued in future work.

CRediT authorship contribution statement

Fengrui Zhang: Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Data curation, Conceptualization.
Yulia T. Peet: Writing – review & editing, Supervision, Resources, Project administration, Funding acquisition, Formal analysis, Conceptualization.

Declaration of 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.

Acknowledgments

This work was supported by NSF CAREER-1944568, NSF AGS-2207115 and AFOSR FA9550-23-1-0762 awards. Computing time on Stampede 3 supercomputer of the Texas Advanced Computing Center enabled by NSF ACCESS program is acknowledged.

Appendix A. Derivation of split-form DGSEM discretization

To arrive at Eq. (24) in Section 2.5.1, we start with the definition,

$$\left(\nabla_{\xi} \cdot \tilde{\mathbf{F}}, \phi_{ijk}^H \right)_{\Omega} = \int_{-1}^1 \int_{-1}^1 \int_{-1}^1 \left(\sum_{l=1}^3 \frac{\partial \tilde{\mathbf{F}}^l}{\partial \xi^l} \right) \phi_{ijk}^H d\xi^1 d\xi^2 d\xi^3. \quad (\text{A.1})$$

For the integral in the right-hand side of the above equation, we can change the order of the integration and the summation operators, and consider the component with $l = 1$. We apply integration-by-parts,

$$\int_{-1}^1 \int_{-1}^1 \int_{-1}^1 \frac{\partial \tilde{\mathbf{F}}^1}{\partial \xi^1} \phi_{ijk}^H d\xi^1 d\xi^2 d\xi^3 = \int_{-1}^1 \int_{-1}^1 \left(\tilde{\mathbf{F}}^1 \phi_{ijk}^H \right) \Big|_{\xi^1=-1}^1 d\xi^2 d\xi^3 - \int_{-1}^1 \int_{-1}^1 \int_{-1}^1 \frac{\partial \phi_{ijk}^H}{\partial \xi^1} \tilde{\mathbf{F}}^1 d\xi^1 d\xi^2 d\xi^3. \quad (\text{A.2})$$

In the same way, we can manipulate the other two components in the summation operator in Eq. (A.1). Subsequently, we employ the GLL quadrature and the cardinality property of the Lagrangian basis functions, to yield a discretization,

$$\begin{aligned} \left(\nabla_{\xi} \cdot \tilde{\mathbf{F}}, \phi_{ijk}^H \right)_{\hat{\Omega}} &= w_j w_k \left(l_i(1) \tilde{\mathbf{F}}_{Njk}^{1*} - l_i(-1) \tilde{\mathbf{F}}_{1jk}^{1*} \right) \\ &\quad + w_i w_k \left(l_j(1) \tilde{\mathbf{F}}_{ink}^{2*} - l_j(-1) \tilde{\mathbf{F}}_{1ik}^{2*} \right) + w_i w_j \left(l_k(1) \tilde{\mathbf{F}}_{ijN}^{3*} - l_k(-1) \tilde{\mathbf{F}}_{ij1}^{3*} \right) \\ &\quad - \sum_{n=1}^N \tilde{\mathbf{F}}_{njk}^1 l'_i(\xi_n^1) w_n w_j w_k - \sum_{n=1}^N \tilde{\mathbf{F}}_{ink}^2 l'_j(\xi_n^2) w_n w_i w_k - \sum_{n=1}^N \tilde{\mathbf{F}}_{ijN}^3 l'_k(\xi_n^3) w_n w_i w_j, \end{aligned} \quad (\text{A.3})$$

where $\tilde{\mathbf{F}}^*$ is the surface flux calculated as an approximate solution to a Riemann problem [75]. At this stage, we have discretized the convective term following the standard DGSEM formulation.

To introduce the split-form DGSEM, we start with the SBP operator derived from the following equation,

$$\int_{-1}^1 l_n(\xi) l'_i(\xi) d\xi = [l_n(\xi) l_i(\xi)]_{-1}^1 - \int_{-1}^1 l'_n(\xi) l_i(\xi) d\xi. \quad (\text{A.4})$$

After GLL quadrature is applied to discretize the above integrals, the integral-by-parts operation is transferred into a summation-by-parts operation,

$$\sum_{k=1}^N l_n(\xi_k) l'_i(\xi_k) w_k = l_n(\xi_N) l_i(\xi_N) - l_n(\xi_1) l_i(\xi_1) - \sum_{k=1}^N l'_n(\xi_k) l_i(\xi_k) w_k. \quad (\text{A.5})$$

Because of the cardinality property of the Lagrangian basis function, the above equation can be further manipulated,

$$l'_i(\xi_n) w_n = l_n(\xi_N) l_i(\xi_N) - l_n(\xi_1) l_i(\xi_1) - l'_n(\xi_i) w_i. \quad (\text{A.6})$$

We can then substitute the derivative of the basis function in Eq. (A.3) with the above relation, and subsequently apply the cardinality property of the Lagrangian basis function. The first summation term in Eq. (A.3) becomes,

$$\begin{aligned} \sum_{n=1}^N \tilde{\mathbf{F}}_{njk}^1 l'_i(\xi_n^1) w_n w_j w_k &= \sum_{n=1}^N \tilde{\mathbf{F}}_{njk}^1 \left[l_n(\xi_N^1) l_i(\xi_N^1) - l_n(\xi_1^1) l_i(\xi_1^1) - l'_n(\xi_l^1) w_l \right] w_j w_k \\ &= w_j w_k \left[\tilde{\mathbf{F}}_{Njk}^1 l_i(\xi_N^1) - \tilde{\mathbf{F}}_{1jk}^1 l_i(\xi_1^1) \right] - w_i w_j w_k \sum_{n=1}^N \tilde{\mathbf{F}}_{njk}^1 l'_n(\xi_i^1). \end{aligned} \quad (\text{A.7})$$

As shown in [7,15,76], because of the summation-by-parts property of the Lagrangian basis function (Eq. (A.6)), the right-most term in Eq. (A.7) can be reformulated in a telescopic flux form,

$$\sum_{n=1}^N \tilde{\mathbf{F}}_{njk}^1 l'_n(\xi_i^1) = \frac{\tilde{\mathbf{F}}_{(i+1)jk}^1 - \tilde{\mathbf{F}}_{ijk}^1}{w_i}, \quad (\text{A.8})$$

where $\tilde{\mathbf{F}}$ are the auxiliary fluxes defined at the GLL points. It was proven [15,76] that the telescopic flux form can be further extended to a high-order approximation as

$$\frac{\tilde{\mathbf{F}}_{(i+1)jk}^1 - \tilde{\mathbf{F}}_{ijk}^1}{w_i} = 2 \sum_{n=1}^N \tilde{\mathbf{F}}^{1\#}(\mathbf{U}_{njk}, \mathbf{U}_{ijk}) l'_n(\xi_i^1), \quad (\text{A.9})$$

where $\tilde{\mathbf{F}}^{1\#}(\mathbf{U}_{njk}, \mathbf{U}_{ijk})$ is a suitable two-point flux function. We can approximate the other two summation terms in Eq. (A.3) with the same method. Plugging Eq. (A.7), (A.8), (A.9) back into Eq. (A.3), we obtain Eq. (24).

Appendix B. Details of the discretization of diffusion terms

Starting from Eq. (17b) and noting that the vector gradient $\tilde{\nabla}_{\xi} \mathbf{U}$ in curvilinear coordinates can be rewritten as

$$\tilde{\nabla}_{\xi} \mathbf{U} = \sum_{l=1}^3 J \mathbf{a}^l \frac{\partial \mathbf{U}}{\partial \xi^l} = \sum_{l=1}^3 \left(\frac{\partial J \mathbf{a}^l \mathbf{U}}{\partial \xi^l} - \mathbf{U} \frac{\partial J \mathbf{a}^l}{\partial \xi^l} \right) = \sum_{l=1}^3 \frac{\partial J \mathbf{a}^l \mathbf{U}}{\partial \xi^l}, \quad (\text{B.1})$$

due to a vector identity [46],

$$\sum_{l=1}^3 \frac{\partial J \mathbf{a}^l}{\partial \xi^l} = 0, \quad (\text{B.2})$$

we obtain Eq. (48) in Section 2.6. After separating the terms in Eq. (48) into a “high-order” and a “low-order” component, for the inner products of the high-order component (the first term in the right-hand side of Eq. (49)), let us take $l = 1$ as an example (the other two directions share a similar form). To evaluate the inner product, we integrate by parts,

$$\begin{aligned} \left(\frac{\partial J \mathbf{a}^1 \mathbf{U}}{\partial \xi^1}, \phi_{ijk}^H \right)_{\hat{\Omega}} &= \int_{-1}^1 \int_{-1}^1 \left(\int_{-1}^1 \frac{\partial J \mathbf{a}^1 \mathbf{U}}{\partial \xi^1} l_i(\xi) d\xi \right) l_j(\eta) l_k(\zeta) d\eta d\zeta \\ &= \int_{-1}^1 \int_{-1}^1 \left[(\mathbf{J} \mathbf{a}^1 \mathbf{U})_{\xi=1} l_i(1) - (\mathbf{J} \mathbf{a}^1 \mathbf{U})_{\xi=-1} l_i(-1) \right] l_j(\eta) l_k(\zeta) d\eta d\zeta \\ &\quad - \int_{-1}^1 \int_{-1}^1 \int_{-1}^1 \mathbf{J} \mathbf{a}^1 \mathbf{U} l'_i(\xi) l_j(\eta) l_k(\zeta) d\xi d\eta d\zeta. \end{aligned} \quad (\text{B.3})$$

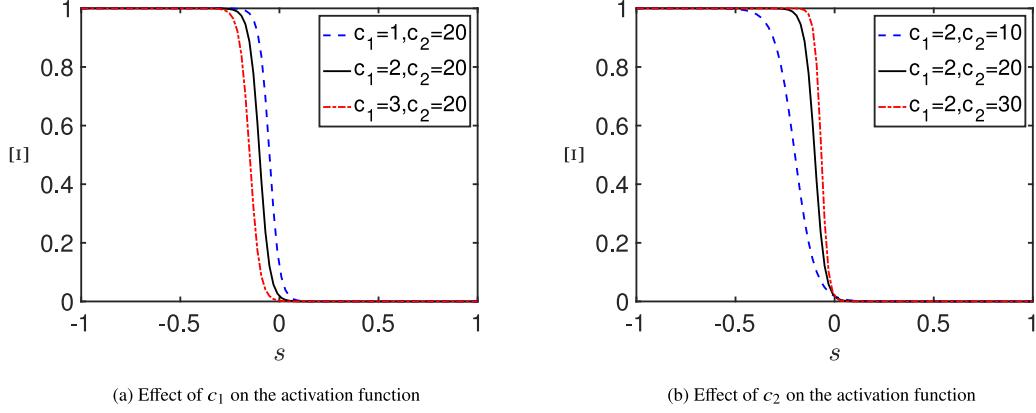


Fig. C.16. Shape of the activation function, $\Xi(s) = 0.5 [1 - \tanh(c_1 + c_2 s)]$, for different values of parameters c_1 and c_2 .

We then discretize the integrals with the GLL quadrature rule and apply the cardinality property of the Lagrangian basis functions to yield

$$\begin{aligned} \left(\frac{\partial J\mathbf{a}^1 \mathbf{U}}{\partial \xi^1}, \phi_{ijk}^H \right)_{\hat{\Omega}} &= \left[(J\mathbf{a}^1 \mathbf{U})_{Njk}^* l_i(1) - (J\mathbf{a}^1 \mathbf{U})_{1jk}^* l_i(-1) \right] w_j w_k \\ &\quad - \sum_{n=1}^N (J\mathbf{a}^1 \mathbf{U})_{njk} l'_i(\xi_n) w_n w_j w_k, \end{aligned} \quad (\text{B.4})$$

where $(J\mathbf{a}^1 \mathbf{U})_{\{1,N\}jk}^*$ represent surface fluxes at the element surfaces. We employ BR1 scheme [37] to discretize the surface fluxes as

$$(J\mathbf{a}^1 \mathbf{U})^* = \{(J\mathbf{a}^1 \mathbf{U})\}_{+,-} = \frac{1}{2} \left((J\mathbf{a}^1 \mathbf{U})^+ + (J\mathbf{a}^1 \mathbf{U})^- \right), \quad (\text{B.5})$$

where $(\cdot)^+$ and $(\cdot)^-$ notations were introduced in Section 2.5.1. Following [37], we define the jumps across the interfaces as in Eq. (51). Substituting Eq. (51) into Eq. (B.5) and, subsequently, Eq. (B.4), we obtain

$$\begin{aligned} \left(\frac{\partial J\mathbf{a}^1 \mathbf{U}}{\partial \xi^1}, \phi_{ijk}^H \right)_{\hat{\Omega}} &= \left[(J\mathbf{a}^1 \mathbf{U})_{Njk}^{**} l_i(1) - (J\mathbf{a}^1 \mathbf{U})_{1jk}^{**} l_i(-1) \right] w_j w_k \\ &\quad + \left[(J\mathbf{a}^1 \mathbf{U})_{Njk} l_i(1) - (J\mathbf{a}^1 \mathbf{U})_{1jk} l_i(-1) - \sum_{n=1}^N (J\mathbf{a}^1 \mathbf{U})_{njk} l'_i(\xi_n) w_n \right] w_j w_k, \end{aligned} \quad (\text{B.6})$$

We further apply the SBP operator to the terms in the second square bracket of Eq. (B.6) to yield a final form of the discretization as in Eq. (50) and, subsequently, Eq. (52).

To discretize the high-order viscous term in Eq. (54), we take Eq. (55) as a starting point and restrict our attention to the first component in the sum. After integrating by parts, one obtains

$$\begin{aligned} \left(\frac{\partial \tilde{\mathbf{F}}_V^1}{\partial \xi^1}, \phi_{ijk}^H \right)_{\hat{\Omega}} &= \int_{-1}^1 \int_{-1}^1 \left[(\tilde{\mathbf{F}}_V^1)_{\xi=1} l_i(1) - (\tilde{\mathbf{F}}_V^1)_{\xi=-1} l_i(-1) \right] d\eta d\zeta \\ &\quad - \int_{-1}^1 \int_{-1}^1 \int_{-1}^1 \tilde{\mathbf{F}}_V^1 l_i(\xi) l_j(\eta) l_k(\zeta) d\xi d\eta d\zeta. \end{aligned} \quad (\text{B.7})$$

To obtain a discrete form, we, again, utilize the GLL quadrature rule and the cardinality property of the Lagrangian basis functions to yield Eq. (56).

The other two components in the sum on the right-hand side of Eq. (55) are discretized following the same procedure that yields the final discretization of the diffusion term in Eq. (58) and, ultimately, Eq. (59).

Appendix C. Sensitivity to adjustable parameters

The sensitivity of the blending scheme to the two adjustable parameters c_1 and c_2 in the modified Ducros sensor is tested here on the example of a Riemann problem case 3 as a representative problem with strong two-dimensional shocks. The parameters control the shape of the activation function $\Xi(s)$ (Eq. (42)) that multiplies the Ducros sensor (Eq. (41)) in the blending coefficient scheme (Eq. (40)). The shape of the activation function for the baseline values, $c_1 = 2$, $c_2 = 20$, and for the tested off-design values $c_1 = \{1, 3\}$, $c_2 = \{10, 30\}$ is compared in Fig. C.16. The effect of the parameter c_1 is to shift the activation function to the right for lower c_1 and to the left for higher c_1 . The shift to the right indicates that small positive values of the argument (positive dilatation) now activate the shock sensor and allow for a low-order flux blending which introduces excessive damping of expansion waves. The shift to the left, to the contrary, under-dissipates the regions of negative dilatation (compression waves), resulting in a more aggressive refinement of shock structures. The effect of c_2 is similar, providing more damping at higher c_2 (right shift) and less damping at lower c_2 (left shift), albeit the shape of $\Xi(s)$ with lower c_2 is more affected.

Blending coefficient for the Riemann problem case 3 from simulations with different values of c_1 and c_2 is shown in Fig. C.17 for $h = 0.01$, $N = 4$ and $t = 0.3$, which are the same parameters as in Fig. 4(c) (except for c_1 and c_2 values). Compared to the baseline values of $c_1 = 2$, $c_2 = 20$ in Fig. 4(c), keeping c_2 constant and lowering c_1 diffuses the shock, while keeping c_2 constant and increasing c_1 makes the shock sharper. These

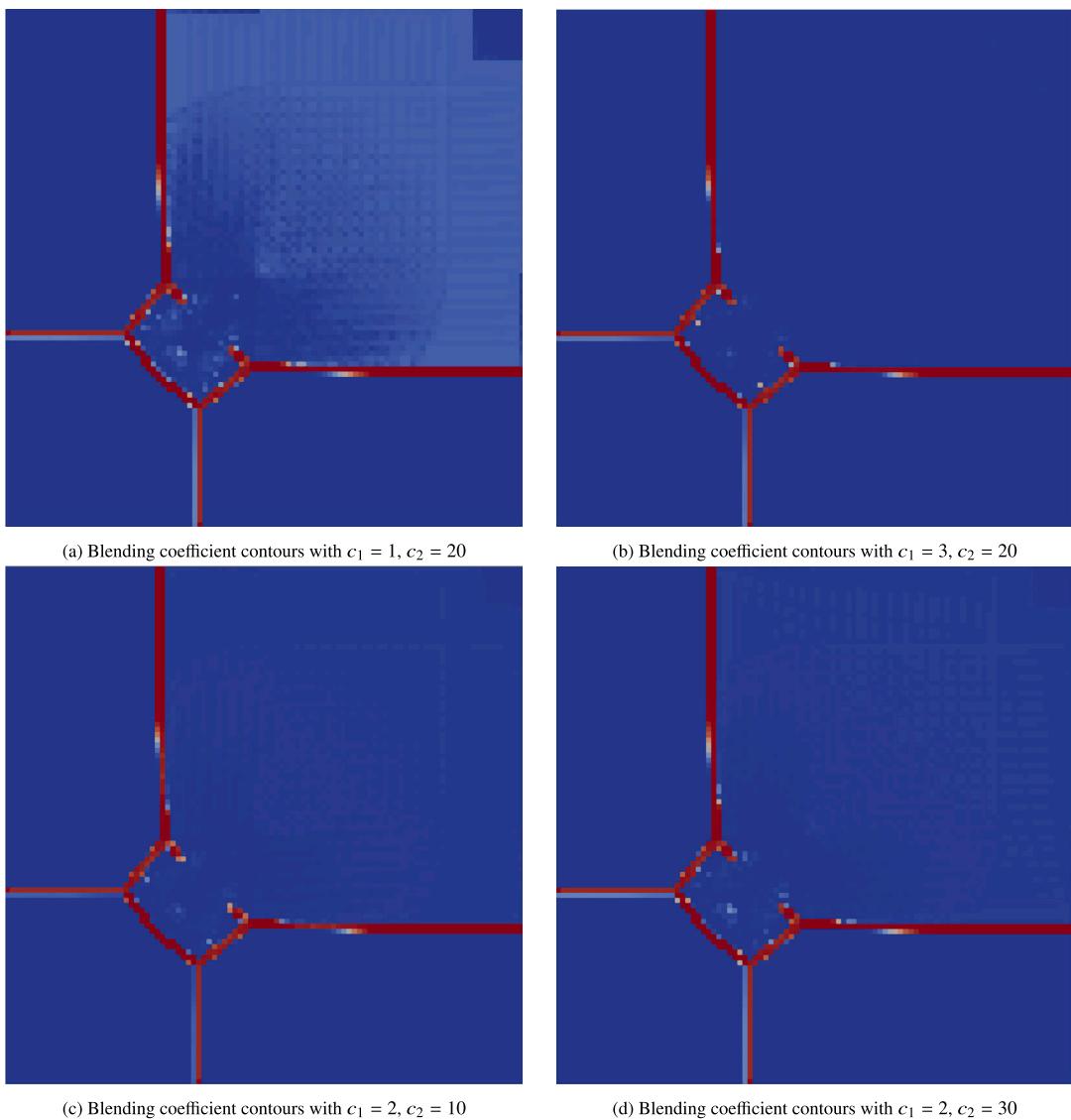


Fig. C.17. Blending coefficient (on a scale of 0 to 1) contours with different c_1 and c_2 combinations for the 2D Riemann problem case 3 with $h = 0.01$ and $N = 4$ at $t = 0.3$.

trends are as predicted by the shape of the activation function in Fig. C.16. To the contrary, keeping c_1 constant and lowering c_2 refines the shock structure, while keeping c_1 constant and increasing c_2 smooths it out, again, as predicted by Fig. C.16. While the parameter choice that leads to an excessive smearing of the shock structure points to an unnecessary dissipation by the scheme, the parameter choice that makes the shock more refined, while works in this test case, may lead to instabilities with higher Mach numbers and stronger shocks. This demonstrates that the chosen values of $c_1 = 2, c_2 = 20$ provide a robust shock capturing scheme while limiting an excessive dissipation.

Data availability

Data will be made available on request.

References

- [1] Wang ZJ, Fidkowski K, Abgrall R, Bassi F, Caraeni D, Cary A, Deconinck H, Hartmann R, Hillewaert K, Huynh H, Kroll N, May G, Persson P-O, van Leer B, Visbal M. High-order CFD methods: Current status and perspective. *Internat J Numer Methods Fluids* 2012;1–42.
- [2] Reed WH, Hill TR. Triangular mesh methods for the neutron transport equation. Technical report, Los Alamos Scientific Lab, New Mexico, USA; 1973.
- [3] Cockburn B, Karniadakis GE, Shu C-W. Discontinuous Galerkin methods: theory, computation and applications, vol. 11. Springer Science & Business Media; 2012.
- [4] Mossier P, Beck A, Munz C-D. A p-adaptive discontinuous Galerkin method with hp-shock capturing. *J Sci Comput* 2022;91(4):1–36.
- [5] Kirby RM, Karniadakis GE. De-aliasing on non-uniform grids: Algorithms and applications. *J Comput Phys* 2003;191:249–64.
- [6] Gassner GJ, Beck AD. On the accuracy of high-order discretizations for underresolved turbulence simulations. *Theor Comput Fluid Dyn* 2013;27:221–37.
- [7] Gassner GJ, Winters AR, Kopriva DA. Split form nodal discontinuous Galerkin schemes with summation-by-parts property for the compressible Euler equations. *J Comput Phys* 2016;327:39–66.

- [8] Canuto C, Hussaini MY, Quarteroni A, Zang TA. Spectral methods in fluid dynamics. Springer-Verlag; 1998.
- [9] Mengaldo G, De Grazia D, Moxey D, Vincent PE, Sherwin SJ. Dealiasing techniques for high-order spectral element methods on regular and irregular grids. *J Comput Phys* 2015;299:56–81.
- [10] Abbassi H, Mashayek F, Jacobs GB. Shock capturing with entropy-based artificial viscosity for staggered grid discontinuous spectral element method. *Comput & Fluids* 2014;98:152–63.
- [11] Pirozzoli S. Generalized conservative approximations of split convective derivative operators. *J Comput Phys* 2010;229:7180–90.
- [12] Mattsson K, Nordström J. Summation by parts operators for finite difference approximations of second derivatives. *J Comput Phys* 2004;199(2):503–40.
- [13] Kopriva DA, Gassner DG. An energy stable discontinuous Galerkin spectral element discretization for variable coefficient advection problems. *SIAM J Sci Comput* 2014;36(4):A2076–99.
- [14] Pritchard CM, Nadarajah S. Fully discrete entropy-stable flux reconstruction scheme for compressible flows through the relaxation Runge-Kutta method. In: AIAA SCITECH 2023 forum. 2023, p. 0663.
- [15] Fisher TC, Carpenter MH. High-order entropy stable finite difference schemes for nonlinear conservation laws: Finite domains. *J Comput Phys* 2013;252:518–57.
- [16] Chan J. On discretely entropy conservative and entropy stable discontinuous Galerkin methods. *J Comput Phys* 2018;362:346–74.
- [17] Chan J, Lin Y, Warburton T. Entropy stable modal discontinuous Galerkin schemes and wall boundary conditions for the compressible Navier-Stokes equations. *J Comput Phys* 2022;448:110723.
- [18] Singh S, Karchani A, Chourashi T, Myong RS. A three-dimensional modal discontinuous Galerkin method for the second-order Boltzmann-Curtiss-based constitutive model of rarefied and microscale gas flows. *J Comput Phys* 2022;457:111052.
- [19] Guermond JL, Pasquetti R, Popov B. Entropy viscosity method for nonlinear conservation laws. *J Comput Phys* 2011;230(11):4248–67.
- [20] Cockburn B, Shu C-W. TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws. II. General framework. *Math Comp* 1989;52(186):411–35.
- [21] Klöckner A, Warburton T, Hesthaven JS. Viscous shock capturing in a time-explicit discontinuous Galerkin method. *Math Model Nat Phenom* 2011;6(3):57–83.
- [22] Baumann CE, Oden JT. A discontinuous hp finite element method for the Euler and Navier-Stokes equations. *Internat J Numer Methods Fluids* 1999;31(1):79–95.
- [23] Cockburn B, Shu CW. The Runge-Kutta discontinuous Galerkin method for conservation laws V: Multidimensional systems. *J Comput Phys* 1998;141:199–224.
- [24] Qiu J, Shu C-W. Hermite WENO schemes and their application as limiters for Runge-Kutta discontinuous Galerkin method: one-dimensional case. *J Comput Phys* 2004;193(1):115–35.
- [25] Sonntag M, Munz C-D. Efficient parallelization of a shock capturing for discontinuous Galerkin methods using finite volume sub-cells. *J Sci Comput* 2017;70(3):1262–89.
- [26] Abgrall R, Shu C-W. Development of residual distribution schemes for the discontinuous Galerkin method: the scalar case with linear elements. *Commun Comput Phys* 2009;5(2–4):376–90.
- [27] Huerta A, Casoni E, Peraire J. A simple shock-capturing technique for high-order discontinuous Galerkin methods. *Internat J Numer Methods Fluids* 2012;69(10):1614–32.
- [28] Persson P-O, Stamm B. A discontinuous Galerkin method for shock capturing using a mixed high-order and sub-grid low-order approximation space. *J Comput Phys* 2022;449:110765.
- [29] Hennemann S, Rueda-Ramírez AM, Hindenlang FJ, Gassner GJ. A provably entropy stable subcell shock capturing approach for high order split form DG for the compressible Euler equations. *J Comput Phys* 2021;426:109935.
- [30] Rueda-Ramírez AM, Hennemann S, Hindenlang FJ, Winters AR, Gassner GJ. An entropy stable nodal discontinuous Galerkin method for the resistive MHD equations. Part II: Subcell finite volume shock capturing. *J Comput Phys* 2021;444:110580.
- [31] Rueda-Ramírez AM, Gassner GJ. A subcell finite volume positivity-preserving limiter for DGSEM discretizations of the Euler equations. In: WCCM ECCOMAS 2020. 2021.
- [32] Pazner W. Sparse invariant domain preserving discontinuous Galerkin methods with subcell convex limiting. *Comput Methods Appl Mech Engrg* 2021;382:113876.
- [33] Zanotti O, Fambri F, Dumbrer M, Hidalgo A. Space-time adaptive ADER discontinuous Galerkin finite element schemes with a posteriori sub-cell finite volume limiting. *Comput & Fluids* 2015;118:204–24.
- [34] Lin Y, Chan J. High order entropy stable discontinuous Galerkin spectral element methods through subcell limiting. *J Comput Phys* 2024;498:112677.
- [35] Rueda-Ramírez AM, Pazner W, Gassner GJ. Subcell limiting strategies for discontinuous Galerkin spectral element methods. *Comput & Fluids* 2022;247:105627.
- [36] Vilar F. A posteriori correction of high-order discontinuous Galerkin scheme through subcell finite volume formulation and flux reconstruction. *J Comput Phys* 2019;387:245–79.
- [37] Bassi F, Rebay S. A high-order accurate discontinuous finite element method for the numerical solution of the compressible Navier-Stokes equations. *J Comput Phys* 1997;131:267–79.
- [38] Gao M, Kuhn T, Munz C-D. On the investigation of oblique shock-wave/turbulent boundary-layer interactions with a high-order discontinuous Galerkin method. *Internat J Numer Methods Fluids* 2022;94(8):1331–57.
- [39] Persson PO, Peraire J. Sub-cell shock capturing for discontinuous Galerkin methods. In: AIAA 2006-112. 2006.
- [40] Bhagatwala A, Lele SK. A modified artificial viscosity approach for compressible turbulence simulations. *J Comput Phys* 2009;228(14):4965–9.
- [41] Ducros F, Ferrand V, Nicoud F, Weber C, Darraq D, Gachereau C, Poinsot T. Large-eddy simulation of the shock/turbulence interaction. *J Comput Phys* 1999;152(2):517–49.
- [42] Johnsen E, Larsson J, Bhagatwala AV, Cabot WH, Moin P, Olson BJ, Rawat PS, Shankar SK, Sjögren B, Yee HC, Zhong X, Lele SK. Assessment of high-resolution methods for numerical simulations of compressible turbulence with shock waves. *J Comput Phys* 2010;229(4):1213–37.
- [43] Pirozzoli S. Numerical methods for high-speed flows. *Annu Rev Fluid Mech* 2011;43:163–94.
- [44] Jameson A, Schmidt W, Turkel E. Numerical solution of the Euler equations by finite volume methods using Runge Kutta time stepping schemes. In: 14th fluid and plasma dynamics conference. 1981, p. 1259.
- [45] Fischer P, Ronquist EM, Dewey D, Patera AT. Spectral element methods: Algorithms and architectures. Technical report, 1988.
- [46] Kopriva DA. Implementing spectral methods for partial differential equations: Algorithms for scientists and engineers. 1st ed.. Springer Publishing Company, Incorporated; 2009.
- [47] Zhang F, Peet YT. Discontinuous Galerkin spectral element method for shock capturing with summation by parts properties. *J Comput Phys*: X 2023;100123.
- [48] Gassner GJ. A skew-symmetric discontinuous Galerkin spectral element discretization and its relation to SBP-SAT finite difference methods. *SIAM J Sci Comput* 2013;35(3):A1233–53.
- [49] Chandrashekhar P. Kinetic energy preserving and entropy stable finite volume schemes for compressible Euler and Navier-Stokes equations. *Commun Comput Phys* 2013;14(5):1252–86.
- [50] Roe PL. Characteristic-based schemes for the Euler equations. *Annu Rev Fluid Mech* 1986;18(1):337–65.
- [51] Gassner GJ, Winters AR, Hindenlang FJ, Kopriva DA. The BR1 scheme is stable for the compressible Navier-Stokes equations. *J Comput Phys* 2018;77:154–200.
- [52] Gottlieb S, Shu CW. Total variation diminishing Runge-Kutta schemes. *Math Comp* 1998;67:73–85.
- [53] Sod G. A survey of several finite difference methods for systems of nonlinear hyperbolic conservation laws. *J Comput Phys* 1978;27:606–14.
- [54] DeVore RA. Nonlinear approximation. *Acta Numer* 1998;7:51–150.
- [55] Schulz-Rinne CW, Collins JP, Glaz HM. Numerical solution of the Riemann problem for two-dimensional gas dynamics. *SIAM J Sci Comput* 1993;14:1394–414.
- [56] Lax PD, Liu XD. Solution of two-dimensional Riemann problems of gas dynamics by positive schemes. *SIAM J Sci Comput* 1998;19(2):319–40.
- [57] Woodward P, Colella P. The numerical simulation of two-dimensional fluid flow with strong shocks. *J Comput Phys* 1984;54:115–73.
- [58] Kemm F. On the proper setup of the double Mach reflection as a test case for the resolution of gas dynamics codes. *Comput & Fluids* 2016;132:72–5.
- [59] Bohm M, Schermeng S, Winters AR, Gassner GJ, Jacobs GB. Multi-element SIAC filter for shock capturing applied to high-order discontinuous Galerkin spectral element methods. *J Sci Comput* 2019;81:820–44.
- [60] Hakkinen R, Greber I, Trilling L, Abarbanel S. The interaction of an oblique shock wave with a laminar boundary layer. (NASA-MEMO-2-18-59W). 1959.
- [61] Katzer E. On the lengthscales of laminar shock/boundary-layer interaction. *J Fluid Mech* 1989;206:477–96.
- [62] Yao Y, Krishnan L, Sandham N, Roberts G. The effect of Mach number on unstable disturbances in shock/boundary-layer interactions. *Phys Fluids* 2007;19(5):054104.
- [63] White FM, Majdalani J. Viscous fluid flow, vol. 3. McGraw-Hill New York; 2006.
- [64] Daru V, Tenaud C. Numerical simulation of the viscous shock tube problem by using a high resolution monotonicity-preserving scheme. *Comput & Fluids* 2009;38(3):664–76.
- [65] Lopez-Morales M, Bull J, Crabil J, Economou TD, Manosalvas D, Romero J, Sheshadri A, Watkins JE, Williams D, Palacios F, et al. Verification and validation of HiFiLES: a high-order LES unstructured solver on multi-GPU platforms. In: 32nd AIAA applied aerodynamics conference, atlanta, Georgia, USA. 2014, p. 16–20.

- [66] DeBonis J. Solutions of the Taylor-Green vortex problem using high-resolution explicit finite difference methods. In: 51st AIAA aerospace sciences meeting including the new horizons forum and aerospace exposition. 2013, p. 382.
- [67] Bailey A, Hiatt J. Sphere drag coefficients for a broad range of Mach and Reynolds numbers. *AIAA J* 1972;10(11):1436–40.
- [68] Uddin H, Kramer RM, Pantano C. A Cartesian-based embedded geometry technique with adaptive high-order finite differences for compressible flow around complex geometries. *J Comput Phys* 2014;262:379–407.
- [69] Al-Marouf M, Samtaney R. A versatile embedded boundary adaptive mesh method for compressible flow in complex geometry. *J Comput Phys* 2017;337:339–78.
- [70] Mao C, Jin T, Luo K, Fan J. Investigation of supersonic turbulent flows over a sphere by fully resolved direct numerical simulation. *Phys Fluids* 2019;31(5):056102.
- [71] Capuano F, Beratlis N, Zhang F, Peet Y, Squires K, Balaras E. Cost vs accuracy: DNS of turbulent flow over a sphere using structured immersed-boundary, unstructured finite-volume, and spectral-element methods. *Eur J Mech B Fluids* 2023;102:91–102.
- [72] Pierce T, Kauffman C, Nicholls J. Mechanism of ignition in shock wave interactions with reactive liquid droplets. In: 13th aerospace sciences meeting. 1975, p. 163.
- [73] Chakraborty P, Balachandar S, Adrian RJ. On the relationships between local vortex identification schemes. *J Fluid Mech* 2005;535:189.
- [74] Fischer P, Lottes J, Kerkemeier S, Marin O, Heisey K, Obabko A, Merzari E, Peet Y. Nek5000 user's manual. Technical report, 2015.
- [75] Toro EF. Riemann solvers and numerical methods for fluid dynamics. 1st ed.. Berlin: Springer Verlag; 1999.
- [76] Carpenter MH, Fisher TC, Nielsen EJ, H. FS. Entropy stable spectral collocation schemes for the Navier-Stokes equations: Discontinuous interfaces. *SIAM J Sci Comput* 2014;36(5):B835–67.