POLITECNICO DI TORINO
Repository ISTITUZIONALE

General order virtual element approximation for the Smagorinsky turbulence model

Original
General order virtual element approximation for the Smagorinsky turbulence model / Berrone, S., Cascavita Mellado, K.L., Delgado Ávila, E., Rubino, S., Strazzullo, M., Vicini, F.. - In: MATHEMATICS AND COMPUTERS IN SIMULATION. - ISSN 0378-4754. - ELETTRONICO. - 249:(2026), pp. 958-972. [10.1016/j.matcom.2026.06.009]

Availability:
This version is available at: 11583/3015709 since: 2026-09-19T21:39:40Z

Publisher:
Elsevier

Published
DOI:10.1016/j.matcom.2026.06.009

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright

(Article begins on next page)

30 October 2026

Mathematics and Computers in Simulation 249 (2026) 958–972

Contents lists available at ScienceDirect

Mathematics and Computers in Simulation

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

Original articles

General order virtual element approximation for the Smagorinsky turbulence model

Stefano Berrone [a,b], Karol L. Cascavita [a,b,*], Enrique Delgado Ávila [c], Samuele Rubino [d], Maria Strazzullo [a,b], Fabio Vicini [a,b]

[a] Politecnico di Torino, Department of Mathematical Sciences "Giuseppe Luigi Lagrange", Corso Duca degli Abruzzi, 24, Turin, 10129, Italy
[b] INdAM-GNCS group member, Italy
[c] Departamento Matemática Aplicada II, Universidad de Sevilla, Avda. Reina Mercedes s/n, Sevilla, 42012, Spain
[d] Departamento EDAN & IMUS, Universidad de Sevilla, Avda. Reina Mercedes s/n, Sevilla, 42012, Spain

ARTICLE INFO

MSC:
35Q30
65M22
76F25

Keywords:
Navier–Stokes equations
Smagorinsky model
Convection-dominated flows
Virtual element method

ABSTRACT

In this paper, we investigate a Smagorinsky model in a virtual element framework to simulate convection-dominated Navier–Stokes equations. We conduct a two-dimensional numerical investigation to assess the performance of the general order virtual element approximation in this context. First, we examine numerically the convergence of the method with respect to the meshsize to certify the novel virtual element numerical discretization, which includes, for the first time, a discretization of the Smagorinsky term. Moreover, we present a numerical study of a lid-driven cavity for different Reynolds numbers (up to 10000) and meshes (uniform, anisotropic, and isotropic with hanging nodes). The results highlight the main advantage of using the virtual elements method in this context: the isotropic refinement with hanging nodes enhances the accuracy of the solution compared to the anisotropic mesh, uses fewer degrees of freedom with respect to the uniform mesh, and yields the most stable behavior in terms of convergence of the Newton solver.

1. Introduction

The solution of the Navier–Stokes Equations (NSE) is an ubiquitous goal across a wide range of industrial and scientific fields. The main challenges arise in the convection-dominated regimes, where the meshsize should comply with the Kolmogorov scale to accurately capture the complex flow features [1,2]. However, this requirement yields prohibitively computational costs, and using coarser meshes is not a practical alternative, as it might result in inaccurate simulations with spurious numerical oscillations or failure of the Newton solver to converge. To address these problems, numerical stabilization has been classically employed: the NSE model is modified according to the chosen stabilization strategy, allowing the use of a coarser mesh while still achieving accurate results and yet reducing computational efforts. For a survey on numerical stabilization approaches, we refer to the monograph [3]. Moreover, the reader may refer to the following list for several works on numerical stabilization of convection-dominated flows successfully applied in various contexts [4–7]. This contribution focuses on the Smagorinsky model, which can be traced back to Joseph Smagorinsky [8] and was conceived for the numerical simulation of turbulence in weather forecasting. It is known to be the basic Large Eddy Simulation (LES) model [9,10] to simulate turbulence and has been applied in numerous settings due to its simplicity [11–19].

* Corresponding author at: Politecnico di Torino, Department of Mathematical Sciences "Giuseppe Luigi Lagrange", Corso Duca degli Abruzzi, 24, Turin, 10129, Italy.
E-mail address: karol.cascavita@polito.it (K.L. Cascavita).

https://doi.org/10.1016/j.matcom.2026.06.009
Received 9 October 2025; Received in revised form 21 May 2026; Accepted 8 June 2026
Available online 15 June 2026
0378-4754/© 2026 The Authors. Published by Elsevier B.V. on behalf of International Association for Mathematics and Computers in Simulation (IMACS). This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

Specifically, we propose and analyze a discrete Smagorinsky model in a virtual element framework. Virtual Element Methods (VEM) were introduced in 2013 with the seminal works [20,21] to solve linear diffusion problems. It is nowadays considered and popularized as an extension of the Finite Element Methods (FEM) to more general polytopal meshes. The cornerstone of this method is the ability to build shape functions for general polygons, which is a key aspect to constructing a robust method that accommodates arbitrary shapes and orders. The use of the *virtual* label arises due to the presence of non-polynomial shape functions, whose pointwise values and closed-form expressions are not explicitly known. Computations are possible through local projection operators and suitable degrees of freedom. A non-exhaustive list of VEM applications across various fields follows [22–29]. Moreover, a comprehensive documentation discussing the research developed over the last decade was recently published [30].

In the realm of fluid dynamics, virtual element discretizations were first devised for the Stokes problem [31,32] and later extended to the Navier–Stokes equation in [33]. In the latter, the authors introduced the enhancement condition in the divergence-free virtual setting. Namely, a condition that allows L^2-projectors to be computable for higher orders. The key novelty of this VEM family lies in the divergence-free property, which is usually only weakly preserved in conventional discretization techniques. In this work, we take advantage of both this important feature and the versatility of VE meshes, which have great potential to accurately solve fluid dynamics problems in convection-dominated settings.

We propose a standard convergence analysis of the VEM approximation to the exact solution of the Smagorinsky model to certify the novel VEM discretization. Moreover, we test the remarkable properties of VEM in a lid-cavity problem with Reynolds numbers up to 10000, comparing three types of meshes: (i) uniform, (ii) anisotropic, and (iii) isotropic with hanging nodes, i.e., points between two consecutive aligned edges of the tessellation. The ease in using hanging nodes on general meshes is a peculiarity of polygonal approximations, and in our tests, the isotropic mesh with hanging nodes proved to be the best option in terms of accuracy, computational efficiency, and robustness with respect to the Newton solver convergence. All numerical tests have been performed for two-dimensional spatial domains.

The main novelties of this contribution are as follows:

- the discretization of the Smagorinsky model via general order VEM approximation with numerical validation through several convergence tests;
- the exploitation of the advantageous general polygonal VEM meshes in convection-dominated regimes combined with Smagorinsky stabilization in comparison with other mesh choices representing the state of the art in the literature.

The rest of the paper is organized as follows. The model problem is presented in its continuous form in Section 2. Section 3 is devoted to the virtual element setting. Finally, in Section 4, we present the numerical results. We close the paper by drawing some conclusions in Section 5.

2. Problem formulation

We introduce the notation that will be used throughout the manuscript. We use the notation $(\cdot,\cdot)_{\mathcal{O}}$ to denote the L^2-scalar product for any domain[1] $\mathcal{O} \subset \mathbb{R}^2$. Although the presented VEM methodology applies to a general framework that also accommodates $\mathcal{O} \subset \mathbb{R}^3$, we restrict the presentation to the bidimensional case. In addition, let $|\cdot|_{\ell^2}$ represents the Euclidean norm for vectors, and, with a slight abuse of notation, let also denote the Frobenius norm defined for all $\boldsymbol{\gamma} \in \mathbb{R}^{d\times d}$ as $|\boldsymbol{\gamma}|_{\ell^2} := (\boldsymbol{\gamma} : \boldsymbol{\gamma})^{\frac{1}{2}}$. We indicate with $(1:n)$ the ordered set of natural numbers from 1 to any integer $n > 0$. Regarding physical notations, we use the dimensionless Reynolds number $Re := \overline{U}L/\nu$, where $\nu > 0$, $\overline{U}$, and L denote the kinematic viscosity, the characteristic velocity, and the characteristic length of the problem at hand, respectively. This dimensionless number describes the flow regime. For example, a large Reynolds number indicates a convection-dominated flow and turbulent behavior.

2.1. The incompressible Navier–Stokes equations

We aim to find the velocity $\mathbf{u}$ and the ratio between the fluid pressure and its density p on the domain $\Omega \subset \mathbb{R}^2$, such that

$$\begin{cases} -\nabla\cdot(\nu\nabla\mathbf{u}) + (\nabla\mathbf{u})\mathbf{u} + \nabla p = \mathbf{f} & \text{in } \Omega, \\ \nabla\cdot\mathbf{u} = 0 & \text{in } \Omega, \\ \mathbf{u} = \mathbf{0} & \text{on } \Gamma_D, \\ \boldsymbol{n}\cdot[p\mathbf{I} - \nu\nabla\mathbf{u}] = \mathbf{0} & \text{on } \Gamma_N, \end{cases} \tag{1}$$

with $\mathbf{f}$ an external forcing term per unit mass [34]. For the sake of simplicity, we refer to p and $\mathbf{f}$ as the pressure variable and the forcing term, respectively. We apply homogeneous Dirichlet and Neumann boundary conditions in the distinct portions of the domain boundary $\partial\Omega$, denoted Γ_D and Γ_N, respectively, with $\Gamma_D \neq \emptyset$ and $\Gamma_N := \partial\Omega \setminus \Gamma_N$. For simplicity, we will consider $\Gamma_D = \partial\Omega$ and $\Gamma_N = \emptyset$ from now on. Naturally, other possible boundary conditions can be employed.

We define the velocity and pressure spaces as

$$\mathbb{U} := [H_0^1(\Omega)]^2 \quad \text{and} \quad \mathbb{Q} := L_0^2(\Omega) = \{q \in L^2(\Omega) \mid (q,1)_\Omega = 0\},$$

[1] Specifically, considering a discrete spatial domain Ω, $\mathcal{O}$ may represent the whole domain, an element of the discretization, or even the boundary of the element.

and equipped with the norm $\|\boldsymbol{v}\|_{\mathbb{U}} := |\boldsymbol{v}|_{[H^1(\Omega)]^2}$ for all $\boldsymbol{v} \in \mathbb{U}$ and $\|q\|_{\mathbb{Q}} := \|q\|_{L^2(\Omega)}$ for all $q \in \mathbb{Q}$. We remark that if $\Gamma_N \neq \emptyset$, then $\mathbb{Q} := L^2(\Omega)$. Thus, the weak formulation of the model problem (1) reads as follows: find $(\mathbf{u}, p) \in \mathbb{U} \times \mathbb{Q}$ such that:

$$\begin{cases} va(\mathbf{u}, \mathbf{v}) + c(\mathbf{u}, \mathbf{u}, \mathbf{v}) + b(\mathbf{v}, p) = (\mathbf{f}, \mathbf{v})_\Omega & \forall \mathbf{v} \in \mathbb{U}, \\ b(\mathbf{u}, q) = 0 & \forall q \in \mathbb{Q}, \end{cases} \tag{2}$$

with $\mathbf{f} \in [L^2(\Omega)]^2$, and with bilinear forms $a : \mathbb{U} \times \mathbb{U} \to \mathbb{R}$, $b : \mathbb{U} \times \mathbb{Q} \to \mathbb{R}$, and trilinear form $c : \mathbb{U} \times \mathbb{U} \times \mathbb{U} \to \mathbb{R}$ defined as

$$\begin{aligned} a(\mathbf{w}, \mathbf{v}) &:= (\nabla \mathbf{w}, \nabla \mathbf{v})_\Omega, \\ b(\mathbf{w}, q) &:= -(q, \nabla \cdot \mathbf{w})_\Omega, \\ c(\mathbf{w}, \mathbf{z}, \mathbf{v}) &:= ((\nabla \mathbf{z})\mathbf{w}, \mathbf{v})_\Omega, \end{aligned}$$

for every $\mathbf{w}, \mathbf{v}, \mathbf{z} \in \mathbb{U}$ and $q \in \mathbb{Q}$.

2.2. The Smagorinsky turbulence model

The Smagorinsky model introduces an artificial diffusion to the NSE, acting as a general stabilizing term for convection-dominated flows. The key feature of this model is that it approximates the stresses of the unresolved subgrid scales (due to turbulence) by a relation between the strain-rate and an Eddy viscosity. This model is inherently discrete, and we delay the discussion of the discrete Smagorinsky model until Section 3.7. Building up on the ideas in [35], the Smagorinsky model can be reinterpreted as the discretized version of the following continuous problem: given a spatial domain Ω, find $(\mathbf{u}, p)$ such that

$$\begin{cases} -\nabla \cdot \left(\left(\nu + \nu_S(\mathbf{u}) \right) \nabla \mathbf{u} \right) + (\nabla \mathbf{u})\mathbf{u} + \nabla p = \boldsymbol{f} & \text{in } \Omega, \\ \nabla \cdot \mathbf{u} = 0 & \text{in } \Omega, \\ \mathbf{u} = \mathbf{0} & \text{on } \partial\Omega. \end{cases} \tag{3}$$

Namely, compared to the strong formulation of NSE in (1), an Eddy viscosity term $\nu_S(\mathbf{u})$ is added. Let $\mathcal{T}$ denote a tessellation of the domain Ω, decomposed in elements denoted by T with diameter h_T. Thus, the Eddy viscosity term in the Smagorinsky model is defined as $\nu_S : \mathbb{U} \to \mathbb{R}$ such that

$$\nu_S(\mathbf{u}) = C_S^2 \sum_{T \in \mathcal{T}_h} h_T^2 \left| \nabla \mathbf{u}_{|T} \right|_{\ell^2} \chi_T, \tag{4}$$

where χ_T is the characteristic function over the tessellation element, while C_S is the Smagorinsky constant [36], which is classically set (guided by numerical experiments [37]) as $C_S = 0.1$: in this paper we follow this standard choice. The Smagorinsky term (4) is strictly related to the discretization of the space. We remark that for $h_T \to 0$, the Eddy viscosity term $\nu_S(\mathbf{u}) \to 0$. This totally complies with the Kolmogorov scale turbulence description: a refined mesh can capture the complex behavior of the flow and does not need stabilization, which is needed for a larger meshsize. In the next section, we specify the approximation by employing the VEM numerical discretization.

3. VEM approximation

We briefly present the enhanced formulation for the divergence-free VEM introduced in [33]. We first give some preliminaries regarding the mesh assumptions and the discrete setting.

3.1. Mesh assumptions

Let $(\mathcal{T}_h)_h$ denote a h-refined mesh sequence of Ω, with $\mathcal{T}_h$ being a decomposition into a tessellation with mesh elements denoted by T. The boundary and faces of the element are denoted by ∂T and F, respectively. We remark that in this paper, we use the word face to denote the edges of the polygon. The notation h_T and h_F stands for the elemental and the face diameter, respectively. We define the global meshsize $h := \max_{T \in \mathcal{T}_h} h_T$. Furthermore, we assume that there exists a real number $\rho > 0$, such that:

1. for all $\mathcal{T}_h$ in the sequence, for all $T \in \mathcal{T}_h$, T is star-shaped with respect to a ball of radius $r_B \geq \rho h_T$ (star-shaped property);
2. for all $T \in \mathcal{T}_h$, and for any face F of T, h_F is comparable with h_T as $\rho h_T \leq h_F \leq h_T$ (shape regularity property).

Alternatively, relaxed mesh assumptions can be adopted; we refer the reader to [38,39] for further details.

3.2. Polynomial decomposition

Let $\mathbb{P}^k(\mathcal{O})$ denote the space of polynomials on a domain $\mathcal{O}$ of degree up to k. A key idea in devising projectors in VEM for the vector formulation is to decompose vector polynomials over a cell T into a gradient component and their orthogonal component. Thus, polynomial decomposition reads as

$$[\mathbb{P}^k(T)]^2 = \mathcal{G}^k(T) \oplus \mathcal{G}^{\perp,k}(T),$$

where $\mathcal{G}^k(T) := \nabla\mathbb{P}^{k+1}(T)$ and $\mathcal{G}^{\perp,k}(T) := \mathbf{x}^\perp[\mathbb{P}^{k-1}(T)]$ with $\mathbf{x}^\perp := (y, -x)$ [40]. Namely, by the multiplication action of $\mathbf{x}^\perp$, we build the orthogonal component to the gradients. The scalar scaled monomial $m_{\boldsymbol{\alpha}} \in \mathbb{P}^k(T)$ with $|\boldsymbol{\alpha}|_{\ell^2} \leq k$ is defined as:

$$m_{\boldsymbol{\alpha}} := \prod_{i=1}^{2} \left(\frac{x_i - x_i^T}{h_T} \right)^{\alpha_i}, \tag{5}$$

where we use the classic notation of the multi-index $\boldsymbol{\alpha} := (\alpha_1, \alpha_2)$ and we denote with x_i^T the ith coordinate of the barycenter of T. Vector scaled monomials are defined as $\boldsymbol{m}_{(\boldsymbol{\alpha}_x, \boldsymbol{\alpha}_y)} := (m_{\boldsymbol{\alpha}_x}, m_{\boldsymbol{\alpha}_y}) \in [\mathbb{P}^k(T)]^2$.

Remark 1 (*Vector Monomial Decomposition and Scaling*)**.** Explicit closed forms of the polynomial decomposition can be obtained for vector scale monomials, see [40] for a complete presentation. For the sake of completeness, we recall the 2D case: let $\boldsymbol{\mathcal{B}} := (\beta_x, \beta_y)$, be a vector gathering the coefficients β_x, β_y for each direction, then

$$\boldsymbol{m}_{(\boldsymbol{\mathcal{B}},\mathbf{0})} = \frac{h_T}{(|\boldsymbol{\mathcal{B}}|_{\ell^2}+1)} \nabla m_{(\beta_x+1,\beta_y)} + \frac{\beta_y}{(|\boldsymbol{\mathcal{B}}|_{\ell^2}+1)} \mathbf{x}^\perp m_{(\beta_x,\beta_y-1)},$$
$$\boldsymbol{m}_{(\mathbf{0},\boldsymbol{\mathcal{B}})} = \frac{h_T}{(|\boldsymbol{\mathcal{B}}|_{\ell^2}+1)} \nabla m_{(\beta_x,\beta_y+1)} - \frac{\beta_x}{(|\boldsymbol{\mathcal{B}}|_{\ell^2}+1)} \mathbf{x}^\perp m_{(\beta_x-1,\beta_y)}.$$

See also [40] for detailed guidelines on how to build VEM. Moreover, a recent review on the proper scaling of normalized monomials and its influence on the conditioning was reported in [41].

A monomial scalar and vector basis are denoted as $\mathcal{M}^k(T) := \left\{ m_{\boldsymbol{\alpha}} : |\boldsymbol{\alpha}|_{\ell^2} \leq k \right\}$ and $[\mathcal{M}^k(T)]^2$, respectively.

3.3. Local spaces

Let a polynomial of degree $k \geq 2$ be fixed. The local virtual element space introduced originally in [31] is

$$\begin{aligned} \mathcal{U}_T^k := \{ \mathbf{v} \in [H^1(T)]^2 \,|\, (i) \quad & \mathbf{v}_{|\partial T} \in [\mathbb{B}^k(T)]^2, \\ (ii) \quad & -\boldsymbol{\Delta}\mathbf{v} - \nabla q \in \mathcal{G}^{\perp,k}(T) \text{ for some } q \in L^2(T), \\ (iii) \quad & \nabla \cdot \mathbf{v} \in \mathbb{P}^{k-1}(T) \}, \end{aligned}$$

where the first condition in the virtual space concerns the conformity prescription on the boundary with

$$\mathbb{B}^k(T) := \{ v \in C^0(\partial T) | v_{|F} \in \mathbb{P}^k(F), \forall F \subset \partial T \}.$$

Then, the enhanced local virtual space introduced in [33] is defined as

$$\mathcal{V}_T^k := \{ \mathbf{v} \in \mathcal{U}_T^k | (\mathbf{v} - \Pi_k^\nabla \mathbf{v}, \mathbf{g}^\perp)_T = 0, \forall \mathbf{g}^\perp \in \mathcal{G}^{\perp,k}(T) \setminus \mathcal{G}^{\perp,k-2}(T) \}. \tag{6}$$

The enhancement uses an additional condition, formulated in terms of the VE projector Π_k^∇, that enables the computation of higher-order projectors, e.g., the L^2-projector of order k, $\Pi_k^0 \mathbf{v}_h$. The definition of the local projectors is postponed to Section 3.5.

Let $\{ \boldsymbol{\varphi}^i \}_{1 \leq i \leq dim(\mathcal{V}_T^k)}$ be a basis of $\mathcal{V}_T^k$ made of virtual functions which are not explicitly known. One property of $\mathcal{V}_T^k$ is the polynomial inclusion $[\mathbb{P}^k(T)]^2 \subseteq \mathcal{V}_T^k$.

3.4. Degrees of freedom (DoFs)

Let $N_{dof} := dim(\mathcal{V}_T^k)$ denote the local number of degrees of freedom. Let N^v denote the number of vertices in a polygon T and $N^\circ = 2N^v$ be the number of DoFs related to the vertices. Let $N^- := 2N^v(k-1)$ denote the number of DoFs set on ∂T, excluding vertices, let $N^\partial := N^\circ + N^-$ denote the total DoFs on ∂T. Moreover, let $N^{\nabla\cdot}$ and N^Δ denote the number of DoFs related to the divergence and the Laplacian operators, respectively. The latter two are integrals evaluated inside the element acting as DoFs. The velocity local degrees of freedom function $d\circ\ell_l^T : \mathcal{V}_T^k \to \mathbb{R}^{N_{dof}}$ is defined as:

$$d\circ\ell_l^T(\mathbf{v}) := \begin{cases} \mathbf{v}\left(\mathbf{x}_{T,i}^\circ\right)_d, & i \in (1 : \frac{1}{2}N^\circ),\ l = \frac{1}{2}N^\circ(d-1)+i, \\ \mathbf{v}\left(\mathbf{x}_{T,i}^-\right)_d, & i \in (1 : \frac{1}{2}N^-),\ l = \frac{1}{2}N^-(d-1)+N^\circ+i, \\ (\nabla\cdot\mathbf{v}, m_i)_T, & i \in (1 : N^{\nabla\cdot}),\ l = N^\partial + i, \\ (\mathbf{v}, \boldsymbol{m}_i)_T, & i \in (1 : N^\Delta),\ l = N^\partial + N^{\nabla\cdot} + i, \end{cases}$$

where $d \in \{1, 2\}$ denotes the vector component index, for all $m_i \in \mathcal{M}^{k-1}(T) \setminus \{1\}$ and $i \in (1 : N^{\nabla\cdot})$, for all $\boldsymbol{m}_i \in \mathbf{x}^\perp\mathcal{M}^{k-3}(T)$ and $i \in (1 : N^\Delta)$, and for all $\mathbf{v} \in \mathcal{V}_T^k$ and $l \in (1 : N_{dof})$, with $\mathbf{x}_{T,i}^\circ$ the ith vertex of the polygon T and $\mathbf{x}_{T,i}^-$ the ith edge point, usually a Gauss–Lobatto quadrature point (excluding vertices) located at the edge. The advantage of directly using the values at the quadrature rule points as DoFs is the ease of computing the integrals. An important feature of the virtual basis is that $d\circ\ell_l^T(\boldsymbol{\varphi}^j) = \delta_{lj}$.

3.5. Local projectors

For all $T \in \mathcal{T}_h$, the H^1-projection operator $\Pi_k^\nabla : \mathcal{V}_T^k \to [\mathbb{P}^k(T)]^2$ is defined such that, for all $\mathbf{v}_h \in \mathcal{V}_T^k$,

$$\begin{cases} (\nabla \Pi_k^\nabla \mathbf{v}_h, \nabla \mathbf{p})_T = (\nabla \mathbf{v}_h, \nabla \mathbf{p})_T & \forall\, \mathbf{p} \in [\mathbb{P}^k(T)]^2 \setminus [\mathbb{P}^0(T)]^2, \\ (\Pi_k^\nabla \mathbf{v}_h, \mathbf{p})_{\partial T} = (\mathbf{v}_h, \mathbf{p})_{\partial T} & \forall\, \mathbf{p} \in [\mathbb{P}^0(T)]^2. \end{cases} \tag{7}$$

In a similar fashion, the L^2-projection operator $\Pi_k^0 : \mathcal{V}_T^k \to [\mathbb{P}^k(T)]^2$ is obtained by solving the following problem: for all $\mathbf{v}_h \in \mathcal{V}_T^k$,

$$(\Pi_k^0 \mathbf{v}_h, \mathbf{p})_T = (\mathbf{v}_h, \mathbf{p})_T \qquad \forall\, \mathbf{p} \in [\mathbb{P}^k(T)]^2. \tag{8}$$

Finally, we define the L^2-projector for tensors, specifically for the velocity gradient, $\boldsymbol{\Pi}_{k-1}^0 \nabla : \mathcal{V}_T^k \to [\mathbb{P}^{k-1}(T)]^{2\times 2}$ such that, for all $\mathbf{v}_h \in \mathcal{V}_T^k$

$$(\boldsymbol{\Pi}_{k-1}^0 \nabla \mathbf{v}_h, \boldsymbol{\tau})_T = (\nabla \mathbf{v}_h, \boldsymbol{\tau})_T \qquad \forall \boldsymbol{\tau} \in [\mathbb{P}^{k-1}(T)]^{2\times 2}. \tag{9}$$

3.6. Discrete global weak formulation

Let $k > 1$ be the order of VEM, then the global enhanced spaces are defined as follows

$$\mathcal{V}_h^k := \{\mathbf{v}_h \in [H_0^1(\Omega)]^2 | \mathbf{v}_{h|T} \in \mathcal{V}_T^k, \forall T \in \mathcal{T}_h\},$$
$$P_h^{k-1} := \{q_h \in L_0^2(\Omega) | q_{h|T} \in \mathbb{P}^{k-1}(T), \forall T \in \mathcal{T}_h\}.$$

Notice that the conformity prescription is only applied to the discrete velocity, while the discrete global pressure is a piecewise polynomial that can jump. The discrete version of the weak problem reads as: find $(\mathbf{u}_h, p_h) \in \mathcal{V}_h^k \times P_h^{k-1}$ such that

$$\begin{cases} va_h(\mathbf{u}_h, \mathbf{v}_h) + c_h(\mathbf{u}_h, \mathbf{u}_h, \mathbf{v}_h) + b_h(\mathbf{v}_h, p_h) = l_h(\mathbf{v}_h), & \forall \mathbf{v}_h \in \mathcal{V}_h^k, \\ b_h(\mathbf{u}_h, q_h) = 0, & \forall q_h \in P_h^{k-1}, \end{cases} \tag{10}$$

for the global linear form $l_h : \mathcal{V}_h^k \to \mathbb{R}$ formulated as

$$l_h(\mathbf{v}_h) := \sum_{T \in \mathcal{T}_h} (\mathbf{f}, \Pi_k^0 \mathbf{v}_h)_T,$$

and the global bilinear forms $a_h : \mathcal{V}_h^k \times \mathcal{V}_h^k \to \mathbb{R}$ and $b_h : \mathcal{V}_h^k \times P_h^{k-1} \to \mathbb{R}$ consisting of the collection of local contributions as

$$a_h(\mathbf{w}_h, \mathbf{v}_h) := \sum_{T \in \mathcal{T}_h} (\nabla \Pi_k^\nabla \mathbf{w}_h, \nabla \Pi_k^\nabla \mathbf{v}_h)_T + s_h(\mathbf{w}_h, \mathbf{v}_h),$$
$$b_h(\mathbf{v}_h, q_h) := -\sum_{T \in \mathcal{T}_h} (q_h, \nabla \cdot \mathbf{v}_h)_T.$$

The stabilization term s_h is needed to ensure coercivity in the discrete system. We use the standard dofi-dofi stabilization $s_h : \mathcal{V}_h^k \times \mathcal{V}_h^k \to \mathbb{R}$ defined as

$$s_h(\mathbf{w}_h, \mathbf{v}_h) := \sum_{T \in \mathcal{T}_h} \sum_{l}^{N_{dof}} dof_l^T((\mathbf{I} - \Pi_k^\nabla)\mathbf{w}_h) dof_l^T((\mathbf{I} - \Pi_k^\nabla)\mathbf{v}_h),$$

where $\mathbf{I}$ denotes the identity matrix. The trilinear form related with the nonlinear term $c_h : \mathcal{V}_h^k \times \mathcal{V}_h^k \times \mathcal{V}_h^k \to \mathbb{R}$ is defined such that, for all $(\mathbf{w}_h, \mathbf{x}_h, \mathbf{v}_h) \in \mathcal{V}_h^k \times \mathcal{V}_h^k \times \mathcal{V}_h^k$,

$$c_h(\mathbf{w}_h, \mathbf{x}_h, \mathbf{v}_h) := \sum_{T \in \mathcal{T}_h} ((\boldsymbol{\Pi}_{k-1}^0 \nabla \mathbf{x}_h) \Pi_k^0 \mathbf{w}_h, \Pi_k^0 \mathbf{v}_h)_T. \tag{11}$$

Different configurations of linear forms can arise from the various available projectors. We use here the original setting presented in [33].

3.7. The discrete Smagorinsky model

The discretized Smagorinsky form $a_h^S : \mathcal{V}_h^k \times \mathcal{V}_h^k \times \mathcal{V}_h^k \to \mathbb{R}$, in a VEM framework, is written as follows

$$a_h^S(\mathbf{w}_h; \mathbf{x}_h, \mathbf{v}_h) := \sum_{T \in \mathcal{T}_h} (\nu_h^S(\mathbf{w}_h) \nabla \Pi_k^\nabla \mathbf{x}_h, \nabla \Pi_k^\nabla \mathbf{v}_h)_T,$$

where we have also introduced the discrete version of the Smagorinsky viscosity adapted to VEM and defined as $\nu_h^S : \mathcal{V}_h^k \to \mathbb{R}$ such that

$$\nu_h^S(\mathbf{w}_h) := C_S^2 \sum_{T \in \mathcal{T}_h} h_T^2 |\boldsymbol{\Pi}_{k-1}^0 \nabla \mathbf{w}_h|_{\ell^2} \chi_T. \tag{12}$$

The global problem, including the Smagorinsky diffusive term, has the following form: find $(\mathbf{u}_h, p_h) \in \mathcal{V}_h^k \times P_h^{k-1}$ such that

$$\begin{cases} \nu a_h(\mathbf{u}_h, \mathbf{v}_h) + a_h^S(\mathbf{u}_h; \mathbf{u}_h, \mathbf{v}_h) + c_h(\mathbf{u}_h, \mathbf{u}_h, \mathbf{v}_h) + b_h(\mathbf{v}_h, p_h) = l_h(\mathbf{v}_h) & \forall \mathbf{v}_h \in \mathcal{V}_h^k, \\ b_h(\mathbf{u}_h, q_h) = 0 & \forall q_h \in P_h^{k-1}. \end{cases} \tag{13}$$

We use the shorthand notation $\boldsymbol{\mathcal{F}}_h^{NS}(\mathbf{u}_h, p_h) = \mathbf{0}$ for the Navier–Stokes problem (10) and $\boldsymbol{\mathcal{F}}_h(\mathbf{u}_h, p_h) = \mathbf{0}$ for the global problem including the Smagorinsky term (13). To solve the nonlinear global problem, we use the Newton scheme detailed in Algorithm 1. For completeness, we write the directional derivative of the Smagorinsky term along the direction $\delta\mathbf{u}_h$, readily verified to be of the following form:

$$\partial_{\mathbf{u}_h} a_h^S(\mathbf{u}_h; \mathbf{u}_h, \mathbf{v}_h)[\delta\mathbf{u}_h] = \sum_{T\in\mathcal{T}_h} \left(\nu_h^S(\mathbf{u}_h)\nabla\Pi_k^\nabla \delta\mathbf{u}_h, \nabla\Pi_k^\nabla \mathbf{v}_h\right)_T + C_S^2 \sum_{T\in\mathcal{T}_h} h_T^2 \left(t, (\nabla\Pi_k^\nabla \mathbf{u}_h : \nabla\Pi_k^\nabla \mathbf{v}_h)\right)_T,$$

with $t = (\boldsymbol{\Pi}_{k-1}^0 \nabla\mathbf{u}_h : \boldsymbol{\Pi}_{k-1}^0 \nabla\delta\mathbf{u}_h)/|\boldsymbol{\Pi}_{k-1}^0 \nabla\mathbf{u}_h|_{\ell^2}$.

Algorithm 1 The Newton solver for the Smagorinsky model

1: Choose $\mathbf{U}_h^0 := (\mathbf{u}_h{}^0, p_h^0)$
2: Choose $\epsilon > 0$, set $\boldsymbol{R}_h = \infty$ and $m = 0$
3: **while** $||\boldsymbol{R}_h|| > \epsilon$ **do**
4: Seek $\boldsymbol{\delta}_h := (\delta\mathbf{u}_h, \delta p_h)$ solving for all $(\mathbf{v}_h, q_h) \in \mathcal{V}_h^k \times P_h^{k-1}$ such that

$$\partial_{\mathbf{U}_h} \boldsymbol{\mathcal{F}}_h[\boldsymbol{\delta}_h] = -\boldsymbol{\mathcal{F}}_h,$$

or explicitly written

$$\begin{cases} \partial_{\mathbf{u}_h} \boldsymbol{\mathcal{F}}_h^{NS}[\boldsymbol{\delta}_h] + \partial_{\mathbf{u}_h} a_h^S(\mathbf{u}_h{}^m; \mathbf{u}_h{}^m, \mathbf{v}_h)[\boldsymbol{\delta}_h] & = -R_1^s(\mathbf{U}_h^m), \\ \partial_{p_h} \boldsymbol{\mathcal{F}}_h^{NS}[\boldsymbol{\delta}_h] & = -R_2(\mathbf{U}_h^m), \end{cases}$$

where the residual is defined as $\boldsymbol{R}_h := (R_1^s(\mathbf{U}_h), R_2(\mathbf{U}_h))$ and

$$\begin{aligned} R_1^s(\mathbf{U}_h) &:= R_1(\mathbf{U}_h) + a_h^S(\mathbf{u}_h; \mathbf{u}_h, \mathbf{v}_h), \\ R_1(\mathbf{U}_h) &:= c_h(\mathbf{u}_h, \mathbf{u}_h, \mathbf{v}_h) + a_h(\mathbf{u}_h, \mathbf{v}_h) + b_h(\mathbf{v}_h, p_h) - l_h(\mathbf{v}_h), \\ R_2(\mathbf{U}_h) &:= b_h(\mathbf{u}_h, q_h). \end{aligned}$$

5: Update the solution $\mathbf{U}_h^{m+1} = \mathbf{U}_h^m + \boldsymbol{\delta}_h$
6: $m = m + 1$
7: **end while**

4. Numerical results

In this section, we illustrate numerical results for the Smagorinsky model combined with the VEM approximation. We will show that the Smagorinsky model is necessary to simulate these Reynolds numbers on coarse meshes. Indeed, we conducted some numerical tests solving the standard NSE using either VEM (employing an in-house code based on [42]) or FEM (employing the FEniCS library [43]) and the Newton solver did not converge in the Reynolds number range $1000 \le Re \le 2000$.

We consider first some test cases assessing the accuracy of the model against manufactured or analytical solutions, and second, we consider the benchmark of the lid-driven cavity for increasing Reynolds numbers (up to $Re = 10000$). All problems are solved in the unitary square Ω. The following metrics will be used for the computation of the errors for the virtual element approximation of velocity and pressure

$$\|\nabla\mathbf{u} - \boldsymbol{\Pi}_{k-1}^0 \nabla\mathbf{u}_h\|_{\mathcal{T}_h} := \left(\sum_{T\in\mathcal{T}_h} \|\nabla\mathbf{u} - \boldsymbol{\Pi}_{k-1}^0 \nabla\mathbf{u}_h\|_{[L^2(T)]^{2\times 2}}^2\right)^{\frac{1}{2}},$$

$$\|\mathbf{u} - \Pi_k^0 \mathbf{u}_h\|_{\mathcal{T}_h} := \left(\sum_{T\in\mathcal{T}_h} \|\mathbf{u} - \Pi_k^0 \mathbf{u}_h\|_{[L^2(T)]^2}^2\right)^{\frac{1}{2}},$$

$$\|p - p_h\|_{\mathcal{T}_h} := \left(\sum_{T\in\mathcal{T}_h} \|p - p_h\|_{L^2(T)}^2\right)^{\frac{1}{2}}.$$

4.1. Accuracy tests

We consider two test cases here labeled as the irrotational force test and the polynomial P2P1 test. Let us detail the data of the two experiments:

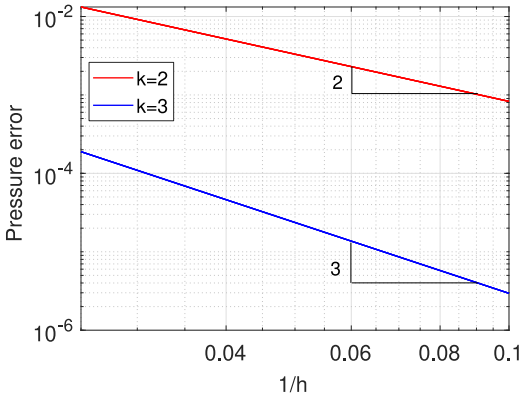


Fig. 1. *Irrotational force test.* Pressure errors $\|p - p_h\|_{\mathcal{T}_h}$ for $Re = 1$ and $k \in \{2, 3\}$.

Table 1
Polynomial P2P1 test. Errors and convergence rates using the discrete Smagorinsky model, $Re = 10000$ and $k = 2$.

h	Velocity				Pressure	
	$\|\nabla \mathbf{u} - \boldsymbol{\Pi}^0_{k-1} \nabla \mathbf{u}_h\|_{\mathcal{T}_h}$	rate	$\|\mathbf{u} - \Pi^0_k \mathbf{u}_h\|_{\mathcal{T}_h}$	rate	$\|p - p_h\|_{\mathcal{T}_h}$	rate
0.100	7.3571E−03		4.3530E−04		3.8144E−05	
0.050	1.7860E−03	2.043	4.1754E−05	3.382	6.5666E−06	2.539
0.025	4.8040E−04	1.894	5.7269E−06	2.866	1.7309E−06	1.924

1. *Irrotational force test*: let $(\mathbf{u}, p) \in [\mathbb{P}^2(\Omega)]^2 \times \mathbb{P}^3(\Omega)$ with velocity $\mathbf{u}(\mathbf{x}) = (-y, x)$, pressure $p(\mathbf{x}) = \lambda x^3 + 0.5(x^2 + y^2) - 1/3 - \lambda/4$, force $\mathbf{f}(\mathbf{x}) = (3\lambda x^2, 0)$ and $\lambda \geq 0$. We also set the parameter $\lambda = 10$ and the viscosity $\nu = 1$. The velocity gradient is constant throughout the domain.
2. *Polynomial P2P1 test*: let $(\mathbf{u}, p) \in [\mathbb{P}^2(\Omega)]^2 \times \mathbb{P}^1(\Omega)$ with velocity $\mathbf{u}(\mathbf{x}) = (-x + y^2, y - x^2)$, pressure $p(\mathbf{x}) = 2x - 2y$ and the corresponding $\mathbf{f}(\mathbf{x})$ obtained inserting the expression in Eq. (3) for $Re = 10000$.

For clarity in the results analysis, we briefly recall the error estimates from [33] for the norms defined in Section 2.1: assuming $\mathbf{u}, \mathbf{f} \in [H^{s+1}(\Omega)]^2$, with $0 < s \leq k$, then

$$\|\mathbf{u} - \mathbf{u}_h\|_{\mathbb{U}} \leq h^s \mathcal{J}(\mathbf{u}; \nu) + h^{s+2} \mathcal{H}(\mathbf{f}, \nu), \tag{14}$$

$$\|p - p_h\|_{\mathbb{Q}} \leq C h^s |p|_{H^s(\Omega)} + C h^{s+2} |\mathbf{f}|_{[H^{s+1}(\Omega)]^2} + C h^s \mathcal{K}(\mathbf{u}; \nu), \tag{15}$$

with the functions $\mathcal{J}$, $\mathcal{H}$, $\mathcal{K}$, and the constant C independent of the meshsize. In particular, $\mathcal{J}$ and $\mathcal{H}$ depend on $|\mathbf{u}|_{[H^1(\Omega)]^2}$ and $|\mathbf{f}|_{[H^{s+1}(\Omega)]^2}$, respectively. In the *Irrotational Force Test*, where both the velocity field and the forcing term are polynomials of degree less than or equal to k, we expect the discrete velocity to be reproduced exactly. On the other hand, this property is not expected in the *Polynomial P2P1 Test* due to the presence of the Smagorinsky term. Finally, we remark that, in these convergence examples, we employ the estimates for the Navier–Stokes equations proposed in [33], since the Smagorinsky term vanishes for sufficiently small meshsize.

We consider uniformly refined mesh sequences. At each recurrence of the mesh refinement, the forcing term can be obtained from the analytical solution using the Smagorinsky model in its strong form (3).

Therefore, for each mesh, we compare the corresponding manufactured analytical solution and the numerical solution. Since we expect the Smagorinsky term to vanish as the mesh size tends to zero due to (12), we also expect the error to converge to the physical analytical solution as in (14). In the irrotational force test, provided that uniform meshes are used, and since the velocity gradient is constant, ν_S remains constant. Since the Laplacian and the gradient of ν_S vanish, the force balances only with the pressure gradient and the convective term. This implies that the physical solution and the manufactured problem coincide, with the latter being independent of h_T.

In Fig. 1 and Table 1, we report the errors and rates of convergence for the irrotational force test and the polynomial P2P1 test cases, respectively. In the irrotational case, we solve using degree $k \in \{2, 3, 4\}$, but we show pressure errors only for $k = 2$ and $k = 3$, as errors for $k = 4$ were close to machine precision, as expected. For the same reason, velocity errors for all degrees were omitted. We recall that VEM approximates exactly pressures in $\mathbb{P}^{k-1}(\Omega)$, which for $k = 4$ it coincides with the polynomial space of the exact pressure. On the other hand, in the second case, the force $\mathbf{f}$ includes non-polynomial components due to the derivative of the Smagorinsky term. This explains the higher velocity errors, although velocity is a polynomial in $[\mathbb{P}^2(\Omega)]^2$. This is consistent with (14) and (15).

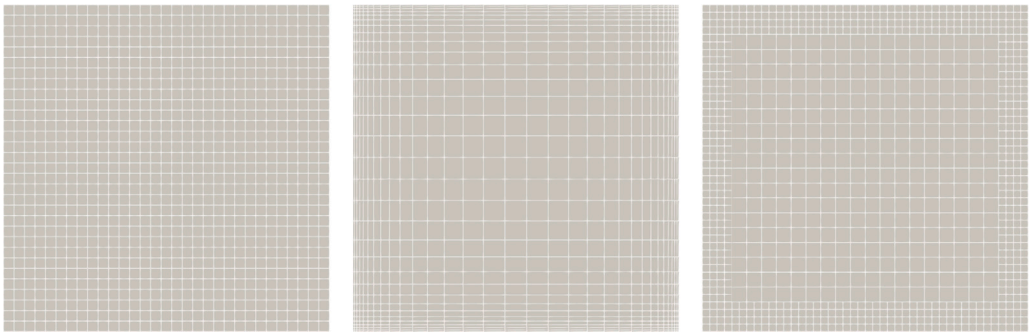


Fig. 2. *Lid-driven cavity flow*. Mesh visualization: 31 × 31 USM (left), 31 × 31 ARM (center), and IMH based on a 22 × 22 USM (right).

4.2. Lid-driven cavity flow

For this test case, we focus on a more applied yet established test case in the turbulence model community. The main goal is to provide a first numerical investigation on how VEM approximation performs in a turbulent setting for several Reynolds numbers, highlighting its effectiveness and strengths.

In this cavity flow test, no-slip boundary conditions are prescribed in Ω for the vertical and the bottom walls, while on the top wall a unitary horizontal velocity is imposed. The external body force is set as $\mathbf{f} = \mathbf{0}$. The transition regime from laminar to turbulent is "expected" in the range $6000 \leq Re \leq 8000$, see [44] and the references therein. To illustrate the benefits of using VEM, we consider $k = 2$ and three different families of meshes: the Uniform Square Meshes (USM), Anisotropic Rectangular Meshes (ARM), and Isotropic Meshes with Hanging nodes (IMH). These families are depicted in Fig. 2, while their description at each refinement level is listed in Table 2. We remark that the comparison among families is conducted by maintaining a comparable number of DoFs at each refinement level (labeled as Coarse, Medium, and Fine). On one hand, the ARMs are obtained by partitioning each side of the domain according to the following function, as in [45]:

$$f(\xi) = 0.5\left(1 + \frac{\tanh(4\xi - 2)}{\tanh(2)}\right). \tag{16}$$

On the other hand, the IMHs refinement is based on sequences of meshes $(\mathcal{T}_l)_{l>0}$ generated iteratively thanks to the following strategy: let $\mathcal{T}_0$ be a uniform squared mesh generated by a discrete parameter $h_0 \in (0, 1]$. We generate the coarsest mesh with hanging nodes of the sequence $\mathcal{T}_1$ by splitting by four all the squares whose maximum distance to the boundary in normal direction is $\delta_0 \in [h_0, 0.5-h_0)$. For our experiments we use $\delta = 0.175$. The consecutive meshes for $\mathcal{T}_l$ are generated using $\delta = \delta_0/2^l$. The transitions from fine to coarse cells are obtained by using hanging nodes. This approach allows for varying mesh sizes without unnecessarily propagating the refinement throughout the domain, as in the uniform case. We remark that the hanging nodes in the IMHs are easily amenable to VEM, which does not require special treatments. More precisely, the original face is just treated as two collinear faces divided by the hanging node, which is treated as another vertex carrying DoFs. Here, we mainly use the first level for three different initial meshes, see Table 2 for more details.

In the literature, the ARMs are commonly used to achieve higher accuracy near the domain walls, where high gradients and small scales are expected as the Reynolds number increases. However, as one can observe from the middle plot of Fig. 2, some elements are much stretched, and this anisotropy of the elements near the boundary of the domain might foster the overestimation of the Smagorinsky viscosity near the walls. Indeed, the Smagorinsky model can be defined as *isotropic*: the model's scaling h_T does not take into account the element anisotropy, treating every direction of the cell similarly. In this specific boundary layer, the cell diameter h_T of each element of the layer is, roughly speaking, comparable to the measure of its largest face. This phenomenon is visible in Fig. 3, where we plot the x- and y- components of the velocity. The ARM solution (blue solid and blue dotted lines) does not match the Erturk et al. reference values (green stars) [44] for $Re = 1000$ (coarse mesh), $Re = 2500$ (medium mesh), and $Re = 3200$ (fine mesh). We show the highest Reynolds number that each mesh can resolve, and for which reference data is available for comparison. We conclude that ARMs are not practical in this setting. Alternatively, uniform meshes, as USM, would require a significantly high number of elements to adequately resolve localized features such as sharp gradients and boundary layers, leading to unnecessary computational cost. Instead, a non-uniform mesh allows for targeted refinement in critical areas, improving accuracy while maintaining overall efficiency. Thus, to assess the turbulent model without the influence of the mesh anisotropy and focusing on the boundary layer refinement, we use the IMH family of meshes.

From Fig. 3, we observe that IMH results (red solid and red dotted lines) better match the reference values with respect to ARM results. We do not report the results for USM since they were comparable to IMH in terms of accuracy. This is expected since the uniform mesh has a good description of the boundary layer, but increases the number of DoFs, more than twice with respect to the finer mesh, i.e., IMH presents 31887 DoFs, while USM presents 69763 DoFs. This is caused by a non-necessary refinement on the central region of the domain.

In Fig. 4, we show the streamlines for a laminar case with $Re = 3200$ and the finest mesh for each mesh family. We qualitatively compare the results and observe that in the case of ARM, it shows the biggest displacements of the vortices with respect to the

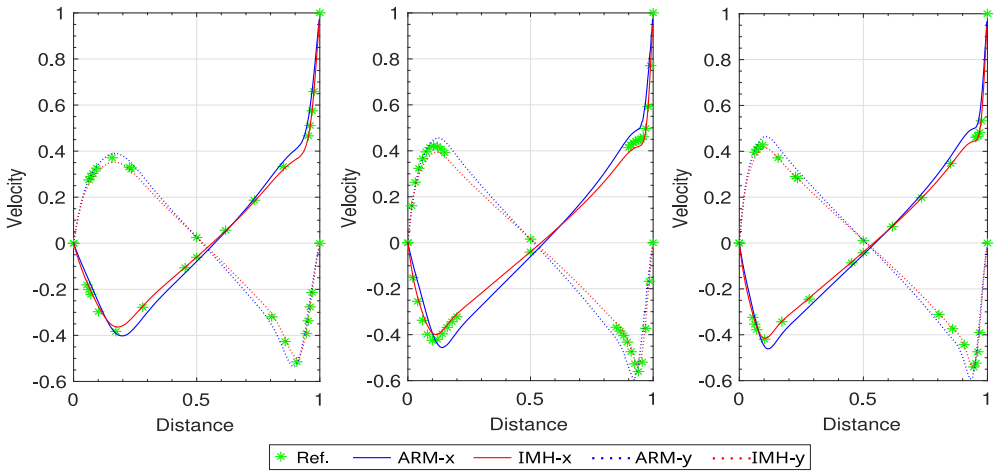


Fig. 3. *Lid-driven cavity flow*. Velocity profiles with various Reynolds numbers: $Re = 1000$ using coarse meshes (left), $Re = 2500$ using medium meshes (center), and $Re = 3200$ using fine meshes (right). The solid and dotted lines represent the x- and y-components of the velocity, respectively.

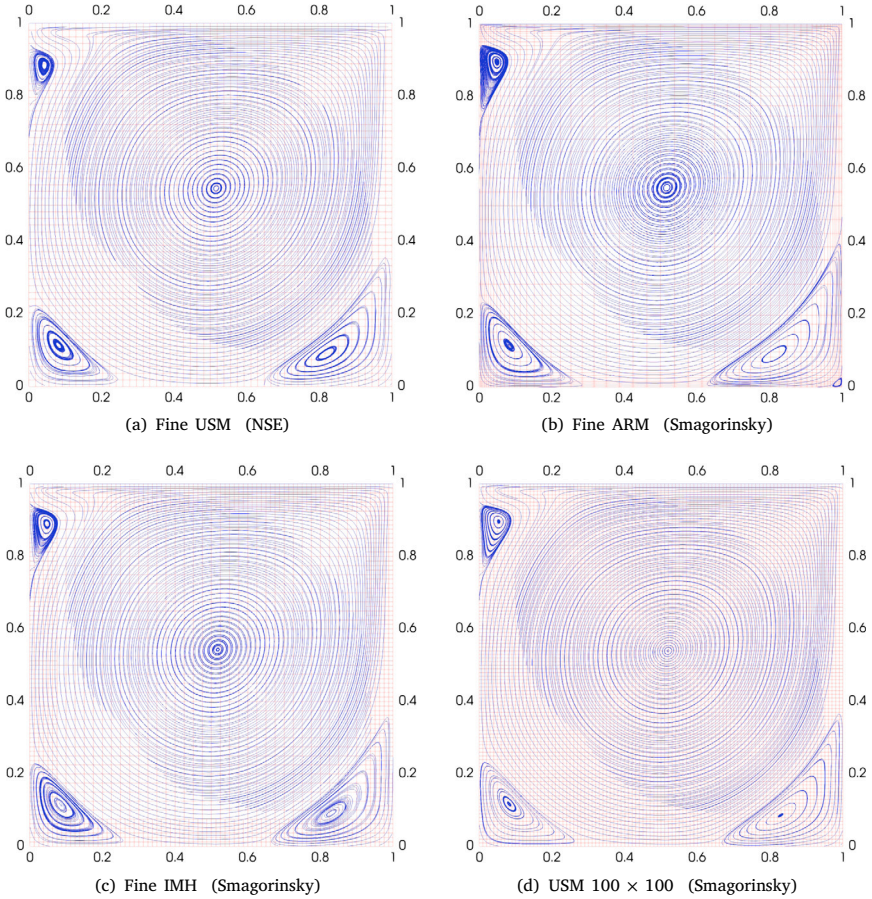


Fig. 4. *Lid-driven cavity flow*. Streamlines for $Re = 3200$ with the Smagorinsky model using the three fine meshes, and the USM 100×100 with NSE for comparison (top left).

Table 2
Lid-driven cavity flow. Mesh setting description, where N denotes the number of segments into which the domain is partitioned in each direction.

	IMH		USM/ARM	
	$N \times N$	# DoFs	$N \times N$	# DoFs
Coarse	22 × 22	10 399	31 × 31	10 526
Medium	28 × 28	15 235	38 × 38	15 583
Fine	40 × 40	31 887	54 × 54	31 647

Table 3
Lid-driven cavity flow. Relative L^2-errors for the velocity variable against the numerical reference solution varying the Re and considering USM.

Mesh	Re		
	5000	7500	10000
Coarse	3.071E−01	–	–
Medium	2.490E−01	3.264E−01	–
Fine	1.809E−01	2.433E−01	2.941E−01

Table 4
Lid-driven cavity flow. Relative L^2-errors for the velocity variable against the numerical reference solution varying the Re and considering IMH.

Mesh	Re		
	5000	7500	10000
Coarse	1.853E−01	2.399E−01	–
Medium	1.449E−01	1.942E−01	2.351E−01
Fine	1.048E−01	1.495E−01	1.868E−01

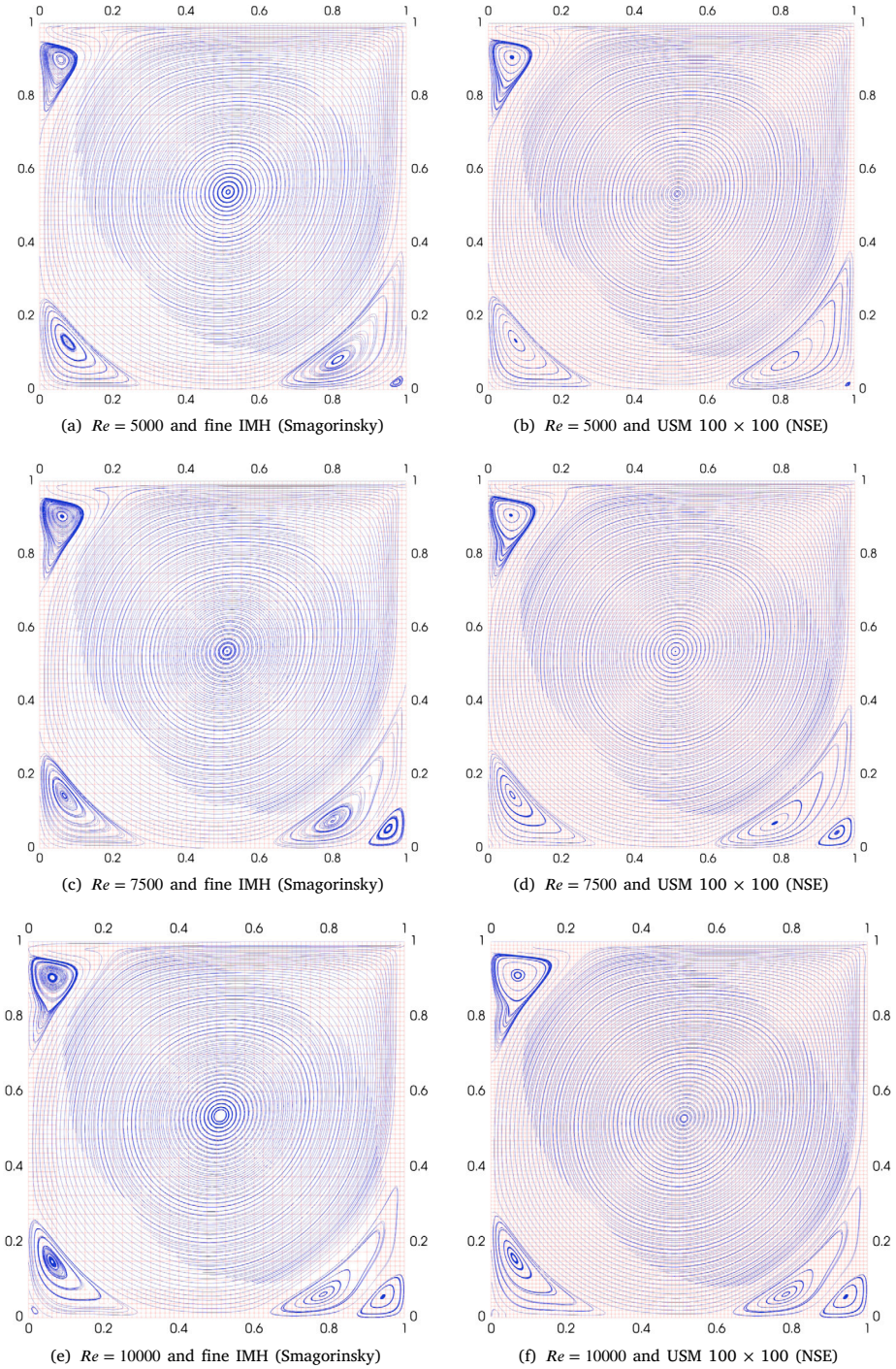
reference, which are twice and even three times the displacements shown by the IMH or USM. In addition, there is a spurious vortex in the bottom-right corner. In Fig. 5, we show the streamlines plot for $Re \in \{5000, 7500, 10000\}$, using the fine IMH for the Smagorinsky model in comparison with reference solutions based on the standard Navier–Stokes problem solved on a USM 100 × 100 with 109203 DoFs. We observe that the Smagorinsky model solution accurately represents the feature of the turbulent flow, and we only see a slight displacement of the vortices of the order of the cell diameter of the reference mesh.

To quantitatively assess the convenience of exploiting IMH in this setting, we compute the L^2-errors of the velocity with respect to the reference configuration for $Re \in \{5000, 7500, 10000\}$. In Tables 3 and 4, we report the relative errors for the Smagorinsky model applied to the USM and IMH families, respectively. Henceforth, the hyphen symbol "-" in the tables indicates a failure of the Newton solver to converge. The IMH family presents smaller errors compared to the USM family and is more robust in terms of the Newton solver convergence: we reach convergence for all the Reynolds numbers and meshes considered, except for $Re = 10000$ with the coarse mesh. In contrast, the USM family encounters more convergence issues ($Re \in \{7500, 10000\}$ with the coarse mesh and $Re = 10000$ with the medium mesh). We will further comment on the convergence of the Newton solver later in this section.

In addition, in Fig. 6, we show the velocity profiles along the vertical and horizontal middle axes for laminar and turbulent (including transitional) flow, respectively. The numerical results are compared with those reported by Erturk et al. [44]. We prefer it compared to Ghia et al. [46] since it privileges data points close to walls, where sharp gradients are especially steep, and gives more relevant information about how well the refinement captures them.

We now propose a quantitative analysis based on the relative ℓ^2-error against the Erturk et al. [44] values for USM meshes, IMH meshes, and ARMs with $Re \in \{1000, 2500, 5000, 7500, 10000\}$. The results are collected in Tables 5, 6, and 7, respectively. Notice that the precision of the error is limited by the precision of the reported reference data in [44,46], i.e., five significant digits. It is clear that the IMH family offer the best performance in terms of accuracy, providing smaller or, in the worst case, comparable errors with respect to USM or ARM, when a comparison is possible, i.e., only when the Newton solver converges for the two considered meshes.

Specifically, we solve the nonlinear system using the Newton solver without preconditioning or relaxation techniques. The Newton solver is well known for exhibiting quadratic convergence, provided that a sufficiently good initialization is supplied. We noticed that, under the same initial guess condition for the solver, the IMH family are the most robust choice and allow for simulations for large Reynolds, up to $Re = 10000$. On the contrary, ARMs family yield convergence issues. In Fig. 7, we show the convergence process for the three coarse meshes for different representative Reynolds. From the plot, we observe that for $Re = 2000$, IMH and USM converge in the same number of iterations, while the ARM already diverges. When we increase the Reynolds number



(a) $Re = 5000$ and fine IMH (Smagorinsky)

(b) $Re = 5000$ and USM 100×100 (NSE)

(c) $Re = 7500$ and fine IMH (Smagorinsky)

(d) $Re = 7500$ and USM 100×100 (NSE)

(e) $Re = 10000$ and fine IMH (Smagorinsky)

(f) $Re = 10000$ and USM 100×100 (NSE)

Fig. 5. *Lid-driven cavity flow*. Streamlines for several Reynolds numbers, using the fine IMH mesh with Smagorinky model compared to the USM 100×100 with NSE.

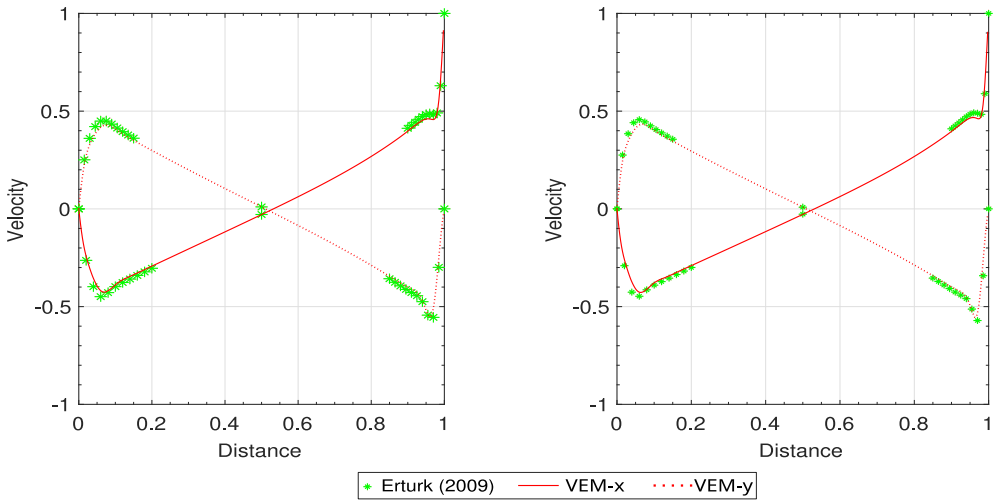


Fig. 6. *Lid-driven cavity flow*. Velocity profiles using the fine IMH for $Re = 7500$ (left) and $Re = 10000$ (right). The solid and dotted lines represent the x- and y-components of the velocity, respectively.

Table 5
Lid-driven cavity flow. Relative ℓ^2-errors for the velocity variable against Erturk et al. [44] varying the Re and considering USM.

Mesh	Re				
	1000	2500	5000	7500	10000
Coarse	9.365E−02	1.538E−01	2.399E−01	–	–
Medium	7.502E−02	1.195E−01	1.859E−01	2.444E−01	–
Fine	5.303E−02	7.594E−02	1.229E−01	1.638E−01	1.993E−01

Table 6
Lid-driven cavity flow. Relative ℓ^2-errors for the velocity variable against Erturk et al. [44] varying the Re and considering IMH.

Mesh	Re				
	1000	2500	5000	7500	10000
Coarse	7.307E−02	1.018E−01	1.445E−01	1.782E−01	1.104E−01
Medium	5.913E−02	7.813E−02	1.104E−01	1.385E−01	1.626E−01
Fine	4.173E−02	5.194E−02	7.070E−02	9.266E−02	1.133E−01

Table 7
Lid-driven cavity flow. Relative ℓ^2-errors for the velocity variable against Erturk et al. [44] varying the Re and considering ARM.

Mesh	Re				
	1000	2500	5000	7500	10000
Coarse	9.429E−02	–	–	–	–
Medium	6.972E−02	1.602E−02	–	–	–
Fine	1.054E−02	1.054E−02	–	–	–

to $Re = 6900$, the USM diverges. The IMH converges in few iterations not only for $Re = 6900$ but also for large Reynolds numbers, up to $Re = 10000$.

Remark 2 (*Anisotropic Scaling*)**.** Let $T \in \mathcal{T}_h$ be an element of the mesh. We recall that $h_F^\star := \min_{F \subset \partial T} h_F$ denotes the smallest face diameter of T. In the numerical investigation we carried out, choosing h_T in the Smagorinsky model for ARMs led to overdiffusive results due to the anisotropic nature of the mesh. For this reason, we tried $h_F^\star$ as an alternative scaling factor and we compared the two approaches for $Re \in \{100, 400, 1000, 2500, 3200\}$. For this test, we cover a range of Reynolds numbers only in the laminar regime for the mere purpose of comparison with the h_T scaling in terms of flow accuracy: indeed, the anisotropic scaling lacks convergence in the Newton solver for higher Reynolds numbers (see Table 7). We report in Tables 8 and 9 the relative ℓ^2-errors for low Reynolds numbers with anisotropic meshes using the scaling $h_F^\star$ and h_T, respectively. We compare with Ghia et al. [46] and

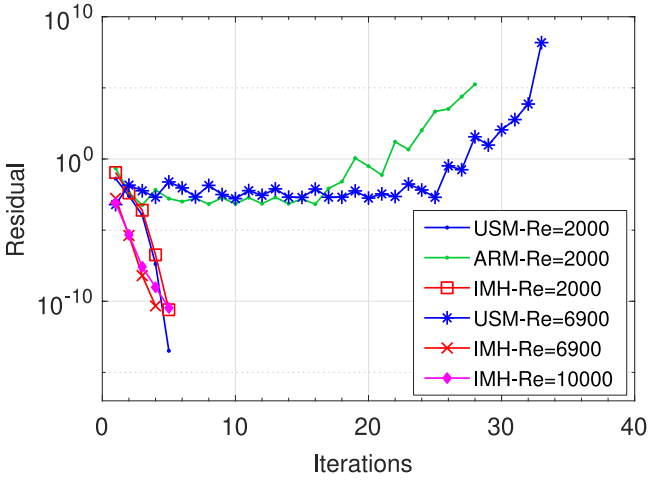


Fig. 7. *Lid-driven cavity flow*. The Newton solver convergence for various Re and the three mesh families.

Table 8
Lid-driven cavity flow. Relative ℓ^2-errors for the velocity variable with various Re and ARM with scaling $h_F^\star$. We compare with Ghia et al. [46] and Erturk et al. [44] depending on the available data.

Scale	Re					
	100 [46]	400 [46]	1000 [46]	1000 [44]	2500 [44]	3200 [46]
Coarse	1.016E−02	8.209E−02	1.659E−02	2.720E−02	–	–
Medium	1.078E−02	8.275E−02	1.324E−02	2.049E−02	–	–
Fine	1.150E−02	8.4074E−02	1.330E−02	1.267E−02	1.970E−02	3.562E−02

Table 9
Lid-driven cavity flow. Relative ℓ^2-errors for the velocity variable for various Re and ARM with scaling h_T. We compare with Ghia et al. [46] and Erturk et al. [44] depending on the available data.

Scale	Re					
	100[46]	400[46]	1000[46]	1000[44]	2500[44]	3200[46]
Coarse	1.449E−02	8.577E−02	1.178E−01	9.429E−02	–	–
Medium	1.363E−02	8.413E−02	8.823E−02	6.972E−02	1.602E−02	–
Fine	1.288E−02	5.055E−02	3.906E−02	1.054E−02	1.054E−02	1.118E−02

Erturk et al. [44] depending on the available data (we explicitly indicate them in the table). The use of h_T shows better behavior to reach the Newton convergence, caused by the overestimation of the Eddy viscosity. However, smaller errors are obtained using $h_F^\star$ scaling in comparison with the results using the h_T scaling, due the overdiffusion of the latter model. We do not exclude that a more tailored guess choice would help the Newton convergence for the $h_F^\star$ scaling; however, this investigation went beyond the goal of the contribution.

5. Conclusions

We proposed the combination of the VEM with the discrete Smagorinsky model to solve convection-dominated NSE. We presented successful numerical tests that show the method's convergence to analytical solutions as the meshsize tends to zero. Moreover, we performed lid-driven cavity tests with different mesh families to assess the potential of VEM approximation for the Smagorinsky model. The IMH mesh, which is easily handled by the VEM, proved to be the best choice in terms of accuracy, efficiency, and robustness of the nonlinear solver when compared to uniform and anisotropic meshes. We also apply a different anisotropic scaling for the Smagorinsky term, i.e., based on the minimum face diameter. This variant was shown to be more accurate and to mitigate overdiffusion. Nonetheless, the scarce convergence behavior of the Newton solver with anisotropic meshes did not improve.

We remark that, to the best of our knowledge, this is the first work dealing with VEM approximation in turbulence modeling. For these reasons, many are the unexplored yet interesting paths of this research. For example, we aim at applying more problem-guided mesh refinement, based on *ad hoc* a posteriori error estimator and/or based on the aspect ratio of the polygons.

CRediT authorship contribution statement

Stefano Berrone: Writing – review & editing, Supervision, Funding acquisition, Conceptualization. **Karol L. Cascavita:** Writing – review & editing, Writing – original draft, Validation, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Enrique Delgado Ávila:** Writing – review & editing, Supervision, Conceptualization. **Samuele Rubino:** Writing – review & editing, Supervision, Conceptualization. **Maria Strazzullo:** Writing – review & editing, Writing – original draft, Supervision, Methodology, Investigation, Formal analysis, Conceptualization. **Fabio Vicini:** Writing – review & editing, Supervision, Investigation, Conceptualization.

Acknowledgments

This manuscript reflects only the authors' views and opinions, and the Ministry cannot be considered responsible for them. The author S.B. kindly acknowledges the funding by the European Union through project Next Generation EU, M4C2, PRIN 2022 PNRR project P2022BH5CB_001 "Polyhedral Galerkin methods for engineering applications to improve disaster risk forecast and management: stabilization-free operator-preserving methods and optimal stabilization methods". The author K.L.C. acknowledges financial support by INdAM-GNCS through the projects 2024 (CUP: E53C23001670001) and 2025 (CUP: E53C24001950001), and the PNRR M4C2 project of CN00000013 National Centre for HPC, Big Data and Quantum Computing (HPC) (CUP: E13C22000990001). Authors E.D. and S.R. acknowledge the Spanish Government - FEDER Fund Project PID2021-123153OBC21 funded by MCIN/AEI/10.13039/501100011033/FEDER, UE. M.S. acknowledges the "20227K44ME - Full and Reduced order modelling of coupled systems: focus on non-matching methods and automatic learning (FaReX)" project – funded by European Union – Next Generation EU within the PRIN 2022 program (D.D. 104 - 02/02/2022 Ministero dell'Università e della Ricerca). Moreover, M.S. thanks the INdAM - GNCS Project "Metodi numerici efficienti per problemi accoppiati in sistemi complessi" (CUP_E53C24001950001) and the ECCOMAS EYIC Grant "CRAFT: Control and Reg-reduction in Applications for Flow Turbulence". The author F.V. acknowledges the financial support by INdAM-research group GNCS, project title: Metodi numerici avanzati per equazioni alle derivate parziali in fenomeni di trasporto e diffusione - CUP E53C24001950001.

References

[1] A.N. Kolmogorov, The local structure of turbulence in incompressible viscous fluid for very large Reynolds numbers, Comptes Rendus de L'Académie Des Sci. de L'URSS 30 (1941) 301–305.
[2] A.N. Kolmogorov, Dissipation of energy in the locally isotropic turbulence, Proc. R. Soc. Lond. Ser. A: Math. Phys. Sci. 434 (1890) (1991) 15–17.
[3] H.G. Roos, M. Stynes, L. Tobiska, Robust numerical methods for singularly perturbed differential equations: Convection-diffusion-reaction and flow problems., second ed., in: Springer Series in Computational Mathematics, vol. 24, Springer, 2008.
[4] A. Ivagnes, M. Strazzullo, M. Girfoglio, T. Iliescu, G. Rozza, Data-driven optimization for the evolve-filter-relax regularization of convection-dominated flows, Internat. J. Numer. Methods Engrg. 126 (9) (2025) e70042, http://dx.doi.org/10.1002/nme.70042.
[5] M. Strazzullo, F. Ballarin, T. Iliescu, T.C. Rebollo, Variational multiscale evolve and filter strategies for convection-dominated flows, Comput. Methods Appl. Mech. Engrg. 438 (2025) http://dx.doi.org/10.1016/j.cma.2025.117811.
[6] F. Zoccolan, M. Strazzullo, G. Rozza, Stabilized weighted reduced order methods for parametrized advection-dominated optimal control problems governed by partial differential equations with random inputs, J. Numer. Math. (2024) http://dx.doi.org/10.1515/jnma-2023-0006.
[7] F. Zoccolan, M. Strazzullo, G. Rozza, A Streamline Upwind Petrov-Galerkin Reduced Order Method for Advection-Dominated Partial Differential Equations Under Optimal Control, Comput. Methods Appl. Math. 25 (1) (2025) 237–260, http://dx.doi.org/10.1515/cmam-2023-0171.
[8] J. Smagorinsky, General circulation experiments with the primitive equations: I. The basic experiment, Mon. Weather Rev. 91 (3) (1963) 99–164, http://dx.doi.org/10.1175/1520-0493(1963)091<0099:GCEWTP>2.3.CO;2.
[9] T. Chacon Rebollo, R. Lewandowski, Mathematical and numerical foundations of turbulence models and applications, in: Modeling and Simulation in Science, Engineering and Technology, Springer New York, New York, NY, 2014, http://dx.doi.org/10.1007/978-1-4939-0050-6.
[10] P. Sagaut, Large Eddy Simulation for Incompressible Flows: An Introduction, first ed., Springer-Verlag Berlin Heidelberg, Berlin, Heidelberg, 2001, http://dx.doi.org/10.1007/978-3-662-04283-3.
[11] L.C. Berselli, T. Iliescu, W.J. Layton, Mathematics of Large Eddy Simulation of Turbulent Flows, Springer, 2006.
[12] M. Germano, U. Piomelli, P. Moin, W.H. Cabot, A dynamic subgrid-scale eddy viscosity model, Phys. Fluids A: Fluid Dyn. 3 (7) (1991) 1760–1765.
[13] T. Iliescu, V. John, W. Layton, G. Matthies, L. Tobiska, A numerical study of a class of LES models, Int. J. Comput. Fluid Dyn. 17 (1) (2003) 75–85.
[14] S. Khani, M.L. Waite, Large eddy simulations of stratified turbulence: the dynamic smagorinsky model, J. Fluid Mech. 773 (2015) 327–344, http://dx.doi.org/10.1017/jfm.2015.249.
[15] D. Lilly, A proposed modification of the germano subgrid-scale eddy viscosity model, Phys. Fluids A 4 (3) (1992) 633–635.
[16] C. Paré; s, Existence, uniqueness and regularity of solution of the equations of a turbulence model for incompressible fluids, Appl. Anal. 43 (3–4) (1992) 245–296.
[17] T.C. Rebollo, F. Hecht, M.G. Mármol, G. Orzetti, S. Rubino, Numerical approximation of the smagorinsky turbulence model applied to the primitive equations of the ocean, Math. Comput. Simulation 99 (2014) 54–70.
[18] W. Rozema, H.J. Bae, R.W.C.P. Verstappen, Local dynamic gradient smagorinsky model for large-eddy simulation, Phys. Rev. Fluids 7 (2022) 074604, http://dx.doi.org/10.1103/PhysRevFluids.7.074604.
[19] G.S. Winckelmans, A.A. Wray, O.V. Vasilyev, H. Jeanmart, Explicit-filtering large-eddy simulation using the tensor-diffusivity model supplemented by a dynamic smagorinsky term, Phys. Fluids 13 (5) (2001) 1385–1403, http://dx.doi.org/10.1063/1.1360192.
[20] L. Beirão da Veiga, F. Brezzi, A. Cangiani, G. Manzini, L.D. Marini, A. Russo, Basic principles of virtual element methods, Math. Models Methods Appl. Sci. 23 (01) (2013) 199–214, http://dx.doi.org/10.1142/S0218202512500492.
[21] L. Beirão da Veiga, F. Brezzi, L.D. Marini, A. Russo, The Hitchhiker's guide to the virtual element method, Math. Models Methods Appl. Sci. 24 (08) (2014) 1541–1573, http://dx.doi.org/10.1142/S021820251440003X.
[22] L. Beirão da Veiga, C. Lovadina, D. Mora, A virtual element method for elastic and inelastic problems on polytope meshes, Comput. Methods Appl. Mech. Engrg. 295 (2015) 327–346, http://dx.doi.org/10.1016/j.cma.2015.07.013.
[23] M.F. Benedetto, S. Berrone, A. Borio, S. Pieraccini, S. Scialò, A hybrid mortar virtual element method for discrete fracture network simulations, J. Comput. Phys. 306 (2016) 148–166, http://dx.doi.org/10.1016/j.jcp.2015.11.034.

[24] A. Borio, F.P. Hamon, N. Castelletto, J.A. White, R.R. Settgast, Hybrid mimetic finite-difference and virtual element formulation for coupled poromechanics, Comput. Methods Appl. Mech. Engrg. 383 (2021) 113917, http://dx.doi.org/10.1016/j.cma.2021.113917.
[25] A. Borio, F. Marcon, M. Strazzullo, General Order Virtual Element Methods for Neumann Boundary Optimal Control Problems in Saddle Point Formulation, 2024, arXiv:2411.08497.
[26] F. Dassi, C. Lovadina, M. Visinoni, Hybridization of the virtual element method for linear elasticity problems, Math. Models Methods Appl. Sci. 31 (14) (2021) 2979–3008, http://dx.doi.org/10.1142/S0218202521500676.
[27] F. Dassi, D. Mora, C. Reales, I. Velásquez, Mixed variational formulations of virtual elements for the polyharmonic operator $(-\Delta)^n$, Comput. Math. Appl. 158 (2024) 150–166, http://dx.doi.org/10.1016/j.camwa.2024.01.013.
[28] S. Kumar, D. Mora, R. Ruiz-Baier, N. Verma, Numerical solution of the Biot/elasticity interface problem using virtual element methods, J. Sci. Comput. 98 (3) (2024) http://dx.doi.org/10.1007/s10915-023-02444-7.
[29] B.-B. Xu, W.-L. Fan, P. Wriggers, High-order 3D virtual element method for linear and nonlinear elasticity, Comput. Methods Appl. Mech. Engrg. 431 (2024) http://dx.doi.org/10.1016/j.cma.2024.117258.
[30] L. Beirão Da Veiga, F. Brezzi, L.D. Marini, A. Russo, The virtual element method, Acta Numer. 32 (2023) 123–202, http://dx.doi.org/10.1017/S0962492922000095.
[31] L. Beirão da Veiga, C. Lovadina, G. Vacca, Divergence free virtual elements for the Stokes problem on polygonal meshes, ESAIM Math. Model. Numer. Anal. 51 (2) (2017) 509–535, http://dx.doi.org/10.1051/m2an/2016032, URL: https://www.numdam.org/articles/10.1051/m2an/2016032/.
[32] G. Manzini, A. Mazzia, Conforming virtual element approximations of the two-dimensional Stokes problem, Appl. Numer. Math. 181 (2022) 176–203, http://dx.doi.org/10.1016/j.apnum.2022.06.002, URL: https://www.sciencedirect.com/science/article/pii/S0168927422001520.
[33] L. Beirão da Veiga, C. Lovadina, G. Vacca, Virtual elements for the Navier–Stokes problem on polygonal meshes, SIAM J. Numer. Anal. 56 (3) (2018) 1210–1242, http://dx.doi.org/10.1137/17M1132811, arXiv:https://doi.org/10.1137/17M1132811.
[34] A. Quarteroni, A. Valli, Numerical Approximation of Partial Differential Equations, vol. 23, Springer Science & Business Media, 2008.
[35] T.C. Rebollo, E.D. Ávila, M.G. Mármol, F. Ballarin, G. Rozza, On a Certified Smagorinsky Reduced Basis Turbulence Model, SIAM J. Numer. Anal. 55 (6) (2017) 3047–3067, http://dx.doi.org/10.1137/17M1118233.
[36] S. Rubino, Numerical Modeling of Turbulence by Richardson Numberbased and VMS Models, (Ph.D. Thesis), 2014.
[37] D. Lilly, The representation of small-scale turbulence in numerical simulation experiments, NCAR Rep. (1966).
[38] S.C. Brenner, L.-Y. Sung, Virtual element methods on meshes with small edges or faces, Math. Models Methods Appl. Sci. 28 (07) (2018) 1291–1336, http://dx.doi.org/10.1142/S0218202518500355.
[39] T. Sorgente, S. Biasotti, G. Manzini, M. Spagnuolo, The role of mesh quality and mesh quality indicators in the virtual element method, Adv. Comput. Math. 48 (1) (2021) 3, http://dx.doi.org/10.1007/s10444-021-09913-3.
[40] F. Dassi, G. Vacca, Bricks for the mixed high-order virtual element method: Projectors and differential operators, Appl. Numer. Math. 155 (2020) 140–159, http://dx.doi.org/10.1016/j.apnum.2019.03.014, URL: https://www.sciencedirect.com/science/article/pii/S0168927419300674, Structural Dynamical Systems: Computational Aspects held in Monopoli (Italy) on June 12-15, 2018..
[41] M. Cicuttin, An implementation detail about the scaling of monomial bases in polytopal finite element methods, Appl. Math. Lett. 159 (2025) 109281, http://dx.doi.org/10.1016/j.aml.2024.109281, URL: https://www.sciencedirect.com/science/article/pii/S089396592400301X.
[42] S. Berrone, A. Borio, G. Teora, F. Vicini, POLYDIM: A C++ library for POLYtopal DIscretization Methods, Comput. Phys. Comm. 320 (2026) 109937, http://dx.doi.org/10.1016/j.cpc.2025.109937.
[43] A. Logg, K. Mardal, G. Wells, Automated Solution of Differential Equations by the Finite Element Method, Springer-Verlag, Berlin, 2012.
[44] E. Erturk, Discussions on driven cavity flow, Internat. J. Numer. Methods Fluids 60 (3) (2009) 275–294, http://dx.doi.org/10.1002/fld.1887, arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/fld.1887, URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/fld.1887.
[45] R. Haferssas, P. Jolivet, S. Rubino, Efficient and scalable discretization of the Navier–Stokes equations with LPS modeling, Comput. Methods Appl. Mech. Engrg. 333 (2018) 371–394, http://dx.doi.org/10.1016/j.cma.2018.01.026.
[46] U. Ghia, K. Ghia, C. Shin, High-Re solutions for incompressible flow using the Navier-Stokes equations and a multigrid method, J. Comput. Phys. 48 (3) (1982) 387–411, http://dx.doi.org/10.1016/0021-9991(82)90058-4, URL: https://www.sciencedirect.com/science/article/pii/0021999182900584.