

# A new family of semi-implicit Finite Volume/Virtual Element methods for incompressible flows on unstructured meshes

Walter Boscheri<sup>a,\*</sup>, Andrea Chiozzi<sup>b,\*</sup>, Michele Giuliano Carlino<sup>a</sup>, Giulia Bertaglia<sup>b</sup>

<sup>a</sup> Department of Mathematics and Computer Science, University of Ferrara, Via Niccolò Machiavelli 30, 44121 Ferrara, Italy

<sup>b</sup> Department of Environmental and Prevention Sciences, University of Ferrara, C.so Ercole I d'Este 32, 44121 Ferrara, Italy

Received 14 April 2023; received in revised form 16 May 2023; accepted 22 May 2023

Available online 8 June 2023

## Abstract

We introduce a new family of high order accurate semi-implicit schemes for the solution of nonlinear time-dependent systems of partial differential equations (PDE) on unstructured polygonal meshes. The time discretization is based on a splitting between explicit and implicit terms that may arise either from the multi-scale nature of the governing equations, which involve both slow and fast scales, or in the context of projection methods, where the numerical solution is projected onto the physically meaningful solution manifold. We propose to use a high order finite volume (FV) scheme for the explicit terms, hence ensuring conservation property and robustness across shock waves, while the virtual element method (VEM) is employed to deal with the discretization of the implicit terms, which typically requires an elliptic problem to be solved. The numerical solution is then transferred via suitable  $L_2$  projection operators from the FV to the VEM solution space and vice-versa. High order time accuracy is then achieved using the semi-implicit IMEX Runge–Kutta schemes, and the novel schemes are proven to be asymptotic preserving (AP) and well-balanced (WB). As representative models, we choose the shallow water equations (SWE), thus handling multiple time scales characterized by a different Froude number, and the incompressible Navier–Stokes equations (INS), which are solved at the aid of a projection method to satisfy the solenoidal constraint of the velocity field. Furthermore, an implicit discretization of the viscous terms is also devised for the INS model, which is based on the VEM technique. Consequently, the CFL-type stability condition on the maximum admissible time step is based only on the fluid velocity and not on the celerity nor on the viscous eigenvalues. A large suite of test cases demonstrates the accuracy and the capabilities of the new family of schemes to solve relevant benchmarks in the field of incompressible fluids.

© 2023 Elsevier B.V. All rights reserved.

**Keywords:** Semi-implicit schemes; Finite volume; Virtual Element Method; High order in space and time; Asymptotic preserving; Incompressible flows

## 1. Introduction

Incompressible flows are mathematically described by non-linear systems of hyperbolic conservation laws, and they cover a wide range of physical phenomena such as environmental, geophysical and meteorological flows, as well as the dynamics of mechanical processes like turbo-machinery. In this article, we consider two different models:

\* Corresponding authors.

E-mail addresses: [walter.boscheri@unife.it](mailto:walter.boscheri@unife.it) (W. Boscheri), [andrea.chiozzi@unife.it](mailto:andrea.chiozzi@unife.it) (A. Chiozzi), [michelegiuliano.carlino@unife.it](mailto:michelegiuliano.carlino@unife.it) (M.G. Carlino), [giulia.bertaglia@unife.it](mailto:giulia.bertaglia@unife.it) (G. Bertaglia).

(i) the shallow water equations (SWE), which are commonly used to simulate storm surges, inundations, dam breaks and river floods; (ii) the incompressible Navier–Stokes (INS) equations, which model atmospheric flows, flows around wings and even pressurized pipe flows. Due to the relevant physical applications, there is a great interest in the numerical solution of these models, which however gives rise to some complexities. In particular, the shallow water equations are concerned with slow and fast time scales, which are linked to the non-linear convective and pressure terms, respectively. Moreover, the energy equation in the incompressible Navier–Stokes model reduces to a constraint on the velocity field that must satisfy a divergence-free condition. The use of unstructured grids is very convenient for real-world applications that deal, for example, with ocean bathymetry, wing and turbine shapes, and river morphology, which should be approximated with high accuracy. Moreover, obtaining high order of accuracy also in time represents an important goal in order to achieve accurate results for unsteady problems.

From the numerical viewpoint, the multi-scale nature of the system poses severe restrictions on the maximum admissible time step in order to make the numerical scheme capable of capturing the fast waves. For explicit time discretization, this stability condition can lead to extremely small time steps that ultimately prevent the method to be effectively adopted for real applications. Furthermore, the amount of numerical dissipation introduced by the scheme dramatically augments, spoiling the accuracy of the solution. An effective strategy is based on flux splitting techniques, which separate the slow and fast scales. An explicit time discretization is employed for the slow scales that typically correspond to the non-linear convective terms and are, therefore, related to the fluid velocity, while an implicit time stepping is retained for the fast scales which involve the acoustic waves like the sound speed or the celerity. On the other hand, the divergence-free constraint is rather complicated to be respected on general unstructured meshes with high order of accuracy, therefore pressure-correction methods are typically used, which first compute an intermediate velocity field that is not solenoidal, and then a correction is performed in order to project the velocity onto the divergence-free manifold. Furthermore, the viscous terms in the INS model are endowed with a parabolic time step restriction, which is also responsible of very small time steps in viscous dominated flows, which might be conveniently treated implicitly, although this is not trivial on unstructured meshes.

By all means, this class of numerical methods belongs to the category of implicit–explicit (IMEX) [1–4] or semi-implicit time schemes [5–7], which imply the solution of an algebraic system for the unknown physical quantity that is discretized implicitly (for example the pressure or the velocity field). This system does normally involve an elliptic equation to be solved on the entire computational domain. The effectiveness of IMEX schemes have been demonstrated in several contexts, including low Mach compressible flows [8–10], magnetized plasma flows [11], pipe flows [12,13], as well as free surface [14,15] and atmospheric [16,17] applications. If Cartesian meshes are adopted for the spatial discretization [9,18], finite difference schemes are likely to be employed for the implicit terms, and the high order extension is straightforward by enlarging the stencil of the finite difference operators. Besides, finite differences can also be used on orthogonal unstructured meshes like Voronoi tessellations [19–21] up to second order of accuracy. In this case, the solution of the implicit algebraic system can be carried out quite easily. However, when general unstructured meshes pave the computational domain, the solution of the algebraic system becomes more difficult, since the spatial discretization might be rather complicated. If finite volume methods are used, the inversion of the reconstruction operator should be embedded in the system solver, leading to additional difficulties. Discontinuous Galerkin methods [22–24] represent a clever option, because they simultaneously provide high accuracy and compactness of the stencil. Nevertheless, finite volume methods can deal with very general control volumes and they exhibit, by construction, conservation and shock capturing properties, which makes such schemes suitable for the discretization of the explicit slow scale terms, like the non-linear convective terms. In addition, since the implicit system is typically concerned with an elliptic equation on the fast scale quantity, like the pressure, finite element schemes could also be used on both quadrilateral and triangular cells in two space dimensions (2D) or cubic and tetrahedral control volumes in three-dimension (3D). This is the reasoning behind the recent emergence of a new class of hybrid finite volume/finite element methods [25–28], which combine the robustness of the explicit finite volume with the versatility of the finite element method. Due to the classical finite element solver, these numerical methods are restricted to simplex meshes in 2D/3D – i.e. meshes that are topologically dual of triangulations. A first attempt of using a hybrid scheme on general polygonal meshes has been forwarded in [29], where the implicit pressure system is solved at the aid of a discontinuous Galerkin approach that acts on a sub-triangulation of the polygonal tessellation.

The goal of this paper is to design a genuinely hybrid finite volume/finite element scheme on arbitrary shaped grids in 2D. The originality of the proposed scheme lies in the usage of the Virtual Element Method (VEM) to

handle the solution of the implicit elliptic equation. The Virtual Element Method (VEM), originally developed in [30–33], is a stabilized Galerkin formulation, originating from mimetic finite difference schemes [34], to solve partial differential equations on very general polygonal and polyhedral meshes that overcomes the many difficulties and challenges that are associated with polygonal finite element formulations. The VEM represents a generalization of the finite element method in which the explicit knowledge of the basis functions is not needed. Instead, the VEM provides projection operators of the basis functions onto polynomial spaces of arbitrary degree that allow to discretize and to suitably approximate the bilinear form and the continuous linear functional deriving from the variational formulation. The discretized bilinear form is conveniently decomposed as the sum of a consistent term, which ensures polynomial consistency, and a correction term, which guarantees stability. The VEM has been extensively proposed for the solution of elliptic problems in many fields of mechanics, such as two- and three-dimensional linear elasticity [35–38], nearly incompressible elasticity [39], linear elastodynamics [40], elastic problems with singularities and discontinuities [41,42], fracture mechanics [43,44], flow problems in porous media [45,46]. Recently, there has been an interest in addressing also steady state partial differential equations (PDE), as for instance the steady Navier–Stokes equations [47–52]. Noticeably, in this application, the divergence-free constraint is directly imposed in the definition of the virtual basis, avoiding any projection (or correction) operator. Nevertheless, unsteady problems have still to be faced recurring to the VEM.

In this paper, we introduce an innovative hybrid finite volume/virtual element solver for hyperbolic PDE on general unstructured polygonal meshes in the context of incompressible flows, solving both the SWE and the INS model. The FV solver is used to treat the explicit terms, while the VEM strategy enters the scheme to solve the implicit algebraic system for the pressure. Furthermore, in the incompressible Navier–Stokes equations, the VEM approach is actually used twice: firstly to implicitly discretize the viscous terms [53,54], and secondly to perform the projection step ensuring a divergence-free velocity field. Remarkably, the entire scheme works on genuinely polygonal tessellations. Finally, high order in time is ensured by applying a semi-implicit IMEX time stepping technique, which has been also employed in the context of compressible fluids [10].

The rest of the paper is organized as follows. In Section 2 the governing equations are presented, studying the multi-scale character of the mathematical models which leads to the flux splitting for SWE or the projection strategy for INS. Section 3 is devoted to the presentation of the numerical scheme, with details on the explicit FV solver and the implicit VEM. The asymptotic and the well-balance property of the scheme are also studied in this section. A large suite of test problems is proposed in Section 4, showing the accuracy and the robustness of the new family of schemes while solving academic benchmarks for SWE and INS systems. Finally, we provide some concluding remarks and an outlook on possible future works in Section 5.

## 2. Mathematical model

Incompressible flows are mathematically described by by nonlinear time-dependent systems of PDE, which can be cast into the following formulation:

$$\frac{\partial \mathbf{Q}}{\partial t} + \nabla \cdot \mathbb{F}(\mathbf{Q}, \nabla \mathbf{Q}) + \mathbb{B} \cdot \nabla \mathbf{Q} = \mathbf{0}, \quad \mathbf{x} \in \Omega \subset \mathbb{R}^d, \quad t \in \mathbb{R}_0^+, \quad \mathbf{Q} \in \Omega_{\mathbf{Q}} \subset \mathbb{R}^\gamma, \quad (1)$$

with  $d = 2$  representing the number of space dimensions. The vector  $\mathbf{x} = (x, y)$  identifies the spatial coordinates in the physical domain  $\Omega$  and  $t$  is the time. The vector of evolutionary variables  $\mathbf{Q} = (q_1, q_2, \dots, q_\gamma)$  is defined in the space of the admissible states  $\Omega_{\mathbf{Q}} \subset \mathbb{R}^\gamma$ , and  $\mathbb{F}(\mathbf{Q}, \nabla \mathbf{Q}) = (\mathbf{f}(\mathbf{Q}, \nabla \mathbf{Q}), \mathbf{g}(\mathbf{Q}, \nabla \mathbf{Q}))$  denotes the conservative non-linear flux tensor ( $\mathbf{f}$  in  $x$ -direction,  $\mathbf{g}$  in  $y$ -direction), while  $\mathbb{B} = (\mathbf{B}_x, \mathbf{B}_y)$  contains the purely non-conservative part of the system written in block-matrix notation.

A widespread model is given by the Incompressible Navier–Stokes (INS) equations, which can be cast into form (1) with absence of non-conservative terms by defining

$$\mathbf{Q} = (0, \mathbf{v}), \quad \mathbb{F}(\mathbf{Q}, \nabla \mathbf{Q}) = (\mathbf{v}, \mathbf{v} \otimes \mathbf{v} - \nu \nabla \mathbf{v} + p \mathbf{I}), \quad \mathbb{B} \cdot \nabla \mathbf{Q} = \mathbf{0}, \quad (2)$$

introducing the identity matrix  $\mathbf{I}$ . The velocity field is  $\mathbf{v}(\mathbf{x}, t) = (u, v)$ , with the components along  $x$ - and  $y$ -direction, and  $p(\mathbf{x}, t) = P(\mathbf{x}, t)/\rho$  denotes the normalized fluid pressure, where  $P(\mathbf{x}, t)$  is the physical pressure and  $\rho$  is the constant fluid density. The kinematic viscosity coefficient is given by  $\nu = \mu/\rho$  with  $\mu$  being the dynamic viscosity of the fluid. The first equation of the INS model (2) is the divergence-free constraint on the velocity field, which corresponds to the energy equation in the low Mach limit of compressible flows [9].

For situations in which the vertical velocity field is negligible with respect to the horizontal one, incompressible flows are governed by the Shallow Water Equations (SWE). By neglecting friction forces, the SWE fits the general formalism (1) by setting

$$\mathbf{Q} = (\eta, \mathbf{q}, b), \quad \mathbb{F}(\mathbf{Q}) = (\mathbf{q}, \mathbf{v} \otimes \mathbf{q}, 0), \quad \mathbb{B} \cdot \nabla \mathbf{Q} = (0, gH\nabla\eta, 0). \quad (3)$$

Here,  $\eta(\mathbf{x}, t) \geq 0$  is the free surface elevation of the fluid and  $b(\mathbf{x})$  defines the bottom bathymetry, measured with respect to a reference plane, thus the total water depth is given by  $H(\mathbf{x}, t) = \eta(\mathbf{x}, t) - b(\mathbf{x}) \geq 0$ . The flow discharge is  $\mathbf{q}(\mathbf{x}, t) = H(\mathbf{x}, t)\mathbf{v}(\mathbf{x}, t)$ , while  $g$  is the gravity acceleration. The SWE model (3) is written in terms of conservative variables and it can reduce to a conservation law ( $\mathbb{B} \cdot \nabla \mathbf{Q} = \mathbf{0}$ ) in the case of flat bottom topography, hence  $\eta = H$  and  $\mathbb{F}(\mathbf{Q}) = (\mathbf{q}, \mathbf{v} \otimes \mathbf{q} + gH^2/2, 0)$ .

These mathematical models can describe phenomena characterized by different time scales (i.e., multi-scale regimes), leading to the resolution of problems that are notoriously complicated to approximate numerically due to the associated stiffness. For example, low Froude number  $Fr = |\mathbf{v}|/\sqrt{gH}$  flows in the SWE are characterized by a rather low convective speed  $|\mathbf{v}|$  compared to the acoustic-gravity celerity  $\sqrt{gH}$ . For the INS equations, the viscous dominated flows induce a stiffness, which is then responsible of a severe CFL-type stability restriction on the time step for explicit time discretizations. To identify the terms responsible for the stiffness, it is convenient to write the governing equations in dimensionless form [9,29]. Let the variables of (2) and (3) be rescaled as

$$\tilde{\mathbf{x}} = \frac{\mathbf{x}}{L_0}, \quad \tilde{t} = \frac{t}{T_0}, \quad \tilde{\mathbf{v}} = \frac{\mathbf{v}}{U_0}, \quad \tilde{\eta} = \frac{\eta}{H_0}, \quad \tilde{H} = \frac{H}{H_0}, \quad \tilde{b} = \frac{b}{H_0}, \quad \tilde{\mathbf{q}} = \frac{\mathbf{q}}{U_0H_0}, \quad \tilde{p} = \frac{pL_0}{\nu U_0}, \quad (4)$$

where  $L_0, T_0, U_0 = L_0/T_0, H_0$  are the characteristic length, time, velocity and depth, respectively, of the problem under consideration. By omitting the tilde symbol for ease of reading, the dimensionless form of the INS equations (2) writes

$$\mathbf{Q} = (\mathbf{0}, \mathbf{v}), \quad \mathbb{F}(\mathbf{Q}, \nabla \mathbf{Q}) = \left( \mathbf{v}, \mathbf{v} \otimes \mathbf{v} - \frac{1}{Re} (\nabla \mathbf{v} - p\mathbf{I}) \right), \quad (5)$$

where the Reynolds number  $Re = U_0L_0/\nu$  appears in front of the viscosity and pressure terms of the momentum equation.

Analogously, we can derive the dimensionless form of the SWE system (3) that is given by

$$\mathbf{Q} = (\eta, \mathbf{q}, b), \quad \mathbb{F}(\mathbf{Q}) = (\mathbf{q}, \mathbf{v} \otimes \mathbf{q}, \mathbf{0}), \quad \mathbb{B} \cdot \nabla \mathbf{Q} = \left( 0, \frac{H}{Fr^2} \nabla \eta, 0 \right), \quad (6)$$

where it can be observed the presence of the Froude number  $Fr = U_0/\sqrt{gH_0}$  in the pressure gradient term.

In both cases, the dimensionless systems can be characterized by a stiffness parameter  $\varepsilon$ , which is the reference Reynolds number  $\varepsilon = Re$  in (5) or the reference Froude number  $\varepsilon = Fr$  in (6). The asymptotic limit of the models can consequently be studied using the asymptotic parameter  $\varepsilon$ , as it will be carried out in Section 3.8.

### 2.1. Flux splitting

In order to develop an efficient and accurate numerical method to solve multi-scale problems as those discussed in the previous section, it is convenient to treat the governing equations with a flux splitting approach, which is based on the separation of the different time scales, fast and slow, in terms of the flux terms (either conservative or non-conservative). In the literature, this strategy has already been followed to treat incompressible flow models [5,29,54–56] but also compressible fluids [9,57,58]. The idea behind this technique consists in dividing the system of governing equations into two sub-systems, where one contains the fluxes dependent on the scaling parameter  $\varepsilon$  and the other one does not. This procedure basically allows to highlight the terms that will be discretized explicitly in time with respect to those that, due to the  $\varepsilon$ -dependence, will encounter an implicit discretization.

The flux splitting applied to a generic system of the type (1) reads

$$\frac{\partial \mathbf{Q}}{\partial t} + \nabla \cdot \mathbb{F}_E(\mathbf{Q}) + \nabla \cdot \mathbb{F}_I(\mathbf{Q}, \nabla \mathbf{Q}) + \mathbb{B}_E \cdot \nabla \mathbf{Q} + \mathbb{B}_I \cdot \nabla \mathbf{Q} = \mathbf{0}, \quad (7)$$

being  $\mathbb{F} = \mathbb{F}_E + \mathbb{F}_I, \mathbb{B} = \mathbb{B}_E + \mathbb{B}_I$ . The subscripts “E” and “I” distinguish the explicit terms related to convective-type phenomena, from the implicit contribution concerned with the fluid pressure and viscosity. As a consequence, we obtain a partitioned system [59] consisting in the following sub-systems.

- Convective sub-system:

$$\frac{\partial \mathbf{Q}}{\partial t} + \nabla \cdot \mathbb{F}_E(\mathbf{Q}) + \mathbb{B}_E \cdot \nabla \mathbf{Q} = \mathbf{0}; \tag{8}$$

- Pressure–viscosity sub-system:

$$\frac{\partial \mathbf{Q}}{\partial t} + \nabla \cdot \mathbb{F}_I(\mathbf{Q}, \nabla \mathbf{Q}) + \mathbb{B}_I \cdot \nabla \mathbf{Q} = \mathbf{0}. \tag{9}$$

In the case of the INS equations (5), the scaling parameter is represented by the Reynolds number,  $\varepsilon = \text{Re}$ , implying that all terms related to the Reynolds number should be treated implicitly. Therefore, we may opt for the following splitting:

$$\mathbb{F}_E(\mathbf{Q}) = (\mathbf{0}, \mathbf{v} \otimes \mathbf{v}), \quad \mathbb{F}_I(\mathbf{Q}, \nabla \mathbf{Q}) = \left( \mathbf{v}, -\frac{1}{\varepsilon} (\nu \nabla \mathbf{v} - p \mathbf{I}) \right), \tag{10}$$

which simultaneously allows the time step stability condition to be independent from the parabolic viscous terms and also to respect the divergence-free constraint on the velocity field. The eigenvalues of the explicit sub-system in normal direction  $\mathbf{n} = (n_x, n_y)$  are simply given by

$$\lambda_1 = 0, \quad \lambda_{2,3,4} = 2\mathbf{v} \cdot \mathbf{n}, \tag{11}$$

thus the time step is limited by the fluid velocity and not by the viscosity coefficient with parabolic cell size.

For the specific case of the SWE (6), we adopt the classical flux splitting originally proposed in [5], hence obtaining

$$\mathbb{F}_E(\mathbf{Q}) = (\mathbf{0}, \mathbf{v} \otimes \mathbf{q}, \mathbf{0}), \quad \mathbb{F}_I(\mathbf{Q}) = (\mathbf{q}, \mathbf{0}, \mathbf{0}), \quad \mathbb{B}_I \cdot \nabla \mathbf{Q} = \left( 0, \frac{H}{\varepsilon^2} \nabla \eta, 0 \right). \tag{12}$$

Even in this case, the eigenvalues of the explicit sub-system only involve the convective fluid velocity:

$$\lambda_{1,2} = 0, \quad \lambda_3 = \mathbf{v} \cdot \mathbf{n}, \quad \lambda_4 = 2\mathbf{v} \cdot \mathbf{n}. \tag{13}$$

Indeed, we recall that the scaling parameter coincides with the Froude number, hence  $\varepsilon = \text{Fr}$ , and all the related terms are taken implicitly.

For both the models, on a computational mesh with characteristic mesh size  $h$ , the time step is limited by a classical CFL stability condition that is only based on the maximum convective eigenvalue, reading

$$\Delta t \leq \text{CFL} \min_{\Omega} \frac{h}{\max |\lambda|}. \tag{14}$$

The above condition yields a milder stability restriction, especially in the asymptotic regime when  $\varepsilon \rightarrow 0$ . In particular, the time step becomes independent of the stiffness parameter, which means that simulations of flows defined by any Reynolds and Froude number can be run with the same computational efficiency. Moreover, since the terms related to the stiffness are treated implicitly, the numerical dissipation is solely proportional to the fluid speed, yielding a numerical scheme that is particularly suited for applications in the asymptotic (zero-relaxation) limit, i.e., when  $\varepsilon \rightarrow 0$ . We remark here that this type of problems could not be simulated adopting a purely explicit scheme in time [60–62].

### 3. Numerical scheme

We start by presenting the discretization of the computational domain in space and time. Next, we define the polynomial spaces adopted for the discretization of the numerical solution, and we present the first order semi-discrete scheme in time, analysing the asymptotic and well-balance properties. Finally, the high order extension in space and time of the new schemes are introduced.

#### 3.1. Discretization of the time computational domain

The time coordinate is defined in the time interval  $t \in [0, t_f]$ , where  $t_f \in \mathbb{R}_0^+$  represents the final time. A sequence of discrete points  $t^n$  approximate the temporal computational domain such that  $t \in [t^n; t^{n+1}]$ , hence

$$t^{n+1} = t^n + \Delta t, \tag{15}$$

where the time step  $\Delta t = t^{n+1} - t^n$  is determined at each time iteration according to the stability condition (14).

### 3.2. Discretization of the space computational domain

The space computational domain  $\Omega \subseteq \mathbb{R}^2$  is discretized by a set of non-overlapping polygonal control volumes  $P_i$  with boundary  $\partial P_i$ , surface area  $|P_i|$  and characteristic mesh size  $h_i = \sqrt{|P_i|}$ . The total number of cells is  $N_P$ , thus  $i = 1, \dots, N_P$ , and the union of all elements is called the tessellation  $\mathcal{T}_\Omega$  of the domain

$$\mathcal{T}_\Omega = \bigcup_{i=1}^{N_P} P_i. \tag{16}$$

To make the notation easier, we drop the subscript  $i$ , bearing in mind that all the geometric quantities are specific to each polygon  $P_i$ . The barycentre  $\mathbf{x}_P := (x_P, y_P)$  of each cell is computed as

$$\mathbf{x}_P = \frac{1}{|P|} \int_P \mathbf{x} d\mathbf{x}. \tag{17}$$

The number of vertices of element  $P$  is denoted by  $N_E$ . The boundary of  $P$  is formed by straight edges, and the polygon  $P$  has  $N_E$  edges. The vertices of the polygonal element  $P$  are oriented in counter-clockwise order and their coordinates are denoted by  $\mathbf{x}_k := (x_k, y_k)$ ,  $k = 1, 2, \dots, N_E$ . For convenience, we also use  $x_k$  as a label for the  $k$ th vertex. We denote the unit normal vector to edge  $e \in \partial P$  by  $\mathbf{n}_{P,e}$ . Each vector  $\mathbf{n}_{P,e}$  points out of  $P$ . We assume that the orientation of all edges is fixed once and for all, so that we can unambiguously introduce  $\mathbf{n}_e$ , the unit normal vector to edge  $e$ , which is independent of the elements  $P$  that share the common edge  $e$ . Moreover, the following assumptions on the regularity of the mesh are adopted.

All the integrals appearing in the numerical scheme are evaluated relying on quadrature formulae of suitable accuracy. For boundary integrals we use Gaussian quadrature rules [63], while for volume integrals we adopt the efficient numerical integration proposed in [64,65].

*Mesh regularity assumptions.* There exists a positive constant  $\varrho$  such that for every polygonal element  $P \in \mathcal{T}_h$  it holds that

- (i)  $P$  is star-shaped with respect to a disk with radius greater than  $\varrho \max_{\mathcal{T}_h} h_P$ ;
- (ii) for every edge  $e \in \partial P$  it holds that  $|e| \geq \varrho h_P$ , with  $|e|$  being the edge length.

The restriction of  $P$  being star-shaped in (i) implies that all the elements are simply connected subsets of  $\mathbb{R}^2$  and that they have a finite number of vertices and edges. The scaling assumption in (ii) implies that the number of edges on the boundary of any element is uniformly bounded over the whole mesh  $\mathcal{T}_h$ , thus avoiding collapsing vertexes that would give rise to degenerate edges.

### 3.3. Conforming virtual element space

On every polygonal element  $P$  with boundary  $\partial P$ , we define the standard local virtual element space of order  $k$ ,  $V_k^h(P)$ , as

$$V_k^h(P) = \left\{ v_h \in H^1(P) : \Delta v_h \in \mathbb{P}_{k-2}(P), v_h|_{\partial P} \in C^0(\partial P), v_h|_e \in \mathbb{P}_k(e) \forall e \in \partial P \right\}, \tag{18}$$

where  $\mathbb{P}_k(P)$  denotes the space of polynomials of degree less than or equal to  $k$  on  $P$ . A straightforward consequence of definition (18) is that  $\mathbb{P}_k(P)$  is a subspace of  $V^h(P)$ . We define

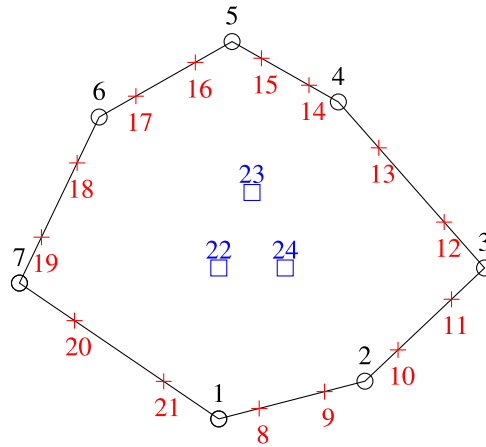
$$n_k := \dim V_k^h(P) = \frac{(k+1)(k+2)}{2}. \tag{19}$$

Let us introduce the multi-index  $\boldsymbol{\kappa} = (\kappa_1, \kappa_2)$ , with the usual notation  $|\boldsymbol{\kappa}| = \kappa_1 + \kappa_2$ . Moreover, if  $\mathbf{x} = (x_1, x_2)$ , then  $\mathbf{x}^{\boldsymbol{\kappa}} = (x_1^{\kappa_1}, x_2^{\kappa_2})$ . We define the scaled monomials  $m_{\boldsymbol{\kappa}}$  of degree equal to  $|\boldsymbol{\kappa}|$  as

$$m_{\boldsymbol{\kappa}} = \left( \frac{\mathbf{x} - \mathbf{x}_P}{h_P} \right)^{\boldsymbol{\kappa}}. \tag{20}$$

Hence, we define the set  $\mathcal{M}_k(P)$  of scaled monomials of degree less than or equal to  $k$ ,

$$\mathcal{M}_k(P) := \{m_{\boldsymbol{\kappa}} : 0 \leq |\boldsymbol{\kappa}| \leq k\}, \tag{21}$$



**Fig. 1.** Degrees of freedom for  $k = 3$ . The internal ones are indicated by a square symbol in blue. The boundary degrees of freedom are located on the vertices (o in black) and along the edges (+ in red).

and recognize that  $\mathcal{M}_k(P)$  is a basis for  $\mathbb{P}_k(P)$ . We also use the  $\alpha$  subscript to denote the  $\alpha$ -th scaled monomial  $m_\alpha$  of  $\mathcal{M}_k(P)$ .

As shown in [30], each function  $v_h$  in  $V_k^h(P)$  is uniquely determined by the following degrees of freedom:

- the  $N_E$  values of  $v_h$  at the vertices of  $P$ ;
- the  $N_E(k - 1)$  values of  $v_h$  on the  $(k + 1)$  Gauss–Lobatto quadrature points on each edge  $e$ ;
- the moments up to degree  $k - 2$  of  $v_h$  in  $P$ , defined as

$$\frac{1}{|P|} \int_P v_h m_\alpha \, d\mathbf{x}, \quad \alpha = 1, \dots, n_{k-2}, \tag{22}$$

where  $n_{k-2}$  is computed from (19) as the dimension of  $\mathbb{P}_{k-2}(P)$ .

A sketch of the internal and boundary degrees of freedom for  $k = 3$  is shown in Fig. 1. The internal  $n_{k-2}$  degrees of freedom have actually no pointwise location, since they are given by moments according to (22), thus they are simply plotted at the interior of the cell to be interpreted as an integral definition. It follows that the dimension of  $V_k^h(P)$  is

$$\dim V_k^h(P) = N_E k + \frac{(k - 1)k}{2}, \tag{23}$$

thus we denote by  $N_P^{\text{dof}} = \dim V_k^h(P)$  the number of degrees of freedoms of each  $v_h \in V_k^h(P)$ . We also denote by  $\text{dof}_i(v_h)$  the value of the  $i$ th degree of freedom of  $v_h$ . Denoting  $\{\varphi\}_{i=1}^{N_P^{\text{dof}}}$  as the canonical basis for  $V_k^h(P)$ , we can represent each  $v_h \in V_k^h(P)$  in terms of its degrees of freedom by means of a Lagrange interpolation:

$$v_h = \sum_{i=1}^{N_P^{\text{dof}}} \text{dof}_i(v_h) \varphi_i, \tag{24}$$

where the usual interpolation property holds true:

$$\text{dof}_i(\varphi_j) = \delta_{ij}, \quad i, j = 1, \dots, N_P^{\text{dof}}. \tag{25}$$

Given the mesh  $\mathcal{T}_\Omega$ , the global conforming virtual element space  $V_k^h \in H^1(\Omega)$  is obtained by gluing together all the elemental spaces  $V_k^h(P)$ , i.e.:

$$V_k^h = \left\{ v_h \in H^1(P) : v_h|_P \in V_k^h(P) \, \forall P \in \mathcal{T}_\Omega \right\}. \tag{26}$$

### 3.4. Elliptic projection operator

Since functions in the virtual element space  $V_k^h(P)$  are not explicitly known, we need to resort to a projection operator  $\Pi_{P,k}^\nabla : V_k^h(P) \rightarrow \mathbb{P}_k(P)$  that maps functions from the virtual element space to the polynomial space of degree  $k$ , which is indeed known. Such projector will play a fundamental role in what follows and it is defined up to a constant by the orthogonality condition

$$\int_P \nabla p_k \cdot \nabla (\Pi_{P,k}^\nabla v_h - v_h) d\mathbf{x} = 0, \quad \forall p_k \in \mathbb{P}_k(P). \tag{27}$$

In order to fix the projection on constants as well, we need to separately define the projector operator  $P_0 : V_k^h(P) \rightarrow \mathbb{P}_0(P)$  requiring that

$$P_0(\Pi_{P,k}^\nabla v_h - v_h) = 0. \tag{28}$$

Among the many possible options for  $P_0$ , as forwarded in [30] we choose to define

$$P_0 v_h := \frac{1}{N_E} \sum_{i=1}^{N_E} v_h(\mathbf{x}_i) \quad \text{for } k = 1, \tag{29}$$

$$P_0 v_h := \frac{1}{|P|} \int_P v_h d\mathbf{x} \quad \text{for } k \geq 2. \tag{30}$$

It turns out that the projection  $\Pi_{P,k}^\nabla v_h$  is computable using only the degrees of freedom of  $v_h$ . Indeed, since  $\mathcal{M}_k(P)$  is a basis for  $\mathbb{P}_k(P)$ , from (27) we can write

$$\int_P \nabla m_\alpha \cdot \nabla (\Pi_{P,k}^\nabla v_h - v_h) d\mathbf{x} = 0, \quad \alpha = 1, \dots, n_k, \tag{31}$$

and since  $\Pi_{P,k}^\nabla v_h$  is an element of  $\mathbb{P}_k(P)$ , it can be represented in the basis  $\mathcal{M}_k(P)$  as well:

$$\Pi_{P,k}^\nabla v_h = \sum_{\beta=1}^{n_k} s^\beta m_\beta. \tag{32}$$

Consequently, Eq. (31) becomes:

$$\sum_{\beta=1}^{n_k} s^\beta \int_P \nabla m_\alpha \cdot \nabla m_\beta d\mathbf{x} = \int_P \nabla m_\alpha \cdot \nabla v_h d\mathbf{x}, \quad \alpha = 1, \dots, n_k, \tag{33}$$

which constitutes a linear system of  $n_k$  equations in the  $n_k$  unknowns  $s^\beta$ . The system is still indeterminate though, and such indeterminacy is overcome by adding conditions (29):

$$\sum_{\beta=1}^{n_k} s^\beta P_0 m_\beta = P_0 v_h. \tag{34}$$

Now, the linear system arising from (33) and (34) is solvable. In fact, the left-hand side of (33) entails the integration of known polynomials over  $P$ . As for the left-hand side, applying integration by parts we get

$$\int_P \nabla m_\alpha \cdot \nabla v_h d\mathbf{x} = - \int_P \Delta m_\alpha v_h d\mathbf{x} + \int_{\partial P} \frac{\partial m_\alpha}{\partial n} v_h dS. \tag{35}$$

The first term is explicitly computed from the internal degrees of freedom of  $v_h$ , while the second term contains a polynomial integrand which can be exactly integrated by evaluating it at the Gauss–Lobatto quadrature points along each edge  $e$ , where the edge degrees of freedom are located. Indeed, we recall that the basis functions  $\{\varphi\}_{i=1}^{N_P^{\text{dof}}}$ , which span  $v_h$ , are of Lagrangian type according to (24).

A projection  $\Pi_{P,k}^\nabla v_h$  can be easily obtained by computing the projections  $\Pi_{P,k}^\nabla \varphi_i$  of each of the virtual basis functions of  $V_k^h(P)$ . In fact, we have

$$\Pi_{P,k}^\nabla \varphi_i = \sum_{\alpha=1}^{n_k} s_i^\alpha m_\alpha, \quad i = 1, \dots, N_P^{\text{dof}},$$



where  $s_i^\alpha$  are the components of the solution vector of the system (33) with the additional condition (34), in which we choose  $v_h = \varphi_i$ . In matrix form, this system writes

$$\mathbf{G}\mathbf{\Pi}_k^{\star \nabla} = \mathbf{B}, \tag{36}$$

where  $\mathbf{G}$  and  $\mathbf{B}$  are respectively  $n_k \times n_k$  and  $n_k \times N_p^{\text{dof}}$  matrices such that:

$$(\mathbf{G})_{\alpha\beta} = P_0 m_\beta, \tag{37} \quad \text{for } \alpha = 1,$$

$$(\mathbf{G})_{\alpha\beta} = \int_P \nabla m_\alpha \cdot \nabla m_\beta \, d\mathbf{x}, \tag{38} \quad \text{for } \alpha \geq 2,$$

$$(\mathbf{B})_{\alpha i} = P_0 \varphi_i, \tag{39} \quad \text{for } \alpha = 1, \quad i = 1, \dots, N_p^{\text{dof}}$$

$$(\mathbf{B})_{\alpha i} = \int_P \nabla m_\alpha \cdot \nabla \varphi_i \, d\mathbf{x}, \tag{40} \quad \text{for } \alpha \geq 2, \quad i = 1, \dots, N_p^{\text{dof}},$$

and  $\mathbf{\Pi}_k^{\star \nabla}$  is the  $n_k \times N_p^{\text{dof}}$  matrix representation of operator  $\Pi_{P,k}^{\nabla}$  in the basis  $\mathcal{M}_k(P)$ , so that  $(\mathbf{\Pi}_k^{\star \nabla})_{\alpha i} = s_i^\alpha$ .

It is also possible to provide a matrix representation  $\mathbf{\Pi}_k^{\nabla}$  of operator  $\Pi_{P,k}^{\nabla}$  in the canonical basis of  $V_k^h(P)$ , by introducing the  $N_p^{\text{dof}} \times n_k$  matrix  $\mathbf{D}$  that performs a change of basis:

$$(\mathbf{D})_{i\alpha} = \text{dof}_i(m_\alpha), \quad i = 1, \dots, N_p^{\text{dof}}, \quad \alpha = 1, \dots, n_k. \tag{41}$$

Thus,  $\mathbf{\Pi}_k^{\nabla}$  is the  $N_p^{\text{dof}} \times N_p^{\text{dof}}$  matrix computed as

$$\mathbf{\Pi}_k^{\nabla} = \mathbf{D}\mathbf{\Pi}_k^{\star \nabla} = \mathbf{D}\mathbf{G}^{-1}\mathbf{B}. \tag{42}$$

### 3.5. $L_2$ projection operator

We also need to resort to the  $L_2$  projection operator  $\Pi_{P,k}^0 : V_k^h(P) \rightarrow \mathbb{P}_k(P)$  that maps functions from the virtual element space to the polynomial space of degree  $k$ . Such projector is defined by the orthogonality condition

$$\int_P p_k(\Pi_{P,k}^0 v_h - v_h) \, d\mathbf{x} = 0, \quad \forall p_k \in \mathbb{P}_k(P). \tag{43}$$

The projection  $\Pi_{P,k}^0 v_h$  can be evaluated using only the degrees of freedom of  $v_h$ . Indeed, since  $\Pi_{P,k}^0 v_h$  is an element of  $\mathbb{P}_k(P)$ , it can be represented in the basis  $\mathcal{M}_k(P)$ :

$$\Pi_{P,k}^0 v_h = \sum_{\beta=1}^{n_k} r^\beta m_\beta. \tag{44}$$

From this, it follows that condition (43) becomes

$$\sum_{\beta=1}^{n_k} r^\beta \int_P m_\alpha m_\beta \, d\mathbf{x} = \int_P m_\alpha v_h \, d\mathbf{x}, \quad \alpha = 1, \dots, n_k, \tag{45}$$

which is a linear system of  $n_k$  equations in the  $n_k$  unknowns  $r^\beta$ . While the left-hand side of (45) is readily computable, involving the integration of known polynomials over  $P$ , the right-hand side clearly is not. In fact, for a general  $k$  we know moments as degrees of freedom only for  $m_\alpha \in \mathbb{P}_{k-2}(P)$ . The idea here is to replace  $v_h$  in the right-hand side with its elliptic projection  $\Pi_{P,k}^{\nabla} v_h$  only for monomials of degree  $k$  and  $k - 1$ . Such system can be written in matrix form to give the  $n_k \times N_p^{\text{dof}}$  matrix representation  $\mathbf{\Pi}_k^{\star 0}$  of  $\Pi_{P,k}^0$  as follows:

$$\mathbf{H}\mathbf{\Pi}_k^{\star 0} = \mathbf{C}. \tag{46}$$

The matrices  $\mathbf{H}$  and  $\mathbf{C}$  are of dimensions  $n_k \times n_k$  and  $n_k \times N_p^{\text{dof}}$ , respectively, and they are explicitly defined as

$$(\mathbf{H})_{\alpha\beta} = \int_P m_\alpha m_\beta \, d\mathbf{x}, \quad \alpha, \beta = 1, \dots, n_k, \tag{47a}$$

$$(\mathbf{C})_{\alpha i} = \begin{cases} \int_P m_\alpha \varphi_i \, d\mathbf{x}, & \alpha = 1, \dots, n_{k-2}, & i = 1, \dots, N_p^{\text{dof}}, \\ \int_P m_\alpha \Pi_{P,k}^{\nabla} \varphi_i \, d\mathbf{x}, & n_{k-2} + 1 \leq \alpha \leq n_k, & i = 1, \dots, N_p^{\text{dof}}. \end{cases} \tag{47b}$$

It is also possible to provide the matrix representation  $\Pi_k^0$  of operator  $\Pi_{P,k}^0$  in the canonical basis of  $V_k^h(P)$ , by using the change of basis matrix  $\mathbf{D}$  (41), which is

$$\Pi_k^0 = \mathbf{D}\hat{\Pi}_k^0 = \mathbf{D}\mathbf{H}^{-1}\mathbf{C}. \tag{48}$$

In a similar fashion, we can also compute the matrix representations  $\Pi_{k-1}^0$  and  $\hat{\Pi}_{k-1}^0$  of the  $L_2$  projection onto  $\mathbb{P}_{k-1}(P)$ . To this end, we solve the linear system

$$\mathbf{H}'\hat{\Pi}_{k-1}^0 = \mathbf{C}', \tag{49}$$

where the  $n_{k-1} \times n_{k-1}$  matrix  $\mathbf{H}'$  is obtained by taking the first  $n_{k-1}$  rows and columns of matrix  $\mathbf{H}$ , defined in (47a), and the  $n_{k-1} \times N_P^{\text{dof}}$  matrix  $\mathbf{C}'$  is obtained by taking the first  $n_{k-1}$  rows of matrix  $\mathbf{C}$ , defined in (47b).

Finally, we obtain

$$\Pi_{k-1}^0 = \mathbf{D}'\hat{\Pi}_{k-1}^0, \tag{50}$$

where the  $N_P^{\text{dof}} \times n_{k-1}$  matrix  $\mathbf{D}'$  is computed by taking the first  $n_{k-1}$  columns of matrix  $\mathbf{D}$ , defined in (41).

### 3.6. Finite volume space

In this work, the leading numerical scheme relies on the finite volume paradigm, which is used to discretize the explicit terms in (7) and to store the numerical solution within each control volume  $P_i$  for every time level  $t^n$ . Finite volume schemes represent the vector of conserved variables  $\mathbf{Q}$  as cell averages referred to each polygonal element  $P$ :

$$\mathbf{Q}_i^n := \frac{1}{|P_i|} \int_{P_i} \mathbf{Q}(\mathbf{x}, t^n) d\mathbf{x}, \tag{51}$$

which obviously belong to the space of polynomials of degree zero ( $k = 0$ ), since they are constant states within each cell. To make notation easier, we omit the subscript  $i$  referring to the cell  $P_i$  and the superscript  $n$  pointing to the current time level.

To achieve higher order of accuracy, a non-linear reconstruction of the finite volume solution (51) is carried out, hence obtaining an approximation of the solution of degree  $k > 0$  which is addressed with high order polynomials  $\mathbf{w}(\mathbf{x})$  while avoiding spurious oscillations in the proximity of eventual discontinuities (further details on the high order reconstruction in the FV space will be given in Section 3.9). Consequently, the reconstructed solution lies in the space of polynomials  $\mathbb{P}_k$  and it is written in terms of an expansion of a set of modal basis functions  $\beta_\ell$  with corresponding  $n_k$  degrees of freedom  $\hat{\mathbf{w}}_\ell$ :

$$\mathbf{w}(\mathbf{x}) = \sum_{\ell=1}^{n_k} \beta_\ell(\mathbf{x}) \hat{\mathbf{w}}_\ell. \tag{52}$$

The basis functions  $\beta_\ell$  are given by a set of conservative Taylor functions that can be directly retrieved from the monomials (20):

$$\beta_\ell(\mathbf{x})|_P = m_\kappa - \frac{c}{|P|} \int_P m_\kappa d\mathbf{x}. \tag{53}$$

The coefficient  $c$  ensures the conservation property of the basis functions, thus we have  $c = 0$  for  $|\kappa| = 0$ , and  $c = 1$  for  $|\kappa| > 0$ . We remark that the conservation property of the basis functions (53) means that

$$\frac{1}{|P|} \int_P \sum_{\ell=1}^{n_k} \beta_\ell(\mathbf{x}) d\mathbf{x} = 1, \tag{54}$$

thus, the first degree of freedom of each element  $P$  (i.e. the one identified by  $\ell = 1$ ) represents the cell average value, in the finite volume sense.

### 3.7. Semi-discrete semi-implicit IMEX scheme in time

Concerning the temporal discretization of the equations, let us start by presenting the first order in time semi-discrete scheme according to the splitting (7), which reads:

$$\frac{\mathbf{Q}^{n+1} - \mathbf{Q}^n}{\Delta t} + \nabla \cdot \mathbb{F}_E(\mathbf{Q}^n) + \nabla \cdot \mathbb{F}_I(\mathbf{Q}^{n+1}, \nabla \mathbf{Q}^{n+1}) + \mathbb{B}_E(\mathbf{Q}^n) \cdot \nabla \mathbf{Q}^n + \mathbb{B}_I(\mathbf{Q}^n) \cdot \nabla \mathbf{Q}^{n+1} = \mathbf{0}. \quad (55)$$

*Incompressible Navier–Stokes model.* For the INS equations (2), the semi-discrete scheme expressed in (55) leads to

$$\nabla \cdot \mathbf{v}^{n+1} = 0, \quad (56a)$$

$$\frac{\mathbf{v}^{n+1} - \mathbf{v}^n}{\Delta t} + \nabla \cdot (\mathbf{v}^n \otimes \mathbf{v}^n) - \nu \Delta \mathbf{v}^{n+1} + \nabla p^{n+1} = \mathbf{0}, \quad (56b)$$

which is solved as follows. From the momentum equation we obtain a provisional velocity field  $\mathbf{v}^*$ , which already contains the computation of both explicit convection and implicit viscous contribution, that is given by

$$\mathbf{v}^* - \Delta t \nu \Delta \mathbf{v}^* = F_v^n - \Delta t \nabla p^n, \quad F_v^n = \mathbf{v}^n - \Delta t \nabla \cdot (\mathbf{v}^n \otimes \mathbf{v}^n). \quad (57)$$

Notice that the discretization of the convection terms is simply referred to with  $F_v^n$  and that we use the pressure gradient at time  $t^n$  to supplement the provisional velocity with physical information about the pressure field, as proposed in [66]. Once the velocity field  $\mathbf{v}^*$  is determined upon the solution of (57), the momentum equation (56b) is given by

$$\frac{\mathbf{v}^{n+1} - \mathbf{v}^*}{\Delta t} + \nabla p^{n+1} - \nabla p^n = \mathbf{0}. \quad (58)$$

The above equation can be inserted into the continuity Eq. (56a) yielding an elliptic equation for the unknown pressure  $p^{n+1}$ :

$$\Delta t \Delta p^{n+1} = \nabla \cdot \mathbf{v}^* + \Delta t \Delta p^n. \quad (59)$$

The pressure system (59) is solved at the aid of the GMRES method [67], where we prescribe a tolerance  $\delta_0 = 10^{-12}$  to stop the iterative procedure. Once the new pressure is known, the velocity field is updated according to (58).

**Theorem 1 (Divergence-free Constraint).** *Assuming periodic boundary conditions on  $\partial\Omega \in \mathbb{R}$ , the semi-discrete scheme (57)–(59) satisfies the divergence-free constraint on the velocity field:*

$$\nabla \cdot \mathbf{v}^{n+1} = 0. \quad (60)$$

**Proof.** We can solve the system by substitution, hence inserting (57) into (59) leads to

$$\mathbf{v}^{n+1} = \mathbf{v}^n - \Delta t \nabla \cdot (\mathbf{v}^n \otimes \mathbf{v}^n) + \Delta t \nu \Delta \mathbf{v}^* - \Delta t \nabla p^n + \Delta t \nabla p^n - \Delta t \nabla p^{n+1}. \quad (61)$$

Taking the divergence of the above equation yields

$$\begin{aligned} \nabla \cdot \mathbf{v}^{n+1} &= \nabla \cdot (\mathbf{v}^n - \Delta t \nabla \cdot (\mathbf{v}^n \otimes \mathbf{v}^n) + \Delta t \nu \Delta \mathbf{v}^* - \Delta t \nabla p^n) + \Delta t \Delta p^n - \Delta t \Delta p^{n+1}, \\ &= \nabla \cdot \mathbf{v}^* + \Delta t \Delta p^n - \Delta t \Delta p^{n+1}. \end{aligned} \quad (62)$$

The right hand side of (62) is indeed the elliptic equation (59), thus it vanishes, and the sought divergence-free constraint is preserved (60).  $\square$

We point out that the semi-discrete scheme (57)–(59) is equivalent to a splitting method in which we consider the following set of equations:

$$\nabla \cdot \mathbf{v}^{n+1} = 0, \quad (63a)$$

$$\frac{\mathbf{v}^* - \mathbf{v}^n}{\Delta t} + \nabla \cdot (\mathbf{v}^n \otimes \mathbf{v}^n) - \nu \Delta \mathbf{v}^* + \nabla p^n = \mathbf{0}. \quad (63b)$$

$$\frac{\mathbf{v}^{n+1} - \mathbf{v}^*}{\Delta t} - \nabla p^n + \nabla p^{n+1} = \mathbf{0}. \quad (63c)$$

Shallow water equations. For the SWE model (3), the first order semi-discrete scheme in time (55) reads:

$$\frac{\eta^{n+1} - \eta^n}{\Delta t} + \nabla \cdot \mathbf{q}^{n+1} = 0, \tag{64a}$$

$$\frac{\mathbf{q}^{n+1} - \mathbf{q}^n}{\Delta t} + \nabla \cdot (\mathbf{v}^n \otimes \mathbf{q}^n) + g H^n \nabla \eta^{n+1} = \mathbf{0}. \tag{64b}$$

Notice that the last equation for the bottom bathymetry is neglected since  $b(\mathbf{x})$  is not time dependent. Following [5], the above system is solved by substitution, inserting (64b) into (64a), which leads to a wave equation for the unknown  $\eta^{n+1}$ :

$$\eta^{n+1} + \Delta t^2 g \nabla \cdot (H^n \nabla \eta^{n+1}) = \eta^n - \Delta t \nabla \cdot F_{\mathbf{q}}^n, \quad F_{\mathbf{q}}^n = \mathbf{q}^n - \Delta t \nabla \cdot (\mathbf{v}^n \otimes \mathbf{q}^n), \tag{65}$$

where  $F_{\mathbf{q}}^n$  denotes again the contribution of the non-linear explicit terms. As for the INS equations, the above linear system is solved with the GMRES method up to the prescribed tolerance  $\delta_0$ . Once  $\eta^{n+1}$  is computed, it can be used to straightforwardly obtain  $\mathbf{q}^{n+1}$  through (64b).

**Theorem 2 (Well-balance Property).** Assuming periodic boundary conditions on  $\partial\Omega \in \mathbb{R}$  and assuming the following initial condition

$$\eta(\mathbf{x}, 0) = \eta_0, \quad \mathbf{v}(\mathbf{x}, 0) = \mathbf{0}, \quad b(\mathbf{x}) \neq 0, \tag{66}$$

the semi-discrete scheme (64) is well-balanced in the sense of [68].

**Proof.** The initial condition implies that  $\eta^n = \eta_0$  and  $\mathbf{v}^n = \mathbf{0}$ , hence  $\mathbf{q}^n = \mathbf{0}$ . Consequently, in the absence of discontinuities in the numerical solution, we get a vanishing non-linear convective contribution, i.e.  $F_{\mathbf{q}}^n = \mathbf{0}$ . More precisely, the numerical dissipation associated to the numerical flux of the convective term  $\nabla \cdot (\mathbf{v}^n \otimes \mathbf{q}^n)$  is exactly zero for any constant state, included  $\mathbf{q}^n = \mathbf{0}$ . Thus, the wave equation (65) reduces to

$$\eta^{n+1} - \Delta t^2 g \nabla \cdot (H^n \nabla \eta^{n+1}) = \eta^n, \quad F_{\mathbf{q}}^n = \mathbf{0}, \tag{67}$$

for which  $\eta^{n+1} = \eta^n = \eta_0$  is an admissible solution. From this it follows that  $\nabla \eta^{n+1} = 0$ . The discharge equation is then updated according to (64b), meaning that

$$\mathbf{q}^{n+1} = \mathbf{0} - \Delta t g H^n \cdot 0 = \mathbf{0}. \tag{68}$$

It is therefore evident that the semi-discrete scheme (64) can preserve stationary solutions of the shallow water system of the form given by (66) with arbitrary bathymetry.  $\square$

High order semi-implicit IMEX schemes. The class of semi-implicit Implicit–Explicit (IMEX) Runge–Kutta methods is used to attain higher order accuracy in time [4]. An IMEX Runge–Kutta scheme is a multi-step method characterized by two  $s \times s$  triangular matrices, the explicit one,  $\tilde{A} = (\tilde{a}_{kj})$ , with  $\tilde{a}_{kj} = 0$  for  $j \geq k$ , and the implicit one,  $A = (a_{kj})$ , with  $a_{kj} = 0$  for  $j > k$ , and by the weights vectors  $\tilde{b} = (\tilde{b}_1, \dots, \tilde{b}_s)^T$ ,  $b = (b_1, \dots, b_s)^T$ , where  $s$  identifies the number of implicit Runge–Kutta stages. They are usually defined by the following explicit (on the left) and implicit (on the right) Butcher tableau:

$$\begin{array}{c|c} \tilde{c} & \tilde{A} \\ \hline & \tilde{b} \end{array} \quad \begin{array}{c|c} c & A \\ \hline & b \end{array}$$

whose characterization provides the time accuracy of the scheme. According to [4], the governing equations (1) are written under the form of an autonomous system, that is

$$\frac{\partial \mathbf{Q}}{\partial t} = \mathcal{H}(\mathbf{Q}_E(t), \mathbf{Q}_I(t)), \quad \forall t > t_0, \quad \text{with} \quad \mathbf{Q}(t_0) = \mathbf{Q}_0, \tag{69}$$

where  $\mathbf{Q}_0$  defines the initial condition at time  $t_0$ . The semi-implicit IMEX technique is a method-of-lines (MOL) strategy, hence the function  $\mathcal{H}$  represents the spatial approximation of the conservative fluxes as well as the non-conservative products in (1). The arguments of  $\mathcal{H}$  undergo an explicit or implicit discretization in accordance to the

flux splitting (7). Consequently, a partitioned system with  $\mathbf{Q} = (\mathbf{Q}_E, \mathbf{Q}_I)$  is obtained:

$$\begin{cases} \frac{\partial \mathbf{Q}_E}{\partial t} = \mathcal{H}(\mathbf{Q}_E, \mathbf{Q}_I) \\ \frac{\partial \mathbf{Q}_I}{\partial t} = \mathcal{H}(\mathbf{Q}_E, \mathbf{Q}_I) \end{cases}, \tag{70}$$

where the number of unknowns has been doubled. However, for specific choices of the IMEX scheme and for autonomous systems this duplication is only apparent [4]. We can easily assume that the governing equations (2) and (3) are autonomous, since no explicit time dependency is present in the function  $\mathcal{H}$ . Therefore, only one set of stage fluxes needs to be computed, which, at each stage  $i = 1, \dots, s$  can be evaluated as

$$k_i = \mathcal{H} \left( \mathbf{Q}_E^n + \Delta t \sum_{j=1}^s \tilde{a}_{ij} k_j, \mathbf{Q}_I^n + \Delta t \sum_{j=1}^s a_{ij} k_j \right), \quad 1 \leq i \leq s. \tag{71}$$

A semi-implicit IMEX Runge–Kutta method is then obtained as follows. Let us first set  $\mathbf{Q}_E^n = \mathbf{Q}_I^n = \mathbf{Q}^n$ , then the stage fluxes for  $i = 1, \dots, s$  are calculated as

$$\mathbf{Q}_E^i = \mathbf{Q}_E^n + \Delta t \sum_{j=1}^{i-1} \tilde{a}_{ij} k_j, \quad 2 \leq i \leq s, \tag{72a}$$

$$\tilde{\mathbf{Q}}_I^i = \mathbf{Q}_E^n + \Delta t \sum_{j=1}^{i-1} a_{ij} k_j, \quad 2 \leq i \leq s, \tag{72b}$$

$$k_i = \mathcal{H}(\mathbf{Q}_E^i, \tilde{\mathbf{Q}}_I^i + \Delta t a_{ii} k_i), \quad 1 \leq i \leq s. \tag{72c}$$

Finally, the numerical solution is updated with

$$\mathbf{Q}^{n+1} = \mathbf{Q}^n + \Delta t \sum_{i=1}^s b_i k_i. \tag{73}$$

We remark that Eq. (72c) gives rise to a formally implicit step with the solution of a system for  $k_i$ , that corresponds to the elliptic equations (59) and (65). The final update of the solution (73) is done using the implicit weights  $b^\top$  that are assumed to be equal to the explicit ones  $\tilde{b}^\top$ . Furthermore, the stage fluxes  $k_i$  in (71) are the same for both explicit and implicit conserved vectors  $\mathbf{Q}_E$  and  $\mathbf{Q}_I$ , therefore the system is actually not doubled, since there is indeed only one set of numerical solution. The Butcher tableaux for the IMEX schemes chosen to reach first, second and third order of accuracy in time are given in Appendix.

### 3.8. Asymptotic-Preserving property

Numerical schemes satisfy the Asymptotic-Preserving (AP) property if they provide a consistent discretization of the asymptotic limit model of the governing equations. To prove the fulfilment of this property, we first derive the asymptotic limit of the PDE systems (2) and (3), and then we study the AP property at the semi-discrete level for the schemes (56) and (64). The high order version of the schemes will also be compliant with the AP property, thanks to the semi-implicit IMEX strategy, as fully detailed in [4]. For this reason, in this framework we limit ourselves to consider only the first order semi-discrete schemes presented in the previous section.

Let us assume the computational domain  $\Omega(\mathbf{x})$  to be assigned with periodic boundary conditions on  $\partial\Omega$  and let us introduce the  $k$ th order Chapman–Enskog expansion of a generic variable  $\phi(\mathbf{x}, t)$  in powers of the non-dimensional stiffness parameter  $\varepsilon$ , that reads

$$\phi(\mathbf{x}, t) = \phi_{(0)}(\mathbf{x}, t) + \varepsilon \phi_{(1)}(\mathbf{x}, t) + \varepsilon^2 \phi_{(2)}(\mathbf{x}, t) + \dots + \mathcal{O}(\varepsilon^k). \tag{74}$$

As previously introduced in Section 2, the Reynolds number  $\text{Re}$  is the stiffness parameter for the incompressible Navier–Stokes equations (5), while the Froude number  $\text{Fr}$  is used to characterize the scaling of the shallow water equations (6).

*Incompressible Navier–Stokes equations.* Application of the expansion (74) to the rescaled governing PDE (5) and collection of the like powers of  $\varepsilon$  yields the following  $k$ th leading order equations for  $k \in \{0, -1\}$ .

- $\mathcal{O}(\varepsilon^0)$ :

$$\nabla \cdot \mathbf{v}_{(0)} = 0, \tag{75a}$$

$$\partial_t \mathbf{v}_{(0)} + \nabla \cdot (\mathbf{v}_{(0)} \otimes \mathbf{v}_{(0)}) + \nabla p_{(1)} - \nu \Delta \mathbf{v}_{(1)} = \mathbf{0}. \tag{75b}$$

- $\mathcal{O}(\varepsilon^{-1})$ :

$$\nabla p_{(0)} - \nu \Delta \mathbf{v}_{(0)} = \mathbf{0}. \tag{76}$$

Notice that at the first order asymptotic expansion we retrieve the diffusive Stokes equation [69].

**Theorem 3** (*Asymptotic-Preserving Property*). *Assuming periodic boundary conditions on  $\partial\Omega \in \mathbb{R}$ , the semi-discrete scheme (56)–(57) is a consistent approximation of the Stokes system (76) at the first order asymptotic expansion in the asymptotic limit ( $\varepsilon \rightarrow 0$ ).*

**Proof.** We assume that the following expansions hold true for the discrete variables at any generic time  $t^n$ :

$$\mathbf{v}^n(\mathbf{x}) = \mathbf{v}_{(0)}^n(\mathbf{x}) + \varepsilon \mathbf{v}_{(1)}^n(\mathbf{x}), \quad p^n(\mathbf{x}) = p_{(0)}^n(\mathbf{x}) + \varepsilon p_{(1)}^n(\mathbf{x}). \tag{77}$$

We recall that the semi-discrete scheme (56)–(57) can be explicitly written as a splitting method for the viscous terms according to (63), and its rescaled version with the dimensionless variables (4) writes

$$\nabla \cdot \mathbf{v}^{n+1} = 0, \tag{78a}$$

$$\frac{\mathbf{v}^* - \mathbf{v}^n}{\Delta t} + \nabla \cdot (\mathbf{v}^n \otimes \mathbf{v}^n) - \frac{\nu}{\varepsilon} \Delta \mathbf{v}^* + \frac{1}{\varepsilon} \nabla p^n = \mathbf{0}. \tag{78b}$$

$$\frac{\mathbf{v}^{n+1} - \mathbf{v}^*}{\Delta t} - \frac{1}{\varepsilon} \nabla p^n + \frac{1}{\varepsilon} \nabla p^{n+1} = \mathbf{0}. \tag{78c}$$

Inserting (77) into the semi-discrete scheme (78) and retaining only first order terms of the expansions leads to

$$-\nu \Delta \mathbf{v}_{(0)}^* + \nabla p_{(0)}^n = \mathbf{0}. \tag{79a}$$

$$-\nabla p_{(0)}^n + \nabla p_{(0)}^{n+1} = \mathbf{0}. \tag{79b}$$

This obviously yields the discrete Stokes model

$$\nabla p_{(0)}^{n+1} = \nu \Delta \mathbf{v}_{(0)}^*, \tag{80}$$

which is a consistent discretization of the limit model (76). Indeed, at the first order asymptotic expansion the divergence-free constraint disappears, and we have that  $\mathbf{v}^* = \mathbf{v}^{n+1}$ .  $\square$

*Shallow water equations.* Following the same reasoning, we can identify the limit model of the shallow water system by using the expansion (77) in the rescaled model (6) and by collecting like powers of  $\varepsilon$  for  $k \in \{0, -1, -2\}$ .

- $\mathcal{O}(\varepsilon^0)$ :

$$\partial_t \eta_{(0)} + \nabla \cdot ((\eta_0 - b) \mathbf{v}_0) = 0, \tag{81a}$$

$$\partial_t \mathbf{q}_{(0)} + \nabla \cdot (\mathbf{v}_{(0)} \otimes \mathbf{q}_{(0)}) + \eta_{(2)} \nabla \eta_{(0)} + \eta_{(1)} \nabla \eta_{(1)} + (\eta_{(0)} - b) \nabla \eta_{(2)} = \mathbf{0}. \tag{81b}$$

- $\mathcal{O}(\varepsilon^{-1})$ :

$$\eta_{(1)} \nabla \eta_{(0)} + (\eta_{(0)} - b) \nabla \eta_{(1)} = \mathbf{0}. \tag{82}$$

- $\mathcal{O}(\varepsilon^{-2})$ :

$$(\eta_{(0)} - b) \nabla \eta_{(0)} = \mathbf{0}. \tag{83}$$

From (83) it directly follows that  $\eta_{(0)} = \eta_{(0)}(t)$ , which can be used in (82) to obtain  $\eta_{(1)} = \eta_{(1)}(t)$  under the assumption of  $H(\mathbf{x}, t) > 0$ . Since we have assumed periodic boundaries, integration of the mass equation over the

computational domain and application of Gauss theorem leads to vanishing fluxes, hence determining that the free surface elevation  $\eta_{(0)}$  is constant in space as well. This means that the total water depth also becomes constant in both space and time, that is  $H(\mathbf{x}, t) = \text{const}$ . Then, the low Froude shallow water system writes

$$\nabla \cdot ((\eta_0 - b) \mathbf{v}_0) = 0, \tag{84a}$$

$$\partial_t \mathbf{q}_{(0)} + \nabla \cdot (\mathbf{v}_{(0)} \otimes \mathbf{q}_{(0)}) + (\eta_{(0)} - b) \nabla \eta_{(2)} = \mathbf{0}. \tag{84b}$$

The reader is referred to [29] for further details on the derivation of the limit model for the SWE system.

**Theorem 4 (Asymptotic Preserving Property).** *Assuming periodic boundary conditions on  $\partial\Omega \in \mathbb{R}$ , the semi-discrete scheme (64) is a consistent approximation of the low Froude shallow water system (84) at the leading order asymptotic expansion in the asymptotic limit ( $\varepsilon \rightarrow 0$ ).*

**Proof.** We assume that the following expansions hold true for the discrete variables at any generic time  $t^n$ :

$$\eta^n(\mathbf{x}) = \eta_{(0)}^n(\mathbf{x}) + \varepsilon^2 \eta_{(2)}^n(\mathbf{x}), \quad \mathbf{v}^n(\mathbf{x}) = \mathbf{v}_{(0)}^n(\mathbf{x}) + \varepsilon \mathbf{v}_{(1)}^n(\mathbf{x}). \tag{85}$$

We remark that  $\eta_{(0)}^n(\mathbf{x}) = \eta_{(0)}$  is constant in space and time and that  $\varepsilon^2 \eta_{(2)}^n(\mathbf{x})$  is a perturbation of the free surface level, thus the constant total water depth at zeroth order is given by  $H_{(0)}^n(\mathbf{x}) = \eta_{(0)}(\mathbf{x}) - b(\mathbf{x})$ . Inserting (85) into the semi-discrete scheme (64) and retaining only zeroth order terms of the expansions yields

$$\nabla \cdot (H_{(0)} \mathbf{v}_{(0)})^{n+1} = 0, \tag{86a}$$

$$\frac{(H_{(0)} \mathbf{v}_{(0)})^{n+1} - (H_{(0)} \mathbf{v}_{(0)})^n}{\Delta t} + \nabla \cdot (\mathbf{v}_{(0)}^n \otimes (H_{(0)} \mathbf{v}_{(0)})^n) + H_{(0)}^n \nabla \eta_{(2)}^{n+1} = \mathbf{0}, \tag{86b}$$

that is a consistent discretization of the low Froude shallow water system (84). Furthermore, if we formally substitute the discharge equation (86b) into the mass equation (86a), we obtain the corresponding rescaled version of the elliptic equation (65) in the asymptotic limit, namely

$$\Delta t H_{(0)} \nabla \cdot \nabla \eta_{(2)}^{n+1} = \nabla \cdot (H_{(0)} \mathbf{v}_{(0)})^n - \Delta t \nabla \cdot (\nabla \cdot (\mathbf{v}_{(0)}^n \otimes (H_{(0)} \mathbf{v}_{(0)})^n)). \tag{87}$$

It is interesting to notice that the above Eq. (87), divided by the constant water depth  $H_{(0)}$ , is the classical Poisson equation for the pressure correction of the projection stage for the incompressible Navier–Stokes model, which we solve in (59). The only difference is that in (59) we use a splitting for the viscous terms, hence yielding the additional term related to the pressure gradient at the current time  $t^n$ . We can therefore conclude that the semi-discrete scheme (64) is a consistent discretization of the limit model (84).  $\square$

### 3.9. Spatial finite volume discretization of the explicit terms

The explicit terms are only concerned with the conservative fluxes  $\mathbb{F}_E(\mathbf{Q})$  in (7), and they are discretized using a classical finite volume scheme tailored for general polygonal meshes. They are constructed upon integration of the governing PDE over the control volume  $P_i$  and application of Gauss theorem, hence leading to

$$\partial_t \int_{P_i} \mathbf{Q}_i d\mathbf{x} = - \int_{\partial P_i} \mathbb{F}_E(\mathbf{Q}) \cdot \mathbf{n} dS, \tag{88}$$

which indeed gives the explicit fluxes  $\mathcal{H}(\mathbf{Q}_E)$  in (69). The contribution of the explicit fluxes is then collected in the term  $F_{\mathbf{Q},i}^n$ , which can be either  $F_{\mathbf{v}}^n$  in (57) or  $F_{\mathbf{q}}^n$  in (65):

$$F_{\mathbf{Q},i}^n = \mathbf{Q}_i^n - \frac{\Delta t}{|P_i|} \sum_{e=1}^{N_E} \int_{\partial P_i, e} \mathcal{F}(\mathbf{w}_e^{-,n}, \mathbf{w}_e^{+,n}, \mathbf{n}_{P,e}) dS. \tag{89}$$

The Rusanov-type numerical flux  $\mathcal{F}$  is a function of three arguments, namely the left and the right state  $\mathbf{w}_e^{-,n}, \mathbf{w}_e^{+,n}$ , with respect to the edge  $e$ , and the associated outward pointing unit normal vector  $\mathbf{n}_{P,e}$ . The numerical flux is thus defined by

$$\mathcal{F}(\mathbf{w}_e^{-,n}, \mathbf{w}_e^{+,n}, \mathbf{n}_{P,e}) = \frac{1}{2} (\mathbb{F}_E(\mathbf{w}_e^{-,n}) + \mathbb{F}_E(\mathbf{w}_e^{+,n})) \cdot \mathbf{n}_{P,e} - \frac{1}{2} |s_{\max}| (\mathbf{w}_e^{+,n} - \mathbf{w}_e^{-,n}). \tag{90}$$

Notice that the numerical dissipation  $s_{\max}$  is given by the maximum eigenvalue of the convective sub-systems, namely (11) or (13), related to the right and left states. Consequently, the numerical diffusion of the scheme is proportional to the flow velocity only, which is a remarkable property in the context of low Froude number flows for the SWE model, since the acoustic-gravity wave speed does not play any role and the numerical dissipation automatically tends to zero for  $\varepsilon \rightarrow 0$ . The left and right states can be simply given by the finite volume solution stored as cell averages (51) within each cell, being

$$\mathbf{w}_e^{-,n} = \mathbf{Q}_i^n, \quad \mathbf{w}_e^{+,n} = \mathbf{Q}_j^n, \tag{91}$$

where  $j$  is the index of the polygon which shares edge  $e$  with the cell  $P_i$ . Of course, the above choice yields only a first order in space accurate scheme. To improve the spatial order of accuracy, we perform a reconstruction technique that can effectively be used to feed the left and right state in the computation of the numerical flux function (90).

*CWENO reconstruction on polygonal meshes.* The aim of the reconstruction operator is to determine the expansion coefficient of the reconstruction polynomial (52), that will be subsequently employed for the computation of the high order numerical fluxes in the finite volume scheme (90). We choose to use the class of Central WENO (CWENO) schemes, which has been originally proposed in [70,71], and subsequently extended to deal with general polygonal meshes in [29,72,73]. The interested reader is referred to the aforementioned references for all the details about the CWENO procedure, in particular we point out Appendix B in [29]. Eventually, we obtain a reconstruction polynomial within each cell of arbitrary degree  $k$ , that is expressed according to (52). This polynomial is then used to evaluate the high order extrapolated states  $\mathbf{w}_e^{-,n}$  and  $\mathbf{w}_e^{+,n}$  in the finite volume scheme (89), hence improving the spatial accuracy of the scheme up to order  $k + 1$ .

### 3.10. Spatial virtual element discretization of the implicit terms

The implicit discretization is based on higher-order conforming virtual element approximations on the polygonal tessellation. Hereafter, we detail the virtual element formulation for both the INS and the SWE models.

*Incompressible Navier–Stokes model.* We consider the implicit contributions in the semi-discrete scheme (56) on domain  $\Omega$  and derive the weak form for both Eqs. (57) and (59), where pure Dirichlet boundary conditions  $\mathbf{v}^* = [g_x(\mathbf{x}), g_y(\mathbf{x})]^T$  and  $p^{n+1} = h(\mathbf{x})$  are applied on  $\partial\Omega$ . We explicit the components of  $\mathbf{v}^*$  on  $\Omega$  by writing  $\mathbf{v}^* = [v_x^*, v_y^*]^T$ . Let us denote by  $H_0^1(\Omega)$  the affine subspace of functions in the Sobolev space  $H^1(\Omega)$  whose trace is zero on  $\partial\Omega$ . Moreover, we denote by  $H_{g_x}^1(\Omega)$ ,  $H_{g_y}^1(\Omega)$  and  $H_h^1(\Omega)$  respectively the affine subspaces of functions in the Sobolev space  $H^1(\Omega)$  whose trace is equal to  $g_x(\mathbf{x})$ ,  $g_y(\mathbf{x})$  and  $h(\mathbf{x})$  on  $\partial\Omega$ . Multiplying component-wise the momentum equation (57) and the pressure equation (59) by a test function  $w \in H_0^1(\Omega)$ , integrating over the domain  $\Omega$  and applying the divergence theorem we get the following weak formulation for the INS model:

$$\begin{aligned} &\text{find } v_x^* \in H_{g_x}^1(\Omega), v_y^* \in H_{g_y}^1(\Omega), p^{n+1} \in H_h^1(\Omega) \text{ s.t.} \\ &\left\{ \begin{aligned} &\int_{\Omega} v_x^* w d\Omega + \Delta t \int_{\Omega} \nabla v_x^* \cdot \nabla w d\Omega = \int_{\Omega} (F_{v,x}^n - \Delta t \frac{\partial p^n}{\partial x}) w d\Omega \quad \forall w \in H_0^1(\Omega), \quad (\text{a}) \\ &\int_{\Omega} v_y^* w d\Omega + \Delta t \int_{\Omega} \nabla v_y^* \cdot \nabla w d\Omega = \int_{\Omega} (F_{v,y}^n - \Delta t \frac{\partial p^n}{\partial y}) w d\Omega \quad \forall w \in H_0^1(\Omega), \quad (\text{b}) \\ &\Delta t \int_{\Omega} \nabla p^{n+1} \cdot \nabla w d\Omega = - \int_{\Omega} (\nabla \cdot \mathbf{v}^* + \Delta t \Delta p^n) w d\Omega \quad \forall w \in H_0^1(\Omega). \quad (\text{c}) \end{aligned} \right. \tag{92} \end{aligned}$$

Given the polygonal tessellation of the space domain  $\mathcal{T}_{\Omega}$  (16), the local virtual element approximations  $v_{x,h}^*, v_{y,h}^* \in V_k^h(P)$  of the velocity field components  $v_x^*, v_y^*$  and the local virtual element approximation  $p_h^{n+1} \in V_k^h(P)$  of the pressure field  $p^{n+1}$  can be respectively represented, on each control volume  $P \in \mathcal{T}_{\Omega}$ , in terms of the canonical basis  $\{\varphi_i\}_{i=1}^{N_P^{\text{dof}}}$  of the local virtual element space  $V_k^h(P)$ , introduced in Section 3.3. Hence, we write

$$v_{x,h}^*(\mathbf{x})|_{x \in P} = \sum_{i=1}^{N_P^{\text{dof}}} \text{dof}_i(v_{x,h}^*) \varphi_i(\mathbf{x}), \tag{93a}$$



$$v_{y,h}^*(\mathbf{x})|_{x \in P} = \sum_{i=1}^{N_P^{\text{dof}}} \text{dof}_i(v_{y,h}^*)\varphi_i(\mathbf{x}), \tag{93b}$$

$$p_h^{n+1}(\mathbf{x})|_{x \in P} = \sum_{i=1}^{N_P^{\text{dof}}} \text{dof}_i(p_h^{n+1})\varphi_i(\mathbf{x}). \tag{93c}$$

Choosing  $w = \varphi_j$  in Eqs. (92) and plugging (93) into (92), we obtain the discrete version of the weak form on the cell  $P$ :

$$\mathbf{M}_P \hat{\mathbf{v}}_{x,P}^* + \Delta t \mathbf{K}_P \hat{\mathbf{v}}_{x,P}^* = \mathbf{F}_{m_x,P}^n, \tag{94a}$$

$$\mathbf{M}_P \hat{\mathbf{v}}_{y,P}^* + \Delta t \mathbf{K}_P \hat{\mathbf{v}}_{y,P}^* = \mathbf{F}_{m_y,P}^n, \tag{94b}$$

$$\Delta t \mathbf{K}_P \hat{\mathbf{p}}_P^{n+1} = \mathbf{F}_{c,P}^n, \tag{94c}$$

where vectors  $\hat{\mathbf{v}}_{x,P}^*$  and  $\hat{\mathbf{v}}_{y,P}^*$  are the degrees of freedom of the velocity field components  $v_{x,h}^*$  and  $v_{y,h}^*$  on  $P$  and vector  $\hat{\mathbf{p}}_P^{n+1}$  collects the degrees of freedom of the pressure field  $p_h^{n+1}$  on  $P$ . In (94), we also introduce the definition of the local mass and stiffness matrices  $\mathbf{M}_P$  and  $\mathbf{K}_P$ , respectively:

$$(\mathbf{M}_P)_{i,j} = \int_P \varphi_i \varphi_j \, d\mathbf{x}, \tag{95}$$

$$(\mathbf{K}_P)_{i,j} = \int_P \nabla \varphi_i \cdot \nabla \varphi_j \, d\mathbf{x}. \tag{96}$$

Furthermore, the local load terms  $\mathbf{F}_{m_x,P}^n$ ,  $\mathbf{F}_{m_y,P}^n$  and  $\mathbf{F}_{c,P}^n$  on the right-hand sides, which are known from the  $n$ -th time step, are given by

$$(\mathbf{F}_{m_x,P}^n)_i = \int_P (F_{v,x}^n - \Delta t \frac{\partial p^n}{\partial x}) \varphi_i \, d\mathbf{x}, \tag{97}$$

$$(\mathbf{F}_{m_y,P}^n)_i = \int_P (F_{v,y}^n - \Delta t \frac{\partial p^n}{\partial y}) \varphi_i \, d\mathbf{x}, \tag{98}$$

$$(\mathbf{F}_{c,P}^n)_i = - \int_P (\nabla \cdot \mathbf{v}^* + \Delta t \Delta p^n) \varphi_i \, d\mathbf{x}. \tag{99}$$

Matrices  $\mathbf{M}_P$  and  $\mathbf{K}_P$ , as well as the load terms  $\mathbf{F}_{m_x,P}^n$ ,  $\mathbf{F}_{m_y,P}^n$  and  $\mathbf{F}_{c,P}^n$ , are not readily computable since they involve functions that are unknown within the element interior. To approximate  $\mathbf{M}_P$ , we observe that for each virtual basis function  $\varphi_i(\mathbf{x})$  on  $P$  we can write the expansion

$$\varphi_i = \Pi_{P,k}^0 \varphi_i + (I - \Pi_{P,k}^0) \varphi_i,$$

where  $I$  is the identity operator, and plugging it into (95) we obtain

$$\begin{aligned} (\mathbf{M}_P)_{i,j} &= \int_P \varphi_i \varphi_j \, d\mathbf{x} = \int_P \Pi_{P,k}^0 \varphi_i \Pi_{P,k}^0 \varphi_j \, d\mathbf{x} + \int_P (I - \Pi_{P,k}^0) \varphi_i (I - \Pi_{P,k}^0) \varphi_j \, d\mathbf{x} \\ &\quad + \int_P \Pi_{P,k}^0 \varphi_i (I - \Pi_{P,k}^0) \varphi_j \, d\mathbf{x} + \int_P (I - \Pi_{P,k}^0) \varphi_i \Pi_{P,k}^0 \varphi_j \, d\mathbf{x}. \end{aligned} \tag{100}$$

We note that the last two terms in (100) vanish due to the orthogonality condition (43), yielding

$$(\mathbf{M}_P)_{i,j} = \int_P \Pi_{P,k}^0 \varphi_i \Pi_{P,k}^0 \varphi_j \, d\mathbf{x} + \int_P (I - \Pi_{P,k}^0) \varphi_i (I - \Pi_{P,k}^0) \varphi_j \, d\mathbf{x}. \tag{101}$$

The first term in (101) guarantees consistency and can be computed exactly, whereas the second one ensures stability and although it cannot be exactly computed, it can be approximated as follows:

$$\int_P (I - \Pi_{P,k}^0) \varphi_i (I - \Pi_{P,k}^0) \varphi_j \, d\mathbf{x} \simeq |P| \sum_{r=1}^{N_P^{\text{dof}}} \text{dof}_r((I - \Pi_{P,k}^0) \varphi_i) \text{dof}_r((I - \Pi_{P,k}^0) \varphi_j). \tag{102}$$

Here, we adopted the so-called  $\text{dof}_i - \text{dof}_i$  stabilization [30], although other choices are possible [74,75]. From simple algebraic manipulations, recalling (48), the sought approximation  $\mathbf{M}_P^h$  of matrix  $\mathbf{M}_P$  can be evaluated as the

sum of a consistency term and a stability term, reading

$$\mathbf{M}_P^h = \mathbf{C}^T \mathbf{H}^{-1} \mathbf{C} + |P|(\mathbf{I} - \Pi_k^0)^T (\mathbf{I} - \Pi_k^0). \tag{103}$$

Analogously, the integral in the definition of the stiffness matrix in (96) needs to be approximated, since its integrand is unknown in the element interior. Plugging the expansion

$$\varphi_i = \Pi_{P,k}^\nabla \varphi_i + (I - \Pi_{P,k}^\nabla) \varphi_i,$$

into (96), we obtain

$$\begin{aligned} (\mathbf{K}_P)_{i,j} = & \int_P \nabla \varphi_i \cdot \nabla \varphi_j \, d\mathbf{x} = \int_P \nabla \Pi_{P,k}^\nabla \varphi_i \cdot \nabla \Pi_{P,k}^\nabla \varphi_j \, d\mathbf{x} + \int_P \nabla (I - \Pi_{P,k}^\nabla) \varphi_i \cdot \nabla (I - \Pi_{P,k}^\nabla) \varphi_j \, d\mathbf{x} \\ & + \int_P \nabla (I - \Pi_{P,k}^\nabla) \varphi_i \cdot \nabla \Pi_{P,k}^\nabla \varphi_j \, d\mathbf{x} + \int_P \nabla \Pi_{P,k}^\nabla \varphi_i \cdot \nabla (I - \Pi_{P,k}^\nabla) \varphi_j \, d\mathbf{x}. \end{aligned} \tag{104}$$

Again, the last two terms in (104) vanish due to orthogonality conditions (27), leading to

$$(\mathbf{K}_P)_{i,j} = \int_P \nabla \Pi_{P,k}^\nabla \varphi_i \cdot \nabla \Pi_{P,k}^\nabla \varphi_j \, d\mathbf{x} + \int_P \nabla (I - \Pi_{P,k}^\nabla) \varphi_i \cdot \nabla (I - \Pi_{P,k}^\nabla) \varphi_j \, d\mathbf{x}. \tag{105}$$

Here, the first term ensures polynomial consistency and the second term is responsible for stability. In particular, the consistency term is exactly computable since it entails the integration of known polynomials, whilst the stability term must be approximated. For such approximation we adopt an expression analogue to (102), thus

$$\int_P \nabla (I - \Pi_{P,k}^\nabla) \varphi_i \cdot \nabla (I - \Pi_{P,k}^\nabla) \varphi_j \, d\mathbf{x} \simeq |P| \sum_{r=1}^{N_P^{\text{dof}}} \text{dof}_r((I - \Pi_{P,k}^\nabla) \varphi_i) \text{dof}_r((I - \Pi_{P,k}^\nabla) \varphi_j). \tag{106}$$

After some algebraic manipulations, recalling (42), the approximation  $\mathbf{K}_P^h$  of matrix  $\mathbf{K}_P$  can be evaluated as the sum of a consistency term and a stability term as follows:

$$\mathbf{K}_P^h = (\hat{\Pi}_k^\nabla)^T \mathbf{G} \hat{\Pi}_k^\nabla + |P|(\mathbf{I} - \Pi_k^\nabla)^T (\mathbf{I} - \Pi_k^\nabla). \tag{107}$$

Finally, to compute the approximations  $\mathbf{F}_{m_x,P}^{n,h}$ ,  $\mathbf{F}_{m_y,P}^{n,h}$  and  $\mathbf{F}_{c,P}^{n,h}$  of the load terms  $\mathbf{F}_{m_x,P}^n$ ,  $\mathbf{F}_{m_y,P}^n$  and  $\mathbf{F}_{c,P}^n$ , respectively, we resort to the  $L_2$  projector  $\Pi_{P,k-1}^0$  introduced in Section 3.5 and we write

$$(\mathbf{F}_{m_x,P}^{n,h})_i = \int_P (F_{v,x}^n - \Delta t \frac{\partial p^n}{\partial x}) \Pi_{P,k}^0 \varphi_i \, d\mathbf{x}, \tag{108}$$

$$(\mathbf{F}_{m_y,P}^{n,h})_i = \int_P (F_{v,y}^n - \Delta t \frac{\partial p^n}{\partial y}) \Pi_{P,k}^0 \varphi_i \, d\mathbf{x}, \tag{109}$$

$$(\mathbf{F}_{c,P}^{n,h})_i = - \int_P (\nabla \cdot \mathbf{v}^* + \Delta t \Delta p^n) \Pi_{P,k}^0 \varphi_i \, d\mathbf{x}. \tag{110}$$

The implicit terms are then computed by solving the following virtual element linear system:

$$\mathbf{M}_P^h \hat{\mathbf{v}}_{x,P}^* + \Delta t \mathbf{K}_P^h \hat{\mathbf{v}}_{x,P}^* = \mathbf{F}_{m_x,P}^{n,h}, \tag{111a}$$

$$\mathbf{M}_P^h \hat{\mathbf{v}}_{y,P}^* + \Delta t \mathbf{K}_P^h \hat{\mathbf{v}}_{y,P}^* = \mathbf{F}_{m_y,P}^{n,h}, \tag{111b}$$

$$\Delta t \mathbf{K}_P^h \hat{\mathbf{p}}_P^{n+1} = \mathbf{F}_{c,P}^{n,h}. \tag{111c}$$

*Shallow water equations.* For the SWE model, we consider the implicit contributions in the semi-discrete scheme (64). The weak form for Eq. (65), where pure Dirichlet boundary conditions  $\eta^{n+1} = \bar{\eta}(\mathbf{x})$  are applied on  $\partial\Omega$ , is now derived. We denote by  $H_{\bar{\eta}}^1(\Omega)$  the affine subspace of functions in  $H^1(\Omega)$  whose trace is equal to  $\bar{\eta}(\mathbf{x})$  on  $\partial\Omega$ . Multiplying (65) by a test function  $w \in H_0^1(\Omega)$ , integrating over the domain  $\Omega$  and applying the divergence theorem, the weak formulation reads:

$$\begin{aligned} \text{find } \eta^{n+1} \in H_{\bar{\eta}}^1(\Omega) \text{ s.t.} \\ \int_{\Omega} \eta^{n+1} w \, d\Omega - \Delta t^2 g \int_{\Omega} H^n \nabla \eta^{n+1} \cdot \nabla w \, d\Omega = \int_{\Omega} (\eta^n - \Delta t \nabla \cdot \mathbf{F}_q^n) w \, d\Omega \quad \forall w \in H_0^1(\Omega). \end{aligned} \tag{112}$$

Given the polygonal tessellation of the space domain  $\mathcal{T}_\Omega$ , the virtual element approximations of the free surface elevation  $\eta_h$  and the total water depth  $H_h$  can be represented, on each control volume  $P \in \mathcal{T}_\Omega$ , in terms of the canonical basis functions  $\{\varphi_i\}_{i=1}^{N_P^{\text{dof}}}$  of the virtual element space  $V_k^h(P)$  introduced in Section 3.3:

$$\eta_h(\mathbf{x})|_{\mathbf{x} \in P} = \sum_{i=1}^{N_P^{\text{dof}}} \text{dof}_i(\eta_h) \varphi_i(\mathbf{x}), \tag{113}$$

$$H_h(\mathbf{x})|_{\mathbf{x} \in P} = \sum_{i=1}^{N_P^{\text{dof}}} \text{dof}_i(H_h) \varphi_i(\mathbf{x}). \tag{114}$$

Choosing  $w = \varphi_j$ , and plugging (113) into (112) we obtain the discrete version of the weak form on the cell  $P$ , that is

$$\mathbf{M}_P \hat{\boldsymbol{\eta}}_P^{n+1} - \Delta t \mathbf{K}_P^n \hat{\boldsymbol{\eta}}_P^{n+1} = \mathbf{F}_{s,P}^n, \tag{115}$$

where  $\hat{\boldsymbol{\eta}}_P^{n+1}$  is the vector of degrees of freedom of  $\eta_h^{n+1}$  on  $P$ , while matrices  $\mathbf{M}_P$  and  $\mathbf{K}_P^n$  are defined as

$$(\mathbf{M}_P)_{i,j} = \int_P \varphi_i \varphi_j \, d\mathbf{x}, \tag{116}$$

$$(\mathbf{K}_P^n)_{i,j} = \int_P H^n \nabla \varphi_i \cdot \nabla \varphi_j \, d\mathbf{x}, \tag{117}$$

and the load term  $\mathbf{F}_{s,P}^n$  reads

$$(\mathbf{F}_{s,P}^n)_i = \int_P (\eta^n - \Delta t \nabla \cdot \mathbf{F}_q^n) \varphi_i \, d\mathbf{x}. \tag{118}$$

Matrices  $\mathbf{M}_P$  and  $\mathbf{K}_P^n$ , as well as the load term  $\mathbf{F}_{s,P}^n$ , are not readily computable since the respective integrands are unknown within the element interior. For the mass matrix  $\mathbf{M}_P$  we provide the approximation  $\mathbf{M}_P^h$  already defined in (107) for the INS equations. On the other hand, matrix  $\mathbf{K}_P^n$  in (117) differs from matrix  $\mathbf{K}_P$  derived in (96) for the INS model, because it contains the non-constant diffusion coefficient  $H^n$ , which is obviously space dependent. As shown in [33], in such a case, using the elliptic projector  $\Pi_{P,k}^\nabla$  to approximate  $\nabla v_h$ ,  $v_h \in V_k^h(P)$ , would entail the loss of optimal convergence rates. Instead, as suggested in [33], we employ the  $L_2$  projector operator  $\Pi_{P,k-1}^0$  on polynomials of degree  $k-1$  to approximate  $\nabla v_h$  within the consistency term of the corresponding approximated bilinear form. The computation of  $\Pi_{P,k-1}^0 \nabla v_h$  in terms of the degrees of freedom is carried out by constructing the matrix representation of the operator  $\Pi_{P,k-1}^0 \nabla$ . Such representation can be expressed in terms of the two matrices  $\hat{\Pi}_{k-1}^{0,x}$  and  $\hat{\Pi}_{k-1}^{0,y}$  given by

$$\begin{aligned} \hat{\Pi}_{k-1}^{0,x} &= \hat{\mathbf{H}}^{-1} \mathbf{E}^x, \\ \hat{\Pi}_{k-1}^{0,y} &= \hat{\mathbf{H}}^{-1} \mathbf{E}^y, \end{aligned}$$

where matrix  $\hat{\mathbf{H}}$  is obtained by taking the first  $n_{k-1}$  rows and columns of matrix  $\mathbf{H}$  defined in (47a). Matrices  $\mathbf{E}^x$  and  $\mathbf{E}^y$  are defined as follows:

$$(\mathbf{E}^x)_{i\alpha} = \int_P \varphi_{i,x} m_\alpha \, d\mathbf{x}, \quad (\mathbf{E}^y)_{i\alpha} = \int_P \varphi_{i,y} m_\alpha \, d\mathbf{x}, \quad \alpha = 1, \dots, n_{k-1}.$$

Hence, matrix  $\mathbf{K}_P^n$  can be approximated with the matrix  $\mathbf{K}_P^{n,h}$  which is computed as

$$\begin{aligned} (\mathbf{K}_P^{n,h})_{i,j} &= (\mathbf{K}_{c,P}^{n,h})_{i,j} + (\mathbf{K}_{s,P}^{n,h})_{i,j} \\ &= \int_P H^n \Pi_{P,k-1}^0 \nabla \varphi_i \cdot \Pi_{P,k-1}^0 \nabla \varphi_j \, d\mathbf{x} + \bar{H}_h^n \mathcal{S}_P ((I - \Pi_{P,k}^\nabla) \varphi_i, (I - \Pi_{P,k}^\nabla) \varphi_j), \end{aligned} \tag{119}$$

where the first integral represents the consistency term  $\mathbf{K}_{c,P}^{n,h}$  and the second term is the stability term  $\mathbf{K}_{s,P}^{n,h}$ . Moreover,  $\bar{H}_h^n$  is the mean value of  $H_h^n$  over  $P$  and  $\mathcal{S}_P$  is a suitable stabilizing bilinear form. After a few algebraic passages, it can be shown that in (119) the consistency term can be computed from the degrees of freedom in the following manner:

$$\mathbf{K}_{c,P}^{n,h} = (\hat{\Pi}_{k-1}^{0,x})^T \mathbf{H} \hat{\Pi}_{k-1}^{0,x} + (\hat{\Pi}_{k-1}^{0,y})^T \mathbf{H} \hat{\Pi}_{k-1}^{0,y}, \tag{120}$$

where matrix  $\mathbf{H}^H$  is defined as:

$$(\mathbf{H}^H)_{\alpha,\beta} := \int_P H^n m_\alpha m_\beta d\mathbf{x}, \quad 1 \leq \alpha, \beta \leq n_{k-1}. \quad (121)$$

In the stability term  $\mathbf{K}_{s,P}^{n,h}$ , we choose to adopt the dof<sub>i</sub> – dof<sub>i</sub> stabilization for the bilinear form  $\mathcal{S}_P$ . Therefore,  $\mathbf{K}_{s,P}^{n,h}$  matrix is eventually determined as

$$\mathbf{K}_{s,P}^{n,h} = \bar{H}(\mathbf{I} - \mathbf{\Pi}_k^\nabla)^T(\mathbf{I} - \mathbf{\Pi}_k^\nabla). \quad (122)$$

The implicit terms are then computed by solving the following virtual element linear system:

$$\mathbf{M}_P^h \hat{\boldsymbol{\eta}}_P^{n+1} - \Delta t \mathbf{K}_P^{n,h} \hat{\boldsymbol{\eta}}_P^{n+1} = \mathbf{F}_{s,P}^{n,h}. \quad (123)$$

### 3.11. Projection operators between finite volume and virtual element space

Let us define the  $L_2$ –projection operators used to transfer data from one approximation space to the other and viceversa. In particular, at the beginning of the implicit step of the numerical scheme, data are available in the finite volume space as cell averages (51), thus they need to be projected onto the virtual element space to solve the elliptic equation, either (59) or (65). To that aim, we first perform a CWENO reconstruction step [29,73] to obtain a high order representation of the numerical solution given by (52), that is in terms of the Taylor modal basis functions (53). This is then mapped onto the VEM space by the  $N_P^{\text{dof}} \times n_k$  operator  $\mathbf{V}_P$ , which is defined for each element as

$$\mathbf{V}_P = (\mathbf{M}_P^h)^{-1} \int_P \Pi_*^0 \varphi_i m_\beta d\mathbf{x}, \quad i = 1, \dots, N_P^{\text{dof}}, \quad \alpha = 1, \dots, n_k, \quad (124)$$

where the VEM mass matrix is computed according to (103). Analogously, the inverse  $n_k \times N_P^{\text{dof}}$  operator  $\mathbf{C}_P$ , which transfers data from the virtual element to the finite volume space at the end of the implicit step, writes

$$\mathbf{C}_P = \left( \int_P m_\alpha m_\beta d\mathbf{x} \right)^{-1} \int_P m_\alpha \Pi_*^0 \varphi_i d\mathbf{x}, \quad i = 1, \dots, N_P^{\text{dof}}, \quad \alpha = 1, \dots, n_k. \quad (125)$$

It is clear that these operators verify the consistency relations

$$\mathbf{C}_P \mathbf{V}_P = \mathbf{I}_{[n_k \times n_k]}, \quad \mathbf{V}_P \mathbf{C}_P = \mathbf{I}_{[N_P^{\text{dof}} \times N_P^{\text{dof}}]}. \quad (126)$$

## 4. Numerical results

The novel numerical methods are applied to several test cases in order to thoroughly assess their convergence, stability and accuracy properties. Whenever possible, a comparison against the reference solution is proposed. The quantitative analysis is carried out by measuring the errors between the numerical and the exact solution, denoted by  $q_h$  and  $q_{ex}$ , respectively, for a generic variable  $q(\mathbf{x}) : \Omega \rightarrow \mathbb{R}^2$ . The  $L_2$  and  $L_\infty$  norms are defined as

$$L_2(q) = \sqrt{\int_\Omega (q_h - q_{ex})^2 d\mathbf{x}} \quad \text{and} \quad L_\infty(q) = \max_\Omega |q_h - q_{ex}|.$$

The novel hybrid semi-implicit finite volume/virtual element methods are labelled with SI-FVVEM. The CFL number is assumed to be  $\text{CFL} = 0.9$  in (14). We remark that the time step is independent from the fast scales of the problem under consideration, which are discretized implicitly. We also highlight that if the velocity field is initialized with zero, the first time step is determined according to the CFL condition of fully explicit schemes, hence including the eigenvalues of the full system of governing equations. The first six applications refer to the SWE, while the remaining seven test problems deal with the INS equations. Different Froude numbers and viscosity coefficients are considered for shallow water and viscous flows, respectively, to numerically show that they do not affect neither the accuracy nor the efficiency of the proposed SI-FVVEM schemes. If not stated otherwise, the default configuration is the third order SI-FVVEM scheme in space and time.

**Table 1**

Numerical convergence results of the SI-FVVEM scheme with second and third order of accuracy in space and time using the steady shallow water vortex problem on Voronoi meshes. The errors are measured in  $L_2$  norm and refer to the free surface elevation  $\eta$  and velocity component  $u$  at time  $t_f = 0.1$ . The asymptotic preserving (AP) property of the schemes is studied by considering different Froude numbers  $Fr = \{10^{-1}, 10^{-2}, 10^{-4}, 10^{-6}\}$ .

$h(\Omega)$	SI-FVVEM $\mathcal{O}(2)$				SI-FVVEM $\mathcal{O}(3)$			
	$L_2(\eta)$	$\mathcal{O}(\eta)$	$L_2(u)$	$\mathcal{O}(u)$	$L_2(\eta)$	$\mathcal{O}(\eta)$	$L_2(u)$	$\mathcal{O}(u)$
Fr = $10^{-1}$ (double precision)								
5.0184E-01	6.5520E-04	–	4.3967E-02	–	5.5807E-04	–	4.1527E-02	–
2.4761E-01	1.2273E-04	2.37	9.5032E-03	2.17	8.0381E-05	2.74	5.9083E-03	2.76
1.6167E-01	5.3189E-05	1.96	4.3203E-03	1.85	2.7156E-05	2.55	1.9278E-03	2.63
1.2464E-01	2.9092E-05	2.32	2.4833E-03	2.13	1.0751E-05	3.56	8.4473E-04	3.17
Fr = $10^{-2}$ (quadruple precision)								
5.0184E-01	6.7318E-06	–	4.4214E-02	–	5.7630E-06	–	4.1836E-02	–
2.4761E-01	1.6369E-06	2.00	9.6380E-03	2.16	8.0877E-07	2.78	5.9892E-03	2.77
1.6167E-01	7.4398E-07	1.85	4.5656E-03	1.75	2.4626E-07	2.79	1.8936E-03	2.70
1.2464E-01	4.0295E-07	2.36	2.5566E-03	2.23	9.7144E-08	3.58	8.5621E-04	3.05
Fr = $10^{-4}$ (quadruple precision)								
5.0184E-01	6.7348E-10	–	4.4216E-02	–	5.7672E-10	–	4.1839E-02	–
2.4761E-01	1.7694E-10	1.89	9.6401E-03	2.16	8.1477E-11	2.77	5.9903E-03	2.75
1.6167E-01	7.7779E-11	1.93	4.5693E-03	1.75	2.4780E-11	2.79	1.8942E-03	2.70
1.2464E-01	4.0266E-11	2.53	2.5578E-03	2.23	9.7662E-12	3.58	8.5635E-04	3.05
Fr = $10^{-6}$ (quadruple precision)								
5.0184E-01	6.7386E-14	–	4.4281E-02	–	5.7694E-14	–	4.1908E-02	–
2.4761E-01	1.7780E-14	1.89	9.7891E-03	2.14	8.5738E-15	2.78	7.0563E-03	2.52
1.6167E-01	7.7780E-15	1.94	4.5707E-03	1.79	2.5189E-15	2.79	2.3166E-03	2.61
1.2464E-01	4.0491E-15	2.51	2.5657E-03	2.22	9.9750E-16	3.58	1.0207E-04	3.15

4.1. Convergence rates study (SWE)

This subsection examines the numerical convergence of the SI-FVVEM schemes on the steady shallow water vortex initially proposed in [19]. The computational domain is the square  $\Omega = [-5; 5]^2$  with a flat bottom ( $b = 0$ ), considering periodic boundary conditions. For this test case, the exact solution in polar coordinates is defined by the following free surface elevation and velocity field

$$\eta(\mathbf{x}) = H_0 - \frac{1}{2g} e^{-(r^2-1)}, \quad \mathbf{v}(\mathbf{x}) = \begin{bmatrix} u(\mathbf{x}) \\ v(\mathbf{x}) \end{bmatrix} = \begin{bmatrix} -u_\alpha \sin(\alpha) \\ u_\alpha \cos(\alpha) \end{bmatrix}, \tag{127}$$

where the radius  $r$  and the polar angle  $\alpha$  are given by the inverse polar map

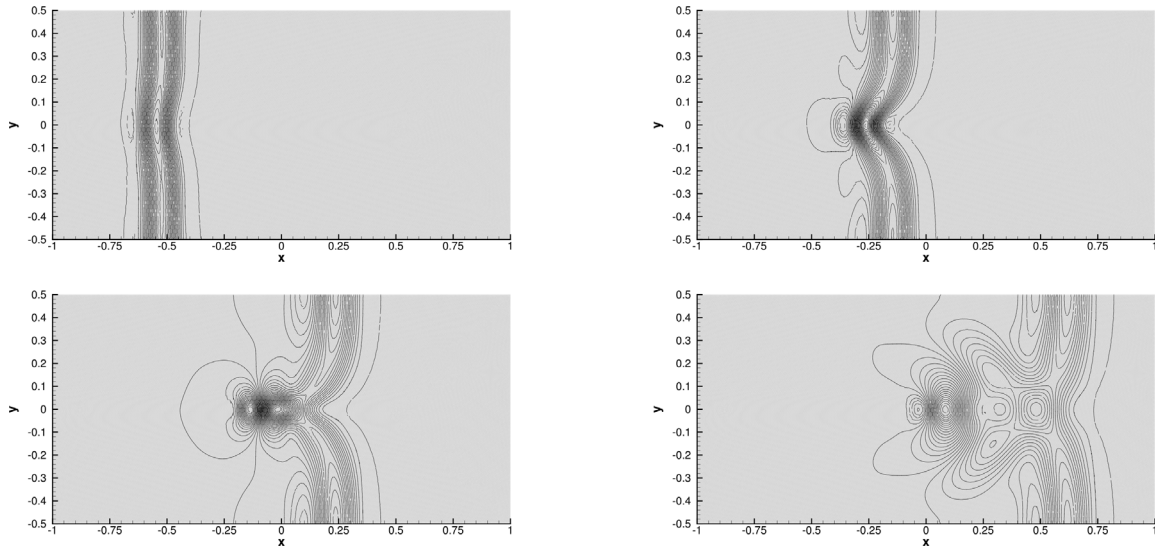
$$r = \sqrt{x^2 + y^2} \quad \text{and} \quad \alpha = \text{atan2}(y, x).$$

In (127), the water depth for infinite radius is identified by  $H_0$  and the angular velocity  $u_\alpha = r e^{-(r^2-1)/2}$  allows the centrifugal forces ( $u_\alpha/gr$ ) to balance the pressure forces ( $\partial\eta/\partial r$ ). In this simulation, the initial conditions are given by the exact solution in (127). The Froude number can be varied by adjusting the value of the constant  $H_0$ . In particular, a gravitational constant  $g = 10$  is used in order to ease this computation. For polynomial space dimension  $M = 2, 3$  the convergence is performed on four refined computational meshes, each with four different Froude numbers  $Fr \in \{0.32, 10^{-2}, 10^{-4}, 10^{-6}\}$  (corresponding to an asymptotic free surface  $H_0 \in \{10^0, 10^3, 10^7, 10^{11}\}$ , respectively). Each mesh is identified by the characteristic length  $h(\Omega)$ , taken as the maximum element size in the computational grid. The analysis takes into account the  $L_2$ -norm for the free surface  $\eta$  and the velocity  $u$  in  $x$ -direction at final time  $t_f = 0.1$ , with quadruple precision finite arithmetic [28]. Table 1 sums up the results and reports the convergence rates. We can see that the formal order of accuracy is correctly achieved, and that the scheme is both asymptotic preserving and accurate, with the order of accuracy being independent of the Froude number.

**Table 2**

Well-balance test with double and quadruple finite arithmetic precision. Errors measured in  $L_2$  and  $L_\infty$  norms for the free surface elevation  $\eta$  and momentum component  $Hu$  at the final time  $t_f = 0.1$ .

Precision	$L_2(\eta)$	$L_\infty(\eta)$	$L_2(Hu)$	$L_\infty(Hu)$
Double	1.5221E-15	3.5527E-15	2.010E-18	9.7600E-17
Quadruple	9.3593E-34	4.5259E-33	4.833E-22	1.3078E-20



**Fig. 2.** Well-balance test with small perturbation of the free surface ( $\delta = 10^{-2}$ ). 50 equidistant contour lines in the interval  $\eta = [0.995; 1.008]$  are shown at output times  $t = 0.12, t = 0.24, t = 0.36$  and  $t = 0.48$  (from top left to bottom right panel).

#### 4.2. Well-balance test (SWE)

To test the well-balance property of the scheme (also known as the C-property) proven in [Theorem 2](#), we use a benchmark proposed in [\[68\]](#). This test case assesses whether a numerical scheme can maintain stationary equilibrium solutions of the governing equations up to machine precision. The particular water-at-rest equilibrium solution of the SWE is characterized by a constant free surface elevation  $\eta(\mathbf{x}, t) = \eta_0$  and zero fluid velocity, i.e.,  $\mathbf{v}(\mathbf{x}, t) = \mathbf{0}$ , with an arbitrary bottom topography different from the trivial profile  $b(\mathbf{x}) = 0$ . Following [\[68\]](#), we use a computational domain  $\Omega = [-2; 1] \times [-0.5; 0.5]$  with Dirichlet boundary conditions in the  $x$ -direction and periodic boundaries along the  $y$ -direction. The domain is paved with  $N_p = 8633$  Voronoi cells of mesh size  $h = 1/50$ . The bathymetry and the initial free surface elevation are defined as follows:

$$b(\mathbf{x}) = 0.5 \cdot e^{-5(x+0.1)^2 - 50y^2}, \quad \eta(\mathbf{x}, 0) = \begin{cases} 1 + \delta & \text{if } -0.95 \leq x \leq -0.85 \\ 1 & \text{elsewhere} \end{cases}. \tag{128}$$

Here, the fluid is initially at rest, and we first set the perturbation amplitude  $\delta = 0$ , thus  $\eta_0 = 1$ . The simulation is run until the final time  $t_f = 0.1$ , using double and quadruple finite arithmetic. The errors with respect to the initial condition are reported in [Table 2](#). The results show that the scheme is well-balanced up to machine accuracy.

In accordance with [\[68\]](#), a slight perturbation is introduced in the free surface elevation by setting  $\delta = 10^{-2}$  in [\(128\)](#). To properly track the wave propagation, a constant time step of  $\Delta t = 0.01$  is employed. [Fig. 2](#) displays the results at various output times, demonstrating that the presence of the bottom bump does not generate any spurious oscillations. Moreover, the flow structure agrees well with the outcomes found in the literature [\[22,28,76\]](#).

**Table 3**

Initialization of Riemann problems. Initial left (L) and right (R) states are reported as well as the final time of the simulation  $t_f$ , the computational domain  $[x_L; x_R]$ , the position of the initial discontinuity  $x_d$  and the characteristic mesh size  $h$ .

Test	$\eta_L$	$u_L$	$b_L$	$\eta_R$	$u_R$	$b_R$	$x_L$	$x_R$	$x_d$	$h$	$t_f$
RP1 [78]	1	0	0	2	0	0	-0.5	0.5	0	1/200	0.075
RP2	$10^3$	0	0	1	0	0	-15	15	0	1/400	0.09
RP3 [79]	1	0	0.2	0.5	0	0	-5	5	0	1/200	1
RP4 [79]	1.46184	0	0	0.30873	0	0.2	-5.0	5.0	0	1/200	1

### 4.3. Circular dambreak (SWE)

To simulate shock waves, we will use the circular dam break problem over a bottom step presented in [7,22]. The computational domain is a circle with radius  $r = |\mathbf{x}| \leq 2$ , denoted by  $\Omega = \{\mathbf{x} \in \mathbb{R}^2 : r = |\mathbf{x}| \leq 2\}$ , with Dirichlet boundary conditions everywhere. The computational grid has a total number of  $N_p = 34477$  cells, with a characteristic mesh size  $h = 1/50$ . The initial condition for the problem reads

$$\eta(\mathbf{x}, 0) = \begin{cases} 1.0 & \text{if } r \leq 1 \\ 0.5 & \text{if } r > 1 \end{cases}, \quad b(\mathbf{x}) = \begin{cases} 0.2 & \text{if } r \leq 1 \\ 0.0 & \text{if } r > 1 \end{cases}, \quad \mathbf{v}(\mathbf{x}, 0) = \mathbf{0}. \tag{129}$$

The simulation is carried out until the final time  $t_f = 0.2$ , at which the solution exhibits a contact wave travelling towards the centre of the domain, as well as a shock wave heading towards the outer boundary. Additionally, due to the presence of the bottom step, there is a discontinuity in the flow at  $r = 1$ . Fig. 3 depicts the results of the simulation along with a comparison against a reference solution. The reference solution has been computed by solving the one-dimensional (1D) SWE in the radial direction with geometric reaction source terms, using a classical shock capturing MUSCL-TVD finite volume scheme with 10 000 cells [77]. The figure shows that there is a very good agreement between the numerical and reference solutions, with no spurious oscillations occurring in the plateau between the different discontinuities. Note that numerical dissipation is only present in the finite volume solver for the convective terms and not in the pressure solver [29], as required in [28].

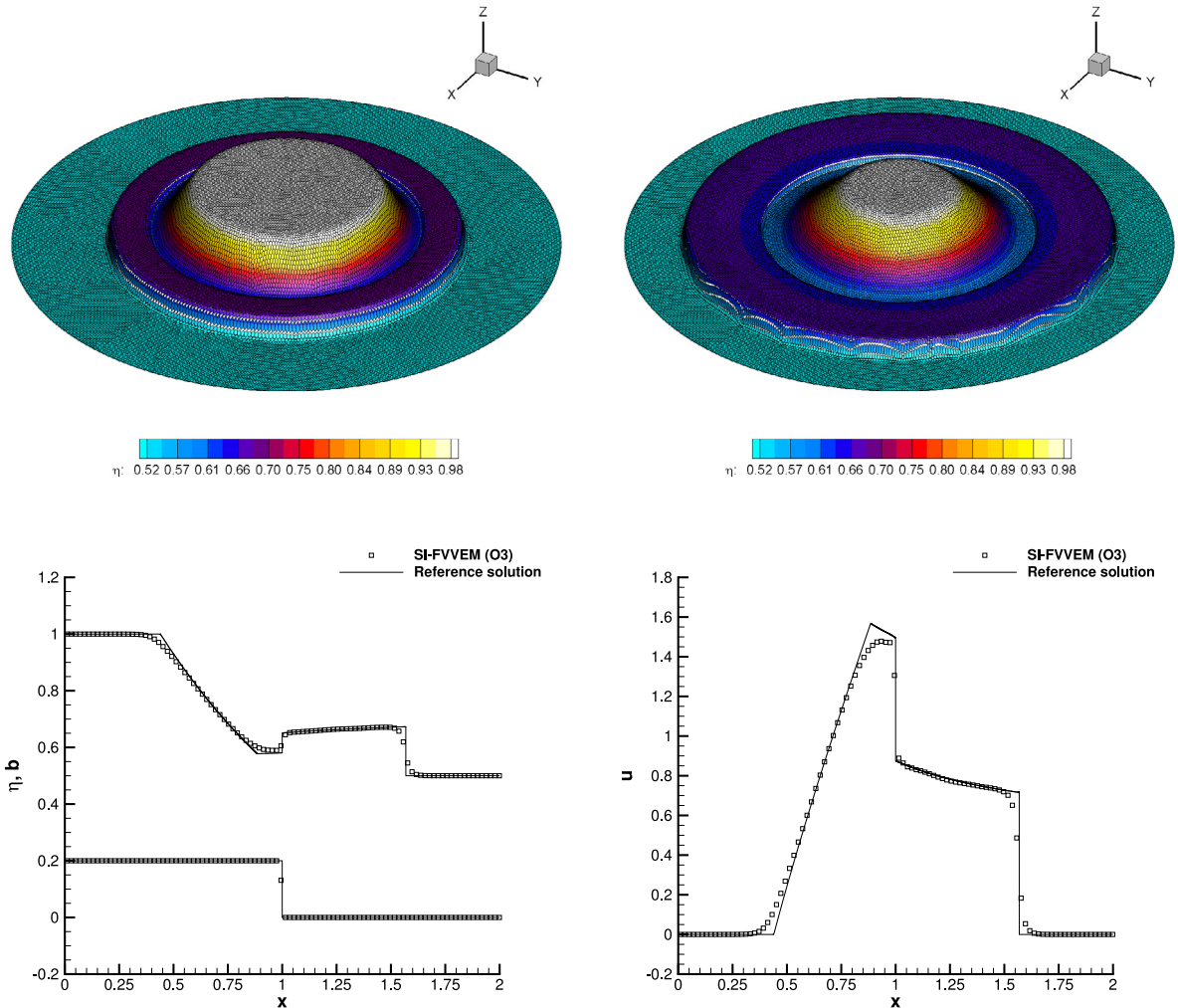
### 4.4. Riemann problems (SWE)

To verify shock-capturing and conservation properties, a set of Riemann problems is considered, which include both flat and variable bottom topography. The exact solution is obtained using the Riemann solver proposed in [78,79] for flat and variable bottom, respectively. The initial condition is given in terms of two states  $\mathbf{Q}_L = (\eta_L, u_L, b_L)$  and  $\mathbf{Q}_R = (\eta_R, u_R, b_R)$  separated by a discontinuity located at position  $x = x_d$ :

$$\mathbf{Q}(\mathbf{x}, 0) = \begin{cases} \mathbf{Q}_L & \text{if } x \leq x_d \\ \mathbf{Q}_R & \text{if } x > x_d \end{cases}. \tag{130}$$

Data of the computational domain and the initial condition for the chosen four Riemann problems are summarized in Table 3. The computational domain in these test cases is  $\Omega = [x_L, x_R] \times [x_L/10, x_R/10]$ , and it is discretized using an unstructured Voronoi mesh of size  $h$ . This makes the computation multidimensional, despite the 1D setup of the test problems.

Fig. 4 shows the comparison between the numerical and the reference solution through a 1D cut of 200 equidistant points along the  $x$ -axis of the computational domain at  $y = 0$ . The first two Riemann problems (RP1 and RP2) assume a constant flat bathymetry and involve shock and rarefaction waves. The remaining Riemann problems (RP3 and RP4) are concerned with a jump in the bottom elevation of height  $\Delta b = 0.2$ , which generates contact discontinuities. The SI-FVVEM scheme exhibits an excellent agreement and can handle supercritical flows with Froude numbers greater than one. For RP2, the maximum Froude number is  $Fr = 5.73$ . The FV strategy used to discretize the non-linear convective terms makes the scheme robust. The implicit treatment of the free surface elevation still ensures a stable scheme for all four Riemann problems, without any need of additional numerical dissipation. We also remark that the proposed approach is conservative by construction, correctly capturing moving shocks and the values of the plateau between two discontinuities. The 1D symmetry of the problem is preserved even in the context of arbitrary shaped polygonal cells, as depicted in the 3D views of the free surface elevation in Fig. 4.



**Fig. 3.** Circular dam break problem. Top: 3D view of the free surface elevation at time  $t = 0.1$  (left) and  $t = 0.2$  (right). Bottom: 1D cut along the line  $y = 0$  of the numerical solution compared with the reference solution at time  $t_f = 0.2$  for the free surface and bottom profile (left) as well as for the horizontal velocity component (right).

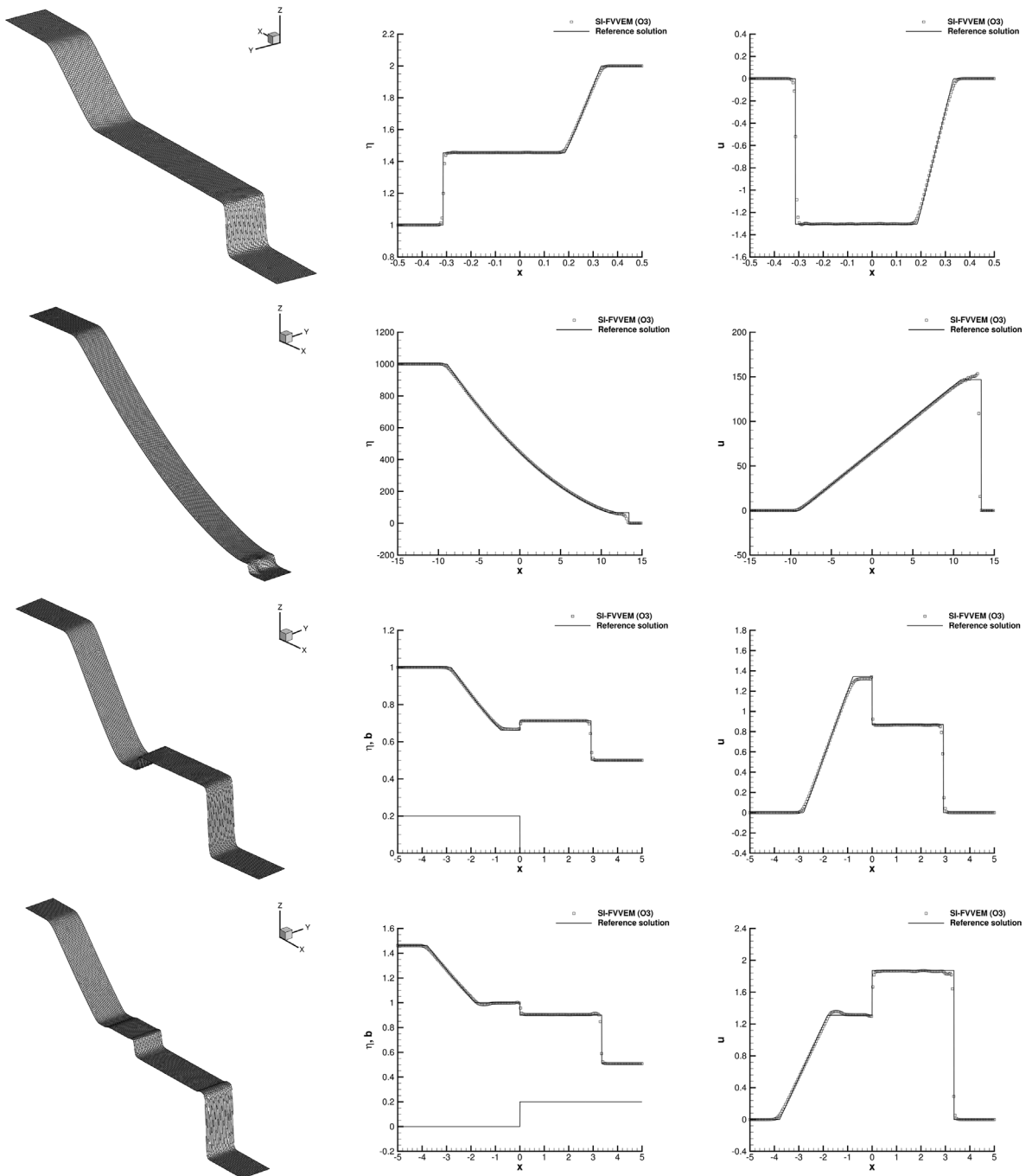
#### 4.5. Smooth surface wave propagation (SWE)

In this study, the propagation of a wave on the free surface is examined, following the configuration outlined in [7]. The computational domain is the square  $\Omega = [-1; 1]^2$ , with Dirichlet boundary conditions enforced on all sides. The domain is discretized into  $N_p = 15717$  Voronoi cells, with a characteristic mesh size  $h = 1/50$ . The initial condition is defined by

$$\eta(\mathbf{x}, 0) = 1 + e^{-\frac{r^2}{2\sigma^2}}, \quad \mathbf{v}(\mathbf{x}, 0) = \mathbf{0}, \quad b(\mathbf{x}) = 0, \quad (131)$$

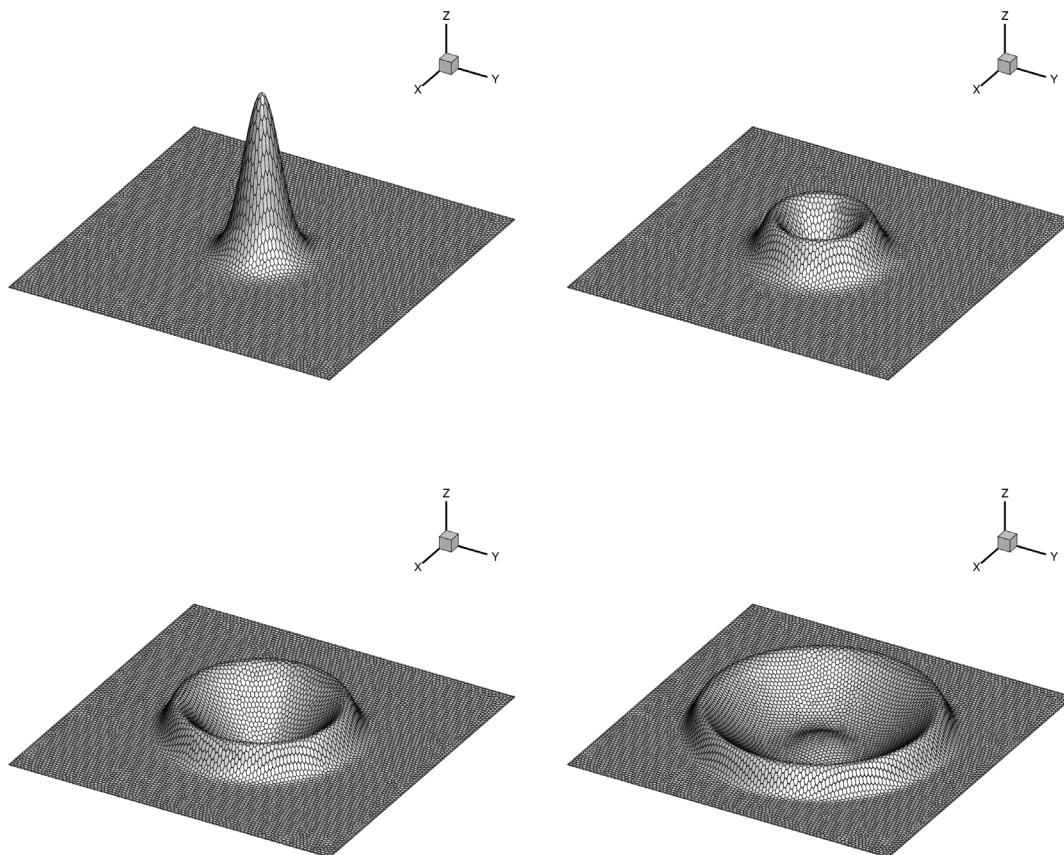
where  $\sigma = 0.1$ . The time step is set to  $\Delta t = 0.001$ , and the simulation is run until the time  $t_f = 0.15$ , at which the wave profile becomes stiff and a shock wave appears. Fig. 5 shows a 3D view of the free surface elevation at different output times, illustrating the ability of the SI-FVEM scheme to well preserve the symmetry of the solution even when the computational grid is not symmetric. The reference solution is computed using a 1D MUSCL-TVD scheme on a very fine mesh, as done for the circular dam break problem. In Fig. 6 we plot a comparison between the numerical solution and the reference solution for the free surface elevation and horizontal velocity component.



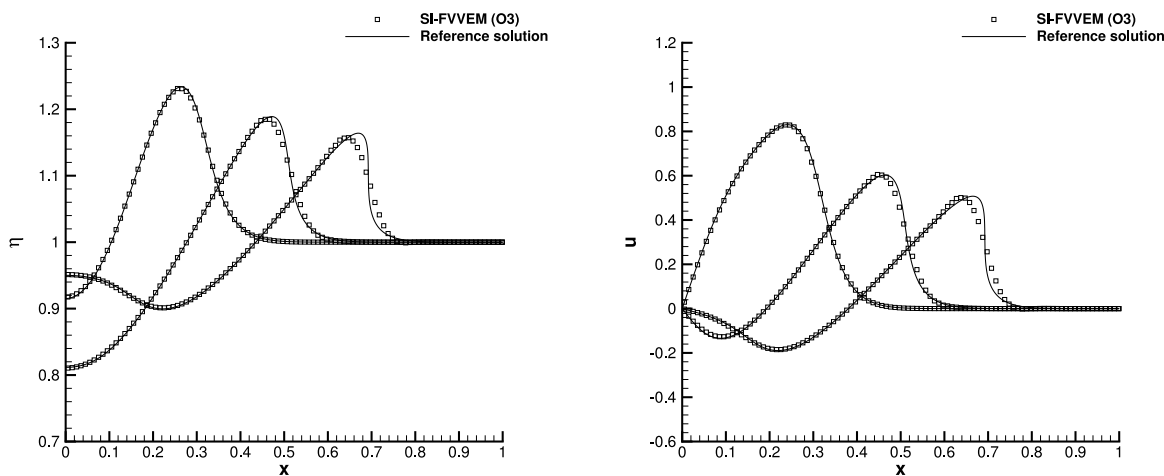


**Fig. 4.** Riemann problems RP1, RP2, RP3 and RP4 (from top to bottom row). 3D view of the free surface elevation with the Voronoi computational mesh (left) and comparison against the reference solution for the variables  $\eta$  (middle) and  $u$  (right).

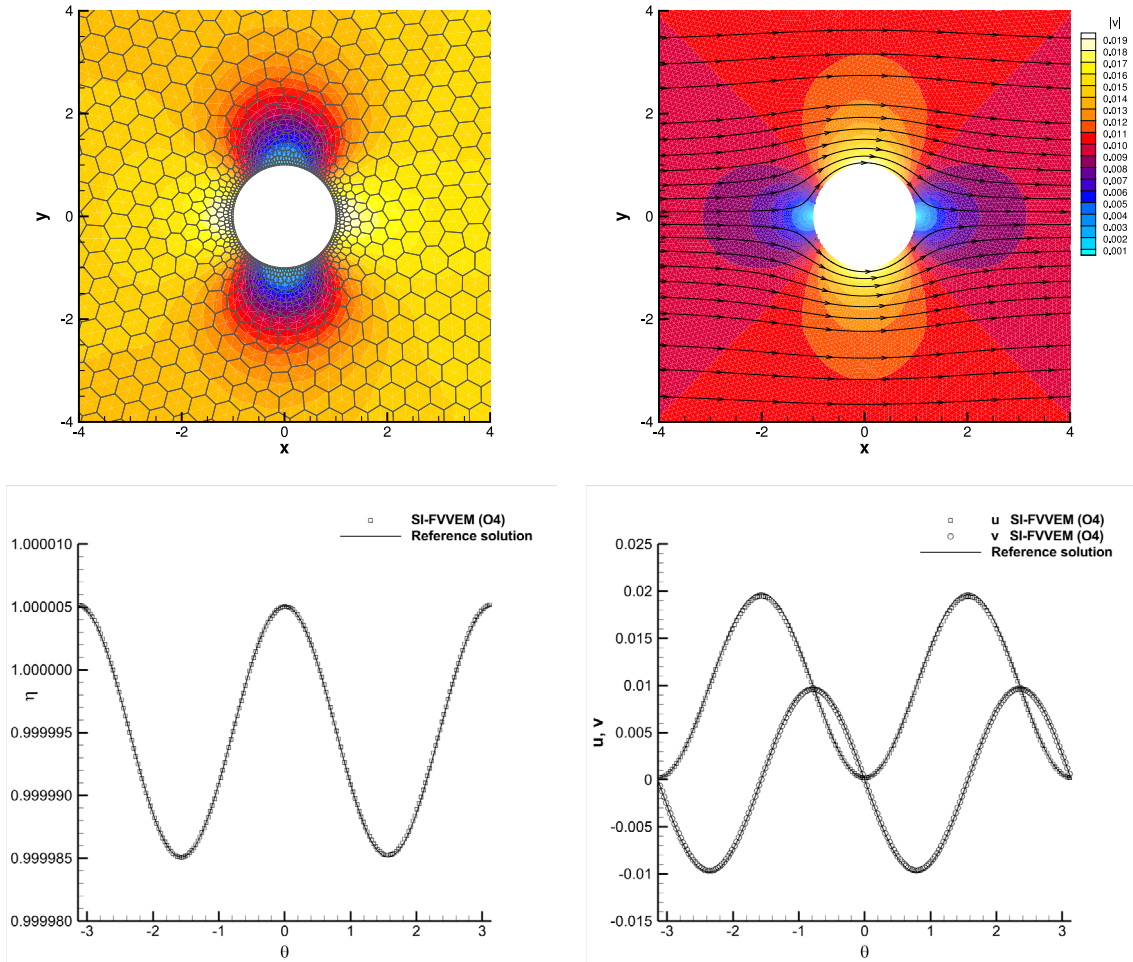
A very good agreement can be appreciated, particularly up to time  $t = 0.1$ , when the flow is still smooth. At time  $t = 0.15$ , the shock is smoothed by the explicit finite volume convective solver, and the SI-FVVEM scheme remains stable, without any spurious oscillations.



**Fig. 5.** Smooth surface wave propagation problem. 3D view of the free surface elevation and Voronoi computational mesh at output times  $t = 0$ ,  $t = 0.05$ ,  $t = 0.1$  and  $t = 0.15$  (from top left to bottom right panel).



**Fig. 6.** Smooth surface wave propagation problem. Comparison between numerical and reference solution at output times  $t = 0.05$ ,  $t = 0.1$  and  $t = 0.15$  for the free surface elevation  $\eta$  (left) and the horizontal velocity component  $u$  (right).



**Fig. 7.** Flow around a circular cylinder at time  $t_f = 10$ . Top: zoom around the cylinder of the contour plot of the free surface elevation with the Voronoi computational mesh (left) and of the magnitude of the velocity field with stream-traces (right). Bottom: comparison of the numerical and exact solutions for the free surface (left) and velocity components (right) at radius  $r = 1.01$ .

4.6. Low Froude number flow around a circular cylinder (SWE)

As last test case for the SWE, we propose to simulate a low Froude ( $Fr = 3.19 \cdot 10^{-3}$ ) flow that passes around a circular cylinder of radius  $r_c = 1$ , as studied in [28,80]. The computational domain is  $\Omega = [-16; 16]^2$ , where the circular cylinder is defined as  $\mathbf{x} \in \mathbb{R}^2$  such that  $r \leq r_c$ , where  $r = \sqrt{x^2 + y^2}$  is the radial coordinate, and the bottom is assumed to be flat ( $b = 0$ ). The mesh consists of Voronoi cells with a characteristic mesh size  $h = 1/20$  near the cylinder, increasing regularly in diameter until  $h = 1/2$  at the domain boundaries (see Fig. 7). This choice results necessary in order to properly approximate the internal boundary without relying on an isoparametric description of the physical geometry, as proposed in [22]. The mesh contains a total of  $N_p = 15516$  cells.

The exact velocity field and free surface elevation of the test problem can be expressed using polar coordinates  $(r, \theta)$  as follows:

$$v_r = v_m \left( 1 - \frac{r_c^2}{r^2} \right) \cos(\theta), \quad v_\theta = -v_m \left( 1 + \frac{r_c^2}{r^2} \right) \sin(\theta), \quad \eta = \eta_0 + \frac{1}{2} v_m^2 g \left( \frac{2r_c^2}{r^2} \cos(2\theta) - \frac{r_c^4}{r^2} \right), \quad (132)$$

where  $\eta_0 = 1$  and  $v_m = 10^{-2}$ . The initial condition for the simulation is a flat free surface,  $\eta(\mathbf{x}, 0) = \eta_0$ , while the exact solution is imposed at all boundaries except for the rightmost side of the domain ( $x = 16$ ), where an outflow

**Table 4**

Errors in  $L_2$  and  $L_\infty$  norms for the horizontal velocity  $u$  for the stationary Poiseuille flow. The errors refer to the polynomial space dimension  $M$  at final time  $t_f = 5$ .

$M$	$L_2(u)$	$L_\infty(u)$
1	3.0966E-02	8.5265E-2
2	4.1826E-13	1.6733E-13
3	4.2756E-13	3.0695E-13

condition is set. The simulation is run until the final time  $t_f = 10$  to reach the stationary state. In order to exploit the high order of accuracy of the SI-FVVEM methods, we use the fourth order version in space, with a first order time discretization. The velocity field with associated streamlines at the final time are shown in Fig. 7 (top). The comparison against the reference solution is presented in Fig. 7 (bottom) along the circumference of radius  $r = 1.01$  centred at the origin, where we observe that the fourth order in space SI-FVVEM method properly retrieves the exact profile of both the free surface elevation and velocity components.

#### 4.7. Stationary Poiseuille flow (INS)

This first test case for the INS model concerns an incompressible viscous flow between two parallel plates. It is a particular solution of the Stokes system, i.e., the convective fluxes are neglected. The computational domain is the channel  $\Omega = [x_L; x_R] \times [y_B; y_T] = [0; 3] \times [0; 1]$  discretized with  $N_p = 1949$  cells, and the stationary solution writes

$$\frac{\partial p}{\partial x} = \frac{p_R - p_L}{x_R - x_L}, \quad \mathbf{v} = \begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} \frac{(p_R - p_L)}{2L\nu} y(y - y_T) \\ 0 \end{pmatrix}, \quad \nu = 10^{-2}, \tag{133}$$

where  $p_L = -1$  and  $p_R = -5.8$  are the constant pressures applied along  $x_L$  and  $x_R$ , respectively. Wall boundary conditions are imposed in  $y$ -direction, while the Dirichlet boundaries are set along the  $x$ -direction. The initial condition is given by the exact solution in (133). Table 4 collects the errors in  $L_2$  and  $L_\infty$ -norms between the recovered and exact horizontal velocity  $u$  with respect to the polynomial space dimension  $M$  at final time  $t_f = 5$ . As expected, when  $M \geq 2$ , i.e. the polynomial space contains parabolic functions, the numerical errors are up to machine accuracy.

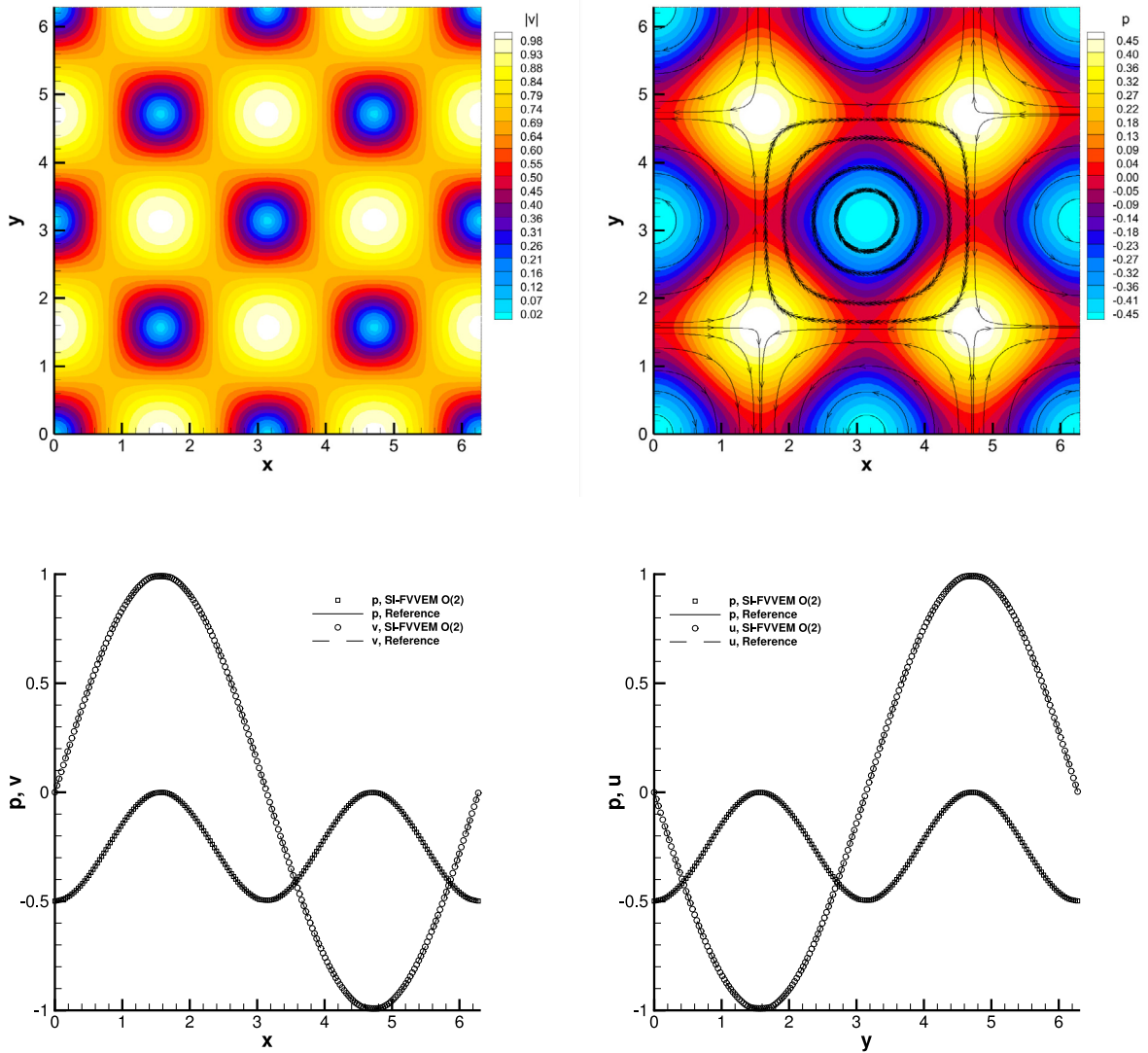
#### 4.8. 2D Taylor–Green vortex (INS)

The Taylor–Green vortex describes the flow of an incompressible viscous fluid in a 2D square domain  $\Omega = [0; 2\pi]^2$  with periodic boundary conditions. The exact solution for this problem is

$$p = -\frac{e^{-4\nu t}}{4} (\cos(2x) + \cos(2y)), \quad \mathbf{v} = \begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} \sin(x) \cos(y) \\ -\cos(x) \sin(y) \end{pmatrix} e^{-2\nu t}, \quad \nu = 10^{-2}. \tag{134}$$

The initial condition is defined by the exact solution (134) at time  $t = 0$ . A second order SI-FVVEM method is run on a mesh composed of  $N_p = 5705$  polygonal cells with characteristic size  $h = 1/20$ . Fig. 8 depicts the numerical solution obtained at the final time of the simulation,  $t_f = 0.2$ . The velocity magnitude and the pressure distribution with the stream-traces are shown and a comparison of 1D cuts along the  $x$ - and the  $y$ -axis of the numerical solution against the exact one is proposed for both pressure and velocity, retrieving an excellent matching.

This test case is also used to assess the convergence property of the proposed SI-FVVEM scheme. We consider a sequence of successively refined meshes, and we compute the errors in  $L_2$  norm with respect to the analytical solution (134). The convergence rates are reported in Table 5 for different values of the Reynolds number, in order to evaluate the asymptotic behaviour of the scheme. As demonstrated by Theorem 3, the schemes satisfy the AP property and the errors are independent of the stiffness parameter  $Re$ , achieving the formal second order of accuracy in space and time.



**Fig. 8.** Taylor–Green vortex at time  $t = 0.2$  with viscosity  $\nu = 10^{-2}$ . Top: velocity magnitude (left) and pressure with the stream-traces (right). Bottom: 1D cuts with 200 equidistant points along the  $x$ -axis (left) and the  $y$ -axis (right) of the numerical solution against the exact solution.

4.9. First problem of Stokes (INS)

In this test case, we consider a problem only dominated by viscous effects, known as the first problem of Stokes, for which an exact solution of the unsteady Navier–Stokes equations has been derived [69]. An infinite incompressible shear layer is modelled starting from the following velocity field:

$$u = 0, \quad v = v_0 \operatorname{erf}\left(\frac{1}{2} \frac{x}{\sqrt{\nu t}}\right), \quad v_0 = 0.1. \tag{135}$$

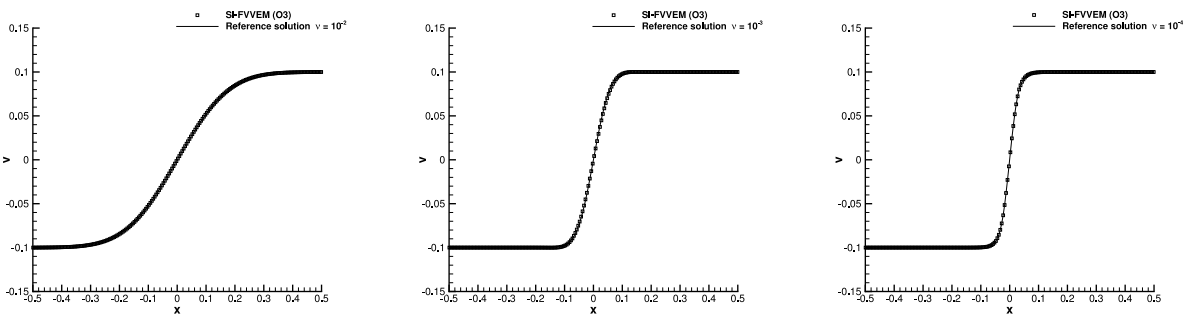
The initial condition then writes

$$p = 1, \quad u = 0, \quad v = \begin{cases} -v_0, & x \leq 0 \\ v_0, & x > 0. \end{cases} \tag{136}$$

**Table 5**

Numerical convergence results of the SI-FVEM scheme with second order of accuracy in space and time using the Taylor–Green vortex on general polygonal meshes. The errors are measured in  $L_2$  norm and refer to the pressure  $p$  and velocity component  $u$  at time  $t_f = 0.2$ . The asymptotic preserving (AP) property of the scheme is studied by considering different Reynolds numbers  $Re = \{10^2, 10^3, 10^4, 10^5\}$ .

$h(\Omega)$	$L_2(p)$	$\mathcal{O}(p)$	$L_2(u)$	$\mathcal{O}(u)$
$Re = 10^2$				
3.1483E-01	1.8321E-02	–	4.6596E-02	–
1.7729E-01	6.2191E-03	1.88	9.2174E-03	2.82
1.2069E-01	1.7224E-03	3.34	3.7950E-03	2.31
8.9945E-02	9.2102E-04	2.13	2.1919E-03	1.87
$Re = 10^3$				
3.1483E-01	1.8501E-02	–	4.6894E-02	–
1.7729E-01	6.0593E-03	1.94	9.3017E-03	2.82
1.2069E-01	1.7371E-03	3.25	3.8958E-03	2.26
8.9945E-02	9.2967E-04	2.13	2.2858E-03	1.81
$Re = 10^4$				
3.1483E-01	1.8520E-02	–	4.6924E-02	–
1.7729E-01	6.0464E-03	1.95	9.3105E-03	2.82
1.2069E-01	1.7387E-03	3.24	3.9069E-03	2.26
8.9945E-02	9.3060E-04	2.13	2.2964E-03	1.81
$Re = 10^5$				
3.1483E-01	1.8522E-02	–	4.6927E-02	–
1.7729E-01	6.0451E-03	1.95	9.3114E-03	2.82
1.2069E-01	1.7389E-03	3.24	3.9080E-03	2.26
8.9945E-02	9.3069E-04	2.13	2.2975E-03	1.81

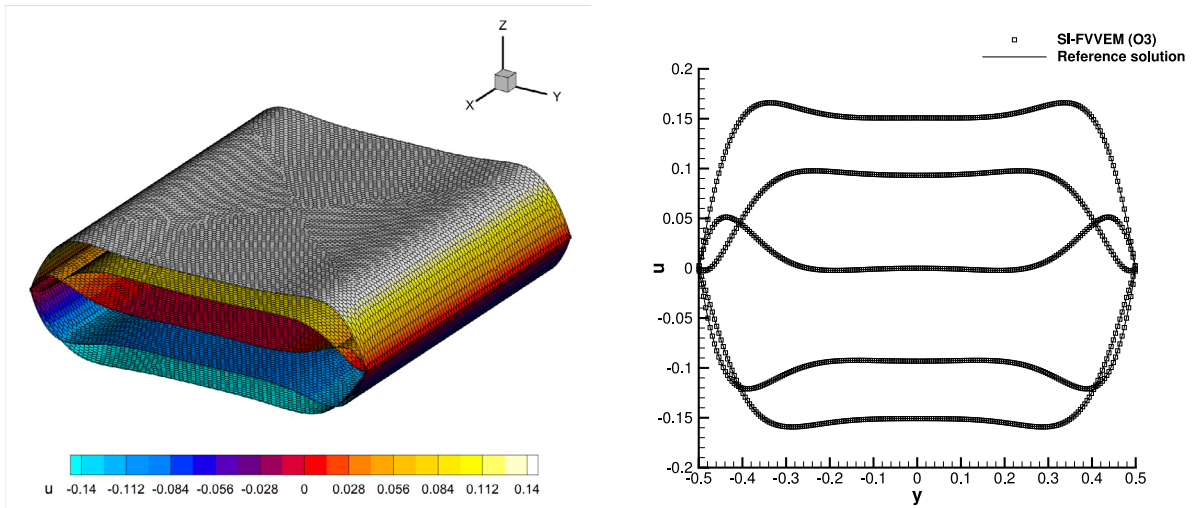


**Fig. 9.** First problem of Stokes at time  $t_f = 1$ . Comparison against the analytical solution with viscosity  $\nu = 10^{-2}$  (left),  $\nu = 10^{-3}$  (middle) and  $\nu = 10^{-4}$  (right). The numerical solution is interpolated with 200 equidistant points at  $y = 0$ .

The computational domain is  $\Omega = [-0.5; 0.5] \times [-0.1; 0.1]$  and it is paved with a grid of characteristic mesh size  $h = 1/100$  along the  $x$ -direction, while  $h = 1/10$  is used along the  $y$ -direction. At the boundaries, the initial condition (136) is imposed along the  $y$ -direction while periodic conditions close the problem on the remaining sides. The first problem of Stokes is solved using three values of viscosity, namely  $\nu = 10^{-3}$ ,  $\nu = 10^{-4}$  and  $\nu = 10^{-5}$ . Fig. 9 depicts the recovered solutions for the vertical velocity  $v$  at  $y = 0$  over the exact solution at the final time  $t_f = 1$ . The figures show a perfect overlapping between numerical and analytical solutions. We remark that the time step is independent of the value of the viscosity coefficient, thanks to the implicit discretization of the viscous terms. Therefore, the three simulations ran for the same amount of computational time.

4.10. Oscillatory viscous flow between two flat plates (INS)

Here, we consider a viscous incompressible flow between two flat plates. The solution solves the Stokes equations and was derived by Womersley in [81]. The computational domain is the square  $\Omega = [x_L; x_R]^2 = [-0.5; 0.5]^2$



**Fig. 10.** Womersley flow with  $\nu = 2 \cdot 10^{-2}$ . Left: 3D view of the horizontal velocity  $u$  over the entire computational domain at different output times. Right: numerical solution interpolated along 200 equidistant spatial points at  $y = 0$  compared with the exact solution at output times  $t = 0.2, t = 0.4, t = 1.0, t = 0.6, t = 0.8$  (from the bottom to the top).

with wall boundaries in the  $y$ -direction and periodic sides elsewhere. The employed mesh has a total number of  $N_p = 14368$  polygons with a characteristic size  $h = 1/80$ . The non-linear convective terms are neglected according to [82], hence obtaining the analytical solution for the fluid velocity and pressure gradient as a function of time and the distance from the plate:

$$u = \frac{A}{i\omega} \left[ 1 - \frac{\cosh(\alpha_W \sqrt{i} y/R)}{\cosh(\alpha_W \sqrt{i})} \right] e^{i\omega t}, \quad \frac{\partial p}{\partial x} = \frac{p(x_R) - p(x_L)}{L} = A e^{i\omega t}. \tag{137}$$

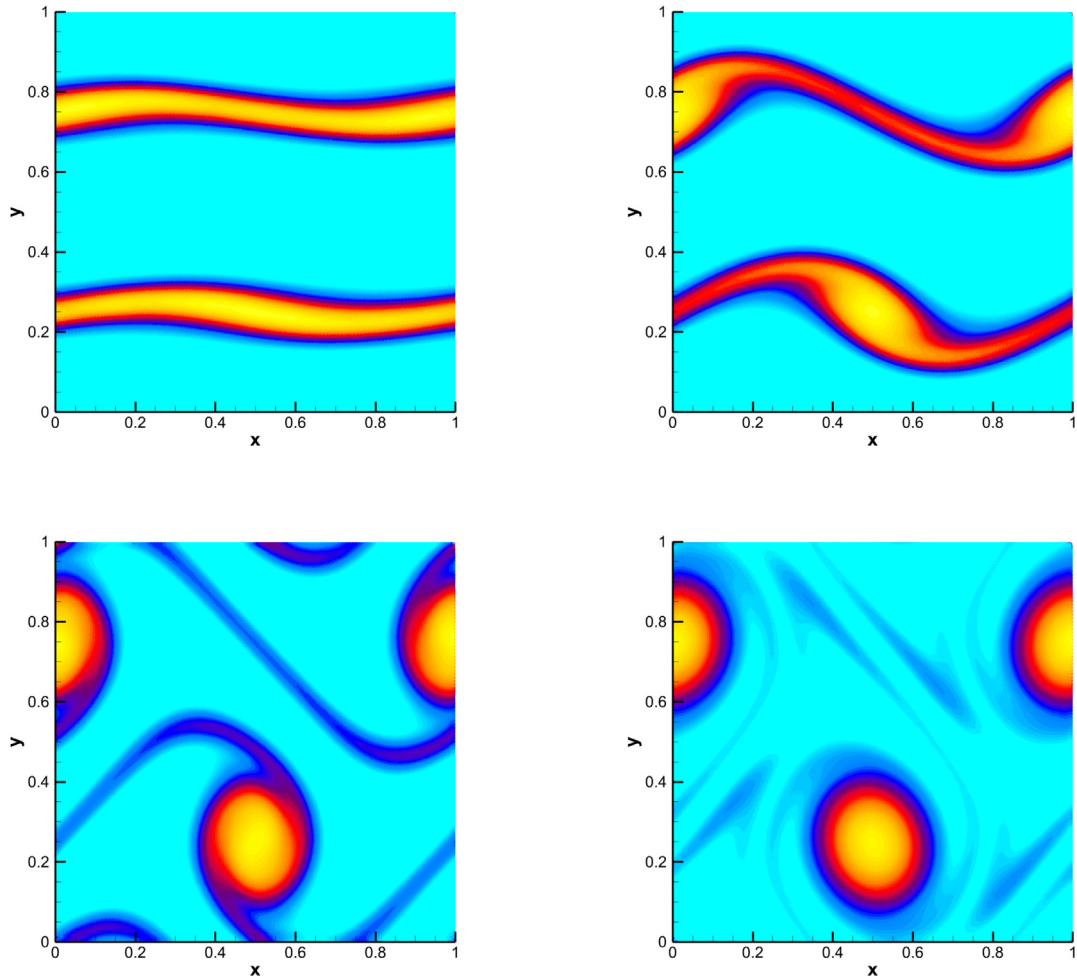
Here,  $\alpha_W = R\sqrt{\omega/\nu}$  is the Womersley number, with  $R$  being the half distance between the two plates and  $\omega = 2\pi$  denoting the frequency of the oscillations. Moreover, the amplitude of the sinusoidal pressure gradient is set to  $A = 1$ , the imaginary unit is addressed as usual with  $i = \sqrt{-1}$ , and  $L = x_R - x_L$  is the total length of the computational domain along the  $x$ -direction. The pressure gradient is imposed at the aid of a source term that is discretized explicitly in time as already done in [53]. The viscosity coefficient is  $\nu = 2 \cdot 10^{-2}$  and the final time of the simulation is  $t_f = 1$ . To properly follow the oscillatory dynamics, a time step  $\Delta t = 0.01$  is imposed. Fig. 10 shows a 3D view of the obtained numerical horizontal velocity  $u$  for time instances  $t \in \{0.2, 0.4, 0.6, 0.8\}$ . We also compare the numerical horizontal velocity profile along the cut at  $x = 0$  against the exact solution (137), achieving and excellent agreement for all output times.

#### 4.11. Double shear layer (INS)

This test, originally introduced in [83], is concerned with the incompressible flow of a double shear layer solving the Navier–Stokes equations. The physical dynamics consists in a horizontal flow jet of a fluid given by a small vertical pressure gradient balancing the convective forces. As a consequence, several vortices develop during the simulation. The computational domain is the square  $\Omega = [0; 1]^2$  with periodic boundaries. At the initial time the following pressure and velocity field are prescribed:

$$p = 1, \quad u = \begin{cases} \tanh(\theta(y - 1/4)), & y \leq 1/2 \\ \tanh(\theta(3/4 - y)), & y > 1/2 \end{cases}, \quad v = \delta \sin(2\pi x), \tag{138}$$

with parameters  $\theta = 30$  and  $\delta = 0.05$ . A Reynolds number  $Re = 5000$  is chosen, and the final time of the simulation is fixed at  $t_f = 1.8$ . The computational grid counts  $N_p = 15684$  polygonal cells with  $h = 1/100$ . Fig. 11 depicts



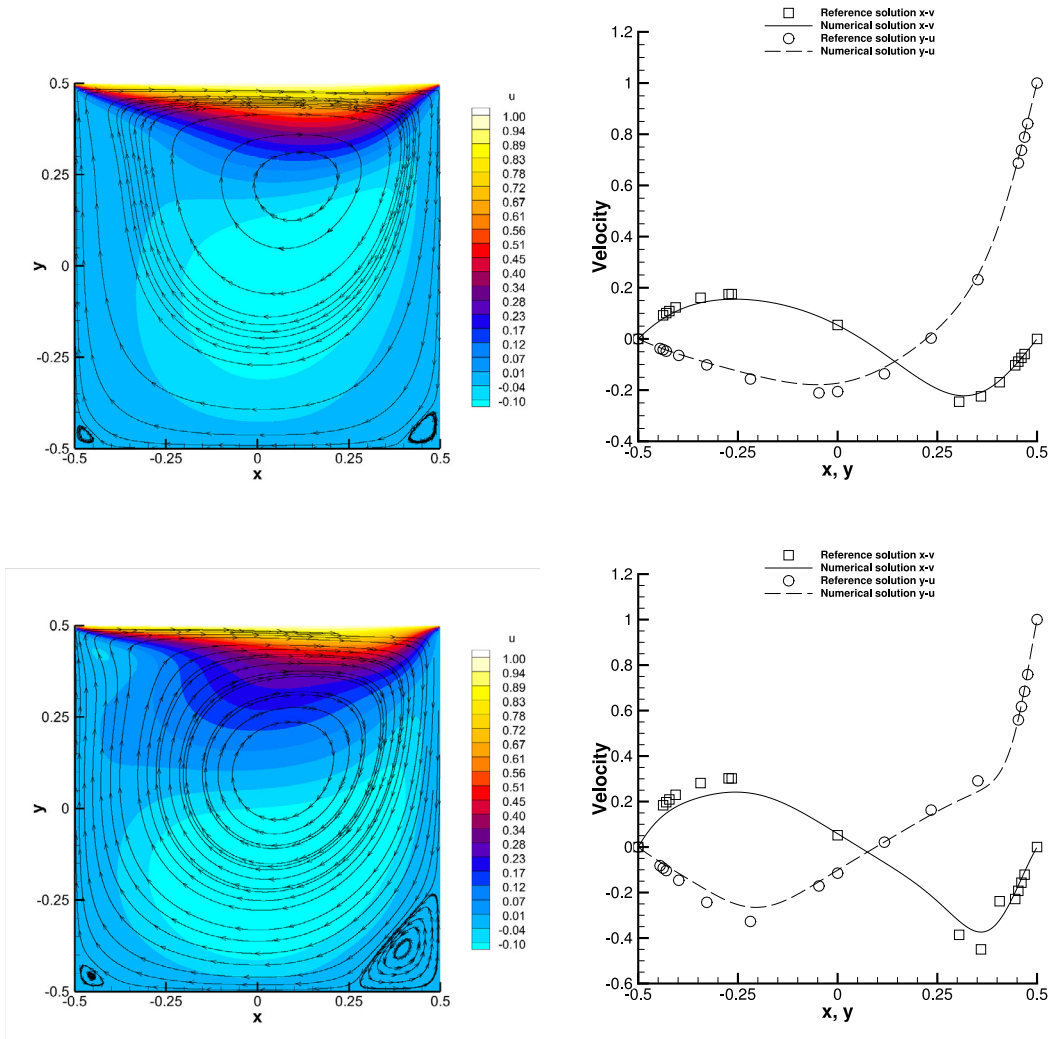
**Fig. 11.** Double shear layer. Vorticity magnitude at output times  $t = 0.4$  (top left),  $t = 0.8$  (top right),  $t = 1.2$  (bottom left) and  $t = 1.8$  (bottom right). The results are plotted with 41 contours in the interval  $[0; 26]$ .

the vorticity magnitude at different time instances. Even in this case, numerical results are qualitatively in very good agreement with reference solutions available in the literature [53,84,85].

#### 4.12. Lid-driven cavity flow (INS)

We further test the novel SI-FVVEM scheme on the classical benchmark of the lid driven cavity flow in the computational domain  $\Omega = [-0.5, 0.5]^2$ . In this case, the target is to find the steady hydrodynamics state for a fluid initially at rest, i.e.,  $p = 0$  and  $\mathbf{v} = \mathbf{0}$ . Wall boundary conditions are defined on the vertical sides ( $x = \pm 0.5$ ) and at the bottom ( $y = -0.5$ ) of the domain, while on the top side ( $y = 0.5$ ) the velocity field  $\mathbf{v} = (1, 0)$  is imposed. The characteristic mesh size is  $h = 1/80$ , which is rather coarse for this test case and yields a total number of  $N_p = 10093$  polygonal cells. The final time of the simulation is  $t_f = 25$ . Two different values of the Reynolds number are considered, namely  $Re = 100$  and  $Re = 400$ . All the simulations are run using the third order version of the scheme in space, while relying on a cheaper first order time discretization, because the goal is to catch the steady state. Fig. 12 shows the distribution of the horizontal velocity at the final time as well as the stream-traces of the velocity field, which permit to notice the generation of small vortical flows with a counter orientation with respect to the main vortex led by the cavity. We also show the numerical velocity components  $u$  and  $v$  along the cuts  $x = 0$  and  $y = 0$ , respectively, comparing them with the results found in [86]. From the plots, it is possible



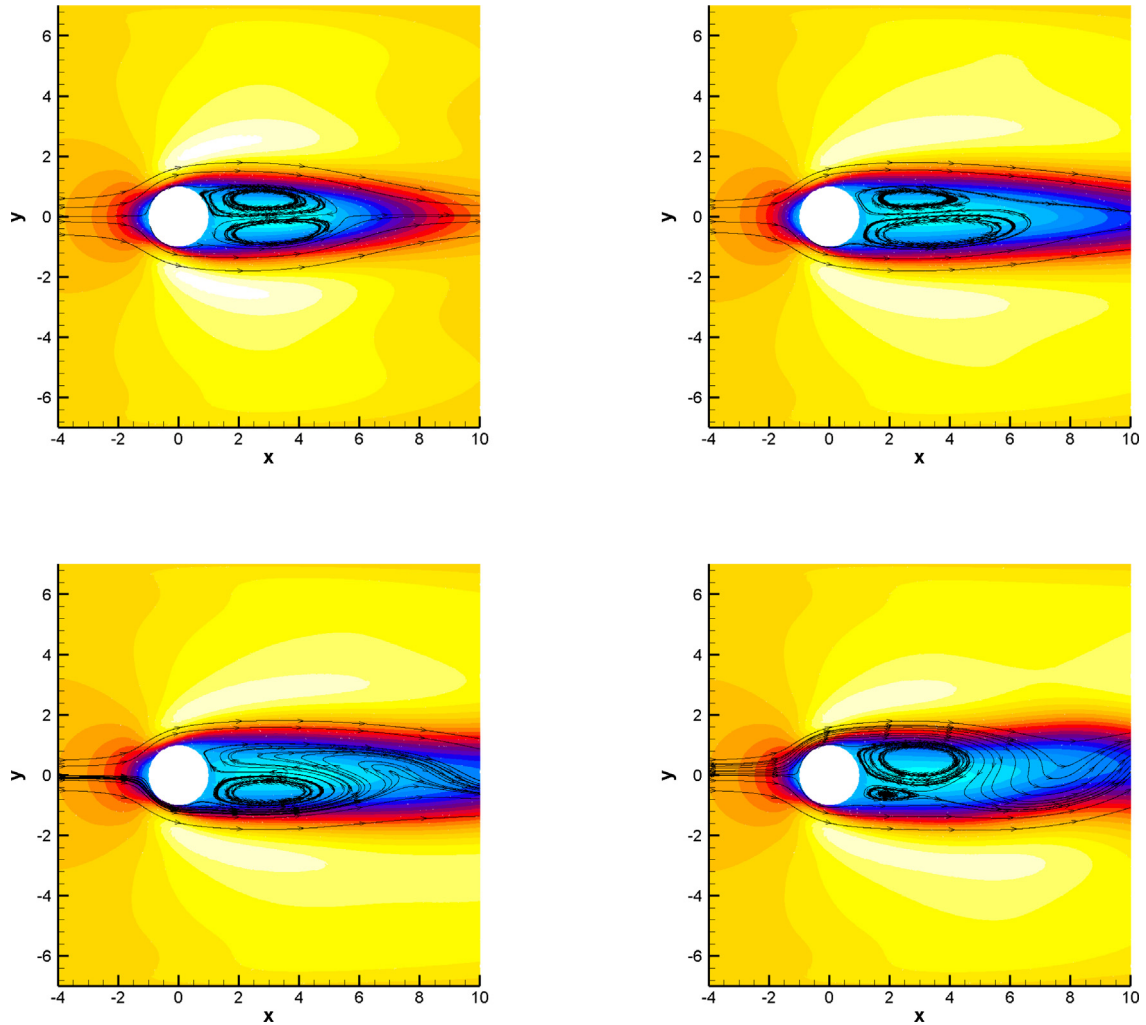


**Fig. 12.** Lid-driven cavity flow at time  $t = 25$  with a Reynolds number  $Re = 100$  (top row) and  $Re = 400$  (bottom row). Left: horizontal velocity contours and streamlines. Right: comparison with the reference solution of [86] for the velocity components  $u$  and  $v$  along the lines  $y = 0$  and  $x = 0$ .

to appreciate a good matching between our numerical results and data available in the literature for both Reynolds numbers.

#### 4.13. Laminar flow over a cylinder (INS)

We finally solve the problem related to a laminar flow over a circular cylinder [87], which generates a von Karman vortex that is shed behind the obstacle. The computational domain is defined as  $\Omega = [-10; 40] \times [-7; 7]$ , in which a cylinder of radius  $r_c = 1$  is located with centre at  $\mathbf{x}_c = (0, 0)$ . The computational mesh is made of  $N_p = 25390$  Voronoi cells with characteristic mesh size  $h = 0.25$ . On the left side of the domain we prescribe an inflow velocity of  $\mathbf{v} = (0.5, 0)$ , while the remaining sides are assigned with transmissive boundary conditions. On the obstacle, no-slip conditions are used. A constant pressure  $p = 1$  is imposed at the initial time, and the flow is characterized by a Reynolds number of  $Re = 100$ . A plot of the resulting stream-traces is reported in Fig. 13 at different output times, showing the generation of vortical flow patterns past the cylinder. Fig. 14 depicts the vorticity and the horizontal velocity distribution at the final time  $t_f = 300$ , clearly identifying two main vortexes departing from the obstacle that further generates the Von Karman street.



**Fig. 13.** Laminar flow over a cylinder. Stream-traces along the circular cylinder at times,  $t = 25, t = 50, t = 100$  and  $t = 200$  (from top left to bottom right). Colour map of the horizontal velocity  $u$  with 21 contour levels in the range  $[-0.05; 0.65]$ . (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

To give a quantitative interpretation, the same test case is also run using an inflow velocity of  $\mathbf{v} = (1, 0)$ , while keeping the same Reynolds number. The time evolution of the pressure is measured at the point  $\mathbf{x} = (15, 0)$ , from which we extract the associated frequency spectrum  $f$ . The results are depicted in Fig. 15, showing that we obtain a Strouhal number  $St = f r_c / u$  of  $St = 0.1401$  for  $u = 0.5$  and  $St = 0.1680$  for  $u = 1$  which is in reasonable good agreement with the value of  $St = 0.1649$  reported in [88].

### 5. Conclusions

The purpose of this work was to devise a new family of semi-implicit schemes on general polygonal meshes for the solution of multi-scale PDE. To this aim, the fast and the slow scales of the mathematical models are separated, so that a conservative shock capturing finite volume scheme is used to deal with the non-linear explicit convective terms, while the virtual element method is employed for discretizing the implicit sub-system. *Ad hoc* projector operators have been designed to transfer the numerical solution from one approximation space to another. A high order numerical solution in space and time is obtained by a CWENO spatial reconstruction and an IMEX Runge–Kutta time integrator. Moreover, high order of accuracy is granted in the implicit solver by the VEM paradigm. Well-balancing and asymptotic-preservation properties of the first order semi-discrete scheme have been demonstrated.

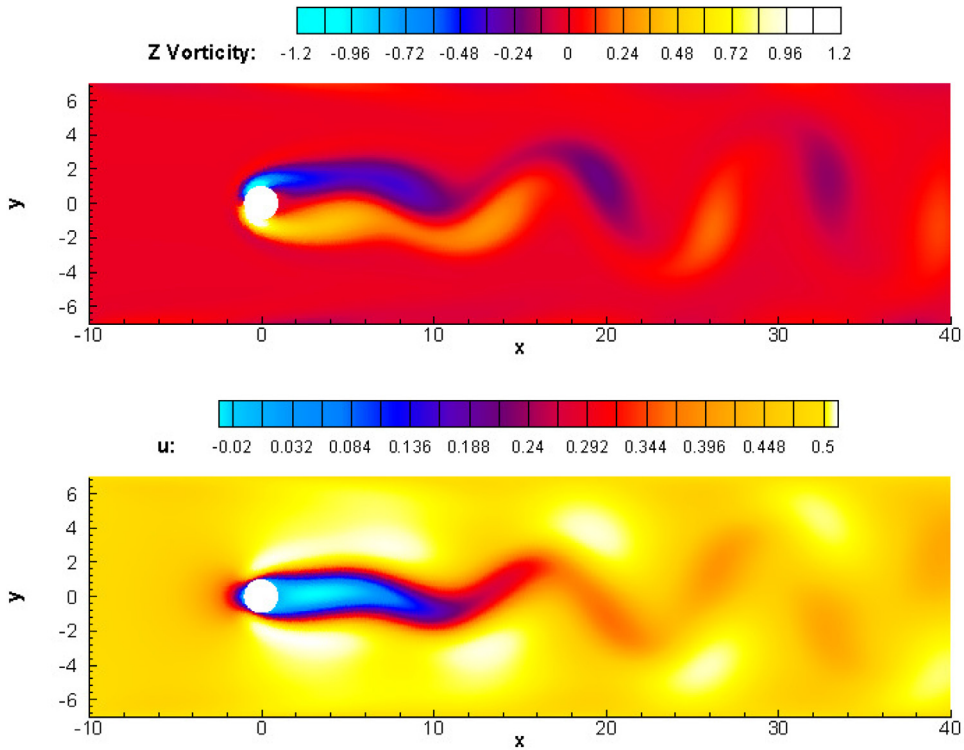


Fig. 14. Laminar flow over a cylinder at time  $t_f = 300$ . Map of the vorticity (top) and the horizontal velocity (bottom).

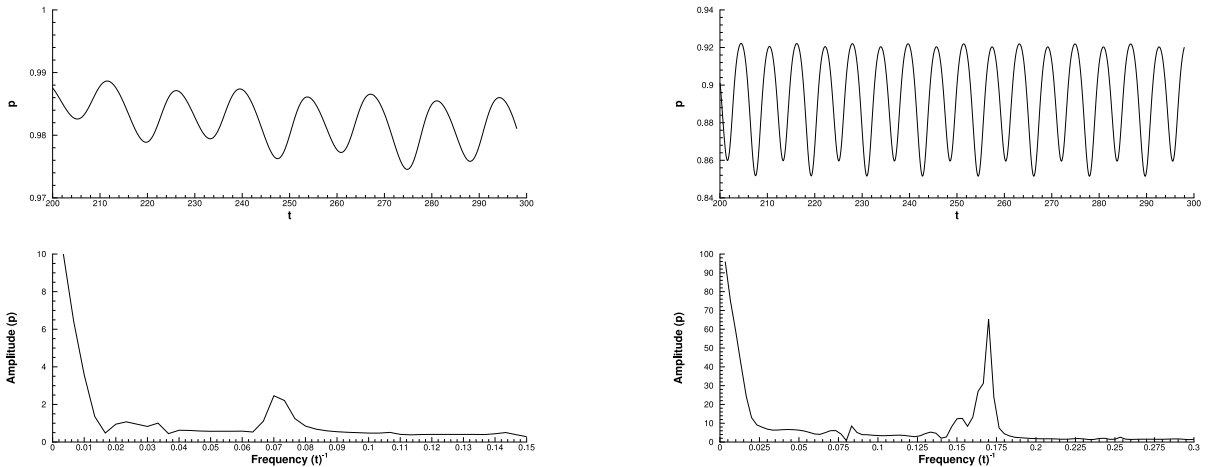


Fig. 15. Laminar flow over a cylinder at time  $t_f = 300$ . Pressure and related frequency modes measured at  $\mathbf{x} = (15, 0)$  for a flow with  $Re = 100$  and inflow velocity of  $u = 0.5$  (left) and  $u = 1$  (right).

As prototype examples, we have considered two non-linear systems of hyperbolic PDE that govern the motion of incompressible fluids, namely the shallow water equations and the incompressible Navier–Stokes model. The novel numerical scheme has been thoroughly tested against a wide range of academic benchmarks, showing accuracy and robustness capabilities. The VEM approach naturally allows to achieve high order space discretizations on arbitrary shaped meshes, that is here combined with a robust finite volume solver for treating shock waves and strong discontinuities. The resulting numerical method is stable, accurate and robust, being suitable for the treatment of complex geometries as well.

In the future we plan to extend the approach presented in this work to compressible flows, along the lines of [10]. Discontinuous Galerkin schemes in the framework of the virtual element space will also be investigated in order to devise a novel class of nonconforming virtual element methods. Finally, the development of purely VEM schemes in the context of semi-implicit time marching techniques is also foreseen, where even the convective sub-system is solved at the aid of the VEM paradigm, supplemented with suitable limiters to ensure stability and conservation at the same time.

**Declaration of competing interest**

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

**Data availability**

Data will be made available on request

**Acknowledgements**

WB and MGC received financial support by Fondazione Cariplo and Fondazione CDP (Italy) under the project No. 2022-1895. WB and GB are members of the GNCS-INdAM (*Istituto Nazionale di Alta Matematica*) group. AC and GB have been partially funded by the call “Bando Giovani anno 2022 per progetti di ricerca finanziati con il contributo 5x1000 anno 2020” of the University of Ferrara. This work was partially carried out at the Institute des Mathematiques de Bordeaux (IMB, Bordeaux-France) during the visiting program of WB and MGC.

**Appendix. IMEX Runge–Kutta schemes**

The Butcher tableau for the IMEX Runge–Kutta schemes used in this work are reported hereafter. They have been derived in [3,89,90] and each IMEX scheme is described with a triplet  $(s, \tilde{s}, p)$  which characterizes the number  $s$  of stages of the implicit method, the number  $\tilde{s}$  of stages of the explicit method and the order  $p$  of the resulting scheme. The acronym SA stands for Stiffly Accurate, LS implies L-Stability, while DIRK refers to Diagonally Implicit Runge–Kutta schemes.

- SP(1,1,1)

$$\begin{array}{c|c} 0 & 0 \\ \hline 1 & 1 \end{array} \quad \begin{array}{c|c} 1 & 1 \\ \hline 1 & 1 \end{array} \tag{A.1}$$

- LSDIRK2(2,2,2)  $\gamma = 1 - 1/\sqrt{2}$ ,  $\beta = 1/(2\gamma)$

$$\begin{array}{c|ccc|ccc} 0 & 0 & 0 & \gamma & \gamma & 0 \\ \beta & \beta & 0 & 1 & 1-\gamma & \gamma \\ \hline & 1-\gamma & \gamma & & 1-\gamma & \gamma \end{array} \tag{A.2}$$

- SA DIRK (3,4,3)  $\gamma = 0.435866$

$$\begin{array}{c|ccc|ccc|ccc} 0 & 0 & 0 & 0 & 0 & \gamma & \gamma & 0 & 0 & 0 \\ \gamma & \gamma & 0 & 0 & 0 & \gamma & 0 & \gamma & 0 & 0 \\ 0.717933 & 1.437745 & -0.719812 & 0 & 0 & 0.717933 & 0 & 0.282066 & \gamma & 0 \\ 1 & 0.916993 & 1/2 & -0.416993 & 0 & 1 & 0 & 1.208496 & -0.644363 & \gamma \\ \hline & 0 & 1.208496 & -0.644363 & \gamma & & 0 & 1.208496 & -0.644363 & \gamma \end{array} \tag{A.3}$$

Notice that the first order scheme (A.1) corresponds to the implicit Euler method.

## References

- [1] U.M. Ascher, S.J. Ruuth, R.J. Spiteri, Implicit-explicit Runge-Kutta methods for time-dependent partial differential equations, *Appl. Numer. Math.* 25 (1982) 151–167.
- [2] S. Boscarino, L. Pareschi, On the asymptotic properties of IMEX Runge–Kutta schemes for hyperbolic balance laws, *J. Comput. Appl. Math.* 316 (2017) 60–73.
- [3] L. Pareschi, G. Russo, Implicit-explicit Runge-Kutta schemes and applications to hyperbolic systems with relaxation, *J. Sci. Comput.* 25 (2005) 129–155.
- [4] S. Boscarino, F. Filbet, G. Russo, High order semi-implicit schemes for time dependent partial differential equations, *J. Sci. Comput.* 68 (2016) 975–1001.
- [5] V. Casulli, Semi-implicit finite difference methods for the two-dimensional shallow water equations, *J. Comput. Phys.* 86 (1990) 56–74.
- [6] V. Casulli, R. Cheng, Semi-implicit finite difference methods for three-dimensional shallow water flow, *Internat. J. Numer. Methods Fluids* 15 (1992) 629–648.
- [7] M. Dumbser, V. Casulli, A staggered semi-implicit spectral discontinuous Galerkin scheme for the shallow water equations, *Appl. Math. Comput.* 219 (2013) 8057–8077.
- [8] J. Park, C.-D. Munz, Multiple pressure variables methods for fluid flow at all Mach numbers, *Internat. J. Numer. Methods Fluids* 49 (2005) 905–931.
- [9] W. Boscheri, L. Pareschi, High order pressure-based semi-implicit IMEX schemes for the 3D Navier-Stokes equations at all Mach numbers, *J. Comput. Phys.* 434 (2021) 110206.
- [10] W. Boscheri, M. Tavelli, High order semi-implicit schemes for viscous compressible flows in 3D, *Appl. Math. Comput.* 434 (2022) 127457.
- [11] F. Fambri, A novel structure preserving semi-implicit finite volume method for viscous and resistive magnetohydrodynamics, *Internat. J. Numer. Methods Fluids* 93 (2021) 3447–3489.
- [12] M. Ioriatti, M. Dumbser, Semi-implicit staggered discontinuous Galerkin schemes for axially symmetric viscous compressible flows in elastic tubes, *Comput. & Fluids* 167 (2018) 166–179.
- [13] M. Ioriatti, M. Dumbser, A posteriori sub-cell finite volume limiting of staggered semi-implicit discontinuous Galerkin schemes for the shallow water equations, *Appl. Numer. Math.* 135 (2019) 443–480.
- [14] V. Casulli, A semi-implicit finite difference method for non-hydrostatic free-surface flows, *Int. J. Numer. Methods Fluids* 30 (1999) 425–440.
- [15] G. Tumolo, L. Bonaventura, M. Restelli, A semi-implicit, semi-Lagrangian, p-adaptive discontinuous Galerkin method for the shallow water equations, *J. Comput. Phys.* 232 (2013) 46–67.
- [16] G. Tumolo, L. Bonaventura, A semi-implicit, semi-Lagrangian discontinuous Galerkin framework for adaptive numerical weather prediction, *Q. J. R. Meteorol. Soc.* 141 (692) (2015) 2582–2601.
- [17] G. Orlando, T. Benacchio, L. Bonaventura, An IMEX-DG solver for atmospheric dynamics simulations with adaptive mesh refinement, *J. Comput. Appl. Math.* 427 (2023) 115124.
- [18] F. Fambri, M. Dumbser, Semi-implicit discontinuous Galerkin methods for the incompressible Navier–Stokes equations on adaptive staggered Cartesian grids, *Comput. Methods Appl. Mech. Engrg.* 324 (2017) 170–203.
- [19] W. Boscheri, M. Dumbser, M. Righetti, A semi-implicit scheme for 3D free surface flows with high-order velocity reconstruction on unstructured Voronoi meshes, *Internat. J. Numer. Methods Fluids* 72 (2013) 607–631.
- [20] W. Boscheri, G. Pisaturo, M. Righetti, High order divergence-free velocity reconstruction for free surface flows on unstructured Voronoi meshes, *Internat. J. Numer. Methods Fluids* 90 (2019) 296–321.
- [21] W. Boscheri, A space-time semi-Lagrangian advection scheme on staggered Voronoi meshes applied to free surface flows, *Comput. & Fluids* 202 (2020) 104503.
- [22] M. Tavelli, M. Dumbser, A high order semi-implicit discontinuous Galerkin method for the two dimensional shallow water equations on staggered unstructured meshes, *Appl. Math. Comput.* 234 (2014) 623–644.
- [23] M. Tavelli, M. Dumbser, A staggered space-time discontinuous Galerkin method for the three-dimensional incompressible Navier-Stokes equations on unstructured tetrahedral meshes, *J. Comput. Phys.* 319 (2016) 294–323.
- [24] G. Orlando, P.F. Barbante, L. Bonaventura, An efficient IMEX-DG solver for the compressible Navier-Stokes equations for non-ideal gases, *J. Comput. Phys.* 471 (2022) 111653.
- [25] S. Busto, J. Ferrín, E. Toro, M. Vázquez-Cendón, A projection hybrid high order finite volume/finite element method for incompressible turbulent flows, *J. Comput. Phys.* 353 (2018) 169–192.
- [26] A. Bermúdez, S. Busto, M. Dumbser, J. Ferrín, L. Saavedra, M. Vázquez-Cendón, A staggered semi-implicit hybrid FV/FE projection method for weakly compressible flows, *J. Comput. Phys.* 421 (2020) 109743.
- [27] S. Busto, L.D. Río, M. Vázquez-Cendón, M. Dumbser, A semi-implicit hybrid finite volume / finite element scheme for all Mach number flows on staggered unstructured meshes, *Appl. Math. Comput.* 402 (2021) 126117.
- [28] S. Busto, M. Dumbser, A staggered semi-implicit hybrid finite volume/finite element scheme for the shallow water equations at all Froude numbers, *Appl. Numer. Math.* 175 (2022) 108–132.
- [29] W. Boscheri, M. Tavelli, C. Castro, An all Froude high order IMEX scheme for the shallow water equations on unstructured Voronoi meshes, *Appl. Numer. Math.* 185 (2023) 311–335.
- [30] L. Beirão da Veiga, F. Brezzi, A. Cangiani, L. Marini, G. Manzini, A. Russo, The basic principles of Virtual Elements Methods, *Math. Models Methods Appl. Sci.* 23 (1) (2013) 199–214.
- [31] B. Ahmad, A. Alsaedi, F. Brezzi, L. Marini, A. Russo, Equivalent projectors for virtual element methods, *Comput. Math. Appl.* 66 (3) (2013) 376–391.

- [32] L. Beirão da Veiga, G. Manzini, A virtual element method with arbitrary regularity, *IMA J. Numer. Anal.* 34 (2) (2013) 759–781.
- [33] L. Beirão da Veiga, F. Brezzi, D. Marini, A. Russo, Virtual element methods for general second order elliptic problems on polygonal meshes, *Math. Models Methods Appl. Sci.* 26 (2016) 729–750.
- [34] K. Lipnikov, G. Manzini, M. Shashkov, Mimetic finite difference method, *J. Comput. Phys.* 257 (Part B) (2014) 1163–1227.
- [35] L. Beirão da Veiga, F. Brezzi, L. Marini, Virtual elements for linear elasticity problems, *SIAM J. Numer. Anal.* 51 (2) (2013) 794–812.
- [36] A. Gain, C. Talischì, G. Paulino, On the virtual element method for three-dimensional linear elasticity problems on arbitrary polyhedral meshes, *Comput. Methods Appl. Mech. Engrg.* 282 (2014) 132–160.
- [37] E. Artioli, L.B. da Veiga, F. Dassi, Curvilinear virtual elements for 2D solid mechanics applications, *Comput. Methods Appl. Mech. Engrg.* 359 (2020) 112667.
- [38] F. Dassi, C. Lovadina, M. Visinoni, A three-dimensional hellinger–Reissner virtual element method for linear elasticity problems, *Comput. Methods Appl. Mech. Engrg.* 364 (2020) 112910.
- [39] H. Chi, L.B. da Veiga, G. Paulino, Some basic formulations of the virtual element method VEM for finite deformations, *Comput. Methods Appl. Mech. Engrg.* 318 (2017) 142–190.
- [40] P. Antonietti, G. Manzini, I. Mazzieri, H. Mourad, M. Verani, The arbitrary-order virtual element method for linear elastodynamics models: Convergence, stability and dispersion-dissipation analysis, *Internat. J. Numer. Methods Engrg.* 122 (2021) 934–971.
- [41] E. Benvenuti, A. Chiozzi, G. Manzini, N. Sukumar, Extended virtual element method for the Laplace problem with singularities and discontinuities, *Comput. Methods Appl. Mech. Engrg.* 356 (2019) 571–597.
- [42] E. Benvenuti, A. Chiozzi, G. Manzini, N. Sukumar, Extended virtual element method for two-dimensional linear elastic fracture, *Comput. Methods Appl. Mech. Engrg.* 390 (2022) 114352.
- [43] V. Nguyen-Thanh, X. Zhuang, H. Nguyen-Xuan, T. Rabczuk, P. Wriggers, A virtual element method for 2D linear elastic fracture analysis, *Comput. Methods Appl. Mech. Engrg.* 340 (2018) 366–395.
- [44] A. Hussein, F. Aldakheel, B. Hudobivnik, P. Wriggers, P. Guidault, O. Allix, A computational framework for brittle crack-propagation based on efficient virtual element method, *Comput. Methods Appl. Mech. Engrg.* 159 (2019) 15–32.
- [45] A. Borio, F. Hamon, N. Castelletto, J. White, R. Settgast, Hybrid mimetic finite-difference and virtual element formulation for coupled poromechanics, *Comput. Methods Appl. Mech. Engrg.* 383 (2021) 113917.
- [46] M. Benedetto, A. Borio, F. Kyburg, J. Mollica, S. Scialò, An arbitrary order Mixed Virtual Element formulation for coupled multi-dimensional flow problems, *Comput. Methods Appl. Mech. Engrg.* 391 (2022) 114204.
- [47] G. Wang, L. Mu, Y. Wang, Y. He, A pressure-robust virtual element method for the Stokes problem, *Comput. Methods Appl. Mech. Engrg.* 382 (2021) 113879.
- [48] L.B. 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.
- [49] A. Chernov, C. Marcati, L. Mascotto, P- and hp- virtual elements for the Stokes problem, *Adv. Comput. Math.* 47 (24) (2021).
- [50] Y. Wang, G. Wang, F. Wang, An adaptive virtual element method for incompressible flow, *Comput. Math. Appl.* 101 (2021) 63–73.
- [51] L. Beirão da Veiga, F. Dassi, D.A. Di Pietro, J. Droniou, Arbitrary-order pressure-robust DDR and VEM methods for the Stokes problem on polyhedral meshes, *Comput. Methods Appl. Mech. Engrg.* 397 (2022) 115061.
- [52] P. Antonietti, G. Vacca, M. Verani, Virtual element method for the Navier–Stokes equation coupled with the heat equation, *IMA J. Numer. Anal.* (2022) <http://dx.doi.org/10.1093/imanum/drac072>.
- [53] W. Boscheri, G. Dimarco, M. Tavelli, An efficient second order all Mach finite volume solver for the compressible Navier–Stokes equations, *Comput. Methods Appl. Mech. Engrg.* 374 (2021) 113602.
- [54] W. Boscheri, M. Tavelli, N. Paoluzzi, High order Finite Difference/Discontinuous Galerkin schemes for the incompressible Navier–Stokes equations with implicit viscosity, *Commun. Appl. Ind. Math.* 13 (2022) 21–38.
- [55] E. Toro, C. Castro, D. Vanzo, A. Siviglia, A flux-vector splitting scheme for the shallow water equations extended to high-order on unstructured meshes, *Internat. J. Numer. Methods Fluids* 1 (2022) 1–27.
- [56] S. Vater, R. Klein, A semi-implicit multiscale scheme for shallow water flows at low Froude number, *Commun. Appl. Math. Comput. Sci.* 13 (2018) 303–336.
- [57] E. Toro, M. Vázquez-Cendón, Flux splitting schemes for the Euler equations, *Comput. & Fluids* 70 (2012) 1–12.
- [58] M. Dumbser, V. Casulli, A conservative, weakly nonlinear semi-implicit finite volume scheme for the compressible Navier–Stokes equations with general equation of state, *Appl. Math. Comput.* 272 (2016) 479–497.
- [59] S. Osher, F. Solomon, A partially implicit method for large stiff systems of Ode’s with only few equations introducing small time-constants, *SIAM J. Numer. Anal.* 13 (1976) 645–663.
- [60] H. Guillard, A. Murrone, On the behavior of upwind schemes in the low Mach number limit : II. Godunov type schemes, *Comput. & Fluids* 33 (2004) 655–675.
- [61] H. Guillard, C. Viozat, On the behavior of upwind schemes in the low Mach limit, *Comput. & Fluids* 28 (1999) 63–86.
- [62] S. Dellacherie, Analysis of Godunov type schemes applied to the compressible Euler system at low Mach number, *J. Comput. Phys.* 229 (2010) 978–1016.
- [63] A. Stroud, *Approximate Calculation of Multiple Integrals*, Prentice-Hall Inc., Englewood Cliffs, New Jersey, 1971.
- [64] A. Sommariva, M. Vianello, Gauss–Green cubature and moment computation over arbitrary geometries, *J. Comput. Appl. Math.* 231 (2009) 886–896.
- [65] A. Martinez, F. Piazzon, A. Sommariva, M. Vianello, Quadrature-based polynomial optimization, *Optim. Lett.* 14 (2020) 1027–1036.
- [66] M. Tavelli, M. Dumbser, A staggered semi-implicit discontinuous Galerkin method for the two dimensional incompressible Navier–Stokes equations, *Appl. Math. Comput.* 248 (2014) 70–92.
- [67] Y. Saad, M. Schultz, GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems, *SIAM J. Sci. Stat. Comput.* 7 (1986) 856–869.

- [68] R. LeVeque, Balancing source terms and flux gradients in high-resolution Godunov methods: the quasi-steady wave-propagation algorithm, *J. Comput. Phys.* 146 (1998) 346–365.
- [69] H. Schlichting, K. Gersten, *Grenzschichttheorie*, Springer Verlag, 2005.
- [70] D. Levy, G. Puppo, G. Russo, Central WENO schemes for hyperbolic systems of conservation laws, *M2AN Math. Model. Numer. Anal.* 33 (3) (1999) 547–571.
- [71] D. Levy, G. Puppo, G. Russo, Compact central WENO schemes for multidimensional conservation laws, *SIAM J. Sci. Comput.* 22 (2) (2000) 656–672.
- [72] W. Boscheri, G. Dimarco, High order central WENO-Implicit-Explicit Runge Kutta schemes for the BGK model on general polygonal meshes, *J. Comput. Phys.* 422 (2020) 109766.
- [73] W. Boscheri, G. Dimarco, High order finite volume schemes with IMEX time stepping for the Boltzmann model on unstructured meshes, *Comput. Methods Appl. Mech. Engrg.* 387 (2021) 114180.
- [74] L. Beirão da Veiga, F. Dassi, A. Russo, High-order virtual element method on polyhedral meshes, *Comput. Methods Appl. Mech. Engrg.* 74 (2017) 1110–1122..
- [75] L. Mascotto, Ill-conditioning in the virtual element method: Stabilizations and bases, *Numer. Methods Partial Differential Equations* 34 (4) (2018) 1258–1281.
- [76] A. Canestrelli, M. Dumbser, A. Siviglia, E. Toro, Well-balanced high-order centered schemes on unstructured meshes for shallow water equations with fixed and mobile bed, *Adv. Water Resour.* 33 (3) (2010) 291–303.
- [77] E. Toro, *Riemann Solvers and Numerical Methods for Fluid Dynamics: A Practical Introduction*, Springer, 2009.
- [78] E. Toro, *Shock-Capturing Methods for Free-Surface Shallow Flows*, John Wiley & Sons, 2001.
- [79] R. Bernetti, V. Titarev, E. Toro, Exact solution of the Riemann problem for the shallow water equations with discontinuous bottom geometry, *J. Comput. Phys.* 227 (2008) 3212–3243.
- [80] F. Bassi, S. Rebay, High-order accurate discontinuous finite element solution of the 2D Euler equations, *J. Comput. Phys.* 138 (1997) 251–285.
- [81] J.R. Womersley, Method for the calculation of velocity, rate of flow and viscous drag in arteries when the pressure gradient is known, *J. Physiol.* 127 (3) (1955) 553.
- [82] C. Loudon, A. Tordesillas, The use of the dimensionless Womersley number to characterize the unsteady nature of internal flow, *J. Theoret. Biol.* 191 (1998) 63–78.
- [83] J.B. Bell, P. Colella, H.M. Glaz, A second-order projection method for the incompressible Navier-Stokes equations, *J. Comput. Phys.* 85 (2) (1989) 257–283.
- [84] M. Dumbser, I. Peshkov, E. Romenski, O. Zanotti, High order ADER schemes for a unified first order hyperbolic formulation of continuum mechanics: viscous heat-conducting fluids and elastic solids, *J. Comput. Phys.* 314 (2016) 824–862.
- [85] W. Boscheri, M. Dumbser, M. Ioriatti, I. Peshkov, E. Romenski, A structure-preserving staggered semi-implicit finite volume scheme for continuum mechanics, *J. Comput. Phys.* 424 (2021) 109866.
- [86] U. Ghia, K.N. Ghia, C.T. Shin, High-Re solutions for incompressible flow using Navier-Stokes equations and multigrid method, *J. Comput. Phys.* 48 (1982) 387–411.
- [87] M. Tavelli, M. Dumbser, A staggered space–time discontinuous Galerkin method for the incompressible Navier-Stokes equations on two-dimensional triangular meshes, *Comput. & Fluids* 119 (2015) 235–249.
- [88] L. Qu, C. Norberg, L. Davidson, S.-H. Peng, F. Wang, Quantitative numerical analysis of flow past a circular cylinder at Reynolds number between 50 and 200, *J. Fluids Struct.* 39 (2013) 347–370.
- [89] L. Pareschi, G. Russo, High order asymptotically strong-stability-preserving methods for hyperbolic systems with stiff relaxation, in: T. Hou, E. Tadmor (Eds.), *Hyperbolic Problems: Theory, Numerics, Applications*, Springer, Berlin, 2003, pp. 241–251.
- [90] S. Boscarino, J. Qiu, G. Russo, T. Xiong, High order semi-implicit WENO schemes for all-mach full Euler system of gas dynamics, *SIAM J. Sci. Comput.* 44 (2022) B368–B394.