Computers and Fluids 257 (2023) 105869

Contents lists available at ScienceDirect

Computers and Fluids

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

A modified equation analysis for immersed boundary methods based on volume penalization: Applications to linear advection–diffusion equations and high-order discontinuous Galerkin schemes

Victor J. Llorente [a,b], Jiaqing Kou [a,*], Eusebio Valero [a,b], Esteban Ferrer [a,b]

[a] ETSIAE-UPM - School of Aeronautics, Universidad Politécnica de Madrid, Plaza Cardenal Cisneros 3, E-28040 Madrid, Spain
[b] Center for Computational Simulation, Universidad Politécnica de Madrid, Campus de Montegancedo, Boadilla del Monte, 28660 Madrid, Spain

ARTICLE INFO

Keywords:
Discontinuous Galerkin
Immersed boundary method
Modified equation analysis
Volume penalization

ABSTRACT

The Immersed Boundary Method (IBM) is a popular numerical approach to impose boundary conditions without relying on body-fitted grids, thus reducing the costly effort of mesh generation. To obtain enhanced accuracy, IBM can be combined with high-order methods (e.g., discontinuous Galerkin). For this combination to be effective, an analysis of the numerical errors is essential. In this work, we apply, for the first time, a modified equation analysis to the combination of IBM (based on volume penalization) and high-order methods (based on nodal discontinuous Galerkin methods) to analyze *a priori* numerical errors and obtain practical guidelines on the selection of IBM parameters. The analysis is performed on a linear advection–diffusion equation with Dirichlet boundary conditions. Three ways to penalize the immersed boundary are considered, the first penalizes the solution inside the IBM region (classic approach), whilst the second and third penalize the first and second derivatives of the solution. We find optimal combinations of the penalization parameters, including the first and second penalizing derivatives, resulting in minimum errors. We validate the theoretical analysis with numerical experiments for one- and two-dimensional advection–diffusion equations.

1. Introduction

Despite successful applications of Immersed Boundary Method (IBM) in simulating complex flows [1,2] and fluid–structure interaction problems [3–5], understanding and controlling numerical errors in the IBM approach remains a challenge. IBM refers to a group of numerical strategies that handle the boundary condition when the solid is immersed in the computational domain, avoiding body-fitted meshes and enabling the use of simple meshes (e.g., Cartesian or Octree). The IBM approach originates from the idea of Peskin [6], where singular forces represented by delta functions were positioned at solid boundaries to mimic the effect of physical boundaries. Since IBM reduces the complexity of mesh generation and handles moving boundaries efficiently, it has received a lot of attention over the past few decades. In general, IBM treatment can be achieved using the cut-cell approach [7–10] or by introducing source terms, e.g., ghost cell [11], projection method [12], direct forcing [13,14] or volume penalization [15,16]. Although the cut-cell approach shows better convergence properties, the extension to moving boundaries and the treatment of different types of cut-cells remain challenging. A more flexible approach is the IBM based on Volume Penalization (VP). The latter shows advantages in robustness, ease of implementation, and theoretical convergence estimates [15,17].

Volume penalization is a classic IBM treatment based on modeling the solid as a porous medium with low permeability [18,19]. The method imposes the boundary condition by introducing a source term (or penalty term) to the computational nodes located inside the solid. This approach dates back to the work of Courant [20], where a penalty method was used to transform constrained optimization problems into a constraint-free problem. Volume penalization methods for the Navier–Stokes equations were first proposed by Arquis and Caltagirone [17] with a Brinkman-type penalization for the momentum equations. After that, Angot et al. [15] and Carbou and Fabrie [21] proved the convergence of volume penalization, showing that as the penalization parameter η approaches 0, the model error converges if no-slip boundary conditions are considered. Subsequently, the volume penalization was extended to allow Neumann boundary conditions [18,19] and Robin boundary conditions [22], as well as spatially varying Neumann and Robin boundary conditions [23]. The volume penalization method was extended to compressible flows by Liu and Vasilyev [24], Brown–Dymkoski et al. [25], Abgrall et al. [26] and Abalakin et al. [27]. This

* Corresponding author.
E-mail address: jiaqing.kou@alumnos.upm.es (J. Kou).

https://doi.org/10.1016/j.compfluid.2023.105869
Received 22 December 2022; Received in revised form 20 February 2023; Accepted 17 March 2023
Available online 21 March 2023

method has been applied to a variety of problems, including flapping wings [16], two-phase flows [28], aeroacoustics [29], fluid–structure interactions [30] and thermal flows [31]. So far, IBM research for high-order methods has been relatively unexplored, with efforts focused on Poisson problems [32,33] and cut-cell approaches [9,34]. In the context of volume penalization, we have recently extended this approach to high-order flux reconstruction schemes [35,36], and now to the high-order Discontinuous Galerkin Spectral Element Method (DGSEM) in this work.

There have been several attempts to analyze the errors of the IBM approach. Bever and Leveque [37] analyzed the error of traditional IBM applied to one-dimensional problems, and highlighted the importance of choosing appropriate discrete delta functions to maintain optimal accuracy. Following a similar strategy, the immersed interface method [38], which modifies the finite difference scheme with a jump condition for the immersed boundary, was derived [39]. Tornberg and Engquist [40] performed error analyses of traditional IBM with regularization and found first-order convergence for the standard central difference scheme with smoothing discrete delta functions. This error analysis was then extended to Stokes flows by Mori [41], Chen et al. [42], and Liu and Mori [43] where error estimates for velocity and pressure were reported. Most analyses focus on the traditional IBM method, where the numerical property is based on the selection of the appropriate delta functions. Error analyses for the direct forcing approach were also explored in [44,45], where the importance of maintaining smoothness in the solution and the choice of a suitable temporal and spatial resolution was highlighted. For these types of approach, the discretization error from space–time discretization and the modeling error from particular IBM treatment are coupled. In contrast, volume penalization has the advantage that the modeling error and the numerical error can be handled separately. The convergence of modeling errors was studied rigorously by Angot et al. [15] and Carbou and Fabrie [21]. The modeling errors for the incompressible flow past a cylinder and a sphere were analyzed by Zhang and Zheng [46]. Therefore, the main concern is the discretization error, which depends on the numerical scheme used, and where a detailed error analysis is lacking, especially in the context of high-order methods.

In the present work, we perform error analyses of the IBM based on combination of volume penalization and nodal DGSEM, and propose new penalties to cancel spatial errors to improve the accuracy of the solution. In particular, we use the modified equation analysis [47,48], which has been extensively used to analyze the stability and accuracy of low-order numerical discretization, and to obtain high-order schemes [49,50]. The relationship between the errors introduced by the IBM based on volume penalization and high-order schemes remains unclear and motivates this work. First, using a modified equation analysis for volume penalization using DGSEM, we determine the shape and relationship of the dominant errors (i.e. dissipative/dispersive character). Second, we design the volume penalization scheme by including additional penalty terms that cancel the undesired numerical errors. In recent work, we have attempted to damp the numerical errors that arise from the volume penalization approach, using second-order derivatives [51] or combining it with a frequency damping technique [52]. These studies have tried to minimize errors, without explicitly considering the causes of such errors, and therefore can be considered *a posteriori* palliative treatment. In this work, we consider a different perspective and analyze the source of these errors. By doing so, we are able to cancel the errors at the source. This approach can be considered an *a priori* error control. Note that we limit our analysis to linear advection–diffusion equations. The results from our analysis can thus be extended to linear systems (or linearized version of nonlinear equations). Examples include acoustics [53] and stability analysis [54].

The article is organized as follows. In Section 2, the volume penalization method and the DGSEM technique are introduced for the governing equation. Next, Section 3 introduces the principal errors in a volume penalization approach to investigate the discretization error using the modified equation analysis in Section 4. The numerical results are shown in Section 5, to validate the conclusions of the analysis, where one and two-dimensional advection–diffusion equations are investigated. Finally, conclusions are given in Section 6.

2. Motivation

2.1. The governing equation

Let us introduce the problem by considering the following time-dependent 1D advection–diffusion equation for the transported solution $u = u(x,t)$,

$$\frac{\partial u}{\partial t} + c\frac{\partial u}{\partial x} - \nu\frac{\partial^2 u}{\partial x^2} = 0, \quad \text{for } x \in (0,L),\ t > 0, \tag{1a}$$

where the flow parameters are constant: velocity field c and kinetic viscosity $\nu \geq \nu_{\min} > 0$. The PDE is completed with the set of initial and boundary conditions,

$$u(x,0) = u^0(x), \qquad 0 \leq x \leq L, \tag{1b}$$

$$u(0,t) = u_0(t), \quad u(L,t) = u_L(t), \qquad t \geq 0. \tag{1c}$$

The transport problem (1) is discretized in space based on a high-order DG method; in time, a Runge–Kutta method; and some of the solution points would be penalized by additional source terms to impose the IBM conditions.

2.2. The volume penalization approach

Motivated by the characteristic-based VP [25] and the inclusion of local dissipation [51], we consider the governing equation with penalization terms for the solution (classic volume penalization [15,18,30,55]) and additional first-order and second-order penalization terms:

$$\frac{\partial u}{\partial t} + c\frac{\partial u}{\partial x} - \nu\frac{\partial^2 u}{\partial x^2} + \frac{\chi}{\eta_1}(u - u_s) + \frac{\partial}{\partial x}\left(\frac{\chi}{\eta_2}u\right) + \frac{\partial^2}{\partial x^2}\left(\frac{\chi}{\eta_3}u\right) = 0, \tag{2a}$$

The additional term in Eq. (2a) helps to impose the immersed boundary condition on a given region of the domain $\Omega = [0, L]$. In this work, we consider a boundary condition of homogeneous Dirichlet type, namely $u_s = u_s(x,t) = 0$ (that is, a no-slip wall). The other two parameters are penalized terms determined in *the modified equation analysis* section, where we focus only on the spatial errors of the discretization. The penalization parameters for variable, first and second order derivatives are η_1, η_2, and η_3 respectively. In the classic volume penalization approach [15,18,30,55], the mask function χ is sharp and discontinuous, which is 1 in the solid and 0 in the fluid. Here, to facilitate the analysis, a continuous mask function represented by a hyperbolic tangent function is used, defined as

$$\chi = \chi(x,t) = \begin{cases} \left[\tanh\left(d/\delta\right) + 1\right]/2, & \text{If } x \in \Omega_s \\ \left[\tanh\left(-d/\delta\right) + 1\right]/2, & \text{Otherwise} \end{cases}, \tag{2b}$$

which distinguishes between the fluid, Ω_f, and the solid, Ω_s, regions such that $\Omega = \Omega_f \cup \Omega_s$. The distance of any solution point from the boundary interface is defined as $d = d(x,t)$. This smooth mask ensures that spatial derivatives can be calculated. The width of the hyperbolic tangent function is defined as δ, which should be infinitely small to reduce the modeling error and approximate the sharp mask function. The mask function at different δ is compared in Fig. 1, where the sharp mask function can be well represented when $\delta \approx 10^{-3}$. Note that in the figure, the mask function at $\delta = 10^{-3}$ almost overlaps with the sharp mask function. Later in Section 5.1, we will also show that both sharp and smooth masks result in similar behavior given a sufficiently small δ (e.g., $\delta < 10^{-2}$).

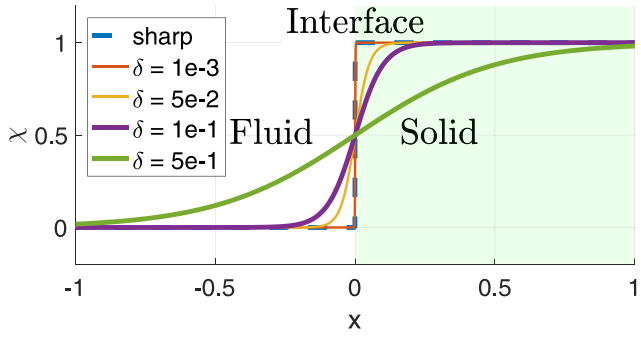

Fig. 1. Comparison of sharp and smooth mask function at different δ. The interface lies at $x = 0$, therefore $d = |x|$. The sharp mask function is well represented when $\delta < 10^{-3}$.

2.3. The VP-DG discrete equation

Eq. (2a) is discretized using the DG spectral element method. We group the terms in Eq. (2a) that leads to the penalized equation:

$$\frac{\partial u}{\partial t} + \mathcal{L}u = 0, \tag{3a}$$

where the second-order differential operator is represented by

$$\mathcal{L}u = \frac{\partial}{\partial x}\left(\hat{c}u - \hat{\nu}\frac{\partial u}{\partial x}\right) + \frac{\chi}{\eta_1}u, \tag{3b}$$

where $\hat{c}$ = and $\hat{\nu}$ are the VP velocity field and the VP viscosity, respectively. Note that here we consider the element to be either a fully solid or a fully fluid one. This implies that the solid boundary aligns with the element interface, which is a natural choice for the DG method when using the local r-refinement, e.g. [56].

The domain Ω is divided into multiple subdomains named elements $\Omega^k = [x_{k-1}, x_k]$, $k = 1, 2, \dots, K$, as can be seen in Fig. 2, and mapped to the reference interval $\xi \in [-1, 1]$. The global solution is assumed to be approximated by a piecewise polynomial defined as the direct sum $\oplus$ of the K local polynomial solutions,

$$u(x,t) \simeq u_h(x,t) = \bigoplus_{k=1}^{K} u_h^k(x(\xi),t), \tag{4}$$

also for the test function and the VP flux function. On each element, we describe the local solution by the Legendre orthogonal interpolating polynomial, which is written in the Lagrange form,

$$u_h^k(\xi,t) = \sum_{j=0}^{N} u_{h,j}^k(t)\, l_j(\xi). \tag{5}$$

We select Gauss–Lobatto (GL) points, as they are becoming very popular in newly energy-stable and entropy-conserving schemes [57,58]. Using GL points, the nodal (grid point) values become $u_{h,j}^k(t) = u_h^k(\xi_j, t)$, and $l_j(\xi)$ is the Nth order Lagrange interpolating polynomial,

$$l_j(\xi) := \prod_{\substack{i=0 \\ i\neq j}}^{N} \frac{\xi - \xi_i}{\xi_j - \xi_i}, \tag{6}$$

which satisfies $l_j(\xi_i) = \delta_{ij}$, being δ_{ij} the Kronecker delta. After obtaining weak forms of Eq. (3a) and applying the Gaussian quadrature to the inner product in the reference interval (see Appendix A for details), the semi-discrete equation writes as follows:

$$\frac{\mathrm{d}u_{h,j}^k}{\mathrm{d}t} + \sum_{i=0}^{N} D_{ij}^k u_{h,i}^k = S_j^k, \tag{7a}$$

for $k = 1, 2, \dots, K$ and $j = 0, 1, \dots, N$ where

$$D_{ij}^k := \frac{\chi_j^k}{\eta_1}\delta_{ij} - \frac{2}{\Delta x_k}\hat{c}_i^k \frac{w_i}{w_j} l_j'(\xi_i) - \left(\frac{2}{\Delta x_k}\right)^2 \hat{\nu}_i^k \frac{w_i}{w_j}\sum_{r=0}^{N} l_j'(\xi_r) l_r'(\xi_i), \tag{7b}$$

is the VP-DG derivative, and

$$S_j^k := \frac{2}{\Delta x_k}\frac{\mathcal{F}_{-1}^k l_j(-1) - \mathcal{F}_1^k l_j(1)}{w_j} + \left(\frac{2}{\Delta x_k}\right)^2 \frac{\hat{\nu}_0^k \mathcal{U}_{-1}^k l_j'(-1) - \hat{\nu}_N^k \mathcal{U}_1^k l_j'(1)}{w_j}, \tag{7c}$$

the numerical source. The weights of the GL quadrature are $\{w_i\}_{i=0}^N$ and l' represents the derivative of the Lagrange polynomial. In the previous formula, we define the numerical fluxes as $\mathcal{F}$ and $\mathcal{U}$ (see Appendix A). These fluxes are given by:

$$\mathcal{F}_{-1}^k = \mathfrak{f}_2^{k-1} u_{h,2}^{k-1} + \mathfrak{f}_0^k u_{h,0}^k, \qquad \mathcal{U}_{-1}^k = \mathfrak{g}_2^{k-1} u_{h,2}^{k-1} + \mathfrak{g}_0^k u_{h,0}^k, \tag{8a}$$

$$\mathcal{F}_1^k = \mathfrak{f}_2^k u_{h,2}^k + \mathfrak{f}_0^{k+1} u_{h,0}^{k+1}, \qquad \mathcal{U}_1^k = \mathfrak{g}_2^k u_{h,2}^k + \mathfrak{g}_0^{k+1} u_{h,0}^{k+1}. \tag{8b}$$

The weights $\mathfrak{f}$ and $\mathfrak{g}$ depend on $\hat{c}$, $\hat{\nu}$, and some numerical parameters that determine the advective/diffusive flux scheme used; see Appendix A for details . To calculate the viscous flux, we have considered the Bassi–Rebay 1 (BR1) scheme [59] and the Local discontinuous Galerkin (LDG) scheme [60].

3. Errors in volume penalization

Rigorous proofs of the convergence of modeling errors have been provided in previous work [15,21], showing that the numerical error introduced from the penalization term can be controlled *a priori* [25]. Analysis of volume penalization suggests that the two contributions to total error are modeling and discretization errors [30]:

$$\left\| u^{\text{exact}} - u_\eta^{\text{num.}} \right\| \leq \left\| u^{\text{exact}} - u_\eta \right\| + \left\| u_\eta - u_\eta^{\text{num.}} \right\|, \tag{9}$$

where u^{exact} is the exact solution of the governing equation, u_η and $u_\eta^{\text{num.}}$ are the exact and numerical solutions of the penalized equation, respectively, and $\|\cdot\|$ is the L_p norm used to quantify the error. The modeling error depends on the penalization parameter [55]:

$$\left\| u^{\text{exact}} - u_\eta \right\| \propto \eta_1^\alpha. \tag{10}$$

This explains that the convergence of the solution to the exact solution requires the error norm to approach zero for small penalization parameter limit, i.e.,

$$\lim_{\eta_1 \to 0} \left\| u^{\text{exact}} - u_\eta \right\| = 0. \tag{11}$$

According to Angot et al. [15] and Carbou and Fabrie [21], the volume penalization gives $\alpha = 1/2$, indicating that the penalization error has a decay rate of $\mathcal{O}(\sqrt{\eta_1})$ for Dirichlet boundary conditions. For the Neumann boundary conditions, a decay rate of $\mathcal{O}(\eta_1)$ can be obtained [61]. It should be noted that the theory is based on the classical volume penalization approach, where a sharp mask function is considered. In the present theoretical analysis, the result holds when we consider δ in Eq. (2b) to be infinitely small approximating the sharp mask.

The second part of the overall error is the discretization error, which refers to the error between the exact solution and the numerical solution of the penalized equation. As pointed out by Schneider et al. [18, 55], the discretization error is not only determined by the numerical scheme, but also limited by the regularity of the penalized solution. Regularity is characterized by the smoothness/continuity of the exact penalized solution u_η at the boundary of the penalized problem. The order of convergence in the discretization error becomes the minimum order between the numerical scheme and the regularity of the exact penalized solution. In the modified equation analysis, we are interested in the error of the numerical scheme. Details are given in the next section.

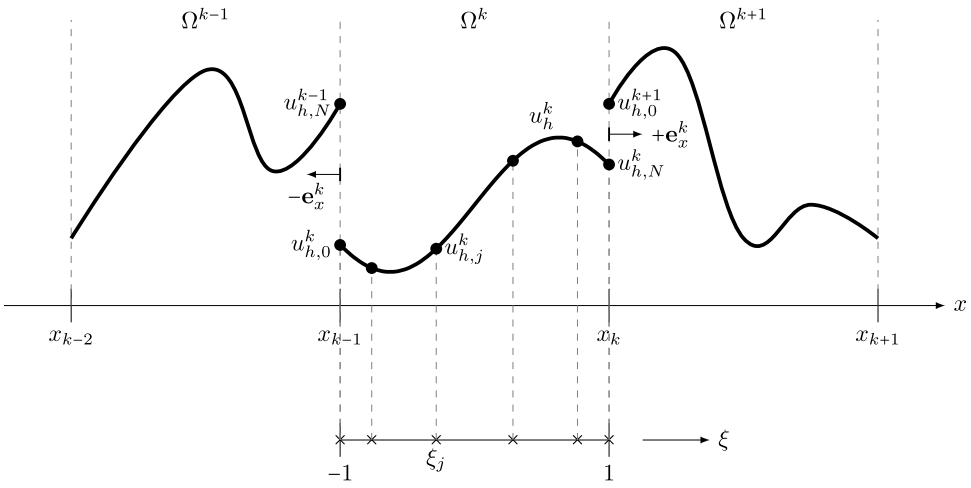

Fig. 2. Domain decomposition and reference interval in the DGSEM technique.

4. The modified equation analysis

When we discretize the penalized equation (3a) numerically, we translate it into a semi-discrete system (7a) for each element. This scheme is an approximation of our original equation. A different view is that the discrete system is the solution of modified differential equations but with some extra terms. This equation is named as the modified equation (or reduced PDE):

$$\frac{\partial u_h}{\partial t} + \mathcal{L}_h u + HOT = s(u_h^\star) \tag{12}$$

and is obtained by expanding the solutions in Eq. (7a) with a Taylor series around a point in the mesh $x(\xi_j)$. We omit the superscript k. HOT is the high-order term due to the Taylor series. $\mathcal{L}_h$ is the operator $\mathcal{L}$ applied at $x(\xi_j)$ and $s(u_h^\star)$ is a function of $u_h^\star$ that is the solution transported from the element Ω^k at the interfaces; see Fig. 2. The purpose of the modified equation is to obtain the local spatial truncation error, TE_j, which is defined as the difference between the original equation and the modified equation. The overall: $TE = \sum_j TE_j$. With a consistent discretization and a stable numerical scheme, the discretization error or the TE_j term writes as follows:

$$TE_j = C_0 h^p + C_1 h^{p+1} + C_2 h^{p+2} + C_3 h^{p+3} + \cdots = \mathcal{O}(h^p), \tag{13}$$

where h is a geometric discretization parameter representative of the grid spacing, p the order of accuracy of the numerical scheme, and C_0, C_1, C_2, … some constants that are independent of h and p. However, as discussed in the previous section, the discretization error is not only determined by the numerical scheme, but is also limited by the regularity of the penalized solution [18,55]. For high-order methods, with good regularity of the penalized solution u_η (see the previous section) at the interface, the high-order convergence property can be recovered, that is, $\mathcal{O}(h^{N+1})$. This has been shown in the recent work of Kou et al. [36]. Here, we further analyze the spatial discretization errors of the numerical scheme (the first source of the discretization error) to control and reduce errors and improve accuracy.

Note that finite-volume/difference methods are local by nature. In contrast, DGSEM is local within the whole domain but global within the element. Due to the non-local character of DG within each element, the solution depends on every point at the GL mesh. To perform the analysis, we center the solution at the same point of the source for each component of the discrete equation. Let us simplify the analysis to three GL points ($N = 2$), which are located at $\xi_0 = -1$, $\xi_1 = 0$ and $\xi_2 = 1$ with weights $w_0 = 1/3$, $w_1 = 4/3$ and $w_2 = 1/3$ respectively. The Lagrange polynomials are

$$l_0 = \frac{1}{2}\xi(\xi-1), \quad l_1 = -(\xi+1)(\xi-1), \quad l_2 = \frac{1}{2}\xi(\xi+1), \tag{14}$$

and the VP-DG matrix,

$$\begin{pmatrix} D^k_{00} & D^k_{10} & D^k_{20} \\ D^k_{01} & D^k_{11} & D^k_{21} \\ D^k_{02} & D^k_{12} & D^k_{22} \end{pmatrix} = \frac{1}{\eta_1}\begin{pmatrix} \chi^k_0 & 0 & 0 \\ 0 & \chi^k_1 & 0 \\ 0 & 0 & \chi^k_2 \end{pmatrix} - \frac{2}{\Delta x_k}\begin{pmatrix} -\frac{3}{2}\hat{c}^k_0 & -2\hat{c}^k_1 & \frac{1}{2}\hat{c}^k_2 \\ \frac{1}{2}\hat{c}^k_0 & 0 & -\frac{1}{2}\hat{c}^k_2 \\ -\frac{1}{2}\hat{c}^k_0 & 2\hat{c}^k_1 & \frac{3}{2}\hat{c}^k_2 \end{pmatrix} - \left(\frac{2}{\Delta x_k}\right)^2 \begin{pmatrix} \hat{v}^k_0 & 4\hat{v}^k_1 & \hat{v}^k_2 \\ -\frac{1}{2}\hat{v}^k_0 & -2\hat{v}^k_1 & -\frac{1}{2}\hat{v}^k_2 \\ \hat{v}^k_0 & 4\hat{v}^k_1 & \hat{v}^k_2 \end{pmatrix}. \tag{15}$$

coming from Eq. (7b). Due to the non-local character of DG within each element, the solution depends on every point at the GL mesh. To perform the analysis, we center the solution at the same point of the source for each component of the discrete equation. For example, the discrete equation for the $j = 0$ component is

$$\frac{\mathrm{d}u^k_{h,0}}{\mathrm{d}t} + D^k_{00}u^k_{h,0} + D^k_{10}u^k_{h,1} + D^k_{20}u^k_{h,2} = S^k_0. \tag{16a}$$

and the numerical source,

$$S^k_0 = \frac{2}{\Delta x_k}\left[3\mathcal{F}^k_{-1} - \frac{3}{\Delta x_k}\left(3\hat{v}^k_0\mathcal{U}^k_{-1} + \hat{v}^k_2\mathcal{U}^k_1\right)\right]. \tag{16b}$$

Expanding $u^k_{h,1}$ and $u^k_{h,2}$ around $u^k_{h,0}$, we have

$$u^k_{h,1} = u^k_{h,0} + \Delta\xi\left.\frac{\partial u^k_h}{\partial\xi}\right|_{\xi_0} + \frac{\Delta\xi^2}{2!}\left.\frac{\partial^2 u^k_h}{\partial\xi^2}\right|_{\xi_0} + \frac{\Delta\xi^3}{3!}\left.\frac{\partial^3 u^k_h}{\partial\xi^3}\right|_{\xi_0} + \cdots \tag{17}$$

$$u^k_{h,2} = u^k_{h,0} + 2\Delta\xi\left.\frac{\partial u^k_h}{\partial\xi}\right|_{\xi_0} + \frac{2^2\Delta\xi^2}{2!}\left.\frac{\partial^2 u^k_h}{\partial\xi^2}\right|_{\xi_0} + \frac{2^3\Delta\xi^3}{3!}\left.\frac{\partial^3 u^k_h}{\partial\xi^3}\right|_{\xi_0} + \cdots \tag{18}$$

being $\Delta\xi = \xi_1 - \xi_0 = \xi_2 - \xi_1 = 1$, we get

$$\frac{\partial u^k_{h,0}}{\partial t} + \left(D^k_{00} + D^k_{10} + D^k_{20}\right)u^k_{h,0} + \sum_{m=1}^{\infty}\left(D^k_{10} + 2^m\left(D^k_{20} + \frac{6}{\Delta x_k^2}\hat{v}^k_2\mathfrak{g}^k_2\right)\right)\frac{\Delta\xi^m}{m!}\left.\frac{\partial^m u^k_h}{\partial\xi^m}\right|_{\xi_0} = S^k_0, \tag{19}$$

where the numerical source now reads:

$$S^k_0 = \frac{2}{\Delta x_k}\left[\left(3\mathfrak{f}^{k-1}_2 - \frac{9}{\Delta x_k}\hat{v}^k_0\mathfrak{g}^{k-1}_2\right)u^{k-1}_{h,2} - \frac{3}{\Delta x_k}\hat{v}^k_2\mathfrak{g}^{k+1}_0 u^{k+1}_{h,0} + \left(3\mathfrak{f}^k_0 - \frac{3}{\Delta x_k}\left(3\hat{v}^k_0\mathfrak{g}^k_0 + \hat{v}^k_2\mathfrak{g}^k_2\right)\right)u^k_{h,0}\right]. \tag{20}$$

Neither $u^{k-1}_{h,2}$ nor $u^{k+1}_{h,0}$ can be expanded using Taylor series due to the discontinuous nature of DG. The terms in brackets of Eq. (19) simplify

Table 1
The reaction parameter and the coefficient $\mathcal{K}$ in the modified equations for a three-point GL grid.

j	ξ_j	$\widetilde{r}_j^k$	$\mathcal{K}_j^{(m)k}$
0	-1	$\frac{3\hat{c}_0^k+4\hat{c}_1^k-\hat{c}_2^k}{\Delta x_k}-4\frac{\hat{v}_0^k+4\hat{v}_1^k+\hat{v}_2^k}{\Delta x_k^2}$	$\frac{2^{2-m}\hat{c}_1^k-\hat{c}_2^k}{\Delta x_k^{1-m}}-4\frac{2^{2-m}\hat{v}_1^k+\hat{v}_2^k\left(1-{}^3\!/_2\mathfrak{g}_2^k\right)}{\Delta x_k^{2-m}}$
1	0	$-\frac{\hat{c}_0^k-\hat{c}_2^k}{\Delta x_k}+2\frac{\hat{v}_0^k+4\hat{v}_1^k+\hat{v}_2^k}{\Delta x_k^2}$	$-2^{-m}\frac{(-1)^m\hat{c}_0^k-\hat{c}_2^k}{\Delta x_k^{1-m}}+2^{1-m}\frac{(-1)^m\hat{v}_0^k\left(1+3\mathfrak{g}_0^k\right)+\hat{v}_2^k\left(1+3\mathfrak{g}_2^k\right)}{\Delta x_k^{2-m}}$
2	1	$\frac{\hat{c}_0^k-4\hat{c}_1^k-3\hat{c}_2^k}{\Delta x_k}-4\frac{\hat{v}_0^k+4\hat{v}_1^k+\hat{v}_2^k}{\Delta x_k^2}$	$(-1)^m\left(\frac{\hat{c}_0^k-2^{2-m}\hat{c}_1^k}{\Delta x_k^{1-m}}-4\frac{\hat{v}_0^k\left(1-{}^3\!/_2\mathfrak{g}_2^k\right)+2^{2-m}\hat{v}_1^k}{\Delta x_k^{2-m}}\right)$

to

$$D_{00}^k+D_{10}^k+D_{20}^k=\frac{\chi_0^k}{\eta_1}+\frac{3\hat{c}_0^k+4\hat{c}_1^k-\hat{c}_2^k}{\Delta x_k}-4\frac{\hat{v}_0^k+4\hat{v}_1^k+\hat{v}_2^k}{\Delta x_k^2}=:\frac{\chi_0^k}{\eta_1}+\widetilde{r}_0^k, \quad (21)$$

and

$$D_{10}^k+2^m\left(D_{20}^k+\frac{6}{\Delta x_k^2}\hat{v}_2^k\mathfrak{g}_2^k\right)=3\frac{2^{m+1}}{\Delta x_k^2}\hat{v}_2^k\mathfrak{g}_2^k+4\frac{\hat{c}_1^k-2^{m-2}\hat{c}_2^k}{\Delta x_k}-16\frac{\hat{v}_1^k+2^{m-2}\hat{v}_2^k}{\Delta x_k^2}=:\left(\frac{2}{\Delta x_k}\right)^m\mathcal{K}_0^{(m)k}. \quad (22)$$

For the Taylor term of first order, $\widetilde{c}_0^k=\mathcal{K}_0^{(1)k}$; and the second order term, $\widetilde{v}_0^k=-\mathcal{K}_0^{(2)k}/2$. Finally, we find the modified equation at the left-boundary element,

$$\frac{\partial u_{h,0}^k}{\partial t}+\Delta\xi\,\widetilde{c}_0^k\frac{2}{\Delta x_k}\left.\frac{\partial u_h^k}{\partial\xi}\right|_{\xi_0}-\Delta\xi^2\,\widetilde{v}_0^k\left(\frac{2}{\Delta x_k}\right)^2\left.\frac{\partial^2 u_h^k}{\partial\xi^2}\right|_{\xi_0}+\frac{\chi_0^k}{\eta_1}u_{h,0}^k+HOT_0^k=s_{\mathrm{DG},0}^k, \quad (23a)$$

where

$$s_{\mathrm{DG},0}^k=S_0^k-\widetilde{r}_0^k u_{h,0}^k, \quad (23b)$$

and

$$HOT_0^k=\sum_{m=3}^{\infty}\left(\frac{2}{\Delta x_k}\right)^m\mathcal{K}_0^{(m)k}\frac{\Delta\xi^m}{m!}\left.\frac{\partial^m u_h^k}{\partial\xi^m}\right|_{\xi_0}. \quad (23c)$$

Additionally, the original PDE at the left-boundary element is

$$\frac{\partial u_{h,0}^k}{\partial t}+\hat{c}_0^k\frac{2}{\Delta x_k}\left.\frac{\partial u_h^k}{\partial\xi}\right|_{\xi_0}-\hat{v}_0^k\left(\frac{2}{\Delta x_k}\right)^2\left.\frac{\partial^2 u_h^k}{\partial\xi^2}\right|_{\xi_0}+\frac{\chi_0^k}{\eta_1}u_{h,0}^k=0. \quad (23d)$$

and, therefore, the truncation error at the left-boundary element becomes:

$$TE_0^k=s_{\mathrm{DG},0}^k+\left(\hat{c}_0^k-\Delta\xi\,\widetilde{c}_0^k\right)\frac{2}{\Delta x_k}\left.\frac{\partial u_h^k}{\partial\xi}\right|_{\xi_0}-\left(\hat{v}_0^k-\Delta\xi^2\,\widetilde{v}_0^k\right)\left(\frac{2}{\Delta x_k}\right)^2\left.\frac{\partial^2 u_h^k}{\partial\xi^2}\right|_{\xi_0}-HOT_0^k. \quad (23e)$$

We can proceed in a similar manner to obtain the modified equations and truncation errors for the inner point, $j=1$, and the right-boundary point, $j=2$. For $j=1$, $u_{h,0}^k$ and $u_{h,2}^k$ are centered on $u_{h,1}^k$; for $j=2$, $u_{h,0}^k$ and $u_{h,1}^k$ are centered on $u_{h,2}^k$. Their formulae can be written using Eqs. (23), but with differences in the reactive parameter, $\widetilde{r}_j^k$, the coefficient $\mathcal{K}$ and the numerical source, S_j^k, for $j=0,1,2$, see Tables 1 and 2. The source of the DG, $s_{\mathrm{DG},j}^k$, arises from the discontinuous nature of the DG approach (discontinuous boundary values) and the selected diffusive scheme.

Now suppose that an element, Ω^k, belongs to a solid region, Ω_s, then $\chi_j^k=1$ for $j=0,1,2$, and the truncation error still remains inside the solid. If we want to eliminate all the error terms in TE_j^k for $j=0,1,2$, we need to solve the system:

$$\begin{cases} s_{\mathrm{DG},j}^k=0,\\ \hat{c}_j^k-\Delta\xi\,\widetilde{c}_j^k=0,\\ \hat{v}_j^k-\Delta\xi\,\widetilde{v}_j^k=0,\\ \mathcal{K}_j^{(m)k}=0, \end{cases} \quad (24)$$

for $j=0,1,2$ and $m\geq 3$. However, the problem is given by Eq. (24) has an infinite number of equations and a finite number of unknowns. In total, there are 10 unknowns, which are the 4 weights $\mathfrak{f}$ and $\mathfrak{g}$, $\hat{c}_0^k=\hat{c}_1^k=\hat{c}_2^k=c+1/\eta_2=:\hat{c}$ and $\hat{v}_0^k=\hat{v}_1^k=\hat{v}_2^k=\nu-1/\eta_3=:\hat{v}$. The solution to the system is as follows:

$$\begin{cases} \eta_2=-1/c, \quad \eta_3=1/\nu\\ \mathfrak{f}_2^{k-1}=\mathfrak{f}_0^k=\mathfrak{f}_2^k=\mathfrak{f}_0^{k+1}=0\\ \text{for all } \mathfrak{g}_2^{k-1},\mathfrak{g}_0^k,\mathfrak{g}_2^k \text{ and } \mathfrak{g}_0^{k+1} \end{cases} \quad (25a)$$

This solution will be referred to as the trivial solution of the problem. At this point, one may wonder if there is any other set, a nontrivial family, that cleans up almost all the errors within the solid region. To investigate this, a determined system should be formed. Ideal errors remaining within the solid region should be:

$$TE_j^k\sim HOT_j^k\sim\mathcal{O}(\Delta x_k^m), \quad j=0,1,2, \quad (26)$$

for $m\geq 3$ as a representation of high order. However, the investigation of non-trivial solutions did not meet the previous requirement; see more details in Appendix B. Table 3 summarizes both the trivial and non-trivial solutions that have been found.

The trivial solution is the condition for DGSEM to compensate (or kill) spatial truncation errors within the body region, but additional insight can be obtained. If we substitute the values for η_2 and η_3 in our penalized equation and isolate χ, we get the following equation:

$$\frac{\partial u}{\partial t}+\underbrace{\frac{\partial}{\partial x}\left[(1-\chi)\left(cu-\nu\frac{\partial u}{\partial x}\right)\right]}_{\text{Physical term}}+\frac{\chi}{\eta_1}u=0. \quad (27)$$

If we are in the solid region, $\chi=1$, then the physical contribution of the PDE is removed, so this region is modeled with only the reaction penalization term, and therefore only time integration methods will lead to errors within the solid. This result agrees with the use of a typical characteristic-based volume penalization approach [25,27], where the RHS term vanishes to smooth out the errors in the solid region but without a theoretical explanation, which is provided here. Cancellation of particular terms can reduce the error inside the solid, thus improving the accuracy in the fluid region. To find the overall accuracy in both the solid and the fluid regions, the present results can be coupled with the classic modified equation analysis for the fluid region [48]. The local error in both fluid and solid elements is coupled across elements via the numerical fluxes. This topic is currently out of scope and is worth investigating in future works.

5. Numerical results

In this section we introduce two numerical experiments to evaluate and validate the trivial solution derived from the modified equation

Table 2
Numerical source for the DG source, $s^k_{\mathrm{DG},j} = S^k_j - \widetilde{r}^k_j u^k_{h,j}$.

j	S^k_j
0	$\frac{2}{\Delta x_k}\left[\left(3\mathfrak{f}_2^{k-1} - \frac{9}{\Delta x_k}\widehat{\nu}_0^k \mathfrak{g}_2^{k-1}\right) u_{h,2}^{k-1} - \frac{3}{\Delta x_k}\widehat{\nu}_2^k \mathfrak{g}_0^{k+1} u_{h,0}^{k+1} + \left(3\mathfrak{f}_0^k - \frac{3}{\Delta x_k}\left(3\widehat{\nu}_0^k \mathfrak{g}_0^k + \widehat{\nu}_2^k \mathfrak{g}_2^k\right)\right) u_{h,0}^k\right]$
1	$\frac{6}{\Delta x_k^2}\left[\widehat{\nu}_0^k \mathfrak{g}_2^{k-1} u_{h,2}^{k-1} + \widehat{\nu}_2^k \mathfrak{g}_0^{k+1} u_{h,0}^{k+1} + \left(\widehat{\nu}_0^k \mathfrak{g}_0^k + \widehat{\nu}_2^k \mathfrak{g}_2^k\right) u_{h,1}^k\right]$
2	$-\frac{2}{\Delta x_k}\left[\frac{3}{\Delta x_k}\widehat{\nu}_0^k \mathfrak{g}_2^{k-1} u_{h,2}^{k-1} + \left(3\mathfrak{f}_0^{k+1} + \frac{9}{\Delta x_k}\widehat{\nu}_2^k \mathfrak{g}_0^{k+1}\right) u_{h,0}^{k+1} + \left(3\mathfrak{f}_2^k + \frac{3}{\Delta x_k}\left(3\widehat{\nu}_2^k \mathfrak{g}_2^k + \widehat{\nu}_0^k \mathfrak{g}_0^k\right)\right) u_{h,2}^k\right]$

Table 3
Summary of family of solutions for VP-IBM DGSEM, the trivial solution is the last row. $*$ means equivalent to a continuous Galerkin (CG) method.

$\widehat{c} = c + \frac{1}{\eta_2}$	$\widehat{\nu} = \nu - \frac{1}{\eta_3}$	$\mathfrak{f}$s	$\mathfrak{g}$s	$\equiv$ CG $*$	TE_j^k
η_2 free	η_3 free	free	$\mathfrak{g}_2^k = \mathfrak{g}_0^k = 2$ $\mathfrak{g}_2^{k-1} = \mathfrak{g}_0^{k+1} = 0$		$\mathcal{O}\left(\Delta x_k^0\right)$
η_2 free	η_3 free	$\mathfrak{f}_2^{k-1} = -\mathfrak{f}_2^k = \widehat{c} + \frac{4}{\Delta x_k}\widehat{\nu}$ $\mathfrak{f}_0^k = \mathfrak{f}_0^{k+1} = 0$	$\mathfrak{g}_2^k = \mathfrak{g}_0^k = 2$ $\mathfrak{g}_2^{k-1} = \mathfrak{g}_0^{k+1} = 0$	✓	$\mathcal{O}\left(\Delta x_k^0\right)$
$\widehat{c} + \frac{4}{\Delta x_k}\widehat{\nu} = 0$		0	$\mathfrak{g}_2^k = \mathfrak{g}_0^k = 2$ $\mathfrak{g}_2^{k-1} = \mathfrak{g}_0^{k+1} = 0$		$\mathcal{O}\left(\Delta x_k^0\right)$
η_2 free	0	$\mathfrak{f}_0^k = -\mathfrak{f}_2^k = \widehat{c}$ $\mathfrak{f}_0^k = \mathfrak{f}_0^{k+1} = 0$	free	✓	$\mathcal{O}\left(\Delta x_k^2\right)$
η_2 free	0	free	free		$\mathcal{O}\left(\Delta x_k^0\right)$ Boundary $\mathcal{O}\left(\Delta x_k^2\right)$ Inner
0 ($\eta_2 = -1/c$)	0 ($\eta_3 = 1/\nu$)	0	free		$\mathcal{O}\left(\Delta x_k^\infty\right)$

analysis. The first group of cases is the one-dimensional advection–diffusion equation, where the influence of penalization parameters is studied in detail. The optimal parameters obtained from the numerical experiments and the analysis of the modified equations are then applied to the two-dimensional advection–diffusion equation.

5.1. One-dimensional advection–diffusion equation

We start from the one-dimensional advection equation, where a no-slip wall is placed in the middle of the computational domain. This problem has been formulated in previous works [51,52], which is illustrated in Fig. 3. Periodic boundary conditions are imposed on both sides of the computational domain, while a sinusoidal wave with a given wavenumber is considered as the initial condition. The advection speed is set to $c = 1$ and the computational domain is defined in $x \in [-1, 1]$, discretized by K equispaced elements with mesh size Δx. The solution points are selected according to the Gauss–Lobatto quadrature rule, which is consistent with the previous analysis. An upwind flux for the advection term is selected. The solid region is defined as a no-slip wall, i.e., $u_s = 0$. It lies in the middle of the computational domain and starts from $x = 0$, whose width is defined as Δ_s, leading to the solid region $0 \le x \le \Delta_s$.

For consistency with the analysis of the modified equations, we consider $\Delta_s = \Delta x$, which means that the solid boundaries lie exactly at the interface between the elements (if we have an even number of elements). This allows us to define the solid ratio $r = 1/K$ as the ratio between the solid region and the computational domain. As shown in Fig. 3, the initial wavelike solution passes through the no-slip wall in the middle, and the damped solution moves to the right as time evolves. Since periodic boundary conditions are considered, the solution will eventually become 0. If the no-slip wall boundary condition is exactly imposed, the solutions coming out of the wall will be zero. However, in practice, the classic volume penalization

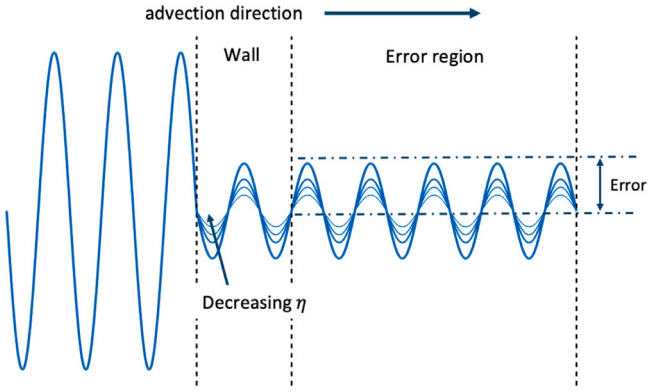

Fig. 3. Schematic illustration of the advection problem with IBM.

(where only the solution is penalized) is unable to cancel out all waves, when η_1 approaches zero, the modeling error still exists [51,52]. The transported solution coming out of the wall (in the initial transient state so that the solution passes through the wall only once) depends on the damping provided by the volume penalization approach, where a smaller η_1 makes the solution closer to zero. Therefore, the accuracy of the IBM imposition can be evaluated by comparing the exact solution (zero) and the damped solution in both the flow and solid regions (e.g., within a short advection time $0 < x < t$).

We first perform the numerical experiment of a linear advection equation with a wavelike initial condition. The initial condition is defined as a sinusoidal wave with wavenumber ω, which is nondimensionalized by the mesh size Δx and the polynomial order N, defined as $\omega \Delta x/(N+1)$. Furthermore, due to the existence of a solid wall, the actual fluid domain is shorter than the entire computational domain;

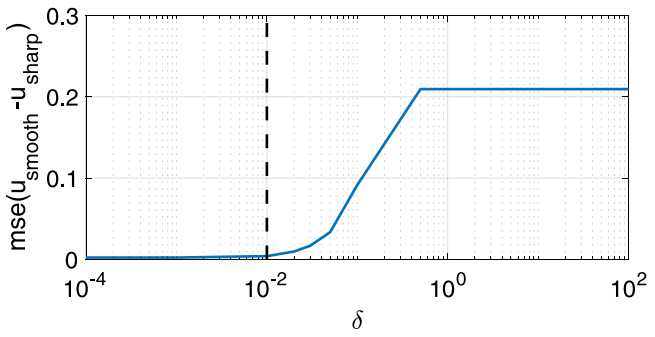

Fig. 4. Mean squared error in the flow, between sharp and smooth mask function with increasing mask width δ.

therefore, the effective wavenumber in the fluid region is greater than ω. This effective wavenumber is rescaled by the solid ratio r, defined as $\overline{\omega} = \omega/(1-r)$ [51]. We consider a spatial discretization with $K = 40$ elements in the computational domain ($\Delta x = 0.05$). Based on this mesh, we set $\Delta_s = \Delta x$ with $r = 1/40$ and choose $N = 3$ as a representative order for high-order methods. The initial condition with wavenumber $\overline{\omega}\Delta x/(N+1) = 0.3223$ is considered, which lies in the resolved wavenumber region of the scheme. The time integration is based on the third-order Runge–Kutta scheme. To reduce the temporal error, a sufficiently small time step is set to $\Delta t = 10^{-5}$. The final time is set to 1.1 to obtain a sufficiently penalized solution in the right region of the computational domain.

Different combinations of parameters (with and without the first-order term) are considered. To evaluate accuracy, the error (in the flow) is defined as the error in $x \in [\Delta_s, 1]$ and the penalized value $u_s = 0$. Defining the number of solution points inside the flow domain of interest as $N_p = (N+1)K$, we have the L_2-norm of the error as

$$\text{error} = \sqrt{\frac{1}{N_p}\sum_{i=1}^{N_p}[u(x_i) - u^{\text{exact}}(x_i)]^2}\ ,\ x_i \in [\Delta_s, 1],\ u^{\text{exact}} = 0, \tag{28}$$

and the L_2-norm of the error in the solid is defined as

$$\text{error}_{\text{solid}} = \sqrt{\frac{1}{N_p}\sum_{i=1}^{N_p}[u(x_i) - u^{\text{exact}}(x_i)]^2}\ ,\ x_i \in [0, \Delta_s],\ u^{\text{exact}} = 0, \tag{29}$$

First, a numerical study is performed to justify the equivalence of using sharp or smooth mask functions (given the small width δ for the smooth mask function to reduce the modeling error). The mask function in Eq. (2b) is a smooth mask function, while a sharp mask function is used in the classic volume penalization [15,18,30,55]:

$$\chi(x,t) = \begin{cases} 1, & \text{if } x \in \Omega_s \\ 0, & \text{Otherwise} \end{cases}.$$

We run the simulation at different widths δ with the penalization parameter $\eta_1 = 10^{-3}$, until the final time 1.1, and compare the mean squared error in the flow between the results from the sharp and the smooth mask function. This error is compared in Fig. 4, where the results based on the sharp or smooth mask are almost identical at small δ, and the difference becomes dominant when δ is sufficiently large, $\delta > 0.01$. Similar results are obtained when $\delta < 0.01$, which is sufficient to guarantee the equivalence of the analysis for smooth and sharp masks. As δ further increases, the mask becomes too smooth and the penalized region occupies the flow, resulting in additional modeling errors due to the wrong representation of the interface. Therefore, since the equivalence of sharp and smooth mask functions exists for a small range of δ, in the numerical tests, a sharp mask function for the classic volume penalization [15,18,30,55] is used.

A comparison of the solution at the final time is shown in Fig. 5. Four cases are tested, where the first three cases contain only the volume penalization term for the solution, while the first-order penalization term with $\eta_2 = -1/c$ is added to the last case. The figure shows that as the penalization parameter η_1 decreases, the solution approaches zero, indicating that the boundary condition is imposed more accurately. Note that in the last case, a large penalization parameter (i.e. weaker penalization) $\eta_1 = 10^{-3}$ is used. In addition, when the first-order term is added, improved accuracy is seen as the solution is closer to zero. The errors in the fluid region of the four cases are $3.071 \cdot 10^{-2}$, $5.385 \cdot 10^{-3}$, $5.698 \cdot 10^{-4}$, and $1.022 \cdot 10^{-4}$, respectively. This indicates that by introducing the first-order term with a proper selection of the penalization parameter, it is possible to improve the accuracy.

Furthermore, to study the effect of η_2, we run additional simulations for a range of η_2, and show the errors in Fig. 6. The errors in the flow and solid regions for $\eta_1 = 10^{-3}$, $\eta_1 = 10^{-4}$, and $\eta_1 = 10^{-5}$ are shown in Figs. 6a, 6b, 6c and 6d, respectively. For consistency with the analysis of modified equations, the first group of cases is performed in polynomial order $N = 2$, and the second group of cases is performed in polynomial order $N = 3$. Improved accuracy is seen when the penalization parameter is decreased. In addition, there exists an optimal η_2 that leads to minimal errors in both the flow and solid regions, which is the same for all penalization parameters. This optimal value is $\eta_2 = -1/c$, indicating that inside the solid the first-order penalization term becomes $-\partial u/\partial x$ thus the physical advection is canceled out. From Fig. 6b and 6d, this cancellation will lead to almost zero error inside the solid, indicating that the boundary condition is satisfied exactly. At a larger η_1, this optimal value remains valid, but the optimal error increases, as shown in Fig. 6c and 6e. Therefore, to reach the optimal accuracy, we need to use a small penalization parameter η_1, in combination with the optimal η_2. These findings are consistent with the theory that the modeling error of volume penalization converges with $\eta_1 \to 0$. Furthermore, the conclusion of the modified equation analysis is also validated, since choosing $\eta_2 = -1/c$ leads to improved accuracy and almost satisfies the boundary conditions exactly. In addition, numerical tests on the same problem, with non-body-fitted grid are given in Appendix C, where the same conclusions as this example can be drawn.

To investigate the effect of the viscous term, the advection–diffusion equation is investigated. Since the optimal η_2 in the advection equation has been obtained, $\eta_2 = -1/c$ is selected for all cases. We proceed as for the advection equation, by setting $K = 40$ elements, $\Delta_s = \Delta x$, $r = 1/40$ and $N = 3$. The initial condition with wavenumber $\overline{\omega}\Delta x/(N+1) = 0.3223$ is used and marched in time to $t = 1.5$. Taking into account the effect of diffusion, the error in the flow region is limited to $x \in [\Delta_s, 0.7]$.

For the discretization of the viscous flux, either the BR1 or the LDG scheme is considered. The results for two physical viscosities $\nu = 0.001$ and $\nu = 0.01$ are shown in Figs. 7 and Figs. 8, respectively. For both cases, it is observed that the optimal second-order coefficient η_3 exists and can lead to a minimal error within the solid (as shown in Fig. 7b and 7d and Fig. 8b and 8d). This optimal value shows the relationship $1/\eta_3 = \nu$, which also indicates the cancelation of the viscous term inside the solid. This agrees with the optimal η_3 derived from the modified equation analysis. However, when looking at the error inside the flow, the optimal second-order penalization term does not lead to the lowest error when the BR1 scheme is used. This highlights the importance of choosing appropriate Riemann solvers to maintain good accuracy in the flow region. When the LDG scheme is selected, the optimal η_3 will reach the lowest error in the flow region, indicating that this flux is more suitable for the present problem. Therefore, when handling the viscous term, the LDG scheme is preferred, which gives consistent results of errors in the solid and in the fluid. In summary, the one-dimensional test case shows that the optimal penalization parameters derived from the modified equation analysis achieve minimal numerical errors in imposing the boundary conditions.

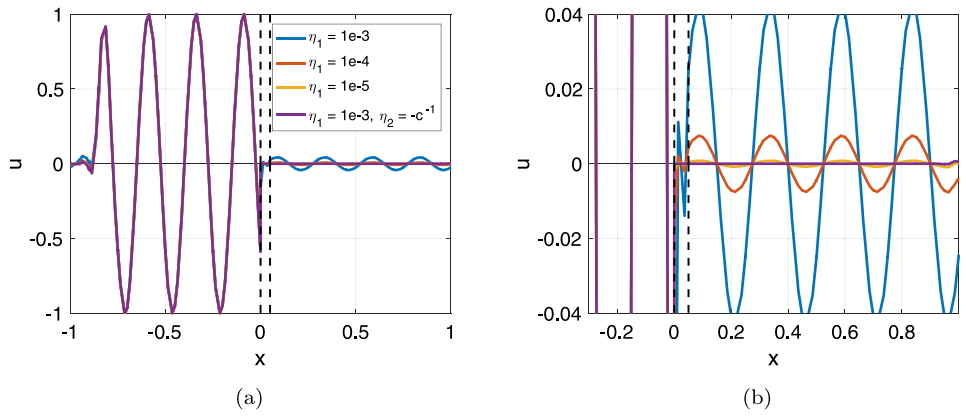

Fig. 5. Simulation under different penalization parameters ($r = 1/40$, $N = 3$, initial wavenumber $\overline{\omega}\Delta x/(N+1) = 0.3223$, $K = 40$) at $t = 1.1$: (a) Global view; (b) Enlarged view.

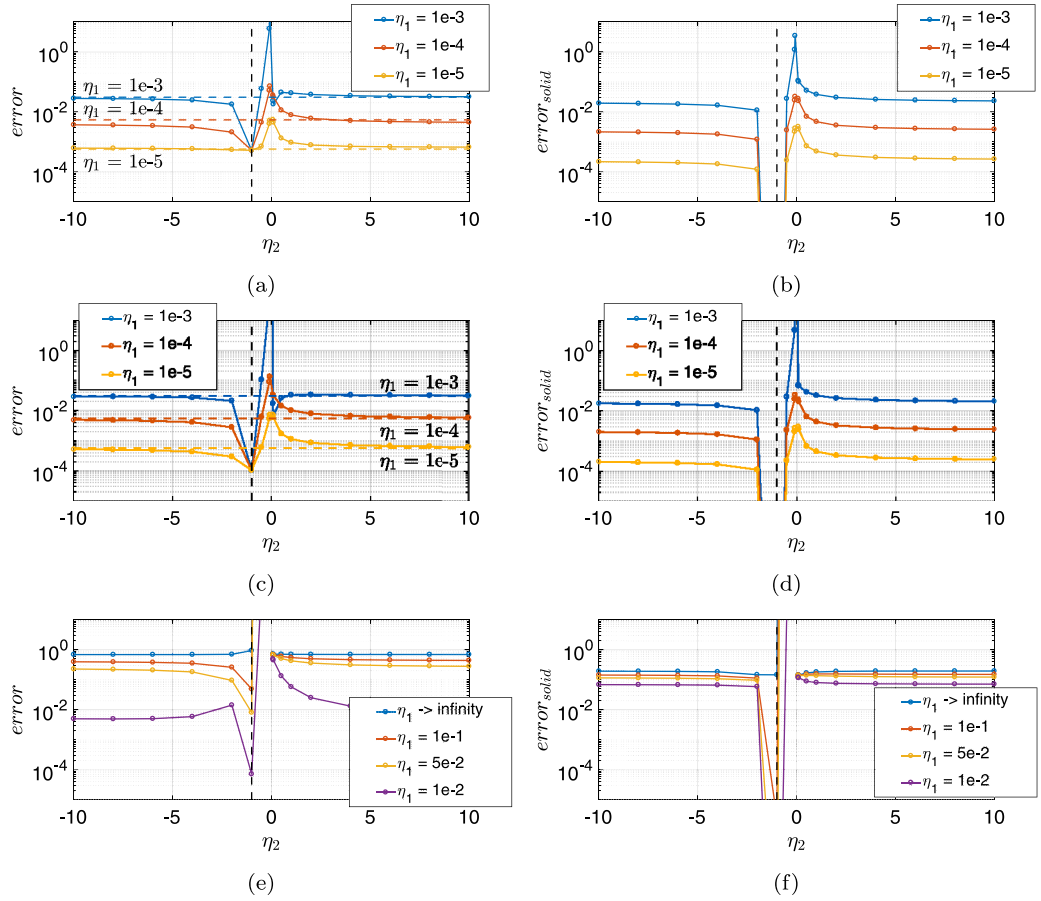

Fig. 6. Error comparison for the advection equation, vertical dashed line refers to $\eta_2 = -1/c$, and horizontal dashed line refers to $\eta_2 \to \infty$. (a) Error in the flow ($N = 2$). (b) Error in the solid, the optimal solution is zero ($N = 2$). (c) Error in the flow ($N = 3$). (d) Error in the solid, the optimal solution is zero ($N = 3$). (e) Error in the flow (larger penalization parameter, $N = 3$). (f) Error in the solid (larger penalization parameter, $N = 3$).

5.2. Two-dimensional advection–diffusion equation

In this section, a numerical experiment is performed for the two-dimensional advection–diffusion equation, using the conclusions of the modified equation analysis. The one-dimensional test case in the previous section is extended to two space directions. Again, periodic boundary conditions are imposed on both sides of the computational domain, while a sinusoidal wave with a given wavenumber is considered as the initial condition. Therefore, the optimal parameters derived from one-dimensional test cases are then dependent on each space direction. Extensions for GL points can be found in [62]. The governing equation is (the solid region is the no-slip wall):

$$\frac{\partial u}{\partial t} + \nabla \cdot (\mathbf{f}_{\text{adv}} + \mathbf{f}_{\text{diff}}) + \frac{\chi}{\eta_1} u + \nabla \cdot (\mathbf{g}\chi u) + \nabla \cdot (\mathbf{H}\nabla(\chi u)) = 0, \tag{30}$$

where the advection flux is $\mathbf{f}_{\text{adv}} = (c_x u, c_y u)^T$, the diffusion flux is $\mathbf{f}_{\text{diff}} = (-\nu_x \partial u/\partial x, -\nu_y \partial u/\partial y)^T$, $\mathbf{g} = (1/\eta_{2,x}, 1/\eta_{2,y})^T$, and $\mathbf{H} = \text{diag}\,(1/\eta_{3,x}, 1/\eta_{3,y})$. The first-order and second-order penalization parameters in each direction is denoted by the second subscript. Here we set $c_x = c_y = 1$ and $\nu_x = \nu_y = 0.001$, therefore, the optimal parameters satisfy $\eta_{2,x} = \eta_{2,y}$ and $\eta_{3,x} = \eta_{3,y}$. Note that, for the present linear equation, the extension to different advection velocities and viscosities

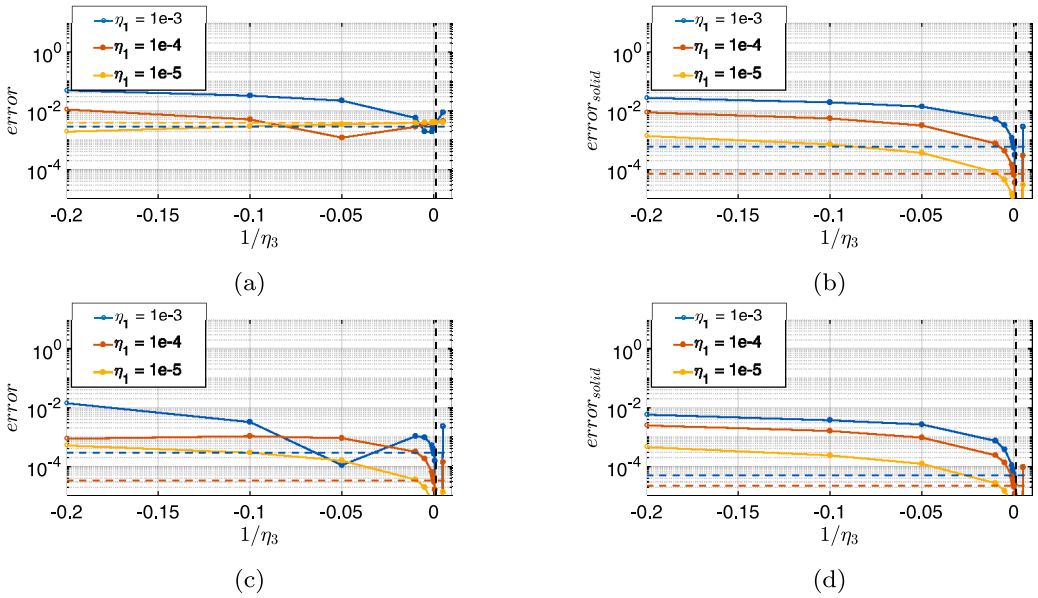

Fig. 7. Error comparison for the advection–diffusion equation ($\nu = 0.001$), vertical dashed line refers to $1/\eta_3 = \nu = 0.001$, horizontal dashed line refers to $\eta_3 \to \infty$ (without second-order term) (a) Error in the flow ($\eta_2 = -1/c$), BR1. (b) Error in the solid ($\eta_2 = -1/c$), BR1. (c) Error in the flow ($\eta_2 = -1/c$), LDG. (d) Error in the solid ($\eta_2 = -1/c$), LDG.

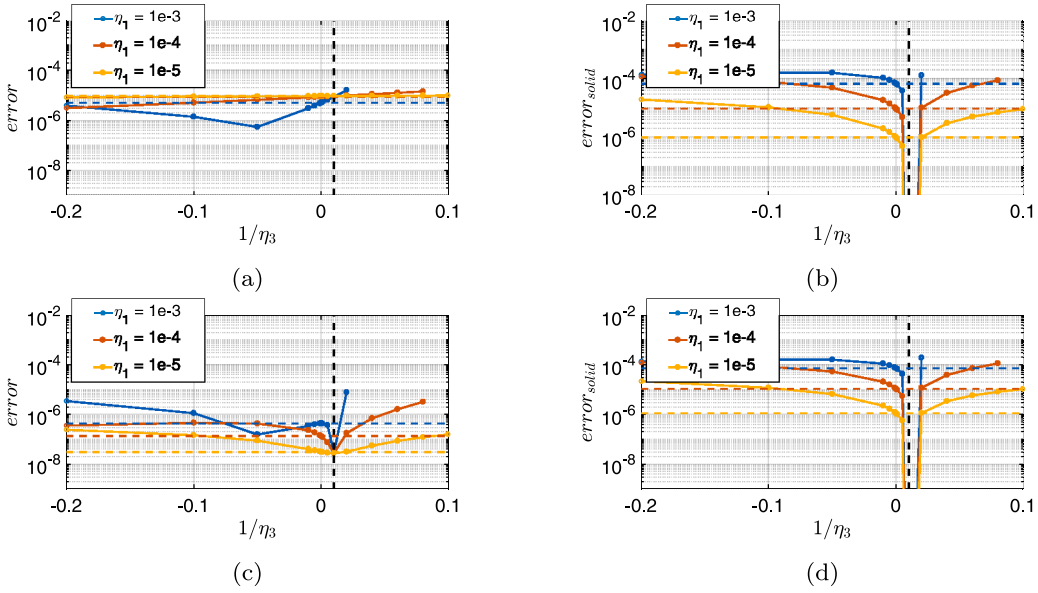

Fig. 8. Error comparison for the advection–diffusion equation ($\nu = 0.01$), vertical dashed line refers to $1/\eta_3 = \nu = 0.01$, horizontal dashed line refers to $\eta_3 \to \infty$ (without second-order term). (a) Error in the flow ($\eta_2 = -1/c$), BR1. (b) Error in the solid ($\eta_2 = -1/c$), BR1. (c) Error in the flow ($\eta_2 = -1/c$), LDG. (d) Error in the solid ($\eta_2 = -1/c$), LDG.

in each direction is straightforward, while the optimal penalization parameter (i.e., trivial solution from modified equation analysis) also varies in different directions. As in the one-dimensional test case, the solid wall is considered in the middle of the computational domain. A schematic illustration of the two-dimensional problem for the present study is shown in Fig. 9. The solid no-slip region has an L shape, which is centered in the middle of the square domain, making the top right region amplified by the wall. If the advection direction is set appropriately, the initial wave will move towards the wall. After that, we can solve the equation until all the solutions in the top right region have been penalized (which are then expected to be zero), and compute the error in this region. The error is again the difference between the numerical and exact solution (here set to zero).

We consider a square computational domain in $x \in [-0.1, 0.1]$ and $y \in [-0.1, 0.1]$, with periodic boundary conditions. The domain is discretized into 20 equispaced elements in both the x and the y directions, resulting in 400 square elements in total and uniform mesh size $\Delta x = \Delta y = 0.01$. The penalization parameter and the explicit time step is set to $\eta_1 = \Delta t = 10^{-4}$. The polynomial order $N = 3$ is selected. Due to the preset flow advection parameters, the advection moves towards the top right direction. The width of the solid region is set to the size of a uniform grid $\Delta_s = \Delta x$, resulting in the solid ratio $r = 1/20$. We use the wavelike initial condition $u(x, y) = \sin(\omega x + \omega y)$, where a nondimensional wavelength $\overline{\omega}\Delta x/(N+1) = 0.3307$ is selected. Again, like the one-dimensional test case, we are only interested in the initial transient state (i.e., $t \approx 0.1$) where the solution is damped by

Table 4
Error comparison (error region in Fig. 9) of the two-dimensional advection–diffusion equation with IBM wall under different diffusive flux schemes and different combinations of penalization parameters.

Diffusive flux scheme	$\eta_1 = 10^{-4}$	$\eta_1 = 10^{-4}$ $\eta_2 = -1$	$\eta_1 = 10^{-4}$ $\eta_2 = -1$ $\eta_3 = 10^3$
BR1	1.6610×10^{-4}	1.3513×10^{-4}	1.5874×10^{-4}
LDG	6.4091×10^{-5}	7.1993×10^{-6}	2.2669×10^{-7}

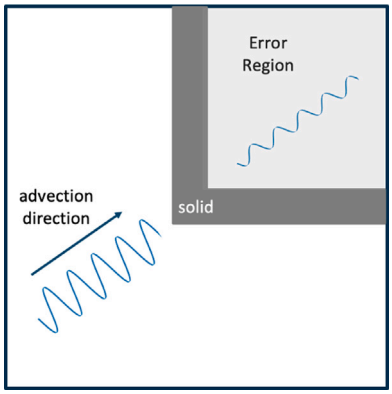

Fig. 9. Schematic illustration of the advection problem with IBM.

the solid only once. We check the accuracy by comparing the solution inside the error region (after it all gets damped by the wall) against the expected solution in the solid (e.g., zero in the present example).

The first simulation for pure advection problem is performed when only the first penalization term (η_1 for u) is included. Fig. 10 shows three typical solution fields at different times. As shown in the figure, the penalized solution will move towards the top right corner, and finally dominate the entire domain due to the periodic boundary conditions. To compare the accuracy of simulation, the final simulation time is set to 0.11. Two solution fields, without and with the optimal first-order penalization term, are shown in Fig. 11, where the values of η_2, x and η_2, y are set to -1 to match the physical advection speed. The improved accuracy from adding the optimal first-order term is seen in both the solution field and in the error. The error in the fluid region has been greatly reduced from 0.0207 to $1.4616 \cdot 10^{-5}$. To test the proposed analysis, a more challenging test case is included in Appendix D, where different velocities and diffusivities are considered in each direction. In this case, when penalizing with optimal parameters also leads to minimal error. This numerical experiment extends and validates the conclusions obtained from the modified equation analysis, where the optimal first-order penalization term cancels the advection term and leads to improved accuracy.

Additional numerical experiments are performed for the advection–diffusion equation. The space and time discretizations remain the same as in the advection case. The final time is set to $t = 0.15$. Three strategies are considered: (1) only volume penalization for the no-slip wall boundary condition, (2) volume penalization for both the value and the first-order term, and (3) volume penalization for all terms. Two types of viscous fluxes with either the BR1 or the LDG scheme are considered. The errors inside the fluid region are compared in Table 4, where conclusions similar to one-dimensional advection can be drawn. When the BR1 scheme is used, adding additional first-order and second-order penalization terms improves the overall accuracy, compared with the standard case (the first strategy). However, the addition of a second-order term does not lead to improved accuracy in the flow region. When the LDG scheme is used, adding first- and second-order terms will lead to a greater reduction of the error. This is consistent with the observations for the one-dimensional advection equation, where the LDG scheme is shown to provide more accurate results than the BR1 scheme. This numerical experiment validates the proposed modified equation analysis for the second-order derivative in two-dimensional linear equations.

6. Conclusions

This study contributes to a better understanding of the numerical errors for Immersed Boundary Methods based on volume penalization, in combination with a high-order nodal discontinuous Galerkin scheme. For this purpose, an analysis of the modified equation is provided.

The modified equation is a useful tool to analyze dissipative/dispersive errors related to the numerical discretizations. In this paper, we focus on the spatial errors introduced by the Immersed Boundary Method. Nodal solutions are expanded as Taylor series, and by rearranging the pseudo-differential equation new terms arise. These terms allow us to obtain insight into the dissipative/dispersive characteristics of the errors and guidelines for their minimization. For example, the inclusion of extra penalization terms of the first and second derivatives, in addition to the classic penalization of the variable, is considered. Through this analysis, we provide optimal values for the first- and second-order penalization parameters to cancel the advection/diffusive errors inside the solid, which in turn lead to improved errors in the flow.

Numerical experiments validate the theoretical findings obtained from the analysis of modified equations, where optimal penalization parameters can lead to minimal errors (with a sufficiently small penalization parameter η_1). When combined with an appropriate numerical scheme (here, Local discontinuous Galerkin for viscous terms), minimal errors in the flow region are reached.

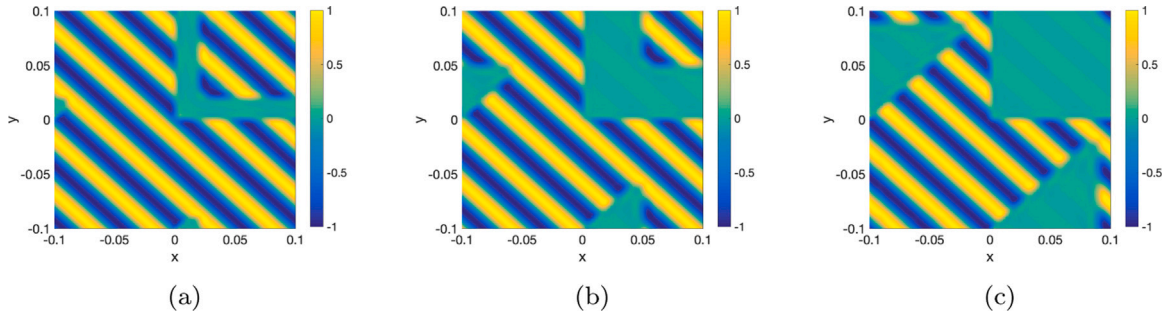

Fig. 10. Simulation under different parameters ($r = 1/20$, $N = 3$, initial wavenumber $\overline{\omega}\Delta x/(N+1) = 0.3307$, $K = 20$). (a) t = 0.01. (b) t = 0.04. (c) t = 0.08.

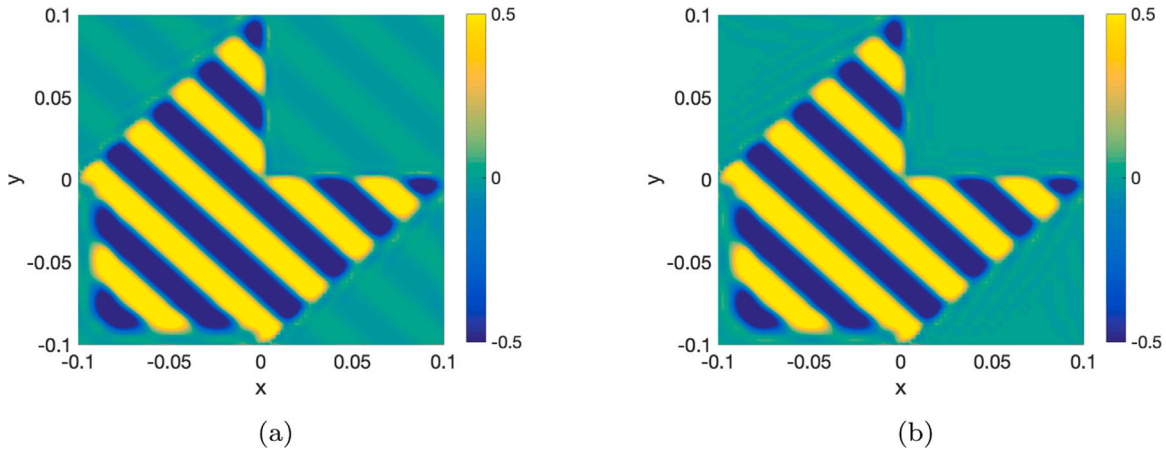

Fig. 11. Simulation under different parameters ($r = 1/20$, $N = 3$, initial wavenumber $\overline{\omega}\Delta x/(N+1) = 0.3307$, $K = 20$). The difference lies in the upper right flow region. (a) error = 0.0207, $\text{error}_{\text{solid}} = 0.0552$. (b) error = $1.4616 \cdot 10^{-5}$, $\text{error}_{\text{solid}} = 0$.

Future work will extend these findings to systems of partial differential equations with non-linearities, and extend the theoretical analysis to multi-dimensional systems.

CRediT authorship contribution statement

Victor J. Llorente: Conceptualization, Investigation, Methodology, Software, Writing – original draft, Writing – review & editing. **Jiaqing Kou:** Conceptualization, Investigation, Methodology, Software, Writing – original draft, Writing – review & editing. **Eusebio Valero:** Conceptualization, Investigation, Methodology, Funding acquisition, Project administration, Supervision. **Esteban Ferrer:** Conceptualization, Investigation, Methodology, Software, Writing – original draft, Writing – review & editing, Funding acquisition, Project administration, Supervision.

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.

Data availability

Data will be made available on request.

Acknowledgments

JK and EF acknowledge the financial support of the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement (MSCA ITN-EID-GA ASIMIA No. 813605). VJL, EF, and EV acknowledge financial support from the European High-Performance Computing Joint Undertaking (JU) under grant agreement (No. 956104). The JU receives support from the European Union's Horizon 2020 research and innovation programme under grant agreement (No. 823844) and Spain, France, Germany. EF would like to thank the support of the Spanish Ministry MCIN/AEI/10.13039/501100011033 and the European Union NextGeneration EU/PRTR for the grant "Europa Investigación 2020" EIN2020-112255, and also the Comunidad de Madrid through the call Research Grants for Young Investigators from the Universidad Politécnica de Madrid, Spain. Finally, all authors gratefully acknowledge the Universidad Politécnica de Madrid (www.upm.es) for providing computing resources on Magerit Supercomputer.

Appendix A. The DGSEM technique

We re-write Eq. (3) in its weak form:

$$\int_0^L \left(\frac{\partial u}{\partial t} + \frac{\partial \widehat{f}}{\partial x} + \frac{\chi}{\eta_1} u \right) \psi \, \mathrm{d}x = 0, \tag{A.1}$$

where $\psi = \psi(x,t)$ is a local smooth test function. Given that $\Omega = [0, L]$ is divided into K elements, the integral is split into the sum of element integrals:

$$\sum_{k=1}^{K} \left\{ \int_{x_{k-1}}^{x_k} \left(\frac{\partial u}{\partial t} + \frac{\partial \widehat{f}}{\partial x} + \frac{\chi}{\eta_1} u \right) \psi \, \mathrm{d}x \right\} = 0. \tag{A.2}$$

Each element, $x = x(\xi)$, is transformed according to: $x = x_{k-1} + (\xi + 1)\Delta x_k/2$, where $\Delta x_k = x_k - x_{k-1}$ and $-1 \le \xi \le 1$. Then, $\mathrm{d}x = (\Delta x_k/2)\mathrm{d}\xi$ and $\partial/\partial x = (2/\Delta x_k)\partial/\partial\xi$. Thus the weak form becomes:

$$\sum_{k=1}^{K} \left\{ \frac{\Delta x_k}{2} \int_{-1}^{1} \left(\frac{\partial u}{\partial t} + \frac{2}{\Delta x_k} \frac{\partial \widehat{f}}{\partial \xi} + \frac{\chi}{\eta_1} u \right) \psi \, \mathrm{d}\xi \right\} = 0. \tag{A.3}$$

Assuming that global variables are represented by K local polynomial variables and substituting the Lagrange interpolation of the test function into the Galerkin weak form, $\psi = \sum \psi_j l_j$, we get the following:

$$\frac{\Delta x_k}{2} \int_{-1}^{1} l_j \frac{\partial u_h^k}{\partial t} \mathrm{d}\xi + \int_{-1}^{1} l_j \frac{\partial \widehat{f}_h^k}{\partial \xi} \mathrm{d}\xi + \frac{\Delta x_k}{2\eta_1} \int_{-1}^{1} l_j \chi^k u_h^k \mathrm{d}\xi = 0, \tag{A.4}$$

for $k = 1, 2, \ldots, K$ and $j = 0, 1, \ldots, N$. The first and third integrals are evaluated as follows:

$$\int_{-1}^{1} l_j \frac{\partial u_h^k}{\partial t} \mathrm{d}\xi = \sum_{i=0}^{N} \int_{-1}^{1} l_i l_j \mathrm{d}\xi \frac{\mathrm{d}u_{h,i}^k}{\mathrm{d}t}, \tag{A.5}$$

$$\int_{-1}^{1} l_j \chi^k u_h^k \mathrm{d}\xi = \sum_{i=0}^{N} \int_{-1}^{1} \chi^k l_i l_j \mathrm{d}\xi u_{h,i}^k, \tag{A.6}$$

whereas the second integral is integrated by parts,

$$\int_{-1}^{1} l_j \frac{\partial \widehat{f}_h^k}{\partial \xi} \mathrm{d}\xi = l_j \mathcal{F}^k \Big|_{-1}^{1} - \int_{-1}^{1} l_j' \widehat{f}_h^k \mathrm{d}\xi, \tag{A.7}$$

being $l_j' = \mathrm{d}l_j/\mathrm{d}\xi$. The VP flux function in the first term is substituted by a numerical flux, i.e.,

$$\mathcal{F}_1^k := \mathcal{F}(u_{h,N}^k, u_{h,0}^{k+1}; +\mathbf{e}_x^k), \tag{A.8}$$

$$\mathcal{F}_{-1}^k := \mathcal{F}(u_{h,N}^{k-1}, u_{h,0}^k; -\mathbf{e}_x^k), \tag{A.9}$$

depending on the normal at the boundary, $\pm\mathbf{e}_x^k$, and the solution at two adjacent elements. We discuss the choice of the numerical flux later.

The remaining integral is divided as follows:

$$\int_{-1}^{1} l'_j \widehat{f}_h^k \mathrm{d}\xi = \sum_{i=0}^{N} \int_{-1}^{1} \widehat{c}^k l_i l'_j \mathrm{d}\xi\, u_{h,i}^k + \sum_{i=0}^{N} \int_{-1}^{1} l_i l'_j \mathrm{d}\xi\, \widehat{f}_{\mathrm{diff}\,h,i}^k. \tag{A.10}$$

with $\widehat{f}_{\mathrm{diff}} = -\widehat{\nu}\partial u/\partial x$ the VP diffusive flux. Substituting the integrals (A.5), (A.6), (A.7), and (A.10), we get the following:

$$\sum_{i=0}^{N} \left\{ \frac{\Delta x_k}{2} \langle l_i, l_j \rangle \frac{\mathrm{d}u_{h,i}^k}{\mathrm{d}t} + \left(\frac{\Delta x_k}{2\eta_1} \langle \chi^k l_i, l_j \rangle - \langle \widehat{c}^k l_i, l'_j \rangle \right) u_{h,i}^k - \langle l_i, l'_j \rangle \widehat{f}_{\mathrm{diff}\,h,i}^k \right\} = - \left. l_j \mathcal{F}^k \right|_{-1}^{1}, \tag{A.11}$$

for $k = 1, 2, \ldots, K$ and $j = 0, 1, \ldots, N$ where the inner product of the given functions $a = a(\xi)$ and $b = b(\xi)$ is defined as follows:

$$\langle a, b \rangle := \int_{-1}^{1} ab \,\mathrm{d}\xi. \tag{A.12}$$

Additionally, the VP diffusive flux involves the derivative of u and must be discretized consistently with the rest of the scheme. If we write the VP diffusive flux in weak form,

$$\sum_{k=1}^{K} \left\{ \frac{\Delta x_k}{2} \int_{-1}^{1} \left(\widehat{f}_{\mathrm{diff}} + \widehat{\nu} \frac{2}{\Delta x_k} \frac{\partial u}{\partial \xi} \right) \psi \,\mathrm{d}\xi \right\} = 0. \tag{A.13}$$

and repeat the interpolating and integration-by-part procedures, we get:

$$\sum_{i=0}^{N} \left\{ \frac{\Delta x_k}{2} \langle l_i, l_j \rangle \widehat{f}_{\mathrm{diff}\,h,i}^k - \langle \widehat{\nu}^k l_i, l'_j \rangle u_{h,i}^k \right\} = - l_j \widehat{\nu}^k \mathcal{U}^k \Big|_{-1}^{1}, \tag{A.14}$$

for $k = 1, 2, \ldots, K$ and $j = 0, 1, \ldots, N$ where $\mathcal{U}^k$ is another numerical flux for the solution, that is,

$$\mathcal{U}_1^k := \mathcal{U}(u_{h,N}^k, u_{h,0}^{k+1}), \tag{A.15}$$

$$\mathcal{U}_{-1}^k := \mathcal{U}(u_{h,N}^{k-1}, u_{h,0}^k). \tag{A.16}$$

The computation of the inner products is done via Gaussian quadrature:

$$\langle l_i, l_j \rangle \approx \sum_{m=0}^{N} w_m l_i(\xi_m) l_j(\xi_m) = w_j \delta_{ij}, \tag{A.17}$$

$$\langle l_i, l'_j \rangle \approx \sum_{m=0}^{N} w_m l_i(\xi_m) l'_j(\xi_m) = w_i l'_j(\xi_i), \tag{A.18}$$

$$\langle \chi^k l_i, l_j \rangle \approx \sum_{m=0}^{N} w_m \chi_m^k l_i(\xi_m) l_j(\xi_m) = w_j \chi_j^k \delta_{ij}, \tag{A.19}$$

$$\langle \widehat{c}^k l_i, l'_j \rangle \approx \sum_{m=0}^{N} w_m \widehat{c}_m^k l_i(\xi_m) l'_j(\xi_m) = w_i \widehat{c}_i^k l'_j(\xi_i), \tag{A.20}$$

$$\langle \widehat{\nu}^k l_i, l'_j \rangle \approx \sum_{m=0}^{N} w_m \widehat{\nu}_m^k l_i(\xi_m) l'_j(\xi_m) = w_i \widehat{\nu}_i^k l'_j(\xi_i), \tag{A.21}$$

where w_m are the Gauss–Lobatto weights ($\sum_{m=0}^{N} w_m = 2$) and $\chi_m^k = \chi^k(\xi_m, t)$, $\widehat{c}_m^k = \widehat{c}^k(\xi_m, t) = c + \chi_m^k/\eta_2$, and $\widehat{\nu}_m^k = \widehat{\nu}^k(\xi_m, t) = \nu - \chi_m^k/\eta_3$. When all of them are combined, Eqs. (7) are obtained.

Finally, the last stage of a DGSEM is the calculation of $\mathcal{F}$ and $\mathcal{U}$ to reproduce the physics of advection and diffusion. A variety of fluxes are available for DG, and most are summarized by Arnold et al. [63]. Here, we use a unifying function:

$$W_{\pm}^k(a, b; \lambda) := \{\{ab\}\}_{\pm}^k - \frac{1}{2}\lambda [\![|a| b]\!]_{\pm}^k, \tag{A.22}$$

for $\lambda \in \mathbb{R}$. If $\lambda = 0$ the discretization becomes a central scheme; $\lambda = -1$, upwind; $\lambda = 1$, downwind. The subscript "+" means the right boundary element and "–" the left boundary element. We also denote $\{\{\cdot\}\}$ as the averaging operator:

$$\{\{a\}\}_+^k := \frac{a_{h,N}^k + a_{h,0}^{k+1}}{2}, \quad \{\{a\}\}_-^k := \frac{a_{h,0}^k + a_{h,N}^{k-1}}{2}, \tag{A.23}$$

and $[\![\cdot]\!]$ as the jump operator:

$$[\![a]\!]_+^k := a_{h,N}^k - a_{h,0}^{k+1}, \quad [\![a]\!]_-^k := a_{h,N}^{k-1} - a_{h,0}^k. \tag{A.24}$$

Once these operators are defined, we divide the numerical flux into an advective term and a diffusive term: $\mathcal{F} = \mathcal{F}_{\mathrm{adv}} + \mathcal{F}_{\mathrm{diff}}$. The computation of the advective numerical flux is as follows:

$$\mathcal{F}_{1,\mathrm{adv}}^k = W_+^k(\widehat{c}, u; \alpha), \tag{A.25}$$

$$\mathcal{F}_{-1,\mathrm{adv}}^k = W_-^k(\widehat{c}, u; \alpha), \tag{A.26}$$

and the diffusive numerical flux is:

$$\mathcal{F}_{1,\mathrm{diff}}^k = W_+^k(1, \widehat{f}_{\mathrm{diff}}; \beta), \tag{A.27}$$

$$\mathcal{F}_{-1,\mathrm{diff}}^k = W_-^k(1, \widehat{f}_{\mathrm{diff}}; \beta). \tag{A.28}$$

The values of $\widehat{f}_{\mathrm{diff}}$ at the boundary elements are computed with:

$$\widehat{f}_{\mathrm{diff}\,h,0}^k = W_-^k\left(\widehat{\nu}, \frac{u}{\Delta x_k}; \gamma\right), \qquad \widehat{f}_{\mathrm{diff}\,h,N}^{k-1} = W_+^k\left(\widehat{\nu}, \frac{u}{\Delta x_{k-1}}; \gamma\right), \tag{A.29}$$

$$\widehat{f}_{\mathrm{diff}\,h,N}^k = W_+^k\left(\widehat{\nu}, \frac{u}{\Delta x_k}; \gamma\right), \qquad \widehat{f}_{\mathrm{diff}\,h,0}^{k+1} = W_-^k\left(\widehat{\nu}, \frac{u}{\Delta x_{k+1}}; \gamma\right). \tag{A.30}$$

Finally, $\mathcal{U}$ is computed as

$$\mathcal{U}_1^k = W_+^k(1, u; \delta), \tag{A.31}$$

$$\mathcal{U}_{-1}^k = W_-^k(1, u; \delta). \tag{A.32}$$

BR1 is recovered by setting $\alpha = -1$ and $\beta = \gamma = \delta = 0$, while LDG is obtained by setting $\alpha = \gamma = -1$ and $\beta = -\delta = -1$. The weights $\mathfrak{f}$ and $\mathfrak{g}$ of (8) are obtained by finding the us at x_{k-1} and x_k from the function W described in Eq. (A.22).

Appendix B. Non-trivial solutions

In all the cases, the considered element is inside the solid region. The first case is related to an inviscid problem without a second derivative penalty term or a viscid problem with $\eta_3 = 1/\nu$. The second case includes second derivatives and is therefore more general.

Case 1: Problem with $\widehat{\nu} = 0$

The parameters TE and HOT are listed in Tables B.5 and B.6. The main findings include the following:

$$\widetilde{c}_j^k = \mathfrak{K}_j^{(1)k} = \widehat{c}, \quad \forall j = 0, 1, 2 \tag{B.1}$$

$$\widetilde{\nu}_j^k = -\frac{1}{2}\mathfrak{K}_j^{(2)k} = 0, \quad \forall j = 0, 1, 2 \tag{B.2}$$

$$\mathfrak{K}_1^{(2p)k} = 0, \quad p \in \mathbb{N} \tag{B.3}$$

In total, we have five unknowns ($\widehat{c}$, $\mathfrak{f}_2^{k-1}$, $\mathfrak{f}_0^k$, $\mathfrak{f}_2^k$, $\mathfrak{f}_0^{k+1}$) and, therefore, a determined system would be:

$$\begin{cases} \mathfrak{f}_2^{k-1} u_{h,2}^{k-1} + \left(\mathfrak{f}_0^k - \widehat{c}\right) u_{h,0}^k = 0, \\ \mathfrak{f}_0^{k+1} u_{h,0}^{k+1} + \left(\mathfrak{f}_2^k + \widehat{c}\right) u_{h,2}^k = 0, \\ \mathfrak{K}_0^{(3)k} = 0, \\ \mathfrak{K}_1^{(3)k} = 0, \\ \mathfrak{K}_2^{(3)k} = 0. \end{cases} \tag{B.4}$$

whose errors are $TE_j^k \sim \mathfrak{K}_j^{(4)k} \sim \mathcal{O}\left(\Delta x_k^3\right)$ for $j = 0, 1, 2$ within Ω_s. However, the unique solution of the system is the trivial one. If we leave η_2 free, the system that determines the numerical flux weights becomes

$$\begin{cases} \mathfrak{f}_2^{k-1} u_{h,2}^{k-1} + \left(\mathfrak{f}_0^k - \widehat{c}\right) u_{h,0}^k = 0, \\ \mathfrak{f}_0^{k+1} u_{h,0}^{k+1} + \left(\mathfrak{f}_2^k + \widehat{c}\right) u_{h,2}^k = 0, \end{cases} \tag{B.5}$$

being $TE_j^k \sim \mathfrak{K}_j^{(3)k} \sim \mathcal{O}\left(\Delta x_k^2\right)$ for $j = 0, 1, 2$. A non-trivial solution would be:

$$\mathfrak{f}_2^{k-1} = \mathfrak{f}_0^{k+1} = 0, \quad \mathfrak{f}_0^k = -\mathfrak{f}_2^k = \widehat{c}, \tag{B.6}$$

Table B.5
The reaction parameter and the coefficient $\mathcal{K}$ in the modified equations for a three-point GL grid and a problem with $\hat{v} = 0$.

j	ξ_j	$\tilde{r}_j^k$	$\mathcal{K}_j^{(m)k}$
0	-1	$\frac{6}{\Delta x_k}\hat{c}$	$\frac{2^{2-m}-1}{\Delta x_k^{1-m}}\hat{c}$
1	0	0	$-2^{-m}\frac{(-1)^m-1}{\Delta x_k^{1-m}}\hat{c}$
2	1	$-\frac{6}{\Delta x_k}\hat{c}$	$(-1)^m\frac{1-2^{2-m}}{\Delta x_k^{1-m}}\hat{c}$

Table B.6
Numerical source in the DG source, $s_{\mathrm{DG},j}^k = S_j^k - \tilde{r}_j^k u_{h,j}^k$, for a problem with $\hat{v} = 0$.

j	S_j^k
0	$\frac{6}{\Delta x_k}\left(\mathfrak{f}_2^{k-1}u_{h,2}^{k-1} + \mathfrak{f}_0^k u_{h,0}^k\right)$
1	0
2	$-\frac{6}{\Delta x_k}\left(\mathfrak{f}_0^{k+1}u_{h,0}^{k+1} + \mathfrak{f}_2^k u_{h,2}^k\right)$

Alternatively, if the upwinding numerical flux is the solution, i.e.

$$\mathfrak{f}_0^k = \mathfrak{f}_0^{k+1} = 0, \quad \mathfrak{f}_2^{k-1} = -\mathfrak{f}_2^k = \hat{c}. \tag{B.7}$$

Then the system becomes:

$$\hat{c}\left(u_{h,2}^{k-1} - u_{h,0}^k\right) = 0, \tag{B.8}$$

whose truncation error leads to:

$$\begin{cases} TE_j^k \sim \mathcal{O}\left(\Delta x_k^2\right), \forall j = 0,1,2, & \text{If } u_{h,2}^{k-1} = u_{h,0}^k \\ TE_0^k \sim \mathcal{O}\left(\Delta x_k^0\right), TE_1^k, TE_2^k \sim \mathcal{O}\left(\Delta x_k^2\right), & \text{If } u_{h,2}^{k-1} \neq u_{h,0}^k, \end{cases} \tag{B.9}$$

Setting $u_{h,2}^{k-1} = u_{h,0}^k$ is very similar to using a Continuous Galerkin (CG) method. A downwind numerical flux mimics the results of upwinding, but for the right-hand boundary element. Other numerical fluxes leave:

$$TE_0^k, TE_2^k \sim \mathcal{O}\left(\Delta x_k^0\right), \quad TE_1^k \sim \mathcal{O}\left(\Delta x_k^2\right). \tag{B.10}$$

Case 2: Problem with $\hat{v} \neq 0$

In this second case (with second derivatives) we consider η_3 free. The parameters of TE and HOT are listed in Tables B.7 and B.8. Again, we conclude that

$$\tilde{c}_0^k = \mathcal{K}_0^{(1)k} = \hat{c} - 4\frac{3 - {}^3/_2\mathfrak{g}_2^k}{\Delta x_k}\hat{v}, \tag{B.11}$$

$$\tilde{c}_1^k = \mathcal{K}_1^{(1)k} = \hat{c} + 3\frac{\mathfrak{g}_2^k - \mathfrak{g}_0^k}{\Delta x_k}\hat{v}, \tag{B.12}$$

$$\tilde{c}_2^k = \mathcal{K}_2^{(1)k} = \hat{c} + 4\frac{3 - {}^3/_2\mathfrak{g}_0^k}{\Delta x_k}\hat{v}, \tag{B.13}$$

and

$$\tilde{v}_0^k = -\frac{1}{2}\mathcal{K}_0^{(2)k} = (4 - 3\mathfrak{g}_2^k)\hat{v}, \tag{B.14}$$

$$\tilde{v}_1^k = -\frac{1}{2}\mathcal{K}_1^{(2)k} = -\frac{1}{4}\left(2 + 3\left(\mathfrak{g}_0^k + \mathfrak{g}_2^k\right)\right)\hat{v}, \tag{B.15}$$

$$\tilde{v}_2^k = -\frac{1}{2}\mathcal{K}_2^{(2)k} = (4 - 3\mathfrak{g}_0^k)\hat{v}, \tag{B.16}$$

Since $\tilde{c}_j^k \neq \hat{c}$ for $j = 0,1,2$, the term $\hat{c}_j^k - \Delta\xi\tilde{c}_j^k$ in the truncation error should be suppressed since it is $\mathcal{O}(\Delta x_k^{-1})$. The choice is $\mathfrak{g}_2^k = \mathfrak{g}_0^k = 2$. However, $\hat{v}_j^k - \Delta\xi\tilde{v}_j^k \neq 0$ for $j = 0,1,2$ and therefore $TE_j^k \sim \mathcal{O}\left(\Delta x_k^0\right)$. If we want to find an optimal value of η_3 to increase the order of the scheme, we come to the conclusion that $\hat{v} = 0$, but this case was already discussed previously. Keeping $\mathfrak{g}_2^k = \mathfrak{g}_0^k = 2$ and η_3 free, $\mathfrak{g}_2^{k-1} = \mathfrak{g}_0^{k+1} = 0$ kills $s_{\mathrm{DG},1}^k$, to have $s_{\mathrm{DG},0}^k, s_{\mathrm{DG},2}^k = 0$,

$$\begin{cases} \mathfrak{f}_2^{k-1}u_{h,2}^{k-1} + \left(\mathfrak{f}_0^k - \hat{c} - \frac{4}{\Delta x_k}\hat{v}\right)u_{h,0}^k = 0, \\ \mathfrak{f}_0^{k+1}u_{h,0}^{k+1} + \left(\mathfrak{f}_2^k + \hat{c} + \frac{4}{\Delta x_k}\hat{v}\right)u_{h,2}^k = 0. \end{cases} \tag{B.17}$$

In this case, a solution of the system is as follows:

$$\mathfrak{f}_2^{k-1} = \mathfrak{f}_0^{k+1} = 0, \quad \mathfrak{f}_0^k = -\mathfrak{f}_2^k = \hat{c} + \frac{4}{\Delta x_k}\hat{v}, \tag{B.18}$$

Additionally, if upwind in such a way that

$$\mathfrak{f}_0^k = \mathfrak{f}_0^{k+1} = 0, \quad \mathfrak{f}_2^{k-1} = -\mathfrak{f}_2^k = \hat{c} + \frac{4}{\Delta x_k}\hat{v}, \tag{B.19}$$

the second equation of the system is met, but the first one becomes:

$$\left(\hat{c} + \frac{4}{\Delta x_k}\hat{v}\right)\left(u_{h,2}^{k-1} - u_{h,0}^k\right) = 0. \tag{B.20}$$

To eliminate this term, a relation of ηs is obtained, $\hat{c} + (4/\Delta x_k)\hat{v} = 0$, since in a DG method $u_{h,2}^{k-1} \neq u_{h,0}^k$. Note that in a CG method, it is not necessary to fill this relation, since the solution is continuous between elements. In all cases described previously, $TE_j^k \sim \mathcal{O}(\Delta x_k^0)$ for $j = 0,1,2$.

A summary of all the conditions derived can be found in Table 3.

Appendix C. One-dimensional advection problem based on non-body-fitted mesh

In this section, we perform numerical tests for the one-dimensional advection problem on a non-body-fitted mesh. We consider a solid region length $1.5\Delta x$. The purpose of this case is twofold: (1) analyzing the effect of a solid region that spans several Δx; (2) mimic the non-body-fitted grid where the boundary interface lies within an element. The solid domain spans from $x = -0.75\Delta x$ to $x = 0.75\Delta x$. The total number of element remains to be $N = 40$, leading to a solid ratio $r = 3/80$. The same initial condition with $\omega\Delta x/(N+1) = 0.3142$ is selected and the final solution time is set to $t = 1.1$. The same simulation is reproduced for the advection problem, considering a range of η_1 and η_2. The error comparison is shown in Fig. C.12.

From Fig. C.12, we can draw the same conclusions as the body-fitted case. Firstly, as η_1 decreases, the error in both the fluid and the solid regions decreases, since the modeling error is reduced. Secondly, the optimal η_2 leads to the minimal error both in the fluid and the solid regions at small η_1. For different polynomial orders, when the optimal parameter $\eta_2 = -1/c$ is used, the boundary condition is satisfied almost exactly.

Appendix D. Two-dimensional advection–diffusion problem with different parameters

In this section, a more challenging two-dimensional problem is studied. We simulate the two-dimensional advection–diffusion with different parameters (velocities and diffusivities) in different directions. We introduce two combinations of parameter. We fix the other problem settings, while the time step is reduced to 10^{-5} to avoid numerical instability. The LDG scheme is used for the viscous flux since it is more accurate. The penalization parameter η_1 is set to 10^{-4}, while the other penalization parameters η_2 and η_3 (if the corresponding terms are imposed) are set to the optimal values. It should be noted that in these cases, the optimal parameters η_2 and η_3 are different in each direction, obtained from the corresponding velocity and diffusivity. Numerical errors are compared in Table D.9, where different types of penalization are included.

As shown in the table, we can observe the same trend as in Section 5.2. The largest error can be seen for the classic volume penalization, where only the solution is penalized. In addition, adding additional penalization on first-order and second-order terms (with the optimal penalization parameters) can largely improve the overall

Table B.7
The reaction parameter and the coefficient $\mathcal{K}$ in the modified equations for a three-point GL grid and a problem with $\hat{v} \neq 0$.

j	ξ_j	$\tilde{r}_j^k$	$\mathcal{K}_j^{(m)k}$
0	-1	$\frac{6}{\Delta x_k}\hat{c} - 4\frac{6}{\Delta x_k^2}\hat{v}$	$\frac{2^{2-m}-1}{\Delta x_k^{1-m}}\hat{c} - 4\frac{2^{2-m}+1-{}^3/_2\mathfrak{g}_2^k}{\Delta x_k^{2-m}}\hat{v}$
1	0	$2\frac{6}{\Delta x_k^2}\hat{v}$	$-2^{-m}\frac{(-1)^m-1}{\Delta x_k^{1-m}}\hat{c} + 2^{1-m}\frac{(-1)^m\left(1+3\mathfrak{g}_0^k\right)+1+3\mathfrak{g}_2^k}{\Delta x_k^{2-m}}\hat{v}$
2	1	$-\frac{6}{\Delta x_k}\hat{c} - 4\frac{6}{\Delta x_k^2}\hat{v}$	$(-1)^m\left(\frac{1-2^{2-m}}{\Delta x_k^{1-m}}\hat{c} - 4\frac{2^{2-m}+1-{}^3/_2\mathfrak{g}_0^k}{\Delta x_k^{2-m}}\hat{v}\right)$

Table B.8
Numerical source in the DG source, $s_{\mathrm{DG},j}^k = S_j^k - \tilde{r}_j^k u_{h,j}^k$, for a problem with $\hat{v} \neq 0$.

j	S_j^k
0	$\frac{2}{\Delta x_k}\left[3\left(\mathfrak{f}_2^{k-1}u_{h,2}^{k-1} + \mathfrak{f}_0^k u_{h,0}^k\right) - \frac{3}{\Delta x_k}\left(3\mathfrak{g}_2^{k-1}u_{h,2}^{k-1} + \mathfrak{g}_0^{k+1}u_{h,0}^{k+1} + \left(3\mathfrak{g}_0^k + \mathfrak{g}_2^k\right)u_{h,0}^k\right)\hat{v}\right]$
1	$\frac{6}{\Delta x_k^2}\left[\mathfrak{g}_2^{k-1}u_{h,2}^{k-1} + \mathfrak{g}_0^{k+1}u_{h,0}^{k+1} + \left(\mathfrak{g}_0^k + \mathfrak{g}_2^k\right)u_{h,1}^k\right]\hat{v}$
2	$\frac{2}{\Delta x_k}\left[-3\left(\mathfrak{f}_2^k u_{h,2}^k + \mathfrak{f}_0^{k+1}u_{h,0}^{k+1}\right) - \frac{3}{\Delta x_k}\left(\mathfrak{g}_2^{k-1}u_{h,2}^{k-1} + 3\mathfrak{g}_0^{k+1}u_{h,0}^{k+1} + \left(3\mathfrak{g}_2^k + \mathfrak{g}_0^k\right)u_{h,2}^k\right)\hat{v}\right]$

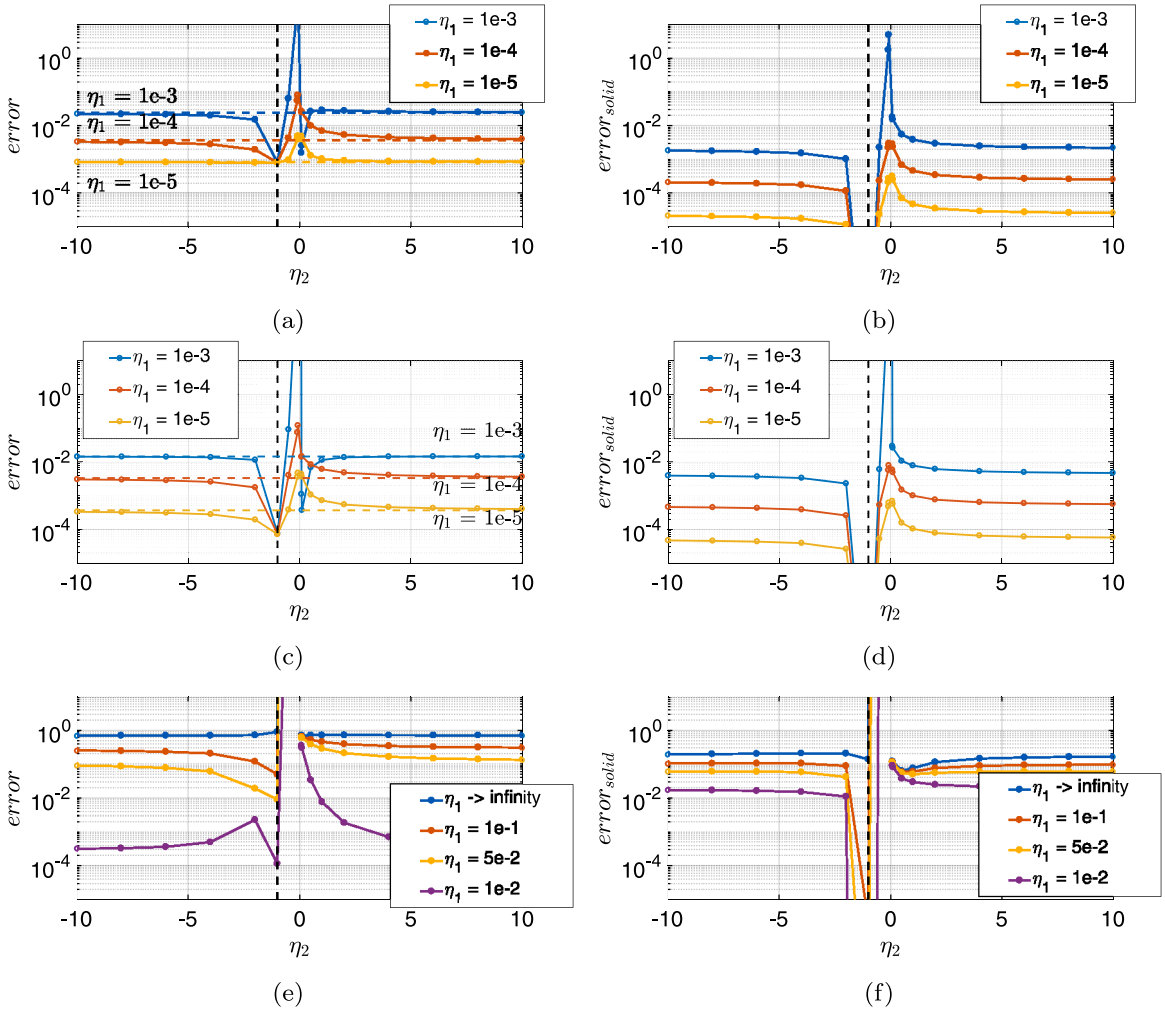

Fig. C.12. Error comparison for the advection equation based on the non-body-fitted mesh, vertical dashed line refers to $\eta_2 = -1/c$, and horizontal dashed line refers to $\eta_2 \to \infty$. (a) Error in the flow ($N = 2$). (b) Error in the solid, the optimal solution is zero ($N = 2$). (c) Error in the flow ($N = 3$). (d) Error in the solid, the optimal solution is zero ($N = 3$). (e) Error in the flow (larger penalization parameter, $N = 3$). (f) Error in the solid (larger penalization parameter, $N = 3$).

Table D.9
Error comparison (error region in Fig. 9) of the two-dimensional advection–diffusion equation with IBM wall under different flow parameters and different penalization terms.

Parameters	η_1 term	η_1, η_2 terms	η_1, η_2, η_3 terms
$c_x = 1,\ c_y = 1.5,$ $\nu_x = 0.0015,\ \nu_y = 0.001$	1.7970×10^{-4}	1.6067×10^{-5}	5.4686×10^{-7}
$c_x = 1,\ c_y = 2,$ $\nu_x = 0.001,\ \nu_y = 0.002$	4.4634×10^{-5}	5.6013×10^{-6}	6.8056×10^{-7}

accuracy, where the best performance is observed when all three types of penalization are considered.

References

[1] Mittal R, Iaccarino G. Immersed boundary methods. Annu Rev Fluid Mech 2005;37:239–61.
[2] Huang W-X, Tian F-B. Recent trends and progress in the immersed boundary method. Proc Inst Mech Eng C 2019;233(23–24):7617–36.
[3] Sotiropoulos F, Yang X. Immersed boundary methods for simulating fluid–structure interaction. Prog Aerosp Sci 2014;65:1–21.
[4] Kim W, Choi H. Immersed boundary methods for fluid-structure interaction: A review. Int J Heat Fluid Flow 2019;75:301–9.
[5] Griffith BE, Patankar NA. Immersed methods for fluid–structure interaction. Annu Rev Fluid Mech 2020;52:421–48.
[6] Peskin CS. Flow patterns around heart valves: a numerical method. J Comput Phys 1972;10(2):252–71.
[7] Ye T, Mittal R, Udaykumar H, Shyy W. An accurate cartesian grid method for viscous incompressible flows with complex immersed boundaries. J Comput Phys 1999;156(2):209–40.
[8] Udaykumar H, Mittal R, Rampunggoon P, Khanna A. A sharp interface cartesian grid method for simulating flows with complex moving boundaries. J Comput Phys 2001;174(1):345–80.
[9] Fidkowski KJ, Darmofal DL. A triangular cut-cell adaptive method for high-order discretizations of the compressible Navier–Stokes equations. J Comput Phys 2007;225(2):1653–72.
[10] Sticko S, Kreiss G. Higher order cut finite elements for the wave equation. J Sci Comput 2019;80(3):1867–87.
[11] Majumdar S, Iaccarino G, Durbin P. RANS solvers with adaptive structured boundary non-conforming grids. Annu Res Briefs 2001;1.
[12] Taira K, Colonius T. The immersed boundary method: a projection approach. J Comput Phys 2007;225(2):2118–37.
[13] Fadlun E, Verzicco R, Orlandi P, Mohd-Yusof J. Combined immersed-boundary finite-difference methods for three-dimensional complex flow simulations. J Comput Phys 2000;161(1):35–60.
[14] Luo H, Dai H, de Sousa PJF, Yin B. On the numerical oscillation of the direct-forcing immersed-boundary method for moving boundaries. Comput & Fluids 2012;56:61–76.
[15] Angot P, Bruneau C-H, Fabrie P. A penalization method to take into account obstacles in incompressible viscous flows. Numer Math 1999;81(4):497–520.
[16] Kolomenskiy D, Schneider K. A Fourier spectral method for the Navier–Stokes equations with volume penalization for moving solid obstacles. J Comput Phys 2009;228(16):5687–709.
[17] Arquis E, Caltagirone J. Sur les conditions hydrodynamiques au voisinage d'une interface milieu fluide-milieu poreux: applicationa la convection naturelle. CR Acad Sci Paris II 1984;299:1–4.
[18] Kadoch B, Kolomenskiy D, Angot P, Schneider K. A volume penalization method for incompressible flows and scalar advection–diffusion with moving obstacles. J Comput Phys 2012;231(12):4365–83.
[19] Sakurai T, Yoshimatsu K, Okamoto N, Schneider K. Volume penalization for inhomogeneous Neumann boundary conditions modeling scalar flux in complicated geometry. J Comput Phys 2019;390:452–69.
[20] Courant R. Variational methods for the solution of problems of equilibrium and vibrations. Bull Amer Math Soc 1943;49:1–23. http://dx.doi.org/10.1090/S0002-9904-1943-07818-4.
[21] Carbou G, Fabrie P. Boundary layer for a penalization method for viscous incompressible flow. Adv Differential Equations 2003;8(12):1453–80.
[22] Ramière I, Angot P, Belliard M. A general fictitious domain method with immersed jumps and multilevel nested structured meshes. J Comput Phys 2007;225(2):1347–87.
[23] Thirumalaisamy R, Patankar NA, Bhalla APS. Handling Neumann and robin boundary conditions in a fictitious domain volume penalization framework. 2021, arXiv preprint arXiv:2101.02806.
[24] Liu Q, Vasilyev OV. A Brinkman penalization method for compressible flows in complex geometries. J Comput Phys 2007;227(2):946–66.
[25] Brown-Dymkoski E, Kasimov N, Vasilyev OV. A characteristic based volume penalization method for general evolution problems applied to compressible viscous flows. J Comput Phys 2014;262:344–57.
[26] Abgrall R, Beaugendre H, Dobrzynski C. An immersed boundary method using unstructured anisotropic mesh adaptation combined with level-sets and penalization techniques. J Comput Phys 2014;257:83–101.
[27] Abalakin I, Zhdanova N, Kozubskaya T. Immersed boundary method implemented for the simulation of an external flow on unstructured meshes. Math Models Comput Simul 2016;8(3):219–30.
[28] Horgue P, Prat M, Quintard M. A penalization technique applied to the "Volume-Of-Fluid" method: Wettability condition on immersed boundaries. Comput & Fluids 2014;100:255–66.
[29] Komatsu R, Iwakami W, Hattori Y. Direct numerical simulation of aeroacoustic sound by volume penalization method. Comput & Fluids 2016;130:24–36.
[30] Engels T, Kolomenskiy D, Schneider K, Sesterhenn J. Numerical simulation of fluid–structure interaction with the volume penalization method. J Comput Phys 2015;281:96–115.
[31] Cui X, Yao X, Wang Z, Liu M. A coupled volume penalization-thermal lattice Boltzmann method for thermal flows. Int J Heat Mass Transfer 2018;127:253–66.
[32] Lew AJ, Buscaglia GC. A discontinuous-Galerkin-based immersed boundary method. Internat J Numer Methods Engrg 2008;76(4):427–54. http://dx.doi.org/10.1002/nme.2312, URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/nme.2312.
[33] Lew AJ, Negri M. Optimal convergence of a discontinuous-Galerkin-based immersed boundary method. ESAIM Math Model Numer Anal 2011;45(4):651–74. http://dx.doi.org/10.1051/m2an/2010069.
[34] Müller B, Krämer-Eis S, Kummer F, Oberlack M. A high-order discontinuous Galerkin method for compressible flows with immersed boundaries. Internat J Numer Methods Engrg 2017;110(1):3–30.
[35] Kou J, Joshi S, Hurtado-de Mendoza A, Puri K, Hirsch C, Ferrer E. High-order flux reconstruction based on immersed boundary method. In: 14th WCCM-ECCOMAS congress 2020, vol. 700. 2021.
[36] Kou J, Joshi S, Hurtado-de Mendoza A, Puri K, Hirsch C, Ferrer E. Immersed boundary method for high-order flux reconstruction based on volume penalization. J Comput Phys 2022;448:110721.
[37] Beyer RP, LeVeque RJ. Analysis of a one-dimensional model for the immersed boundary method. SIAM J Numer Anal 1992;29(2):332–64.
[38] Li Z. On convergence of the immersed boundary method for elliptic interface problems. Math Comp 2015;84(293):1169–88.
[39] LeVeque RJ, Li Z. The immersed interface method for elliptic equations with discontinuous coefficients and singular sources. SIAM J Numer Anal 1994;31(4):1019–44.
[40] Tornberg A-K, Engquist B. Numerical approximations of singular source terms in differential equations. J Comput Phys 2004;200(2):462–88.
[41] Mori Y. Convergence proof of the velocity field for a Stokes flow immersed boundary method. Comm Pure Appl Math: J Issued Inst Math Sci 2008;61(9):1213–63.
[42] Chen K-Y, Feng K-A, Kim Y, Lai M-C. A note on pressure accuracy in immersed boundary method for Stokes flow. J Comput Phys 2011;230(12):4377–83.
[43] Liu Y, Mori Y. L^p Convergence of the immersed boundary method for stationary Stokes problems. SIAM J Numer Anal 2014;52(1):496–514.
[44] Guy RD, Hartenstine DA. On the accuracy of direct forcing immersed boundary methods with projection methods. J Comput Phys 2010;229(7):2479–96.
[45] Zhou K, Balachandar S. An analysis of the spatio-temporal resolution of the immersed boundary method with direct forcing. J Comput Phys 2021;424:109862.
[46] Zhang M, Zheng ZC. High-order immersed-boundary simulation and error analysis for flow around a porous structure. In: ASME 2017 international mechanical engineering congress and exposition. American Society of Mechanical Engineers Digital Collection; 2017.
[47] Shyy W. A study of finite difference approximations to steady-state, convection-dominated flow problems. J Comput Phys 1985;57(3):415–38.
[48] Moura RC, Sherwin S, Peiró J. Modified equation analysis for the discontinuous Galerkin formulation. In: Kirby R, Berzins M, Hesthaven J, editors. Spectral and high order methods for partial differential equations. Lecture notes in computational science and engineering, vol. 106, Cham, Germany: Springer; 2015.
[49] Warming RF, Hyett B. The modified equation approach to the stability and accuracy analysis of finite-difference methods. J Comput Phys 1974;14(2):159–79.
[50] Shubin GR, Bell JB. A modified equation approach to constructing fourth order methods for acoustic wave propagation. SIAM J Sci Stat Comput 1987;8(2):135–51.

[51] Kou J, Hurtado-de Mendoza A, Joshi S, Le Clainche S, Ferrer E. Eigensolution analysis of immersed boundary method based on volume penalization: applications to high-order schemes. J Comput Phys 2022;449:110817.
[52] Kou J, Ferrer E. A combined volume penalization/selective frequency damping approach for immersed boundary methods applied to high-order schemes. J Comput Phys 2023;472:111678.
[53] Seo JH, Moon YJ. Linearized perturbed compressible equations for low Mach number aeroacoustics. J Comput Phys 2006;218(2):702–19.
[54] Sipp D, Marquet O, Meliga P, Barbagallo A. Dynamics and control of global instabilities in open-flows: A linearized approach. Appl Mech Rev 2010;63(3). http://dx.doi.org/10.1115/1.4001478, arXiv:https://asmedigitalcollection.asme.org/appliedmechanicsreviews/article-pdf/63/3/030801/5442879/030801_1.pdf, 030801.
[55] Schneider K. Immersed boundary methods for numerical simulation of confined fluid and plasma turbulence in complex geometries: a review. J Plasma Phys 2015;81:435810601. http://dx.doi.org/10.1017/S0022377815000598.
[56] Marcon J, Castiglioni G, Moxey D, Sherwin SJ, Peiró J. rp-adaptation for compressible flows. Internat J Numer Methods Engrg 2020;121(23):5405–25.
[57] Manzanero J, Rubio G, Kopriva DA, Ferrer E, Valero E. An entropy–stable discontinuous Galerkin approximation for the incompressible Navier–Stokes equations with variable density and artificial compressibility. J Comput Phys 2020;408:109241.
[58] Wintermeyer N, Winters AR, Gassner GJ, Kopriva DA. An entropy stable nodal discontinuous Galerkin method for the two dimensional shallow water equations on unstructured curvilinear meshes with discontinuous bathymetry. J Comput Phys 2017;340:200–42.
[59] 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.
[60] Goldstein DB, Handler RA, Sirovich L. The local discontinuous Galerkin method for time-dependent convection-diffusion systems. J Comput Phys 1993;105:354–66.
[61] Kolomenskiy D, Schneider K, et al. Analysis and discretization of the volume penalized Laplace operator with Neumann boundary conditions. Appl Numer Math 2015;95:238–49.
[62] Hesthaven JS, Warburton T. Nodal discontinuous galerkin methods: Algorithms, analysis, and applications. Springer Science & Business Media; 2007.
[63] Arnold DN, Brezzi F, Cockburn B, Marini LD. Unified analysis of discontinuous Galerkin methods for elliptic problems. J Numer Anal 2002;39:1749–79.