



# Development and analysis of entropy stable no-slip wall boundary conditions for the Eulerian model for viscous and heat conducting compressible flows

Mohammed Sayyari<sup>1</sup> · Lisandro Dalcin<sup>1</sup> · Matteo Parsani<sup>1</sup>

Received: 1 July 2021 / Accepted: 4 October 2021 / Published online: 19 October 2021  
© The Author(s) 2021

## Abstract

Nonlinear entropy stability analysis is used to derive entropy stable no-slip wall boundary conditions for the Eulerian model proposed by Svärd (Phys A Stat Mech Appl 506:350–375, 2018). The spatial discretization is based on entropy stable collocated discontinuous Galerkin operators with the summation-by-parts property for unstructured grids. A set of viscous test cases of increasing complexity are simulated using both the Eulerian and the classic compressible Navier–Stokes models. The numerical results obtained with the two models are compared, and similarities and differences are then highlighted. However, the differences are very small and probably smaller than what the current experimental technology allows to measure.

**Mathematics Subject Classification** 65M70

## 1 Introduction

The classical compressible Navier–Stokes (CNS) equations can be derived based on the material (Lagrangian) derivative formulation [29]. In the Lagrangian sense, diffusion between gas pockets is non-existent, and thus, the continuity equation is hyperbolic. On the other hand, in the Eulerian model of Svärd [32], air molecules diffuse into other parts of the domain, and thus, the continuity equation is modeled as a parabolic equation.

---

This article is part of the section “Computational Approaches” edited by Siddhartha Mishra.

✉ Mohammed Sayyari  
mohammed.alsayyari@kaust.edu.sa

Lisandro Dalcin  
dalcinl@gmail.com

Matteo Parsani  
matteo.parsani@kaust.edu.sa

<sup>1</sup> King Abdullah University of Science and Technology (KAUST), Computer Electrical and Mathematical Science and Engineering Division (CEMSE), Extreme Computing Research Center (ECRC), Thuwal 23955-6900, Saudi Arabia

Generally speaking, entropy conservation and entropy stability are used to preserve the second law of thermodynamics in the mathematical sense, i.e., the mathematical entropy function,  $S$ , decreases monotonically outside the equilibrium. This yields entropy estimates and bounds on  $S$ , which can be translated into bounds on the conservative variables,  $q$ , of the underlying model [12,31]. In this work, nonlinear entropy stability and the summation-by-parts (SBP) framework are used to derive entropy stable wall boundary conditions for the Eulerian model for the viscous and heat-conducting compressible flows proposed by Svärd [32] and their semidiscrete counterpart. As done in [25] for the classical CNS equations, a semidiscrete entropy estimate for the entire domain is achieved when the new boundary conditions are coupled with an entropy stable discrete interior operator. The data at the boundary are weakly imposed using a penalty flux approach and a simultaneous-approximation-term (SAT) penalty technique. At the semidiscrete level, the work in [25] was sharpened in [13] by first constructing entropy conservative wall boundary conditions and then adding a precise interior penalty term. Here, we follow the semidiscrete analysis and derivation presented in [13,25] by using a collocated discontinuous Galerkin framework based on SBP operators constructed at the Legendre–Gauss–Lobatto (LGL) points. We verify the entropy conservation property of the baseline boundary conditions by simulating the flow around a rotating sphere placed in a cubic domain. In addition, to verify the accuracy of the proposed boundary condition implementation, we present the convergence study for a three-dimensional test case constructed using the method of manufactured solutions.

In Svärd [32], Dolejší and Svärd [15], numerical simulations were presented to highlight the difference between the classical CNS equations and the Eulerian model. Here, we also use a set of test cases of increasing complexity simulated using both the CNS and Eulerian models. We use the  $hp$ , fully-discrete entropy stable SSDC solver described, validated, and verified in [26]. The numerical results of the two models in regions near solid walls are compared, and differences and similarities are then highlighted.

The paper is organized as follows. In Sect. 2, we present the Eulerian model in a general form, and we show its entropy stability analysis. Then, in Sect. 3, we derive the wall boundary conditions at the continuous. Later, in Sect. 4, we discretize the system using SBP-SAT operators and present a discrete entropy analysis of the Eulerian model. The latter step sets the context for the construction of discrete entropy conservative and entropy stable wall boundary conditions. Furthermore, in Sect. 5, a common SAT procedure is presented to allow the use of a single subroutine for imposing interface coupling and wall boundary conditions. In Sect. 6, we present a set of numerical results that demonstrate the efficacy and accuracy of the new boundary conditions. In addition, we present a few more test cases that highlight similarities and differences between the two models slightly away from solid walls. Finally, conclusions are drawn in Sect. 7.

## 2 Entropy analysis of the Eulerian model

To analyze the Eulerian model, we begin by presenting its general form and then, using the global entropy, its entropy analysis. The latter step will set the context for deriving solid wall boundary conditions that preserve the nonlinear entropy stability.

### 2.1 General form of the Eulerian model

Svärd [32] arrives at the following form of the Eulerian model for viscous and heat conducting compressible flows:

$$\frac{\partial \rho}{\partial t} + \frac{\partial \rho \mathcal{U}_j}{\partial x_j} = \frac{\partial}{\partial x_j} \left( v \frac{\partial \rho}{\partial x_j} \right), \tag{1a}$$

$$\frac{\partial \rho \mathcal{U}_i}{\partial t} + \frac{\partial \rho \mathcal{U}_i \mathcal{U}_j}{\partial x_j} + \frac{\partial p}{\partial x_i} = \frac{\partial}{\partial x_j} \left( v \frac{\partial \rho \mathcal{U}_i}{\partial x_j} \right), \tag{1b}$$

$$\frac{\partial \rho \mathcal{E}}{\partial t} + \frac{\partial \rho \mathcal{E} \mathcal{U}_j}{\partial x_j} + \frac{\partial p \mathcal{U}_j}{\partial x_j} = \frac{\partial}{\partial x_j} \left( v \frac{\partial \rho \mathcal{E}}{\partial x_j} \right), \tag{1c}$$

where  $i, j = 1 : N_{\text{DIM}}$  (in MATLAB notation), and  $\rho, \mathcal{U}_i, p,$  and  $\mathcal{E}$  are the density, the velocity component in the  $x_i$  direction, the thermodynamic pressure and the specific total energy, respectively.

In what follows, we assume thermodynamically perfect (or ideal) gas. Thus, the thermodynamic pressure,  $p,$  is given by

$$p = \rho R T, \tag{2}$$

where  $R$  is the gas constant. The specific total energy,  $\mathcal{E},$  is given by the following equation

$$\mathcal{E} = c_v T + \frac{1}{2} \mathcal{U}_i \mathcal{U}_i, \tag{3}$$

where  $c_v$  is the heat capacity at constant volume and  $T$  is the temperature. The heat capacity at constant pressure,  $c_p,$  is related to  $c_v$  through the gas constant, i.e.,  $R = c_p - c_v.$  Furthermore, the generalized form of the kinematic viscosity,  $v,$  is given by

$$v = \frac{\alpha \mu}{\rho(\mathbf{x}, t)} + \beta(\rho, T), \tag{4}$$

where  $\mu$  is the dynamic viscosity,  $\alpha \in [1, \frac{4}{3}],$  and  $\beta$  is an additional diffusion coefficient. Finally, the speed of sound,  $c,$  is defined as

$$c = \sqrt{\gamma R T}, \tag{5}$$

where  $\gamma = \frac{c_p}{c_v}.$

**Remark 1** Equation (4) is the most general expression of the kinematic viscosity of the Eulerian model. However, in practice, Svärd [32] chooses  $\beta = 0$  and thus, Eq. (4) reduces to a scaled kinematic viscosity coefficient

$$v = \frac{\alpha \mu}{\rho(\mathbf{x}, t)}. \tag{6}$$

### 2.2 Entropy analysis

We cover the entropy analysis of the Eulerian model (1) by rewriting it using the following compact form,

$$\frac{\partial q}{\partial t} + \frac{\partial f_i^{(I)}}{\partial x_i} = \frac{\partial f_i^{(V)}}{\partial x_i}, \tag{7}$$

where  $f_i^{(I)}$  and  $f_i^{(V)}$  are the inviscid and viscous fluxes in the  $x_i$  direction, respectively.

The next theorem ensures that the Eulerian model is entropy stable.

**Theorem 1** *The following boundary integral*

$$\int_{\Gamma} \left( w^\top \left( v \frac{dq}{dw} \right) \frac{\partial w}{\partial x_i} - \mathcal{F}_i \right) d\Gamma \tag{8}$$

*bounds the time derivative of the entropy function,  $S$ , of the Eulerian model (1).*

**Proof** Given an entropy pair  $(\mathcal{S}, \mathcal{F}_i)$  that symmetrizes the Eulerian model (see [32]), we define the entropy variable  $w^\top = \frac{d\mathcal{S}}{dq}$ . Then, using the matrix  $\frac{dq}{dw}$ , we change the variables of system (1) to  $q := q(w)$ . These steps yield the symmetric form

$$\frac{dq}{dw} \frac{\partial w}{\partial t} + \frac{\partial g_i^{(I)}}{\partial x_i} = \frac{\partial g_i^{(V)}}{\partial x_i}, \tag{9}$$

where  $g_i^{(I)}$ , and  $g_i^{(V)}$  are the symmetrized inviscid and viscous fluxes, respectively. Next, we multiply the symmetric system (9) from the left by the entropy variables. Thus, using the chain rule, the term associated with the time derivative reads

$$w^\top \frac{dq}{dw} \frac{\partial w}{\partial t} = \frac{d\mathcal{S}}{dq} \frac{dq}{dw} \frac{\partial w}{\partial t} = \frac{\partial \mathcal{S}}{\partial t}. \tag{10}$$

Further, as shown by Tadmor [35], the contribution of the symmetrized inviscid flux term yields

$$w^\top \frac{\partial g_i^{(I)}}{\partial x_i} = \frac{\partial \mathcal{F}_i}{\partial x_i}, \tag{11}$$

i.e., the divergence of the entropy flux. Lastly, the viscous term contribution can be manipulated to obtain the following expression:

$$w^\top \frac{\partial g_i^{(V)}}{\partial x_i} = w^\top \frac{\partial}{\partial x_i} \left( \left( v \frac{dq}{dw} \right) \frac{\partial w}{\partial x_i} \right). \tag{12}$$

Therefore, using Eqs. (10)–(12), the scalar partial differential equation for the entropy function,  $\mathcal{S}$ , integrated over a generic domain  $\Omega$  with boundary  $\Gamma$  reads

$$\int_{\Omega} \frac{\partial \mathcal{S}}{\partial t} + \frac{\partial \mathcal{F}_i}{\partial x_i} dx = \int_{\Omega} w^\top \frac{\partial}{\partial x_i} \left( \left( v \frac{dq}{dw} \right) \frac{\partial w}{\partial x_i} \right) dx. \tag{13}$$

Now, using the fundamental theorem of calculus on  $\frac{\partial \mathcal{F}_i}{\partial x_i}$ , and the integration-by-parts rule on  $w^\top \frac{\partial g_i^{(V)}}{\partial x_i}$  gives

$$\int_{\Omega} \frac{\partial \mathcal{S}}{\partial t} = \int_{\Gamma} \left( w^\top \left( v \frac{dq}{dw} \right) \frac{\partial w}{\partial x_i} - \mathcal{F}_i \right) d\Gamma - DT, \tag{14}$$

where

$$DT = \int_{\Omega} \frac{\partial w^\top}{\partial x_i} \left( v \frac{dq}{dw} \right) \frac{\partial w}{\partial x_i} dx. \tag{15}$$

Because the kinematic viscosity,  $\nu$ , is positive and the matrix  $\frac{dq}{dw}$  is symmetric positive-definite, the product  $\left( v \frac{dq}{dw} \right)$  is also a positive-definite matrix. Thus,  $DT$  is positive and appropriately bounded (see [31]). We obtain the following expression

$$\int_{\Omega} \frac{\partial \mathcal{S}}{\partial t} dx \leq \int_{\Gamma} \left( w^\top \left( v \frac{dq}{dw} \right) \frac{\partial w}{\partial x_i} - \mathcal{F}_i \right) d\Gamma \tag{16}$$

which completes the proof.

In the next section, we derive solid wall boundary conditions such that the boundary integral on the left-hand side of (16) is bounded by data.

### 3 Entropy stable wall boundary condition

In this section, we derive the boundary conditions for preserving the entropy stability of the system (1) for a solid wall. The analysis presented next closely follows the works of [13,25].

The entropy–entropy flux pair  $(\mathcal{S}, \mathcal{F}_i) = (-\rho s, -\rho \mathcal{U}_i s)$ , used in [13,25], is the only pair that admits a diffusive entropy flux for the CNS model [32]. Therefore, it is the only suitable pair for a comparison between the CNS and Eulerian systems of equations. This pair is also used in this work. Thus, the entropy variables are computed as [9,17]

$$w = \left( \frac{h}{T} - s - \frac{\mathcal{U}_i \mathcal{U}_i}{2T}, -\frac{\mathcal{U}_1}{T}, -\frac{\mathcal{U}_2}{T}, -\frac{\mathcal{U}_3}{T}, -\frac{1}{T} \right)^\top, \tag{17}$$

where  $h = c_{pr} T$  is the enthalpy.

**Remark 2** The entropy–entropy flux pair for the Eulerian model is not necessarily restricted to  $(\mathcal{S}, \mathcal{F}_i) = (-\rho s, -\rho \mathcal{U}_i s)$ . Svård [32] mentions that the Eulerian model admits a diffusive entropy flux for all Harten’s generalized entropies.

To simplify, we rewrite the Eulerian form (7) as

$$\frac{\partial q}{\partial t} + \frac{\partial f_i^{(l)}}{\partial x_i} = \frac{\partial}{\partial x_i} \left( v \frac{\partial q}{\partial x_i} \right). \tag{18}$$

From the proof of Theorem 1, we arrive at

$$\frac{\partial \mathcal{S}}{\partial t} + \frac{\partial \mathcal{F}_i}{\partial x_i} = w^\top \frac{\partial}{\partial x_i} \left( \left( v \frac{dq}{dw} \right) \frac{\partial w}{\partial x_i} \right). \tag{19}$$

Integrating the previous expression over a cubical domain,  $\Omega = (0, 1) \times (0, 1) \times (0, 1)$ , and, without loss of generality, considering a solid wall located at  $x_i = 0$ , we get the following contributions [25]

$$\int_{\Omega} \frac{\partial \mathcal{S}}{\partial t} dx = - \int_{x_i=0} \left[ w^\top \left( v \frac{dq}{dw} \right) \frac{\partial w}{\partial x_i} - \mathcal{F}_i \right] dx_j dx_k - DT, \tag{20}$$

where  $j, k \neq i$ . From the proof of Theorem 1,  $DT \geq 0$ . Thus, we get the bound

$$\int_{\Omega} \frac{\partial \mathcal{S}}{\partial t} dx \leq - \int_{x_i=0} \left[ w^\top \left( v \frac{dq}{dw} \right) \frac{\partial w}{\partial x_i} - \mathcal{F}_i \right] dx_j dx_k. \tag{21}$$

To bound the time derivative of the entropy, the right-hand-side of (21) requires boundary data. For a solid viscous wall, assuming linear analysis, five independent boundary conditions are required [32]. Four boundary conditions are the three no-slip boundary conditions, as the CNS case, and one on the density flux presented below. The fifth condition is the gradient of the temperature normal to the wall (Neumann boundary condition; e.g., the adiabatic wall), or the temperature of the wall (the Dirichlet or isothermal wall boundary condition), or a mixture of Dirichlet and Neumann conditions (the Robin boundary condition). The last condition is the same as for the CNS model [3,32,33]. These five boundary conditions lead to linear stability of both the CNS and Eulerian model [3,32,33]. In the remainder of this section, we will show the type and the form of the wall boundary conditions that have to be imposed to bound the estimate (21) and, hence, to attain entropy stability.

### 3.1 Inviscid contribution

The inviscid contribution to the time rate of change of the entropy function,

$$\int_{x_i=0} \mathcal{F}_i \, dx_j dx_k, \tag{22}$$

appearing on the RHS of (21), is treated as in [13, Theorem 2.2]. However, for clarity of presentation and completeness, we report it here.

**Theorem 2** *The no-slip boundary conditions  $\mathcal{U}_i = 0$  and  $\mathcal{U}_j = \mathcal{U}_j^{wall}$ , where  $j \neq i$ , bound the inviscid contribution of the time derivative of the entropy function, i.e.*

$$\mathcal{F}_i = 0, \tag{23}$$

for an inviscid solid wall with an outfacing normal vector pointing in the  $x_i$  direction.

**Proof** The proof of this theorem can be found in [13, Theorem 2.2].

### 3.2 Viscous contribution

In this subsection, we derive the viscous boundary conditions for a no-slip wall associated to the first term of the RHS of (21), i.e.,

$$- \int_{x_i=0} w^\top \left( v \frac{dq}{dw} \right) \frac{\partial w}{\partial x_i} \, dx_j dx_k. \tag{24}$$

The main result is provided in the following theorem.

**Theorem 3** *The boundary conditions*

$$g(t) = \mu \frac{1}{T} \frac{\partial T}{\partial x_i}, \quad \text{and} \quad \frac{1}{\rho} \frac{\partial \rho}{\partial x_i} = 0, \tag{25}$$

bound the viscous contribution to the time derivative of the entropy function for a no-slip solid wall with an outfacing normal vector pointing in the  $x_i$  direction.

**Proof** The explicit evaluation of the viscous contribution gives

$$- w^\top \left( v \frac{dq}{dw} \right) \frac{\partial w}{\partial x_i} = \mu \frac{1}{T} \frac{\partial T}{\partial x_i} - \mu(s + (\gamma - 1)) \frac{1}{\rho} \frac{\partial \rho}{\partial x_i} \tag{26}$$

that contributes positively to the time derivative of the entropy function. Then, setting  $\frac{1}{\rho} \frac{\partial \rho}{\partial x_i} = 0$  yields the boundary condition

$$- w^\top \left( v \frac{dq}{dw} \right) \frac{\partial w}{\partial x_i} = \mu \frac{1}{T} \frac{\partial T}{\partial x_i} = g(t) \tag{27}$$

which completes the proof.

For the CNS model, the viscous contribution to the time derivative of the entropy function can be bounded by only setting  $\kappa \frac{1}{T} \frac{\partial T}{\partial x_i}$  [25]. However, for the Eulerian model, we require the imposition of two boundary conditions, i.e.  $g(t) = \mu \frac{1}{T} \frac{\partial T}{\partial x_i}$  and  $\frac{1}{\rho} \frac{\partial \rho}{\partial x_i} = 0$ .

In practice with the above boundary conditions, the time derivative of the entropy function satisfies the following relation

$$\int_{\Omega} \frac{\partial S}{\partial t} dx \leq \mu \frac{1}{T} \frac{\partial T}{\partial x_i} = g(t) = \text{DATA}. \tag{28}$$

Bound (28) can be translated into a bound for the conserved quantities [12,31], and hence, for the primitive variables.

### 4 Semidiscrete entropy stable framework

Herein, using summation-by-parts (SBP) operators [34] and the simultaneous-approximation-technique (SAT) [8,22], we provide an entropy stable framework of any order for the semidiscretization of the Eulerian model (1) using unstructured grids.

#### 4.1 SBP operators

The one-dimensional SBP operator for the first derivative in the direction  $x_i$  is defined as the following.

**Definition 1** Summation-by-parts (SBP) operator for the first derivative: A matrix operator with constant coefficients,  $\mathcal{D} \in \mathbb{R}^{N \times N}$ , is a linear SBP operator of degree  $p$  approximating the derivative  $\frac{\partial}{\partial x_i}$  on the domain  $x_i \in [a, b]$  with nodal distribution  $x$  having  $N$  nodes, if

1.  $\mathcal{D}x_i^j = jx_i^{j-1}$ ,  $j = 0, 1, \dots, p$ ;
2.  $\mathcal{D} = \mathcal{P}^{-1}\mathcal{Q}$ , where the norm matrix  $\mathcal{P}$  is symmetric positive-definite;
3.  $\mathcal{Q} + \mathcal{Q}^T = \mathcal{B}$ , where  $\mathcal{B} = \text{diag}[-1, 0, \dots, 0, 1]$ .

In other words, an SBP operator of degree  $p$  is one that exactly differentiates monomials up to degree  $p$ .

In this work, a collocated discontinuous Galerkin approach is used. Specifically, diagonal norm SBP operators are constructed on the LGL nodes. The one-dimensional SBP operators used in this work are explicitly constructed in [10] with  $N = p + 1$ . Their extension to two- and three-dimensions is achieved using tensor product operations [10,25]:

$$\begin{aligned} \mathcal{D}_{x_1} &= \mathcal{D}_N \otimes \mathcal{I}_N \otimes \mathcal{I}_N \otimes \mathcal{I}_5, & \dots & \quad \mathcal{D}_{x_3} = \mathcal{I}_N \otimes \mathcal{I}_N \otimes \mathcal{D}_N \otimes \mathcal{I}_5, \\ \mathcal{Q}_{x_1} &= \mathcal{Q}_N \otimes \mathcal{I}_N \otimes \mathcal{I}_N \otimes \mathcal{I}_5, & \dots & \quad \mathcal{Q}_{x_3} = \mathcal{I}_N \otimes \mathcal{I}_N \otimes \mathcal{Q}_N \otimes \mathcal{I}_5, \\ \mathcal{B}_{x_1} &= \mathcal{B}_N \otimes \mathcal{I}_N \otimes \mathcal{I}_N \otimes \mathcal{I}_5, & \dots & \quad \mathcal{B}_{x_3} = \mathcal{I}_N \otimes \mathcal{I}_N \otimes \mathcal{B}_N \otimes \mathcal{I}_5, \\ \Delta_{x_1} &= \Delta_N \otimes \mathcal{I}_N \otimes \mathcal{I}_N \otimes \mathcal{I}_5, & \dots & \quad \Delta_{x_3} = \mathcal{I}_N \otimes \mathcal{I}_N \otimes \Delta_N \otimes \mathcal{I}_5, \\ \mathcal{P}_{x_1} &= \mathcal{P}_N \otimes \mathcal{I}_N \otimes \mathcal{I}_N \otimes \mathcal{I}_5, & \dots & \quad \mathcal{P}_{x_3} = \mathcal{I}_N \otimes \mathcal{I}_N \otimes \mathcal{P}_N \otimes \mathcal{I}_5, \\ \mathcal{P}_{x_1,x_2} &= \mathcal{P}_N \otimes \mathcal{P}_N \otimes \mathcal{I}_N \otimes \mathcal{I}_5 & \dots & \quad \mathcal{P}_{x_2,x_3} = \mathcal{I}_N \otimes \mathcal{P}_N \otimes \mathcal{P}_N \otimes \mathcal{I}_5, \\ & & & \quad \mathcal{P}_{x_1,x_2,x_3} = \mathcal{P}_N \otimes \mathcal{P}_N \otimes \mathcal{P}_N \otimes \mathcal{I}_5, \end{aligned} \tag{29}$$

where  $\mathcal{D}_N, \mathcal{Q}_N, \mathcal{B}_N, \Delta_N$  and  $\mathcal{P}_N$  are the one-dimensional SBP operators, and  $\mathcal{I}_N$  is the identity operator.<sup>1</sup> In this context, we choose a diagonal  $\mathcal{P}_N$ . By the definition of SBP operators  $\mathcal{D}_N = \mathcal{P}_N^{-1}\mathcal{Q}_N$  and  $\mathcal{B}_N = \mathcal{Q}_N^T + \mathcal{Q}_N$ . The matrices  $\mathcal{B}_{(\cdot)}$  pick off the interface terms in the

<sup>1</sup> In this work, we use  $N = 5$  because in three dimensions the number of partial differential equations is five for both the CNS and Eulerian model.

respective directions. For the spectral element discretization considered in this paper, the  $\mathcal{B}_{(\cdot)}$  matrices take on a particularly simple form. For example the matrix  $\mathcal{B}_{x_1}$  reads

$$\begin{aligned} \mathcal{B}_{x_1} &= \mathcal{B}_{x_1}^+ - \mathcal{B}_{x_1}^-, \\ \mathcal{B}_{x_1}^- &= \text{diag}(1, 0, \dots, 0) \otimes \mathcal{I}_N \otimes \mathcal{I}_N \otimes \mathcal{I}_5, \\ \mathcal{B}_{x_1}^+ &= \text{diag}(0, \dots, 0, 1) \otimes \mathcal{I}_N \otimes \mathcal{I}_N \otimes \mathcal{I}_5. \end{aligned}$$

For a high-order accurate scheme constructed on tensor product cell, these matrices pick off the values of the vector they act on (typically the solution or the flux vectors) at the nodes of the two opposite faces multiplied by the orthogonal component of the unit normal.

When applying any of the operators (29) to the scalar entropy equation in space, a hat will be used to differentiate the scalar operator from the full vector operator, e.g.

$$\widehat{\mathcal{P}} = (\mathcal{P}_N \otimes \mathcal{P}_N \otimes \mathcal{P}_N).$$

We finally note that in the present work, the quadrature nodes and solution nodes are collocated.

### 4.2 Semidiscretization of the Eulerian model

Using the operators shown in (29), we can write the semidiscretization of (1) as

$$\frac{\partial \mathbf{q}_i}{\partial t} = \mathcal{D}_{x_i} \left( \mathbf{f}_i^{(V)} - \mathbf{f}_i^{(I)} \right) + \mathcal{P}_{x_i}^{-1} \left( \mathbf{g}_{x_i}^{(b)} + \mathbf{g}_{x_i}^{(In)} \right), \tag{30}$$

where vectors  $\mathbf{g}_{x_i}^{(b)}$  enforce the boundary conditions, while  $\mathbf{g}_{x_i}^{(In)}$  patches interfaces together using a SAT approach [25]. Boldface letters represent quantities,  $\mathbf{q}$ , and functions,  $\mathbf{f}$  and  $\mathbf{g}$  for all nodes in an element. Note that the definition  $\mathbf{g}_{x_i}$  involves the use of operators  $\mathcal{B}^-$  or  $\mathcal{B}^+$  (see the following section), which nonzero out only the contribution of the nodes located at the interfaces.

Following [9,17], we use the telescoping property of an SBP operator,

$$\mathcal{D}_{x_i} \mathbf{f}_i^{(I)} = \mathcal{P}_{x_i}^{-1} \mathcal{Q}_{x_i} \mathbf{f}_i^{(I)} = \mathcal{P}_{x_i}^{-1} \Delta_{x_i} \mathcal{I}_{StoF} \mathbf{f}_i^{(I)} = \mathcal{P}_{x_i}^{-1} \Delta_{x_i} \tilde{\mathbf{f}}_i^{(I)}, \tag{31}$$

to re-write (30) as

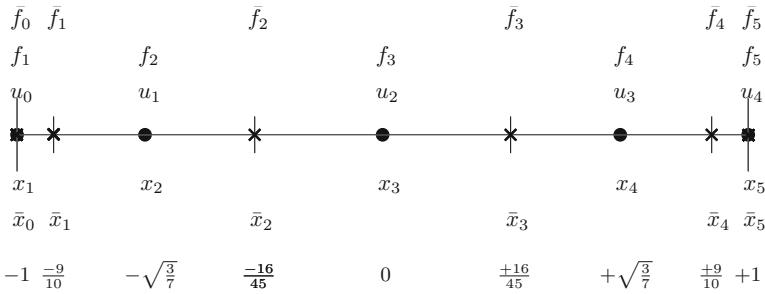
$$\frac{\partial \mathbf{q}}{\partial t} = \left( \mathcal{D}_{x_i} \mathbf{f}_i^{(V)} - \mathcal{P}_{x_i}^{-1} \Delta_{x_i} \tilde{\mathbf{f}}_i^{(I)} \right) + \mathcal{P}_{x_i}^{-1} \left( \mathbf{g}_{x_i}^{(b)} + \mathbf{g}_{x_i}^{(In)} \right). \tag{32}$$

The vector  $\tilde{\mathbf{f}}_i^{(I)}$  is defined as  $\tilde{\mathbf{f}}_i^{(I)} = \mathcal{I}_{StoF} \mathbf{f}_i^{(I)}$  and the one-dimensional telescoping operator,  $\Delta_N$ , is given by

$$\Delta_N = \begin{pmatrix} -1 & 1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 & 0 \\ 0 & 0 & \ddots & \ddots & 0 & 0 \\ 0 & 0 & 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & 0 & -1 & 1 \end{pmatrix}.$$

The operator  $\mathcal{I}_{StoF}$  interpolates the value at the solution nodes to the interfaces between nodes as shown in Fig. 1.





**Fig. 1** One dimensional discretization using LGL points of order  $p = 4$ .  $\cdot$  and  $\times$  denote solution and flux points, respectively. Reprinted from [9] with permission

Additionally, we define  $\left[ \frac{dq}{dw} \right]$  as a block diagonal matrix applied to all LGL points in an element. The kinematic viscosity  $\nu$  is a scalar and we can bring it inside the matrix  $\left[ \frac{dq}{dw} \right]$ , i.e.,  $\left[ \nu \frac{dq}{dw} \right]$ .

Using the procedure based on local discontinuous Galerkin (LDG) and interior penalty (IP) approach described in [13,25], the semidiscretization (32) can be recast as

$$\frac{\partial \mathbf{q}_i}{\partial t} + \mathcal{P}_{x_i}^{-1} \Delta_{x_i} \bar{\mathbf{f}}_{i,sc}^{(I)} - \mathcal{D}_{x_i} \left[ \nu \frac{dq}{dw} \right] \Theta_{x_i} = \mathcal{P}_{x_i}^{-1} \left( \mathbf{g}_{x_i}^{(b),q} + \mathbf{g}_{x_i}^{(In),q} \right), \tag{33a}$$

$$\Theta_{x_i} - \mathcal{D}_{x_i} \mathbf{w} = \mathcal{P}_{x_i}^{-1} \left( \mathbf{g}_{x_i}^{(b),\Theta} + \mathbf{g}_{x_i}^{(In),\Theta} \right), \tag{33b}$$

where the symbol  $\Theta_{x_i}$  is the penalized gradient of the entropy variables in the  $x_i$  direction [25]. The terms  $\mathbf{g}_{x_i}^{(b),q}$ ,  $\mathbf{g}_{x_i}^{(In),q}$ , and  $\mathbf{g}_{x_i}^{(b),\Theta}$ ,  $\mathbf{g}_{x_i}^{(In),\Theta}$  are the SAT penalty boundary ( $b$ ) and interface ( $In$ ) terms on the conservative variables,  $q$ , and the gradient of the entropy variables,  $\Theta$ , respectively. The contributions of the interface penalty terms are non-zero only for the interface nodes in the normal direction to the interface.

**Remark 3** To build a high-order entropy conservative semidiscretization, the linear interpolation operator  $\mathcal{I}_{StoF}$  is replaced with a nonlinear interpolation operator [9,17,25]. Thus, the operator  $\mathcal{P}_{x_i}^{-1} \Delta_{x_i} \bar{\mathbf{f}}_{i,sc}^{(I)}$  is a nonlinear operator that is a discrete counterpart to the term that appears in Theorem 2. Thus, we arrive at the following relation

$$\mathbf{1}^\top \mathcal{P} \mathcal{P}_{x_i}^{-1} \Delta_{x_i} \bar{\mathbf{f}}_{i,sc}^{(I)} = \mathbf{1}^\top \widehat{\mathcal{P}}_{x_j, x_k} \widehat{\mathcal{B}}_{x_i} \mathbf{F}_{x_i} \simeq \int_{x_i=1} \mathcal{F}_i dx_j dx_k - \int_{x_i=0} \mathcal{F}_i dx_j dx_k. \tag{34}$$

The flux vector  $\bar{\mathbf{f}}_{i,sc}^{(I)}$  is constructed using an entropy conservative two-point flux [35], and therefore, it satisfies the relation (11). In the case of an entropy stable flux,  $\bar{\mathbf{f}}_{i,ss}^{(I)}$ , the resulting term is entropy dissipative [13].

### 4.3 Time derivative of entropy function

Following the entropy analysis detailed in [9,24,25], the semidiscretization (33) yields the following expression for the time derivative of the entropy function,  $\mathcal{S}$ :

$$\frac{d}{dt} \mathbf{1}^\top \widehat{\mathcal{P}} \mathcal{S} + \mathbf{D} \mathbf{T} = \Xi \tag{35}$$

where

$$\mathbf{DT} = \left\| \left[ v \frac{dq}{dw} \right]^{\frac{1}{2}} \boldsymbol{\Theta}_{x_i} \right\|_{\mathcal{P}}^2, \tag{36}$$

and

$$\begin{aligned} \boldsymbol{\Xi} &= -\mathbf{1}^\top \widehat{\mathcal{P}}_{x_j, x_k} \widehat{\mathcal{B}}_{x_i} \mathbf{F}_{x_i} \\ &+ \mathbf{w}^\top \mathcal{P}_{x_j, x_k} \mathcal{B}_{x_i} \left[ v \frac{dq}{dw} \right] \boldsymbol{\Theta}_{x_i} \\ &+ \mathbf{w}^\top \mathcal{P}_{x_j, x_k} \left( \mathbf{g}_{x_i}^{(b),q} + \mathbf{g}_{x_i}^{(In),q} \right) \\ &+ \left( \left[ v \frac{dq}{dw} \right] \boldsymbol{\Theta}_{x_i} \right)^\top \mathcal{P}_{x_j, x_k} \left( \mathbf{g}_{x_i}^{(b),\Theta} + \mathbf{g}_{x_i}^{(In),\Theta} \right). \end{aligned} \tag{37}$$

The full derivation of (35) is detailed in Appendix A.

Now, consider a cubic element of length 1 with a solid wall with a normal vector in the  $x_i$  direction, we have that the interface penalty terms are zero, i.e.,  $\mathbf{g}_{x_i}^{(In),q} = \mathbf{g}_{x_i}^{(In),\Theta} = 0$ . Therefore, Eq. (37) reduces to

$$\begin{aligned} \boldsymbol{\Xi} &= -\mathbf{1}^\top \widehat{\mathcal{P}}_{x_j, x_k} \widehat{\mathcal{B}}_{x_i} \mathbf{F}_{x_i} \\ &+ \mathbf{w}^\top \mathcal{P}_{x_j, x_k} \mathcal{B}_{x_i} \left[ v \frac{dq}{dw} \right] \boldsymbol{\Theta}_{x_i} \\ &+ \mathbf{w}^\top \mathcal{P}_{x_j, x_k} \mathbf{g}_{x_i}^{(b),q} \\ &+ \left( \left[ v \frac{dq}{dw} \right] \boldsymbol{\Theta}_{x_i} \right)^\top \mathcal{P}_{x_j, x_k} \mathbf{g}_{x_i}^{(b),\Theta}. \end{aligned} \tag{38}$$

#### 4.4 Entropy stable wall boundary conditions for the semidiscrete system

The boundary condition penalty term with respect to the conservative variables is split into three design-order terms plus one source boundary term:

$$\mathbf{g}_{x_i}^{(b),q} = \mathbf{g}_{x_i}^{(b,I),q} + \mathbf{g}_{x_i}^{(b,V),q} + \mathcal{M}^{(b,V)} + \mathcal{L}^{(b,V)}. \tag{39}$$

The first component of each term is computed from the numerical solution, and the second component is constructed from a combination of the numerical solution and five independent boundary data, as done in [25]. Without loss of generality, in this section, we assume a wall at  $x_i = 0$ ; thus, we use the operator  $\mathcal{B}^-$  to single out the contribution of the cell nodes to the wall nodes.

In the following, we define the inviscid,  $\mathbf{g}_{x_i}^{(b,I),q}$ , and viscous,  $\mathbf{g}_{x_i}^{(b,V),q}$  and  $\mathbf{g}_{x_i}^{(b,V),\Theta}$ , contributions in cell notation. Then, we use the point-wise notation to define the remaining terms,  $\mathcal{M}^{(b,V)}$  and  $\mathcal{L}^{(b,V)}$ , which are only applied to the boundary interface nodes using the operators  $\mathcal{B}^-$ , or  $\mathcal{B}^+$ , i.e.,

$$\mathcal{M}^{(b,V)} = -\mathcal{B}^- \mathcal{M}^{(b,V)}, \tag{40}$$

and

$$\mathcal{L}^{(b,V)} = -\mathcal{B}^- \mathcal{L}^{(b,V)}. \tag{41}$$

The first term enforces Euler no-penetration wall

$$\mathbf{g}_{x_i}^{(b,I),q} = -\frac{1}{2} \mathcal{B}_{x_i}^- \left( \mathbf{f}_i^{(I)} - \mathbf{f}_{i,sc}^{(b,I)}(v^{(E)}, v^{(b,I)}) \right), \tag{42}$$

where  $\mathbf{f}_{i,sc}^{(b,I)}$  is the entropy conservative flux and  $v$  represent the primitive variables. The manufactured inviscid states are defined as in [25] for  $i = 1$ , without loss of generality,

$$v^{(E)} = \text{diag}([1, -1, 1, 1, 1]), \quad \text{and} \quad v^{(b,I)} = (\rho, -\mathcal{U}_1, \mathcal{U}_2, \mathcal{U}_3, \mathcal{T})^\top. \tag{43}$$

The viscous boundary term is given as

$$\mathbf{g}_{x_i}^{(b,V),q} = \frac{1}{2} \mathcal{B}_{x_i}^- \left( \left[ v \frac{dq}{dw} \right] \Theta_{x_i} - \mathbf{f}_i^{(b,V)} \right), \tag{44}$$

where  $\mathbf{f}_i^{(b,V)}$  will be defined later in this section. Together with (44), the following term

$$\mathbf{g}_{x_i}^{(b,V),\Theta} = \frac{1}{2} \mathcal{B}_{x_i}^- \left( \mathbf{w} - \mathbf{w}^{(b,V)} \right), \tag{45}$$

enforce the no-slip boundary condition weakly. The analysis of the previous viscous terms match the analysis described by [13,25] since the entropy variables are the same, and the matrix  $\left[ v \frac{dq}{dw} \right]$  is symmetric positive-definite.

The remaining terms are defined in point-wise notation, and thus the bold notation is dropped. The analysis of their contributions is summarized next.

- The manufactured viscous flux is defined as

$$f_i^{(b,V)} = \left( v \frac{dq}{dw} \left( v^{(b,V)} \right) \right) \tilde{\Theta}_{x_i}, \tag{46}$$

where the manufactured viscous boundary primitive variables,  $v^{(b,V)}$ , is defined, as in [13], by

$$v^{(b,V)} = (\rho, -\mathcal{U}_1 + 2\mathcal{U}_1^{\text{wall}}, -\mathcal{U}_2 + 2\mathcal{U}_2^{\text{wall}}, -\mathcal{U}_3 + 2\mathcal{U}_3^{\text{wall}}, \mathcal{T})^\top. \tag{47}$$

The change of variables matrix,  $\frac{dq}{dw}(v)$ , as a function of the primitive variables,  $v$ , is defined in Appendix B. The term  $\Theta_{x_i}$  is the manufactured gradient of the entropy variables. The construction of  $\tilde{\Theta}_{x_i}$  is summarized in Appendix C.

- The dissipative IP term is an averaged state of the viscous flux evaluated at the primitive state,  $v$ , and the manufactured primitive state,  $v^{(b,V)}$ ,

$$\mathcal{M}^{(b,V)} = -\beta \frac{f^{(V)}(v) + f^{(V)}(v^{(b,V)})}{2}, \tag{48}$$

where  $\beta$  is a positive constant that is scaled with the inverse of the element length in the normal direction, controlling the strength of the penalty term [13].

- The source term

$$\mathcal{L}^{(b,V)} = -(0, 0, 0, 0, 1)^\top g(t), \tag{49}$$

imposes the heat flux boundary condition, as done in [25].

We summarize the results for the RHS of (35) in the following three theorems. The first theorem is a result from [25] and is reprinted here for completeness

**Theorem 4** *The penalty inviscid flux contribution,  $\mathbf{g}^{(b,I),q}$ , is entropy conservative if the vector  $v^{(b,I)}$  is defined as in (43).*

**Proof** The proof of this theorem can be found in [25, Theorem 5.1].

**Theorem 5** *The penalty terms for the viscous flux in the conserved variables,  $\mathbf{g}^{(b,V),q}$ , and the gradient of entropy variables,  $\mathbf{g}^{(b,V),\Theta}$ , are*

- entropy conservative if the wall is adiabatic, i.e.  $g(t) = 0$ ,
- entropy stable in the presence of a heatflux,  $g(t) \neq 0$ , where  $g(t)$  is a given  $L^2$  function.

**Proof** Substituting the expressions for  $\mathbf{g}^{(b,V),q}$ , and,  $\mathbf{g}^{(b,V),\Theta}$  into (38), yields

$$\frac{d}{dt} \mathbf{1}^\top \widehat{\mathcal{P}} \mathbf{S} + \mathbf{D}\mathbf{T} = \mathbf{1}^\top \widehat{\mathcal{P}}_{x_j, x_k} g(t). \tag{50}$$

The function  $g(t)$  for an adiabatic wall is  $g(t) = 0$ . Thus the RHS is zero, and the scheme is entropy conservative. Otherwise, the term  $g(t)$  is bounded by data. Hence, the RHS is bounded, completing the proof.

The full details on the computation of (50) are reported in Appendix C.

**Theorem 6** *The IP term,  $\mathcal{M}^{(b,V)}$ , added to (39) is entropy dissipative.*

**Proof** By expanding the penalty terms with respect to the conservative variables,  $\mathbf{g}^{(b,V),q}$ , and focusing only on the dissipative IP term,  $\mathcal{M}^{(b,V)}$ , in (35), we arrive at

$$w^\top \mathcal{M}^{(b,V)} = -\frac{2\beta\alpha\mu}{RT^2} \|\Delta\mathcal{U}\|^2 (\|\Delta\mathcal{U}\|^2 + RT). \tag{51}$$

Thus,  $\mathcal{M}^{(b,V)}$  is entropy dissipative because all the parameters and variables appearing on the RHS of (51) are positive. This completes the proof.

The full details on the computation of (51) are reported in Appendix D.

### 5 A common SAT procedure for the imposition of wall boundary conditions and interior interface coupling

The proposed approach for imposing the solid wall boundary conditions allow for a SAT implementation that is identical to the interface treatment shown in [24]. We can use a single subroutine with different inputs corresponding to the imposition of the interior interface couplings, or the adiabatic solid wall, or the wall with a prescribed heat entropy flow. In fact, the interior interface coupling can be written as (see equations (16a–16d) in [24])

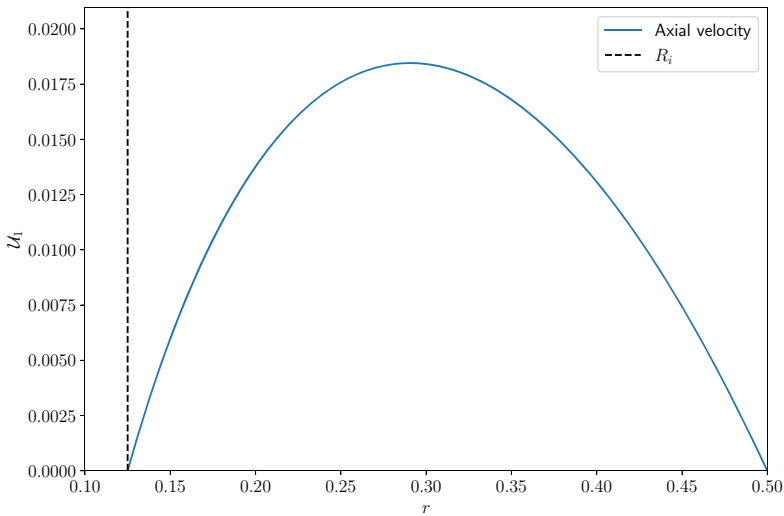
$$\frac{\partial \mathbf{q}_l}{\partial t} + \left( \mathcal{P}_{x_{i,l}}^{-1} \Delta_{x_{i,l}} \bar{\mathbf{f}}_{i,l,sc}^{(I)} - \mathcal{D}_{x_{i,l}} \left[ v \frac{dq}{dw} \right] \Theta_{x_{i,l}} \right) = \mathcal{P}_{x_{i,l}}^{-1} \mathbf{g}_{x_{i,l}}^{(In),q}, \tag{52a}$$

$$\Theta_{x_{i,l}} - \mathcal{D}_{x_{i,l}} \mathbf{w} = \mathcal{P}_{x_{i,l}}^{-1} \mathbf{g}_{x_{i,l}}^{(In),\Theta}, \tag{52b}$$

$$\frac{\partial \mathbf{q}_r}{\partial t} + \left( \mathcal{P}_{x_{i,r}}^{-1} \Delta_{x_{i,r}} \bar{\mathbf{f}}_{i,r,sc}^{(I)} - \mathcal{D}_{x_{i,r}} \left[ v \frac{dq}{dw} \right] \Theta_{x_{i,r}} \right) = \mathcal{P}_{x_{i,r}}^{-1} \mathbf{g}_{x_{i,r}}^{(In),q}, \tag{52c}$$

$$\Theta_{x_{i,r}} - \mathcal{D}_{x_{i,r}} \mathbf{w} = \mathcal{P}_{x_{i,r}}^{-1} \mathbf{g}_{x_{i,r}}^{(In),\Theta}. \tag{52d}$$

The above equations have exactly the same structure as the LDG-IP approach used for the imposition of the solid wall boundary conditions except for the boundary penalty interface terms,  $\mathbf{g}_{x_{i,r}}^{(b),\cdot}$  in Eq. (33), which are replaced by the interior penalty interface coupling terms,  $\mathbf{g}_{x_{i,r}}^{(In),\cdot}$  in Eqs. (52).



**Fig. 2** Axial velocity,  $U_1$ , in a pipe with annular section; Eulerian model with  $Re = 1$ ,  $Ma = 1e-3$

## 6 Numerical results

In this section, we numerically investigate the proposed entropy stable wall boundary conditions. The numerical experiments reported in this paper are performed with the entropy stable collocated Discontinuous Galerkin algorithm and relaxation Runge–Kutta schemes implemented in the *hp*-adaptive, unstructured, curvilinear grid framework SSDC [26]. SSDC is developed in the AANSLab, which is part of the Extreme Computing Research Center at King Abdullah University of Science and Technology (KAUST). The core entropy stable adaptive algorithms of SSDC are built on top of the Portable and Extensible Toolkit for Scientific computing (PETSc) [2], its mesh topology abstraction (DMPLEX) [20], and its scalable ODE/DAE solver library [1]. The SSDC framework uses a non-dimensional formalism; thus, all quantities are scaled to units. For the Eulerian model, the kinematic viscosity is scaled using  $\alpha = 1$ . We use the two-point entropy conservative flux presented by [11]. Furthermore, all the simulations have been performed in double (machine) precision. For all the cases, we use the Runge–Kutta scheme of [5] with adaptive time-step and both relative and absolute tolerances set to  $10^{-8}$ . The meshes are generated using Gmsh [19], and Pointwise V18.3 released in September, 2019. The SSDC Eulerian and SSDC CNS data computed for this section is available in <http://doi.org/10.5281/zenodo.5041436>.

### 6.1 Convergence study

In this section, we use the method of manufactured solutions (MMS) to verify the accuracy and correct implementation of the proposed boundary conditions. We use a pipe with an annular section and length 1. Similarly to [13], this case is considered for two reasons. First, it has an analytic solution for incompressible flow that cannot be represented in polynomial space [27]. Because we use a compressible formulation instead, we calculate the appropriate source term, giving us the manufactured solution. Second, it allows exercising the high-order mesh capabilities and better approximate the circular geometry of the pipe. The following

**Table 1** Convergence study for the flow in a pipe with annular cross-section at  $Re = 1$  and  $Ma = 1e-3$

Grid	$L^1$	Rate	$L^2$	Rate	$L^\infty$	Rate
4	8.36276e-04	–	1.77874e-03	–	5.70764e-03	–
8	1.25039e-04	-2.742	3.44481e-04	-2.368	1.49599e-03	-1.932
16	1.99325e-05	-2.649	5.46416e-05	-2.656	3.12259e-04	-2.260
32	2.64237e-06	-2.915	7.50755e-06	-2.864	5.07517e-05	-2.621
64	3.31631e-07	-2.994	9.65089e-07	-2.960	7.22735e-06	-2.812
128	4.12357e-08	-3.008	1.21565e-07	-2.989	9.60653e-07	-2.911

Axial velocity error; solution polynomial degree:  $p = 2$

**Table 2** Convergence study for the flow in a pipe with annular cross-section at  $Re = 1$  and  $Ma = 1e-3$

Grid	$L^1$	Rate	$L^2$	Rate	$L^\infty$	Rate
4	2.00203e-04	–	3.85016e-04	–	1.54031e-03	–
8	2.71540e-05	-2.882	5.67563e-05	-2.762	3.16887e-04	-2.281
16	2.15610e-06	-3.655	5.69093e-06	-3.318	4.42491e-05	-2.840
32	1.24768e-07	-4.111	4.40714e-07	-3.691	4.63001e-06	-3.257
64	6.57969e-09	-4.245	2.86670e-08	-3.942	4.05012e-07	-3.515
128	2.95557e-10	-4.477	1.63672e-09	-4.131	3.12241e-08	-3.697

Axial velocity error; solution polynomial degree:  $p = 3$

**Table 3** Convergence study for the flow in a pipe with annular cross-section at  $Re = 1$  and  $Ma = 1e-3$

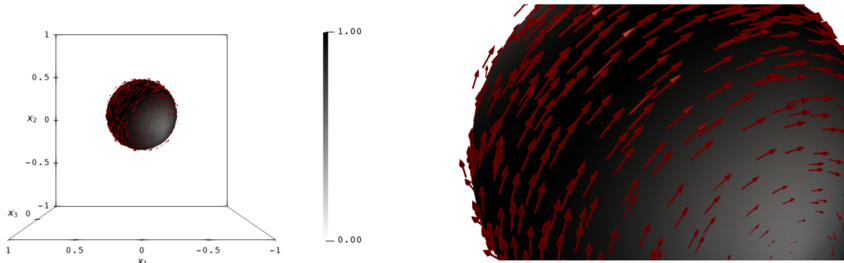
Grid	$L^1$	Rate	$L^2$	Rate	$L^\infty$	Rate
4	4.54267e-05	–	9.55211e-05	–	4.70479e-04	–
8	4.47553e-06	-3.343	9.58936e-06	-3.316	6.31346e-05	-2.898
16	1.88611e-07	-4.569	5.68142e-07	-4.077	5.14343e-06	-3.618
32	5.97100e-09	-4.981	2.34227e-08	-4.600	2.75719e-07	-4.221
64	1.85738e-10	-5.007	7.90368e-10	-4.889	1.13276e-08	-4.605
128	5.76894e-12	-5.009	2.49773e-11	-4.984	3.99322e-10	-4.826

Axial velocity error; solution polynomial degree:  $p = 4$

solution is used for the axial velocity:

$$u_1 = \frac{G}{4\mu} \left( (R_1^2 - r^2) + (R_2^2 - R_1^2) \frac{\log(r/R_1)}{\log(R_2/R_1)} \right), \tag{53}$$

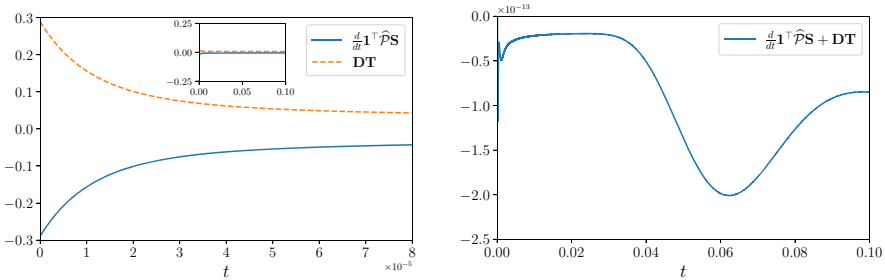
where  $R_o = 0.5$ ,  $R_o/R_i = 4$ , and  $G/\mu = 1$  (Fig. 2). We consider uniform density and temperature, and zero nonaxial velocities. The parameters used are  $Re = 1$ ,  $Ma = 1e-3$  and  $\alpha = 1$ . No-slip adiabatic wall boundary conditions are used on the outer and inner cylinder whereas, periodic boundary conditions are used on the remaining boundaries. We use Mathematica to compute the source term of the Eulerian model [36].



(a) Overview of the spinning sphere.

(b) A closer look at the velocity field near the spinning sphere.

**Fig. 3** Spinning sphere enclosed in a cubical box; Eulerian model with  $Re = 1$ , and  $Ma = 0.05$ . The magnitude of the point-wise velocity is used to scale the arrow glyphs vectors and to color the surface of the sphere



(a) Evolution of the time derivative of the entropy function and the dissipation term.

(b) Evolution of the sum of the time derivative of the entropy function and the dissipation term.

**Fig. 4** Verification of the entropy conservation for the spinning sphere enclosed in a cubical box; Eulerian model with  $Re = 1$ , and  $Ma = 0.05$

The error calculation uses the following discrete norms

$$\begin{aligned}
 \text{Discrete } L^1: \|\mathbf{u}\|_{L^1} &= \|\Omega\|^{-1} \sum_{k=1}^K \mathbf{1}_{N_k}^\top \mathcal{P}_k \mathcal{J}_k \text{abs}(\mathbf{u}_k), \\
 \text{Discrete } L^2: \|\mathbf{u}\|_{L^2}^2 &= \|\Omega\|^{-1} \sum_{k=1}^K \mathbf{u}_k^\top \mathcal{P}_k \mathcal{J}_k \mathbf{u}_k, \\
 \text{Discrete } L^\infty: \|\mathbf{u}\|_{L^\infty} &= \max_{k=1 \dots K} \text{abs}(\mathbf{u}_k),
 \end{aligned}
 \tag{54}$$

where  $\|\Omega\|$  represents the volume of the computational domain,  $\mathcal{J}_k$  is the metric Jacobian of the curvilinear transformation from physical space to computational space of the  $k$ -th hexahedral element, and  $K$  is the total number of non-overlapping hexahedral elements in the mesh.

The results of the convergence study are shown in Tables 1, 2 and 3, where the first column represents the number of elements in the radial and angular coordinates. We observe that the computed order of accuracy is approximately  $(p + 1)$ .

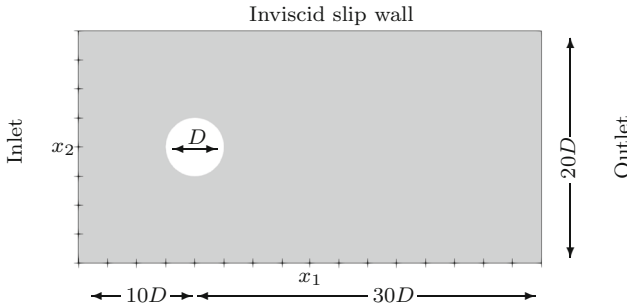


Fig. 5 Cylinder domain with  $D = 1$

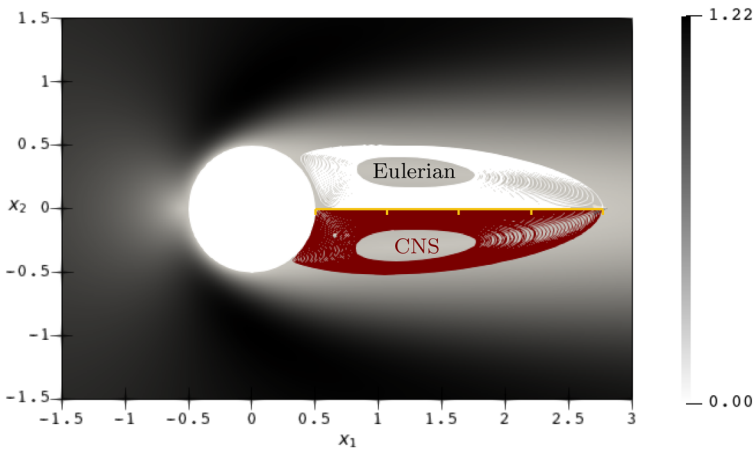


Fig. 6 Wake region of the flow past a cylinder at  $Re = 40$  and  $Ma = 0.07$ . The background is colored by the magnitude of the point-wise velocity vector solution of the Eulerian model. The wake streamlines for the CNS model are imported from a CNS solution for the same testcase

### 6.2 Spinning sphere: verification of the semidiscrete entropy balance

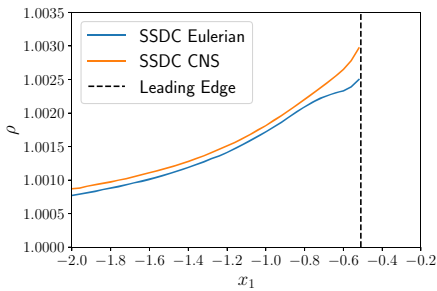
To verify that the scheme enforces no entropy flow through the boundary, we simulate a spinning sphere enclosed in a cubic domain. The sphere rotates at a constant angular velocity around a unit vector given by  $\hat{a}_r = \frac{a_r}{\|a_r\|}$ , where  $a_r = (1, 1, 1)^\top$ . The domain size is  $2 \times 2 \times 2$  with a sphere of diameter  $D = 0.6$  located at the center of it. Solid wall boundary conditions are imposed on the sphere surface and all six faces of the cubic box. The mesh is composed of 4,374 hexahedral elements. The sphere surface is discretized with quadratic boundary elements. A solution polynomial degree  $p = 5$  is used. We run the Eulerian model with  $Re = 1$ , and  $Ma = 0.05$ . In Fig. 3, the velocity vector field near the surface of the sphere is shown. The sphere is also colored based on the module of its point-wise velocity vector.

Figure 4a shows the time derivative of the entropy function of the semidiscretization of the Eulerian model (30),  $\frac{d}{dt} \mathbf{1}^\top \hat{\mathcal{P}} \mathcal{S}$ , and the dissipation term,  $\mathbf{DT}$ , as a function of time,  $t$  (see Eq. (35)). As shown in Fig. 4b, the two terms cancel out, up to machine precision. Therefore, the procedure proposed to impose the boundary conditions is entropy conservative if the IP term in (39) is set to zero. This simulation is a strong numerical verification of what is proven in Sect. 4.4.

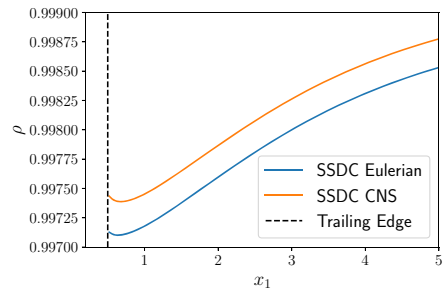


**Table 4** Bubble length,  $L_{\text{bubble}}$ , and drag coefficient,  $C_D$

	$L_{\text{bubble}}$	$C_D$
Park et al. [23]	2.25	1.51
Ye et al. [37]	2.27	1.52
Fornberg [18]	2.24	2.5
Dennis and Chang [14]	2.35	1.52
Calhoun [7]	2.18	1.62
Russell and Wang [28]	2.29	1.6
SSDC CNS	2.254	1.608
SSDC Eulerian	2.256	1.609

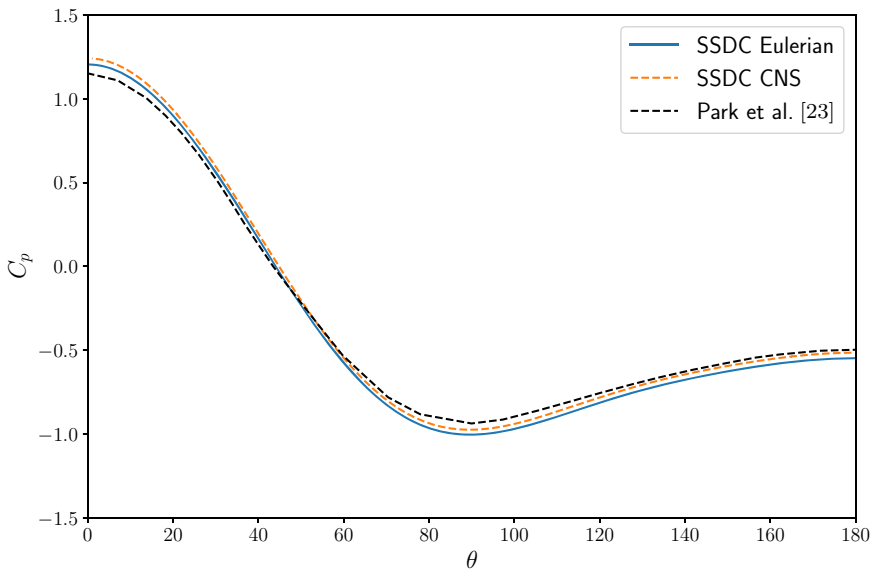


(a) Leading edge.

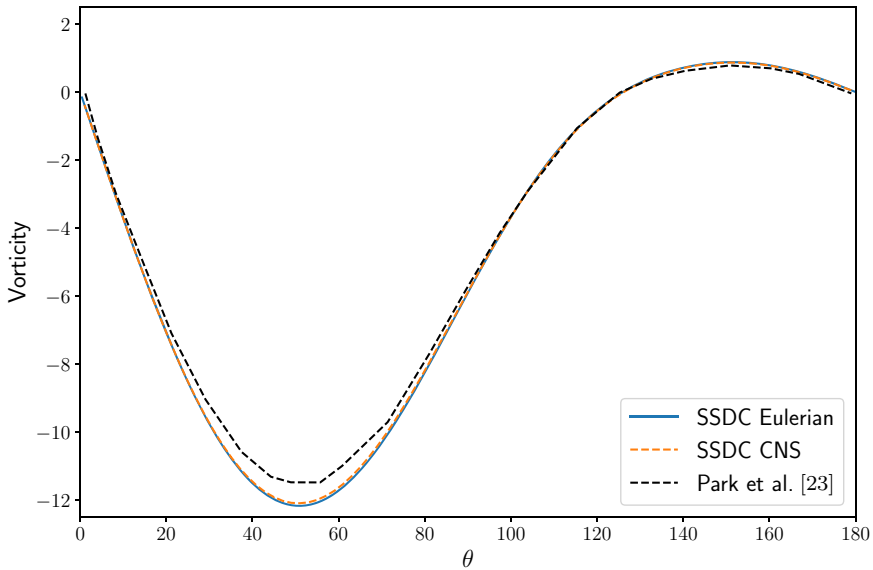


(b) Trailing edge.

**Fig. 7** The density profile for the flow past a cylinder at  $Re = 40$  and  $Ma = 0.07$ . The figures are scaled so that differences between the Eulerian and CNS solution densities are visible



**Fig. 8** Wall pressure coefficient for the flow past a cylinder at  $Re = 40$  and  $Ma = 0.07$ . The reference data are obtained from [23]



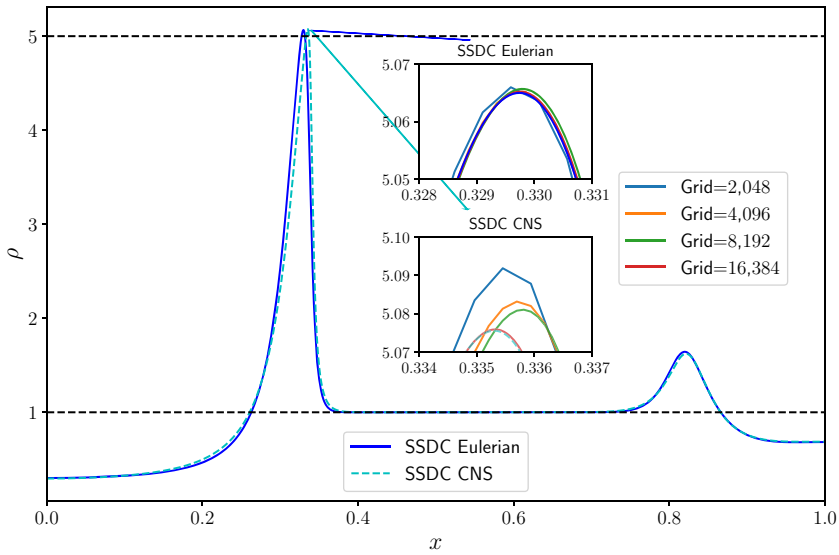
**Fig. 9** Vorticity at the wall for the flow past a cylinder at  $Re = 40$  and  $Ma = 0.07$ . The reference data was obtained from [23]

### 6.3 Laminar flow around a cylinder $Re = 40$

In this section, we present the two-dimensional flow around a cylinder of diameter  $D = 1$  at  $Re = 40$ , and  $Ma = 0.07$ . Figure 5 shows the computational domain and summarizes the boundary conditions. The initial condition is a uniform flow in the  $x_1$  direction. The case is run until convergence to a steady-state. Specifically, we stop the simulation when the residual reaches  $10^{-14}$  and plateau around that value. Entropy stable adiabatic no-slip wall boundary conditions are enforced on the cylinder surface. Far-field boundary conditions are used for the remaining boundaries. The mesh is composed of 1140 elements with quadratic edges for representing the surface of the cylinder. We use a solution polynomial degree  $p = 7$ . Therefore, the total number of degrees of freedom is 72,960. The solution computed with this setup is denoted here as a “numerically converged solution” in the sense that if we increase the order of accuracy of the method,  $p$ , or refine the mesh, the difference between two consecutive numerical solutions for all the primitive variables is machine precision.

First, we compute the length of the recirculation bubble in the wake region. Figure 6 shows the streamlines of the circulation bubble downstream of the sphere. The background is the magnitude of the point-wise velocity vector. The top part of the plot corresponds to the solution computed with the Eulerian model, whereas the bottom part is the solution obtained using the CNS model. Qualitatively, the two solutions look identical. Furthermore, both models show a wake length of  $2.25D$ . The first column in Table 4 compares the length of the bubble with several results published in the literature. The agreement is good. The second column of Table 4 compares the drag coefficient,  $C_D$ . The values obtained with both models are very close to each other and agree well with that reported by Calhoun [7] and Russell and Wang [28].

Next, we present some measurements in the boundary layer of the cylinder wall. Figure 7 shows the density profile for both models, where for the Eulerian model, we enforce a zero



**Fig. 10** Density distribution for the blastwave at time  $t = 0.01$  computed with the Eulerian and the CNS models. The base figure uses a grid of 32,768 elements

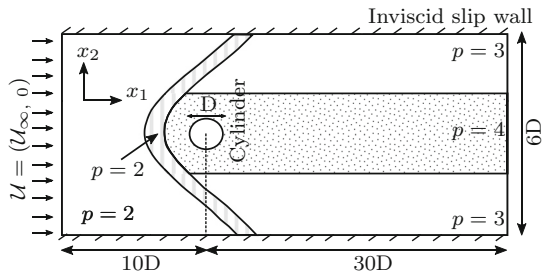
normal density gradient at the wall. Furthermore, for both models, we use adiabatic solid wall boundary conditions. In Fig. 7a, the difference between the two solution densities is shown, and is smaller than  $1e-3$ . The difference observed is accumulated at the leading edge and remains consistent along with the tailwind, as shown in Fig. 7. Note that we have different density ranges for each figure. This was done since a unified range does not make differences visible. We highlight that it is remarkable how close the results of the two models are, given their very different structures of the diffusive terms and boundary conditions.

Figure 8 shows the wall pressure coefficient,  $C_p$ . The curves obtained with the Eulerian model and the CNS model are very close to each other and follow relatively well the DNS results published by Park et al. [23]. Note that in Park et al. [23], the scheme uses “the second-order central-difference scheme in space.” A similar comparison is made in Fig. 9 for the vorticity at the wall. Again, the curves obtained with the Eulerian model and the CNS model are very close to each other and follow well the DNS results except around  $50^\circ$  where they both undershoot the reference results.

### 6.4 Blast-wave

Svärd [32] presents the one-dimensional blastwave as an example where the implementation of the CNS model converges to an unphysical solution. Here, we investigate the same test case. We use  $Re = 10$  and  $Ma = 0.07$ , and the same initial conditions as in [32]. All the simulations are performed with  $p = 1$ . We present the results for a sequence of five uniform nested grids, with respectively 2048, 4096, 8192, 16,384, and 32,768 elements. The solutions obtained with the finest grid are denoted here as “numerically converged solutions”. Solid wall adiabatic no-slip boundary conditions with zero density gradient for the Eulerian model are imposed at the left and right boundaries of the domain.

**Fig. 11** Computational domain,  $hp$ -nonconforming grid, and solution polynomial degree distribution,  $p$ , for the simulation of the supersonic flow past a circular cylinder. The underlying figure was adapted from [26]

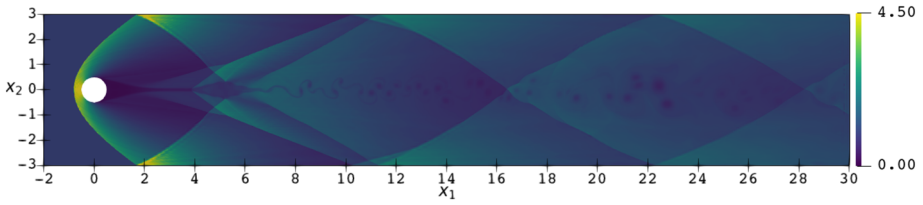


Svärd [32] observes that since the density flux is not bounded, the solution can converge to a state where the wall dissipates density. This is a non-physical solution and is computed with a non-entropy stable implementation of the CNS model [32]. Our discretization of the Eulerian and CNS models are entropy stable. As we will see shortly, we are unable to reproduce the “lagging” effect observed in [32].

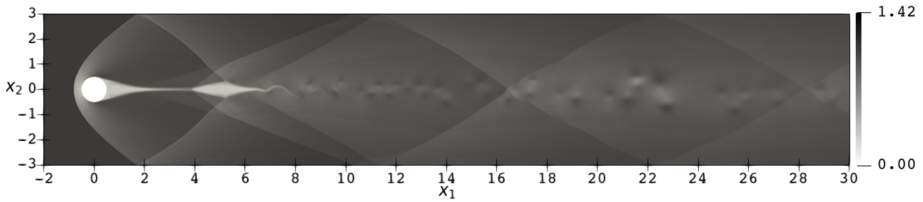
The main plot in Fig. 10 compares the density profiles computed with the Eulerian and the CNS models on the finest grid with 8192 elements. We can easily observe that the two curves do not lie on top of each other. Specifically, we notice a clear difference in the region near the maximum peak, whose position along the  $x$  axis also appears to be influenced by the model. Furthermore, differently from [32], we do not notice any difference at the left and right boundaries of the domain between the solution computed with the Eulerian and the CNS models. In Fig. 10, we also report two zoom views of the maximum peaks. There, we plot the density profiles obtained with the five nested grids. We can observe how the solutions approach the “numerically converged solution”. In particular, it appears that the solution obtained with the Eulerian model reaches the “converged state” faster than the solution obtained with the CNS model. Furthermore, we notice that the value of the maximum peak obtained with the Eulerian model is slightly smaller than the same value computed by the CNS model.

## 6.5 Supersonic flow around a cylinder

In this last test case, we present some results for the flow past a two-dimensional cylinder enclosed between two solid walls [6]. The similarity parameters are  $Re = 10,000$  and  $Ma = 3.5$ . The initial condition is a uniform flow with unit density,  $\rho$ , unit temperature,  $T$ , and only a non-zero unit velocity component in the  $x_1$  direction,  $u_1$ . The flow is computed with a non-uniform distribution of the solution polynomial degree. Therefore, the solution is computed using the entropy stable  $p$ -nonconforming interface technology [16]. The solution polynomial degree distribution is shown in Fig. 11, and it is the same setup used in [26]. Entropy stable adiabatic no-slip wall boundary conditions are enforced on the cylinder surface. Inviscid (slip) wall boundary conditions are imposed on the top and bottom horizontal boundaries, whereas far-field boundary conditions are used for the (left) inlet and the (right) outlet boundaries. The mesh consists of 5067 quadrangles. The solution computed with this setup is denoted here as a “numerically converged solution” in the sense that if we increase the order of accuracy of the method,  $p$ , or we refine the mesh, the difference in the solution for all the primitive variables is machine precision. Figure 12 shows the contour plot of the density,  $\rho$ , and the velocity component in the  $x_1$  direction,  $u_1$ , of the developed solution.

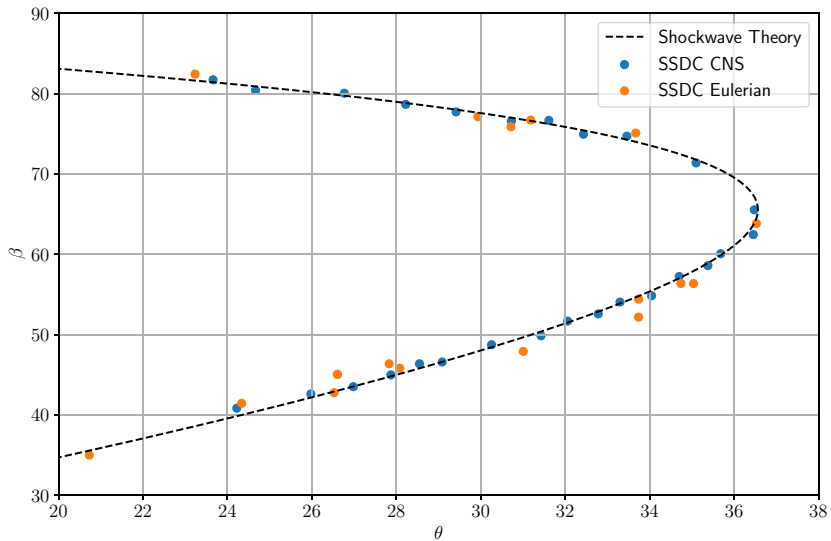


(a) Density,  $\rho$ .



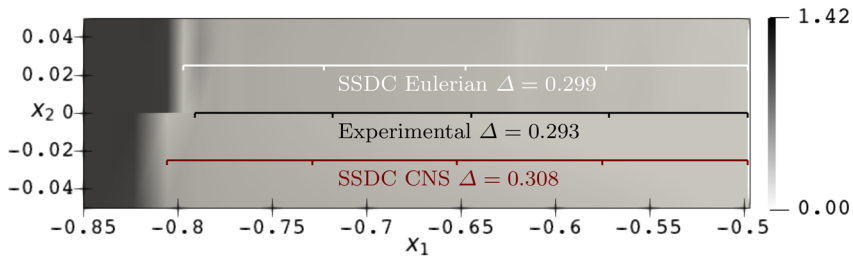
(b) Velocity component in the  $x_1$  direction,  $U_1$ .

**Fig. 12** Supersonic flow past a cylinder enclosed between two plates at  $Re = 10,000$  and  $Ma = 3.5$



**Fig. 13**  $\theta$ - $\beta$ - $Ma$  plot for the flow past a cylinder at  $Re = 10,000$ , and  $Ma = 3.5$

A quantitative analysis of the numerical results can be performed using the oblique shock wave theory for the Euler equations [30]. At an oblique shock, the flow changes direction, and there are three directions of interest: the upstream and downstream flow directions and the shock wave direction itself. The most useful relation of oblique shock wave theory is the one providing the deflection angle,  $\theta$ , as a function of the shock wave angle,  $\beta$ , and local



**Fig. 14** Normalized distance between the shock and the cylinder leading edge for the flow past a cylinder at  $Re = 10,000$ , and  $Ma = 3.5$ . The background is colored by the point-wise velocity magnitude

Mach number,  $Ma$  [21]:

$$\tan(\theta) = 2 \cot(\beta) \left[ \frac{Ma^2 \sin^2(\beta) - 1}{Ma^2(\gamma + \cos(2\beta)) + 2} \right]. \tag{55}$$

The pair  $\theta - \beta$  obtained by postprocessing the solutions computed with the Eulerian and CNS models are shown in Fig. 13. We can see that the results for both models are in good agreement with the theoretical curve. Finally, we report the normalized distance between the shock and the leading edge of the cylinder, as shown in Fig. 14. The value of the latter quantity is  $\Delta/D = 0.299$  and  $\Delta/D = 0.308$  for the Eulerian and CNS models, respectively. These values are in good agreement with the approximate value, which reads  $\Delta/D \simeq 0.193 \exp\left(\frac{4.67}{Ma^2}\right) = 0.293$  [4].

## 7 Conclusion

Guided by the entropy stability analysis, we have derived solid wall boundary conditions that preserve the entropy stability of the Eulerian model for viscous compressible fluids proposed by Svård [32]. In that context, using summation-by-parts operators and the simultaneous-approximation term technique, we have constructed entropy conservative and entropy stable solid wall boundary conditions for the semidiscretized system, which mimics the continuous entropy analysis results. The proposed boundary conditions have been validated in terms of accuracy, entropy conservation, and entropy stability for a set of test cases of increasing complexity. The numerical results obtained with the Eulerian model have been compared with the solutions computed using the classic compressible Navier–Stokes equations also discretized with an entropy stable methodology constructed using summation-by-parts operators. Differences and similarities have then been highlighted and the models are practically indistinguishable. This study is a first attempt to understand better the effects of the viscous flux term introduced in the conservation of mass equation.

**Acknowledgements** The research reported in this paper was funded by King Abdullah University of Science and Technology. We are thankful for the computing resources of the Supercomputing Laboratory and the Extreme Computing Research Center at King Abdullah University of Science and Technology.

**Data availability** The datasets generated during and/or analysed during the current study are available in the Zenodo repository, <http://doi.org/10.5281/zenodo.5041436>

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give

appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article’s Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article’s Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

### Appendix A: Derivation of the time derivative of the entropy

In this section, we provide the derivation of Eq. (35). To obtain the expression of the time derivative of the entropy function,  $\mathcal{S}$ , we multiply (33a) and (33b) by  $\mathbf{w}^\top \mathcal{P}$  and  $\left(\left[v \frac{dq}{dw}\right] \Theta_{x_j}\right)^\top \mathcal{P}$ , respectively. This step yields

$$\begin{aligned} & \frac{d}{dt} \mathbf{1}^\top \widehat{\mathcal{P}} \mathcal{S} + \mathbf{1}^\top \widehat{\mathcal{P}}_{x_j, x_k} \widehat{\mathcal{B}}_{x_i} \mathbf{F}_{x_i} - \mathbf{w}^\top \mathcal{P} \mathcal{D}_{x_i} \left[ v \frac{dq}{dw} \right] \Theta_{x_j} \\ &= \mathbf{w}^\top \mathcal{P}_{x_j, x_k} \left( \mathbf{g}_{x_i}^{(b), q} + \mathbf{g}_{x_i}^{(In), q} \right), \end{aligned} \tag{56a}$$

$$\begin{aligned} & \left( \left[ v \frac{dq}{dw} \right] \Theta_{x_j} \right)^\top \mathcal{P} \Theta_{x_i} - \left( \left[ v \frac{dq}{dw} \right] \Theta_{x_j} \right)^\top \mathcal{P} \mathcal{D}_{x_i} \mathbf{w} \\ &= \left( \left[ v \frac{dq}{dw} \right] \Theta_{x_j} \right)^\top \mathcal{P}_{x_j, x_k} \left( \mathbf{g}_{x_i}^{(b), \Theta} + \mathbf{g}_{x_i}^{(In), \Theta} \right). \end{aligned} \tag{56b}$$

Using the properties of SBP operators, we can write

$$\mathbf{w}^\top \mathcal{P} \mathcal{D}_{x_i} \left[ v \frac{dq}{dw} \right] \Theta_{x_j} = \mathbf{w}^\top \mathcal{P} \mathcal{P}_{x_i}^{-1} \mathcal{B}_{x_i} \left[ v \frac{dq}{dw} \right] \Theta_{x_j} - \mathbf{w}^\top \mathcal{P} \mathcal{P}_{x_i}^{-1} \mathcal{Q}_{x_i}^\top \left[ v \frac{dq}{dw} \right] \Theta_{x_j}. \tag{57}$$

Since  $\mathcal{P}$  is diagonal, we have

$$\mathcal{P} \mathcal{P}_{x_i}^{-1} \mathcal{Q}_{x_i}^\top = \mathcal{P} \mathcal{Q}_{x_i}^\top \mathcal{P}_{x_i}^{-1} = \mathcal{P} \mathcal{D}_{x_i}^\top = \mathcal{D}_{x_i}^\top \mathcal{P}. \tag{58}$$

Thus, Eq. (56a) is recast in the following form:

$$\begin{aligned} & \frac{d}{dt} \mathbf{1}^\top \widehat{\mathcal{P}} \mathcal{S} + \mathbf{1}^\top \widehat{\mathcal{P}}_{x_j, x_k} \widehat{\mathcal{B}}_{x_i} \mathbf{F}_{x_i} + (\mathcal{D}_{x_i} \mathbf{w})^\top \mathcal{P} \left[ v \frac{dq}{dw} \right] \Theta_{x_j} \\ &= \mathbf{w}^\top \mathcal{P}_{x_j, x_k} \mathcal{B}_{x_i} \left[ v \frac{dq}{dw} \right] \Theta_{x_j} + \mathbf{w}^\top \mathcal{P}_{x_j, x_k} \left( \mathbf{g}_{x_i}^{(b), q} + \mathbf{g}_{x_i}^{(In), q} \right). \end{aligned} \tag{59}$$

Additionally, because  $\left[ v \frac{dq}{dw} \right]$  is SPD, we have

$$\left( \left[ v \frac{dq}{dw} \right] \Theta_{x_j} \right)^\top \mathcal{P} \Theta_{x_i} = \left( \left[ v \frac{dq}{dw} \right]^{\frac{1}{2}} \Theta_{x_j} \right)^\top \mathcal{P} \left( \left[ v \frac{dq}{dw} \right]^{\frac{1}{2}} \Theta_{x_i} \right) = \left\| \left[ v \frac{dq}{dw} \right]^{\frac{1}{2}} \Theta_{x_i} \right\|_{\mathcal{P}}^2. \tag{60}$$

Now, substituting (56b) into (59) gives the following expression

$$\begin{aligned}
 & \frac{d}{dt} \mathbf{1}^\top \widehat{\mathcal{P}} \mathcal{S} + \left\| \left[ v \frac{dq}{dw} \right]^\frac{1}{2} \Theta_{x_i} \right\|_{\mathcal{P}}^2 \\
 &= -\mathbf{1}^\top \widehat{\mathcal{P}}_{x_j, x_k} \widehat{\mathcal{B}}_{x_i} \mathbf{F}_{x_i} \\
 &+ \mathbf{w}^\top \mathcal{P}_{x_j, x_k} \mathcal{B}_{x_i} \left[ v \frac{dq}{dw} \right] \Theta_{x_j} \\
 &+ \mathbf{w}^\top \mathcal{P}_{x_j, x_k} \left( \mathbf{g}_{x_i}^{(b), q} + \mathbf{g}_{x_i}^{(In), q} \right) \\
 &+ \left( \left[ v \frac{dq}{dw} \right] \Theta_{x_j} \right)^\top \mathcal{P}_{x_j, x_k} \left( \mathbf{g}_{x_i}^{(b), \Theta} + \mathbf{g}_{x_i}^{(In), \Theta} \right).
 \end{aligned} \tag{61}$$

By defining

$$\mathbf{DT} = \left\| \left[ v \frac{dq}{dw} \right]^\frac{1}{2} \Theta_{x_i} \right\|_{\mathcal{P}}^2, \tag{62}$$

and

$$\begin{aligned}
 \mathcal{E} &= -\mathbf{1}^\top \widehat{\mathcal{P}}_{x_j, x_k} \widehat{\mathcal{B}}_{x_i} \mathbf{F}_{x_i} \\
 &+ \mathbf{w}^\top \mathcal{P}_{x_j, x_k} \mathcal{B}_{x_i} \left[ v \frac{dq}{dw} \right] \Theta_{x_j} \\
 &+ \mathbf{w}^\top \mathcal{P}_{x_j, x_k} \left( \mathbf{g}_{x_i}^{(b), q} + \mathbf{g}_{x_i}^{(In), q} \right) \\
 &+ \left( \left[ v \frac{dq}{dw} \right] \Theta_{x_j} \right)^\top \mathcal{P}_{x_j, x_k} \left( \mathbf{g}_{x_i}^{(b), \Theta} + \mathbf{g}_{x_i}^{(In), \Theta} \right).
 \end{aligned} \tag{63}$$

Equation (61) can be further simplified to obtain (35), i.e.,

$$\frac{d}{dt} \mathbf{1}^\top \widehat{\mathcal{P}} \mathcal{S} + \mathbf{DT} = \mathcal{E}. \tag{64}$$

### Appendix B: Change of variables matrix, $\frac{dq}{dw}(\mathbf{v})$

The Jacobian of the conservative variables,  $q$ , in terms of the entropy variables,  $w$ , as a function of the primitive variables,  $v$ , is given by

$$\frac{dq}{dw}(\mathbf{v}) = \begin{bmatrix} \frac{\rho}{R} & \frac{\mathcal{U}_1 \rho}{R} & \frac{\mathcal{U}_2 \rho}{R} & \frac{\mathcal{U}_3 \rho}{R} & \frac{dq}{dw}(\mathbf{v})_{1,5} \\ \frac{\mathcal{U}_1 \rho}{R} & \left( \mathcal{T} + \frac{\mathcal{U}_1^2}{R} \right) \rho & \frac{\mathcal{U}_1 \mathcal{U}_2 \rho}{R} & \frac{\mathcal{U}_1 \mathcal{U}_3 \rho}{R} & \frac{dq}{dw}(\mathbf{v})_{2,5} \\ \frac{\mathcal{U}_2 \rho}{R} & \frac{\mathcal{U}_1 \mathcal{U}_2 \rho}{R} & \left( \mathcal{T} + \frac{\mathcal{U}_2^2}{R} \right) \rho & \frac{\mathcal{U}_2 \mathcal{U}_3 \rho}{R} & \frac{dq}{dw}(\mathbf{v})_{3,5} \\ \frac{\mathcal{U}_3 \rho}{R} & \frac{\mathcal{U}_1 \mathcal{U}_3 \rho}{R} & \frac{\mathcal{U}_2 \mathcal{U}_3 \rho}{R} & \left( \mathcal{T} + \frac{\mathcal{U}_3^2}{R} \right) \rho & \frac{dq}{dw}(\mathbf{v})_{4,5} \\ \frac{dq}{dw}(\mathbf{v})_{1,5} & \frac{dq}{dw}(\mathbf{v})_{2,5} & \frac{dq}{dw}(\mathbf{v})_{3,5} & \frac{dq}{dw}(\mathbf{v})_{4,5} & \frac{dq}{dw}(\mathbf{v})_{5,5} \end{bmatrix}, \tag{65}$$

where

$$\frac{dq}{dw}(\mathbf{v})_{1,5} = \frac{\rho \left( -2RT + \|\mathcal{U}\|^2 + 2\mathcal{T}c_{pr} \right)}{2R}, \tag{66}$$



$$\frac{dq}{dw}(v)_{2,5} = \frac{\mathcal{U}_1 \rho (\|\mathcal{U}\|^2 + 2\mathcal{T}c_{pr})}{2R}, \tag{67}$$

$$\frac{dq}{dw}(v)_{3,5} = \frac{\mathcal{U}_2 \rho (\|\mathcal{U}\|^2 + 2\mathcal{T}c_{pr})}{2R}, \tag{68}$$

$$\frac{dq}{dw}(v)_{4,5} = \frac{\mathcal{U}_3 \rho (\|\mathcal{U}\|^2 + 2\mathcal{T}c_{pr})}{2R}, \tag{69}$$

$$\frac{dq}{dw}(v)_{5,5} = \frac{\rho \left( (\|\mathcal{U}\|^2)^2 + 4\mathcal{T}c_{pr}(-RT + \|\mathcal{U}\|^2 + \mathcal{T}c_{pr}) \right)}{4R}. \tag{70}$$

### Appendix C: Viscous boundary penalty

This section constructs the viscous boundary penalty term for an adiabatic wall and a wall with heat entropy flux. We then show that the resulting expressions lead to an entropy conservative and entropy stable penalty terms, respectively.

Assuming a dissipative internal penalty term,  $\mathcal{M} = 0$ , the RHS of (35) becomes

$$\frac{d}{dt} \mathbf{1}^\top \widehat{\mathcal{P}} \mathcal{S} + \mathbf{D}\mathbf{T} = w^\top \widehat{\mathcal{P}}_{x_j, x_k} \left( g_{x_i}^{(b, V), q} \right) + \Theta_{x_i}^\top \left( v \frac{dq}{dw} \right) \widehat{\mathcal{P}}_{x_j, x_k} g_{x_i}^{(b, \Theta)}. \tag{71}$$

Define  $\mathbf{W} = \text{diag}(w)$  and write

$$w^\top \widehat{\mathcal{P}}_{x_j, x_k} \chi = \mathbf{1}^\top \mathbf{W} \widehat{\mathcal{P}}_{x_j, x_k} \chi = \mathbf{1}^\top \widehat{\mathcal{P}}_{x_j, x_k} \mathbf{W} \chi, \tag{72}$$

where  $\mathbf{W} \mathcal{P}_{x_j, x_k} = \mathcal{P}_{x_j, x_k} \mathbf{W}$ , since  $\mathbf{W}$  is diagonal, and  $\chi$  is a vector on a node. We use (72) for both terms in the RHS of (71). The first term reads

$$\frac{1}{2} \mathbf{1}^\top \widehat{\mathcal{P}}_{x_j, x_k} \widehat{\mathcal{B}}_{x_i}^- \mathbf{W} \left( v \frac{dq}{dw} \right) \Theta_{x_i} - \frac{1}{2} \mathbf{1}^\top \widehat{\mathcal{P}}_{x_j, x_k} \widehat{\mathcal{B}}_{x_i}^- \mathbf{W} \left( v \frac{dq}{dw} \left( v^{(b, V)} \right) \right) \widetilde{\Theta}_{x_i}, \tag{73}$$

where both  $\frac{dq}{dw} \left( v^{(b, V)} \right)$  and  $\widetilde{\Theta}$  are evaluated using the manufactured primitive state

$$v^{(b, V)} = \left( \rho, -\mathcal{U}_1 + 2\mathcal{U}_1^{\text{wall}}, -\mathcal{U}_2 + 2\mathcal{U}_2^{\text{wall}}, -\mathcal{U}_3 + 2\mathcal{U}_3^{\text{wall}}, T \right)^\top. \tag{74}$$

The kinematic viscosity,  $\nu$ , depends only on the density  $\rho$ . Thus, its evaluation is the same for  $v$  and  $v^{(b, V)}$ . The procedure for computing  $\widetilde{\Theta}_{x_i}$  is detailed in [13] and is summarized in the following equation. The manufactured entropy variables gradient is defined as

$$\widetilde{\Theta}_{x_i} = \frac{dw}{dv} \left( v^{(b, V)} \right) \widetilde{\Pi} = \frac{dw}{dv} \left( v^{(b, V)} \right) \text{diag}(-1, 1, 1, 1, -1) \Pi, \tag{75}$$

where  $\Pi$  is the gradient of primitive variables. The second term on the RHS of (71) is written as

$$\frac{1}{2} \mathbf{1}^\top \widehat{\mathcal{P}}_{x_j, x_k} \widehat{\mathcal{B}}_{x_i}^- \mathbf{W} \left( v \frac{dq}{dw} \right) \Theta_{x_i} - \frac{1}{2} \mathbf{1}^\top \widehat{\mathcal{P}}_{x_j, x_k} \widehat{\mathcal{B}}_{x_i}^- \mathbf{W} \left( v^{(b, V)} \right) \left( v \frac{dq}{dw} \right) \Theta_{x_i}. \tag{76}$$

The second terms in (73) and (76) cancel out. On the other hand, the first terms add up to

$$\mathbf{1}^\top \widehat{\mathcal{P}}_{x_j, x_k} \widehat{\mathcal{B}}_{x_i}^- \mathbf{W} \left( v \frac{dq}{dw} \right) \Theta_{x_i} = \mathbf{1}^\top \widehat{\mathcal{P}}_{x_j, x_k} g(t). \tag{77}$$

Therefore, we arrive at (50), i.e.,

$$\frac{d}{dt} \mathbf{1}^\top \widehat{\mathcal{P}} \mathcal{S} + \mathbf{D}\mathbf{T} = \mathbf{1}^\top \widehat{\mathcal{P}}_{x_j, x_k} g(t). \tag{78}$$

For an adiabatic wall  $g(t) = 0$ , and thus, the RHS is zero yielding an entropy conservative term. For a wall with non-zero heat entropy flux,  $g(t)$  is data, and the RHS is bounded.

## Appendix D: Interior penalty term, $\mathcal{M}^{(b,V)}$

In this section, we compute the contribution to time derivative of the entropy function,  $\mathcal{S}$ , from the interior penalty term  $\mathcal{M}^{(b,V)}$ . Assuming a uniform hexahedral element (i.e., a cube) of length one and a wall in the  $x_i$  direction, we use the manufactured state of the vector of primitive variables at the wall (47) to construct the jump in the entropy variables in the normal direction to the wall,  $\hat{n}$ ,

$$\Delta w^{(b,V)} = (w - w^{(b,V)}) \cdot \hat{n}^\top, \quad (79)$$

and the viscous boundary flux state

$$f^{(V)}(v^{(b,V)}) = \left( v \frac{dq}{dw}(v^{(b,V)}) \right) \Delta w^{(b,V)} \cdot \hat{n}. \quad (80)$$

Then, we use an average of the viscous flux of the numerical solution state at the wall and the manufactured state

$$\mathcal{M}^{(b,V)} = -\beta \frac{f^{(V)}(v) + f^{(V)}(v^{(b,V)})}{2}, \quad (81)$$

to compute the dissipative term with Mathematica [36]

$$w^\top \mathcal{M}^{(b,V)} = -\frac{2\beta\alpha\mu}{RT^2} \|\Delta \mathcal{U}\|^2 (\|\Delta \mathcal{U}\|^2 + RT), \quad (82)$$

where  $\Delta \mathcal{U}$  is the jump in the velocity of the numerical solution state at the wall and the manufactured state. The term is negative for any non-zero jump in velocity, and thus, is entropy dissipative.

## References

1. Abhyankar, S., Brown, J., Constantinescu, E.M., Ghosh, D., Smith, B.F., Zhang, H.: PETSc/TS: a modern scalable ODE/DAE solver library (2018). [arXiv:1806.01437](https://arxiv.org/abs/1806.01437)
2. Balay, S., Abhyankar, S., Adams, M.F., Brown, J., Brune, P., Buschelman, K., Dalcin, L., Dener, A., Eijkhout, V., Gropp, W.D., Karpeyev, D., Kaushik, D., Knepley, M.G., May, D.A., McInnes, L.C., Mills, R.T., Munson, T., Rupp, K., Sanan, P., Smith, B.F., Zampini, S., Zhang, H., Zhang, H.: PETSc users manual. Technical Report. ANL-95/11-Revision 3.15, Argonne National Laboratory (2021)
3. Berg, J., Nordström, J.: Stable robin solid wall boundary conditions for the Navier–Stokes equations. *J. Comput. Phys.* **230**(19), 7519–7532 (2011)
4. Billig, F.S.: Shock-wave shapes around spherical-and cylindrical-nosed bodies. *J. Spacecr. Rockets* **4**(6), 822–823 (1967)
5. Bogacki, P., Shampine, L.: A 3(2) pair of Runge–Kutta formulas. *Appl. Math. Lett.* **2**(4), 321–325 (1989)
6. Boukharfane, R., Ribeiro, F.H.E., Bouali, Z., Mura, A.: A combined ghost-point-forcing/direct-forcing immersed boundary method (IBM) for compressible flow simulations. *Comput. Fluids* **162**, 91–112 (2018)
7. Calhoun, D.: A Cartesian grid method for solving the two-dimensional streamfunction-vorticity equations in irregular regions. *J. Comput. Phys.* **176**(2), 231–275 (2002)
8. Carpenter, M.H., Gottlieb, D., Abarbanel, S.: Time-stable boundary conditions for finite-difference schemes solving hyperbolic systems: methodology and application to high-order compact schemes. *J. Comput. Phys.* **111**(2), 220–236 (1994)
9. Carpenter, M.H., Fisher, T.C., Nielsen, E.J., Frankel, S.H.: Entropy stable spectral collocation schemes for the Navier–Stokes equations: discontinuous interfaces. *SIAM J. Sci. Comput.* **36**(5), B835–B867 (2014)

10. Carpenter, M.H., Parsani, M., Fisher, T.C., Nielsen, E.J.: Entropy stable staggered grid spectral collocation for the Burgers' and compressible Navier–Stokes equations. NASA/TM-2015-218990 (2015)
11. Chandrashekar, P.: Kinetic energy preserving and entropy stable finite volume schemes for compressible Euler and Navier–Stokes equations. *Commun. Comput. Phys.* **14**(5), 1252–1286 (2013)
12. Dafermos, C.M., Dafermos, C.M., Dafermos, C.M., Mathématicien, G., Dafermos, C.M., Mathematician, G.: *Hyperbolic Conservation Laws in Continuum Physics*, vol. 3. Springer, Berlin (2005)
13. Dalcin, L., Rojas, D., Zampini, S., Fernández, D.C.D.R., Carpenter, M.H., Parsani, M.: Conservative and entropy stable solid wall boundary conditions for the compressible Navier–Stokes equations: adiabatic wall and heat entropy transfer. *J. Comput. Phys.* **397**, 108775 (2019)
14. Dennis, S., Chang, G.Z.: Numerical solutions for steady flow past a circular cylinder at Reynolds numbers up to 100. *J. Fluid Mech.* **42**(3), 471–489 (1970)
15. Dolejší, V., Svärd, M.: Numerical study of two models for viscous compressible fluid flows. *J. Comput. Phys.* **427**, 110068 (2021)
16. Fernández, D.C.D.R., Carpenter, M.H., Dalcin, L., Fredrich, L., Winters, A.R., Gassner, G.J., Parsani, M.: Entropy-stable p-nonconforming discretizations with the summation-by-parts property for the compressible Navier–Stokes equations. *Comput. Fluids* **210**, 104631 (2020)
17. Fisher, T.C.: High-order  $L^2$  stable multi-domain finite difference method for compressible flows. PhD thesis, Purdue University (2012)
18. Fornberg, B.: A numerical study of steady viscous flow past a circular cylinder. *J. Fluid Mech.* **98**(4), 819–855 (1980)
19. Geuzaine, C., Remacle, J.F.: Gmsh: a 3-D finite element mesh generator with built-in pre-and post-processing facilities. *Int. J. Numer. Methods Eng.* **79**(11), 1309–1331 (2009)
20. Knepley, M.G., Karpeev, D.A.: Mesh algorithms for Sieve I: mesh distribution. *Sci. Program.* **17**(3), 215–230 (2009). <https://doi.org/10.3233/SPR-2009-0249>
21. Liepmann, H.W., Roshko, A.: *Elements of Gasdynamics*. Courier Corporation, Gloucester (2001)
22. Nordström, J., Carpenter, M.H.: Boundary and interface conditions for high-order finite-difference methods applied to the Euler and Navier–Stokes equations. *J. Comput. Phys.* **148**(2), 621–645 (1999)
23. Park, J., Kwon, K., Choi, H.: Numerical solutions of flow past a circular cylinder at Reynolds numbers up to 160. *KSME Int. J.* **12**(6), 1200–1205 (1998)
24. Parsani, M., Carpenter, M.H., Nielsen, E.J.: Entropy stable discontinuous interfaces coupling for the three-dimensional compressible Navier–Stokes equations. *J. Comput. Phys.* **290**, 132–138 (2015a)
25. Parsani, M., Carpenter, M.H., Nielsen, E.J.: Entropy stable wall boundary conditions for the three-dimensional compressible Navier–Stokes equations. *J. Comput. Phys.* **292**, 88–113 (2015b)
26. Parsani, M., Boukharfane, R., Nolasco, I.R., Fernández, D.C.D.R., Zampini, S., Hadri, B., Dalcin, L.: High-order accurate entropy-stable discontinuous collocated Galerkin methods with the summation-by-parts property for compressible CFD frameworks: scalable SSDC algorithms and flow solver. *J. Comput. Phys.* **424**, 109844 (2020)
27. Rosenhead, L.: *Laminar Boundary Layers; An Account of the Development, Structure, and Stability of Laminar Boundary Layers in Incompressible Fluids, Together with a Description of the Associated Experimental Techniques*, 1st edn. Clarendon Press, Oxford (1963)
28. Russell, D., Wang, Z.J.: A Cartesian grid method for modeling multiple moving objects in 2D incompressible viscous flow. *J. Comput. Phys.* **191**(1), 177–205 (2003)
29. Schneiderbauer, S., Krieger, M.: What do the Navier–Stokes equations mean? *Eur. J. Phys.* **35**(1), 015020 (2013)
30. Shapiro, A.H.: *The Dynamics and Thermodynamics of Compressible Fluid Flow*. Engineering Special Collection, vol. 1. Ronald Press Company, New York (1953)
31. Svärd, M.: Weak solutions and convergent numerical schemes of modified compressible Navier–Stokes equations. *J. Comput. Phys.* **288**, 19–51 (2015)
32. Svärd, M.: A new Eulerian model for viscous and heat conducting compressible flows. *Phys. A Stat. Mech. Appl.* **506**, 350–375 (2018)
33. Svärd, M., Nordström, J.: A stable high-order finite difference scheme for the compressible Navier–Stokes equations: no-slip wall boundary conditions. *J. Comput. Phys.* **227**(10), 4805–4824 (2008)
34. Svärd, M., Nordström, J.: Review of summation-by-parts schemes for initial-boundary-value problems. *J. Comput. Phys.* **268**, 17–38 (2014)
35. Tadmor, E.: Entropy stability theory for difference approximations of nonlinear conservation laws and related time-dependent problems. *Acta Numer.* **12**, 451–512 (2003)
36. Wolfram Research, Inc: *Mathematica*, Version 12.1. <https://www.wolfram.com/mathematica>, Champaign, IL (2020). Accessed 11 Apr 2020
37. Ye, T., Mittal, R., Udaykumar, H., Shyy, W.: An accurate Cartesian grid method for viscous incompressible flows with complex immersed boundaries. *J. Comput. Phys.* **156**(2), 209–240 (1999)