POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

Verification and validation of quasi-static and dynamic VMS-LES methods for the numerical simulation of turbulent flows

Tesi di Laurea Magistrale in
Energy Engineering - Ingegneria Energetica

Author: **Arancia Malaspina**

Student ID: 971706
Advisor: Prof. Alfio Quarteroni
Co-advisors: Prof. Luca Dedé, Dr. Alberto Zingaro, Lorenzo Bennati
Academic Year: 2021-22

Abstract

One of the most challenging topic in the field of computational fluid dynamics (CFD), is the solution to the Navier-Stokes equations for turbulent flows. Over the years, mainly three approaches have been developed: Direct Numerical Simulations (DNS), Large Eddy Simulations (LES) and Reynolds Averaged Navier-Stokes (RANS).
In this study, we analyse the Variational Multiscale-Large Eddy Simulations (VMS-LES) method, which was introduced as an improvement of standard LES methods. The idea behind the VMS-LES is to split the solution into the direct sum of a coarse scale and a fine scale term, instead of using a filtering procedure, as in traditional LES methods. This approach allows to stabilize advection dominated regimes, to satisfy the inf-sup condition and to model the turbulent regime of the blood flow according to the LES paradigm.
While the coarse scale solution is approximated by standard spatial discretization methods (we use the finite element method), many different definitions are available in the literature for the fine scales.
In this work, we analyse two of them, performing the software verification and the model validation of the VMS-LES with either quasi-static or dynamic approximation of fine scales. Differently from the quasi-static, the dynamic VMS-LES method encompasses an additional ODE that models the time evolution of the fine scale velocity. By doing so, the dynamic VMS-LES method solves in a natural way the issue of the pressure stabilization parameter degeneration, characterizing the quasi-static version of the method.
To perform verification and validation we consider the Beltrami-flow and the Taylor-Green-Vortex flow respectively. The first one is a simple test case used to check if the implementation of the method is correct. While the second one is a turbulence benchmark, used to check if the method is able to actually reproduce the physical flow motion. Moreover, in the last chapter of this thesis, we test the two methods an a real application: a patience-specific stenotic carotid.
We assess the performances of the two methods by testing several configurations of the flow conditions and of the numerical setup, to present a systematic comparison between the two on benchmark flows and on a real case. This allows us to investigate to which extent the dynamic is an improvement of the quasi-static VMS-LES method, as often

suggested in the literature.
In this regard, we highlight that the dynamic method overcomes the inconsistency related to the pressure stabilization parameter degeneration. Moreover, both the quasi-static and the dynamic VMS-LES methods with an implicit treatment of the non-convective term, perform well in verification and therefore, they should be free from implementation errors. However, in the model validation only the first one has accurately captured the turbulence indicators.

Keywords: Computational fluid dynamics, verification, validation, numerical methods, turbulent flows, Large Eddy Simulation, VMS-LES

Abstract in lingua italiana

Una delle sfide più impegnative nel campo della fluidodinamica computazionale (CFD) è la risoluzione delle equazioni di Navier-Stokes in regime turbolento. Per risolvere questo problema, nel corso degli anni, sono stati sviluppati diversi metodi, tra i quali i tre principali sono: le Direct Numerical Simulations (DNS), le Large Eddy Simulations (LES) e le Reynolds Averaged Navier-Stokes (RANS).
In questo studio analizzeremo il metodo Variational Multiscale-Large Eddy Simulation (VMS-LES), un metodo LES nato per migliorare alcuni aspetti critici dei tradizionali metodi LES.
Il punto di partenza dei VMS-LES è quello di definire la soluzione come la somma diretta di un termine rappresentativo della scala grande ed uno rappresentativo della scala fine, in contrapposizione ai LES tradizionali che si basano su un approccio di filtraggio. In tal modo, è possibile stabilizzare il regime avvettivo, soddisfare la condizione di inf-sup e modellare la turbolenza dei flussi sanguigni in analogia ai metodi LES. Nel metodo VMS-LES la scala grande viene approssimata in spazio utilizzando i metodi tradizionali quali, per esempio, il metodo agli elementi finiti, mentre per la scala fine è possibile adottare diverse formulazioni. In questa tesi ne analizzeremo due, affrontando la verificazione e la validazione del metodo VMS-LES con approssimazione della scala fine sia quasi-statica sia dinamica. A differenza del quasi-statico, nel metodo VMS-LES dinamico, la soluzione fine della velocità è ricavata risolvendo una equazione differenziale ordinaria. Così facendo, si dovrebbe risolvere l'inconsistenza del parametro di stabilizzazione della pressione, che nel metodo quasi-statico tende ad infinito quando si raffina la discretizzazione temporale.
Per la verificazione e la validazione dei metodi abbiamo utilizzato il flusso di Beltrami ed il Taylor-Green-Vortex, rispettivamente, due casi test con proprietà differenti. Il primo è un caso semplice per cui è nota la soluzione analitica, utilizzato per valutare se il metodo è stato implementato correttamente. Il secondo è un benchmark di turbolenza che permette di verificare se il metodo è in grado di catturare i meccanismi principali di quest'ultima. Poichè non ci sono ancora molti studi che confrontano i risultati dei due metodi, in questa tesi proponiamo un' analisi sistematica sui due casi test appena citati, con l'obbiettivo di confrontare quantitativamente il metodo dinamico rispetto al VMS-LES quasi-statico, al

variare delle condizioni del flusso e del setup numerico. Infine, per fornire un riscontro pratico, applicheremo i due metodi ad un caso reale.
Per concludere, possiamo affermare che il metodo dinamico sta effettivamente risolvendo l'inconsistenza legata al parametro di stabilizzazione della pressione. Inoltre, dalla verificazione si evince che i metodi VMS-LES quasi-statico e dinamico, combinati con uno schema implicito per l'approssimazione del termine non lineare, non contengono errori di implementazione. Tuttavia, dalla validazione del modello, riscontriamo che solamente il primo è in grado di simulare accuratamente la fisica del problema.

Parole chiave: Fluidodinamica computazionale, verificazione, validazione, metodi numerici, turbolenza, Large-Eddy-Simulations, VMS-LES

Contents

1 | Introduction

The most challenging topic to explain to people unfamiliar with numerical methods, is which is *magic trick* connecting physics, equations and models. In this introduction, we try to clarify this link, so that the reader, regardless his/her background, will be able to grasp the sense of our study.

Everybody is aware of the bubbling wakes behind boats or planes, but how do we translate them in mathematics and calculations?
The answer is in the Navier-Stokes equations, the mathematical model which represents the motion of a fluid in terms of velocity and pressure, for a given set of boundary and initial conditions, and for any viscosity and density characterizing the specific fluid at hand. Navier-Stokes (NS) equations are derived by applying mass conservation and momentum balance to an incompressible fluid[1], thanks to a limit process acting on infinitesimal control volumes. The solutions are functions in space and time, which represent the evolution of pressure and velocity in every point of the domain.
However, unless for very simple situations, an analytical solution which solves the equations exactly, cannot be found [20]. Therefore, we have only two options: solving exactly strongly simplified cases, or approximating more challenging (and realistic) problems.
Accordingly, since our goal is to make prediction and computations of real applications, we need to approximate the original problem. This means that, instead of solving equations in a continuous domain (i.e. in every point of the domain) we compute the solution only on some points, and we assume that for an arbitrary small portion of the space around these points the same solution applies. The smaller is that portion, the more accurate is the solution. In this regard, instead of computing solutions on an infinite number of points we compute them on a finite subset of them, switching from a continuos to a discrete approach. To link the continuos and the discrete approach we need the so-called "numerical methods".
A numerical method is the mathematical procedure that legitimates and describes the

[1] In this work we consider only incompressible fluids, with constant density. However, Navier-Stokes equations are defined also for compressible fluids

discrete approach, and the set of operations to compute the approximated solution. Notice that, the more we approximate a problem, the less computations we need to solve, however the quality of the numerical solution becomes very poor. Indeed, one of the main challenge of numerical mathematics is finding the best compromise between a good approximation and the time of resolution, i.e. between accuracy and computational cost. In the next paragraphs, we will go through this problem, in regard of computational fluid dynamic methods for turbulent flows.

The regime of fluids in motion is described by the Reynolds number (Re), a dimensionless quantity which depends on the physical properties of the flow. Every fluid has a critical Re, which identifies the transition from a laminar to a turbulent flow, depending on geometrical and physical quantities. Turbulent conditions are the most challenging to solve, due to their chaotic nature. Among the difficulties, we mention that the flow becomes unsteady and the non-linear term in NS equations cannot be neglected.
Many different approaches have been proposed, to model turbulent flows. The most popular are: Direct Numerical Simulations (DNS), Reynolds-Averaged Navier-Stokes equations (RANS) and Large Eddy Simulation (LES) [22, 33].
In the first case, Navier-Stokes equations are solved without any additional modelling, therefore all ranges of spatial and temporal scales are fully resolved. This is the approach with the highest accuracy, but the highest computational costs.
In the second case, we solve only the main flow and we average the effects of turbulence through extra-terms computed with statistical models (Figure 1.1). Therefore, the computational effort of this method is the lowest, as well as accuracy, unfortunately.
The third case is an intermediate between the first two: small scales are modelled, while large scales are solved directly. This has shown to be the approach with the greatest potential, especially in the scope of turbulent flows. Indeed, if compared with Direct Numerical Simulation (DNS) LES methods have much lower computational costs, since part of the problem is only modelled, but they are more accurate than RANS.
The methods presented in this work belong to the family of LES.

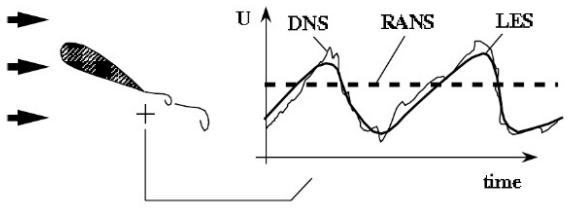


Figure 1.1: DNS, RANS, LES approximations of Navier-Stokes equations.
Figure from [22].

So far, we have presented different numerical methods, focusing on accuracy and computational costs as main features. However, besides these, there are other properties that a numerical method needs to satisfy.
As we know from the Lax-Richtmyer Equivalence Theorem (1.1), often called *"Fundamental Theorem of Numerical Analysis"*, a numerical method has to respect the constrains of consistency and stability in order to be convergent [32].

$$consistency + stability \Longleftrightarrow convergence \tag{1.1}$$

Therefore, when a new method or an improvement to an existing one is implemented, its validity has to be proved.
Among the procedures to test a numerical method we have verification and validation (V&V). To give an heuristic overview of these two techniques, verification aims to answer the question "are we building the product correctly?", while validation aims to answer the question "are we building the right product?". The detailed description of V&V is reported in ASME V&V-20 [15] , which is a document that defines all requirements and guidelines for computational fluid dynamics analysis and provides a valuable reference for a state of the art verification and validation. We summarize here the keypoints:

(i.) The verification (or software verification) aims to check if the computational model accurately represents the underlying mathematical method and its solution. In particular, with this test we check either if our computational model and its implementation are correct, or we evaluate the accuracy of the solutions. For these purposes, we have to assess consistency and convergence and to compare the numerical solution with the analytic one, to compute errors and perform a grid convergence analysis. The test-cases used for verification are not necessary representative of real applications, however, they must have an analytical solution [1, 15].

(ii.) The aim of a model validation is to check if the method is able to represent the real world with the expected degree of precision. The numerical solutions are compared either with an experimental solution or with the results of a DNS (as in our case). The quantities used for the comparison are physical quantities, like kinetic energy or enstrophy and the fluid dynamic benchmarks have to represent all the main feature of real applications, which in our case is turbulence.

The objective of this work is to present the software verification and the model validation of the $quasi-static$ and $dynamic$ VMS-LES methods to solve Navier-Stokes equations for incompressible fluids, at different Reynolds numbers.

As suggested by the name, VMS-LES methods are LES methods to which a variational multi-scale approach is applied.
The theoretical idea behind LES methods is to split the solutions in two terms: a coarse scale term and a fine scale term. However, while traditional LES are based on a filtering procedure and, therefore, small and fine scale solutions are linked with closure equations, the VMS-LES are projectional (i.e. variational) methods. Accordingly, solutions are split into the direct sum of a small scale and a large scale term, hence the original problem is decomposed into two sub-problems of coupled non-linear variational equations.
Among the advantages of this approach, with respect to traditional LES, we mention that: [5, 25]

1. In a projectional formulation the fine scales solution is defined as function of the coarse scale solution. Therefore, we don't need closure equations to connect fine and coarse scales, as in the filtering procedure.

2. It is possible to avoid the use of extra terms such as eddy viscosity (an additional viscosity to model turbulence effects) to represent fine-scale dissipations. These terms introduce a consistency error, that makes the coarse scale equations of traditional LES inconsistent [8, 43].

3. In a variational formulation, equations of large scales are strongly consistent, i.e. replacing the approximated solutions with the exact ones, fluctuations and residuals will be equal to zero. Therefore, a method which gives a good approximation of the large scale solution has small residual and small fluctuations, which means that the fine scale solution is small.

4. In the traditional LES methods the fine scale has to be modelled and it is defined $a - priori$ as the filtering scale. In a variational formulation we model only the terms of the subscale, that cannot be caught by the numerical solver;

5. In the VMS-LES methods we define an analytical expression for the fine scale solution which depends on the coarse scale solution, and we substitute it in the original equations. Hence, the error will *only* be determined by the approximation error due to space and time discretization of the problem.

Further details concerning this theory can be found in papers [5, 25].

In this thesis, we analyse the quasi-static and the dynamic VMS-LES methods, which differ in the approximation of the fine scale solutions. In the first case, the functions which approximate the fine scale solutions are constant in time, i.e. their time derivative

is equal to zero [5, 21]. In the second case, we consider a time dependent solution for the velocity fine scale, computed by solving an additional ordinary differential equation (ODE) [13, 14]. The reason that prompted researchers to develop the dynamic method is that the quasi-static approach leads to an inconsistency of the pressure stabilization term, since the pressure stabilization parameter τ_C goes to infinite when reducing the time-step (see Section 2.3). On the contrary, the dynamic method is correcting in a natural way the behaviour of τ_C [14].

Objectives

In this thesis we go through the software verification and the model validation of the quasi-static and dynamic VMS-LES methods for turbulent flows. First, we perform the software verification on the Beltrami-flow test case, than the model validation on the Taylor-Green-Vortex (TGV) flow. The first one is a simple test case, not fully representative of turbulence even at high Re, for which the analytical solution is known. The second one is a fluid dynamic benchmark, for which a DNS solution is known. The TGV is commonly used for validation, since it represents the main feature of turbulence, such as the grinding down of eddies into smaller eddies, or the kinetic energy decay due to viscous forces [46].
The aims of V&V are either to check if the implementation of the software is free of mistakes (verification), or to check whether the mathematical model appropriately represent the physics of the problem (validation). Besides these, since there are still not many studies of this kind comparing the quasi-static and dynamic VMS-LES methods, we perform a systematic analysis of the performance of the two methods, changing a variety of assumptions.
To sum up the aims of our thesis, we distinguish two levels of objectives: $ex-post$ and $ex-ante$.
By $ex-ante$ objectives, we refer to the aims of verification and validation, which are established before starting the work. On the other side, the $ex-post$ achievements are those resulting from the analysis of the simulations and hence, what we find out evaluating the performances of the two methods.

Main achievements

Downstream our analysis, the dynamic VMS-LES method has demonstrated to solve the inconsistency arising in the quasi-static, but it still has some issues to be solved or further investigated. In this regard, we highlight that its semi-implicit formulation may have some

implementation errors, while the implicit formulation is introducing an extra dissipation of kinetic energy, which is not representative of the physics of the problem. Nevertheless, we believe that fixing these issues, the dynamic VMS-LES methods may improve the results of the quasi-static VMS-LES method.

Organization of the thesis

This thesis is organized as follows: in Chapter 2, we start from Navier-Stokes equations and we introduce step-by-step the formulation of the quasi-static and dynamic VMS-LES methods; in Chapter 3, we perform the Software verification; in Chapter 4, we validate the model; in Chapter 5, we test the two methods on a real application. Finally, in Chapter 6, we draw conclusions.

2 | Navier-Stokes VMS-LES modelling

Navier-Stokes equations are a set of partial differential equations describing the motion of a viscous fluid. The physics laws behind Navier-Stokes equations are mass conservation and momentum balance applied to a continuos fluid. Due to their complexity, it is not yet possible to find an analytical solution, unless several simplifying assumptions apply. Indeed, real applications are tackeled by solving NS equations with numerical methods.
We formulate the VMS-LES method starting from the strong formulation of the fluid dymamic problem, according to the following procedure: in Section 2.1, we define the weak formulation; in Section 2.2, we introduce the space discretization of the macro scale; in Section 2.3, we present two ways to approximate the fine scale terms. Finally, in Section 2.4, we present the time-discretization of the problem [21, 39].

2.1. The incompressible Navier-Stokes equations and the weak formulation of the problem

Let us consider a domain $\Omega \subset \mathbb{R}^d$ where $d = 3$, with a sufficiently regular boundary $\partial\Omega \equiv \Gamma$ such that $\partial\Omega = \Gamma_{\mathrm{N}} \cup \Gamma_{\mathrm{D}}$ and $\Gamma_{\mathrm{N}} \cap \Gamma_{\mathrm{D}} = \emptyset$. We define Γ_{N} the portion of the boundary where Neumann boundary conditions are applied and Γ_{D} the portion related to Dirichlet boundary data.
The Navier-Stokes equations for incompressible fluids read as follow:

$$\rho\frac{\partial \mathbf{u}}{\partial t} + \rho(\mathbf{u}\cdot\nabla)\mathbf{u} - \nabla\cdot\sigma(\mathbf{u},p) = \mathbf{f} \qquad \text{in } \Omega \times (0,T), \tag{2.1}$$

$$\nabla\cdot\mathbf{u} = 0 \qquad \text{in } \Omega \times (0,T), \tag{2.2}$$

$$\mathbf{u} = \mathbf{g} \qquad \text{on } \Gamma_{\mathrm{N}} \times (0,T), \tag{2.3}$$

$$\sigma(\mathbf{u},p)\hat{\mathbf{n}} = \mathbf{h} \qquad \text{on } \Gamma_{\mathrm{N}} \times (0,T), \tag{2.4}$$

$$\mathbf{u} = \mathbf{u}_0 \qquad \text{in } \Omega \times \{0\} \tag{2.5}$$

We denote by $\mathbf{u}$ and p velocity and pressure, respectively, which are the unknowns of the problem, and by ρ the fluid density.
The term $\sigma(\mathbf{u}$,p) is the total fluid stress tensor, which is defined for Newtonian, incompressible and viscous fluids, as:

$$\sigma(\mathbf{u}, p) = -p\mathbf{I} + 2\mu\epsilon(\mathbf{u}) \tag{2.6}$$

In equation (2.6) μ is the dynamic viscosity and $\epsilon(\mathbf{u})$ the strain-rate tensor:

$$\epsilon(\mathbf{u}) = \frac{1}{2}(\nabla\mathbf{u} + (\nabla\mathbf{u})^T) \tag{2.7}$$

hence:

$$\nabla \cdot \sigma(\mathbf{u}, p) = -\nabla p + \mu\Delta\mathbf{u} \tag{2.8}$$

The functions **f, g** and **h** are forcing term, Dirichlet and Neumann data, respectively and $\mathbf{u}_0$ is the initial datum.
The spaces of solutions $\mathcal{V}_g$, $\mathcal{Q}$, $\boldsymbol{\mathcal{V}}_g$ and test functions $\mathcal{V}_0$, $\mathcal{Q}$, $\boldsymbol{\mathcal{V}}_0$, used to develop the weak formulation, are infinite-dimensional functional spaces, since no approximation has already been introduced. They are defined as:

$$\mathcal{V}_g := \{\mathbf{v} \in [H^1(\Omega)]^d \;\; \mathbf{v} = \mathbf{g} \text{ on } \Gamma_{\mathrm{D}}\} \tag{2.9}$$
$$\mathcal{V}_0 := \{\mathbf{v} \in [H^1(\Omega)]^d \;\; \mathbf{v} = \mathbf{0} \text{ on } \Gamma_{\mathrm{D}}\} \tag{2.10}$$
$$\mathcal{Q} := L^2(\Omega) \tag{2.11}$$
$$\boldsymbol{\mathcal{V}}_g := \mathcal{V}_g \times \mathcal{Q} \tag{2.12}$$
$$\boldsymbol{\mathcal{V}}_0 := \mathcal{V}_0 \times \mathcal{Q} \tag{2.13}$$
$$\tag{2.14}$$

The weak formulation of the problem is obtained by first multiplying all members of equation (2.1) by a test function $\mathbf{v} \in \mathcal{V}_0$ and all members of (2.2) by the test function $q \in \mathcal{Q}$ and than integrating every term over Ω.

Therefore, the weak formulation of (2.1) reads as:

$$\int_\Omega \rho\frac{\partial\mathbf{u}}{\partial t} \cdot \mathbf{v}\; d\Omega + \int_\Omega \rho(\mathbf{u}\cdot\nabla)\mathbf{u}\cdot\mathbf{v}\; d\Omega - \int_\Omega (\nabla\cdot\sigma(\mathbf{u},p))\cdot\mathbf{v}\; d\Omega = \int_\Omega \mathbf{f}\cdot\mathbf{v}\; d\Omega \tag{2.15}$$

with $\displaystyle\int_\Omega \nabla\cdot\sigma\mathbf{v}\; d\Omega = -\int_\Omega \sigma\cdot\nabla\mathbf{v}\; d\Omega + \int_{\partial\Omega} \sigma\mathbf{v}\cdot\hat{\mathbf{n}}\; d\Gamma$

and the weak formulation of (2.2) reads as:

$$\int_\Omega q \ \nabla \cdot \mathbf{u} \ d\Omega = 0 \tag{2.16}$$

Plugging (2.6), (2.7) and boundary conditions in (2.15) and (2.16), we define:

$$\begin{cases} a(\mathbf{u}, \mathbf{v}) = \displaystyle\int_\Omega \rho(\mathbf{u} \cdot \nabla)\mathbf{u} \cdot \mathbf{v} \ d\Omega + \int_\Omega \mu \nabla \mathbf{u} \cdot \nabla \mathbf{v} \ d\Omega \\ b(\mathbf{u}, q) = -\displaystyle\int_\Omega q \nabla \cdot \mathbf{u} \ d\Omega \\ F(\mathbf{v}) = \displaystyle\int_{\Gamma_N} \mathbf{h} \cdot \mathbf{v} \ d\gamma + \int_\Omega \mathbf{f} \cdot \mathbf{v} \ d\Omega. \end{cases} \tag{2.17}$$

Let us now introduce the form:

$$A(\mathbf{u}, \mathbf{v}) = \int_\Omega \rho \frac{\partial \mathbf{u}}{\partial t} \cdot \mathbf{v} d\Omega + a(\mathbf{u}, \mathbf{v}) + b(\mathbf{v}, p), \tag{2.18}$$

By means of definition (2.18), we rearrange the weak formulation of the problem as follows: given $\mathbf{u}_0 \in \mathcal{V}_g$, for any $t \in (0, T)$ find $\mathbf{U} = \mathbf{U}(\mathbf{x}, t) = \{\mathbf{u}, p\} \in \boldsymbol{\mathcal{V}}_\mathbf{g}$ such that:

$$\begin{cases} A(\mathbf{u}, \mathbf{v}) = F(\mathbf{v}) & \text{in } \Omega \\ b(\mathbf{u}, q) = 0 & \text{in } \Omega, \end{cases} \tag{2.19}$$

for all $\{\mathbf{v}, q\} \in \boldsymbol{\mathcal{V}}_0$

2.2. Spatial discretization: Finite element method and VMS-LES modelling

The discretization of the weak problem decribed in (2.19) is obtained essentially in three steps: first, the solution is split in two contributes related to fine and coarse scales, according to the VMS approach [5, 25]; second, the coarse scale solution is approximated with piecewise polynomial functions, according to the Finite Elements Method (FEM); third, the fine scale is approximated with either the quasi-static or the dynamic method. The first two steps are presented in this section, while the last one in the next section.

Let us consider T_h a triangulation of the domain Ω made of hexaedra, the space of the finite elements approximating functions is:

$$X_h^r = \{v_h \in C^0(\Omega) : v_h^k|_K \in \mathbb{P}_r, \forall K \in T_h\} \tag{2.20}$$

where $r \geq 1$ is the polynomial degree of the approximating function over the element K and h is the size of $K \in T_h$.
According to the VMS theory [21, 26], we split the spaces defined in (2.10), (2.11) and (2.12) into the direct sum of two subspaces, such that:

$$\boldsymbol{\mathcal{V}}_g = \boldsymbol{\mathcal{V}}_g^h \oplus \boldsymbol{\mathcal{V}}_g', \tag{2.21}$$
$$\boldsymbol{\mathcal{V}}_0 = \boldsymbol{\mathcal{V}}_0^h \oplus \boldsymbol{\mathcal{V}}_0'. \tag{2.22}$$

The spaces $\boldsymbol{\mathcal{V}}_g^h = \mathcal{V}_g^h \times \mathcal{Q}^h$ and $\boldsymbol{\mathcal{V}}_0^h = \mathcal{V}_0^h \times \mathcal{Q}^h$ are the finite-dimensional spaces related to the coarse scales, which are approximated with the FEM, such that:

$$\mathcal{V}_g^h = \mathcal{V}_g \cap [X_h^r]^d, \tag{2.23}$$
$$\mathcal{Q}^h = \mathcal{Q} \cap X_h^r, \tag{2.24}$$

while the spaces of the fine scale are the following infinite-dimensional spaces:

$$\boldsymbol{\mathcal{V}}_g' = \mathcal{V}_g' \times \mathcal{Q}', \tag{2.25}$$
$$\boldsymbol{\mathcal{V}}_{\mathbf{0}} = \mathcal{V}_0' \times \mathcal{Q}'. \tag{2.26}$$

Analogously to the spaces decomposition, we split the solution and the test function into the direct sum of a coarse scale and a fine scale solution:

$$\mathbf{U} = \mathbf{U}^h + \mathbf{U}', \tag{2.27}$$
$$\mathbf{V} = \mathbf{V}^h + \mathbf{V}'. \tag{2.28}$$

Where subscripts h and $'$ indicate the projection of the solution on the coarse and the fine scale respectively, such that: $\mathbf{U}^h \in \boldsymbol{\mathcal{V}}_g^h, \mathbf{U}' \in \boldsymbol{\mathcal{V}}_g', \mathbf{V}^h \in \boldsymbol{\mathcal{V}}_0^h, \mathbf{V}' \in \mathcal{V}_0'$.
By plugging the spatial decomposition into the weak problem and thanks to the bilinearity of (2.18), we have:

$$A(\mathbf{U}^h + \mathbf{U}', \mathbf{V}^h) = F(\mathbf{V}^h), \tag{2.29}$$

$$A(\mathbf{U}^h + \mathbf{U}', \mathbf{V}') = F(\mathbf{V}'), \tag{2.30}$$

and therefore:

$$A(\mathbf{U}^h, \mathbf{V}^h) + A(\mathbf{U}', \mathbf{V}^h) = F(\mathbf{V}^h), \tag{2.31}$$
$$A(\mathbf{U}^h, \mathbf{V}^h) + A(\mathbf{U}', \mathbf{V}^h) = F(\mathbf{V}'), \tag{2.32}$$

The bilinear form $A(\mathbf{U}, \mathbf{V})$ can be written as the sum of a bilinear and a trilinear form, such that: $A(\mathbf{U}, \mathbf{V}) = A_1(\mathbf{U}, \mathbf{V}) + A_2(\mathbf{U}, \mathbf{U}, \mathbf{V})$,
where:

$$A_1(\mathbf{u}, \mathbf{v}) = \int_\Omega \rho \frac{\partial \mathbf{u}}{\partial t} \cdot \mathbf{v} d\Omega + \int_\Omega \mu \nabla \mathbf{u} \cdot \nabla \mathbf{v} \; d\Omega - \int_\Omega q \nabla \cdot \mathbf{u} \; d\Omega \tag{2.33}$$
$$A_2(\mathbf{u}, \mathbf{u}, \mathbf{v}) = \int_\Omega \rho (\mathbf{u} \cdot \nabla) \mathbf{u} \cdot \mathbf{v} \; d\Omega \tag{2.34}$$

Plugging these definitions in (2.29):

$$\begin{aligned} &A_1(\mathbf{U}^h, \mathbf{V}^h) + A_1(\mathbf{U}', \mathbf{V}^h) + \\ &A_2(\mathbf{U}^h, \mathbf{U}^h, \mathbf{V}^h) + A_2(\mathbf{U}', \mathbf{U}^h, \mathbf{V}^h) + A_2(\mathbf{U}^h, \mathbf{U}', \mathbf{V}^h) + A_2(\mathbf{U}', \mathbf{U}', \mathbf{V}^h) \\ &= F(\mathbf{V}^h) \end{aligned} \tag{2.35}$$

The two terms: $A_2(\mathbf{U}^h, \mathbf{U}', \mathbf{V}^h)$ and $A_2(\mathbf{U}', \mathbf{U}^h, \mathbf{V}^h)$ are the cross-stress terms, which represent the energy dissipation between coarse and fine scale, while $A_2(\mathbf{U}', \mathbf{U}', \mathbf{V}^h)$ is the Reynolds-stress term, related to viscous forces.

We consider the following assumptions [5]:

- $\mathbf{u}' = \mathbf{0}$ on Γ;
- $(\nabla \mathbf{v}^h, \mu \nabla \mathbf{u}') = \mathbf{0}$ in Ω.

where the syntax (P, Q) indicate: $(P, Q) = \int_\Omega P \cdot Q \; d\Omega$, and we integrate by part equation (2.35). Therefore, the weak problem (2.19) will read as:

given $\mathbf{u}_0$, for any $t \in (0,T)$, find $\mathbf{U}^h = \mathbf{U}^h(\mathbf{x},t) = \{\mathbf{u}^h, p^h\} \in \boldsymbol{\mathcal{V}}^h_{\mathbf{g}}$ such that:

$$\begin{aligned}
&\left(\mathbf{v}^h, \rho\frac{\partial \mathbf{u}^h}{\partial t}\right) + (\mathbf{v}^h, \rho(\mathbf{u}^h\cdot\nabla)\mathbf{u}^h) + (\nabla\mathbf{v}^h, \mu\nabla\mathbf{u}^h) - (\nabla\cdot\mathbf{v}^h, p^h) + (q^h, \nabla\cdot\mathbf{u}^h)\\
&\underbrace{-(\rho\mathbf{u}^h\cdot\nabla\mathbf{v}^h + \nabla q^h, \mathbf{u}') - (\nabla\cdot\mathbf{v}^h, p')}_{\text{I}}\\
&\underbrace{-(\rho\mathbf{u}^h\cdot(\nabla\mathbf{w}^h)^T, \mathbf{u}')}_{\text{II}}\underbrace{-(\rho\nabla\mathbf{v}^h, \mathbf{u}'\otimes\mathbf{u}')}_{\text{III}}\\
&= (\mathbf{v}^h, \mathbf{f}) + (\mathbf{v}^h, \mathbf{h})_{\Gamma_{\text{N}}}
\end{aligned} \tag{2.36}$$

for all $\mathbf{V}^h = \{\mathbf{v}^h, q^h\} \in \boldsymbol{\mathcal{V}}^h_{\mathbf{0}}$.

In the first and the last rows of equation (2.36) we have the terms related to the Navier-Stokes equations, while the in the second and in the third rows we have additional terms related to stabilization: (I) is the Streamline Upwind Petrov Galerkin - Pressure Stabilizing Petrov Galerkin (SUPG-PSPG) term, (II) is an additional stabilization term arising from the VMS modelling and (III) is the LES term which models the Reynolds stress [21] Applying the same procedure to equation (2.30) we have:

$$\begin{aligned}
&A_1(\mathbf{U}',\mathbf{V}') + A_1(\mathbf{U}^h,\mathbf{V}') +\\
&A_2(\mathbf{U}^h,\mathbf{U}^h,\mathbf{V}') + A_2(\mathbf{U}^h,\mathbf{U}',\mathbf{V}') + A_2(\mathbf{U}',\mathbf{U}^h,\mathbf{V}') + A_2(\mathbf{U}',\mathbf{U}',\mathbf{V}')\\
&= F(\mathbf{V}')
\end{aligned} \tag{2.37}$$

and by moving some terms to the right-hand-side:

$$\begin{aligned}
&A_1(\mathbf{U}',\mathbf{V}') + A_2(\mathbf{U}^h,\mathbf{U}',\mathbf{V}') + A_2(\mathbf{U}',\mathbf{U}^h,\mathbf{V}') + A_2(\mathbf{U}',\mathbf{U}',\mathbf{V}',)\\
&= \underbrace{F(\mathbf{V}') - A_1(\mathbf{U}^h,\mathbf{V}') - A_2(\mathbf{U}^h,\mathbf{U}^h,\mathbf{V}',)}_{RHS}
\end{aligned} \tag{2.38}$$

Notice that, in equation (2.38) RHS is the residual of the coarse scale equation (2.29) projected on the fine scale space $\boldsymbol{\mathcal{V}}'_0$. By defining it as $\mathbf{R}(\mathbf{U}^h,\mathbf{V}')$ we rewrite equation (2.38) as:

$$\begin{aligned}
&A_1(\mathbf{V}',\mathbf{U}') + A_2(\mathbf{V}',\mathbf{U}^h,\mathbf{U}') + A_2(\mathbf{V}',\mathbf{U}',\mathbf{U}^h) + A_2(\mathbf{V}',\mathbf{U}',\mathbf{U}')\\
&= \mathbf{R}(\mathbf{U}^h,\mathbf{V}')
\end{aligned} \tag{2.39}$$

Accordingly, the solution of (2.39) is the fine scale solution, which we define by means of the functional $\mathcal{F}'$ as function of the coarse scale solution and the coarse scale residual.

Therefore, changing the expressions of $\mathcal{F}'$ leads to different VMS-LES methods.

$$\mathbf{U}' = \mathcal{F}'(\mathbf{U}^h, \mathbf{R}(\mathbf{U}^h, \mathbf{V}')) \tag{2.40}$$

Thanks to this definition, it is possible to write the weak problem for the coarse scale in a closed form. Indeed, by plugging (2.40) in (2.29) we obtain the following formulation, which depends only on coarse scale:

Find $\mathbf{U}^h \in \boldsymbol{\mathcal{V}}^h_{\mathbf{g}}$ such that:

$$A(\mathbf{U}^h + \mathcal{F}'(\mathbf{U}^h, \mathbf{R}(\mathbf{U}^h, \mathbf{V}'), \mathbf{V}^h) = F(\mathbf{V}^h) \tag{2.41}$$

for all $\mathbf{V}^h \in \boldsymbol{\mathcal{V}}^h_0$

In order to solve problem (2.41) we have to introduce a numerical approximation of $\mathcal{F}'$, such that $\widetilde{\mathcal{F}'} \approx \mathcal{F}'$. This affects the fine scale solution and hence, also coarse scale solutions, which parametrically depends on the first one. Therefore, from now on, we should consider the numerical approximations of the solutions: $\widetilde{\mathbf{U}'} \approx \mathbf{U}'$ and $\widetilde{\mathbf{U}^h} \approx \mathbf{U}^h$. However, to not overburden the notation, all symbols will be written without tilde, but implicitly referring to it.

2.3. Definition of fine scale: Quasi-Static and Dynamic methods

To approximate the fine scale solution it is possible to adopt different approaches, which differ in the definition of $\widetilde{\mathcal{F}'}$. In this section we introduce the quasi-static and the dynamic methods. While the first one has already been widely analysed [5, 21], the second one is still under investigation, see [13, 14].

2.3.1. Quasi-static

The quasi-static approximation of the fine scale is defined for both pressure and velocity as [5]:

$$\mathbf{U}' \approx -\mathcal{L}(\mathbf{U}^h)\mathbf{R}(\mathbf{U}^h) \tag{2.42}$$

The idea behind this expression is to let the fine scales solution depend on the coarse scale solution, by computing the fine scale velocity and pressure as the product between the element-wise stabilization operator τ and the residual of the coarse scale equation:

$$\mathbf{R}(\mathbf{U}^h) = \left\{ \begin{array}{c} \mathbf{r}_M(\mathbf{u}^h, p^h) \\ r_C(\mathbf{u}^h) \end{array} \right\} \tag{2.43}$$

$$\mathcal{L}(\mathbf{U}^h) = \left[\begin{array}{cc} \tau_M(\mathbf{u}^h)\mathbf{I}_{3\times 3} & \mathbf{0}_{3\times 3} \\ \mathbf{0}_{3\times 3}^T & \tau_C(\mathbf{u}^h) \end{array} \right] \tag{2.44}$$

therefore, according to the definition of $\mathbf{U}'$ we obtain:

$$\mathbf{u}' \approx -\tau_M(\mathbf{u}^h)\mathbf{r}_M(\mathbf{u}^h, p^h) \tag{2.45}$$

$$p' \approx -\tau_C(\mathbf{u}^h) r_C(\mathbf{u}^h) \tag{2.46}$$

Equations (2.45) and (2.46) depend on $\mathbf{r}_M(\mathbf{u}^h, p^h)$ and $r_C(\mathbf{u}^h)$ which represent the two strong residuals of coarse scale momentum and continuity equations, respectively:

$$\mathbf{r}_M(\mathbf{u}^h, p^h) = \rho \left(\frac{\partial \mathbf{u}^h}{\partial t} + \mathbf{u}^h \cdot \nabla \mathbf{u}^h \right) + \nabla p^h - \mu \Delta \mathbf{u}^h - \mathbf{f} \tag{2.47}$$

$$r_C(\mathbf{u}^h) = \nabla \cdot \mathbf{u}^h \tag{2.48}$$

while $\tau_M(\mathbf{u}^h)$ and $\tau_C(\mathbf{u}^h)$ are the stabilization parameters defined as:

$$\tau_M(\mathbf{u}^h) = \left(\frac{\sigma_t^2 \rho^2}{\Delta t^2} + \rho^2 \mathbf{u}^h \cdot \overline{\mathbf{G}} \mathbf{u}^h + C_r \mu^2 \overline{\mathbf{G}} : \overline{\mathbf{G}} \right)^{-\frac{1}{2}} \tag{2.49}$$

$$\tau_C(\mathbf{u}^h) = (\tau_M(\mathbf{u}^h) \overline{\mathbf{g}} \cdot \overline{\mathbf{g}})^{-1} \tag{2.50}$$

In equation (2.49) σ_t represent the order of the time discretization and Δt is the time discretization parameter (see section 2.4). The constant $C_r = 15 \cdot 2^r$ is derived from an element wise inverse estimator, related to the inverse inequality, and it depends on r which is the polynomial degree of the finite elements (FE) approximating functions [21]. The terms $\overline{\mathbf{G}}$ and $\overline{\mathbf{g}}$ are metric tensor and metric vector, respectively, to map the physical element K in the parametric space ($\widetilde{K}$).
By plugging (2.45) and (2.46) in (2.36), we obtain the VMS-LES, semi-discrete formulation of the Navier-Stokes equations with a quasi-static approximation of fine scales:

given $\mathbf{u}_0$, for any $t \in (0,T)$, find $\mathbf{U}^h = \mathbf{U}^h(t) = \{\mathbf{u}^h, p^h\}(t) \in \boldsymbol{\mathcal{V}}^h_{\mathbf{g}}$ such that:

$$\begin{aligned}
&\left(\mathbf{v}^h, \rho\frac{\partial \mathbf{u}^h}{\partial t}\right) + (\mathbf{v}^h, \rho(\mathbf{u}^h \cdot \nabla)\mathbf{u}^h) + (\nabla \mathbf{v}^h, \mu \nabla \mathbf{u}^h) - (\nabla \cdot \mathbf{v}^h, p^h) + (q^h, \nabla \cdot \mathbf{u}^h) \\
&+(\rho \mathbf{u}^h \cdot \nabla \mathbf{v}^h + \nabla q^h, \tau_M(\mathbf{u}^h)\mathbf{r}_M(\mathbf{u}^h, p^h)) - (\nabla \cdot \mathbf{v}^h, \tau_C(\mathbf{u}^h) r_M(\mathbf{u}^h)) \\
&-(\rho \mathbf{u}^h \cdot (\nabla \mathbf{v}^h)^T, \tau_M(\mathbf{u}^h)\mathbf{r}_M(\mathbf{u}^h, p^h)) \\
&+(\rho \nabla \mathbf{v}^h, \tau_M(\mathbf{u}^h)\mathbf{r}_M(\mathbf{u}^h, p^h) \otimes \tau_M(\mathbf{u}^h)\mathbf{r}_M(\mathbf{u}^h, p^h)) \\
&= (\mathbf{v}^h, \mathbf{f}) + (\mathbf{v}^h, \mathbf{h})_{\Gamma_N}
\end{aligned} \tag{2.51}$$

for all $\mathbf{V}^h = \{\mathbf{v}^h, q^h\} \in \boldsymbol{\mathcal{V}}^h_0$

A major issue in this approach is the behaviour of the stabilization parameter τ_C when $\Delta t \to 0$. If $\Delta t = \mathcal{O}\left(\frac{h}{|\mathbf{u}^h|}\right)$ there will not be any critical degeneration. However, reducing Δt with a fixed ratio $\frac{h}{|\mathbf{u}^h|}$, the first term in equation (2.49) will predominate over the others, leading to have $\tau_M \to 0$ and $\tau_C \to \infty$. This makes the quasi-static method inconsistent for lower values of the time discretization parameter. [5]
Therefore, the dynamic method was developed to avoid the degeneration of the stabilization parameter [13].

2.3.2. Dynamic

In the previous section we have introduced the quasi-static method and its limitations, in this section we introduce the dynamic VMS-LES method. The main difference between the two approaches, is that in the quasi-static $\mathbf{u}'$ is time-dependent since it depends on $\mathbf{u}_h$, while in the dynamic we introduce an additional ODE to track its evolution in time.

To derive this ODE from the Navier-Stokes equations, we plug (2.8) in (2.1) and we substitute in equation (2.1) and (2.2) $\{\mathbf{u}, p\} = \{\mathbf{u}' + \mathbf{u}^h, p' + p^h\}$ [13, 14]. Moving the coarse scale terms on right-hand-side, we obtain:

$$\rho\frac{\partial \mathbf{u}'}{\partial t} + \rho(\mathbf{u}^h + \mathbf{u}') \cdot \nabla \mathbf{u}' - \mu \Delta \mathbf{u}' + \nabla p' = -\mathbf{r}_M(\mathbf{u}^h, p^h) \tag{2.52}$$

$$\nabla \cdot \mathbf{u}' = -r_C(\mathbf{u}^h) \tag{2.53}$$

Where r_C and $\mathbf{r}_M$ are the residuals of the finite element solutions $\mathbf{u}_h$ and p_h, projected on the spaces of the subscale [14]:

$$\mathbf{r}_M(\mathbf{u}^h, p^h) = \left(\rho\frac{\partial \mathbf{u}^h}{\partial t} + \rho(\mathbf{u}^h + \mathbf{u}') \cdot \nabla \mathbf{u}^h - \mu\Delta\mathbf{u}^h + \nabla p^h - \mathbf{f}\right) \tag{2.54}$$

$$r_C(\mathbf{u}^h) = \nabla \cdot \mathbf{u}^h \tag{2.55}$$

From equations (2.52) and (2.53) we extrapolate the matrix formulation:

$$\mathbf{M}\partial_t \mathbf{U}' + \widetilde{\mathcal{L}}(\mathbf{U}') = \mathbf{R}(\mathbf{U}^h) \tag{2.56}$$

where $\mathbf{M} = \text{diag}(\mathbf{I}_d, 0)$, $\mathbf{R}(\mathbf{U}^h)$ is the residuals vector as in (2.43) and $\widetilde{\mathcal{L}}$ is defined in an analogous fashion as $\mathcal{L}$ in the quasi-static, such that:

$$\widetilde{\mathcal{L}}(\mathbf{U}^h) = \begin{bmatrix} \widetilde{\tau_M}(\mathbf{u}^h)\mathbf{I}_{3\times 3} & \mathbf{0}_{3\times 3} \\ \mathbf{0}^T_{3\times 3} & \widetilde{\tau_C}(\mathbf{u}^h) \end{bmatrix} \tag{2.57}$$

Hence, performing calculations:

$$\frac{\partial \mathbf{u}'}{\partial t} + (\widetilde{\tau}_M(\mathbf{u}^h))^{-1}\mathbf{u}' \approx -\mathbf{r}_M(\mathbf{u}^h, p^h), \tag{2.58}$$

$$(\widetilde{\tau}_C(\mathbf{u}^h))^{-1}p' \approx -r_C(\mathbf{u}^h). \tag{2.59}$$

According to the Fourier analysis proposed by R. Codina et al. [13], we define the two stabilization parameters denoted with a $\widetilde{\ }$ as:

$$\widetilde{\tau}_M(\mathbf{u}^h) = (\rho^2\mathbf{u}^h \cdot \overline{\mathbf{G}}\mathbf{u}^h + Cr\mu^2\overline{\mathbf{G}} : \overline{\mathbf{G}})^{-\frac{1}{2}} \tag{2.60}$$

$$\widetilde{\tau}_C(\mathbf{u}^h) = (\widetilde{\tau}_M(\mathbf{u}^h)\overline{\mathbf{g}} \cdot \overline{\mathbf{g}})^{-1} \tag{2.61}$$

We will show in section 2.4 that the VMS-LES fully-discrete dynamic problem can be derived from the quasi-static, by simple substituting τ_C and τ_M with $\widetilde{\tau}_C$ and $\widetilde{\tau}_M$.

2.4. Time discretization

The time-discretization of the problem is formulated according to the Backward Differentiation Formulas (BDF) of order σ_t, which are a family of implicit single-stage methods. Implicit methods are more expensive if compared to explicit ones. However, thanks to the fact that BDF is a single-stage method, it is capable to increase computational efforts only marginally, when increasing accuracy [35].

Let us consider a partition of the time domain $(0, T)$ into N_t subintervals of equal size $\Delta t = \frac{T}{N_t}$. The subscript n denote the time-step, such that $n = 0, ..., N_t$ and $\Delta t = t^{n+1} - t^n$.
We write the approximation of the time-derivative as:

$$\frac{\partial \mathbf{u}^h}{\partial t} \approx \frac{\alpha_{\mathrm{BDF}} \mathbf{u}^h_{n+1} - \mathbf{u}^h_{n,\mathrm{BDF}}}{\Delta t} \tag{2.62}$$

such that, the temporal derivative is replaced with a difference quotient: at the numerator we have the increment of $\mathbf{u}^h$, where $\mathbf{u}^h_n$ is the quantity $\mathbf{u}^h$ evaluated at the time t^n; at the denominator we have the timestep.

We define $\mathbf{u}^h_{n,\mathrm{BDF}}$ as follow:

$$\mathbf{u}^h_{n,\mathrm{BDF}} = \begin{cases} \mathbf{u}^h_n & \text{if } n \geq 1, \text{ for } \sigma_t = 1 \text{ (BDF1)} \\ 2\mathbf{u}^h_n - \frac{1}{2}\mathbf{u}^h_{n-1} & \text{if } n \geq 2, \text{ for } \sigma_t = 2 \text{ (BDF2)} \\ 3\mathbf{u}^h_n - \frac{3}{2}\mathbf{u}^h_{n-1} + \frac{1}{3}\mathbf{u}^h_{n-2} & \text{if } n \geq 3, \text{ for } \sigma_t = 3 \text{ (BDF3)} \end{cases} \tag{2.63}$$

and α_{BDF}:

$$\alpha_{\mathrm{BDF}} = \begin{cases} 1 & \text{for } \sigma_t = 1 \text{ (BDF1)} \\ \frac{3}{2} & \text{for } \sigma_t = 2 \text{ (BDF2)} \\ \frac{11}{6} & \text{for } \sigma_t = 3 \text{ (BDF3)} \end{cases} \tag{2.64}$$

Therefore, BDF1 corresponds to the the Backward Euler scheme.
To discretize in time the non-linear term of the Navier-Stokes equations with the BDF approximation, we can choose between three different schemes: fully-implicit, semi-implicit and explicit [39]. In this work, we analyse the fully-implicit and the semi-implicit methods.
In the first approach we let the non-linear term and all other terms deriving from it, depending on $\mathbf{u}^{n+1}_h$ and p^{n+1}_h, such that:

$$(\mathbf{v}^h, \rho(\mathbf{u}^h_{n+1} \cdot \nabla)\mathbf{u}^h_{n+1}) \tag{2.65}$$

To compute the solution with the implicit method we need to solve a non-linear system and to perform many iterations, which require high computational costs [42]. Moreover, we cannot prove the uniqueness of the solution [7]. However, this approach is unconditionally stable, which implies that we do not have any restriction on the choice of the timestep to preserve stability.

In the semi-implicit method, we linearize the convective term and all related terms with the following scheme:

$$(\mathbf{v}^h, \rho(\mathbf{u}_n^h \cdot \nabla)\mathbf{u}_{n+1}^h) \tag{2.66}$$

This method improves the aforementioned criticalities of the implicit, however, it is only conditionally stable [39]. As a matter of fact, in this work, we do not mention the stability condition, since no stability issues arise.
We discuss all details of implicit and semi-implicit formulations in the next two paragraphs.

2.4.1. Fully-implicit, BDF formulation

To get the fully-discrete fully-implicit BDF VMS-LES formulation of the Navier-Stokes equations, we plug (2.62) in the semi-discrete problem. We combine with this scheme either the quasi-static or the dynamic discretization of fine scale. In the following list we resume the two formulations.

- *Quasi − static*, fully-implicit scheme:
  By plugging (2.62) in (2.51) we obtain a VMS-LES fully-discrete, fully-implicit BDF formulation of the Navier-Stokes equations with a quasi-static approximation of fine scales:

  given $\mathbf{u}_n^h, ..., \mathbf{u}_{n+1-\sigma_t}^h$, for any $n = 0, ..., N_t - 1$, find $(\mathbf{u}_{n+1}^h, p_{n+1}^h) \in \mathcal{V}_g^h \times \mathcal{Q}^h$ such that:

$$\begin{aligned}
&(\mathbf{v}^h, \rho\frac{\alpha_{\text{BDF}}\mathbf{u}_{n+1}^h - \mathbf{u}_{n,\text{BDF}}^h}{\Delta t}) + (\mathbf{v}^h, \rho(\mathbf{u}_{n+1}^h \cdot \nabla)\mathbf{u}_{n+1}^h) + (\nabla\mathbf{v}^h, \mu\nabla\mathbf{u}_{n+1}^h) \\
&-(\nabla \cdot \mathbf{v}^h, p_{n+1}^h) + (q^h, \nabla \cdot \mathbf{u}_{n+1}^h) \\
&+(\rho\mathbf{u}_{n+1}^h \cdot \nabla\mathbf{v}^h + \nabla q^h, \tau_M(\mathbf{u}_{n+1}^h)\mathbf{r}_M(\mathbf{u}_{n+1}^h, p_{n+1}^h)) \\
&+(\nabla \cdot \mathbf{v}^h, \tau_C(\mathbf{u}_{n+1}^h) r_C(\mathbf{u}_{n+1}^h)) \\
&+(\rho\mathbf{u}_{n+1}^h \cdot (\nabla\mathbf{v}^h)^T, \tau_M(\mathbf{u}_{n+1}^h)\mathbf{r}_M(\mathbf{u}_{n+1}^h, p_{n+1}^h)) \\
&-(\rho\nabla\mathbf{v}^h, \tau_M(\mathbf{u}_{n+1}^h)\mathbf{r}_M(\mathbf{u}_{n+1}^h, p_{n+1}^h) \otimes \tau_M(\mathbf{u}_{n+1}^h)\mathbf{r}_M(\mathbf{u}_{n+1}^h, p_{n+1}^h)) \\
&= (\mathbf{v}^h, \mathbf{f}_{n+1}) + (\mathbf{v}^h, \mathbf{h}_{n+1})_{\Gamma_N}
\end{aligned} \tag{2.67}$$

  for all $(\mathbf{v}^h, q^h) \in \mathcal{V}_0^h \times \mathcal{Q}^h$, for all $n \geq \sigma_t - 1$
  Notice that the expression of stabilization parameters and residuals follow definitions

(2.50)-(2.49) and (2.48)-(2.47) respectively, and the two terms on the right hand side of the equation are $\mathbf{f}_{n+1} = \mathbf{f}(t_{n+1})$ and $\mathbf{h}_{n+1} = \mathbf{h}(t_{n+1})$

- *Dynamic*, fully-implicit scheme:
  As introduced in section section 2.3.2, the dynamic approach is characterised by an additional ODE to define the fine scale velocity. To solve it in time, we apply the same scheme of equation (2.62) to $\mathbf{u}'$:

$$\frac{\partial \mathbf{u}'}{\partial t} \approx \frac{\alpha_{\mathrm{BDF}}\mathbf{u}'_{n+1} - \mathbf{u}'_{n,\mathrm{BDF}}}{\Delta t} \tag{2.68}$$

and we plug it in (2.58):

$$\rho\frac{\alpha_{\mathrm{BDF}}\mathbf{u}'_{n+1} - \mathbf{u}'_{n,\mathrm{BDF}}}{\Delta t} + (\widetilde{\tau_M}(\mathbf{u}^h_{n+1}))^{-1}\mathbf{u}'_{n+1} = -\mathbf{r}_M(\mathbf{u}^h_{n+1}, p^h_{n+1}) \tag{2.69}$$

explicating $\mathbf{u}'_{n+1}$ we have:

$$\mathbf{u}'_{n+1} = -\left(\rho\frac{\alpha_{BDF}}{\Delta t} + \frac{1}{\widetilde{\tau_M}(\mathbf{u}^h_{n+1})}\right)^{-1}\left(\mathbf{r}_M(\mathbf{u}^h_{n+1}, p^h_{n+1}) - \frac{\rho}{\Delta t}\mathbf{u}'_{n,BDF}\right) \tag{2.70}$$

To obtain the same structure of the quasi-static scheme the two factors of the previous equation are defined as:

$$\breve{\tau_M}(\mathbf{u}^h_{n+1}) = \left(\rho\frac{\alpha_{BDF}}{\Delta t} + \frac{1}{\widetilde{\tau_M}(\mathbf{u}^h_{n+1})}\right)^{-1} \tag{2.71}$$

$$\breve{\mathbf{r}_M}(\mathbf{u}^h_{n+1}, p^h_{n+1}, \mathbf{u}'_{n,BDF}) = \mathbf{r}_M(\mathbf{u}^h_{n+1}, p^h_{n+1}) - \frac{\rho}{\Delta t}\mathbf{u}'_{n,BDF} \tag{2.72}$$

Plugging this two definitions in equation (2.70) $\mathbf{u}'_{n+1}$ and p'_{n+1} become:

$$\mathbf{u}'_{n+1}(\mathbf{u}^h_{n+1}, p^h_{n+1}, \mathbf{u}'_{n,BDF}) = -\breve{\tau_M}(\mathbf{u}^h_{n+1})\breve{\mathbf{r}_M}(\mathbf{u}^h_{n+1}, p^h_{n+1}, \mathbf{u}'_{n,BDF}) \tag{2.73}$$

$$p'_{n+1}(\mathbf{u}^h_{n+1}) = -\widetilde{\tau_C}(\mathbf{u}^h_{n+1})\mathbf{r}_C(\mathbf{u}^h_{n+1}) \tag{2.74}$$

The final formulation is obtained in the same way as (2.67) by simple substituting the definition of $\mathbf{u}'_{n+1}$ with equation (2.70):
given $\mathbf{u}^h_n, ..., \mathbf{u}^h_{n+1-\sigma_t}$ and $\mathbf{u}'_n, ..., \mathbf{u}'_{n+1-\sigma_t}$, for any $n = 0, ..., N_t - 1$, find $(\mathbf{u}^h_{n+1}, p^h_{n+1}) \in$

$V_g^h \times Q^h$ such that:

$$\begin{aligned}
&\left(\mathbf{v}^h, \rho\frac{\alpha_{BDF}\mathbf{u}_{n+1}^h - \mathbf{u}_{n,BDF}^h}{\Delta t}\right) + (\mathbf{v}^h, \rho(\mathbf{u}_{n+1}^h \cdot \nabla)\mathbf{u}_{n+1}^h) + (\nabla\mathbf{v}^h, \mu\nabla\mathbf{u}_{n+1}^h) \\
&-(\nabla\cdot\mathbf{v}^h, p_{n+1}^h) + (q^h, \nabla\cdot\mathbf{u}_{n+1}^h) \\
&+(\rho\mathbf{u}_{n+1}^h\cdot\nabla\mathbf{v}^h + \nabla q^h, \breve{\tau_M}(\mathbf{u}_{n+1}^h)\mathbf{r}_M(\mathbf{u}_{n+1}^h, p_{n+1}^h)) \\
&+(\nabla\cdot\mathbf{v}^h, \widetilde{\tau}_C(\mathbf{u}_{n+1}^h)r_C(\mathbf{u}_{n+1}^h)) \\
&+(\rho\mathbf{u}_{n+1}^h\cdot(\nabla\mathbf{v}^h)^T, \breve{\tau_M}(\mathbf{u}_{n+1}^h)\breve{\mathbf{r}_M}(\mathbf{u}_{n+1}^h, p_{n+1}^h)) \\
&-(\rho\nabla\mathbf{v}^h, \breve{\tau_M}(\mathbf{u}_{n+1}^h)\breve{\mathbf{r}_M}(\mathbf{u}_{n+1}^h, p_{n+1}^h)\otimes\breve{\tau_M}(\mathbf{u}_{n+1}^h)\breve{\mathbf{r}_M}(\mathbf{u}_{n+1}^h, p_{n+1}^h)) \\
&= (\mathbf{v}^h, \mathbf{f}_{n+1}) + (\mathbf{v}^h, \mathbf{h}_{n+1})_{\Gamma_N}
\end{aligned} \tag{2.75}$$

for all $(\mathbf{v}^h, q^h) \in V_o^h \times Q^h$, for all $n \geq \sigma_t - 1$

2.4.2. Semi-implicit, BDF formulation

To define the semi-implicit, BDF formulation we start from the fully-implicit scheme by treating the non-linear terms according to the Newton-Gregory approach. Without entering in details, for which we address to [12], the non-linear term is linearized through the so called "extrapolated variables" denoted as '$n+1$,EXT. These comes from the Newton-Gregory backwards polynomials with a degree equal to the time-discretisation order, such that:

$$u_{n+1,EXT}^h = \begin{cases} u_n^h & \text{if } n \geq 0, \text{ for } \sigma_t = 1 \text{ (BDF1)} \\ 2u_n^h - u_{n-1}^h & \text{if } n \geq 2, \text{ for } \sigma_t = 2 \text{ (BDF2)} \\ 3u_n^h - 3u_{n-1}^h + u_{n-2}^h & \text{if } n \geq 3, \text{ for } \sigma_t = 3 \text{ (BDF3)} \end{cases} \tag{2.76}$$

$$p_{n+1,EXT}^h = \begin{cases} p_n^h & \text{if } n \geq 0, \text{ for } \sigma_t = 1 \text{ (BDF1)} \\ 2p_n^h - p_{n-1}^h & \text{if } n \geq 2, \text{ for } \sigma_t = 2 \text{ (BDF2)} \\ 3p_n^h - 3p_{n-1}^h + p_{n-2}^h & \text{if } n \geq 3, \text{ for } \sigma_t = 3 \text{ (BDF3)} \end{cases} \tag{2.77}$$

As for the implicit scheme, we resume in the following list the two methods analysed in next chapters.

- *Quasi − static*, semi-implicit scheme:

given $\mathbf{u}_n^h, ..., \mathbf{u}_{n+1-\sigma_t}^h$, for any $n = 0, ..., N_t - 1$, find $(\mathbf{u}_{n+1}^h, p_{n+1}^h) \in \mathcal{V}_g^h \times \mathcal{Q}^h$ such

that:

$$\begin{aligned}
&\left(\mathbf{v}^h, \rho\frac{\alpha_{BDF}\mathbf{u}^h_{n+1} - \mathbf{u}^h_{n,BDF}}{\Delta t}\right) + (\mathbf{v}^h, \rho(\mathbf{u}^h_{n+1,EXT}\cdot\nabla)\mathbf{u}^h_{n+1}) + (\nabla\mathbf{v}^h, \mu\nabla\mathbf{u}^h_{n+1})\\
&-(\nabla\cdot\mathbf{v}^h, p^h_{n+1}) + (q^h, \nabla\cdot\mathbf{u}^h_{n+1})\\
&+(\rho\mathbf{u}^h_{n+1,EXT}\cdot\nabla\mathbf{v}^h + \nabla q^h, \tau_M(\mathbf{u}^h_{n+1,EXT})\mathbf{r}_{M,EXT}(\mathbf{u}^h_{n+1}, p^h_{n+1}))\\
&+(\nabla\cdot\mathbf{v}^h, \tau_C(\mathbf{u}^h_{n+1,EXT})r_C(\mathbf{u}^h_{n+1}))\\
&+(\rho\mathbf{u}^h_{n+1,EXT}\cdot(\nabla\mathbf{v}^h)^T, \tau_M(\mathbf{u}^h_{n+1,EXT})\mathbf{r}_{M,EXT}(\mathbf{u}^h_{n+1}, p^h_{n+1}))\\
&-(\rho\nabla\mathbf{v}^h, \tau_M(\mathbf{u}^h_{n+1,EXT})\mathbf{r}^{LHS}_{M,EXT}(\mathbf{u}^h_{n+1}, p^h_{n+1})...\\
&... \otimes \tau_M(\mathbf{u}^h_{n+1,EXT})\mathbf{r}_{M,EXT}(\mathbf{u}^h_{n+1,EXT}, p^h_{n+1,EXT}))\\
&+(\rho\nabla\mathbf{v}^h, \tau_M(\mathbf{u}^h_{n+1,EXT})\mathbf{r}^{RHS}_{M} \otimes \tau_M(\mathbf{u}^h_{n+1,EXT})\mathbf{r}_{M,EXT}(\mathbf{u}^h_{n+1}, p^h_{n+1}))\\
&= (\mathbf{v}^h, \mathbf{f}_{n+1}) + (\mathbf{v}^h, \mathbf{h}_{n+1})_{\Gamma_N}
\end{aligned} \tag{2.78}$$

for all $(\mathbf{v}^h, q^h) \in \mathcal{V}^h_0 \times \mathcal{Q}^h$, for all $n \geq \sigma_t - 1$
where we define the extrapolated residual as:

$$\begin{aligned}
\mathbf{r}_{M,EXT}(\mathbf{u}^h_{n+1}, p^h_{n+1}) = \rho\left(\frac{\alpha_{BDF}\mathbf{u}^h_{n+1} - \mathbf{u}^h_{n,BDF}}{\Delta t} + \mathbf{u}^h_{n+1,EXT}\cdot\nabla\mathbf{u}^h_{n+1}\right) +\\
\nabla p^h_{n+1} - \mu\Delta\mathbf{u}^h_{n+1} - \mathbf{f}_{n+1}
\end{aligned} \tag{2.79}$$

$$\mathbf{r}_{M,EXT}(\mathbf{u}^h_{n+1}, p^h_{n+1}) = \mathbf{r}^{LHS}_{M,EXT}(\mathbf{u}^h_{n+1}, p^h_{n+1}) - \mathbf{r}^{RHS}_{M} \tag{2.80}$$

$$\begin{aligned}
\mathbf{r}^{LHS}_{M,EXT}(\mathbf{u}^h_{n+1}, p^h_{n+1}) = \rho\left(\frac{\alpha_{BDF}\mathbf{u}^h_{n+1}}{\Delta t} + \mathbf{u}^h_{n+1,EXT}\cdot\nabla\mathbf{u}^h_{n+1}\right) +\\
\nabla p^h_{n+1} - \mu\Delta\mathbf{u}^h_{n+1}
\end{aligned} \tag{2.81}$$

$$\mathbf{r}^{RHS}_{M,EXT} = \rho\frac{\mathbf{u}^h_{h,BDF}}{\Delta t} + \mathbf{f}_{n+1} \tag{2.82}$$

while the stabilization parameters for this formulation are:

$$\tau_M(\mathbf{u}^h_{n+1,EXT}) = \left(\frac{\sigma_t^2\rho^2}{\Delta t^2} + \rho^2\mathbf{u}^h_{n+1,EXT}\cdot\overline{\mathbf{G}}\mathbf{u}^h_{n+1,EXT} + Cr\mu^2\overline{\mathbf{G}} : \overline{\mathbf{G}}\right)^{-\frac{1}{2}} \tag{2.83}$$

$$\tau_C(\mathbf{u}^h_{n+1,EXT}) = (\tau_M(\mathbf{u}^h_{n+1,EXT})\overline{\mathbf{g}}\cdot\overline{\mathbf{g}})^{-1} \tag{2.84}$$

- *Dynamic*, semi-implicit scheme

given $\mathbf{u}_n^h, ..., \mathbf{u}_{n+1-\sigma_t}^h$, for any $n = 0, ..., N_t - 1$, find $(\mathbf{u}_{n+1}^h, p_{n+1}^h) \in \mathcal{V}_g^h \times \mathcal{Q}^h$ such that:

$$
\begin{aligned}
&\left(\mathbf{v}^h, \rho\frac{\alpha_{BDF}\mathbf{u}_{n+1}^h - \mathbf{u}_{n,BDF}^h}{\Delta t}\right) + (\mathbf{v}^h, \rho(\mathbf{u}_{n+1,EXT}^h \cdot \nabla)\mathbf{u}_{n+1}^h) + (\nabla\mathbf{v}^h, \mu\nabla\mathbf{u}_{n+1}^h)\\
&-(\nabla\cdot\mathbf{v}^h, p_{n+1}^h) + (q^h, \nabla\cdot\mathbf{u}_{n+1}^h)\\
&+(\rho\mathbf{u}_{n+1,EXT}^h\cdot\nabla\mathbf{v}^h + \nabla q^h, \breve{\tau}_M(\mathbf{u}_{n+1,EXT}^h)\breve{\mathbf{r}}_{M,EXT}(\mathbf{u}_{n+1}^h, p_{n+1}^h; \mathbf{u}'_{n,BDF}))\\
&+(\nabla\cdot\mathbf{v}^h, \widetilde{\tau}_C(\mathbf{u}_{n+1,EXT}^h) r_C(\mathbf{u}_{n+1}^h))\\
&+(\rho\mathbf{u}_{n+1,EXT}^h\cdot(\nabla\mathbf{v}^h)^T, \breve{\tau}_M(\mathbf{u}_{n+1,EXT}^h)\breve{\mathbf{r}}_{M,EXT}(\mathbf{u}_{n+1}^h, p_{n+1}^h; \mathbf{u}'_{n,BDF}))\\
&-(\rho\nabla\mathbf{v}^h, \breve{\tau}_M(\mathbf{u}_{n+1,EXT}^h)\breve{\mathbf{r}}_{M,EXT}^{LHS}(\mathbf{u}_{n+1}^h, p_{n+1}^h)...\\
&... \otimes \breve{\tau}_M(\mathbf{u}_{n+1,EXT}^h)\breve{\mathbf{r}}_{M,EXT}(\mathbf{u}_{n+1,EXT}^h, p_{n+1,EXT}^h; \mathbf{u}'_{n,BDF}))\\
&+(\rho\nabla\mathbf{v}^h, \breve{\tau}_M(\mathbf{u}_{n+1,EXT}^h)\mathbf{r}_M^{RHS} \otimes \breve{\tau}_M(\mathbf{u}_{n+1,EXT}^h)\breve{\mathbf{r}}_{M,EXT}(\mathbf{u}_{n+1}^h, p_{n+1}^h; \mathbf{u}'_{n,BDF}))\\
&= (\mathbf{v}^h, \mathbf{f}_{n+1}) + (\mathbf{v}^h, \mathbf{h}_{n+1})_{\Gamma_N}
\end{aligned}
\tag{2.85}
$$

for all $(\mathbf{v}^h, q^h) \in \mathcal{V}_0^h \times \mathcal{Q}^h$, for all $n \geq \sigma_t - 1$

where we define the residuals denoted with $\breve{}$ as:

$$
\begin{aligned}
\breve{\mathbf{r}}_{M,EXT}(\mathbf{u}_{n+1}^h, p_{n+1}^h; \mathbf{u}'_{n,BDF}) &= \mathbf{r}_{M,EXT}(\mathbf{u}_{n+1}^h, p_{n+1}^h) - \rho\frac{\mathbf{u}'_{n,BDF}}{\Delta t} \\
&= \mathbf{r}_{M,EXT}^{LHS}(\mathbf{u}_{n+1}^h, p_{n+1}^h) - \breve{\mathbf{r}}_{M,EXT}^{RHS}(\mathbf{u}'_{n,BDF})
\end{aligned}
\tag{2.86}
$$

$$
\breve{\mathbf{r}}_{M,EXT}^{RHS}(\mathbf{u}'_{n,BDF}) = \rho\frac{\mathbf{u}_{n,BDF}^h + \mathbf{u}'_{n,BDF}}{\Delta t} + \mathbf{f}_{n+1} \tag{2.87}
$$

Notice that the term $\mathbf{r}_{M,EXT}^{LHS}(\mathbf{u}_{n+1}^h, p_{n+1}^h)$ is equal to the residual of the quasi-static approach, indeed, the contribute of the dynamic is only in the RHS part.
Finally, the stabilization parameters for this method are:

$$
\widetilde{\tau}_M(\mathbf{u}_{n+1,EXT}^h) = \left(\rho^2\mathbf{u}_{n+1,EXT}^h\cdot\overline{\mathbf{G}}\mathbf{u}_{n+1,EXT}^h + Cr\mu^2\overline{\mathbf{G}}:\overline{\mathbf{G}}\right)^{-\frac{1}{2}} \tag{2.88}
$$

$$
\widetilde{\tau}_C(\mathbf{u}_{n+1,EXT}^h) = (\widetilde{\tau}_M(\mathbf{u}_{n+1,EXT}^h)\overline{\mathbf{g}}\cdot\overline{\mathbf{g}})^{-1} \tag{2.89}
$$

$$
\breve{\tau_M}(\mathbf{u}_{n+1,EXT}^h) = \left(\rho\frac{\alpha_{BDF}}{\Delta t} + \frac{1}{\widetilde{\tau_M}(\mathbf{u}_{n+1,EXT}^h)}\right)^{-1} \tag{2.90}
$$

In this chapter, we have formally introduced all the methods under investigation, the next step is to compare their performance on a test case.

3 | Verification of the computational model

In this section, we present the verification of the quasi-static and dynamic VMS-LES methods, which aim is to check their correct implementation and consistency. Verification is a purely mathematical activity, not concerning about the accuracy of physical laws. Therefore, a benchmark problem for software verification is a problem whose exact solution is known, and it is not necessarily correlated with real applications. There are two different families of analytical solutions, which can be used for verification: the first ones come from the Method of Manufactured Solutions, where an exact solution is created *ad hoc* to have the same level of complexity of real applications; the second ones are those of simple test-cases described in literature, where the solution can be computed analytically [41]. In this work we follow the second approach, employing the so called "Beltrami-flow" test-case.
This chapter is organized as follows: in Section 3.1, we define the fluid dynamics benchmark, in Sections 3.2 and 3.3, we introduce the software and we present the numerical setup of the simulations, in Section 3.4, we discuss convergence and in Section 3.5, we analyse consistency.

3.1. Beltrami-flow test case

The "Beltrami-flow" benchmark was named after the Italian mathematician Eugenio Beltrami (1835-1900). He was the theoriser of the Beltrami vector field, a particular field where a vector and its curl are parallel. This concept was introduced in fluid dynamic in 1881 by the Russian scientist Ippolit S. Gromeka to represent a flow with a Lamb vector[1] equal to zero, since velocity and vorticity are parallel.
Ross Ethier and Steinman (1994) [18] solved the Beltrami-Flow test case analytically, defining an analytical solution for a three-dimensional decaying vortex, with the same derivation used for the two-dimensional Taylor decaying vortex.

[1] The cross product of vorticity vector and velocity vector of the flow field

Let us introduce the problem.
We define Ω the domain of the Beltrami-Flow test-case, which is a cube centred in (0,0,0), extending for one unit in every direction:

$$\Omega = (-1, 1)^3$$

We define the boundary of the domain as $\partial\Omega = \Gamma_D \cup \Gamma_N$ where Γ_D is the portion of the boundary where Dirichlet boundary conditions (BCs) applies, while on Γ_N we have Neumann boundary data.
The Beltrami-Flow needs both conditions, indeed, on five of the six faces of the cube we apply homogeneous Dirichlet BCs, while on the remaining face we set a non-homogeneous Neumann boundary data. Since $p \in L^2(\Omega)$, we have a non-zero pressure field on the boundaries, there must be at least one Neumann condition, to ensure the uniqueness of p [39]. Moreover, thanks to this configuration of BCs we avoid some wiggles arising in the velocity solution next to the boundaries [18].
In our model we consider the following Γ_D and Γ_N :

$$\begin{cases} \Gamma_D &= (x, y, -1) \cup (x, 1, z) \cup (x, -1, z) \cup (1, y, z) \cup (-1, y, z) \\ \Gamma_N &= (x, y, 1) \end{cases} \tag{3.1}$$

Since it is not the aim of this work to describe the full derivation of the solution, for which we address to [18, 38, 45], we report the exact solution without performing any computation.
By denoting with u, v, w the three velocity components and p the pressure:

$$u = -a[e^{ax}\sin(ay \pm dz) + e^{az}\cos(ax \pm dy)]e^{-d^2t} \tag{3.2}$$

$$v = -a[e^{ay}\sin(az \pm dx) + e^{ax}\cos(ay \pm dz)]e^{-d^2t} \tag{3.3}$$

$$w = -a[e^{az}\sin(ax \pm dy) + e^{ay}\cos(az \pm dx)]e^{-d^2t} \tag{3.4}$$

$$\begin{aligned} p = -\frac{a^2}{2}[&e^{2ax} + e^{2ay} + e^{2az} + 2\sin(ax \pm dy)\cos(az \pm dx)e^{a(y+z)} + \\ &+2\sin(ay \pm dz)\cos(ax \pm dy)e^{a(x+z)} + \\ &+2\sin(az \pm dx)\cos(ay \pm dz)e^{a(y+x)}]e^{-d^2t} \end{aligned} \tag{3.5}$$

where the coefficients are set as $a = \pi/4$ and $b = \pi/2$.
Notice that, both initial conditions and Dirichlet boundary data are set as equations (3.2)-(3.5) evaluated at $t = 0$ on the respective face.

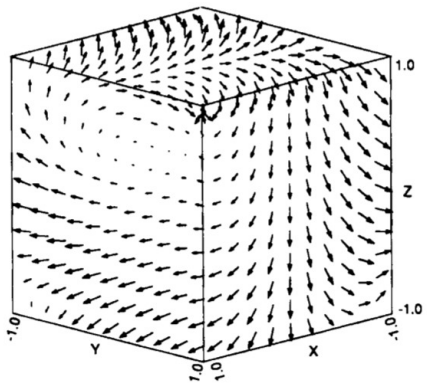


Figure 3.1: Verification. Beltrami-Flow field

3.2. Introduction to the Software

Before entering into details of test case's numerical setup, let us introduce the software. Numerical simulations are performed in `life`^{x}, an open source library for high performance finite element simulations of multiphysics, multiscale and multidomain problems, written in C++ and based on deal.II [2, 4]. The Beltrami flow, as well as the VMS-LES methods, are implemented in a specific branch of this library.

3.3. Numerical setup

In this section we introduce the numerical setup for the verification of the quasi-static and dynamic VMS-LES methods. For this purpose, we remind that we perform two analysis:

- a test modifying the mesh refinement h, fixing the time-step Δt (Section 3.4);
- a test varying the time-step size fixing the mesh refinement(Section 3.5).

The first one is a convergence test, while the second one aims to check consistency and stability.
We define the numerical setup of the simulations, trough a trivial approach, by changing one-by-one the parameters, which have an impact on the solution. We choose the configuration with the best compromise between computational cost and accuracy, i.e. when a parameter does not significantly improve the solution, we take the value which reduces, as much as possible, the simulation time; when a parameter significantly improves the solution without increasing too much the time of resolution, we take the value with the most accurate result.
In table 3.1 we report our choices.

Final time (T)	0.2s
Reynolds number (Re)	1 - 1000
FEM spaces	$\mathbb{Q}1 - \mathbb{Q}1$
BDF order	1
Stopping criterion based on residual	relative, 10^{-8}

Table 3.1: Verification. List of changing parameters

$$Re = \frac{\rho u L}{\mu} \tag{3.6}$$

Remarks

(1) By changing the Reynolds number (3.6) we change the flow regime: according to literature, when $Re = 1$ the Beltrami-flow is in laminar condition, while for $Re = 1000$ it is turbulent [3]. To get an exhaustive overview of the method under investigation, we may consider both cases. Therefore, for laminar conditions, we fix viscosity and density as $\mu = 1$ and $\rho = 1$, respectively and we change Re multiplying ρ by a factor of 10^3, according to equation (3.6), where $L = 1$.

(2) For the space discretization of the coarse scales, we use polynomial FE functions of degree one for both velocity and pressure ($\mathbb{Q}1 - \mathbb{Q}1$). The attentive reader should noticed that, two linear spaces are not stable (i.e. LBB-stable[2]) for the Navier-Stokes equations, since they do not satisfy the inf-sup condition [10]. However, the VMS-LES method is able to control these instabilities [44, 50].

(3) For the time discretization, we test both implicit and semi-implicit schemes. The first case is unconditionally stable, while the second one is only conditionally stable. Therefore the time-step has to be chosen sufficiently low, such that no stability issues arise [39]:

$$\Delta t \leq C \frac{h}{max_\Omega |\mathbf{u}^n|} \tag{3.7}$$

[2]Ladyzhenskaya–Babuška–Brezzi condition [10]

3.4. Mesh convergence analysis

The first aim is to find out whether models are converging reducing the mesh-size and with which rate. We chose three number of refinements (N^{Ref}), which lead to different size (h) and number (N^{el}) of elements, as reported in Table 3.2. We fix the time-step as $\Delta t = 5 \cdot 10^{-3}$, which has shown to be sufficiently low, such that neither instability arises.

N^{Ref}	h	N^{el}
2	0.25	64
3	0.125	512
4	0.0625	4096

Table 3.2: Verification. Mesh overview

To evaluate the convergence of the methods, we analyse pressure and velocity errors.
In particular, let us consider f the generic reference (i.e. exact) solution for either pressure or velocity and f_h the finite element solution, we define the error e such that:

$$e = f - f_h \tag{3.8}$$

We remind that in the VMS-LES formulation the solution f_h includes the effect of the fine scale, which in turns depends on the coarse scale.
The L_2 and H_1 errors are than computed as:

$$\|e\|_{L_2} = \sqrt{\int_\Omega e^2} \tag{3.9}$$

$$\|e\|_{H_1} = \left(\|e\|_{L_2}^2 + \|\nabla \cdot e\|_{L_2}^2\right)^{\frac{1}{2}} \tag{3.10}$$

As the coarse scale is approximated with bilinear polynomial functions and it is stabilized with SUPG/PSPG method, we expect an optimal convergence for velocity errors, which means a convergence of order two for the velocity L^2 errors and order one for the velocity H^1, according to the $a - priori$ error estimator of FEM [39]. For the sake of clarity, we

report the error estimator for the SUPG formulation and we address to [24] for the proof.

$$\mu||\mathbf{u}-\mathbf{u}_h||^2+\sum_{k=1}^{N_{el}}\frac{\alpha^k(h^k)^2}{4\mu}||\nabla(p-p_h)||^2_{\Omega^k}\leq \tag{3.11}$$
$$2\mu C_1 h^{2r}||\mathbf{u}^2||_{r+1}\quad+\left(\frac{C_2}{2\mu}\right)h^{2(l+1)}||p||^2_{l+1}$$

The constant α depends on the elements type (tetrahedra or hexahedra), C_1 and C_2 are non-dimensional parameters independent of the exact solution, while r and l are the polynomial degrees of velocity and pressure approximating functions, respectively.
As it can be observed in (3.11), we cannot bound pressure errors, since the estimator is only bounding the gradient of pressure. Therefore, if $r = l$ we expect a sub-optimal convergence rate for the L_2 pressure errors [24].

3.4.1. Implicit BDF

The convergence tests for the VMS-LES implicit method, with dynamic and quasi-static approximation of fine scale are reported in fig. 3.2.
Regarding velocity, the quasi-static is behaving as expected, indeed L^2 and H^1 errors are converging with order two and one, respectively. About the dynamic, we notice that, we obtain the same rate of convergence of the quasi-static, and negligible differences in the magnitude ($\leq 0.1\%$).
Pressure L^2 errors are behaving differently. First we analyse the dynamic method, then the quasi-static one.
The dynamic case has optimal convergence (i.e. $\mathcal{O}(h^2)$) for high Reynolds and sub-optimal (i.e. $\mathcal{O}(h)$) for $Re = 1$. A possible interpretation of this result is that, to conserve optimal convergence, the stabilization parameters may require a different dependence on h for advection- or convection-dominated regimes [23], something which in our formulation was neglected.
Regarding the quasi-static formulation, we highlight that: at low Reynolds number, the rate of convergence is not yet constant, hence further refinements would be necessary to precisely evaluate the trend; for higher Reynolds, it seems to be super-convergent, but relative errors are much greater than the dynamic case (around 4 to 25 times).
Before concluding this section, we want to introduce another analysis.
By using bilinear elements for p_h and $\mathbf{u}_h$, the Laplacian of the approximate solution $\Delta\mathbf{u}_h$ becomes zero. Therefore, the SUPG stabilization approach becomes equal to the GLS (Galerkin Least Square). To be sure that this simplification is not affecting convergence, we test the couple of elements $\mathbb{Q}_2-\mathbb{Q}_2$, which results are reported in Figure 3.3. Fixing

$Re = 1$ both the quasi-static and the dynamic methods are super-convergent, while for $Re = 1000$ we obtain same performances as using bilinear elements, with optimal convergence for velocity L^2 and H^1 and sub-optimal for pressure L^2 errors. Therefore, we may exclude that bilinear elements interact with stabilization. However, we emphasize that further mesh refinement are needed.
Notice that, choosing the couple $\mathbb{Q}_2 - \mathbb{Q}_2$ accuracy significantly increases, since relative errors are lowered by a factor of 1000.

3.4.2. Semi-Implicit BDF

As it can be seen in fig. 3.4, the semi-implicit approach combined with the quasi-static approximation of fine scales is consistent with the results obtained with the implicit scheme. On the contrary, the dynamic semi-implicit method is converging only at low Reynolds, while for $Re = 1000$ both pressure and velocity errors are divergent with respect to the mesh refinement. At the moment we are not yet able to provide a exhaustive explanation of this non-convergence, which may be caused by an inconsistency in the formulation.
The subject is currently under investigation and we hope to give some updates in further publications.

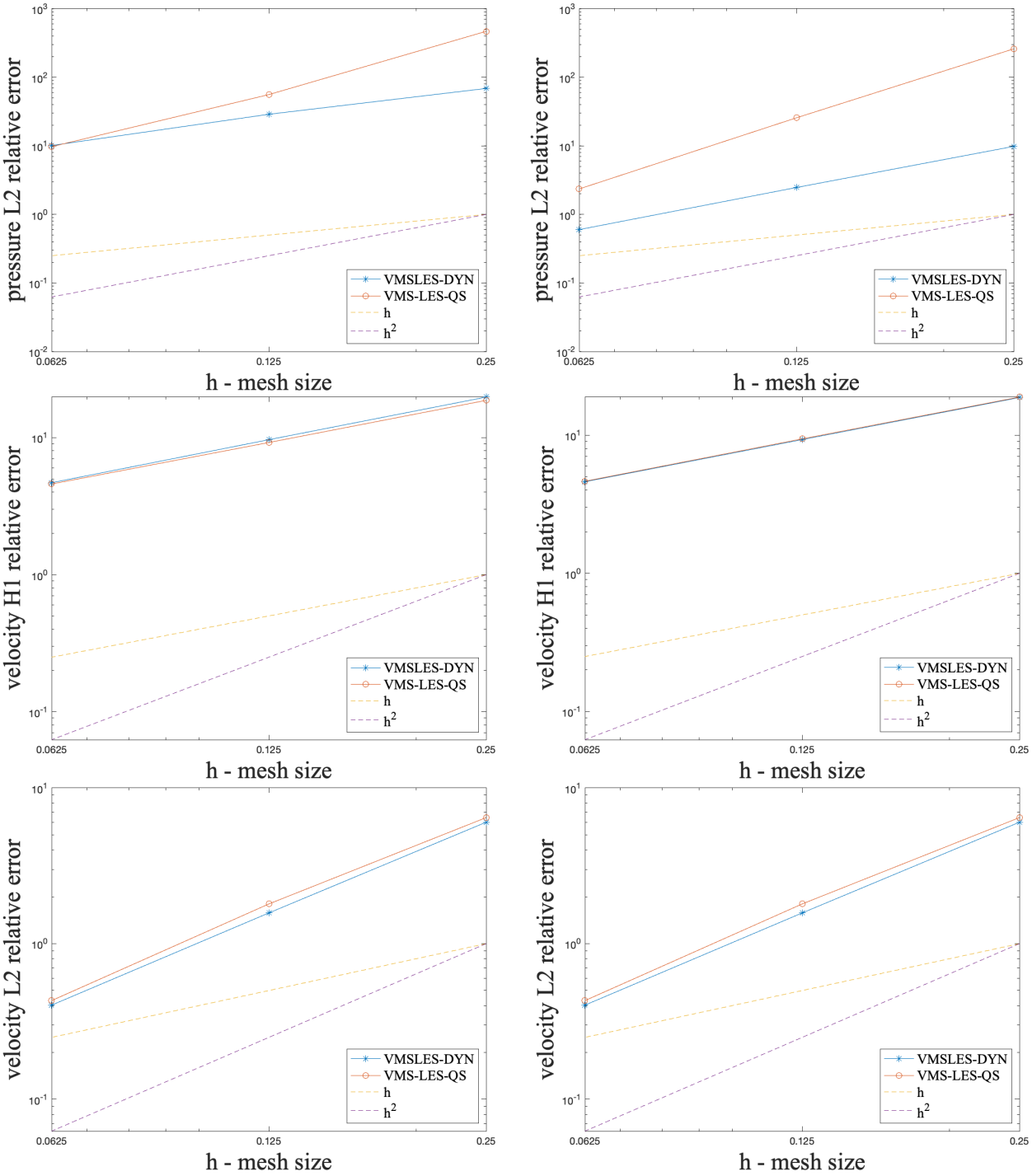


Figure 3.2: Convergence tests for pressure and velocity for implicit, fully discrete quasi-static and dynamic methods, with $\mathbb{Q}_1 - \mathbb{Q}_1$ elements. On the left side $Re = 1$ on the right side $Re = 1000$.

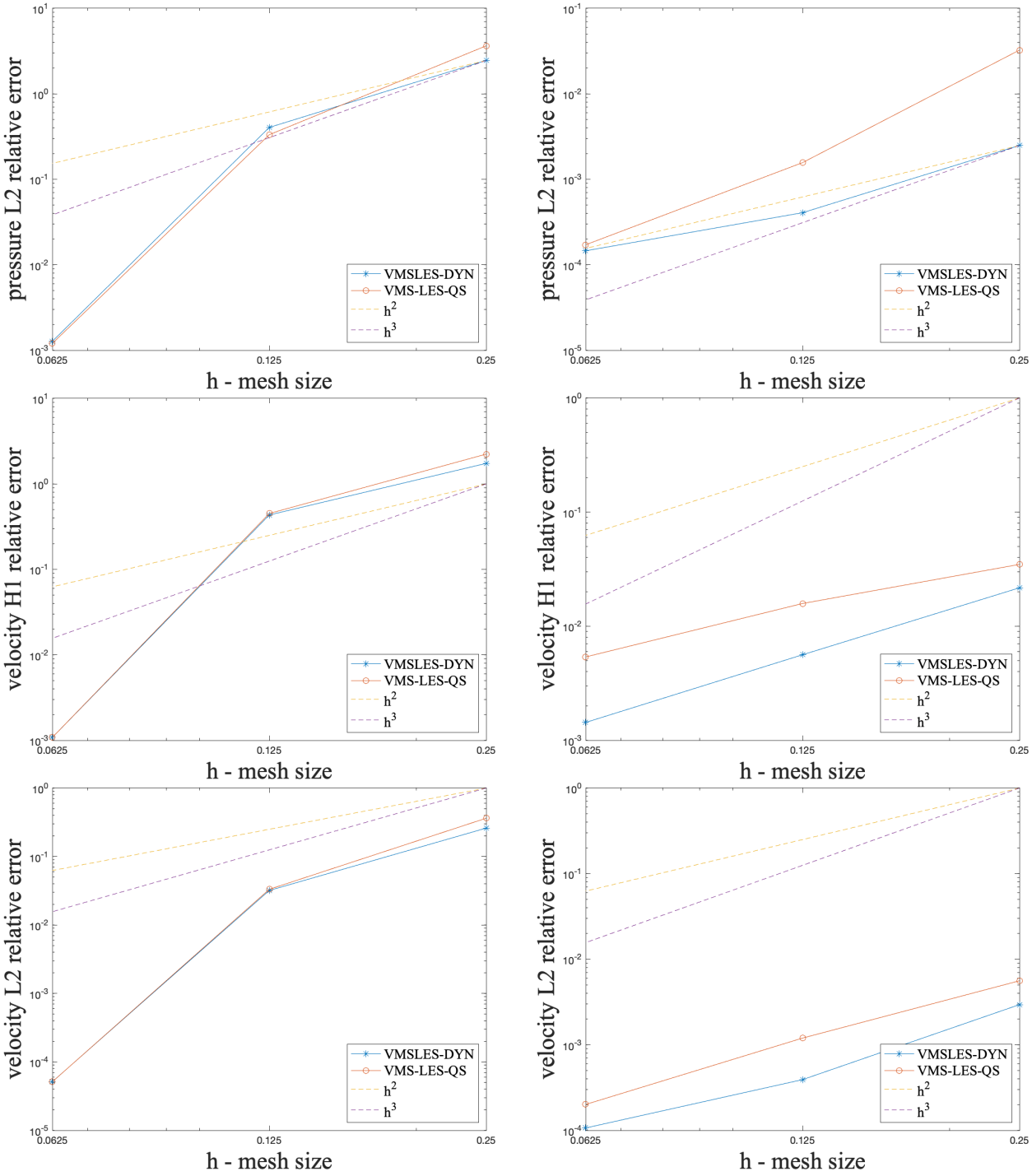


Figure 3.3: Convergence tests for pressure and velocity for implicit, fully discrete quasi-static and dynamic methods, with with $\mathbb{Q}_2 - \mathbb{Q}_2$ elements. On the left side $Re = 1$ on the right side $Re = 1000$.

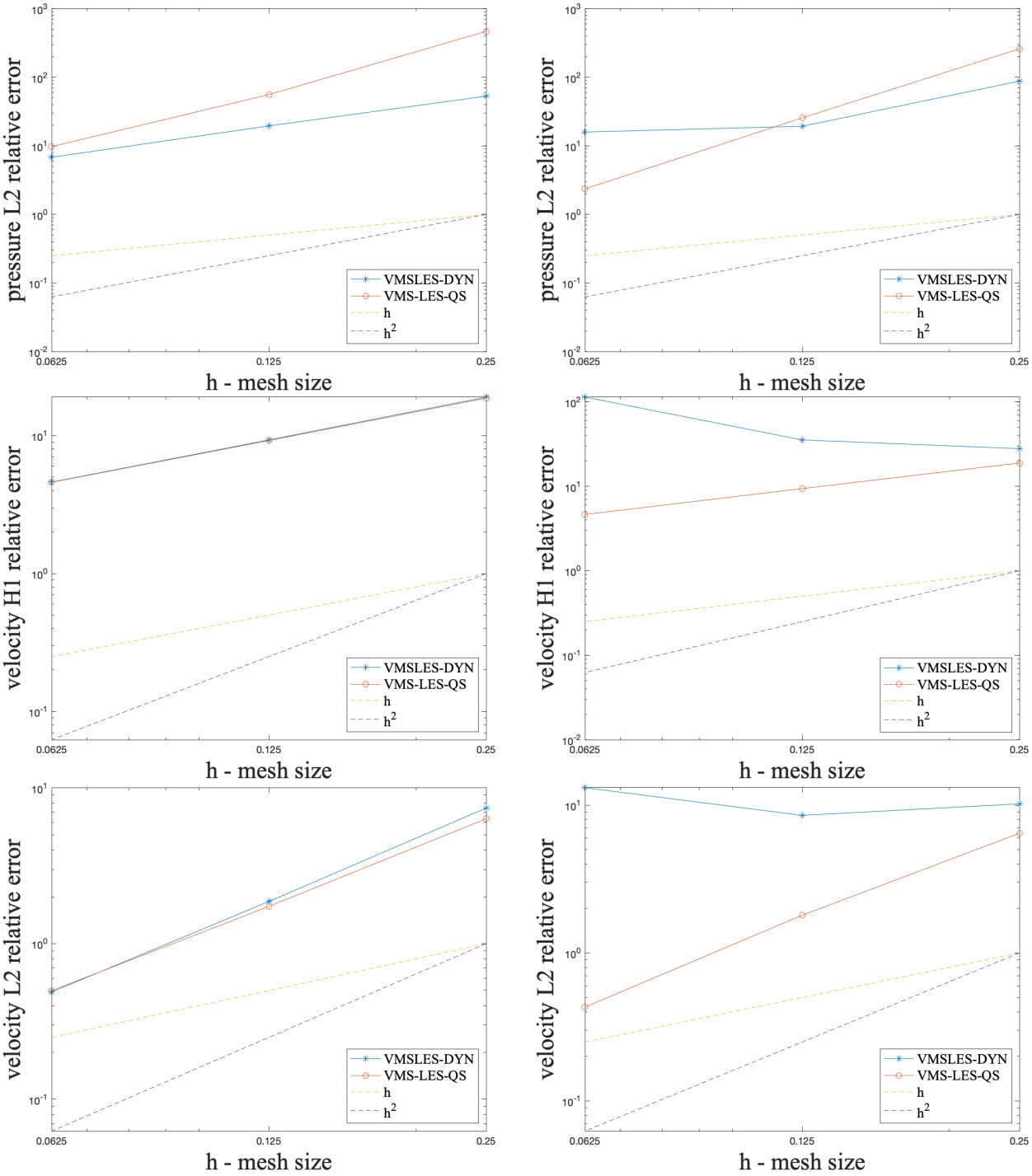


Figure 3.4: Convergence tests for pressure and velocity for semi-implicit, fully discrete quasi-static and dynamic methods, with with $\mathbb{Q}_1 - \mathbb{Q}_1$ elements. On the left side $Re = 1$ on the right side $Re = 1000$.

3.4.3. Discussion of mesh-refinement analysis

In this section we summarize the results of the mesh-convergence analysis and we give some hints for further developments.
First, we observe that the implicit scheme is always convergent, while the semi-implicit dynamic VMS-LES method is divergent for high Reynolds. This behaviour may suggest that there could be some inconsistencies in the implementation of the semi-implicit dynamic method, which becomes significant when convection is dominating on advection.
Second, the dynamic scheme (when not diverging) gives always lower percentage of errors with respect to the quasi-static, which means that it improves accuracy.
Third, to deepen this study it is important to go ahead with the mesh refinement, reducing the size of h.
Fourth, we notice that the effect of stabilization on pressure is different than that on velocity. Indeed, in our study, pressure errors have shown to be sensitive to the change of flow regime or sub-scales discretization approach (dynamic to quasi-static). We think that, an interesting topic for further analysis may be to study the pressure convergence for the VMS-LES method by changing the definition of the stabilization parameters.
Following on from this point, we highlight three remarks:

- A possible improvement to fix the behaviour of pressure, could be a definition of τ which modifies the dependence on h between asymptotic and pre-asymptotic regime as in [23];
- The convergence rate may change between the dynamic and the quasi-static approaches due to the different definition of τ_C (see appendix A);
- By increasing the degree of the approximating functions, it is possible to increase accuracy. However, we did not observe any improvement in convergence for turbulent regimes, which is the focus of this work.

3.5. Consistency analysis

As already introduced, the VMS-LES method with a dynamic approximation of fine scales aims to solve the inconsistency arising in the quasi-static by lowering Δt [5].
To accomplish this goal, in our formulation, we use the procedure described below:

1. In the quasi-static scheme, we define a τ_M depending one the time-step, while in the dynamic $\widetilde{\tau_M}$ is independent from Δt [13];
2. We set τ_C and $\widetilde{\tau_C}$ as function of τ_M (and $\widetilde{\tau_M}$) according to equations (2.50) and

(2.61), respectively;

3. Before plugging $\widetilde{\tau_M}$ in the fully-discrete weak formulation of the problem, we restore its time-dependence, by defining a $\breve{\tau_M}$ which depends on Δt, according to equation (2.90).

Indeed, in equation (2.50) when $\Delta t \longrightarrow 0$, $\tau_C \longrightarrow \infty$, while in equation (2.61) $\widetilde{\tau_C}$ is constant with respect to Δt.
The aim of this section is to prove numerically that the implementation of the dynamic VMS-LES method gives results in agreement with theory.
For this purpose, we fix the space refinement as $N^{Ref} = 3$, the Reynolds number as $Re = 1000$, since the worst case to solve is the turbulent regime, the final time as $T = 0.2\ s$ and the time-step vector as:

$$\Delta t = [10^{-2} \quad 10^{-3} \quad 10^{-4} \quad 10^{-5}]\ s \tag{3.12}$$

Notice that, in this work τ_C and τ_M are averaged parameters: the code print at every time-step an average in space of local values, then we average these values in time.

As it can be seen in Figure 3.5, with the quasi-static method τ_C and τ_M have both a linear trend, however, τ_C is diverging, while τ_M goes to zero.
Conversely, the graph on the left represents the stabilization parameters obtained with the dynamic method. We see that τ_C remains constant for every time-step and $\breve{\tau_M}$ behaves as the τ_M of the quasi-static.

We conclude this chapter highlighting that consistency results are in agreement with theory. Indeed, we see that τ_C behaves in agreement with equation (2.61) and therefore, the dynamic VMS-LES methods solves the inconsistency of the quasi-static.
The attentive reader should observe that, τ_C is not going to zero as τ_M. However, the constrain of consistency requires that all the stabilization terms go to zero and not necessarily the stabilization parameters. In fact, if in (2.51) τ_C is constant, instead of being divergent, the term $-(\nabla \cdot \mathbf{v}^h, \tau_C(\mathbf{u}^h) r_M(\mathbf{u}^h))$ tend to zero when reducing the timestep, since $r_M(\mathbf{u}^h) \to 0$.

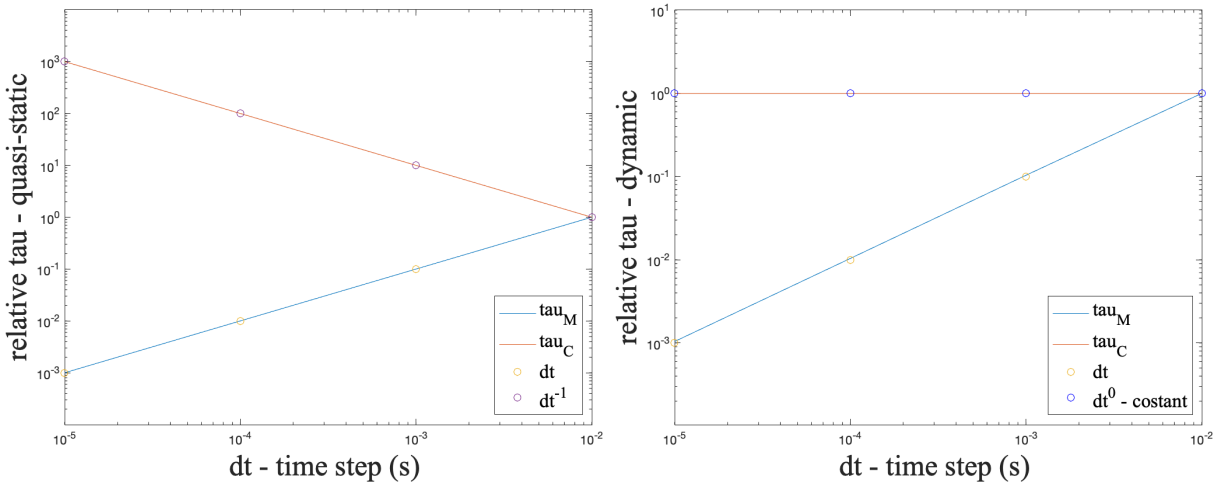


Figure 3.5: Relative τ with quasi-static (left) and dynamic approximation of fine scale (right).

4 | Model validation

In this chapter, we present the model validation of the dynamic and quasi-static VMS-LES methods.
The aim of a validation study is to evaluate if a mathematical model or a numerical method is able to represent the physics of a fluid dynamic problem. Therefore, we need a test case which features the crucial points of turbulence through simple constructs.
For this purpose, we test the aforementioned methods on the Taylor-Green Vortex (TGV) benchmark problem, which details are described in the next sections.
Before going into details of validation, let us resume how this chapter is organized. In Section 4.1 we introduce the TGV problem, in Section 4.2 we introduce the quantities which are used to compare our solution with the reference one. In Section 4.3 we present the numerical setup of the simulations. Finally, Sections 4.4 and 4.5 are devoted to results and discussion, respectively.

4.1. Taylor-Green Vortex test case

The TGV problem was firstly introduced in 1937 by G.I. Taylor and A. E. Green in [46]. In this paper, the authors introduce a flow which aims to mathematically represent the evolution of 3D vortexes arising in turbulent regimes.

The domain of the TGV problem is a cube centered in the origin of the axis, which extends of πL in every directions. However, in our computation, we apply a shifting, such that the domain is defined as:

$$\Omega = [0, 2\pi L]^3 \tag{4.1}$$

We set periodic boundary conditions, which impose that the velocity field on a face is equal to that on the opposite face. Therefore, according to the boundary definition showed

in Figure 4.1, we have:

$$\begin{cases} \mathbf{u}(2\pi L, y, z; t)|_{\Gamma_{\text{right}}} = \mathbf{u}(0, y, z; t)|_{\Gamma_{\text{left}}} \\ \mathbf{u}(x, 2\pi L, z; t)|_{\Gamma_{\text{top}}} = \mathbf{u}(x, 0, z; t)|_{\Gamma_{\text{bottom}}} \\ \mathbf{u}(x, y, 2\pi L; t)|_{\Gamma_{\text{front}}} = \mathbf{u}(x, y, 0; t)|_{\Gamma_{\text{back}}} \end{cases} \tag{4.2}$$

The initial condition of the problem are set as:

$$\begin{cases} u(x, y, z) = A\cos(ax)\sin(by)\sin(cz) \\ v(x, y, z) = B\sin(ax)\cos(by)\sin(cz) \\ w(x, y, z) = C\sin(ax)\sin(by)\cos(cz) \end{cases} \tag{4.3}$$

For these conditions it must hold the following consistency constrain:

$$Aa + Bb + Cc = 0 \tag{4.4}$$

We consider a simplified case in which $a = b = c$, which implies that $A = -B$ and $C = 0$, according to (4.4). We set $A = U_0$ and $a = \dfrac{1}{L}$, where U_0 is the free-stream velocity and L the characteristic length of the domain. Hence, the velocity profile at the time-step $t = 0$ becomes:

$$\mathbf{u}(\mathbf{x}, 0) = \mathbf{u}_0(\mathbf{x}) = \begin{bmatrix} U_0 \sin(\frac{x}{L})\cos(\frac{y}{L})\cos(\frac{z}{L}) \\ -U_0 \cos(\frac{x}{L})\sin(\frac{y}{L})\cos(\frac{z}{L}) \\ 0 \end{bmatrix} \text{ in } \Omega \times \{0\} \tag{4.5}$$

Since the aim of this chapter is to investigate the ability of the methods to represent the physics of turbulence, we set a Reynolds Number $Re = 1600$. This implies that the flow regime is initially laminar (i.e. it has low vorticity), but it evolves in a turbulent flow, where, due to the process of vortex-stretching, large eddies decay in smaller structures with higher vorticity [6].
Notice that, fixing the velocity U_0 and the characteristic length L equal to one, the Reynolds number of the problem is proportional to the cinematic viscosity:

$$Re = \frac{1}{\nu} \tag{4.6}$$

In Figure 4.2 we show the velocity field described by the initial conditions. To go into details of the formulation of the TGV problem, we address the reader to [6, 16, 46]

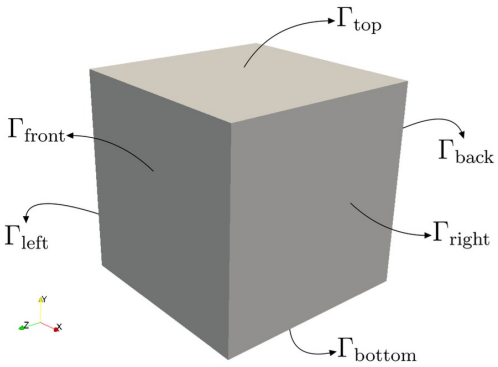


Figure 4.1: TGV. Boundary faces of the domain. Figure from [44]

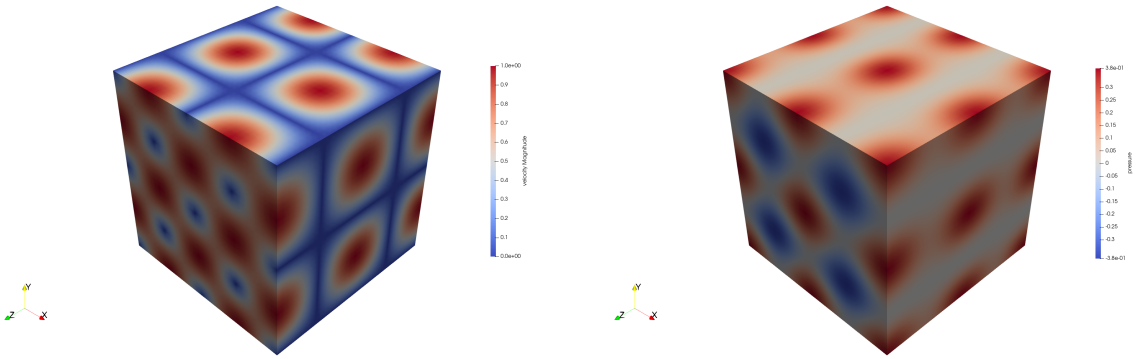


Figure 4.2: TGV. Velocity (left) and pressure (right) field at the time-step $t = 0$.

4.2. Benchmark quantities

In this study, the reference data that we use for validation are provided by a DNS [9, 34]. The retrieved solutions are in terms of kinetic energy, dissipation rate and enstrophy of the flow. Specifically, we compare the temporal evolution of these three integral quantities computed by the DNS, with those computed with the dynamic and quasi-static VMS-LES methods. For the sake of clarity, we report the definitions of these quantities.
The kinetic energy of the flow reads as:

$$E_k(t) = \int_\Omega \frac{1}{2}\rho \mathbf{u}(\mathbf{x}, t) \cdot \mathbf{u}(\mathbf{x}, t) d\Omega \tag{4.7}$$

while the enstrophy is defined as:

$$E_\omega(t) = \int_\Omega \frac{1}{2}\rho \boldsymbol{\omega}(\mathbf{x}, t) \cdot \boldsymbol{\omega}(\mathbf{x}, t) d\Omega \tag{4.8}$$

where $\boldsymbol{\omega} = \nabla \times \mathbf{u}$ is the vorticity of the fluid, defined as the curl of velocity.
However, the DNS solution retrieves dimensionless quantities, obtained by normalizing kinetic energy and enstrophy over domain's volume and density, such that:

$$e_k = \frac{E_k(t)}{\rho|\Omega|} \qquad e_\omega = \frac{E_\omega(t)}{\rho|\Omega|} \tag{4.9}$$

Finally, the dissipation rate of kinetic energy is defined as:

$$\epsilon(t) = -\frac{de_k(t)}{dt} \tag{4.10}$$

Under the assumption of incompressible flow, we can rearrange the (4.10) as function of the enstrophy, such that:

$$\epsilon(t) = 2\frac{\mu}{\rho}e_\omega(t) \tag{4.11}$$

Before moving on, let us introduce the role of kinetic energy in the study of turbulence. According to literature [17, 27–29], there are two regimes which can be distinguished in turbulent flows. On one side, we have the diffusive regime, which is characterized by a non-dissipative decay of eddies with bigger structure into smaller swirls. At this level, it is possible to consider that the vorticity conservation holds. On the other side, when the vortexes' scale is small enough, viscosity becomes predominant over diffusion and therefore, we witness the dissipation of kinetic energy by viscous stresses. In this second regime we cannot consider the conservation of vorticity.
We can observe this process in Figure 4.3, where we report the evolution in time of kinetic energy for the TGV problem: that kinetic energy drops after ten seconds and hence, the dissipation rate blows up. This represents the time when the flow from diffusive becomes viscous.

4.3. Numerical setup

As already introduced in Section 3.2, all calculations have been performed in the `life`$^{\texttt{x}}$ library. However, solving the TGV requires much greater computational efforts than the Beltrami flow. Therefore, we have run all simulations employing the hardware resources available in Politecnico di Milano's Laboratory for Modelling and Scientific Computing (MOX). In particular, we have used Xeon Gold 6238R @2.20GHz with 56 processors running in parallel.
The reference data comes from a Direct Numerical Simulation (DNS) based on a dealiased pseudo-spectral code (developed at Université catholique de Louvain, UCL)[34]. The num-

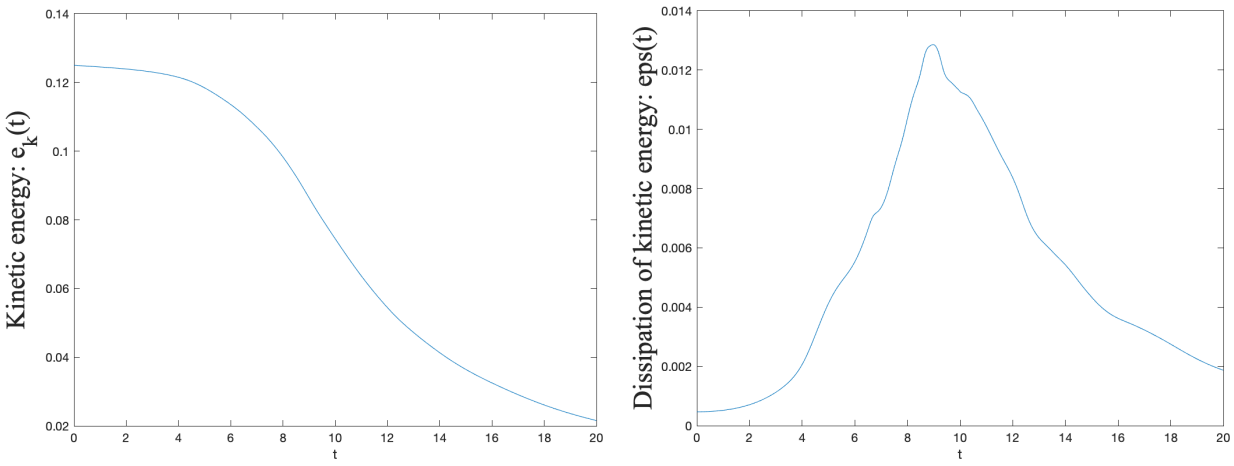


Figure 4.3: DNS solution of kinetic energy (left), and dissipation rate of kinetic energy (right). Data from [34]

ber of refinements of this solution is 7, which consists in a grid with 512 elements per edge, i.e. 512^3 elements in the domain. Moreover, since the Reynolds number is equal to 1600, we set viscosity $\nu = 6.25 \cdot 10^{-4}$ m^2/s, according to (4.6).
The numerical setup of our methods follows from the optimization performed by T. Tassi. in [44]. Hence, we apply the following settings to the four VMS-LES methods resumed in Sections 2.4.1 and 2.4.2: we fix the degree of BDF equal to three, we set biquadratic ($\mathbb{Q}2 - \mathbb{Q}2$) elements for both pressure and velocity coarse scale solutions and we consider a mesh made of hexahedra, with five refinements ($N_{el} = 512$) per edge. According to the conclusions of T.Tassi's work, this setup represents a good compromise between accuracy and computational cost.
However, since the computational cost of the dynamic VMS-LES method is very high, we truncate the simulation at the time $T_f = 5s$ for the semi-implicit methods and $T_f = 2.3s$ for the implicit ones, which require much larger resources. With this choice we are able to capture the beginning of the energy dissipation due to turbulence, with a reasonable time of simulation.

In Table 4.1 we resume the numerical setup of our simulations.

4.4. Numerical results

In this section we present the results of validation. We analyse the kinetic energy, its dissipation rate and enstrophy, separately. Than, in Section 4.5 we discuss our achievement, comparing them with the theoretical results and the results of verification.

Final time (T)	5 - 2.3 s
Reynolds number (Re)	1600
FEM spaces	$\mathbb{Q}2 - \mathbb{Q}2$
BDF order	3
Mesh refinement	$N_{Ref} = 6$

Table 4.1: Validation. Numerical setup.

Kinetic energy

In Figure 4.4 we compare the dynamic and quasi-static, implicit and semi-implicit VMS-LES methods with respect to the kinetic energy results, we can see that:

- The quasi-static implicit and semi-implicit methods are those which better represent the DNS solution. Indeed, we do not detect any additional sources of energy, which may distort the results.
- The dynamic, implicit VMS-LES method shows an higher dissipation of kinetic energy with respect to the reference solution. Therefore, may have an additional dissipation which arise due to the mathematical formulation of the method, which do not represent the physic of the problem.
- The dynamic, semi-implicit VMS-LES method diverges after few timesteps. This may be caused by a non-physical dissipation which arises due to an error in the implementation of the method. Indeed, we remind that this was the case which was diverging also in the code verification. Therefore, we consider this result as an additional proof of the conclusions of Chapter 3.

Dissipation rate of kinetic energy

Let us analyse the dissipation rate of kinetic energy. For this purpose, we compare the results obtained with both definitions (4.10) and (4.11).
According to Figure 4.5, we highlight that:

- The quasi-static VMS-LES method, with either implicit or semi-implicit treatment of non-linearities is following the DNS solution without any remarkable detachment for both definitions of ϵ_k.

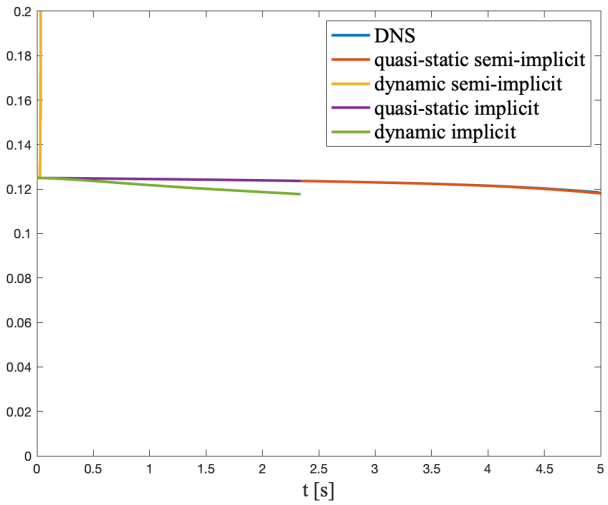


Figure 4.4: Comparison of kinetic energy $e_k(t)$ for the dynamic and quasi-static VMS-LES methods with implicit and semi-implicit treatment of non linearities

- The dynamic semi-implicit VMS-LES method is strongly divergent, in agreement with all previous results.
- The dynamic, implicit VMS-LES method leads to different solutions for ϵ_k when we compute it as the derivative of kinetic energy or through enstrophy. Indeed, by focusing on the graph on the left of Figure 4.5, we can see that the macroscopic trend of the dissipation rate is higher than the one of the reference solution. However, this is coherent with Figure 4.4 since an higher dissipation means that we have a faster decrease of energy, and this is exactly what we see in Figure 4.4. Notice that we have mentioned a "macroscopic trend" since in Figure 4.5 ϵ_k shows an oscillating pattern around a main trend. Besides this, we see from the graph on the right of Figure 4.5 that no issues arise, when computing the dissipation rate of kinetic energy as function of enstrophy, and all methods are almost overlapping the DNS solution.

Enstrophy

Lastly, we analyse enstrophy. As it can be seen in Figure 4.6, we can state that:

- For the quasi-static implicit and semi-implicit, and for the dynamic semi-implicit VMS-LES methods, we see the same behaviours as for the kinetic energy and its dissipation rate.
- On the other side, the dynamic, implicit VMS-LES method shows a better representation of the reference solution of enstrophy, with respect to the other two

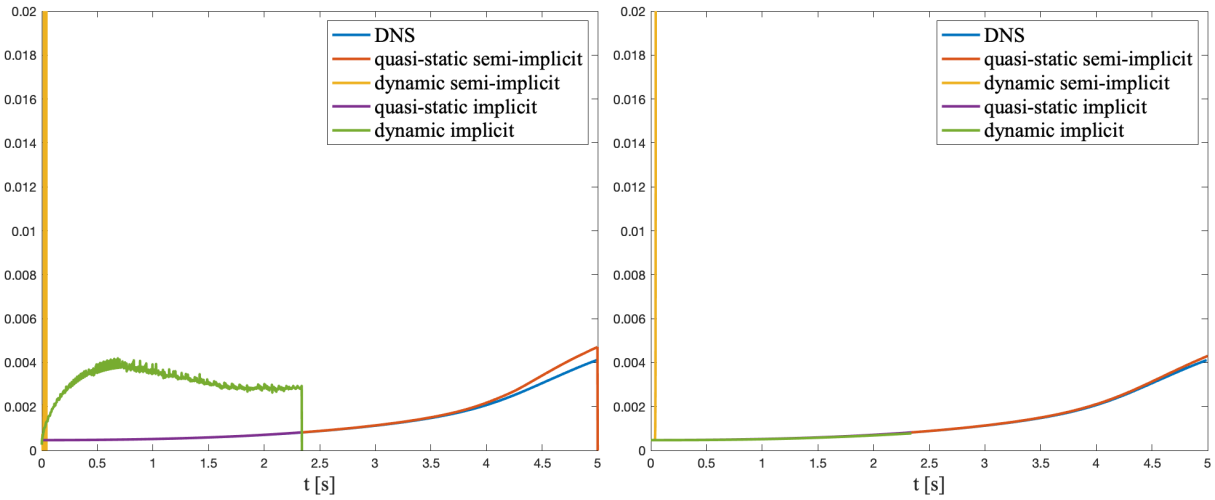


Figure 4.5: Dissipation rate of kinetic energy, computed with equation (4.10) on the left and with (4.11) on the right.

quantities. This may suggest a possible issue in the computation of velocity, which may be corrected or neglected when computing its rotor (i.e. vorticity).

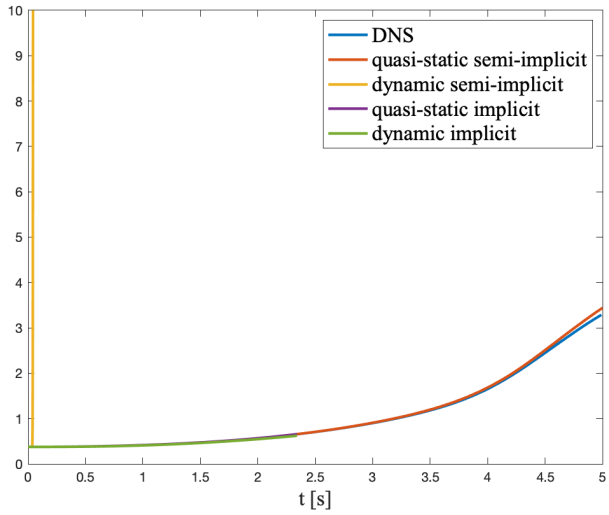


Figure 4.6: Comparison of enstrophy $e_\omega(t)$ for the dynamic and quasi-static VMS-LES methods with implicit and semi-implicit treatment of non linearities

4.5. Discussion of TGV results

The results of validation show that the quasi-static VMS-LES methods give a satisfying approximation of the problem, while in some cases the dynamic semi-implicit and implicit VMS-LES methods are less accurate. Let us go into details.

By focusing on kinetic energy, we identify two issues. On the one side, the dynamic semi-implicit method introduce a huge kinetic energy production, such that the solution already explodes to infinite in few timesteps and therefore the solver is not able to complete the simulations. On the other side, the dynamic implicit method shows an extra dissipation of kinetic energy, which results in a worsening of the solution with respect to the quasi-static method, when compared to the DNS reference data.
We believe that the nature of this two issues is profoundly different, since the first one occurs also in the verification, while the second one arises only in the model validation. Indeed, the dynamic implicit VMS-LES method may be consistent limited to the implementation, but not fully representative of the physic, while the dynamic semi-implicit method may have some implementation issues. For this purpose, from now on in this discussion we will not include the results of the dynamic semi-implicit VMS-LES method.

Let us focus on kinetic energy dissipation rate. First, we notice that regardless the method we choose, we have different results when using equation (4.10) or (4.11). However, this is due to the fact that when we compute the time derivative of kinetic energy, we introduce a numerical error related to the finite difference formula, which is avoided employing equation (4.11) [30, 44]. Second, we see that the dissipation rate computed by the dynamic implicit method trough equation (4.10) is much larger than the one computed by the other two methods and the DNS. However, this may be related to the combination of the two aforementioned issues: the extra dissipation of kinetic energy introduced by this method and the numerical error of the finite differences formula.

Finally, focusing on the enstrophy results, the dynamic implicit method does not show any worsening of the solutions with respect to the quasi-static methods, when compared to the DNS reference data. If we consider that kinetic energy depends on the square of velocity, while enstrophy depends on the square of vorticity, as defined in equations (4.4) and (4.8), respectively, we may conclude that there is a dissipation acting only on velocity and not on vorticity, which is its rotor.

5 | Flow past a carotid bifurcation

In Chapters 3 and 4 we have presented verification and validation for the quasi-static and dynamic VMS-LES methods, respectively. Therefore, we have tested the two methods only on idealised cases. In this Chapter, we want to extend our work to a real application, hence we test the two methods on a patience-specific carotid.

5.1. The clinical problem of a stenotic carotid

As reported by the World Health Organization (WHO), in the last decade the leading cause of death were cardiovascular diseases. In particular, a huge percentage of people have died for ischaemic heart diseases and stroke. The common thread of this two pathologies is *atherosclerosis*, the condition which arise when fat deposits, inflammatory cells, and scar tissue settle in the artery causing a reduction of its cross section, called stenosis [19, 31, 36]. In the stenotic region the blood flow may turn to turbulent, due to the pressure gradient induced by the constriction of the vessel (Figure 5.1). The onset of turbulence brings some drawbacks, such as: extra shear stresses on vessels' walls; recirculation regions and a possible interaction between eddies and blood elements due to the energy transfer which may arise if the two have a comparable scale. Therefore, to understand the phenomenon and prevent pathological situation, a branch of research in mathematical model is working to develop numerical methods able to represent turbulent flows in biological applications.

Since the carotids are an ideal place for the formations of stenosis, due to their biological conformation, in this chapter we test the quasi-static and dynamic VMS-LES methods on a stenotic carotids.

The carotids are the two main arteries which bring oxygenated blood from the hearth to the head. We have two carotids, one the left and one of the right (Figure 5.2). The first part of a carotid, is called common carotid artery (CCA). It starts from the aorta

and it goes through the neck until next to the larynx we have its bifurcation in external carotid (ECA) and internal carotid (ICA). The ECA provides blood to: pharynx, esophagus, larynx, mandible, scalp and face, while the ICA is the larger bifurcation which serves most part of the ipsilateral cerebral hemisphere, the eye and its appendices, the forehead and somewhat the nose and the ear [31, 37].

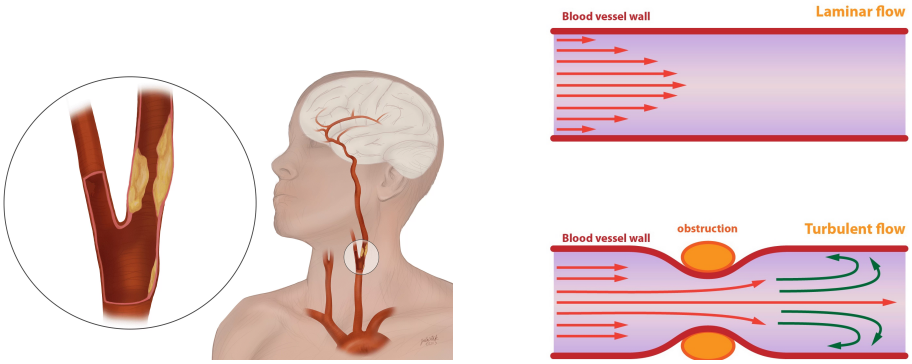


Figure 5.1: Atherosclerotic pathologies. A stenotic carotid (left) and the onset of turbulence due to the vessel's constriction (right). Figure taken from [49] and [48], respectively.

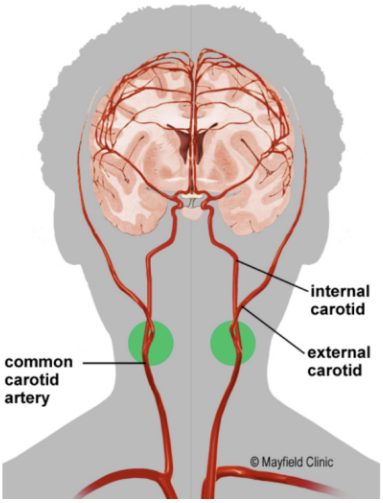


Figure 5.2: Scheme of the human carotids, the green spots highlight the bifurcation region of each CCA into ICA and ECA. Figure taken from [47]

5.2. Numerical setup

The case under consideration is the stenotic carotid presented in [31, 37, 40]. The blood flow was derived thanks to an Echo-Colour-Doppler analysis, while the geometry was derived by medical imagines of computed tomography (Angio Tac).

The stenosis has a diameter of 0.024 m and it is located in the ICA, immediately after the bifurcation of the CCA. At the inlet section of CCA we assume a parabolic velocity profile [11], which models the effect of heart pumping:

$$\mathbf{u}(\mathbf{x},t) = -\frac{2Q(t)}{\pi R^2}\left(1 - \frac{r^2(\mathbf{x})}{R^2}\right)\hat{\mathbf{n}}_{in} \tag{5.1}$$

where R is radius of the inlet surface, $r(\mathbf{x})$ is the radial position, $\hat{\mathbf{n}}_{in}$ is the outward-pointing normal of the surface and $Q(t)$ is the blood flow rate, shown in Figure 5.3.
We define the boundary faces as reported in Figure 5.4. We impose Dirichlet boundary conditions on the inlet section Γ_{CCA}, on the outlet face Γ_{ECA} and on the walls Γ_{walls}, such that: on the first one we fix the blood flow rate; on the second one we impose that the flow is 30% of the input flow of Γ_{CCA} and on the third one we impose a zero flow-rate to represents impermeable walls. Finally, on Γ_{ICA} we impose a Neumann boundary condition, with homogenous pressure.
To model the blood we set the density equal to $\rho = 1060$ kg/m^3 and the dynamic viscosity $\mu = 0.0033$ Pa $\cdot$ s.

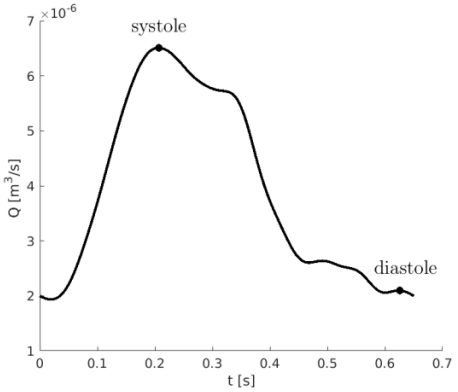


Figure 5.3: Temporal profile of the inlet flow rate $Q(t)$ imposed on Γ_{CCA}. Figure taken from [37]

The domain discretization provides a mesh of tetrahedra made of about 1,5 million active

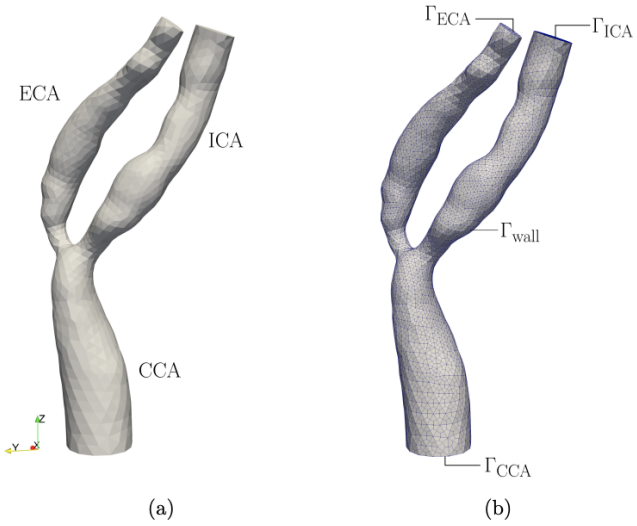


Figure 5.4: Stenotic carotid: geometry (a), boundary faces definition (b). Figure taken from [37]

cells with adaptive size, such that next to the bifurcation we have more refinements. In Table 5.1 and 5.2 we report all mesh specifications.
We set linear FE spaces ($\mathbb{P}1 - \mathbb{P}1$) for both pressure and velocity space approximation, and we perform the time discretization by means of the BDF3 scheme. Moreover, we fix the timestep as $\Delta t = 2 \cdot 10^{-3}$ s and we set a final time of $T_f = 2.2$ s. Notice that, in our study we consider an hearth period of 1.09 s, which means an heart frequency of 65 BPM. Therefore, we are simulating two hearth cycles.

h_{max}	$6.3174 \cdot 10^{-4}$ m
$h_{average}$	$2.4224 \cdot 10^{-4}$ m
h_{min}	$4.9289 \cdot 10^{-5}$ m

Table 5.1: List of cell diameters for the adaptive mesh of the carotid.

5.3. Numerical results

In this section we compare the results obtained by the quasi-static and dynamic, implicit VMS-LES methods. We first analyse the total kinetic energy and its dissipation rate. Than we evaluate the behaviour of velocity and pressure.

velocity DOFs	746184
pressure DOFs	248728
total DOFs	994912
total cells number	1500078

Table 5.2: Carotid. DOF specification

As it can be seen in Figure 5.5 the kinetic energy has a periodic trend with a peak around $t = 0.6s$ and a period of $t = 1.09s$, in agreement with an hearth frequency of 65 BPM. Moreover, the kinetic energy computed by the dynamic VMS-LES method follows the same trend of that one computed by the quasi-static method, but it presents some wiggles which periodically increase, for a small amount, its values. Hence, it becomes meaningless to evaluate the dissipation rate of kinetic energy computed with equation (4.10), since the computation of derivatives amplifies the wiggles (Figure 5.6, left). Indeed, it is better to evaluate the dissipation rate of kinetic energy computed as (4.11), which has a behaviour similar to the kinetic energy, (Figure 5.6, right).

For the solution visualization we focus on the ICA, since it is the vessel where we have the stenosis. We analyse four timestep, which are $t = 0.3s$, $t = 0.6s$, $t = 0.9s$ and $t = 1.09s$ to visualize a complete hearth cycle. In Figure 5.7 and 5.8 we report the velocity solutions. We see that the dynamic VMS-LES method presents a region of high velocity downstream of the stenosis, which does not appear in the solutions computed by the the quasi-static method. However, the nature of this region is ambiguous since it shows a pulsating behaviour. In this regard, by focusing on three consequential instant ($t = 0.89s$, $t = 0.9s$ and $t = 0.91s$) we observe that exactly in that region the velocity is changing magnitude and direction instant by instant (Figure 5.9), which may be related to the wiggles of kinetic energy observed in Figure 5.5. However, we think that this evolution of the velocity solution may be a purely numerical issue of the dynamic, implicit VMS-LES method, not representing any physical behaviour of the blood flow.
Nevertheless, in this work we do not go into details of this problem, which may be caused by the same issue that provokes the oscillations arising in model validation, and we leave it as hint for further researches.

For the sake of completeness we report in Figure 5.10 the pressure solutions, for which no issues arose.

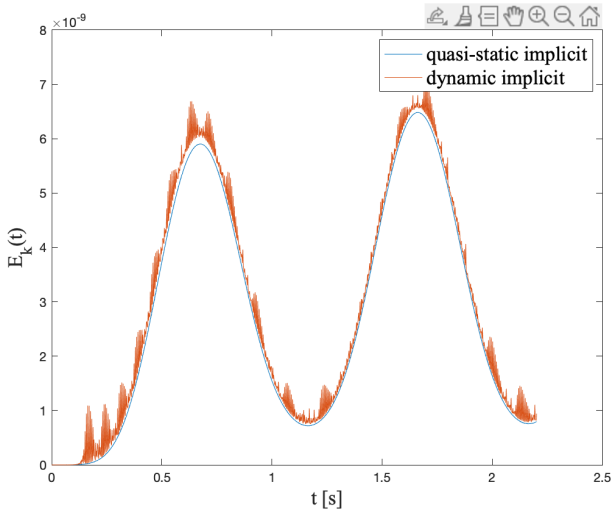


Figure 5.5: Space averaged total kinetic energy $E_k(t)$ (4.4).

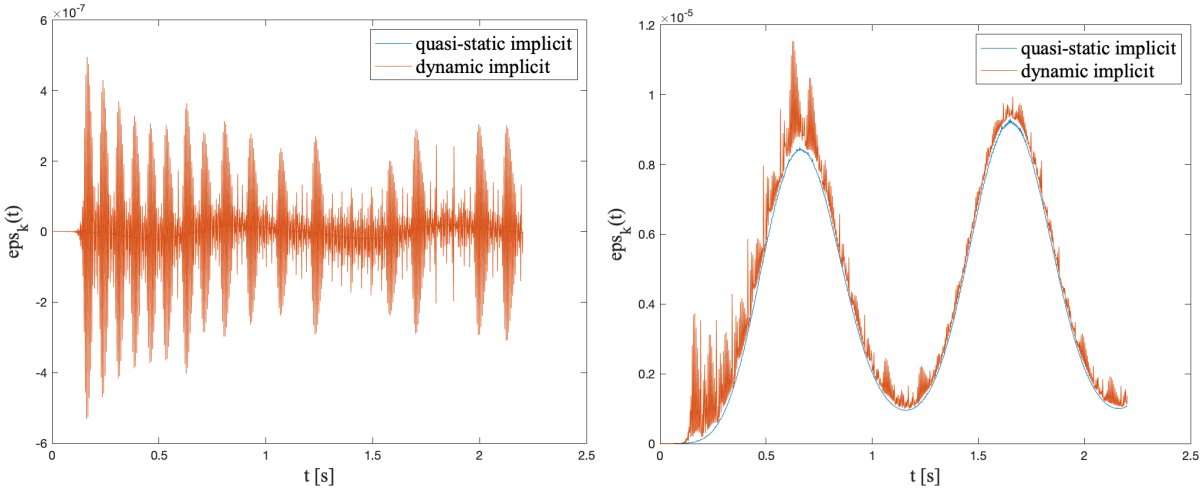


Figure 5.6: Dissipation rate of kinetic energy, computed with equation (4.10) (left) and (4.11) (right) .

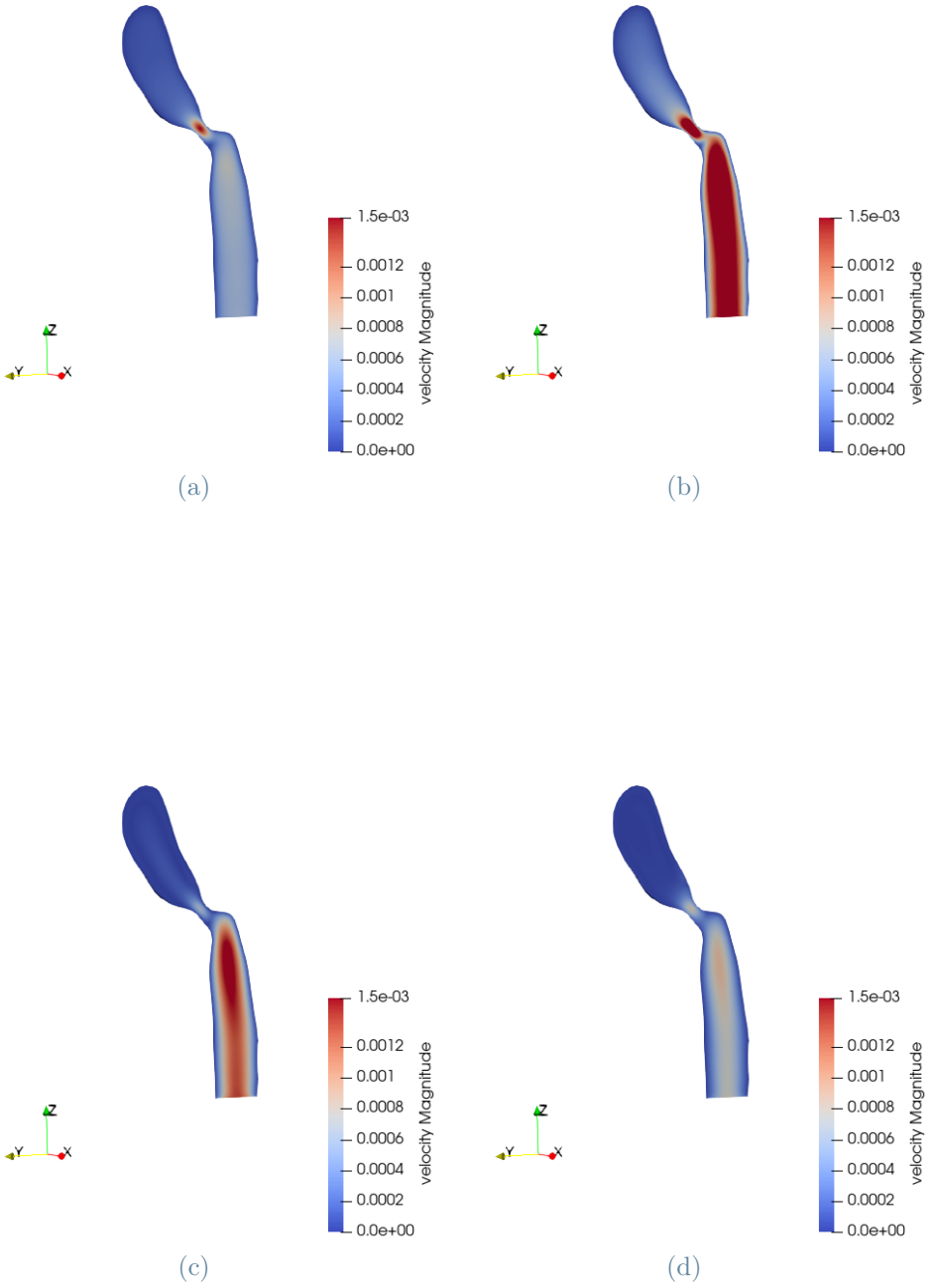


Figure 5.7: Quasi-static, implicit VMS-LES. Velocity profile inside the ICA, across the stenotic region. From (a) to (d) $t = 0.3s$, $t = 0.6s$, $t = 0.9s$, $t = 1.09s$

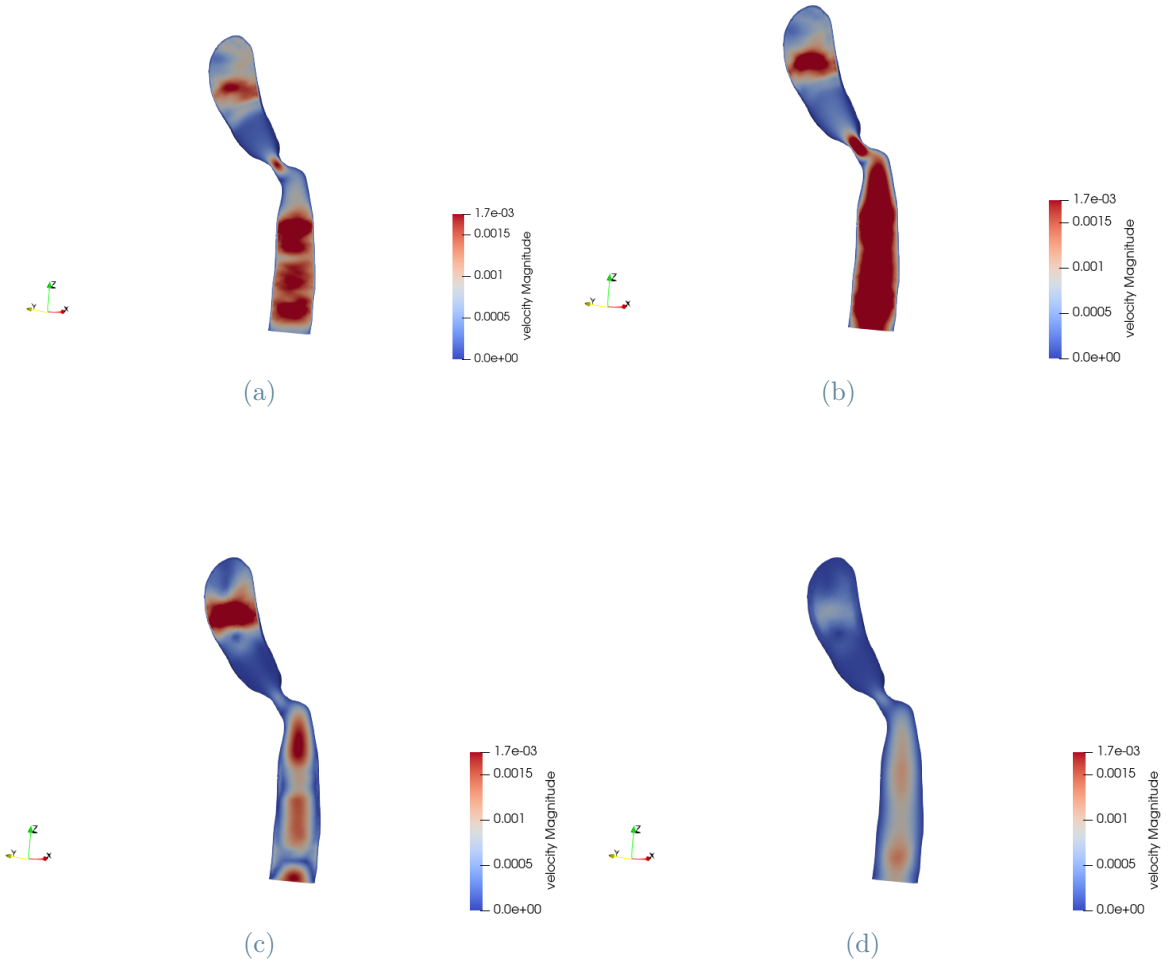
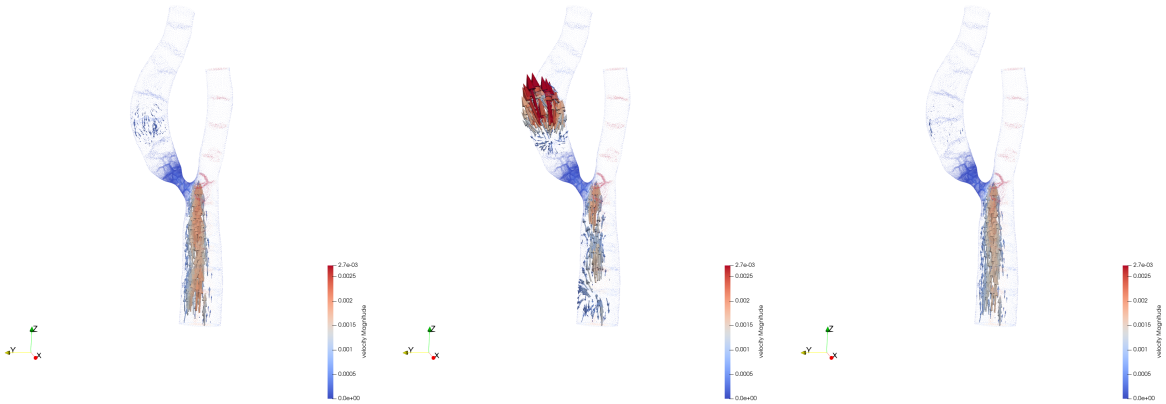


Figure 5.8: Dynamic, implicit VMS-LES. Velocity profile inside the ICA, across the stenotic region. From (a) to (d) $t = 0.3s$, $t = 0.6s$, $t = 0.9s$, $t = 1.09s$



(a) (b) (c)

Figure 5.9: Dynamic, implicit VMS-LES method. Velocity field in the ICA for three consecutive timesteps: $t = 0.89s$ (a), $t = 0.9s$ (b) and $t = 0.91s$ (c).

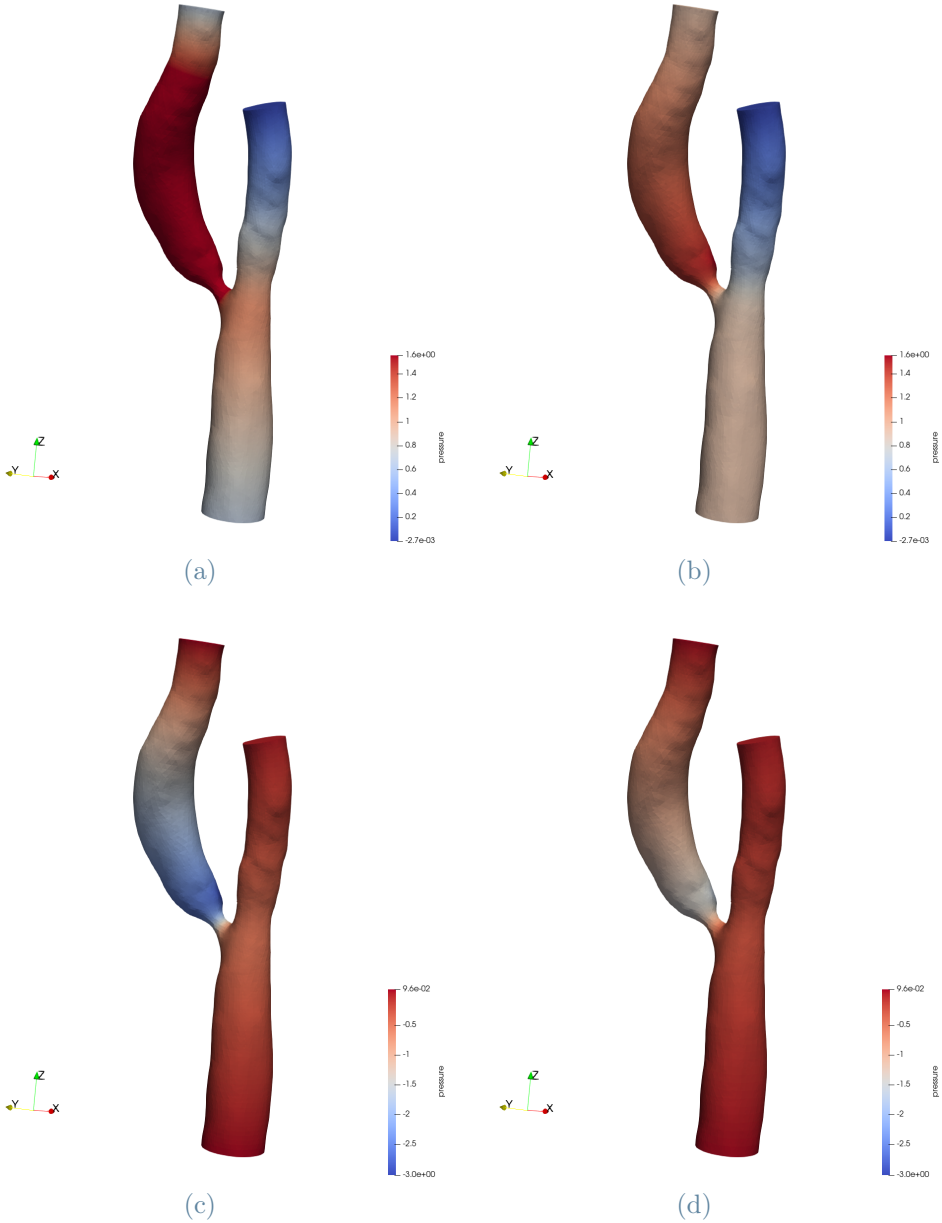


Figure 5.10: Quasi-static, implicit VMS-LES method. Pressure distribution of the carotid. From (a) to (d) $t = 0.3s$, $t = 0.6s$, $t = 0.9s$, $t = 1.09s$

6 | Conclusions

This thesis concerns the software verification and the model validation of quasi-static and dynamic VMS-LES methods for turbulent flows. We present a systematic analysis of the two methods, to evaluate whether the dynamic improves the results of the quasi-static method. However, despite some promising achievements, further analysis would be necessary to completely assess our findings.

Software Verification

We perform the software verification on the Beltrami-flow benchmark, with the aim to evaluate if there are any implementation error and if the methods are consistent. For this purpose, we test both the quasi-static and the dynamic VMS-LES methods, coupled with either implicit or semi-implicit treatment of non-linearities. Our numerical tests highlighted three important conclusions.

First, the errors computed with both quasi-static and dynamic implicit VMS-LES methods are convergent with respect to the mesh refinement. Moreover, compared to the quasi-static, the dynamic method is improving the results in turbulent conditions, since it leads to slightly lower errors.

Second, the dynamic approximation of fine scales is effectively solving the inconsistency against which it has been formulated. Indeed, while in the quasi-static scheme the pressure stabilization parameter τ_C blows up when reducing the time-step (Δt), possibly affecting accuracy and stability of the numerical solution, in the dynamic method $\widetilde{\tau_C}$ is constant for every Δt.

Third, by using the semi-implicit formulation instead of the implicit one, pressure and velocity errors computed by the dynamic VMS-LES method blow up when increasing the mesh-refinement. Therefore, this method, with respect to the software verification, is not converging.

Our hint for future research, is to further analyse the implementation of the dynamic, semi-implicit VMS-LES method. We underline that this approach may represent an important improvement over the implicit scheme, since it allows to significantly reduce the simulation time.

Moreover, we have detected a behaviour of pressure errors which is not in agreement with the theoretical estimates, as are velocity errors, since it is sensible to the choice of the Reynolds number and of the method. Therefore, an interesting topic will be to deepen the effect of stabilization on pressure.

Model Validation

The model validation of the quasi-static and dynamic VMS-LES methods, with semi-implicit and implicit treatment of non-linearities, has retrieved that the quasi-static methods better represent the benchmark quantities than the dynamic.
On the one hand, in the dynamic semi-implicit VMS-LES method, both energy and enstrophy blow up to infinite after few timesteps and therefore, the solver is not able to end the simulations. This may be consistent with the hypothesis of an implementation error, retrieved by the software verification.
On the other hand, the benchmark quantities computed by the dynamic implicit VMS-LES method does not explode to infinite, but they do not represent the reference data with the same accuracy as the quasi-static implicit and semi-implicit methods. In particular, compared to the DNS data, the dynamic implicit method introduces an extra dissipation of kinetic energy, which leads to either a lowering of $E_k(t)$ in time, or an increase of the dissipation rate. Moreover, the dissipation rate (of kinetic energy) shows an oscillating behaviour around a macroscopic trend.
Since these two issues appear only in validation, we believe that they are related to the physics of the problem and in particular, to the ability of the method to represent turbulence.
As hint for further researches, we suggest to perform the model validation changing a variety of physical quantities, to find out whether the extra dissipation and the oscillating behaviour of the results are somehow related to the physical properties.

Real application

In the last chapter of this thesis, we have applied the quasi-static and dynamic implicit VMS-LES methods, to a patience-specific stenotic carotid. We have obtained similar results to those retrieved by the model validation, meaning that our benchmark is representative of real applications.
Indeed, by comparing the physical quantities, we see that both the kinetic energy and the dissipation rate computed by the dynamic method, present an oscillating behaviour around a macroscopic trend, where the macroscopic trend overlaps the solution yielded by the quasi-static method. In this regard, we think that these oscillations may not be

representative of the physics of the problem, but related to a numerical issue.
Therefore, an interesting topic for future researches is to further investigate the nature of these wiggles, which affect the accuracy of the dynamic VMS-LES method, with respect to the quasi-static.

Final consideration and further developments

To conclude this work we want to highlight two remarks, which may be influence further works:

- According to literature, the dynamic VMS-LES method has been introduced after the quasi-static, to correct its inconsistency. Its formulation is obtained from the one of the quasi-static method by simple changing the definition of the stabilization parameters. However, from a theoretical point of view, we may think the contrary and hence, that the quasi-static is a simplification of the dynamic. Indeed, the equations to define the velocity fine-scale solutions are the same for both methods, but in the quasi-static we assume $\frac{\partial \mathbf{u}'}{\partial t} = 0$. Accordingly, in the final formulation of the quasi-static method we simplify the term $\left(\mathbf{v}_h, \rho \frac{\partial \mathbf{u}'}{\partial t}\right)$, but we should include it in the final formulation of the dynamic VMS-LES method.
  However, the actual definition of the dynamic VMS-LES method is neglecting this term.

- Overall the dynamic VMS-LES method has proved to solve the inconsistency against which it was introduced. However, in this work, it has shown to be less accurate and reliable (when changing the non-linearities treatment) than the quasi-static method. In this regard, we believe that it represents an improvement to the quasi-static, but it still need to be "fine-tuned".

Bibliography

[1] *Guide for the Verification and Validation of Computational Fluid Dynamics Simulations (AIAA G-077-1998(2002))*. American Institute of Aeronautics and Astronautics, Inc.

[2] P. C. Africa. lifex: A flexible, high performance library for the numerical solution of complex finite element problems. *SoftwareX*, 20:101252, 2022.

[3] M. Antuono. Tri-periodic fully three-dimensional analytic solutions for the navier–stokes equations. *Journal of Fluid Mechanics*, 890:A23, 2020.

[4] D. Arndt, W. Bangerth, M. Feder, M. Fehling, R. Gassmöller, T. Heister, L. Heltai, M. Kronbichler, M. Maier, P. Munch, J.-P. Pelteret, S. Sticko, B. Turcksin, and D. Wells. The `deal.II` library, version 9.4. *Journal of Numerical Mathematics*, 30 (3):231–246, 2022. URL `https://dealii.org/deal94-preprint.pdf`.

[5] Y. Bazilevs, V. Calo, J. Cottrell, T. Hughes, A. Reali, and G. Scovazzi. Variational multiscale residual-based turbulence modeling for large eddy simulation of incompressible flows. *Computer Methods in Applied Mechanics and Engineering*, 197(1): 173–201, 2007.

[6] R. E. Bensow, M. G. Larson, and P. Westerlund. Vorticity-strain residual-based turbulence modelling of the taylor-green vortex. *International Journal for Numerical Methods in Fluids*, 54(6-8):745–756, June 2007.

[7] L. Berselli and S. Spirito. *Weak solutions to the Navier-Stokes equations constructed by semi-discretization are suitable*, pages 85–97. 01 2016.

[8] J. Blazek. Chapter 7 - turbulence modelling. In J. Blazek, editor, *Computational Fluid Dynamics: Principles and Applications (Second Edition)*, pages 227–270. Elsevier Science, Oxford, second edition edition, 2005.

[9] M. E. Brachet, D. I. Meiron, S. A. Orszag, B. Nickel, R. H. Morf, and U. Frisch. Small-scale structure of the taylor–green vortex. *Journal of Fluid Mechanics*, 130: 411 – 452, 1983.

[10] Brezzi, F. On the existence, uniqueness and approximation of saddle-point problems arising from lagrangian multipliers. *R.A.I.R.O. Analyse Numérique*, 8(2):129–151, 1974.

[11] I. Campbell, J. Ries, S. Dhawan, A. Quyyumi, W. Taylor, and J. Oshinski. Effect of inlet velocity profiles on patient-specific computational fluid dynamics simulations of the carotid bifurcation. *Journal of biomechanical engineering*, 134:051001, 05 2012.

[12] F. E. Cellier and E. Kofman. *Continuous system simulation*. Springer Science & Business Media, 2006.

[13] R. Codina. Stabilized finite element approximation of transient incompressible flows using orthogonal subscales. *Computer Methods in Applied Mechanics and Engineering*, 191:4295–4321, 08 2002.

[14] R. Codina, J. Principe, O. Guasch, and S. Badia. Time dependent subscales in the stabilized finite element approximation of incompressible flow problems. *Computer Methods in Applied Mechanics and Engineering*, 196(21):2413–2430, 2007.

[15] H. Coleman and C. Members. *ASME V and V 20-2009 Standard for Verification and Validation in Computational Fluid Dynamics and Heat Transfer (V and V20 Committee Chair and principal author)*. 01 2009.

[16] O. Colomés, S. Badia, R. Codina, and J. Principe. Assessment of variational multiscale models for the large eddy simulation of turbulent incompressible flows. *Computer Methods in Applied Mechanics and Engineering*, 285, 11 2014.

[17] J. W. Elsner and W. Eisner. On the measurement of turbulence energy dissipation. *Measurement Science and Technology*, 7(10):1334–1348, 1996. Cited By :29.

[18] C. R. Ethier and D. A. Steinman. Exact fully 3D Navier-Stokes solutions for benchmarking. *International Journal for Numerical Methods in Fluids*, 19(5):369–375, Sept. 1994.

[19] E. Falk. Pathogenesis of atherosclerosis. *Journal of the American College of Cardiology*, 47(8, Supplement):C7–C12, 2006. Detection of Vulnerable Plaques.

[20] C. Fefferman. Existence and smoothness of the navier-stokes equation. *The Millennium Prize Problems*, 01 2006.

[21] D. Forti and L. Dedè. Semi-implicit bdf time discretization of the navier–stokes equations with vms-les modeling in a high performance computing framework. *Computers and Fluids*, 117:168–182, 2015.

[22] N. Franck. Unsteady flows modeling and computation. 12 2007.

[23] T.-P. Fries, H. Matthies, et al. *A review of Petrov-Galerkin stabilization approaches and an extension to meshfree methods*. Inst. für wiss. Rechnen, 2004.

[24] T. J. Hughes, L. P. Franca, and M. Balestra. A new finite element formulation for computational fluid dynamics: V. circumventing the babuška-brezzi condition: a stable petrov-galerkin formulation of the stokes problem accommodating equal-order interpolations. *Computer Methods in Applied Mechanics and Engineering*, 59(1): 85–99, 1986.

[25] T. J. R. Hughes, L. Mazzei, and K. E. Jansen. Large eddy simulation and the variational multiscale method. *Comput. Vis. Sci.*, 3(1–2):47–59, may 2000.

[26] D. Irisarri and G. Hauke. A posteriori error estimation and adaptivity based on vms for the incompressible navier–stokes equations. *Computer Methods in Applied Mechanics and Engineering*, 373:113508, 2021.

[27] A. N. Kolmogorov. A refinement of previous hypotheses concerning the local structure of turbulence in a viscous incompressible fluid at high reynolds number. *Journal of Fluid Mechanics*, 13:82 – 85, 1962.

[28] A. N. Kolmogorov. The local structure of turbulence in incompressible viscous fluid for very large reynolds numbers. *Proceedings: Mathematical and Physical Sciences*, 434(1890):9–13, 1991.

[29] R. H. Kraichnan. On kolmogorov's inertial-range theories. *Journal of Fluid Mechanics*, 62(2):305–330, 1974.

[30] A. Krämer, K. Küllmer, D. Reith, W. Joppich, and H. Foysi. Semi-lagrangian off-lattice boltzmann method for weakly compressible flows. *Phys. Rev. E*, 95:023305, 02 2017.

[31] R. M. Lancellotti. Large eddy simulations in haemodinamics: models and applications, 2016.

[32] P. D. Lax and R. D. Richtmyer. Survey of the stability of linear finite difference equations. *Communications on Pure and Applied Mathematics*, 9(2):267–293, 1956.

[33] X. Lü, T. Lu, T. Yang, H. Salonen, Z. Dai, P. Droege, and H. Chen. Improving the energy efficiency of buildings based on fluid dynamics models: A critical review. *Energies*, 14:5384, 08 2021.

[34] NASA. Problem c3.5 direct numerical simulation of the taylor-green vortex at re = 1600, 2023-02-27. URL `https://cfd.ku.edu/hiocfd/case_c3.5.pdf`.

[35] M. Nived, S. S. C. Athkuri, and V. Eswaran. On the application of higher-order backward difference (bdf) methods for computing turbulent flows. *Computers and Mathematics with Applications*, 117:299–311, 2022.

[36] J. o Deguchi, M. Aikawa, C.-H. Tung, E. Aikawa, D.-E. Kim, V. Ntziachristos, R. Weissleder, and P. Libby. Inflammation in atherosclerosis. *Circulation*, 114(1): 55–62, 2006.

[37] S. Perdoncin. Numerical validation of a variational multiscale-les turbulence model for blood flows. Master thesis, Politecnico di Milano, 2020.

[38] V. Průša and K. Rajagopal. A note on the decay of vortices in a viscous fluid. *Meccanica*, 46:875–880, 08 2011.

[39] A. Quarteroni and A. Valli. *Numerical approximation of partial differential equations*, volume 23. Springer Science & Business Media, 2008.

[40] A. Quarteroni, L. Dede', A. Manzoni, and C. Vergara. *Mathematical Modelling of the Human Cardiovascular System: Data, Numerical Approximation, Clinical Applications*. Cambridge Monographs on Applied and Computational Mathematics. Cambridge University Press, 2019.

[41] P. J. Roache. Code Verification by the Method of Manufactured Solutions . *Journal of Fluids Engineering*, 124(1):4–10, 11 2001.

[42] P. Ryzhakov and J. M. Marti. A semi-explicit multi-step method for solving incompressible navier-stokes equations. *Applied Sciences*, 8:119, 01 2018.

[43] F. Schmitt and C. Hirsch. Direct test of boussinesq's hypothesis and the k-transport equation using experimental, dns and les data. In W. RODI and N. FUEYO, editors, *Engineering Turbulence Modelling and Experiments 5*, pages 167–176. Elsevier Science Ltd, Oxford, 2002.

[44] T. Tassi. Stabilization methods for transport dominated problems enhanced by artificial neural networks : towards les turbulence modelling. master thesis, Politecnico di Milano, 2021.

[45] G. Taylor. Lxxv. on the decay of vortices in a viscous fluid. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 46(274):671–674, 1923.

[46] G. I. S. Taylor and A. E. Green. Mechanism of the production of small eddies

from large ones. *Proceedings of The Royal Society A: Mathematical, Physical and Engineering Sciences*, 158:499–521, 1937.

[47] URL:. `https://mayfieldclinic.com/pe-carotidstenosis.htm`, . Online; accessed 2 April 2023.

[48] URL:. `http://fblt.cz/en/skripta/x-srdce-a-obeh-krve/2-krevni-obeh/`, . Online; accessed 2 April 2023.

[49] URL:. `https://www.hopkinsmedicine.org/heart_vascular_institute/vascular-surgery/our-specialties/carotid-artery-disease.html`, . Online; accessed 2 April 2023.

[50] A. Zingaro. *Mathematical and numerical models for the fluid dynamics of the human heart*. Phd thesis, Politecnico di Milano, 2022.

A | Appendix A

In this appendix we want to discuss the relationship between the convergence of p^h with respect to the mesh size and the definition of the stabilization parameters.
We know that:

$$p' \approx -\tau_C(\mathbf{u}^h) r_c(\mathbf{u}^h) \tag{A.1}$$

for the quasi-static approach we have:

$$\tau_C(\mathbf{u}^h) = \left(\frac{\sigma_t^2 \rho^2}{\Delta t^2} + \rho^2 \mathbf{u}^h \cdot \overline{\mathbf{G}} \mathbf{u}^h + C_r \mu^2 \overline{\mathbf{G}} : \overline{\mathbf{G}} \right)^{\frac{1}{2}} (\overline{\mathbf{g}} \cdot \overline{\mathbf{g}})^{-1} \tag{A.2}$$

for the dynamic we have:

$$\widetilde{\tau_C}(\mathbf{u}^h) = \left(\rho^2 \mathbf{u}^h \cdot \overline{\mathbf{G}} \mathbf{u}^h + C_r \mu^2 \overline{\mathbf{G}} : \overline{\mathbf{G}} \right)^{\frac{1}{2}} (\overline{\mathbf{g}} \cdot \overline{\mathbf{g}})^{-1} \tag{A.3}$$

with:

$$r_C(\mathbf{u}^h) = \nabla \cdot \mathbf{u}^h \tag{A.4}$$

being $\Delta t \ll 1$

$$\frac{\sigma_t^2 \rho^2}{\Delta t^2} \gg C r \mu^2 \tag{A.5}$$

moreover, the inequality is stronger if $Re = 1000$ since $\rho \gg \mu$.
Therefore, we can neglect the term $C_r \mu^2 \overline{\mathbf{G}} : \overline{\mathbf{G}}$ both in the quasi - static and in the dynamic methods, such that for the quasi-static we have:

$$p' \sim \left(\frac{\sigma_t^2 \rho^2}{\Delta t^2} + \rho^2 \mathbf{u}^h \cdot \overline{\mathbf{G}} \mathbf{u}^h \right)^{\frac{1}{2}} \nabla \mathbf{u}_h \, (\overline{\mathbf{g}} \cdot \overline{\mathbf{g}})^{-1} \tag{A.6}$$

and for the dynamic:

$$p' \sim \rho \left(\mathbf{u}^h \cdot \overline{\mathbf{G}} \mathbf{u}^h \right)^{\frac{1}{2}} \nabla \cdot \mathbf{u}^h \tag{A.7}$$

Let us derive the polynomial degree of p' from equations (A.6) and (A.7). Assuming elements of degree n for $\mathbf{u}^h$ we approximate $\mathbf{u}^h \sim \mathcal{O}(h^n)$. We neglect the operators $\overline{\mathbf{G}}$ and $\overline{\mathbf{g}}$ since they are purely geometrical transformations which do not affect the order of

p' and we define the constant $K^2 = \dfrac{\sigma_t^2 \rho^2}{\Delta t^2}$.

Accordingly, from equations (A.6) we have:

$$\begin{aligned} p' \quad &\sim \left(K^2 + \rho^2 \mathcal{O}(h^n) \cdot \mathcal{O}(h^n)\right)^{\frac{1}{2}} \mathcal{O}(h^{n-1}) \\ &\sim \left(K^2 + \rho^2 \mathcal{O}(h^{2n})\right)^{\frac{1}{2}} \ \mathcal{O}(h^{n-1}) \end{aligned} \tag{A.8}$$

and from equation (A.7):

$$\begin{aligned} p' \quad &\sim \left(\rho^2 \ \mathcal{O}(h^n) \cdot \mathcal{O}(h^n)\right)^{\frac{1}{2}} \mathcal{O}(h^{n-1}) \\ &\sim \rho \ \mathcal{O}(h^n)\mathcal{O}(h^{n-1}) \\ &\sim \rho \ \mathcal{O}(h^{2n-1}) \end{aligned} \tag{A.9}$$

If we consider a mesh-refinement analysis, h tent to zero. Therefore, we can neglect the second term of the square root in equation (A.8) obtaining the following expression:

$$p' \sim K \ \mathcal{O}(h^{n-1}) \tag{A.10}$$

According to the VMS-LES theory:

$$p_{REF} = p^h + p' \tag{A.11}$$

where p' represent the degree of uncertainty and p_{REF} is the reference, i.e. the analytical, solution. By reducing the number of refinement, the finite element solution (i.e. coarse scale solution) should be more precise, such that for $h \to 0$, $p^h \to p_{REF}$. However, according to equation (A.11), the rate of convergence of p^h may be affected by the asymptotic behaviour p', which in equations (A.10) and (A.9) changes by changing the number of Reynolds and moving from a quasi-static to a dynamic approach.
In this regard, the definition of the stabilization parameters and the fine scale approximation, may interact with the convergence of the method, and the weight of this effect is influenced by the Reynolds number.

List of Figures

List of Tables

Acknowledgements

I want to express my gratitude to Prof. Alfio Quarteroni and Prof. Luca Dedé, who gave me the chance to start this thesis and to get into the world of mathematical models and to Prof. Christian Vergara for making available the geometry of the stenotic carotid. I also want to deeply thank Dr. Alberto Zingaro and Lorenzo Bennati for the precious advices and the support against my countless issues with the simulations, at every time of the night. In the end, I want to thank Duccio who was always close to me when the simulations were crashing, trying to do his best to make me laugh again.