RESEARCH ARTICLE

High-order DG solvers for underresolved turbulent incompressible flows: A comparison of L^2 and $H(\mathrm{div})$ methods

Niklas Fehn[1] | **Martin Kronbichler[1]** | **Christoph Lehrenfeld[2]** | **Gert Lube[2]** | **Philipp W. Schroeder[2]**

[1]Institute for Computational Mechanics, Department of Mechanical Engineering, Technical University of Munich, Garching, Germany

[2]Institute for Numerical and Applied Mathematics, Georg-August-Universität Göttingen, Göttingen, Germany

Correspondence
Christoph Lehrenfeld, Institute for Numerical and Applied Mathematics, Georg-August-Universität Göttingen, 37083 Göttingen, Germany.
Email: lehrenfeld@math.uni-goettingen.de

Summary

The accurate numerical simulation of turbulent incompressible flows is a challenging topic in computational fluid dynamics. For discretisation methods to be robust in the underresolved regime, mass conservation and energy stability are key ingredients to obtain robust and accurate discretisations. Recently, two approaches have been proposed in the context of high-order discontinuous Galerkin (DG) discretisations that address these aspects differently. On the one hand, standard L^2-based DG discretisations enforce mass conservation and energy stability weakly by the use of additional stabilisation terms. On the other hand, pointwise divergence-free $H(\mathrm{div})$-conforming approaches ensure exact mass conservation and energy stability by the use of tailored finite element function spaces. This work raises the question whether and to which extent these two approaches are equivalent when applied to underresolved turbulent flows. This comparative study highlights similarities and differences of these two approaches. The numerical results emphasise that both discretisation strategies are promising for underresolved simulations of turbulent flows due to their inherent dissipation mechanisms.

KEYWORDS

discontinuous Galerkin, high-order finite elements, incompressible Navier-Stokes equations, Taylor-Green vortex, turbulent channel flow, underresolved turbulent flows

1 | INTRODUCTION

1.1 | Motivation

Simulating turbulent flows is still a challenging undertaking, even on today's high-performance computing architectures. Discontinuous Galerkin (DG) discretisations are currently investigated to develop new discretisation methods with inbuilt stabilisation mechanisms rendering these methods robust and accurate when applied to turbulent flow problems. In this contribution, we compare the accuracy of two high-order DG solvers for incompressible flows with a special emphasis on how they perform in the practically relevant situation of only being able to marginally resolve the occurring flow features. While the two "extreme cases" of direct numerical simulation (DNS) on the one hand and Reynolds-averaged Navier-Stokes (RANS) simulations on the other hand are relatively well established and well understood, being able to

perform time-dependent turbulent flow simulations with only a limited amount of fine-scale accuracy usually goes by the name large-eddy simulation (LES). This regime is exactly where we are interested in in this work.

Large-eddy simulations (LES) by now have a rich history in both the engineering and the mathematics literature, and there are numerous promising approaches for different flow situations. Our aim is to be able to compute (incompressible) flows without the need of choosing the "correct" turbulence model or set of parameters. Therefore, we rely on what can be called a "no-model LES" in the sense that no explicit turbulence model shall be incorporated in the simulations in this work. Such an approach, in turn, relies heavily on distinctive robustness and healthy and controllable dissipation properties of the numerical scheme to approximate the incompressible Navier-Stokes equations. While DG methods have reached a mature state in the field of the compressible Navier-Stokes equations[1-7] in the context of low- and moderate-Mach number turbulent flows, designing robust DG discretisations of the incompressible Navier-Stokes equations exhibits some subtleties in the context of underresolved turbulence: Applying well-known numerical flux formulations to the discretisation of convective and viscous terms only[8-13] has been realised to be not robust enough in underresolved scenarios, as pointed out in recent works.[14-17] It is worth mentioning that this lack of robustness is not related to underintegration of nonlinear terms, commonly known as aliasing. Instead, additional techniques are required, which are inherently linked to the nature of the incompressible Navier-Stokes equations and, in particular, the incompressibility constraint. Namely, effort has to be put into ensuring compliance with both the divergence-free and the normal-continuity constraints.

Two promising methods have been proposed recently by Lehrenfeld and Schöberl[18] and Fehn et al,[19] which rely on different stabilisation concepts. The two candidates under investigation are intelligently stabilised L^2-based DG methods[19] and exactly divergence-free H(div)-conforming methods.[18,20] A first connection between both approaches has been pointed out in the works of Fehn et al[19] and Akbas et al.[21] However, it remains rather unexplored how these different stabilisation techniques behave regarding their dissipation mechanism in the underresolved regime, which is particularly relevant for practical LES of incompressible turbulent flows. Therefore, the aim of this work is a comparison of the accuracy of both approaches in underresolved turbulence simulation, which is examined numerically for the well-known 3D benchmark cases of freely decaying turbulence (Taylor-Green vortex) and wall-bounded turbulence in a channel. The numerical results for the H(div)-conforming method have been obtained using NGSolve,[22] the results for the L^2-based DG method using the open-source finite element library `deal.II`.[23]

1.2 | Mathematical model

The underlying mathematical model is the incompressible Navier-Stokes problem[24-26] without any additional terms, ie,

$$
\begin{cases}
\partial_t \boldsymbol{u} - \nu \Delta \boldsymbol{u} + (\boldsymbol{u} \cdot \nabla)\boldsymbol{u} + \nabla p = \boldsymbol{f} & \text{in } (0, T] \times \Omega, & \text{(1a)} \\
\nabla \cdot \boldsymbol{u} = 0 & \text{in } (0, T] \times \Omega, & \text{(1b)} \\
\boldsymbol{u}(0, \cdot) = \boldsymbol{u}_0(\cdot) & \text{in } \Omega. & \text{(1c)}
\end{cases}
$$

In (1), $\Omega \subset \mathbb{R}^3$ denotes a connected bounded Lipschitz domain. Concerning boundary conditions (BCs), we assume that $\partial\Omega = \Gamma_0 \dot{\cup} \Gamma_{\text{per}}$, where the no-slip BC $\boldsymbol{u} = \boldsymbol{0}$ is prescribed on Γ_0 and periodic BCs are imposed on Γ_{per}. Moreover, $\boldsymbol{u}$: $(0, T] \times \Omega \to \mathbb{R}^3$ indicates the velocity field, p : $(0, T] \times \Omega \to \mathbb{R}$ is the (zero-mean) kinematic pressure, $\boldsymbol{f}$: $(0, T] \times \Omega \to \mathbb{R}^3$ represents external body forces, and $\boldsymbol{u}_0$: $\Omega \to \mathbb{R}^3$ stands for a suitable initial condition for the velocity. The underlying fluid is assumed to be Newtonian with constant (dimensionless) kinematic viscosity $0 < \nu \ll 1$.

1.3 | Outline

We introduce both DG discretisations under consideration with a focus on the mathematical formulation and discuss properties such as approximation abilities, mass conservation, and dissipation mechanisms in Section 2. The accuracy of both methods for the 3D Taylor-Green vortex problem as a prototype problem of decaying homogeneous isotropic turbulence is investigated in Section 3. Apart from a direct comparison in terms of the kinetic energy dissipation rate on different meshes, we also investigate the impact of low-order vs high-order discretisations and the impact of variations of the DG formulation such as changes in the stabilisation terms, and the flux formulations for convective and viscous terms. To include wall-bounded turbulent flows in our investigations, we consider the turbulent channel flow problem in Section 4. In that section, we compare the behaviour of both methods under mesh refinement and again discuss the influence of method variations including the handling of anisotropic refinements towards the wall boundary with affine

and isoparametric finite element meshes. The robustness of the considered approaches and their reliability in correctly predicting turbulent flows is further assessed by parameter studies. The final part, Section 5, summarises the observations made in this work and derives conclusions.

2 | TWO DG-BASED SOLVERS FOR TURBULENT INCOMPRESSIBLE FLOWS

2.1 | Notation

Let $\mathcal{T}_h$ be a partition of Ω into hexahedra with mesh size $h = \max_{K \in \mathcal{T}_h} h_K$, where h_K denotes the diameter of the particular element $K \in \mathcal{T}_h$. The skeleton $\mathcal{F}_h$ denotes the set of all facets and $\mathcal{F}_h = \mathcal{F}_h^i \cup \mathcal{F}_h^\partial$, where $\mathcal{F}_h^i$ is the subset of interior plus periodic facets and $\mathcal{F}_h^\partial$ collects all Dirichlet boundary facets $F \subset \Gamma_0$. To any $F \in \mathcal{F}_h$, we assign a unit normal vector $\boldsymbol{n}_F$ where, for $F \subset \partial\Omega$, this is the outer unit normal vector $\boldsymbol{n}$. If $F \in \mathcal{F}_h^i$, there are two adjacent elements K^+ and K^- sharing the facet $F = \partial K^+ \cap \partial K^-$ and $\boldsymbol{n}_F$ points from K^+ to K^-. Let ϕ be any piecewise smooth (scalar-, vector-, or matrix-valued) function with traces from within the interior of $K^\pm$ denoted by $\phi^\pm$, respectively. Then, we define the jump $[\![\cdot]\!]_F$ and average $\{\!\{\cdot\}\!\}_F$ operators across interior and periodic facets $F \in \mathcal{F}_h^i$ by

$$[\![\phi]\!]_F = \phi^+ - \phi^- \quad \text{and} \quad \{\!\{\phi\}\!\}_F = \frac{1}{2}(\phi^+ + \phi^-). \tag{2}$$

These operators act componentwise for vector- and matrix-valued functions. Frequently, the subscript indicating the particular facet is omitted.

As is usual in DG methods, we make use of the broken vector-valued Sobolev space ($m \in \mathbb{N}_0$)

$$\boldsymbol{H}^m(\mathcal{T}_h) = \left\{ \boldsymbol{v} \in \boldsymbol{L}^2(\Omega) : \boldsymbol{v}|_K \in \boldsymbol{H}^m(K), \; \forall \, K \in \mathcal{T}_h \right\},$$

and define the (elementwise) broken Jacobian $\nabla_h : \boldsymbol{H}^1(\mathcal{T}_h) \to \boldsymbol{L}^2(\Omega)$ by $(\nabla_h \boldsymbol{v})|_K = \nabla(\boldsymbol{v}|_K)$ and the (elementwise) broken divergence $\nabla_h \cdot : \boldsymbol{H}^1(\mathcal{T}_h) \to \boldsymbol{L}^2(\Omega)$ by $(\nabla_h \cdot \boldsymbol{v})|_K = \nabla \cdot (\boldsymbol{v}|_K)$ for all $K \in \mathcal{T}_h$.

In the remainder of this subsection, we introduce local and global finite element spaces. For ease of presentation, we consider only hexahedral meshes that consist of straight hexahedra K, which are aligned with the coordinate axes and are the image of the reference hexahedron $\hat{K} = [0, 1]^3$ under affine linear mappings. However, the general case of nonaffinely mapped hexadra can also be dealt with easily, cf the discussion in Section 4.1 below.

In the following, $\mathbb{Q}_k(K)$ (vector-valued $\boldsymbol{Q}_k(K)$) denotes the tensor-product space of all polynomials on K with degree less or equal k in each direction separately, $\mathbb{Q}_k(K) = \mathbb{P}_k \otimes \mathbb{P}_k \otimes \mathbb{P}_k$. The $\boldsymbol{L}^2$-based method makes use of these local spaces by glueing them together discontinuously, resulting in the global spaces

$$\mathbb{Q}_k := \bigoplus_{K \in \mathcal{T}_h} \mathbb{Q}_k(K), \quad \boldsymbol{Q}_k := \bigoplus_{K \in \mathcal{T}_h} \boldsymbol{Q}_k(K). \tag{3}$$

For the $\boldsymbol{H}(\text{div})$-based method, let us first introduce

$$\boldsymbol{H}_0(\text{div}; \Omega) = \{ \boldsymbol{v} \in \boldsymbol{L}^2(\Omega) : \nabla \cdot \boldsymbol{v} \in L^2(\Omega); \boldsymbol{v} \cdot \boldsymbol{n}|_{\Gamma_0} = 0 \}.$$

Note that a function in $\boldsymbol{H}_0(\text{div}; \Omega)$ at least has to be normal continuous. Because we are working on hexahedral meshes in 3D in this work, the local velocity space is based on the Raviart-Thomas element[27]

$$\mathbb{RT}_{[k]}(K) = (\mathbb{P}_{k+1,k,k}(K), \mathbb{P}_{k,k+1,k}(K), \mathbb{P}_{k,k,k+1}(K))^\dagger, \quad \mathbb{P}_{k_1,k_2,k_3}(K) = \mathbb{P}_{k_1} \otimes \mathbb{P}_{k_2} \otimes \mathbb{P}_{k_3}.$$

Note that $\mathbb{RT}_{[k]}(K)$ uses different scalar polynomial spaces for each component. It includes polynomials of order $k + 1$, $\mathbb{P}_{k+1}$, in the coordinate direction aligned to the velocity component, and polynomials of order k, $\mathbb{P}_k$, in the other directions, and thus, much fewer basis functions than $\boldsymbol{Q}_{k+1}(K)$. However, similar to $\boldsymbol{Q}_k(K)$, every component of $\mathbb{RT}_{[k]}(K)$ consists of a simple tensorial construction based on scalar one-dimensional polynomials ($\mathbb{P}_k$ or $\mathbb{P}_{k+1}$). There holds the inclusion $\boldsymbol{Q}_k(K) \subset \mathbb{RT}_{[k]}(K) \subset \boldsymbol{Q}_{k+1}(K)$ We further note that the highest-order polynomials in $\mathbb{RT}_{[k]}(K)$ do not improve the approximation order of the local polynomial space but are added on top of $\boldsymbol{Q}_k(K)$ for the purpose of improving the approximation of the divergence operator and compatibility with the pressure space. Taking the divergence of this space

reduces the polynomial degree by one exactly for those components that have the higher degree $k + 1$. As a result, there holds $\nabla \cdot \mathbb{RT}_{[k]}(K) = \mathbb{Q}_k(K) \supset \nabla \cdot \mathbf{Q}_k(K)$. We define the global finite element spaces for the $\mathbf{H}$(div)-based method as

$$\mathbb{RT}_{[k]} := \mathbf{H}_0(\text{div}; \Omega) \cap \bigoplus_{K \in \mathcal{T}_h} \mathbb{RT}_{[k]}(K). \tag{4}$$

2.2 | L^2-DG method with consistent stabilisation terms

The first DG method investigated in this work is an $\mathbf{L}^2$-based DG incompressible Navier-Stokes solver characterised by consistent divergence and continuity penalty terms acting as stabilisation terms for improved mass conservation and energy stability. This method is based on the formulation proposed in recent works by Fehn et al.[19,28] Both velocity and pressure are approximated by discontinuous functions, where the polynomial space for the velocity is one degree higher than for the pressure for reasons of discrete inf-sup stability:

$$V_h = \mathbf{Q}_k, \quad Q_h = \mathbb{Q}_{k-1} \cap L_0^2(\Omega).$$

To obtain the contribution of boundary face integrals from the weak forms given below, Equation (2) holds where the exterior weighting function is simply set to zero on boundary facets ($v_h^+ = \mathbf{0}$, $\nabla v_h^+ \cdot n = \mathbf{0}$, $q_h^+ = 0$). Homogeneous Dirichlet BCs are prescribed according to the mirror principle $u_h^+ = -u_h^-$ (with $\nabla u_h^+ \cdot n = \nabla u_h^- \cdot n$ for the velocity gradient and $p_h^+ = p_h^-$ for the pressure). The space-semidiscrete weak formulation of the incompressible Navier-Stokes equations (1.2) reads as follows:

$$\begin{cases} \text{Find}(u_h, p_h) : (0, T] \to V_h \times Q_h \text{ with } u_h(0) = u_{0h} \text{ s.t.}, \forall (v_h, q_h) \in V_h \times Q_h, & (5a) \\ (\partial_t u_h, v_h) + \nu a_h(u_h, v_h) + c_h(u_h; u_h, v_h) + j_h(u_h, v_h) + g_h(p_h, v_h) + d_h(u_h, q_h) = (f, v_h). & (5b) \end{cases}$$

The DG discretisation of the viscous term is based on the symmetric interior penalty (SIP) Galerkin method

$$a_h(u_h, v_h) = \int_\Omega \nabla_h u_h : \nabla_h v_h \mathrm{d}x - \sum_{F \in \mathcal{F}_h} \int_F (\{\!\{\nabla u_h\}\!\} \cdot n) \cdot [\![v_h]\!] \mathrm{d}s$$

$$- \sum_{F \in \mathcal{F}_h} \int_F [\![u_h]\!] \cdot (\{\!\{\nabla v_h\}\!\} \cdot n) \mathrm{d}s + \sum_{F \in \mathcal{F}_h} \int_F \tau [\![u_h]\!] \cdot [\![v_h]\!] \mathrm{d}s, \tag{6}$$

where the penalty parameter τ is

$$\tau = \begin{cases} \max(\tau_{K^-}, \tau_{K^+}), & \text{if facet } F \in \mathcal{F}_h^i, \\ \tau_K, & \text{if facet } F \in \mathcal{F}_h^\partial, \end{cases} \tag{7}$$

using the definition according to Hillewaert[29] for hexahedral elements

$$\tau_K = (k + 1)^2 \frac{\left|\partial K \cap \mathcal{F}_h^i\right|_{d-1}/2 + \left|\partial K \cap \mathcal{F}_h^\partial\right|_{d-1}}{|K_d|}. \tag{8}$$

Here, $|\cdot|_d$ denotes the d-dimensional Lebesgue measure. For the convective term, the local Lax-Friedrichs (LF) flux is applied:

$$c_h(w_h; u_h, v_h) = -\int_\Omega (u_h \otimes w_h) : \nabla_h v_h \mathrm{d}x + \sum_{F \in \mathcal{F}_h} \int_F \left(\{\!\{u_h \otimes w_h\}\!\} \cdot n + \frac{\Lambda(w_h)}{2} [\![u_n]\!] \right) \cdot [\![v_h]\!] \mathrm{d}s, \tag{9}$$

where $\Lambda(\boldsymbol{w}_h) = \max(2|\boldsymbol{w}_h^- \cdot \boldsymbol{n}|, 2|\boldsymbol{w}_h^+ \cdot \boldsymbol{n}|)$. Integration by parts of the velocity-pressure coupling terms along with central numerical flux functions yields

$$g_h(p_h, \boldsymbol{v}_h) = -\int_\Omega p_h(\nabla_h \cdot \boldsymbol{v}_h)\mathrm{d}\boldsymbol{x} + \sum_{F \in \mathcal{F}_h}\int_F \{\!\{p_h\}\!\}(\llbracket\boldsymbol{v}_h\rrbracket \cdot \boldsymbol{n})\mathrm{d}\boldsymbol{s}, \tag{10}$$

$$d_h(\boldsymbol{u}_h, q_h) = \int_\Omega \boldsymbol{u}_h \cdot \nabla_h q_h \mathrm{d}\boldsymbol{x} - \sum_{F \in \mathcal{F}_h}\int_F \llbracket q_h \rrbracket(\{\!\{\boldsymbol{u}_h\}\!\} \cdot \boldsymbol{n})\mathrm{d}\boldsymbol{s}. \tag{11}$$

The pressure gradient term and velocity divergence term are implemented in the so-called weak formulation of DG methods,[8] and it holds $g_h(p_h, \boldsymbol{u}_h) = d_h(\boldsymbol{u}_h, p_h)$ in case of exact numerical quadrature; cf also the paragraph on the numerical quadrature below. The consistent stabilisation term $j_h = j_{\mathrm{div},h} + j_{\mathrm{conti},h}$ is composed of a divergence penalty term $j_{\mathrm{div},h}$ and a continuity penalty term $j_{\mathrm{conti},h}$, which are given as[19]

$$j_{\mathrm{div},h}(\boldsymbol{u}_h, \boldsymbol{v}_h) = \int_\Omega \tau_{\mathrm{D}}(\nabla_h \cdot \boldsymbol{u}_h)(\nabla_h \cdot \boldsymbol{v}_h)\mathrm{d}\boldsymbol{x}, \tag{12}$$

$$j_{\mathrm{conti},h}(\boldsymbol{u}_h, \boldsymbol{v}_h) = \sum_{F \in \mathcal{F}_h^i}\int_F \{\!\{\tau_{\mathrm{C}}\}\!\}(\llbracket\boldsymbol{u}_h\rrbracket \cdot \boldsymbol{n})(\llbracket\boldsymbol{v}_h\rrbracket \cdot \boldsymbol{n})\mathrm{d}\boldsymbol{s}, \tag{13}$$

where the penalty parameters of the divergence and continuity penalty terms are defined as $\tau_{\mathrm{D}} = \zeta\overline{\|\boldsymbol{u}_h^{\mathrm{ex}}\|}\frac{h}{k+1}$ and $\tau_{\mathrm{C}} = \zeta\overline{\|\boldsymbol{u}_h^{\mathrm{ex}}\|}$, respectively. Here, $\overline{\|\boldsymbol{u}_h^{\mathrm{ex}}\|}$ is the magnitude of the velocity vector averaged over the element volume and $(\cdot)^{\mathrm{ex}}$ indicates that an extrapolation of the velocity field from previous instants of time is used in the time-discrete setting, resulting in a weak formulation j_h that is linear in $\boldsymbol{u}_h$. This definition of the penalty parameters originates from a dimensional analysis ensuring that all terms in the weak formulation (5) have consistent physical units. It was shown by means of numerical investigation in the work of Fehn et al[19] that with this definition of the penalty parameters, a default value of $\zeta = 1$ ensures robustness for underresolved flows in the sense that stability has been demonstrated for a wide range of Reynolds numbers (and in the inviscid limit), as well as for a wide range of the spatial resolution parameters h, k.

Numerical quadrature

The terms in the weak formulation are integrated numerically with Gaussian quadrature and $k + 1$ quadrature points per coordinate direction, except for the convective term c_h where an over-integration strategy with $\lceil\frac{3k+1}{2}\rceil$ quadrature points is used due to the quadratic nonlinearities of the convective term. On affine element geometries, this setup ensures exact integration. Exact integration allows to apply integration by parts to the integrals in $c_h(\cdot, \cdot), d_h(\cdot, \cdot), g_h(\cdot, \cdot)$, which yields $g_h(p_h, \boldsymbol{u}_h) = d_h(\boldsymbol{u}_h, p_h)$ and $c_h(\boldsymbol{w}_h; \boldsymbol{u}_h, \boldsymbol{u}_h) \geq 0$ for divergence-free and normal-continuous velocities $\boldsymbol{w}_h$ (and suitable BCs) and arbitrary $\boldsymbol{u}_h \in \boldsymbol{V}_h, p_h \in Q_h$.

Time integration and iterative solution of linear systems of equations

Discretisation in time is based on the backward differentiation formula (BDF) time integration method using a second-order accurate formulation (BDF2). The convective term is formulated explicitly in time (using a second-order accurate extrapolation scheme), which results in a CFL-type restriction of the time step size, $\Delta t = \frac{\mathrm{CFL}}{k^{1.5}}\frac{h_{\min}}{\|\boldsymbol{u}\|_{\max}}$, see the work of Fehn et al[19] for details on the time discretisation. In the following, we use CFL = 0.125 with $\|\boldsymbol{u}\|_{\max} = 1$ for the Taylor-Green vortex problem studied in Section 3, and CFL numbers in the range CFL = 0.2 − 1.0 (depending on the spatial resolution used) with $\|\boldsymbol{u}\|_{\max} = 20.1$ (maximum mean velocity known from DNS reference data) for the turbulent channel flow results shown in Section 4. It was found that temporal discretisation errors are negligible for such a setup where the time step size is smaller than the critical time step arising from the CFL condition. For the fully discrete problem, an unsteady Stokes problem has to be solved in each time step. This coupled system of equations including the stabilisation terms is solved iteratively using the generalized minimal residual method with block preconditioning. Other solution strategies for time integration such as projection methods can be used as well to obtain computationally efficient solution algorithms and have been described in the work of Fehn et al[19] in the context of the present $\boldsymbol{L}^2$-based DG discretisation. The solver is stopped once the l_2 norm of the residual has been reduced by 10^6 compared to the initial residual norm (where the initial solution is an extrapolation of the solution from previous time steps), or if the absolute value of the discrete l_2 norm of the residual goes below 10^{-12}. All solver components rely on fast matrix-free operator evaluation

(exploiting the so-called sum-factorisation technique) to achieve optimal computational complexity, and we refer to the work of Fehn et al[30] for a documentation of the computational efficiency of this L^2-based DG solver.

2.3 | Divergence-free H(div)-HDG method

For the H(div)-based method, we basically rely on the method of Lehrenfeld and Schöberl[18]; the underlying FE spaces are given by

$$\boldsymbol{V}_h = \mathbb{R}\mathbb{T}_{[k]}, \quad Q_h = \mathbb{Q}_k \cap L_0^2(\Omega), \quad \widehat{\boldsymbol{V}}_h = \left\{\widehat{\boldsymbol{v}}_h \in \boldsymbol{L}^2(\mathcal{F}_h) : \widehat{\boldsymbol{v}}_h|_F \in \boldsymbol{Q}_{k-1}(F); \widehat{\boldsymbol{v}}_h \cdot \boldsymbol{n}|_F = 0, \forall F \in \mathcal{F}_h; \widehat{\boldsymbol{v}}_h|_{\Gamma_0} = \boldsymbol{0}\right\}, \quad (14)$$

where $\boldsymbol{V}_h$ and Q_h are the discrete velocity and pressure spaces, respectively, and $\widehat{\boldsymbol{V}}_h$ is the hybrid facet space, which contains discrete tangential velocities on the skeleton of $\mathcal{T}_h$. Note that, thus, the global space $\boldsymbol{V}_h$ has order k approximation properties, but in general, locally, the discrete velocity can even be a polynomial of order $k+1$. To obtain the contribution of boundary face integrals on Dirichlet boundary facets $F \in \mathcal{F}_h^\partial$ with homogeneous boundary values, we set $\{\!\{\phi\}\!\}_F = \phi$ and $[\![\phi]\!]_F = \phi$ for the average and jump operators introduced in Equation (2).

Remark 1. To remove some degrees of freedom (DOFs) for the velocity and all pressure unknowns except for piecewise constants, we exploit the a priori knowledge that $\nabla \cdot \boldsymbol{u}_h = 0$; cf Remark 1[18] and Section 2.2.4.2[31] in the works of Lehrenfeld and Schöberl. This can be achieved with a smart choice of basis functions for $\boldsymbol{V}_h$ based on an exact sequence property (de Rham complex) on the discrete level; cf the works of Schöberl and Zaglmayr.[32,33]

The space-semidiscrete weak formulation of the hybridised H(div)-DG method for (1) reads as follows:

$$\begin{cases} \text{Find } (\boldsymbol{u}_h, \widehat{\boldsymbol{u}}_h, p_h) : (0, T] \to \boldsymbol{V}_h \times \widehat{\boldsymbol{V}}_h \times Q_h \text{ with } \boldsymbol{u}_h(0) = \boldsymbol{u}_h \text{ s.t.}, \forall (\boldsymbol{v}_h, \widehat{\boldsymbol{v}}_h, q_h) \in \boldsymbol{V}_h \times \widehat{\boldsymbol{V}}_h \times Q_h, & (15a) \\ (\partial_t \boldsymbol{u}_h, \boldsymbol{v}_h) + \nu a_h((\boldsymbol{u}_h, \boldsymbol{u}_h), (\boldsymbol{v}_h, \widehat{\boldsymbol{v}}_h)) + c_h(\boldsymbol{u}_h; \boldsymbol{u}_h, \boldsymbol{v}_h) + b_h(\boldsymbol{v}_h, p_h) + b_h(\boldsymbol{u}_h, q_h) = (\boldsymbol{f}, \boldsymbol{v}_h). & (15b) \end{cases}$$

Let us now specify the various terms in (15). To discretise the viscous term, a hybridised variant of the SIP method is used. The corresponding bilinear form, with $\sigma_K = 6(k+1)^2$, is given by

$$a_h((\boldsymbol{u}_h, \widehat{\boldsymbol{u}}_h), (\boldsymbol{v}_h, \widehat{\boldsymbol{v}}_h)) = \sum_{K \in \mathcal{T}_h} \left[\int_K \nabla \boldsymbol{u}_h : \nabla \boldsymbol{v}_h \mathrm{d}\boldsymbol{x} - \int_{\partial K} (\nabla \boldsymbol{u}_h \cdot \boldsymbol{n}) \cdot [\![\![\boldsymbol{v}_h]\!]\!] \mathrm{d}\boldsymbol{s} \right.$$
$$\left. - \int_{\partial K} [\![\![\boldsymbol{u}_h]\!]\!] \cdot (\nabla \boldsymbol{v}_h \cdot \boldsymbol{n}) \mathrm{d}\boldsymbol{s} + \int_{\partial K} \frac{\sigma_K}{h_K(\boldsymbol{s})} [\![\![\boldsymbol{u}_h]\!]\!] \cdot [\![\![\boldsymbol{v}_h]\!]\!] \mathrm{d}\boldsymbol{s} \right], \quad (16)$$

where $h_K(\boldsymbol{s}) = |K|_d/|F|_{d-1}$ with $\boldsymbol{s} \in F \subset \partial K$. Here, in contrast to $[\![\cdot]\!]$, the operator $[\![\![\cdot]\!]\!]$ represents the tangential jump between cell and facet velocity projected onto the space $\boldsymbol{Q}_{k-1}(F)$; that is, using the projection onto the tangent plane $\boldsymbol{v}_t = \boldsymbol{v} - (\boldsymbol{v} \cdot \boldsymbol{n})\boldsymbol{n}$,

$$[\![\![\boldsymbol{v}]\!]\!] = \left(\Pi_{\mathcal{F}_h}^{k-1} \boldsymbol{v} - \widehat{\boldsymbol{v}} \right)_t = \Pi_{\mathcal{F}_h}^{k-1} \boldsymbol{v}_t - \widehat{\boldsymbol{v}},$$

where $\Pi_{\mathcal{F}_h}^{k-1}$ is the facet-wise L^2-orthogonal projection onto $\boldsymbol{Q}_{k-1}(F)$. The choice $\sigma_K = 6(k+1)^2$ for the SIP stabilisation parameter renders the SIP penalty scaling comparable to that of the L^2-based method. Due to H(div)-conformity, the pressure-velocity coupling is simply given by

$$b_h(\boldsymbol{u}_h, q_h) = -\int_\Omega q_h(\nabla \cdot \boldsymbol{u}_h) \mathrm{d}\boldsymbol{x}. \quad (17)$$

Here, the property $\nabla \cdot \boldsymbol{V}_h = Q_h$ allows to test with $q_h = \nabla \cdot \boldsymbol{u}_h$, which results in the fact that we are obtaining an exactly divergence-free discrete velocity solution because discretely divergence-free velocities are actually exactly divergence free; ie,

$$-\int_\Omega q_h(\nabla \cdot \boldsymbol{u}_h)\mathrm{d}\boldsymbol{x} = 0, \quad \forall q_h \in Q_h \quad \Longrightarrow \quad \int_\Omega |\nabla \cdot \boldsymbol{u}_h|^2 \mathrm{d}\boldsymbol{x} = 0 \quad \Longrightarrow \quad \nabla \cdot \boldsymbol{u}_h = 0. \quad (18)$$

Finally, for the nonlinear convection term, we use the form

$$c_h(\boldsymbol{w}_h; \boldsymbol{u}_h, \boldsymbol{v}_h) = \int_\Omega (\boldsymbol{w}_h \cdot \nabla_h)\boldsymbol{u}_h \cdot \boldsymbol{v}_h \mathrm{d}\boldsymbol{x} - \sum_{F \in \mathcal{F}_h^i} \int_F (\boldsymbol{w}_h \cdot \boldsymbol{n}) [\![\boldsymbol{u}_h]\!] \cdot \{\!\{\boldsymbol{v}_h\}\!\} \mathrm{d}\boldsymbol{s} + \sum_{F \in \mathcal{F}_h^i} \int_F \frac{\theta}{2} |\boldsymbol{w}_h \cdot \boldsymbol{n}| [\![\boldsymbol{u}_h]\!] \cdot [\![\boldsymbol{v}_h]\!] \mathrm{d}\boldsymbol{s}, \quad (19)$$

where the choice $\theta = 1$ leads to an upwind-stabilised formulation and $\theta = 0$ corresponds to a method without any kind of convection stabilisation. Note that in the convective form (19), instead of the hybrid jump $[\![\![\cdot]\!]\!]$, the "normal" DG jump $[\![\cdot]\!]$ between neighbouring cell velocities is used.

Numerical quadrature

In the realisation of the bilinear forms, numerical quadrature is applied so that on affine linear elements, numerical integration is performed exactly. Due to polynomials of degree $k + 1$ in $\mathbb{RT}_{[k]}$, this requires $k + 2$ Gaussian quadrature points in each direction for the in integrals $a_h(\cdot, \cdot)$; for $b_h(\cdot, \cdot)$ $k + 1$ Gaussian quadrature points suffice as $\nabla \cdot \mathbb{RT}_{[k]} = \mathbb{Q}_k$, whereas due to the nonlinearity in $c_h(\cdot; \cdot, \cdot)$, $\lceil \frac{3k+3}{2} \rceil$ points are required.

Time integration and linear solvers

Concerning time integration, we use the second-order Runge-Kutta variant ARS(2,2,2) of the implicit-explicit (IMEX) method introduced in the work of Ascher et al.[34] Here, the pressure-velocity coupling $b_h(\cdot)$ and the viscosity term $a_h(\cdot)$ is always treated implicitly to maintain the exactly divergence-free property of the method, and the convection term $c_h(\cdot)$ is always treated explicitly. Therefore, only linear systems have to be solved in each time step. We use static condensation to eliminate element-local unknowns and solve the linear systems involving the Schur complement with a balancing domain decomposition by constraints (BDDC)-preconditioned[35] CG solver. As the time step size for the studies of the Taylor-Green vortex problem in Section 3, we take a fixed size of $\Delta t = 10^{-2}$ in all simulations except for the simulations with the LF flux, where the time step size is reduced to $\Delta t = 5 \cdot 10^{-3}$ for stability reasons. For the turbulent channel flow in Section 4, a fixed time step size of $\Delta t = 10^{-3}$ is used for all simulations. In further tests, we have observed that the temporal discretisation errors are negligible for both setups with the chosen time step sizes, ie, a further reduction of the time step size did not change the results significantly.

2.4 | Comparison of approximation spaces under the divergence constraint

For both the $\mathbb{Q}_k$- and the $\mathbb{RT}_{[k]}$-based discretisations, the global number of velocity unknowns on a periodic Cartesian mesh with N elements per coordinate direction in d dimensions is given as

$$N_{\mathrm{DOFs},\boldsymbol{u}}(N, k) = N^d d(k + 1)^d. \quad (20)$$

We note that the global number of unknowns for both velocity spaces is the same despite the fact that the local spaces are different according to (3) and (4).

To characterise the approximation properties of the two methods, it is useful to incorporate the constraints acting on the velocity space. For the $\boldsymbol{H}(\mathrm{div})$ method, the divergence-free condition $b_h(\boldsymbol{u}_h, q_h) = 0$ constrains $(k + 1)^d$ out of the $d(k + 1)^d$ velocity DOFs per element. For the $\boldsymbol{L}^2$-based method, the limit $\tau_\mathrm{C}, \tau_\mathrm{D} \to \infty$ implies continuity in the normal direction and pointwise divergence-free solutions. Both imply constraints on the velocity. In the Appendix, we give a derivation and characterisation of the limit method that the solution of the $\boldsymbol{L}^2$-based method converges to as $\tau_\mathrm{C}, \tau_\mathrm{D} \to \infty$. Normal-continuity ($\tau_\mathrm{C} \to \infty$) corresponds to $(k+1)^{d-1}$ DOFs per coordinate direction, ie, $d(k+1)^{(d-1)}$ DOFs per element, whereas the pointwise divergence-free condition ($\tau_\mathrm{D} \to \infty$) corresponds to $(k+1)^d - 1$ DOFs (cf $\dim(Q_h^{\mathrm{vol}})$ in the Appendix). Furthermore, it can be easily seen that pointwise divergence-free velocities imply the weak divergence-free condition, rendering the weak divergence-free condition $d_h(\boldsymbol{u}_h, q_h) = 0$ in (5) superfluous. Taking these results together and denoting by $N_{\mathrm{DOFs},\boldsymbol{u}}^{\mathrm{L2,c}}$ and $N_{\mathrm{DOFs},\boldsymbol{u}}^{\mathrm{Hdiv,c}}$ the DOFs that remain after taking the constraints into account, we obtain

$$N_{\mathrm{DOFs},\boldsymbol{u}}^{\mathrm{L2,c}}(N, k) = N^d((d - 1)(k + 1)^d - d(k + 1)^{d-1} + 1), \quad (21)$$

for the DG method with infinite stabilisation. Moreover, one obtains the relation

$$N_{\mathrm{DOFs},\boldsymbol{u}}^{\mathrm{L2,c}}(N, k) < N_{\mathrm{DOFs},\boldsymbol{u}}^{\mathrm{Hdiv,c}}(N, k) = N^d(d - 1)(k + 1)^d < N_{\mathrm{DOFs},\boldsymbol{u}}^{\mathrm{L2,c}}(N, k + 1). \quad (22)$$

As a consequence, it makes sense to consider the bracket $\{k, k+1\}$ as polynomial degrees for the L^2-DG method to compare against $\mathbb{RT}_{[k]}$. In the setting $\tau_C < \infty$ and $\tau_D < \infty$, the velocity approximation has some freedom left that could in principle increase the solution quality and move the L^2-DG method of degree k closer to $\mathbb{RT}_{[k]}$ (and beyond). This behaviour again motivates a closer study of the two adjacent polynomial degrees with respect to Raviart-Thomas. We argue that the added solution quality by $\tau_C < \infty$ and $\tau_D < \infty$ can be expected to be minor because the penalty terms are necessary to render the approximation stable, dominating over "spurious" contributions.

2.5 | Mass conservation

To highlight the similarities and differences of the two discretisation approaches in terms of discrete mass conservation, it is instructive to reformulate the discrete continuity equation (11) for the L^2-based approach

$$d_h(\boldsymbol{u}_h, q_h) = -\int_\Omega q_h \nabla_h \cdot \boldsymbol{u}_h \mathrm{d}\boldsymbol{x} + \sum_{F \in \mathcal{F}_h} \int_F \{\!\{q_h\}\!\}([\![\boldsymbol{u}_h]\!] \cdot \boldsymbol{n})\mathrm{d}\boldsymbol{s} = 0.$$

For the $\boldsymbol{H}(\mathrm{div})$-conforming approach, the velocity is normal-continuous across interior facets so that the second term in the above equation vanishes; see Equation (17). By using Raviart-Thomas function spaces, the velocity is exactly divergence-free for the $\boldsymbol{H}(\mathrm{div})$-conforming approach; see Equation (18). Accordingly, it can be seen as a specialisation of the L^2-based approach with additional restrictions for the function spaces of velocity/pressure. In case of the L^2-based approach, these restrictions are not set explicitly; instead, these restrictions are enforced weakly by the use of standard function spaces along with additional stabilisation terms; see Equations (12) and (13).

2.6 | Energy balance and dissipation mechanisms

For a unified presentation below, one can formally write $\hat{\boldsymbol{u}}_h = \hat{\boldsymbol{u}}_h(\boldsymbol{u}_h)$ for the HDG method and obtain a corresponding DG bilinear form $a_h((\boldsymbol{u}_h, \hat{\boldsymbol{u}}_h), (\boldsymbol{u}_h, \hat{\boldsymbol{u}}_h)) \to a_h(\boldsymbol{u}_h, \boldsymbol{u}_h)$ (with abuse of notation), which only depends on the usual velocity variable.

Assuming that $\boldsymbol{f} \equiv \boldsymbol{0}$, testing symmetrically in (5) (respectively in (15) with $\hat{\boldsymbol{u}}_h = \hat{\boldsymbol{u}}_h(\boldsymbol{u}_h)$) leads to the discrete energy balance

$$\underbrace{\frac{1}{2}\frac{\mathrm{d}}{\mathrm{d}t}\|\boldsymbol{u}_h\|_{L^2}^2}_{\text{dt\_ekin}} + \underbrace{\nu a_h(\boldsymbol{u}_h, \boldsymbol{u}_h)}_{\text{visc\_diss}} + \underbrace{c_h(\boldsymbol{u}_h; \boldsymbol{u}_h, \boldsymbol{u}_h)}_{\text{num\_diss\_conv}} + \underbrace{j_h(\boldsymbol{u}_h, \boldsymbol{u}_h)}_{\text{num\_diss\_div + num\_diss\_conti}} = 0. \tag{23}$$

Here, we introduced notations for the total energy dissipation `dt_ekin` that splits into the contributions `visc_diss` for the viscous dissipation, `num_diss_conv` for the (numerical) dissipation of the convection treatment, and `num_diss_div` and `num_diss_conti` for the (numerical) dissipation stemming from the stabilisation forms in the L^2-based approach. Whenever $c_h(\boldsymbol{u}_h; \boldsymbol{u}_h, \boldsymbol{u}_h) > 0$ or $j_h(\boldsymbol{u}_h, \boldsymbol{u}_h) > 0$, the corresponding nonviscosity-related mechanism can clearly be characterised as artificial or numerical dissipation because there is no counterpart in the continuous Navier-Stokes model (1).

It may occur for some discretisations that $c_h(\boldsymbol{u}_h; \boldsymbol{u}_h, \boldsymbol{u}_h)$ is sign-indefinite, which may lead to an artificial increase of kinetic energy. This is often compensated by additional artificial or numerical dissipation in $a_h(\boldsymbol{u}_h, \boldsymbol{u}_h)$ and/or $j_h(\boldsymbol{u}_h, \boldsymbol{u}_h)$. To illustrate this, let us explicitly state the discrete energy balance for the L^2-based approach with local LF flux under the assumption of periodic BCs and vanishing body forces (see the work of Fehn et al[19] for details):

$$\frac{1}{2}\frac{\mathrm{d}}{\mathrm{d}t}\|\boldsymbol{u}_h\|_{L^2}^2 = -\nu a_h(\boldsymbol{u}_h, \boldsymbol{u}_h) - \frac{1}{2}\int_\Omega \nabla_h \cdot \boldsymbol{u}_h(\boldsymbol{u}_h \cdot \boldsymbol{u}_h)\mathrm{d}\boldsymbol{x} + \frac{1}{2}\sum_{F \in \mathcal{F}_h} \int_F [\![\boldsymbol{u}_h]\!] \cdot \boldsymbol{n}\left(\boldsymbol{u}_h^- \cdot \boldsymbol{u}_h^+\right)\mathrm{d}\boldsymbol{s}$$

$$- \sum_{F \in \mathcal{F}_h} \int_F [\![\boldsymbol{u}_h]\!] \cdot \frac{\Lambda}{2}[\![\boldsymbol{u}_h]\!]\mathrm{d}\boldsymbol{s} - j_{\mathrm{div},h}(\boldsymbol{u}_h, \boldsymbol{u}_h) - j_{\mathrm{conti},h}(\boldsymbol{u}_h, \boldsymbol{u}_h). \tag{24}$$

Equation (24) highlights the second main difference between the L^2-based and $\boldsymbol{H}(\mathrm{div})$-based approaches considered in this work. The last two terms in the first row of the aforementioned equation (belonging to the convective term $c_h(\boldsymbol{u}_h; \boldsymbol{u}_h, \boldsymbol{u}_h)$) are sign indefinite for the L^2-based approach. Note that similar terms would also arise for other formulations of the convective term, ie, the convective formulation with upwind flux discretisation. In case of $\boldsymbol{H}(\mathrm{div})$-conforming function spaces with exactly divergence-free velocity, these two terms add up to zero by definition and only the

sign-definite stabilisation term of the convective term contributes to the discrete energy evolution. In case of the stabilised L^2-based approach, the aim of the divergence and continuity penalty terms (which are positive-semidefinite, ie, they exhibit a purely dissipative character) is to compensate sign-indefinite terms of the convective term.

Let us also mention that, in DG discretisations, $a_h(\boldsymbol{u}_h, \boldsymbol{u}_h)$ consists of two types of viscous dissipation: physical and numerical dissipation. Despite its frequent use in the literature, the broken gradient norm $\|\nabla_h \boldsymbol{u}_h\|_{L^2}^2$ is not a good measure for the physical dissipation in underresolved situations as the corresponding numerical dissipation $a_h(\boldsymbol{u}_h, \boldsymbol{u}_h) - \|\nabla_h \boldsymbol{u}_h\|_{L^2}^2$ is in general not sign definite (even if the viscosity bilinear form is coercive). We refer to the work of Lehrenfeld et al[36] for an alternative evaluation of physical and numerical dissipation for DG methods. In the remainder of this work, we circumvent this problem by only considering the sum of physical and numerical viscous dissipation.

3 | DECAYING HOMOGENEOUS ISOTROPIC TURBULENCE: THE 3D TAYLOR-GREEN VORTEX PROBLEM

In the box $\Omega = (0, 2\pi)^3$, equipped with periodic BCs on all faces, consider the case $\boldsymbol{f} \equiv \boldsymbol{0}$ and the space-periodic initial condition[37,38]

$$\boldsymbol{u}_0(\boldsymbol{x}) = \begin{pmatrix} \cos(x_1)\sin(x_2)\sin(x_3) \\ -\sin(x_1)\cos(x_2)\sin(x_3) \\ 0 \end{pmatrix}. \tag{25}$$

This initial condition is imposed and the resulting flow is monitored over time. The resulting Taylor-Green vortex (TGV) problem is possibly the easiest flow system for which one can observe the key physical mechanisms inherent to turbulence: transition, vortex roll-up, 3D vortex stretching and interaction, and finally, in the viscous case, molecular energy dissipation. As done frequently in the literature, simulations are run for $Re = \nu^{-1} = 1600$ until $T = 20$; see other works.[2,3,17] All simulations are performed on structured hexahedral meshes consisting of N^3 cubes and polynomial order k. The accuracy of the underresolved simulations performed here is assessed by comparing the results to accurate direct numerical simulation (DNS) reference data from the work of Fehn et al[30] with $k = 7$ and $N = 128$.

3.1 | Comparison of methods

At first, we want to compare the results obtained by the L^2 method (5) and the $\boldsymbol{H}(\mathrm{div})$ method (15) on the basis of the evolution of kinetic energy $\mathcal{K}(\boldsymbol{u}_h) = \frac{1}{2|\Omega|}\|\boldsymbol{u}_h\|_{L^2}^2$ and (negative) total kinetic energy dissipation rate $-\partial_t \mathcal{K}(\boldsymbol{u}_h)$ for the high-order choice $k = 8$; see also (23). In Figure 1 (top), both quantities are displayed on a sequence of meshes $N \in \{4, 8, 16\}$, which correspond to strongly underresolved, moderately resolved, and essentially resolved simulations. The most important observation is that the results are, in general, not significantly different with respect to the particular quantity. While on the coarse meshes, slight differences between L^2 and $\boldsymbol{H}(\mathrm{div})$ results can be observed; they are basically identical on the finest mesh. Moreover, it is comparably easy to have a good approximation for the kinetic energy, even in a strongly underresolved situation, but to capture the total dissipation rate, significantly more resolution is necessary. In the remainder of this section, we will thus not discuss the evolution of kinetic energy anymore but focus on the more demanding energy dissipation rate.

Let $E(\kappa)$ be defined as the amount of kinetic energy concentrated in the wave number vector $\boldsymbol{\kappa} \in \mathbb{R}^n$ with wave number $\kappa := |\boldsymbol{\kappa}|$, the energy spectrum. Concerning the distribution of $E(\kappa)$ over different wave numbers κ, Figure 1 (bottom) shows the kinetic energy spectrum at $t = 10$ (shortly after the dissipation peak) for both methods under h-refinement. The most important observation is that the L^2- and the $\boldsymbol{H}(\mathrm{div})$-based results are very similar to each other. For the considered Reynolds number and spatial resolutions, the flow exhibits only a comparably small inertial range where the Kolmogorov $-5/3$ rule can be observed. However, the inertial range becomes larger for finer resolutions for which viscous effects are better resolved and, hence, more energy gets dissipated in the fine scales. On the coarse mesh, the resolution is not sufficient to dissipate the fine-scale features on the molecular level.

Next, we refine the above analysis by comparing the $\boldsymbol{H}(\mathrm{div})$-conforming method of degree k with the L^2-conforming method of degrees k and $k + 1$ so that the $\boldsymbol{H}(\mathrm{div})$-conforming method is located in between the L^2-conforming method in terms of DOFs; see Section 2. We show results of this comparison in Figure 2 for $k = 4$ in the top row and for $k = 8$ in the bottom row. For $k = 4$, the L^2-conforming method of degree k appears to be of similar accuracy or slightly more accurate than the $\boldsymbol{H}(\mathrm{div})$-conforming method of degree k. For degree $k = 8$, the L^2-method and $\boldsymbol{H}(\mathrm{div})$-method of degree k also
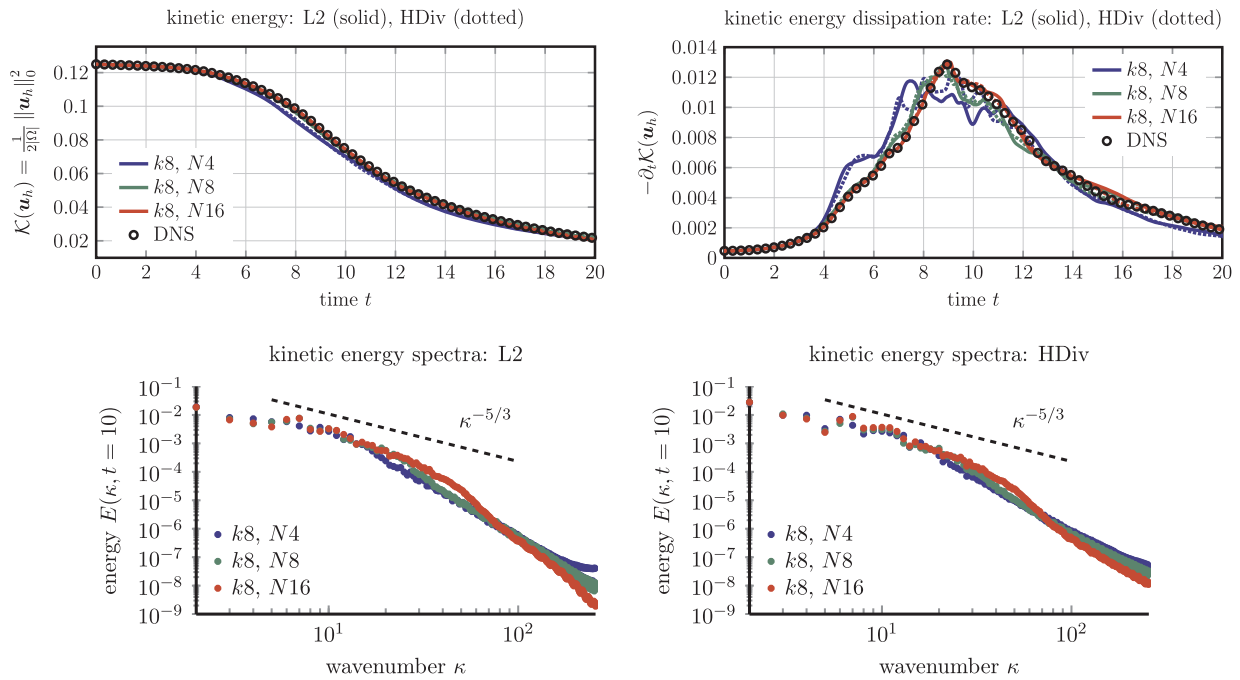
FIGURE 1 Comparisons of high-order ($k = 8$) L^2- and $\boldsymbol{H}$(div)-based simulations for the Taylor-Green vortex. Evolution of kinetic energy (top left), total kinetic energy dissipation rate (top right), and kinetic energy spectra (bottom) at $t = 10$ on different meshes with $N \in \{4, 8, 16\}$. DNS, direct numerical simulation [Colour figure can be viewed at wileyonlinelibrary.com]
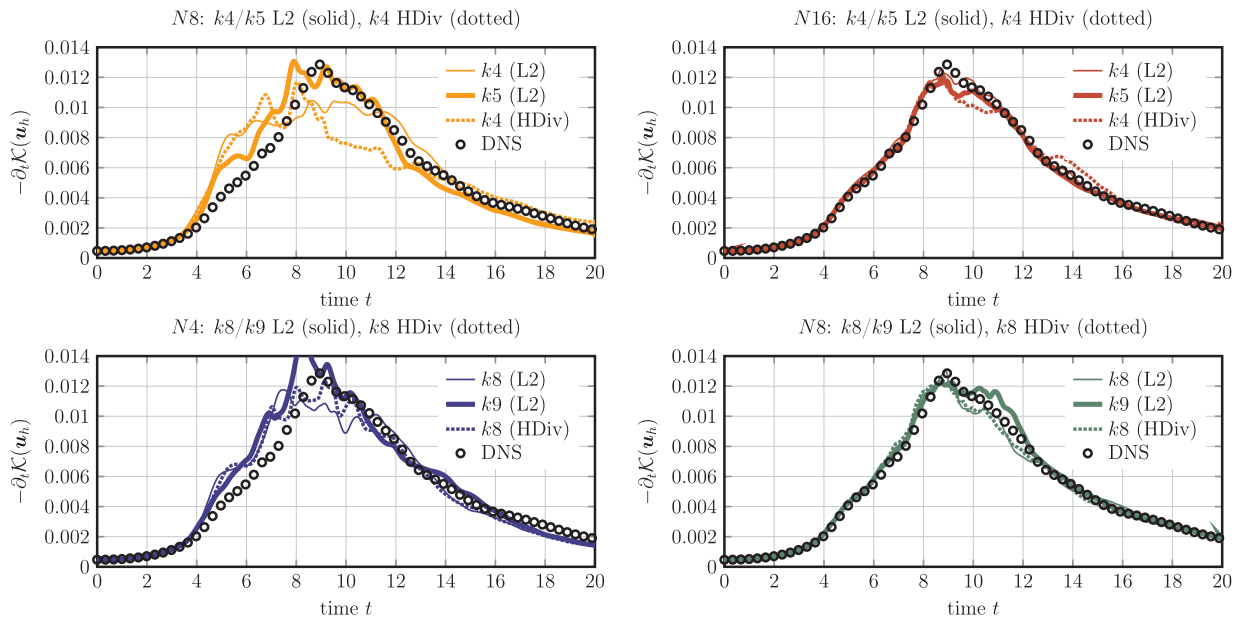

FIGURE 2 Comparison of the $\boldsymbol{H}$(div)-based method of order k with the $\boldsymbol{L}^2$-based method of order k and $k + 1$. Shown are the respective total dissipation rates. DNS, direct numerical simulation [Colour figure can be viewed at wileyonlinelibrary.com]

show similar accuracy, where the $\boldsymbol{H}$(div)-method tends to be slightly more accurate for the $N = 4$ case. The $\boldsymbol{L}^2$-conforming method of degree $k + 1$ seems to be the most accurate one in this comparison, but we want to emphasise that it is difficult to draw precise conclusions due to the wriggling behaviour of the kinetic energy dissipation rate, especially for the coarser resolutions shown in the left part of the figure. Overall, these results suggest that both methods provide a similar level of accuracy for the Taylor-Green vortex problem investigated here, if the function spaces for the individual methods are chosen in a way that they offer comparable resolution capabilities in terms of DOFs. For all following considerations,
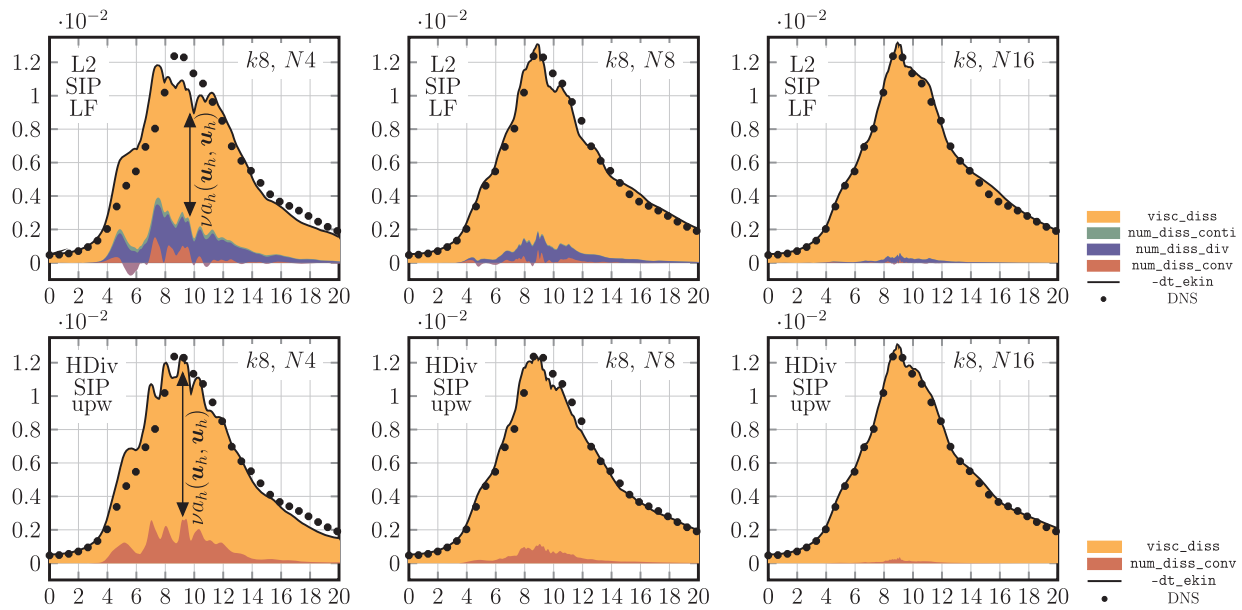
FIGURE 3 Dissipation mechanisms for $k = 8$ under h-refinement (considering meshes with $N \in \{4, 8, 16\}$). The abscissa shows the time t. The Lax-Friedrichs (LF) flux is used for the L^2 method and the upwind (upw) flux is used for the $H(\text{div})$ method. DNS, direct numerical simulation; SIP, symmetric interior penalty [Colour figure can be viewed at wileyonlinelibrary.com]

comparisons between the two methods are made on the basis of the same polynomial degree k, which also results in a fairly comparable number of DOFs according to Section 2.

3.2 | High-order dissipation mechanisms under h-refinement

In Figure 3, we consider a fixed order ($k = 8$) under h-refinement and display the sum of different contributions to the dissipation. Note that the plot is stacked in the sense that the solid line on the top of the orange area represents the sum of all the contributions below. For the L^2-based method (top row, with LF flux), there are four contributions (cf (23)), whereas there are only two (nonzero) contributions for the $H(\text{div})$-based method (bottom row, with upwind (upw) flux). One observes that the sign-indefinite convection parts in the L^2-based methods (where the corresponding part is negative, the area is coloured purple) are compensated by the stabilisation terms. For both methods, one observes that the dominant contribution in the dissipation stems from the viscosity, which eventually is the only relevant dissipation mechanism once the flow is sufficiently resolved, eg, $(k, N) = (8, 16)$.

3.3 | High-order vs low-order dissipation mechanisms for fixed underresolution

Up to now, we only considered a fixed polynomial order of $k = 8$ and did h-refinement procedures. This section is concerned with the question whether high-order computations are actually superior to lower-order ones, given that a comparably fair environment is provided. We already saw that whenever the resolution is sufficiently large, no differences between different methods can be observed anymore with the naked eye. Therefore, here, we restrict ourselves to the strongly underresolved situation $(k, N) = (2, 16), (4, 8), (8, 4)$ while we note that $N_{\text{DOFs},\boldsymbol{u}}(2, 16) > N_{\text{DOFs},\boldsymbol{u}}(4, 8) > N_{\text{DOFs},\boldsymbol{u}}(8, 4)$, cf (20), ie, the low-order methods have slightly more unknowns. In Figure 4, the evolution of the kinetic energy dissipation rate (`-dt_ekin`) shows the differences between high-order or low-order methods. We observe that the resolution improves when going to higher-order (although using slightly fewer unknowns). However, one can also see that the total dissipation of high-order methods is concentrated in the viscous term (`visc_diss`); cf (23). Consequently, in the strongly underresolved situation, low-order methods fundamentally rely on the numerical dissipation provided by stabilisation.

Summarising, we believe that high-order methods are more accurate than low-order methods for freely decaying turbulence problems such as the Taylor-Green vortex. Therefore, in the following, we will restrict ourselves to $k = 8$ and investigate the influence of different treatments of convection, viscosity, and divergence conformity.
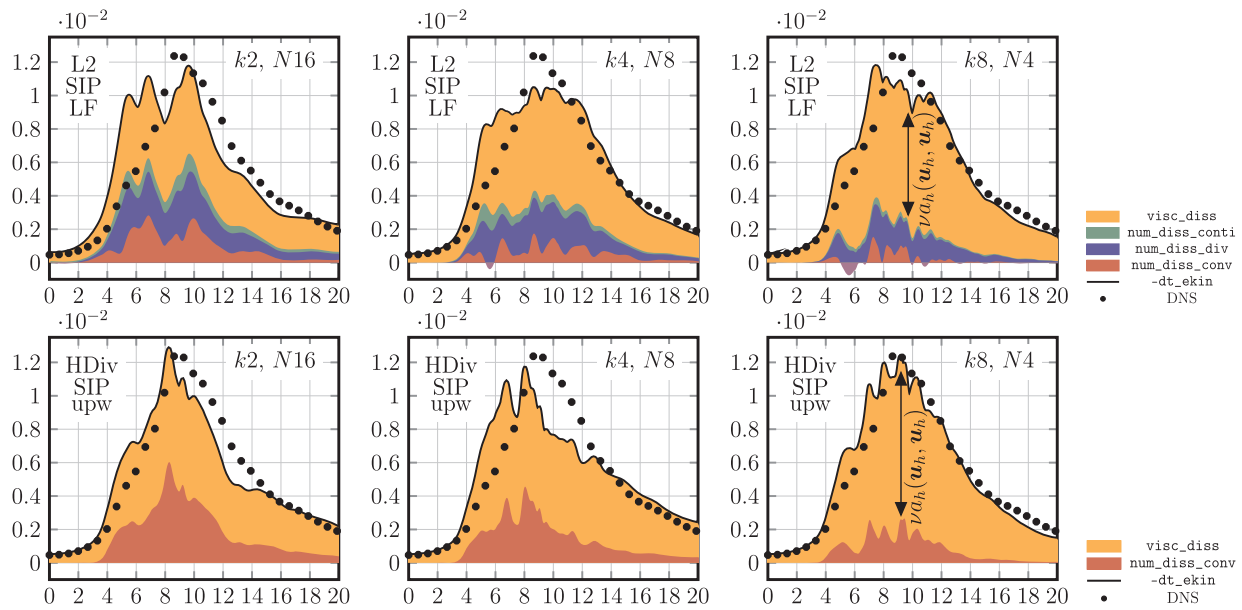
FIGURE 4 Dissipation mechanisms for fixed strong underresolution under k-refinement. The abscissa shows the time t. The Lax-Friedrichs (LF) flux is used for the L^2 method and the upwind (upw) flux is used for the H(div) method. DNS, direct numerical simulation; SIP, symmetric interior penalty [Colour figure can be viewed at wileyonlinelibrary.com]

3.4 | Influence of different treatments of convection, viscosity, and divergence conformity

In this section, we will investigate the influence of small changes to the considered methods. Namely, we compare different fluxes for the discrete convection term for both L^2 and H(div) method, investigate the impact of different viscosity treatments for the H(div) method and vary the penalisation parameter for the L^2 method. As the particular differences decrease with an increasing resolution, we will only consider the strongly and mildly underresolved cases $(k, N) = (8, 4), (8, 8)$.

3.4.1 | Different fluxes for the convection term

Let us briefly explain the different choices of the discrete convective term $c_h(\cdot; \cdot, \cdot)$, which are to be considered and compared here. The basic variants (also used for all computations above) are the LF form (9) for L^2 and the upwind form (19) with $\theta = 1$ for H(div). Now, we will also use the upwind form (19) with $\theta = 1$ and the central flux form (19) with $\theta = 0$ for the L^2 method, where a single-valued normal flux is ensured via $\mathbf{w}_h \cdot \mathbf{n} \rightarrow \{\!\{\mathbf{w}_h\}\!\} \cdot \mathbf{n}$. For the H(div) method, we additionally regard the performance of the central flux form (19) with $\theta = 0$, and the LF form (9), where $\Lambda(\mathbf{w}_h) = 2|\mathbf{w}_h \cdot \mathbf{n}|$. Note that for exactly divergence-free H(div) methods (and under exact numerical integration), the LF form simply emerges from (19) by choosing $\theta = 2$ and performing integration by parts of the corresponding volume term. Put differently, the difference between LF and upwinding for the H(div) method is solely a factor two in the positive semidefinite facet stabilisation term. On the other hand, for the L^2 method, integration by parts using a not exactly divergence-free convective velocity leads to a conceptually different discrete convection term.

The corresponding results for the stacked dissipation rates for $k = 4$, $N = 8$ simulations can be seen in Figure 5. Regarding the total dissipation rate (-dt_ekin), it does not seem to be possible to draw a conclusion as to whether one particular convective term is superior to the others. In fact, all variants result in a comparably accurate solution even in this strongly underresolved situation. Moreover, regarding the LF and upwinding H(div) results, one can observe that the amount of convective dissipation remains approximately the same regardless of which convection stabilisation is used. In particular, the additional factor two in the LF form does not lead to more numerical dissipation. Surprisingly, the third column shows that both the L^2- and the H(div)-based methods can even manage the case without convection stabilisation in form of upwinding or LF for the TGV problem. We conjecture that choosing a seemingly less dissipative convection term is simply counterbalanced by other (numerical) dissipation mechanisms that are present in the treatment of the viscosity[36] and the stabilisation terms of the L^2-based method. Let us also mention that viscous dissipation does not seem to be necessary for stability. In Section 9.1.3 of the work of Schroeder,[20] the inviscid version $\nu = 0$ (ie, $Re = \infty$) of the same Taylor-Green vortex example has been simulated with the H(div)-based method using a central flux for the convection without the need for any additional numerical dissipation in the spatial discretisation. For the L^2-based method, the use
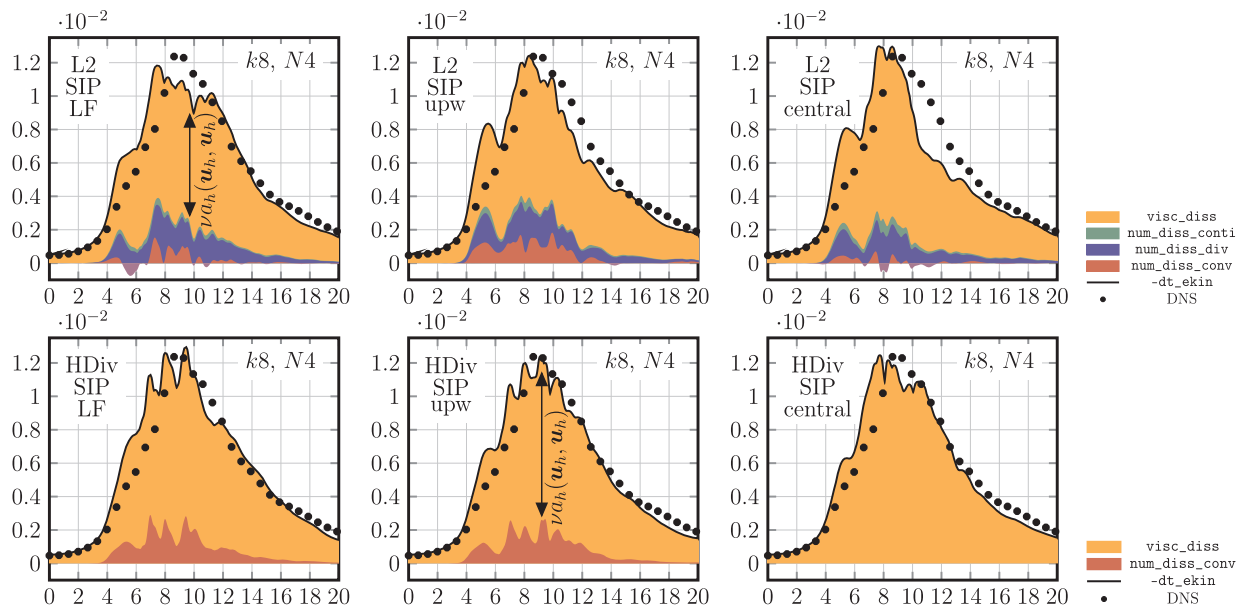
FIGURE 5 Dissipation mechanisms for different convection fluxes for $k = 8$, $N = 4$. Stacked dissipation rates for L^2 results (top) and $H(\text{div})$ results (bottom). Lax-Friedrichs (LF) form (left column), upwind form (middle column), and central flux/no stabilisation (right column). The abscissa shows the time t. DNS, direct numerical simulation; SIP, symmetric interior penalty [Colour figure can be viewed at wileyonlinelibrary.com]

of stabilisation terms discussed here leads to a robust method for the inviscid TGV problem, as shown in the work of Fehn et al,[19] where the LF flux has been used. Summarising, the particular choice of the discretisation of the convection term does not have a large impact on the results for this TGV problem.

3.4.2 | Different viscosity treatment for the $H(\text{div})$-HDG method

In addition to considering variations in the convection term in the last subsection, let us now take a look at what happens when we do not use the popular SIP method for the treatment of the viscosity effects. The motivation for investigating a different method for Laplacian-like terms originates in the frequently heard criticism that the SIP parameter choice can lead to overly large penalties and, therefore, a seemingly large amount of numerical viscous dissipation. To this end, we use a more subtle mechanism to ensure discrete coercivity of the viscosity bilinear form through a so-called lifting technique. The resulting idea/concept goes back to the work of Bassi and Rebay,[39] and we will use it only in the $H(\text{div})$-HDG context; see also the works of Schöberl and Lehrenfeld[35] and Oikawa.[40]

The usual penalty term in the SIP formulation is responsible for controlling the skeleton integrals involving non-quadratic forms. To obtain this control with the penalty integral, the penalty parameter has to be chosen sufficiently large, depending on a constant which, in turn, depends on polynomial trace inverse inequalities. Lifting methods tackle the problem differently by reformulating the problematic skeleton integrals as volume integrals. From this, a stabilisation bilinear form can be characterised that guarantees nonnegativity of the resulting bilinear form without the usual SIP jump penalty. This additional stabilisation bilinear form relies on a new variable, the discrete lifting, whose DOFs can all be eliminated locally by static condensation. Therefore, their efficiency is comparable to the corresponding method without lifting. Similarly, in the context of hybrid mixed DG methods, the additional (dual) diffusive flux variable for approximating $\nabla \boldsymbol{u}$ can also be eliminated locally by static condensation, thereby resulting in a comparable situation in terms of global free DOFs. In fact, many mixed formulations can be expressed as primal formulations involving lifting operations.[41]

We define the discrete ($\boldsymbol{H}^1$) lifting of a vector-valued function living on the boundary of each element.

Definition 1. For all $K \in \mathcal{T}_h$ and $\boldsymbol{\varphi} \in \boldsymbol{L}^2(\partial K)$, the discrete ($\boldsymbol{H}^1$) lifting $\mathcal{L}(\boldsymbol{\varphi}) \in \boldsymbol{V}_h|_K \cap \boldsymbol{L}_0^2(K)$ is defined as

$$\int_K \nabla \mathcal{L}(\boldsymbol{\varphi}) : \nabla \boldsymbol{s}_h \mathrm{d}\boldsymbol{x} = -\int_{\partial K} \boldsymbol{\varphi} \cdot (\nabla \boldsymbol{s}_h) \boldsymbol{n}_K \mathrm{d}\boldsymbol{s}, \quad \forall \boldsymbol{s}_h \in \boldsymbol{V}_h|_K.$$
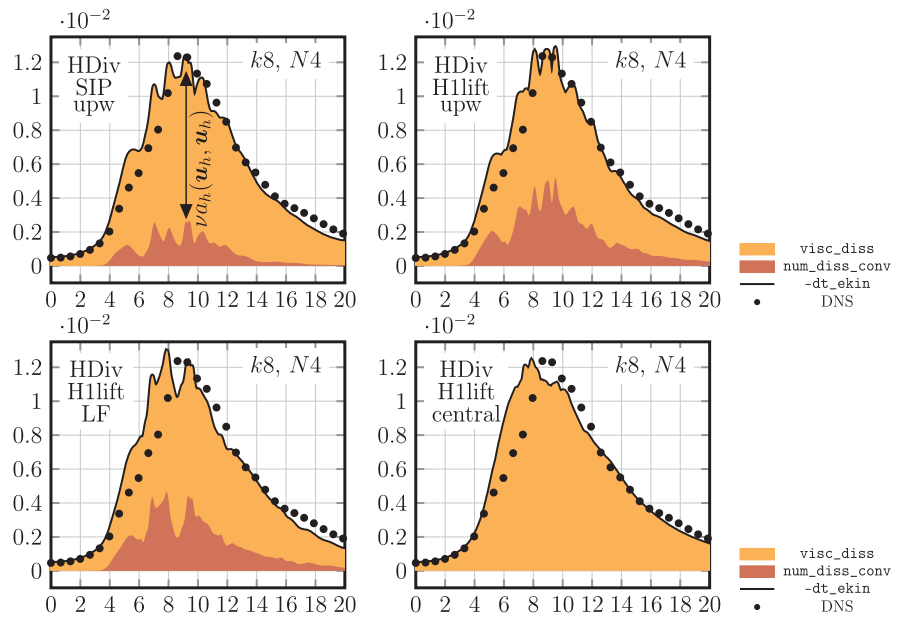
FIGURE 6 $H(\text{div})$ dissipation mechanisms for different viscosity and convection treatment for $(k, N) = (8, 4)$. The (H^1) lifting results are the subject of discussion here; the symmetric interior penalty (SIP) plot is repeated for a clearer presentation. The abscissa shows the time t. DNS, direct numerical simulation; LF, Lax-Friedrichs [Colour figure can be viewed at wileyonlinelibrary.com]

Note that the space for the discrete lifting is chosen such that $\mathcal{L}(\cdot)$ is unique and allows to rewrite $-\int_{\partial K} [\![w_h]\!] \cdot (\nabla v_h) n \, ds = \int_K \nabla \mathcal{L}([\![w_h]\!]) : \nabla v_h \, dx$ in the definition of $a_h(\cdot, \cdot)$. Our $H(\text{div})$-HDG method with (H^1) lifting results from adding the form

$$\nu \sigma_\ell \int_\Omega \nabla_h \mathcal{L}([\![u_h]\!]) : \nabla_h \mathcal{L}([\![v_h]\!]) dx$$

to the left-hand side of (14). One easily checks that this additional term ensures nonnegativity of the bilinear form $a_h(\cdot, \cdot)$ for any $\sigma_\ell \geqslant 1$. Hence, we do not require the penalty parameter σ_K to obey a largeness constraint (usually related to the constant in discrete trace inequalities). In fact, we simply use $\sigma_K = k$ and $\sigma_\ell = 2$, and thus, the (H^1) lifting method enforces weak (tangential) continuity of the discrete $H(\text{div})$ solution in a (much) weaker sense compared to the SIP method.

Figure 6 shows the dissipation rates for the $H(\text{div})$ method for $(k, N) = (8, 4)$ for our basic choice SIP with upwinding, compared against the (H^1) lifting variant with upwinding, LF, and the central flux choice for the discrete convection term. Comparing the two results with upwinding, one can observe that convection dissipation plays a larger role when the (H^1) lifting method is chosen. The (H^1) lifting method is less dissipative in the viscous term, but the convection stabilisation takes over this role such that the total dissipation rate again is not different. In agreement with the last subsection, the same observation holds true for the LF convection term. When the (H^1) lifting method is used without convection stabilisation, surprisingly, it still works and indeed gives comparably accurate results. We conclude that the particular choice of the discretisation of the viscous term does not have a large impact on the results for the TGV problem.

3.4.3 | Variation of penalty factors for the L^2-DG method

Next, we investigate the influence of the penalty factor ζ for the L^2-based method. Figure 7 shows results for values of $\zeta = 0.1, 1, 10, \infty$ (from left to right and top to bottom) considering a spatial resolution of $k = 8, N = 8$. Note that the limit method is characterised in the Appendix. For a penalty factor to $\zeta = 0.1$, the convective term exhibits a negative dissipation rate that is due to sign-indefinite terms in the energy balance (see Equation (24)), and instabilities might occur for smaller penalty factors $\zeta \to 0$. For $\zeta = 1$, only minor undershoots can be observed where the dissipation of the convective term takes a negative value. For a penalty factor of $\zeta = 10$, the numerical dissipation of the convective term is nonnegative for all times and the numerical dissipation of the convective term is larger than that of the penalty terms. Finally, the results for the limit method $\zeta = \infty$ show that, by construction, the solution will be normal-continuous and exactly divergence-free. As a consequence, the only dissipation that is not related to viscosity stems from the convection stabilisation (LF). Moreover, note that for this problem, the total amount of numerical dissipation is similar for the L^2-method with $\zeta = 1, 10$ compared to the limit method with $\zeta \to \infty$.

This parameter study nicely demonstrates the impact of the penalty terms on the dissipation rate of the convective term. However, let us explicitly emphasise that a potential conclusion like "the convective term stabilises the discretisation
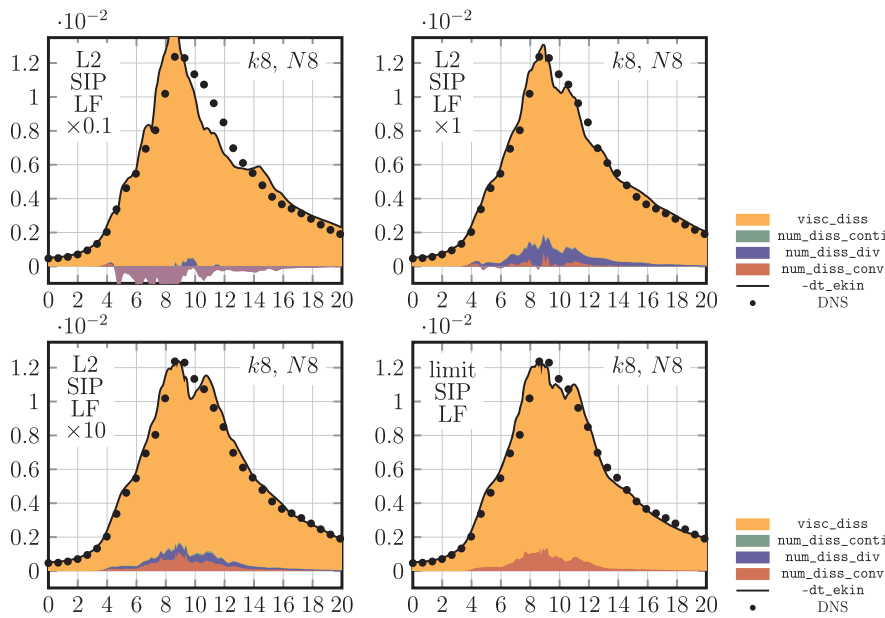
FIGURE 7 L^2 dissipation mechanisms for different penalties for $k = 8$, $N = 8$, where the lower-right plot shows results for the limit method ($\zeta \to \infty$) described in Appendix. The abscissa shows the time t. DNS, direct numerical simulation; LF, Lax-Friedrichs; SIP, symmetric interior penalty [Colour figure can be viewed at wileyonlinelibrary.com]

scheme by providing the required numerical dissipation for underresolved problems" drawn from the $\zeta = 10$ results is misleading. In fact, the convective term would not be able to stabilise the scheme without further action. As demonstrated in the work of Fehn et al,[19] the L^2-based discretisation scheme without additional penalty terms does not lead to a robust discretisation in the underresolved regime potentially leading to a blow-up of the solution. Similarly, the fact that the numerical dissipation of the continuity penalty term is small compared to the overall dissipation rate for the example considered here does not imply that the continuity penalty term is not required to obtain a robust discretisation scheme. As shown in the work of Fehn et al,[19] the continuity penalty term is indeed an essential ingredient for the robustness of the L^2-based method. Finally, we mention that the penalty factor is not intended to be a parameter that can be used to adjust the discretisation scheme. Instead, our goal is a parameter-free turbulent flow solver and designing the penalty parameter in a way to obtain a robust discretisation scheme for a constant penalty factor that is chosen once and for all (the default value is $\zeta = 1$ unless specified otherwise).

4 | TURBULENT CHANNEL FLOW

The 3D turbulent channel flow problem is a frequently used benchmark problem for assessing the ability of flow solvers to deal with wall-bounded turbulence.[42-44] As the domain for all channel flows, we consider the rectangular cuboid $\Omega = (0, L_x) \times (0, L_y) \times (0, L_z)$ with $L_x = 2\pi\delta_c$, $L_y = 2\delta_c$, $L_z = \pi\delta_c$, and channel half-width $\delta_c = 1$. In the x-direction (streamwise) and the z-direction (spanwise), periodic BCs are prescribed, whereas for $y \in \{0, L_y\}$, the no-slip condition $\boldsymbol{u} = \boldsymbol{0}$ is imposed. Due to sharp velocity gradients, it is common practice to stretch the mesh in the y-direction (wall normal direction). We choose the stretching function $\Phi : [0, 1] \to [0, L_y]$,

$$y \mapsto \delta_c \frac{\tanh(C[2y - 1])}{\tanh(C)} + \delta_c, \tag{26}$$

with a constant $C = 1.8$ for all our simulations. Therefore, given a mesh with N elements in each direction, the resulting meshes for the channel flow problem consist of N^3 hexahedra, which will be equidistant in the x- and z-directions and stretched in the y-direction. Usually, the whole motion is driven solely by a constant pressure gradient source term $\boldsymbol{f} = (f_p, 0, 0)^\dagger$ acting in the streamwise direction; cf, for example, Section 13.4 in the work of Chacón Rebollo and Lewandowski.[45]

To distinguish different turbulent channel flow situations, the friction Reynolds number Re_τ is considered most frequently. It is defined as $Re_\tau = u_\tau\delta_c/\nu$ where u_τ denotes the so-called wall friction velocity which, in turn, depends on the wall shear stress τ_w as $u_\tau^2 = \tau_w/\rho$. Under the assumption of a statistically steady state flow and with $\rho = 1$, one obtains $\tau_w = f_p\delta_c$; see Section 2.3 in the work of Dubois et al.[46] Then, by choosing $f_p = 1$, one obtains $u_\tau = 1$, and hence, the friction Reynolds number and viscosity are connected by the simple relation $\nu = 1/Re_\tau$.
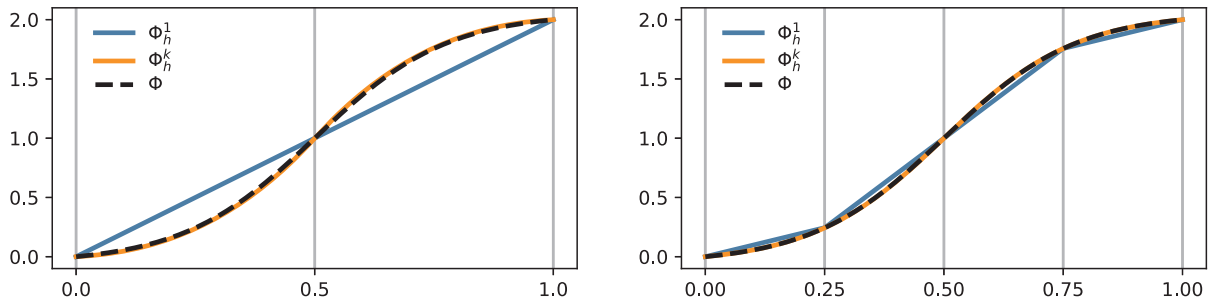
FIGURE 8 Comparison of approximations of the desired mesh stretching (26) with affine linear and isoparametric ($k = 3$) mappings for $N = 2$ (left) and $N = 4$ (right) [Colour figure can be viewed at wileyonlinelibrary.com]

The relevant quantities of interest for the channel flow involve averaging in time $\langle \cdot \rangle_t$ and in spatial directions of homogeneity $\langle \cdot \rangle_s$, where the abbreviation $\langle \cdot \rangle = \langle \langle \cdot \rangle_s \rangle_t$ is used. Then, we consider the following normalised quantities: mean velocities $\langle u_i \rangle^+ = \langle u_i \rangle / u_\tau$, Reynolds stresses $\langle u_i' u_j' \rangle^+ = \langle u_i u_j \rangle / u_\tau^2 - \langle u_i \rangle \langle u_j \rangle / u_\tau^2$, and normalised root mean square (rms) velocities $u_{i,\mathrm{rms}}^+ = |\langle u_i' u_i' \rangle^+|^{1/2}$. Furthermore, the so-called wall units $y^+ = y u_\tau / \nu$ for $y \in [0, \delta_c]$ are used. When normalising the numerical results, u_τ is the friction velocity obtained as a result of the numerical simulation and not the theoretical value $u_\tau = 1$.

4.1 | Affine vs isoparametric finite element meshes

In Section 4.2 below, we will compare results based on finite elements meshes using either *affine* or *isoparametric* mappings. We use this section to explain the possible impact of the choice of finite element meshes.

In the case of affine linear mappings, the stretching function Φ in (26) is approximated by a continuous piecewise linear approximation Φ_h^1 on an equidistant mesh for $[0, 1]$ with N elements, whereas in the isoparametric case, the approximation is done with Φ_h^k, a continuous piecewise polynomial of degree k. In Figure 8, both approximations and the stretching function (26) for $N = 2$ and $N = 4$ are sketched (for $k = 3$) for illustration purposes. One can observe that for $N = 2$ there is no stretching in the affine case and for $N = 4$ the difference between an affine stretching and the isoparametric stretching is significant. From this simple observation, we realise that while the analytical stretching function is used for the purpose of accumulating resolution close to the boundary, the realisation of the corresponding stretching function by the finite element mesh is essential. Note that this aspect is particularly relevant for high-order finite element discretisations considered here as opposed to low-order discretisations for which the number of elements is much larger.

The choice of affine or isoparametric finite element meshes also influences the choice of finite element spaces. In Section 2, we introduced the finite element spaces based on the assumption of affine linear mappings. If this assumption is no longer fulfilled and the element mappings $\Phi_K : \hat{K} \to K$, where $\hat{K} = [0, 1]^3$ is the reference hexahedron, are not affine linear, the local finite element spaces are adapted accordingly. Let $\hat{\mathbb{Q}}_k(\hat{K})$ be the tensor-product polynomial space as introduced (without the hat notation) in Section 2 but w.r.t. $\hat{K} = [0, 1]^3$ and $\hat{\mathbb{RT}}_{[k]}(\hat{K})$, accordingly. Then, we define the mapped polynomial spaces as

$$\mathbb{Q}_k(K) := \hat{\mathbb{Q}}_k(\hat{K}) \circ \Phi_K^{-1}, \quad \mathbf{Q}_k(K) := \mathbf{Q}_k(\hat{K}) \circ \Phi_K^{-1}, \quad \text{and} \quad \mathbb{RT}_{[k]}(K) := (\det(\nabla \Phi_K))^{-1} (\nabla \Phi_K) \left(\mathbb{RT}_{[k]}(\hat{K}) \circ \Phi_K^{-1} \right),$$

where the latter is the well-known Piola transformation. Let us stress that the resulting spaces contain nonpolynomial functions. From here on, we want to focus on the practical impact that a decision between affine linear and isoparametric finite element meshes can have.

Therefore, let us consider a simple 1D configuration with the underlying question of how the approximation properties of affine and isoparametric approaches differ for approximating a typical (averaged) turbulent velocity profile. The following simple (explicit) formulation of a corresponding law of the wall has been derived in the work of Reichardt[47]:

$$\langle u \rangle^+ = 2.5 \ln(1 + 0.4 y^+) + 7.8 \left[1 - \exp\left(-\frac{y^+}{11} \right) - \frac{y^+}{11} \exp(-0.33 y^+) \right]. \tag{27}$$

The accuracy of this formula shall suffice for the intended purpose of showing the different abilities to approximate functions with sharp gradients in a boundary layer as $y \to 0$. Let us explicitly point out that we do not use any form of
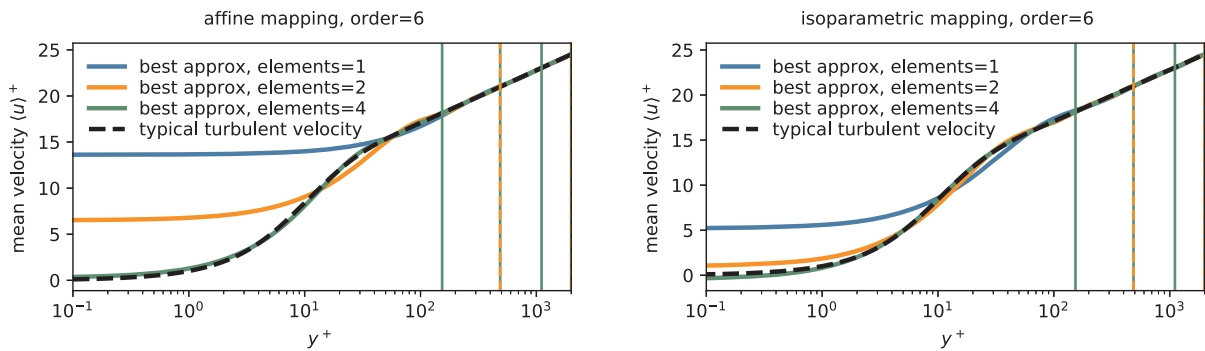
FIGURE 9 Approximation properties resulting from affine (left) and isoparametric (right) finite elements. Simplified 1D example where the L^2-best approximation of a typical turbulent velocity profile is shown on different meshes for $k = 6$. The vertical lines correspond to the location of vertices in the corresponding mesh [Colour figure can be viewed at wileyonlinelibrary.com]

wall modelling in this work (simple no-slip BCs are imposed for all simulation results shown for the turbulent channel problem). Instead, the purpose of this subsection is a theoretical investigation on how well a turbulent velocity profile can be approximated by high-order polynomial function spaces under affine and isoparametric element mappings.

In Figure 9, a simple 1D example demonstrates the different approaches by showing the L^2-best approximation of $\langle u \rangle$ on the interval $y^+ \in (0, 2000)$. Here, the mesh stretching (26) is approximated with an affine (left) and an isoparametric mapping (right) of order $k = 6$ where globally discontinuous polynomials of element-wise order $k = 6$ are used to approximate $\langle u \rangle^+$. The resulting element boundaries are indicated by vertical lines, at least when there are more than one. One can observe that the isoparametric approach roughly allows to use a once less refined mesh in this setting. In the following section, the different approaches will be compared based on the actual 3D turbulent channel flow problem. Note that the corresponding combination of polynomial order and meshes will be identical to the ones chosen in Figure 9.

4.2 | Mesh convergence study for $Re_\tau = 395$

We investigate the accuracy of the two DG methods by comparing statistical quantities of the turbulent channel flow to accurate DNS reference data from the work of Moser et al.[44] Two different polynomial degrees, $k = 3$ and $k = 6$, are studied for three different meshes with $N = 4, 8, 16$ elements for $k = 3$ and $N = 2, 4, 8$ elements for $k = 6$. The results are shown in Figure 10 for $k = 3$ and in Figure 11 for $k = 6$, where the L^2-based discretisation and the $H(\text{div})$-based discretisation are compared, considering both affine mappings (solid lines) and isoparametric mappings (dashed lines) for each discretisation. As a rough indication of the resolution close to the wall we show the location y^* with distance $h/(k+1)$ to the wall corresponding to $k + 1$ unknowns in wall-normal direction, where h is the distance of the first face parallel to the wall. In the case of an isoparametrically transformed mesh, we consider a correspondingly mapped location.

For both polynomial degrees, the numerical results converge towards the DNS reference data under mesh refinement with accurate predictions of the mean velocity profile and Reynolds stresses on the finest meshes. Comparing the $k = 3$ results to those for $k = 6$, the solution quality is comparable for the coarse mesh and the intermediate mesh. Note that on the coarsest mesh, the domain is discretised by only two elements in wall normal direction for each half of the channel for $k = 3$, and by only one element for $k = 6$. Because statistical results rather than instantaneous profiles are shown in Figures 10 and 11, the statistical profiles are mainly continuous between elements despite the discontinuous nature of the shape functions. The weak continuity constraints between elements lead to the typical kinks that can be seen in the statistical profiles for example between the first element and the second element for the $k = 3$ case on the coarsest mesh. No kinks can be observed for the $k = 6$ case on the coarsest mesh because one half of the channel is discretised by a single element. Remarkably, even these strongly underresolved simulations deliver an at least meaningful prediction of the involved statistical quantities, eg, the Reynolds stress tensor. On the fine mesh, the different methods analysed here show less variations in the results for $k = 6$ than for $k = 3$, which can be seen as an indication of an improved accuracy of the high-order simulations with $k = 6$ compared to those with $k = 3$ once the flow becomes resolved, ie, the results are less sensitive to changes in the discretisation scheme for the higher order method (note also that the number of DOFs is slightly lower for the $k = 6$ simulations). In contrast, no clear improvement of the $k = 6$ simulations over the $k = 3$ simulations can be observed in the limit of underresolution.

Comparing the results for the L^2-based discretisation to the $H(\text{div})$-based discretisation, no noticeable differences between the two methods can be observed in terms of accuracy. Furthermore, the results demonstrate rigorously that the
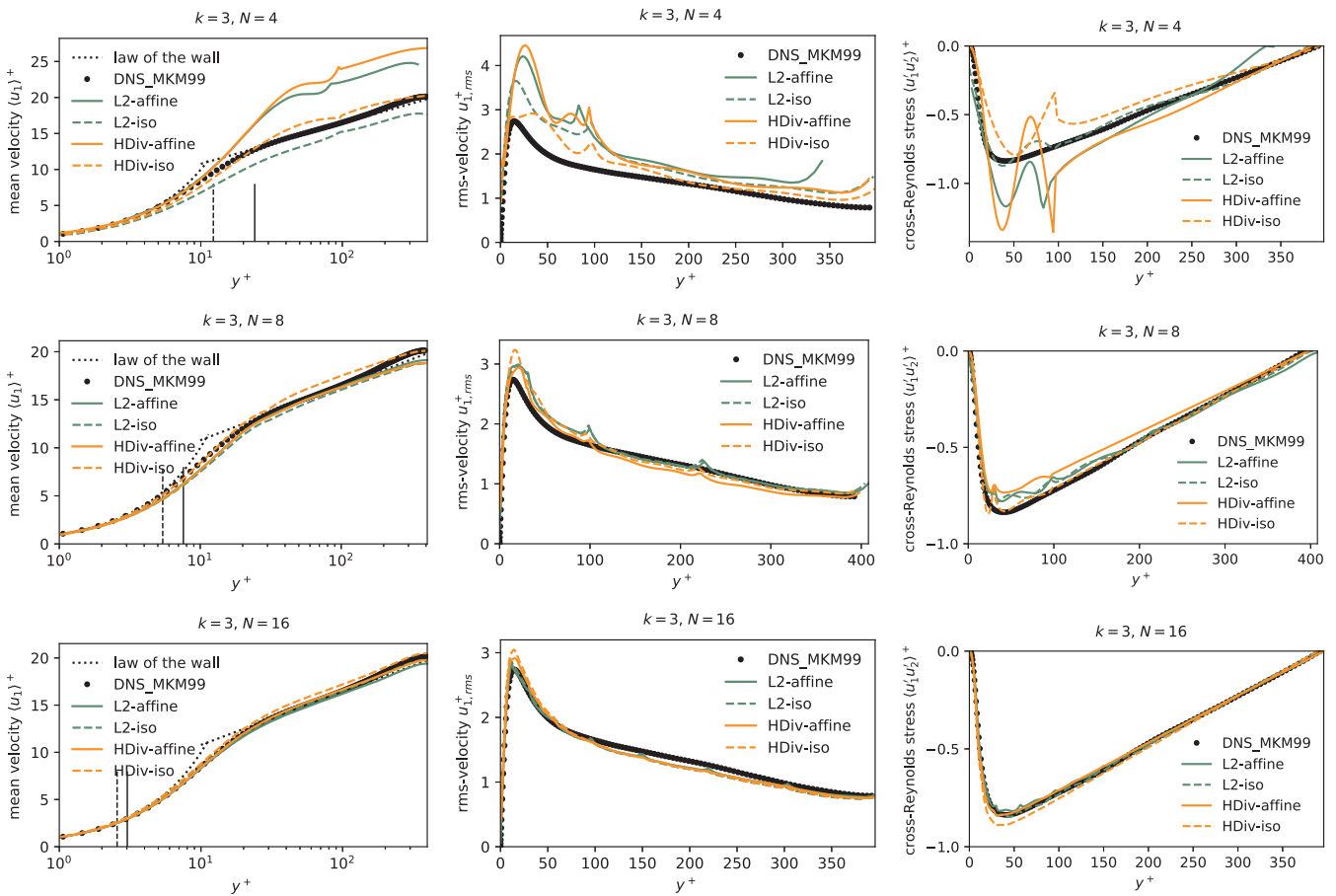
FIGURE 10 Averaged channel profiles for $k = 3$ at $Re_\tau = 395$. Both L^2 and $H(\mathrm{div})$ results are shown for $N = 4, 8, 16$ with affine and isoparametric mapping of the finite element spaces. The vertical lines indicate the location of y^* (solid line for affine and dashed line for isoparametric meshes); cf the explanation in the first paragraph of Section 4.2. DNS. DNS, direct numerical simulation [Colour figure can be viewed at wileyonlinelibrary.com]

spatial resolution described by k and h (or N) is the relevant parameter that drives the accuracy of the results and that shifts the results towards the DNS data, having a much higher influence on the solution quality than the particular DG discretisation method itself.

Regarding the polynomial degree used for the mapping of the geometry, a rather clear trend can be observed on the coarsest meshes: The isoparametric mapping shows more accurate results than the affine mapping. The results for the affine mapping on the coarse mesh appear too inaccurate with the profile for the mean velocity being significantly higher than the DNS reference results and the Reynolds stresses showing larger oscillations compared to the results with isoparametric mapping. Given the underresolution of the flow field on the coarsest mesh, the mean velocity profile is predicted very accurately in case of the isoparametric mapping. However, let us point out that a seemingly exact prediction of the mean velocity profile in terms of the mean velocity in the middle of the channel for these coarse meshes might be coincidence and it cannot be guaranteed that this result is robust under variations of the DG discretisation scheme, eg, when changing the numerical fluxes or the discretisation parameters for the convective and viscous terms. In Section 4.3 below, we explicitly demonstrate the impact of the DG discretisation parameters on the accuracy of the results for the coarsest mesh. With increasing resolution, the differences between the affine and isoparametric mappings diminish. This is in agreement with Figure 9 showing that the affine mapping is of course also able to resolve sharp gradients if the mesh is fine enough.

Due to the above observations, in the following, we restrict ourselves to showing results only for the isoparametric setting. Furthermore, as $k = 6$ does not show a clear advantage compared to $k = 3$ for the coarsest spatial resolutions, all following investigations are performed only for $k = 3$. However, we do not expect different results (qualitatively) for different polynomial orders.
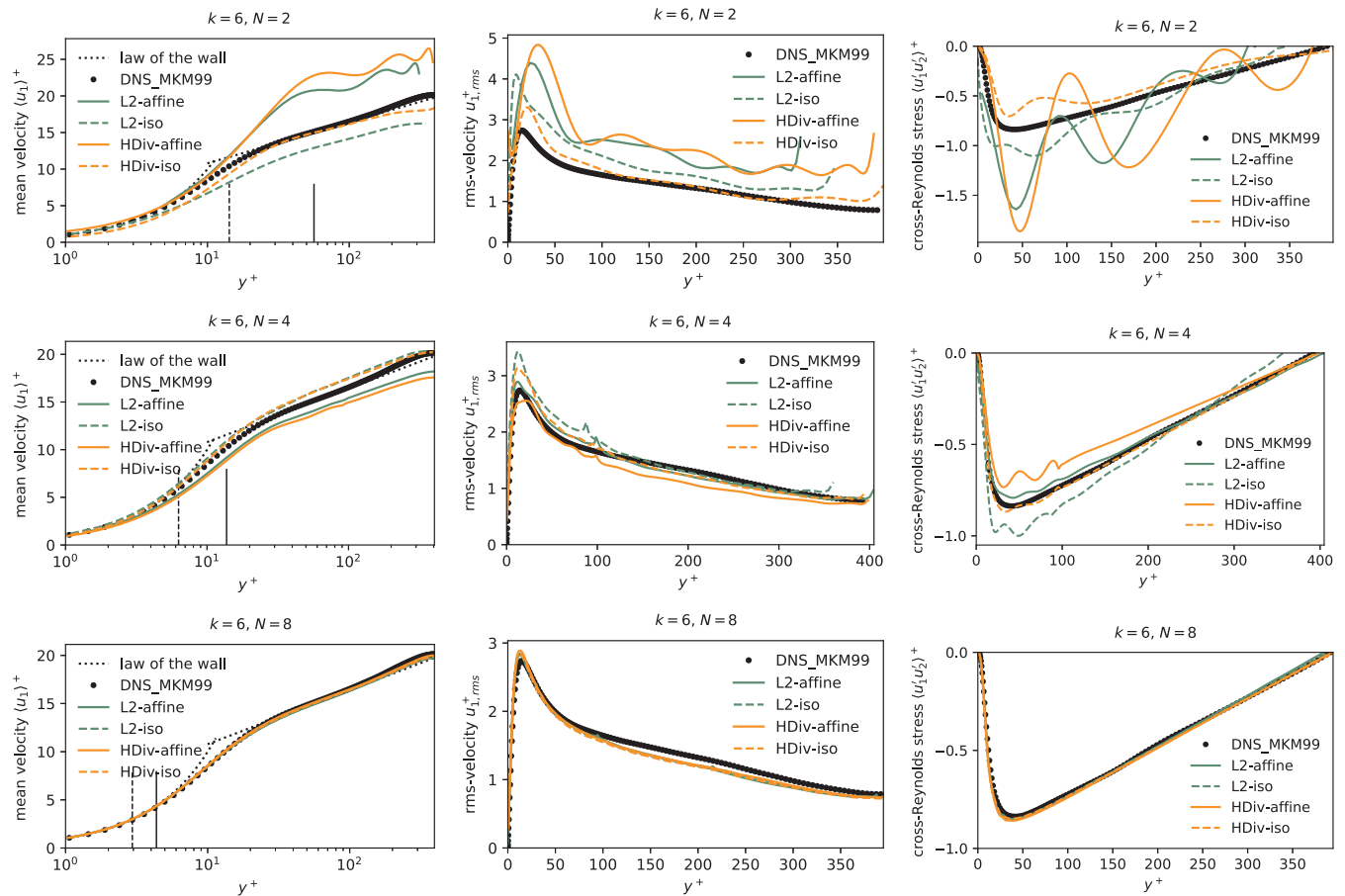
FIGURE 11　Averaged channel profiles for $k = 6$ at $Re_\tau = 395$. Both L^2 and $H(\mathrm{div})$ results are shown for $N = 2, 4, 8$ with affine and isoparametric mapping of the finite element spaces. The vertical lines indicate the location of $y*$ (solid line for affine and dashed line for isoparametric meshes); cf the explanation in the first paragraph of Section 4.2. DNS, direct numerical simulation [Colour figure can be viewed at wileyonlinelibrary.com]

4.3 │ Influence of the SIP parameter and convection stabilisation

Having analysed the convergence behaviour of the considered discretisation schemes towards the reference data under mesh refinement, it remains to demonstrate the sensitivity of the results for a given spatial resolution with respect to the parameters of the discretisation scheme. Instead of tuning the parameters to improve the solution quality for a given mesh, we want to demonstrate the "guaranteed" accuracy of the discretisation schemes, defined by the least accurate results achieved for "unsuitable" or "nonoptimised" parameters. For these investigations, we focus on the coarsest mesh for degree $k = 3$ because the variations in the results (ie, the discretisation error) can be expected to be largest for the lowest spatial resolution. On the one hand, we compare results for "upwind" fluxes to central fluxes for the convective term to investigate its impact for wall-bounded turbulent flows. On the other hand, we study the impact of the SIP penalty factor, which can be expected to be the second main parameter that determines the dissipation properties of the methods. Results for these method variations are shown in Figure 12 for the L^2-method and in Figure 13 for the $H(\mathrm{div})$-method.

For the L^2-method, we multiply our basic SIP penalty parameter described above with factors of 1, 2, 4, 8, and 16, and investigate both the "standard" LF flux formulation (top row in Figure 12) and the convective formulation of the convective term with central flux, as described in Section 3.4.1 (bottom row in Figure 12). For the LF formulation, the SIP penalty factor has a rather small impact on the results. The values of the mean velocity profile increase for increasing penalty factors and the curves move closer to the DNS reference data. Moreover, the wall shear stress increases slightly with increasing penalty factor. Taking into account the results for the Reynolds shear stress, we are not able to identify an optimal SIP penalty factor. For the central flux formulation, the SIP penalty factor has a larger impact. Without upwind stabilisation in the convective term, a SIP penalty factor ≥ 2 was required to render the method stable, which can be explained by the fact that the upwind stabilisation term and the SIP penalty term are the only terms in the discretisation
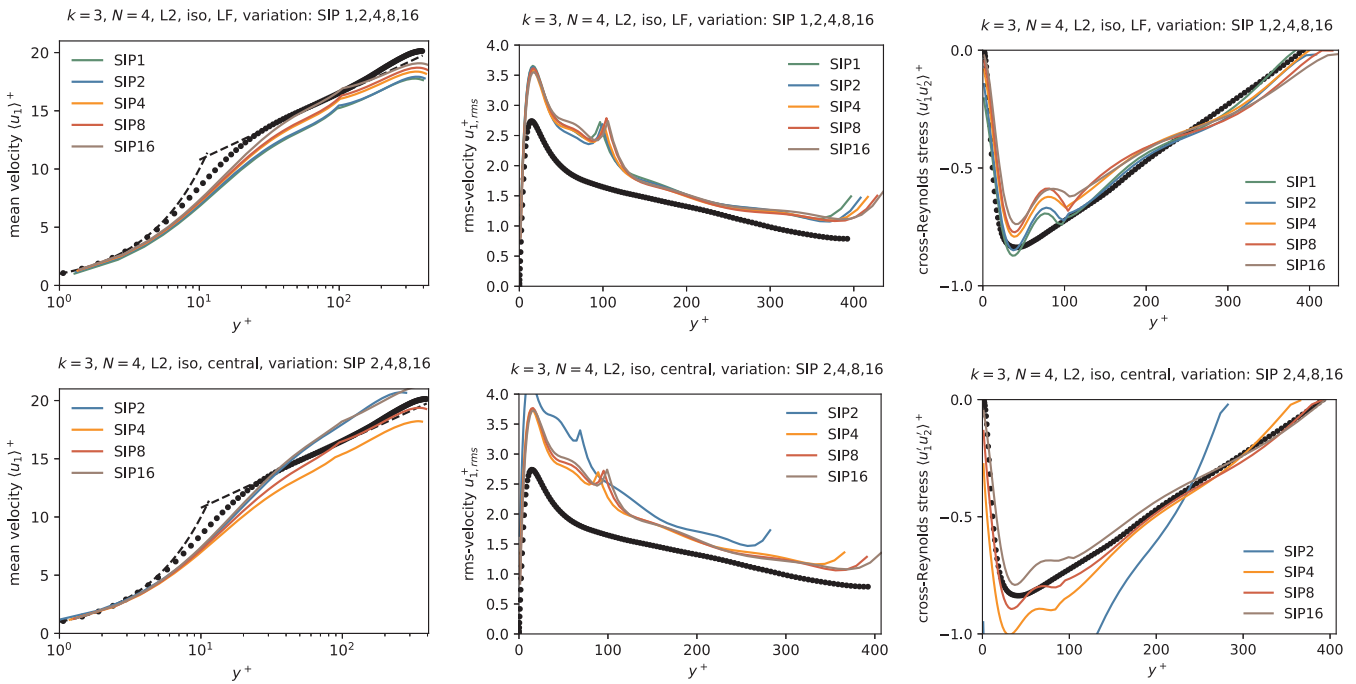
FIGURE 12 L^2: Averaged channel profiles for the most underresolved setting $k = 3$, $N = 4$ with isoparametric mapping. The basic SIP parameter is multiplied by 1, 2, 4, 8, and 16. (Top) computations with the Lax-Friedrichs (LF) stabilisation. (Bottom) convective formulation with central flux for the convection term. SIP, symmetric interior penalty [Colour figure can be viewed at wileyonlinelibrary.com]

scheme that weakly enforce tangential continuity in the velocity field and that the "portion" of the convective term has to be taken over by the SIP term if the upwind stabilisation is switched off. For a small SIP penalty factor of 2, the prediction of the wall shear stress is still inaccurate, but for increasing penalty factors of 4, 8, and 16, the accuracy achieved with the LF formulation is retained. Analogously, switching off the upwind-like stabilisation term in the LF formulation leads to qualitatively similar results as those shown in Figure 12 for the convective formulation with central flux. Because the LF formulation produces better results with the SIP penalty factor as defined in Section 2.2, the LF formulation is the preferred choice here, but we mention that a purely central convective flux can be used as well even though it remains unclear whether this extends to more complex problems. Because we want to avoid parameter adjustments for the present discretisation schemes, the LF formulation with minimal SIP penalty factor is advantageous given the fact that the iteration counts of iterative solvers typically increase with increasing penalty factors and given the fact that parameter tuning should be considered with some care in underresolved turbulence simulations in our opinion because of the important difference of predicting versus reproducing results.

For the H(div) method, our basic SIP penalty parameter described above is multiplied with factors 1, 2, 4, and 16, where 8 is omitted only to improve the visibility of the results. Figure 13 shows the resulting channel profiles with
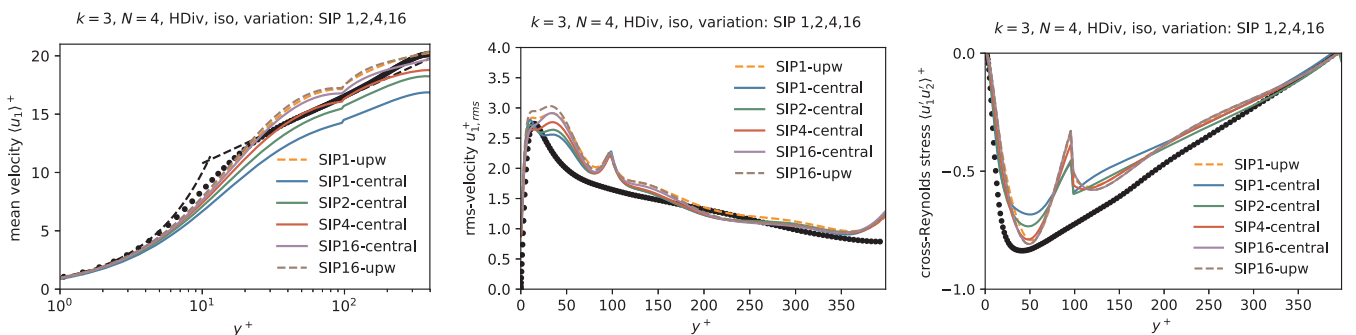

FIGURE 13 H(div): Averaged channel profiles for the most underresolved setting $k = 3$, $N = 4$ with isoparametric mapping. The basic symmetric interior penalty (SIP) parameter is multiplied by 1, 2, 4, and 16, and the two limit cases with and without upwinding are investigated [Colour figure can be viewed at wileyonlinelibrary.com]

different SIP parameters and with and without upwinding (convection stabilisation). The first conclusion can be drawn from comparing SIP1-upw with SIP16-upw (dashed lines), which show that once upwinding is used, the particular parameter for the SIP mechanism does not play a major role anymore. On the other hand, SIP1 without upwinding (central) obviously leads to worse (but still stable) results. Thus, we conclude also for the H(div) method that a certain amount of numerical dissipation is beneficial for channel flow problems. Interestingly, when the SIP parameter is increased for the central flux situation, the curves approach the SIP1-upw curves in a continuous fashion up to the point that SIP16-central is very similar to SIP16-upw. From this behaviour we conclude that for the H(div) method and for this problem, the particular source for numerical dissipation, be it from SIP or from upwinding, is not decisive. A possible explanation for this phenomenon can be derived from regarding how upwinding works (cf (19)): In principle, upwinding is an anisotropic mechanism acting only on facets where inflow or outflow occurs. However, in turbulent channel flow simulations, the turbulent mixing properties of the instantaneous flow basically result in the fact that upwinding acts isotropically. Thus, SIP and upwinding are penalising the tangential discontinuity in a very similar way here.

5 | SUMMARY AND CONCLUSIONS

Two conceptually different DG discretisation methods for the numerical simulation of incompressible turbulent flow problems have been analysed and compared in this work. Fulfilling the incompressibility constraint as well as interelement continuity of the velocity field in the direction normal to element interfaces are of particular importance regarding the numerical robustness and accuracy of the discretisation schemes. Remarkably, the robustness of the considered methods allows to obtain meaningful results even in significantly underresolved settings.

This work shows that these aspects can be addressed successfully by the use of either specialised function spaces in an H(div)-context fulfilling these properties exactly, or plain L^2-conforming spaces equipped with additional stabilisation terms enforcing the above requirements in a weak sense. To assess the properties of the numerical discretisation schemes, the three-dimensional Taylor-Green vortex problem has been investigated as a representative of transitional flows and decaying turbulence, as well as the turbulent channel flow problem as a representative of wall-bounded turbulent flows. The comparison of the L^2-based and H(div)-based methods, as well as studying the impact of different discretisation variants for the convective and viscous terms and the influence of certain discretisation parameters, has several important implications. Overall, variations of the discretisation scheme were found to have only a small or moderate influence on the accuracy of the results. This insensitivity can be seen as an advantage of the methods analysed here in the sense that accurate results are not only achieved for a certain set of parameters but that this accuracy can be expected in general. At the same time and as shown in this work, this implies that the possibilities to optimise high-order DG discretisations by choosing one or the other flux formulation appear to be limited as compared to the influence that other parameters—most importantly the spatial resolution parameters h and k—have on the accuracy of the results.

Interestingly, it is commonly believed that the discretisation of the convective term plays a crucial role in DG discretisations rendering the method stable for convection-dominated problems by the choice of suitable flux functions such as upwind fluxes. Let us emphasise that, even in underresolved settings, basically independent of the chosen convective flux, we still have been able to obtain accurate results with respect to, eg, predictions of statistical quantities such as the Reynolds stress tensor. In this context, a key message of this work is that—for DG discretisations of the incompressible Navier-Stokes equations—measures enforcing mass conservation and energy stability in a weak sense or in a local (exact) sense are more important to achieve a robust discretisation scheme. Indeed, this work shows that convection stabilisation (upwind or LF) is neither necessary to ensure energy stability of the overall method nor crucial for the robustness of the simulations. Whereas the L^2 method relies on divergence and normal-continuity stabilisation for energy-stability, the divergence-free H(div) method is designed to be energy stable without any additional stabilisation. Thus, concerning robustness for incompressible flows, the crucial ingredient turns out to be a meaningful fulfilment of the divergence-free constraint and normal-continuity of the velocity, realised weakly (L^2) or strongly (H(div)) in this work. Nonetheless, the use of upwind-like fluxes for the convective term was found advantageous, especially for the wall-bounded turbulent channel flow problem because alternative central flux formulations have shown larger sensitivity with respect to the discretisation parameters of the viscous term.

We conclude that both L^2-based and H(div)-based discretisations show convincing results in highly underresolved scenarios and are promising candidates as generic turbulent flow solvers. Preconceptions like "high-order methods are not robust" or "no-model LES is not physical" are widespread and the results shown here—albeit promising—are certainly not enough to conclusively demonstrate the opposite. Therefore, given thematurity of the results shown here, the applicability of these novel high-order DG discretisations to complex engineering applications should be considered as part of future work.

ORCID

Martin Kronbichler https://orcid.org/0000-0001-8406-835X
Christoph Lehrenfeld https://orcid.org/0000-0003-0170-8468
Philipp W. Schroeder https://orcid.org/0000-0001-7644-4693

REFERENCES

1. Uranga A, Persson P-O, Drela M, Peraire J. Implicit large eddy simulation of transition to turbulence at low Reynolds numbers using a Discontinuous Galerkin method. *Int J Numer Methods Eng*. 2011;87(1-5):232-261.
2. Gassner GJ, Beck AD. On the accuracy of high-order discretizations for underresolved turbulence simulations. *Theor Comput Fluid Dyn*. 2012;27(3-4):221-237.
3. Carton de Wiart C, Hillewaert K, Duponcheel M, Winckelmans G. Assessment of a discontinuous Galerkin method for the simulation of vortical flows at high Reynolds number. *Int J Numer Methods Fluids*. 2014;74(7):469-493.
4. Beck AD, Bolemann T, Flad D, et al. High-order discontinuous Galerkin spectral element methods for transitional and turbulent flow simulations. *Int J Numer Methods Fluids*. 2014;76(8):522-548.
5. Moura RC, Mengaldo G, Peiró J, Sherwin SJ. On the eddy-resolving capability of high-order discontinuous Galerkin approaches to implicit LES/under-resolved DNS of Euler turbulence. *J Comput Phys*. 2017;330:615-623.
6. Fernandez P, Nguyen NC, Peraire J. The hybridized discontinuous Galerkin method for implicit large-eddy simulation of transitional turbulent flows. *J Comput Phys*. 2017;336:308-329.
7. Winters AR, Moura RC, Mengaldo G, et al. A comparative study on polynomial dealiasing and split form discontinuous Galerkin schemes for under-resolved turbulence computations. *J Comput Phys*. 2018;372:1-21.
8. Hesthaven JS, Warburton T. *Nodal Discontinuous Galerkin Methods: Algorithms, Analysis, and Applications*. New York, NY: Springer Science+Business Media; 2007.
9. Shahbazi K, Fischer PF, Ethier CR. A high-order discontinuous Galerkin method for the unsteady incompressible Navier–Stokes equations. *J Comput Phys*. 2007;222(1):391-407.
10. Botti L, Di Pietro DA. A pressure-correction scheme for convection-dominated incompressible flows with discontinuous velocity and continuous pressure. *J Comput Phys*. 2011;230(3):572-585.
11. Ferrer E, Willden RHJ. A high order discontinuous Galerkin finite element solver for the incompressible Navier–Stokes equations. *Comput Fluids*. 2011;46(1):224-230.
12. Klein B, Kummer F, Keil M, Oberlack M. An extension of the SIMPLE based discontinuous Galerkin solver to unsteady incompressible flows. *Int J Numer Methods Fluids*. 2015;77(10):571-589. https://doi.org/10.1002/fld.3994
13. Ferrer E. An interior penalty stabilised incompressible discontinuous Galerkin–Fourier solver for implicit large eddy simulations. *J Comput Phys*. 2017;348:754-775.
14. Steinmoeller DT, Stastna M, Lamb KG. A short note on the discontinuous Galerkin discretization of the pressure projection operator in incompressible flow. *J Comput Phys*. 2013;251:480-486.
15. Joshi SM, Diamessis PJ, Steinmoeller DT, Stastna M, Thomsen GN. A post-processing technique for stabilizing the discontinuous pressure projection operator in marginally-resolved incompressible inviscid flow. *Comput Fluids*. 2016;139:120-129.
16. Krank B, Fehn N, Wall WA, Kronbichler M. A high-order semi-explicit discontinuous Galerkin solver for 3D incompressible flow with application to DNS and LES of turbulent channel flow. *J Comput Phys*. 2017;348:634-659.
17. Piatkowski M, Müthing S, Bastian P. A stable and high-order accurate discontinuous Galerkin based splitting method for the incompressible Navier–Stokes equations. *J Comput Phys*. 2018;356:220-239.
18. Lehrenfeld C, Schöberl J. High order exactly divergence-free hybrid discontinuous Galerkin methods for unsteady incompressible flows. *Comput Methods Appl Mech Eng*. 2016;307:339-361.
19. Fehn N, Wall WA, Kronbichler M. Robust and efficient discontinuous Galerkin methods for under-resolved turbulent incompressible flows. *J Comput Phys*. 2018;372:667-693.
20. Schroeder PW. *Robustness of High-Order Divergence-Free Finite Element Methods for Incompressible Computational Fluid Dynamics* [PhD thesis]. Göttingen, Germany: Georg-August-Universität Göttingen; 2019.
21. Akbas M, Linke A, Rebholz LG, Schroeder PW. The analogue of grad–div stabilization in DG methods for incompressible flows: limiting behavior and extension to tensor-product meshes. *Comput Methods Appl Mech Eng*. 2018;341:917-938.
22. Schöberl J. *C++11 Implementation of Finite Elements in NGSolve*. ASC Report No. 30/2014. Vienna, Austria: Institute for Analysis and Scientific Computing, Vienna University of Technology; 2014.
23. Alzetta G, Arndt D, Bangerth W, et al. The deal.II library, version 9.0. *J Numer Math*. 2018;26(4):173-183.
24. Tritton DJ. *Physical Fluid Dynamics*. 2nd ed. New York, NY: Oxford University Press; 1988.
25. Schlichting H, Gersten K. *Boundary-Layer Theory*. 8th ed. New York, NY: Springer-Verlag Berlin Heidelberg; 2000.

26. Durst F. *Fluid Mechanics: An Introduction to the Theory of Fluid Flows*. Berlin, Germany: Springer-Verlag Berlin Heidelberg; 2008.

27. Boffi D, Brezzi F, Fortin M. *Mixed Finite Element Methods and Applications*. Berlin, Germany: Springer-Verlag Berlin Heidelberg; 2013.

28. Fehn N, Wall WA, Kronbichler M. On the stability of projection methods for the incompressible Navier–Stokes equations based on high-order discontinuous Galerkin discretizations. *J Comput Phys*. 2017;351:392-421.

29. Hillewaert K. *Development of the Discontinuous Galerkin Method for High-Resolution, Large Scale CFD and Acoustics in Industrial Geometries* [PhD thesis]. Ottignies-Louvain-la-Neuve, Belgium: Université Catholique de Louvain; 2013.

30. Fehn N, Wall WA, Kronbichler M. Efficiency of high-performance discontinuous Galerkin spectral element methods for under-resolved turbulent incompressible flows. *Int J Numer Methods Fluids*. 2018;88(1):32-54.

31. Lehrenfeld C, Schöberl J. *Hybrid Discontinuous Galerkin Methods for Solving Incompressible Flow Problems* [master's thesis]. Aachen, Germany: Rheinisch-Westfälische Technische Hochschule Aachen; 2010.

32. Schöberl J, Zaglmayr S. High order Nédélec elements with local complete sequence properties. *COMPEL*. 2005;24(2):374-384.

33. Zaglmayr S. *High Order Finite Element Methods for Electromagnetic Field Computation* [PhD thesis]. Graz, Germany: Graz University of Technology; 2006.

34. Ascher UM, Ruuth SJ, Spiteri RJ. Implicit-explicit Runge–Kutta methods for time-dependent partial differential equations. *Appl Numer Math*. 1997;25(2-3):151-167.

35. Schöberl J, Lehrenfeld C. Domain decomposition preconditioning for high order hybrid discontinuous Galerkin methods on tetrahedral meshes. In: *Advanced Finite Element Methods and Applications*. Berlin, Germany: Springer-Verlag Berlin Heidelberg; 2013;27-56.

36. Lehrenfeld C, Lube G, Schroeder PW. A natural decomposition of viscous dissipation in DG methods for turbulent incompressible flows. 2018. http://adsabs.harvard.edu/abs/2018arXiv181112769L

37. Taylor GI, Green AE. Mechanism of the production of small eddies from large ones. *Proc Royal Soc A Math Phys Eng Sci*. 1937;158(895):499-521.

38. Brachet ME. Direct simulation of three-dimensional turbulence in the Taylor–Green vortex. *Fluid Dyn Res*. 1991;8(1-4):1-8.

39. 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(2):267-279.

40. Oikawa I. Hybridized discontinuous Galerkin method with lifting operator. *JSIAM Letters*. 2010;2:99-102.

41. Cockburn B, Gopalakrishnan J, Lazarov R. Unified hybridization of discontinuous Galerkin, mixed, and continuous Galerkin methods for second order elliptic problems. *SIAM J Numer Anal*. 2009;47(2):1319-1365.

42. Moin P, Kim J. Numerical investigation of turbulent channel flow. *J Fluid Mech*. 1982;118:341-377.

43. Kim J, Moin P, Moser R. Turbulence statistics in fully developed channel flow at low Reynolds number. *J Fluid Mech*. 1987;177:133-166.

44. Moser RD, Kim J, Mansour NN. Direct numerical simulation of turbulent channel flow up to $Re_\tau = 590$. *Phys Fluids*. 1999;11(4):943-945.

45. Chacón Rebollo T, Lewandowski R. *Mathematical and Numerical Foundations of Turbulence Models and Applications*. New York, NY: Springer Science+Business Media; 2014.

46. Dubois T, Jauberteau F, Temam R. *Dynamic Multilevel Methods and the Numerical Simulation of Turbulence*. New York, NY: Cambridge University Press; 1999.

47. Reichardt H. Vollständige Darstellung der turbulenten Geschwindigkeitsverteilung in glatten leitungen. *ZAMM Zeitschrift Angewandte Math Mech*. 1951;31(7):208-219.

APPENDIX

CHARACTERISATION OF A LIMIT METHOD

The $\boldsymbol{L}^2$-DG method is defined by the velocity and pressure spaces

$$\boldsymbol{V}_h = \mathbb{Q}_k, \quad Q_h = \mathbb{Q}_{k-1} \cap L_0^2(\Omega)$$

and the variational formulation (5), which depends on the stabilisation parameters τ_D and τ_C; cf (12) and (13).

One may ask the question if the limit $\tau_D, \tau_C \to \infty$ can be characterised as a meaningful discretisation. To this end, we first realise that in the limits $\tau_D, \tau_C \to \infty$, the solution $\boldsymbol{u}_h$ has to fulfil $j_{\text{div},h}(\boldsymbol{u}_h, \boldsymbol{v}_h) = 0$ and $j_{\text{conti},h}(\boldsymbol{u}_h, \boldsymbol{v}_h) = 0$ for all $\boldsymbol{v}_h \in \boldsymbol{V}_h$, as both bilinear forms are positive semidefinite. The subspace of functions in $\boldsymbol{V}_h$ that fulfil these constraints is pointwise divergence free and normal continuous across element interfaces. In this sense, the limit shares the

properties of the H(div)-based method. However, the finite element space V_h and, hence, the pointwise divergence-free and normal-continuous subspace

$$V_h^0 = \{u_h \in V_h \,:\, j_{\text{div},h}(u_h, v_h) = j_{\text{conti},h}(u_h, v_h) = 0 \text{ for all } v_h \in V_h\}$$

are different from the H(div)-based method (on nonsimplex meshes).

Let us—for ease of presentation—assume that the mesh consists of straight, affine linearly mapped hexahedral elements. In this case, if $u_h \in V_h^0$, the weak divergence constraint $d_h(u_h, q_h) = 0$ for all $q_h \in Q_h$ (cf (11)) can be rewritten as $d_h(u_h, q_h) = g_h(q_h, u_h) = 0$ for all $q_h \in Q_h$.

To characterise V_h^0 through constraints by Lagrange multiplier functions, we rewrite

$$j_{\text{div},h}(u_h, v_h) = 0 \quad \forall v_h \in V_h \quad \text{as} \quad d_{\text{div},h}(u_h, q_h) = \int_\Omega (\nabla_h \cdot u_h) q_h \mathrm{d}x = 0 \quad \forall q_h \in Q_h^{\text{vol}} = (\nabla_h \cdot V_h) \qquad \text{(A1a)}$$

$$\text{and} \quad j_{\text{conti},h}(u_h, v_h) = 0 \quad \forall v_h \in V_h \quad \text{as} \quad d_{\text{conti},h}(u_h, \hat{q}_h) = \sum_{F \in \mathcal{F}_h^i} \int_F (\llbracket u_h \cdot n \rrbracket) \hat{q}_h \mathrm{d}s = 0 \quad \forall \hat{q}_h \in Q_h^{\text{skel}} = \text{tr}^n|_{\mathcal{F}_h^i} V_h. \quad \text{(A1b)}$$

Here, Q_h^{skel} is the space of functions defined on the (interior) skeleton of the mesh that is obtained by taking the normal trace of functions in V_h. We can explicitly characterise this space as the following space of facet unknowns:

$$Q_h^{\text{skel}} = \left\{ \hat{q}_h \in L^2\left(\mathcal{F}_h^i\right) \,:\, q_h|_F \in \mathbb{Q}_k(F), \quad \forall F \in \mathcal{F}_h^i \right\}.$$

This space is well known in the community of hybrid mixed and hybrid DG methods. Less established is the space Q_h^{vol}, which, however, can also be characterised explicitly as follows:

$$Q_h^{\text{vol}} = \left\{ q_h \in \mathbb{Q}_k \,:\, q_h|_T \in \mathbb{Q}_k(K) \setminus \left\{ \prod_{i=1}^d \xi_i^k \right\}, \forall K \in \mathcal{T}_h \right\} \cap L_0^2(\Omega),$$

where ξ_i, $i = 1, 2, 3$ are the spatial coordinates. We added the zero-mean constraint for uniqueness of solutions in the later given variational formulation. This means Q_h^{vol} is $\mathbb{Q}_k$, except for the element-wise highest-order bubble. Moreover, Q_h^{vol} is a superset of Q_h (because the pressure is approximated by polynomials of degree $k - 1$) so that the weak divergence constraint $d_h(u_h, q_h) = 0$ holds for all $q_h \in Q_h$ (cf (10)) and the corresponding Lagrange multiplier $p_h \in Q_h$ becomes superfluous.

With these characterisations, one can define the limit method as

$$\begin{cases} \text{Find } (u_h, p_h, \hat{p}_h) : (0, T] \to V_h \times Q_h^{\text{vol}} \times Q_h^{\text{skel}} \text{ with } u_h(0) = u_{0h} \text{ s.t., } \forall (v_h, q_h, \hat{q}_h) \in V_h \times Q_h^{\text{vol}} \times Q_h^{\text{skel}}, \\ (\partial_t u_h, v_h) + \nu a_h(u_h, v_h) + c_h(u_h; u_h, v_h) + d_{\text{div},h}(v_h, p_h) + d_{\text{conti},h}(v_h, \hat{p}_h) + d_{\text{div},h}(u_h, q_h) + d_{\text{conti},h}(u_h, \hat{q}_h) = (f, v_h). \end{cases}$$

Let us note that the solution u_h will be in V_h^0, which has dimension $\dim(V_h) - \dim(Q_h^{\text{vol}}) - \dim(Q_h^{\text{skel}})$.

Remark 2. In this section, we assumed affine linear element geometries. This assumption can be dropped if the (discontinuous) finite element space V_h is defined through Piola transformations from the reference element to the physical element; cf Section 4.1. Thereby, the normal traces and the divergence of V_h can be characterised through polynomials again.