Technical Report

TR-2015-003

Deconvolution-based nonlinear filtering for incompressible flows at moderately large Reynolds numbers

by

L. Bertagna, A. Quaini, A. Veneziani

Mathematics and Computer Science

EMORY UNIVERSITY

Deconvolution-based nonlinear filtering for incompressible flows at moderately large Reynolds numbers

L. Bertagna[1‡], A. Quaini[2*], A. Veneziani[1]

[1]*Department of Mathematics and Computer Science, Emory University, Atlanta (GA) USA*
[2]*Department of Mathematics, University of Houston, Houston (TX) USA*

SUMMARY

We consider a Leray model with a deconvolution-based indicator function for the simulation of incompressible fluid flow at moderately large Reynolds number (in the range of few thousand) with under-resolved meshes. For the implementation of the model, we adopt a three-step algorithm called *evolve-filter-relax* (EFR) that requires (i) the solution of a Navier-Stokes problem, (ii) the solution of a Stokes-like problem to filter the Navier-Stokes velocity field, and (iii) a final relaxation step. We take advantage of a reformulation of the EFR algorithm as an operator splitting method to analyze the impact of the filter on the final solution vs a direct simulation of the Navier-Stokes equations. In addition, we provide some direction for tuning the parameters involved in the model based on physical and numerical arguments. Our approach is validated against experimental data for fluid flow in an idealized medical device (consisting of a conical convergent, a narrow throat, and a sudden expansion, as recommended by the U.S. Food and Drug Administration). Numerical results are in good quantitative agreement with the measured axial components of the velocity and pressures for two different flow rates corresponding to turbulent regimes, even for meshes with a mesh size more than 40 times larger than the smallest turbulent scale. After several numerical experiments, we perform a preliminary sensitivity analysis of the computed solution to the parameters involved in the model.

Received ...

KEY WORDS: Computational incompressible fluid dynamics; Leray model; nonlinear filtering; approximate deconvolution; experimental validation

1. INTRODUCTION

The Incompressible Navier-Stokes equations (INSE) feature several challenging aspects, e.g. the saddle-point nature of the problem and nonlinearity, that make their analysis difficult [48]. When numerical methods are used, these challenging features may result in restriction to the possible discretization settings. For instance, the saddle-point nature of the problem leads to the well-known discrete *inf-sup* (or LBB) condition [12], which enforces a restriction on how to approximate fluid velocity and pressure. Additional difficulties are peculiar to the discrete framework, such as instabilities in convection dominated flow. These can be avoided by restrictions on the discretization parameters or by adopting suitable stabilization techniques (e.g., [13]).

As well known, from the phenomenology point of view, strong convective fields compared with viscous forces may trigger flow disturbances up to turbulence. This introduces other numerical challenges. When the Reynolds number - the dimensionless number that weighs the importance of inertial forces vs viscous ones - increases, the flow is characterized by the presence of flow structures

*Correspondence to: 4800 Calhoun Rd. Houston, TX, 77004, E-mail: quaini@math.uh.edu
‡Current affiliation: Department of Scientific Computing, Florida State University, Tallahassee (FL) USA

over a large variety of space scales. Numerical discretization needs to capture all these structures to give an accurate description of the flow. Unfortunately, as the Reynolds number increases, this demands for very fine space reticulations that quickly lead to large algebraic systems. Therefore, this approach (Direct Numerical Simulation - DNS) may be computationally unaffordable.

A possible way to keep an affordable size of the discrete problem without sacrificing the accuracy is to solve the flow average and model properly the effects of the small scales (not directly solved) at the medium and large scales. The Navier-Stokes equations can be averaged in time, leading to the so called Reynolds-Averaged Navier-Stokes equations (RANS), or in space, leading to Large Eddy Simulation (LES) techniques (see, e.g., [17]). In this article, we focus on the latter approach. We consider a variant of the so called *Leray model* [33], where small scale effects are described by a set of equations to be added to the discrete INSE formulated on the unresolved mesh. The extra-problem can be devised in different ways, for instance by a functional splitting of the solved and unresolved scales [5] or by resorting to the concept of "suitability" of weak solutions [26].

The variant of the Leray model we deal with was originally proposed in [9]. Here the extra problem acts as a differential low-pass filter. For its actual implementation, we use the evolve-filter-relax (EFR) algorithm presented in [32]. One of the advantages of this approach is that it is easily implemented in a legacy Navier-Stokes solver, since the filtering step requires the solution of a Stokes-like problem.

We target specifically applications involving incompressible fluid flow at moderately large Reynolds numbers (up to 5000). We reformulate the method of [9, 32] in an operator splitting framework for the INSE. In addition, we provide practical directions to tune the parameters involved in the model based on physical and numerical arguments. We test our reformulation and the parameter tuning on a realistic 3D problem, i.e. flow at different Reynolds numbers through a nozzle which contains all the features commonly encountered in medical devices (flow contraction and expansion, recirculation zones etc., see Figure 1). We selected this problem because it is part of a benchmark issued by the U.S. Food and Drug Administration (FDA) within the "Critical Path Initiative" program [2]. Three independent laboratories were requested by FDA to perform flow visualization experiments on fabricated nozzle models for different flow rates up to Reynolds number 6500 [27]. This resulted in benchmark data available online to the scientific community for the validation of Computational Fluid Dynamics (CFD) simulations [1]. Available experimental measurements enable us to check the effectiveness of the EFR algorithm in simulating average macroscopic quantities. Before the present paper, the EFR algorithm has been applied to academic problems. Here, it is applied for the first time to a realistic problem of practical interest, featuring non-Dirichlet boundary conditions, as usually is the case in engineering applications.

Results of a first CFD study of the FDA nozzle model are reported in [45]. Several groups of CFD professionals participated in the study, following different modeling approaches (turbulence models vs. direct numerical simulations, choice of the boundary conditions, etc.). Overall, the results obtained by different groups had a very large variability also with respect to the experimental results. It was observed that RANS turbulence models were in general unable to correctly estimate the centerline velocities in the throat of the nozzle and downstream of the sudden expansion. None of the participants in the study used a LES approach. Thanks to EFR algorithm and an appropriate choice for the parameter in the Leray model, we manage to achieve good agreement between the computed velocities and pressures and the respective measured quantities, even for meshes with a size more than 40 times larger than the smallest turbulent scale. Moreover, through a large set of numerical experiments, we show the impact of the parameters involved in the model on the computed solution.

The paper is organized as follow. In Section 2, we introduce the continuous Leray model as well as the numerical approximation proposed in [32]. In Section 3, we present possible choices for the so called *indicator function*, a fundamental component of the Leray model. In Section 4, we formulate the algorithm as an operator splitting method. We consider specifically practical problems with any kind of physically relevant boundary conditions. In addition we show how certain parameters of the model can be related to physical and discretization quantities. In Section 5, we describe the details of the discretization and the preconditioners used to solve the resulting linear systems.

The numerical results for the FDA benchmark and the comparison with the experimental data are reported in Section 6. Conclusions are drawn in Section 7.

Software and implementation All the computational results presented in this article have been performed with LifeV [3], an open source library of algorithms and data structures for the numerical solution of partial differential equations with high performance computing techniques. In [37], we validated one Navier-Stoker solver implemented in LifeV with DNS up to Reynolds number 3500 and showed that a properly refined mesh is able to capture accurately the average flow features observed in the experiments. The LES technique used in this article makes the simulation of higher Reynolds numbers computationally affordable without sacrificing accuracy. An important outcome of this work is that the code created for it is incorporated in an open-source library and therefore is readily shared with the community.

2. PROBLEM DEFINITION

2.1. The Navier-Stokes equations

We consider the motion of an incompressible viscous fluid in a time-independent domain Ω over a time interval of interest (t_0, T). The flow is described by the incompressible Navier-Stokes equations

$$\rho\,\partial_t \boldsymbol{u} + \rho\,(\boldsymbol{u}\cdot\nabla)\boldsymbol{u} - \nabla\cdot\boldsymbol{\sigma} = \boldsymbol{f} \text{ in } \Omega\times(t_0,T), \tag{1}$$

$$\nabla\cdot\boldsymbol{u} = 0 \text{ in } \Omega\times(t_0,T), \tag{2}$$

endowed with the boundary and initial conditions

$$\boldsymbol{u} = \boldsymbol{u}_D \text{ on } \partial\Omega_D\times(t_0,T), \tag{3}$$

$$\boldsymbol{\sigma}\cdot\boldsymbol{n} = \boldsymbol{g} \quad \text{on } \partial\Omega_N\times(t_0,T), \tag{4}$$

$$\boldsymbol{u} = \boldsymbol{u}_0 \quad \text{in } \Omega\times\{t_0\},$$

where $\overline{\partial\Omega_D}\cup\overline{\partial\Omega_N} = \overline{\partial\Omega}$ and $\partial\Omega_D\cap\partial\Omega_N = \emptyset$. Here ρ is the fluid density, $\boldsymbol{u}$ is the fluid velocity, ∂_t denotes the time derivative, $\boldsymbol{\sigma}$ is the Cauchy stress tensor, $\boldsymbol{f}$ accounts for possible body forces (such as, e.g., gravity), $\boldsymbol{u}_D, \boldsymbol{g}$ and $\boldsymbol{u}_0$ are given. Equation (1) represents the conservation of the linear momentum, while eq. (2) represents the conservation of the mass. For Newtonian fluids $\boldsymbol{\sigma}$ can be written as

$$\boldsymbol{\sigma}(\boldsymbol{u},p) = -p\mathbf{I} + \mu(\nabla\boldsymbol{u} + \nabla\boldsymbol{u}^T), \tag{5}$$

where p is the pressure and μ is the constant (dynamic) viscosity. Let us introduce operators ∇^s and Δ^s, defined as

$$\nabla^s\boldsymbol{u} \equiv \frac{1}{2}(\nabla\boldsymbol{u} + \nabla\boldsymbol{u}^T) \qquad \Delta^s\boldsymbol{u} \equiv \nabla\cdot\nabla^s\boldsymbol{u}. \tag{6}$$

where ∇^s is the symmetric part of the gradient. For completeness, we also define operator

$$\nabla^{ss}\boldsymbol{u} \equiv \nabla\boldsymbol{u} - \nabla^s\boldsymbol{u} \equiv \frac{1}{2}(\nabla\boldsymbol{u} - \nabla\boldsymbol{u}^T), \tag{7}$$

that we will use later on.

The incompressible Navier-Stokes equations can be rewritten as

$$\rho\,\partial_t \boldsymbol{u} + \rho\,(\boldsymbol{u}\cdot\nabla)\boldsymbol{u} + \nabla p - 2\mu\,\Delta^s\boldsymbol{u} = \boldsymbol{f} \text{ in } \Omega\times(t_0,T), \tag{8}$$

$$\nabla\cdot\boldsymbol{u} = 0 \text{ in } \Omega\times(t_0,T). \tag{9}$$

Remark 2.1
In the continuous formulation, $\nabla \cdot \nabla \boldsymbol{u}^T = \nabla(\nabla \cdot \boldsymbol{u}) = 0$, due to the continuity equation. Therefore, the Navier-Stokes equations are often formulated with the operator Δ rather than Δ^s. However, the contribution of the term $\nabla \boldsymbol{u}^T$ does not vanish when problem (1)-(2) is formulated in its *weak* form - as done in the finite element approximation [25]. We retain therefore this term for the formal correctness of our formulation. This term is particularly important e.g. in Fluid-Structure Interaction problems where we have continuity of the normal stress at the interface between fluid and structure.

To characterize the flow regime under consideration, we define the Reynolds number as

$$Re = \frac{UL}{\nu}, \tag{10}$$

where $\nu = \mu/\rho$ is the *kinematic* viscosity of the fluid, and U and L are characteristic macroscopic velocity and length respectively. For an internal flow in a cylindrical pipe, U is the mean sectional velocity and L is the diameter. For large Reynolds numbers, inertial forces are dominant over viscous forces and vice versa. For moderately large Reynolds numbers the effects of flow disturbances cannot be neglected, and yet Reynolds-averaged Navier-Stokes (RANS) models [38] are generally too crude.

2.2. *Leray model*

To motivate the introduction of the Leray model, it is useful to look at the behavior of the turbulent kinetic energy (TKE) of the fluid, which is the kinetic energy associated with eddies in the turbulent flow. In the framework of the Kolmogorov 1941 (K41) theory [31, 30], the TKE is injected in the system at the large scales (low wave numbers). Since the large scale eddies are unstable, they break down, transferring the energy to smaller eddies. Finally, the TKE is dissipated by the viscous forces at the small scales (high wave numbers). This process is usually referred to as *energy cascade*. The scale at which the viscous forces dissipate energy is referred to as *Kolmogorov scale* and can be expressed as

$$\eta = \left(\frac{\nu^3}{\varepsilon}\right)^{1/4}, \tag{11}$$

where ε is the time-average of the rate at which the energy is dissipated (see e.g. [20]). Formally, ε is defined as

$$\varepsilon := \limsup_{T\to\infty} \frac{1}{T|\Omega|} \int_{t_0}^{T} \nu ||\nabla \boldsymbol{u}||_{L^2}^2 dt.$$

For a flow in developed turbulent regime, the dissipation rate has to be of the same magnitude of the production rate, which is the rate at which the TKE is supplied to the small scales. A common way to express ε in terms of the macro-scale variables is $\varepsilon \sim U^3/L$ [49], leading to the expression

$$\eta = Re^{-3/4} L. \tag{12}$$

This scaling law pinpoints the difficulty of the numerical solution of the Navier-Stokes equation at high Reynolds numbers. In order to correctly capture the dissipated energy, DNS needs a grid with spacing $h \sim \eta$. As the Reynolds number increases, DNS leads to a huge number of unknowns and prohibitive computational costs. On the other hand, when the mesh size h fails to resolve the Kolmogorov scale, the under-diffusion in the simulation leads to nonphysical computed velocities. In some cases, this is detectable simply looking at the velocity field (see [9]), which features nonphysical oscillations. However, as we shall see in Section 6, in some cases the velocity field does not display oscillations, yet it does not correspond to the physical solution (see, e.g., Fig. 11(a), where the velocity computed by DNS in not oscillating, but it grossly underestimates the measured jet length). A possible remedy to this issue is to introduce a model which filters the velocity and conveys the energy lost to resolved scales. Such models can be thought as a way to numerically revert the energy cascade by transferring the energy dissipated at the small (unresolved) scales towards larger (resolved) scales.

The "Leray model" couples the Navier-Stokes equations with a differential filter. The model can be written as

$$\rho\,\partial_t \boldsymbol{u} + \rho\,(\overline{\boldsymbol{u}}\cdot\nabla)\boldsymbol{u} - 2\mu\Delta^s\boldsymbol{u} + \nabla p = \boldsymbol{f} \quad \text{in } \Omega\times(t_0,T), \tag{13}$$

$$\nabla\cdot\boldsymbol{u} = 0 \quad \text{in } \Omega\times(t_0,T), \tag{14}$$

$$-2\delta^2\nabla\cdot(a(\boldsymbol{u})\nabla^s\overline{\boldsymbol{u}}) + \overline{\boldsymbol{u}} + \nabla\lambda = \boldsymbol{u} \quad \text{in } \Omega\times(t_0,T), \tag{15}$$

$$\nabla\cdot\overline{\boldsymbol{u}} = 0 \quad \text{in } \Omega\times(t_0,T)., \tag{16}$$

Here, $\overline{\boldsymbol{u}}$ is the *filtered velocity*, δ can be interpreted as the *filtering radius* (that is, the radius of the neighborhood were the filter extracts information from the unresolved scales), the variable λ is a Lagrange multiplier to enforce the incompressibility constraint for $\overline{\boldsymbol{u}}$ and $a(\cdot)$ is a scalar function such that

$$a(\boldsymbol{u}) \simeq 0 \text{ where the velocity } \boldsymbol{u} \text{ does not need regularization}$$
$$a(\boldsymbol{u}) \simeq 1 \text{ where the velocity } \boldsymbol{u} \text{ does need regularization.}$$

This function, usually referred to as *indicator function*, is crucial for the success of the Leray model. In the next section, we will discuss some of the possible choices for $a(\cdot)$ that have been proposed. Here, we mention that the choice $a(\boldsymbol{u}) \equiv 1$ corresponds to the classic Leray-α model [33]. This choice has the advantage of making the operator in the filter equations linear and constant in time, but its effectivity is rather limited, since it introduces the same amount of regularization in every region of the domain, hence causing overdiffusion.

Equations (15)-(16) require suitable boundary conditions. These are chosen to be

$$\overline{\boldsymbol{u}} = \boldsymbol{u}_D \text{ on } \partial\Omega_D\times(t_0,T), \tag{17}$$

$$(2\delta^2 a(\boldsymbol{u})\nabla^s\overline{\boldsymbol{u}} - \lambda\mathbf{I})\boldsymbol{n} = \mathbf{0} \quad \text{on } \partial\Omega_N\times(t_0,T), \tag{18}$$

while no initial condition is required for $\overline{\boldsymbol{u}}$, since there is no time derivative in eq. (15)-(16). The impact of non-Dirichlet boundary conditions has not been investigated thoroughly in literature. In Section 4, we will discuss the effect of these boundary conditions on the solution of the problem.

Even though (13)-(14) are linear in $(\boldsymbol{u}, p)$ and the filter problem is linear in $(\overline{\boldsymbol{u}}, \lambda)$, the coupling is non-linear, due to the term $(\overline{\boldsymbol{u}}\cdot\nabla)\boldsymbol{u}$ in eq. (13), and $a(\boldsymbol{u})\nabla^s\overline{\boldsymbol{u}}$ in eq. (15) when $a(\cdot)$ is not constant.

Remark 2.2
The structure of the filter problem can be qualified as a generalized Stokes problem with a non-constant viscosity. A solver for the filter can then be obtained by adapting a standard linearized Navier-Stokes solver. This will be discussed in Section 5, where we will address preconditioning of the two saddle point problems.

2.3. Time discrete problem

To discretize in time problem (13)-(16), let $\Delta t \in \mathbb{R}$, $t^n = t_0 + n\Delta t$, with $n = 0, ..., N_T$ and $T = t_0 + N_T\Delta t$. Moreover, we denote by y^n the approximation of a generic quantity y at the time t^n. In the following we will denote by Ω the domain of the equations.

For the time discretization of system (13)-(16), we adopt a Backward Differentiation Formula of order p (BDFp), see e.g. [39]. The Leray system discretized in time reads: given $\boldsymbol{u}^0$, for $n \geq 0$ find the solution $(\boldsymbol{u}^{n+1}, p^{n+1}, \overline{\boldsymbol{u}}^{n+1}, \lambda^{n+1})$ of the system:

$$\rho\,\frac{\alpha}{\Delta t}\,\boldsymbol{u}^{n+1} + \rho\,\overline{\boldsymbol{u}}^{n+1}\cdot\nabla\boldsymbol{u}^{n+1} - 2\mu\Delta^s\boldsymbol{u}^{n+1} + \nabla p^{n+1} = \boldsymbol{b}^{n+1}, \tag{19}$$

$$\nabla\cdot\boldsymbol{u}^{n+1} = 0, \tag{20}$$

$$-2\delta^2\nabla\cdot\left(a(\boldsymbol{u}^{n+1})\nabla^s\overline{\boldsymbol{u}}^{n+1}\right) + \overline{\boldsymbol{u}}^{n+1} + \nabla\lambda^{n+1} = \boldsymbol{u}^{n+1}, \tag{21}$$

$$\nabla\cdot\overline{\boldsymbol{u}}^{n+1} = 0, \tag{22}$$

where α is a coefficient that depends on the order of BDF chosen, and $\boldsymbol{b}^{n+1}$ contains the forcing term $\boldsymbol{f}^{n+1}$ and the solution at the previous time steps used to approximate the time derivative of $\boldsymbol{u}$ at time t^{n+1}. For example, when using BDF2, we have

$$\partial_t \boldsymbol{u} \simeq \frac{3\boldsymbol{u}^{n+1} - 4\boldsymbol{u}^n + \boldsymbol{u}^{n-1}}{2\Delta t}, \tag{23}$$

thus $\alpha = 3/2$ and $\boldsymbol{b}^{n+1} = \boldsymbol{f}^{n+1} + (4\boldsymbol{u}^n - \boldsymbol{u}^{n-1})/(2\Delta t)$.

A monolithic approach for problem (19)-(22) would lead to high computational costs, making the advantage compared to DNS questionable. To decouple the Navier-Stokes system (19)-(20) from the filter system (21)-(22) at the time t^{n+1}, we have two options:

1. *Filter-then-solve*: Solve the filter equations (21)-(22) first, with $\boldsymbol{u}^{n+1}$ replaced by a suitable extrapolation $\boldsymbol{u}^*$ and $a(\boldsymbol{u}^{n+1})$ replaced by $a(\boldsymbol{u}^*)$, and then solve equations (19)-(20) with advection field given by the filtered velocity previously computed.

2. *Solve-then-filter*: Solve equations (19)-(20) first, replacing the advection field $\overline{\boldsymbol{u}}^{n+1}$ with a suitable extrapolation $\boldsymbol{u}^*$, and then solve the filter problem (21)-(22).

In either approach, one could iterate between the two subproblems, using a fixed point approach to the fulfillment of a certain convergence criterion. However, to keep the computational costs low, we adopt a semi-implicit approach, by performing only one iteration per time step. Approach 1 is investigated thoroughly in [9] using the linearly extrapolated Crank-Nicolson approximation $\boldsymbol{u}^* = (3\boldsymbol{u}^n - \boldsymbol{u}^{n-1})/2$. In this paper we will focus on approach 2 with a modified version of the EFR scheme proposed in [32]. The EFR algorithm reads as follows: given the velocities $\boldsymbol{u}^k$ ($k = n-p+1, \dots, n$) needed for the approximation of $\partial_t \boldsymbol{u}$ by BDFp at t^{n+1},

i) *evolve*: find intermediate velocity and pressure $(\boldsymbol{v}^{n+1}, q^{n+1})$ such that

$$\rho \frac{\alpha}{\Delta t} \boldsymbol{v}^{n+1} + \rho\, \boldsymbol{u}^* \cdot \nabla \boldsymbol{v}^{n+1} - 2\mu \Delta^s \boldsymbol{v}^{n+1} + \nabla q^{n+1} = \boldsymbol{b}^{n+1}, \tag{24}$$

$$\nabla \cdot \boldsymbol{v}^{n+1} = 0, \tag{25}$$

where $\boldsymbol{u}^*$ is a suitable approximation of the end-of-step velocity $\boldsymbol{u}^{n+1}$ based on previous time-steps solutions.

ii) *filter*: find $(\overline{\boldsymbol{v}}^{n+1}, \lambda^{n+1})$ such that

$$-2\delta^2 \nabla \cdot \left(a(\boldsymbol{v}^{n+1}) \nabla^s \overline{\boldsymbol{v}}^{n+1}\right) + \overline{\boldsymbol{v}}^{n+1} + \nabla \lambda^{n+1} = \boldsymbol{v}^{n+1}$$
$$\nabla \cdot \overline{\boldsymbol{v}}^{n+1} = \nabla \cdot \boldsymbol{v}^{n+1} = 0$$

iii) *relax*: set

$$\boldsymbol{u}^{n+1} = (1-\chi)\boldsymbol{v}^{n+1} + \chi \overline{\boldsymbol{v}}^{n+1}, \tag{26}$$
$$p^{n+1} = q^{n+1},$$

where $\chi \in [0, 1]$ is a relaxation parameter.

In [32] the authors use energy arguments to support the choice $\chi = O(\Delta t)$, in order to keep the numerical dissipation low. The EFR algorithm has the advantage of modularity: being the problems at steps i) and ii) numerically standard, they can be solved with legacy Navier-Stokes solver. It was shown in [36] that the EFR algorithm is equivalent to a certain viscosity model in Large Eddy Simulation.

3. INDICATOR FUNCTION FOR NONLINEAR FILTERS: PHYSICS VS MATHEMATICS BASED

The breaking down of eddies into smaller ones until they get damped is a highly nonlinear process, however most current models use linear filters to select the eddies to be damped. Linear filter based stabilization, developed by Boyd [10] and Fischer and Mullen [18, 35], has been widely studied over the past years (see, e.g., [34, 50, 21]). Nonlinear filters have been considered in [32], where damping of eddies is based on nonlinearity of real flow problems.

The success of nonlinear filters such as (13)-(16) in simulations ultimately depends on the reliability of the indicator function. One of the most mathematically convenient indicator function is $a(\boldsymbol{u}) = |\nabla \boldsymbol{u}|$ (suitably normalized [7]) because of its strong monotonicity properties. With this choice, we recover a *Smagorinsky-like model*, which is however known to be not sufficiently selective. In fact, it selects laminar shear flow (where $|\nabla \boldsymbol{u}|$ is constant but large) as regions of the domain with large turbulent fluctuations.

In the following we report on some indicator functions that have been proposed in the literature. We group them into two categories: physical phenomenology based and deconvolution based. We highlight their strengths and limitations.

3.1. Physical phenomenology based indicator functions

The indicator functions proposed in [32] are based on physical quantities that are known to vanish for coherent flow structures. One of the most popular methods for eduction of coherent vortices is the Q criterion [29], which identifies as persistent and coherent vortex those regions where:

$$Q(\boldsymbol{u},\boldsymbol{u}) = \frac{1}{2}(\nabla^{ss}\boldsymbol{u} : \nabla^{ss}\boldsymbol{u} - \nabla^{s}\boldsymbol{u} : \nabla^{s}\boldsymbol{u}) > 0,$$

where the spin tensor $\nabla^{ss}\boldsymbol{u}$ and the deformation tensor $\nabla^{s}\boldsymbol{u}$ were defined in (7) and (6), respectively. From the definition it is apparent that $Q > 0$ occurs in those regions where spin dominates deformation. An indicator is obtained by rescaling $Q(\boldsymbol{u},\boldsymbol{u})$ so that the condition $Q(\boldsymbol{u},\boldsymbol{u}) > 0$ implies $a(\boldsymbol{u}) \simeq 0$, that is, regularization is not needed. The indicator function based on the Q-criterion is then given by

$$a_Q(\boldsymbol{u}) = \frac{1}{2} - \frac{1}{\pi}\arctan\left(\delta^{-1}\frac{Q(\boldsymbol{u},\boldsymbol{u})}{|Q(\boldsymbol{u},\boldsymbol{u})| + \delta^2}\right).$$

A second indicator uses an eddy viscosity coefficient formula proposed by Vreman [51] that vanishes for 320 types of flow structures known to be coherent. The Vreman based indicator function reads

$$a_V(\boldsymbol{u}) = \sqrt{\frac{B(\boldsymbol{u})}{|\nabla \boldsymbol{u}|_F^4}},$$

where the subindex F refers to the Frobenius norm and $B(\boldsymbol{u})$ is defined as

$$B = \beta_{11}\beta_{22} - \beta_{12}^2 + \beta_{11}\beta_{33} - \beta_{13}^2 + \beta_{22}\beta_{33} - \beta_{23}^2, \quad \beta_{ij}(\boldsymbol{u}) = \sum_{m=1,2,3}\frac{\partial u_i}{\partial x_m}\frac{\partial u_j}{\partial x_m}.$$

Since $0 \le B(\boldsymbol{u})/|\nabla \boldsymbol{u}|_F^4 \le 1$, $a_V(\boldsymbol{u}) \in [0,1]$. The Vreman based indicator function was shown to be successful in [8].

Another physics based indicator function uses the relative helicity density RH, which is a local quantity, its macroscopic counterpart being the helicity H. The two quantities H and RH are defined respectively as

$$H = \frac{1}{|\Omega|}\int_\Omega \boldsymbol{u}\cdot\boldsymbol{w}\, d\Omega, \quad RH = \frac{\boldsymbol{u}\cdot\boldsymbol{w}}{|\boldsymbol{u}||\boldsymbol{w}|},$$

with $\boldsymbol{w}$ denoting vorticity, i.e. $\boldsymbol{w} = \nabla \times \boldsymbol{u}$. From the Navier-Stokes equations in rotational form, it is possible to see that local high helicity suppresses local turbulent dissipation caused by breakdown of eddies into smaller ones. The helicity based indicator is developed by adjusting relative helicity density so that values of RH near one imply $a(\boldsymbol{u}) \simeq 0$. It reads

$$a_H(\boldsymbol{u}) = 1 - \left| \frac{\boldsymbol{u} \cdot \boldsymbol{w}}{|\boldsymbol{u}||\boldsymbol{w}| + \delta^2} \right|.$$

Notice that other (more selective) indicator functions can also be obtained by taking the geometric average of two (or more) indicator functions.

The indicator functions discussed in this section have the advantage of requiring only algebraic operations on $\boldsymbol{u}$ and its derivatives. Their implementation may be quite straightforward. However, the major drawback is that they do not allow for a rigorous convergence theory to verify the robustness of the associated filtering method.

3.2. Deconvolution based indicator functions

Let F be a linear, invertible, self-adjoint, compact operator from a Hilbert space V (such as $H^1(\Omega)$ or $H^1_0(\Omega)$) to itself. The spectral theorem gives (see, for instance, [11])

$$F\mathbf{x} = \sum_{i=0}^{\infty} \lambda_i \langle \mathbf{x}, \mathbf{e}_i \rangle \mathbf{e}_i, \qquad F^{-1}\mathbf{y} = \sum_{i=0}^{\infty} \frac{1}{\lambda_i} \langle \mathbf{y}, \mathbf{e}_i \rangle \mathbf{e}_i,$$

where λ_i are the eigenvalues of F, and $\mathbf{e}_i$ are the corresponding eigenfunctions, which form an orthonormal basis for V.

Since F is compact, the inverse operator F^{-1} (i.e. the operator s.t. $\mathbf{x} = F^{-1}F\mathbf{x}$) is unbounded. Let D be a bounded regularized approximation of F^{-1}, whose action on $\mathbf{y}$ is given by

$$D\mathbf{y} = \sum_{i=0}^{\infty} \phi\left(\frac{1}{\lambda_i}\right) \langle \mathbf{y}, \mathbf{e}_i \rangle \mathbf{e}_i$$

with

$$\phi\left(\frac{1}{\lambda_i}\right) \simeq \begin{cases} \dfrac{1}{\lambda_i} & \text{if } i \text{ is small,} \\ 0 & \text{if } i \text{ is large.} \end{cases}$$

Then

$$\|\mathbf{x} - DF\mathbf{x}\| \text{ is } \begin{cases} \text{small} & \text{if } \mathbf{x} \text{ is ``smooth'',} \\ \text{large} & \text{if } \mathbf{x} \text{ is not ``smooth'',} \end{cases}$$

where "smooth" is intended with respect to the eigenfunction of the operator F. In particular, $\mathbf{x}$ is smooth if $\langle \mathbf{x}, \mathbf{e}_i \rangle$ is significantly different from zero only for small values of i. The composition of the two operators F and D can be interpreted as a low-pass filter. This motivates the indicator function

$$a_D(\boldsymbol{u}) = |\boldsymbol{u} - D(F(\boldsymbol{u}))|. \tag{27}$$

3.3. The Van Cittert deconvolution operator

A popular choice for D is the Van Cittert deconvolution operator D_N, defined as

$$D_N = \sum_{n=0}^{N} (I - F)^n.$$

The evaluation of a_D with $D = D_N$ (deconvolution of order N) requires then to apply the filter F a total of $N + 1$ times. If $I - F$ is spectrally bounded by 1, then D_N can be seen as the truncated

Neumann expansion of F^{-1}, and D_N approaches F^{-1} as $N \to \infty$. However, this does not introduce a limitation for the application of the method. Since F^{-1} is not bounded, in practice N is chosen to be small, as the result of a trade-off between accuracy (for a regular solution) and filtering (for a non-regular one). In this paper we consider $N = 0, 1$, corresponding to $D_0 = I$ and $D_1 = 2I - F$. For these choices of N, the indicator function (27) becomes

$$a_{D_0}(\boldsymbol{u}) = |\boldsymbol{u} - F(\boldsymbol{u})|\,, \quad a_{D_1}(\boldsymbol{u}) = |\boldsymbol{u} - 2F(\boldsymbol{u}) + F(F(\boldsymbol{u}))|\,. \tag{28}$$

Remark 3.1
In order to ensure that $a(\boldsymbol{u}) \in [0, 1]$, indicator function (27) is rescaled

$$a_D(\boldsymbol{u}) = \frac{|\boldsymbol{u} - D(F(\boldsymbol{u}))|}{\max(1, ||\boldsymbol{u} - D(F(\boldsymbol{u}))||_\infty)}. \tag{29}$$

Notice that the Van Cittert deconvolution D_N can also be interpreted also as the N-th iteration of a Richardson scheme to solve the problem $F(\boldsymbol{u}) = \boldsymbol{b}$. In fact, letting $\boldsymbol{u}_N = D_N(\boldsymbol{b})$, we have

$$\begin{aligned} \boldsymbol{u}_{N+1} &= \sum_{n=0}^{N+1} (I - F)^n(\boldsymbol{b}) = \boldsymbol{b} + \sum_{n=1}^{N+1} (I - F)^n(\boldsymbol{b}) = \\ &= \boldsymbol{b} + (I - F) \sum_{n=0}^{N} (I - F)^n(\boldsymbol{b}) = \boldsymbol{b} + (I - F)(\boldsymbol{u}_N) = \boldsymbol{u}_N + (\boldsymbol{b} - F(\boldsymbol{u}_N)). \end{aligned} \tag{30}$$

From this perspective, we can further support the choice of limiting N. In fact, since the operator F is compact, its eigenvalues accumulate to 0 and its inverse leads to an ill-posed problem. Among the possible regularization techniques used to deal with ill-posed problem, one is precisely to use iterative methods (such as Richardson) with a limited number of iterations. The Van Cittert deconvolution can be thus regarded as an iterative regularization of the inverse problem $F(\boldsymbol{u}) = \boldsymbol{b}$.

3.3.1. The Van Cittert-Helmholtz operator We select F to be the linear Helmholtz filter operator F_H [23] defined by

$$F = F_H \equiv \left(I + \delta^2 \mathcal{L}\right)^{-1}, \qquad \mathcal{L} = -\Delta \equiv -\sum_{i=1}^{3} \frac{\partial^2}{\partial x_j^2}.$$

It is possible to prove [15] that

$$\phi - D_N(F_H(\phi)) = \delta^{2N+2} \mathcal{L}^{N+1} F_H^{N+1} \phi = (-1)^{N+1} \delta^{2N+2} \Delta^{N+1} F_H^{N+1} \phi. \tag{31}$$

Therefore, $a_{D_N}(\boldsymbol{u})$ is close to zero in the regions of the domain where $\boldsymbol{u}$ is smooth. Indicator function (27) with $D = D_N$ and $F = F_H$ has been recently proposed in [9], however the idea of using van Cittert approximate deconvolution in fluid models to increase accuracy is well established and mathematically grounded [46, 47, 15].

Notice that the Van Cittert-Helmholtz deconvolution operator D_N can be conveniently interpreted in a different manner, in view of the theory of maximal monotone operators and their Yosida regularized operator (see [11], Ch. 7). Following this theory, F_H is the *resolvent* J_{δ^2} of $\mathcal{L}$. Correspondingly, the *Yosida approximation* (or *regularization*) of $\mathcal{L}$ reads

$$\mathcal{L}_{\delta^2} \equiv \delta^{-2} \left(I - J_{\delta^2}\right) = \delta^{-2} \left(I - F_H\right)$$

If we let $\boldsymbol{a}_N(\boldsymbol{u}) = \boldsymbol{u} - D_N(F_H(\boldsymbol{u}))$, so that $a_{D_N}(\boldsymbol{u}) = |\boldsymbol{a}_N(\boldsymbol{u})|$, from the recurrence equation (30) it follows that

$$\boldsymbol{a}_N(\boldsymbol{u}) = (I - F_H)\boldsymbol{a}_{N-1}(\boldsymbol{u}) = \ldots = (I - F_H)^N \boldsymbol{a}_0(\boldsymbol{u}) = \delta^{2N+2} \mathcal{L}_{\delta^2}^{N+1} \boldsymbol{u}$$

or, in terms of the (scalar) indicator function $a_{D_N}(\boldsymbol{u})$,

$$a_{D_N}(\boldsymbol{u}) = \delta^{2N+2} \left| \mathcal{L}_{\delta^2}^{N+1} \boldsymbol{u} \right| . \tag{32}$$

Here, we list some properties we infer from Proposition 7.2 in reference [11]. Let $\boldsymbol{v}$ be a generic function in $H^1(\Omega)$. Then,

1. $\mathcal{L}_{\delta^2}\boldsymbol{v} = \mathcal{L}J_{\delta^2}\boldsymbol{v}$
2. $(\mathcal{L}_{\delta^2}\boldsymbol{v}, \boldsymbol{v}) \geq 0$
3. $\lim\limits_{\delta^2 \to 0} J_{\delta^2}\boldsymbol{v} = \boldsymbol{v}$
4. $\delta^2 |\mathcal{L}_{\delta^2}\boldsymbol{v}| \leq |\boldsymbol{v}|$.

Notice that (31) follows promptly from 1. In addition, if $\boldsymbol{v} \in H^2(\Omega)$ we also have

5. $\mathcal{L}_{\delta^2}\boldsymbol{v} = J_{\delta^2}\mathcal{L}\boldsymbol{v}$
6. $|\mathcal{L}_{\delta^2}\boldsymbol{v}| \leq |\mathcal{L}\boldsymbol{v}|$
7. $\lim\limits_{\delta^2 \to 0} \mathcal{L}_{\delta^2}\boldsymbol{v} = \mathcal{L}\boldsymbol{v} \quad (\Rightarrow \lim\limits_{\delta^2 \to 0} \delta^2 \mathcal{L}_{\delta^2}\boldsymbol{v} = 0)$.

Assuming enough regularity for $\boldsymbol{v}$, by induction it is possible to prove that the statement in brackets at point 7 holds for any $N > 0$, more precisely $a_{D_N}(\boldsymbol{u}) = \delta^{2N+2}|\mathcal{L}_{\delta^2}^{N+1}(\boldsymbol{u})|$ vanishes as the filter radius approaches zero.

4. NONLINEAR FILTERING AS AN OPERATOR SPLITTING METHOD

In this Section we consider the EFR scheme introduced in Section 2.3, with the indicator function defined in (29), $D = D_N$ and $F = F_H$, and we propose an operator splitting interpretation. In Section 4.1, we investigate how filtering affects the boundary conditions actually fulfilled by $\boldsymbol{u}^{n+1}$ and p^{n+1}, with particular attention to Neumann conditions. In Section 4.2 we propose a heuristic physical argument to choose the relaxation parameter χ.

For the space discretization we use the Finite Element method. For a given mesh, we introduce the discretization parameter h, which we take to be the length of the shortest edge in the mesh. In particular, we use an *inf-sup* stable Finite Element pair (e.g. Taylor-Hood elements [42]). More details concerning space discretization will be covered in section 5. We denote by $\boldsymbol{f}_h$ the discrete FE approximation of a generic continuous function $\boldsymbol{f}$ accordingly.

To rewrite the EFR scheme as an operator splitting method, let us define $\boldsymbol{u}_h^*$ a suitable approximation of the end-of-step velocity $\boldsymbol{u}_h^{n+1}$ extrapolated from previous time-steps. Let us then denote by $(\overline{\boldsymbol{v}}_h^{n+1}, \overline{q}_h^{n+1})$ the solution to the filter step. In addition we set

$$\overline{\mu}_h = \rho \frac{\delta^2}{\Delta t} a_{D_N}(\boldsymbol{v}_h^{n+1}) = \rho \frac{\delta^{2N+4}}{\Delta t} |\mathcal{L}_{\delta^2}^{N+1} \boldsymbol{v}_h^{n+1}|. \tag{33}$$

Here, the parameter δ is chosen to be of the order of h. Notice that $\overline{\mu}_h$ is dimensionally a dynamic viscosity, a fact that we will exploit in Section 4.2 to establish a practical rule for the selection of the relaxation parameter χ. Finally, we introduce the operators

$$\mathscr{L}_{NS}[\boldsymbol{u}_h^*]\boldsymbol{v}_h = \rho(\boldsymbol{u}_h^* \cdot \nabla)\boldsymbol{v}_h - \nabla \cdot (2\mu \nabla^s \boldsymbol{v}_h), , \tag{34}$$

$$\mathscr{L}_F[\boldsymbol{v}_h]\overline{\boldsymbol{v}}_h = -\nabla \cdot (2\overline{\mu}_h(\overline{\boldsymbol{v}}_h)\nabla^s \overline{\boldsymbol{v}}_h). \tag{35}$$

Here, the notation $A[\boldsymbol{v}]\boldsymbol{u}$ means that the operator A is computed at $\boldsymbol{v}$ and then applied to the function $\boldsymbol{u}$. Notice that the operator $\mathscr{L}_F$ depends on $\boldsymbol{v}_h$ through the artificial viscosity $\overline{\mu}_h$ in (33).

Let $(\boldsymbol{v}_h^{n+1}, q_h^{n+1})$ be the solution of the *evolve* step, that reads

$$\rho\frac{\alpha}{\Delta t}\boldsymbol{v}_h^{n+1} + \mathscr{L}_{NS}[\boldsymbol{u}_h^*]\boldsymbol{v}_h + \nabla q_h^{n+1} = \boldsymbol{b}_h^{n+1}, \tag{36}$$

$$\nabla\cdot\boldsymbol{v}_h^{n+1} = 0, \tag{37}$$

$$\boldsymbol{v}_h^{n+1} = \boldsymbol{u}_D^{n+1} \text{ on } \partial\Omega_D, \tag{38}$$

$$\boldsymbol{\sigma}(\boldsymbol{v}_h^{n+1}, q^{n+1})\cdot\boldsymbol{n} = \boldsymbol{g}^{n+1} \quad \text{on } \partial\Omega_N. \tag{39}$$

We write the filter step as

$$\rho\frac{\overline{\boldsymbol{v}}_h^{n+1}}{\Delta t} + \mathscr{L}_F[\boldsymbol{v}_h]\overline{\boldsymbol{v}}_h + \nabla\overline{q}_h^{n+1} = \rho\frac{\boldsymbol{v}_h^{n+1}}{\Delta t} \tag{40}$$

$$\nabla\cdot\overline{\boldsymbol{v}}_h^{n+1} = 0 \tag{41}$$

$$\overline{\boldsymbol{v}}_h^{n+1} = \boldsymbol{u}_D^{n+1} \text{ on } \partial\Omega_D, \tag{42}$$

$$(2\overline{\mu}_h\nabla^s\overline{\boldsymbol{v}}_h^{n+1} - \overline{q}_h^{n+1}\mathbf{I})\boldsymbol{n} = \boldsymbol{0} \quad \text{on } \partial\Omega_N, \tag{43}$$

The relaxation step for the velocity reads

$$\boldsymbol{u}_h^{n+1} = (1-\chi)\boldsymbol{v}_h^{n+1} + \chi\overline{\boldsymbol{v}}_h^{n+1}. \tag{44}$$

Let us multiply (40) by $\chi\alpha$ and add it to (36) to obtain

$$\rho\frac{\alpha}{\Delta t}((1-\chi)\boldsymbol{v}_h^{n+1} + \chi\overline{\boldsymbol{v}}_h^{n+1}) + \mathscr{L}_{NS}[\boldsymbol{u}_h^*]\boldsymbol{v}_h^{n+1} + \chi\alpha\mathscr{L}_F[\boldsymbol{v}_h^{n+1}]\overline{\boldsymbol{v}}_h^{n+1} + \nabla(q_h + \chi\alpha\overline{q}_h) = \boldsymbol{b}_h^{n+1}. \tag{45}$$

Then, using the relaxation for the velocity (44), we obtain that $\boldsymbol{u}_h^{n+1}$ satisfies

$$\rho\frac{\alpha}{\Delta t}\boldsymbol{u}_h^{n+1} + \mathscr{L}_{NS}[\boldsymbol{u}_h^*]\boldsymbol{v}_h^{n+1} + \chi\alpha\mathscr{L}_F[\boldsymbol{v}_h^{n+1}]\overline{\boldsymbol{v}}_h^{n+1} + \nabla(q_h + \chi\alpha\overline{q}_h) = \boldsymbol{b}_h^{n+1}. \tag{46}$$

Using one more time (44), we obtain

$$\begin{aligned}\rho\frac{\alpha}{\Delta t}\boldsymbol{u}_h^{n+1} + \mathscr{L}_{NS}[\boldsymbol{u}_h^*]\boldsymbol{u}_h^{n+1} + \nabla(q_h^{n+1} + \alpha\chi\overline{q}_h^{n+1}) + \\ \chi(\mathscr{L}_{NS}[\boldsymbol{u}_h^*](\boldsymbol{v}_h^{n+1} - \overline{\boldsymbol{v}}_h^{n+1}) + \alpha\mathscr{L}_F[\boldsymbol{v}_h^{n+1}]\overline{\boldsymbol{v}}_h^{n+1}) = \boldsymbol{b}_h^{n+1}.\end{aligned} \tag{47}$$

The continuity equation for $\boldsymbol{u}_h^{n+1}$ is automatically satisfied, since both $\boldsymbol{v}_h^{n+1}$ and $\overline{\boldsymbol{v}}_h^{n+1}$ are both divergence free.

Equation (47) suggests a relaxation for the end-of-step pressure, namely

$$p_h^{n+1} = q_h^{n+1} + \alpha\chi\overline{q}_h^{n+1}. \tag{48}$$

The operator-splitting scheme involves thus three steps:

i) solve equations (36)-(37) to obtain the intermediate velocity and pressure using the standard Navier-Stokes operator $\mathscr{L}_{NS}$ with the physical viscosity μ;

ii) solve equations (40)-(41) to obtain the filtered velocity and pressure using the filter operator $\mathscr{L}_F$ with the artificial viscosity $\overline{\mu}_h$;

iii) finally, we combine the solutions found at steps i) and ii) with (44) and (48) to get the end-of-step velocity and pressure.

In the third step, the velocity and pressure found at the first step are *corrected* by taking into account the energy dissipated at the scales that were not resolved with the given mesh in the Navier-Stokes step. Notice that equation (47) is a consistent perturbation of the original Navier-Stokes

problem, the perturbation vanishing with coefficient χ that in turn vanishes with the discretization parameters (recall that, as suggested in [32], $\chi = O(\Delta t)$). Notice in particular that for $\delta^2/\Delta t \to 0$ we have that $\overline{\mu}_h \to 0$ and $\overline{\boldsymbol{v}}_h^{n+1} \to \boldsymbol{v}_h^{n+1}$. Recalling that $\delta \simeq h$, this means that for a practical choice $\Delta t \simeq h$ the perturbed equation (47) is consistent with the Navier-Stokes equations. On the other hand, when h falls below the Kolomogorv scale η, all the relevant scales are solved and no filter is needed.

The diffusive effects of the EFR sequence may be stated in the following Proposition.

Proposition 4.1
Assume that we have homogeneous Dirichlet boundary conditions. If $\chi \in [0,1]$, at each time step the following inequality holds

$$\|\boldsymbol{u}_h^{n+1}\|_{L^2} \leq \|\boldsymbol{v}_h^{n+1}\|_{L^2}. \tag{49}$$

Proof
Multiply (40) by $\overline{\boldsymbol{v}}_h^{n+1}$ and integrate by parts. Since $\overline{\mu}_h$ is positive and finite, we have that

$$\|\overline{\boldsymbol{v}}_h^{n+1}\|_{L^2}^2 + C\|\nabla\overline{\boldsymbol{v}}_h^{n+1}\|_{L^2}^2 \leq \|\boldsymbol{v}_h^{n+1}\|_{L^2}^2,$$

where C depends on $\overline{\mu}_h$ ρ, α and Δt. Then, we rewrite the relaxation step as

$$\boldsymbol{v}_h^{n+1} = \boldsymbol{u}_h^{n+1} + \chi(\boldsymbol{v}_h^{n+1} - \overline{\boldsymbol{v}}_h^{n+1}).$$

Notice that

$$\begin{aligned}
(\boldsymbol{v}_h^{n+1} - \overline{\boldsymbol{v}}_h^{n+1}, \boldsymbol{u}_h^{n+1}) &= (\boldsymbol{v}_h^{n+1} - \overline{\boldsymbol{v}}_h^{n+1}, (1-\chi)\boldsymbol{v}_h^{n+1} + \chi\overline{\boldsymbol{v}}_h^{n+1}) \\
&= (\boldsymbol{v}_h^{n+1} - \overline{\boldsymbol{v}}_h^{n+1}, (1-\chi)\boldsymbol{v}_h^{n+1} - (1-\chi)\overline{\boldsymbol{v}}_h^{n+1} + \overline{\boldsymbol{v}}_h^{n+1}) \\
&= (1-\chi)\|\boldsymbol{v}_h^{n+1} - \overline{\boldsymbol{v}}_h^{n+1}\|_{L^2}^2 + (\boldsymbol{v}_h^{n+1} - \overline{\boldsymbol{v}}_h^{n+1}, \overline{\boldsymbol{v}}_h^{n+1}) \\
&= (1-\chi)\|\boldsymbol{v}_h^{n+1} - \overline{\boldsymbol{v}}_h^{n+1}\|_{L^2}^2 + C\|\nabla\overline{\boldsymbol{v}}_h^{n+1}\|_{L^2}^2
\end{aligned}$$

For $\chi \in [0,1]$ (see Remark 6.1), the last line is for sure positive. We have

$$\begin{aligned}
\|\boldsymbol{v}_h^{n+1}\|_{L^2}^2 = (\boldsymbol{v}_h^{n+1}, \boldsymbol{v}_h^{n+1}) &= (\boldsymbol{u}_h^{n+1} + \chi(\boldsymbol{v}_h^{n+1} - \overline{\boldsymbol{v}}_h^{n+1}), \boldsymbol{u}_h^{n+1} + \chi(\boldsymbol{v}_h^{n+1} - \overline{\boldsymbol{v}}_h^{n+1})) \\
&= \|\boldsymbol{u}_h^{n+1}\|_{L^2}^2 + 2\chi(\boldsymbol{v}_h^{n+1} - \overline{\boldsymbol{v}}_h^{n+1}, \boldsymbol{u}_h^{n+1}) + \chi^2\|\boldsymbol{v}_h^{n+1} - \overline{\boldsymbol{v}}_h^{n+1}\|_{L^2}^2 \\
&\geq \|\boldsymbol{u}_h^{n+1}\|_{L^2}^2.
\end{aligned}$$

□

This proposition shows that the filter actually damps the intermediate velocity in the L^2 norm. We speculate that a similar relation formally holds for $\Delta^{N+1}\boldsymbol{u}_h^{n+1}$ and $\Delta^{N+1}\boldsymbol{v}_h^{n+1}$. This will be investigated in the follow-up of the present work.

We rewrite (47) as

$$\begin{aligned}
&\rho\frac{\alpha}{\Delta t}\boldsymbol{u}_h^{n+1} + \mathscr{L}_{NS}[\boldsymbol{u}_h^*]\boldsymbol{u}_h^{n+1} + \nabla\left(q_h^{n+1} + \alpha\chi\overline{q}_h^{n+1}\right) + \\
&\quad \chi\left(\mathscr{L}_{NS}[\boldsymbol{u}_h^*](\boldsymbol{v}_h^{n+1} - \overline{\boldsymbol{v}}_h^{n+1}) + \alpha\mathscr{L}_F[\boldsymbol{u}_h^{n+1}]\overline{\boldsymbol{v}}_h^{n+1} + \alpha\left(\mathscr{L}_F[\boldsymbol{v}_h^{n+1}] - \mathscr{L}_F[\boldsymbol{u}_h^{n+1}]\right)\overline{\boldsymbol{v}}_h^{n+1}\right) = \boldsymbol{b}_h^{n+1}.
\end{aligned} \tag{50}$$

Should our speculation be true, the last term at the left hand side is positive. If we now consider the operator $\mathscr{L}$ defined by

$$\mathscr{L}[\boldsymbol{u}_h^*, \boldsymbol{u}_h^{n+1}]\boldsymbol{u}_h^{n+1} = \mathscr{L}_{NS}[\boldsymbol{u}_h^*]\boldsymbol{u}_h^{n+1} + \alpha\chi\mathscr{L}_F[\boldsymbol{u}_h^{n+1}]\boldsymbol{u}_h^{n+1}. \tag{51}$$

and the perturbed discrete Navier-Stokes momentum equation

$$\rho\frac{\alpha}{\Delta t}\boldsymbol{u}_h^{n+1} + \mathscr{L}[\boldsymbol{u}_h^*, \boldsymbol{u}_h^{n+1}]\boldsymbol{u}_h^{n+1} + \nabla p_h^{n+1} = \boldsymbol{b}_h^{n+1}, \tag{52}$$

our arguments point out that the EFR algorithm is an inexact operator splitting of (52). On the other hand, (52) is the original Navier-Stokes momentum equation plus a viscous term surrogating at the mesh scale the effect of energy dissipation at the unresolved scales.

Remark 4.1
The choice of the end-of-step pressure (48) is somehow arbitrary. It is driven by the strategy of gathering all the effects of the filter to the velocity. Other choices are possible. For instance, one may argue that velocity and pressure should be relaxed in the same way, leading to

$$p_h^{n+1} = (1-\chi)q_h^{n+1} + \chi\overline{q}_h^{n+1}, \tag{53}$$

This approach - that we call relaxation-consistent - introduces a different perturbed equation for the EFR algorithm, i.e.

$$\begin{aligned}
&\rho\frac{\alpha}{\Delta t}\boldsymbol{u}_h^{n+1} + \mathscr{L}_{NS}[\boldsymbol{u}_h^*]\boldsymbol{u}_h^{n+1} + \nabla p_h^{n+1} \\
&+ \chi(\mathscr{L}_{NS}[\boldsymbol{u}_h^*](\overline{\boldsymbol{v}}_h^{n+1} - \boldsymbol{v}_h^{n+1}) + \alpha(1-\chi)\mathscr{L}_F[\boldsymbol{v}_h^{n+1}](\overline{\boldsymbol{v}}_h^{n+1} - \boldsymbol{v}_h^{n+1})) \\
&+ \chi\nabla(q_h^{n+1} + (\alpha-1)\overline{q}_h^{n+1}) = \boldsymbol{b}_h^{n+1}.
\end{aligned} \tag{54}$$

Consistency is still preserved with the same factor $\mathcal{O}(\chi)$. The accuracy and stability of this scheme need to be further investigated (see Sect. 5.1). Thus, the results in the Sect. 6 refer to (48).

4.1. Boundary conditions for the end-of-step solution

To conclude our description of the EFR algorithm, we address the boundary conditions satisfied by the end-of-step solution, so to justify the choice of boundary condition (43).

By combining (44), (38), and (42), we have

$$\boldsymbol{u}_h^{n+1} = \boldsymbol{u}_D^{n+1}, \quad \text{on } \partial\Omega_D.$$

As for the Neumann part of the boundary $\partial\Omega_N$, from (44) and (48), we have

$$\begin{aligned}
(2\mu\nabla^s\boldsymbol{u}_h^{n+1} - p_h^{n+1}\mathbf{I})\boldsymbol{n} &= (2\mu\nabla^s((1-\chi)\boldsymbol{v}_h^{n+1} + \chi\overline{\boldsymbol{v}}_h^{n+1}) - (q_h^{n+1} + \alpha\chi\overline{q}_h^{n+1})\mathbf{I})\boldsymbol{n} \\
&= (2\mu\nabla^s\boldsymbol{v}_h^{n+1} - q_h^{n+1}\mathbf{I})\boldsymbol{n} + \chi(2\mu\nabla^s(\overline{\boldsymbol{v}}_h^{n+1} - \boldsymbol{v}_h^{n+1}) - \alpha\overline{q}_h^{n+1}\mathbf{I})\boldsymbol{n}.
\end{aligned}$$

Then, using (39) and (43), we find that

$$(2\mu\nabla^s\boldsymbol{u}_h^{n+1} - p_h^{n+1}\mathbf{I})\boldsymbol{n} = \boldsymbol{g}^{n+1} - \chi(2\mu\nabla^s(\boldsymbol{v}_h^{n+1} - \overline{\boldsymbol{v}}_h^{n+1}) + 2\alpha\overline{\mu}_h\,\nabla^s\,\overline{\boldsymbol{v}}_h^{n+1})\boldsymbol{n}. \tag{55}$$

This shows that with the choice (43) for the filter, the end-of-step velocity and pressure fulfill a perturbation of the original Neumann conditions, yet consistent with the coefficient χ.

Remark 4.2
In the case of the more general Robin boundary condition of the form

$$(2\mu\nabla^s\boldsymbol{u}_h^{n+1} - p_h^{n+1}\mathbf{I})\boldsymbol{n} + \gamma\boldsymbol{u}_h^{n+1} = \boldsymbol{g}^{n+1} \text{ on } \partial\Omega_R,$$

(where in general $\gamma > 0$), the same consistency as for (55) can still be achieved by imposing the following boundary condition for the filter

$$(2\overline{\mu}_h\,\nabla^s\,\overline{\boldsymbol{v}}_h^{n+1} - \overline{q}_h^{n+1}\mathbf{I})\boldsymbol{n} + \frac{\gamma}{\alpha}\overline{\boldsymbol{v}}_h^{n+1} = \frac{\gamma}{\alpha}\boldsymbol{v}_h^{n+1} \text{ on } \partial\Omega_R.$$

These boundary conditions can arise in particular applications, such as, for instance, domain decomposition techniques in geometric multiscale approaches (see, e.g., [19], Chapter 11), or with the use of the so called 'transpiration' techniques for Fluid-Structure Interaction problems (see, e.g., [14]).

4.2. Quantification of the relaxation parameter χ

As we noticed, for $\delta \simeq h$, $\delta^2/\Delta t \to 0$, $\chi \simeq \Delta t$, the end-of-step pair $(\boldsymbol{u}_h^{n+1}, p_h^{n+1})$ fulfills a consistent perturbation of the Navier-Stokes momentum equation (the mass conservation being fulfilled

exactly). However, we need a practical way to tune the proportionality constant c in the rule $\chi = c\Delta t$. We experienced that the choice $c \approx 1$ does not provide enough numerical dissipation in realistic applications (see Section 6). On the other hand, we postulate that $\chi > 0$ only for $h > \eta$, since for $h \to 0$ we do not need any filtering.

In order to find a proper formula for χ we use a heuristic argument. We set χ such that the viscous stress in (52) on an under-resolved mesh of size h provides the same amount of dissipation as the viscous term in (36) on a fully resolved mesh of size η.

Let us introduce an *equivalent* stress tensor in the perturbed Navier-Stokes equations (52)

$$\boldsymbol{\sigma}^{n+1} = -p_h^{n+1}\mathbf{I} + 2(\mu + \alpha\chi\overline{\mu}_h)\nabla^s \boldsymbol{u}_h^{n+1}.$$

In addition, let ∇^s_ξ denote the symmetric gradient on a mesh with size ξ, where ξ is either h or η. We require that the viscous contribution of this equivalent tensor matches the viscous contribution of the standard stress tensor of a Newtonian fluid (5) on a mesh of size η,

$$(\mu + \alpha\chi\overline{\mu}_h)\nabla^s_h \boldsymbol{u}_h^{n+1} \simeq \mu\nabla^s_\eta \boldsymbol{u}_h^{n+1}, \tag{56}$$

With the approximation ∇^s_ξ with ξ^{-1}, we obtain

$$(\mu + \alpha\chi\overline{\mu}_h)\frac{1}{h} \simeq \mu\frac{1}{\eta},$$

leading to

$$\chi \simeq \frac{\mu}{\alpha\overline{\mu}_h}\left(\frac{h}{\eta} - 1\right) \simeq \frac{\mu}{\alpha\rho}\frac{1}{||a||_\infty \delta^2}\left(\frac{h}{\eta} - 1\right)\Delta t. \tag{57}$$

Here, $||a||_\infty$ is the infinity norm of the indicator function. As for Remark 3.1, in practice $\|a\|_\infty \leq 1$ so that the constant $c = \frac{\mu}{\alpha\rho}\frac{1}{||a||_\infty\delta^2}\left(\frac{h}{\eta} - 1\right)$ is promptly estimated *a priori*.

In practice we may set

$$c = \frac{\mu}{\alpha\rho\eta\delta^2}\max(h - \eta, 0).$$

Remark 4.3
The choice of the filtering radius δ is still an open problem when dealing with non-uniform grids. As we mentioned before, it is a common choice to set δ equal to the space discretization parameter h, which usually refers to the largest diameter of the elements of the mesh. However, when using non-uniform grids, this could lead to eccessive numerical diffusion, since the region where the filter has a significant effect would not be guaranteed to be confined within a single element and may include also one (or more) neighboring elements for the smallest elements. Therefore, in this paper we set h to be the length h_{min} of the shortest edge in the mesh, which guarantees that for each degree of freedom (d.o.f.) the region where the filter has a significant effect is confined within the patch of elements sharing the d.o.f.

5. DISCRETIZATION OF THE OPERATOR SPLITTING ALGORITHM

Let $\mathcal{T}_h$ be a conformal and quasi-uniform partition of Ω. For the approximation of velocity and pressure, we use *inf-sup* stable FE Taylor-Hood pair $\mathbb{P}_2$-$\mathbb{P}_1$ [42, 16]. We do not use any specific numerical stabilization for the convective term.

For the time discretization we use BDF2 (23), with the corresponding convective term extrapolation

$$\boldsymbol{u}_h^* = 2\boldsymbol{u}_h^n - \boldsymbol{u}_h^{n-1}.$$

We denote by M the mass matrix, K the diffusion matrix, N the matrix associated with the discretization of the convective term, and B the matrix associated with the discretization of the

operator $(-\nabla\cdot)$. The full discretization of problem (36)-(39) with BDF2 yields the following system

$$\rho\frac{3}{2\Delta t}M\mathbf{v}^{n+1}+\rho N\mathbf{v}^{n+1}+\mu K\mathbf{v}^{n+1}+B^T\mathbf{q}^{n+1}=\mathbf{b}_u^{n+1}, \tag{58}$$

$$B\mathbf{v}^{n+1}=\mathbf{0}, \tag{59}$$

where $\mathbf{v}^{n+1}$ and $\mathbf{q}^{n+1}$ are the arrays of nodal values for the intermediate velocity and pressure.

The array $\mathbf{b}_u^{n+1}$ accounts for the contributions of solution at the previous time steps and the contribution coming from nonhomogeneous boundary conditions, while the constant $3/2$ in the first term comes from (23).

Remark 5.1
In case of nonhomogeneous Dirichlet boundary conditions, depending on the implementation details, the right hand side of (59) may also be nonzero. This can be the case, for instance, when a "lift function" is used and the degrees of freedom on the Dirichlet boundary are eliminated (via Gaussian reduction). In our implementation, however, the Dirichlet boundary conditions are imposed by setting the corresponding rows in the momentum equation to be $u_i=\boldsymbol{u}_D(\mathbf{x}_i)$. For this reason, the right hand side of (59) is still 0.

Setting $C=\rho\frac{3}{2\Delta t}M+\rho N+\mu K$, we can rewrite (58)-(59) in the form

$$A\mathbf{x}^{n+1}=\mathbf{b}^{n+1}, \tag{60}$$

where

$$A=\begin{bmatrix} C & B^T\\ B & 0\end{bmatrix},\quad \mathbf{x}^{n+1}=\begin{bmatrix}\mathbf{v}^{n+1}\\ \mathbf{q}^{n+1}\end{bmatrix},\quad \mathbf{b}^{n+1}=\begin{bmatrix}\mathbf{b}_u^{n+1}\\ \mathbf{0}\end{bmatrix}. \tag{61}$$

Let $\overline{K}$ be the matrix associated with the discretization of the diffusive term in (40). The full discretization of problem (40)-(43) then yields

$$\frac{\rho}{\Delta t}M\overline{\mathbf{v}}^{n+1}+\overline{K}\overline{\mathbf{v}}^{n+1}+B^T\overline{\mathbf{q}}^{n+1}=\frac{\rho}{\Delta t}M\mathbf{v}^{n+1}, \tag{62}$$

$$B\overline{\mathbf{v}}^{n+1}=\mathbf{0}, \tag{63}$$

where $\overline{\mathbf{v}}^{n+1}$ and $\overline{\mathbf{q}}^{n+1}$ are the nodal values of the filter step velocity and pressure. Setting $\overline{C}=\frac{\rho}{\Delta t}M+\overline{K}$, we can rewrite (62)-(63) in the form

$$\overline{A}\overline{\mathbf{x}}^{n+1}=\overline{\mathbf{b}}^{n+1}, \tag{64}$$

where

$$\overline{A}=\begin{bmatrix}\overline{C} & B^T\\ B & 0\end{bmatrix},\quad \overline{\mathbf{x}}^{n+1}=\begin{bmatrix}\overline{\mathbf{v}}^{n+1}\\ \overline{\mathbf{q}}^{n+1}\end{bmatrix},\quad \overline{\mathbf{b}}^{n+1}=\begin{bmatrix}\frac{\rho}{\Delta t}M\mathbf{v}^{n+1}\\ \mathbf{0}\end{bmatrix}. \tag{65}$$

At every time level t^{n+1}, to solve systems (60) and (64) we use the left preconditioned GMRES method. To precondition both systems, we use an upper-triangular variant of the pressure corrected Yosida splitting [22, 44]. For the matrix A, this preconditioner reads:

$$P_A=\begin{bmatrix} C & B^T\\ 0 & S(S+BH(\mu K+\rho N)HB^T)^{-1}S\end{bmatrix},\quad H=\frac{2\Delta t}{3\rho}M^{-1},\quad S=-BHB^T. \tag{66}$$

The above preconditioner is a suitable approximation of the U factor in the exact block LU factorization of matrix A in (61):

$$A=LU,\quad L=\begin{bmatrix} I & 0\\ BC^{-1} & I\end{bmatrix},\quad U=\begin{bmatrix} C & B^T\\ 0 & -BC^{-1}B^T\end{bmatrix}. \tag{67}$$

See [40, 41, 24] for more details. For the matrix $\overline{A}$, the preconditioner has a similar structure, namely:

$$P_{\overline{A}} = \begin{bmatrix} \overline{C} & B^T \\ 0 & \overline{S}(\overline{S} + B\overline{H}(\overline{K})\overline{H}B^T)^{-1}\overline{S} \end{bmatrix}, \quad \overline{H} = \frac{\Delta t}{\rho} M^{-1}, \quad \overline{S} = -B\overline{H}B^T. \tag{68}$$

The application of the preconditioner requires to solve two linear systems in both C and S for P_A, and two linear systems in both $\overline{C}$ and $\overline{S}$ for $P_{\overline{A}}$. To solve each of these systems, we use a Krylov method with a general purpose preconditioner, such as incomplete LU or algebraic multilevel. It is worth mentioning that, while C is in general non-symmetric because of the convective term, the matrix $\overline{C}$ is symmetric (and positive definite). Therefore, while for P_A the (1,1) block is solved with GMRES method, (1,1) block of $P_{\overline{A}}$ can be solved with the Conjugate Gradient method.

5.1. Matrix formulation of the relaxation consistent EFR scheme

As done for the matrix A we can perform the LU factorization of the matrix $\overline{A}$,

$$\overline{A} = \overline{L}\overline{U}, \quad \overline{L} = \begin{bmatrix} I & 0 \\ B\overline{C}^{-1} & I \end{bmatrix}, \quad \overline{U} = \begin{bmatrix} \overline{C} & B^T \\ 0 & -B\overline{C}^{-1}B^T \end{bmatrix}. \tag{69}$$

With this notation, the relaxation consistent EFR method mentioned in Remark 4.1 can be promptly rewritten in a compact form. According to this scheme we set at each time step

$$\begin{bmatrix} \mathbf{u}^{n+1} \\ \mathbf{p}^{n+1} \end{bmatrix} = (1-\chi) \begin{bmatrix} \mathbf{v}^{n+1} \\ \mathbf{q}^{n+1} \end{bmatrix} + \chi \begin{bmatrix} \overline{\mathbf{v}}^{n+1} \\ \overline{\mathbf{q}}^{n+1} \end{bmatrix},$$

where $\left[\mathbf{u}^{n+1}\,\mathbf{p}^{n+1}\right]^T$ is the end-of-step solution. By getting rid of the filter quantities $\left[\overline{\mathbf{v}}^{n+1}\,\overline{\mathbf{q}}^{n+1}\right]^T$, we obtain

$$\begin{bmatrix} \mathbf{u}^{n+1} \\ \mathbf{p}^{n+1} \end{bmatrix} = \begin{bmatrix} \mathbf{v}^{n+1} \\ \mathbf{q}^{n+1} \end{bmatrix} - \chi \left(I - (\overline{L}\overline{U})^{-1} \begin{bmatrix} \frac{\rho}{\Delta t} M & 0 \\ B & 0 \end{bmatrix} \right) \begin{bmatrix} \mathbf{v}^{n+1} \\ \mathbf{q}^{n+1} \end{bmatrix}.$$

By straightforward (tedious) computations, we can rewrite the matrix in parenthesis (multiplied by χ) as

$$\begin{bmatrix} \left(I - \overline{C}^{-1}B^T\overline{\Sigma}^{-1}B\right)\left(I - (I+Z)^{-1}\right) & 0 \\ \overline{\Sigma}^{-1}B\left(I - (I+Z)^{-1}\right) & I \end{bmatrix}$$

where $\overline{\Sigma} = B\overline{C}^{-1}B^T$ and $Z = \frac{\Delta t}{\rho} M^{-1}\overline{K}$.

This representation of the solution at each time step outlines how the end-of-step solution is the result of a modification of the unfiltered one $\left[\mathbf{v}^{n+1}\,\mathbf{q}^{n+1}\right]^T$ weighted by the relaxation parameter χ. Notice in addition that

$$\left(I - (I+Z)^{-1}\right) = Z(I+Z)^{-1} = \frac{\Delta t}{\rho} M^{-1}\overline{K}(I+Z)^{-1} \simeq \mathcal{O}(\Delta t).$$

This means that the perturbation induced by this scheme on the velocity scales with $\chi\Delta t$ (as opposed to χ "only"). Specific strategies to manage the correct amount of filtering for this relaxation consistent scheme will be investigated in a forthcoming work.

6. NUMERICAL RESULTS

In order to demonstrate the effectiveness of the approach described in the previous sections, we have selected a benchmark from the U.S. Food and Drug Administration (FDA). This benchmark consists in simulating the flow of an incompressible and Newtonian fluid with prescribed density

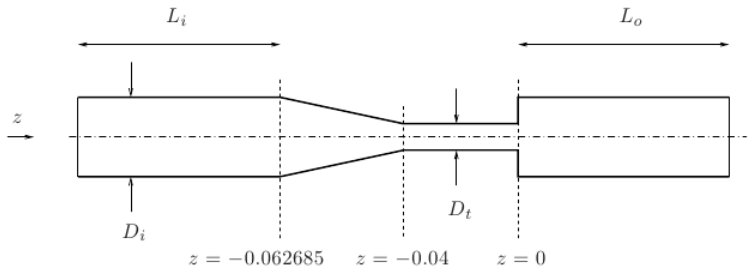


Figure 1. A section of the computational domain, with $D_i = 0.012$, $D_t = 0.004$, $L_i = 4D_i$ and $L_o = 12D_i$. The units are meter.

Re_t	Re_i	flow rate Q (m^3/s)	η (m)
3500	1167	3.6444e-5	2.6371e-5
5000	1667	5.2062e-5	2.0182e-5

Table I. Throat Reynolds number Re_t, inlet Reynolds number Re_i, flow rate Q, and Kolmogorov length scale η for the flow regimes under consideration.

and viscosity ($\rho = 1056$ kg/m^3 and $\mu = 0.0035$ Pa·s) in an idealized medical device shaped like a nozzle (see Fig. 1) at different Reynolds numbers. In the geometry pictured in Fig. 1, the fluid flows from left to right, passing through a cylindrical entrance region, a conical convergent, a cylindrical throat, and a sudden expansion into a larger cylinder.

The complete FDA benchmark requires to study this system for a variety of conditions, including laminar, transitional, and turbulent regimes: the results of the published inter-laboratory experiments refer to values of the Reynolds numbers in the throat (defined as in (10)) of $Re_t = 500, 2000, 3500, 5000, 6500$. In a previous work [37], we have successfully validated LifeV [3] against this benchmark for Re_t up to 3500 using DNS.

To test our methodology, we focus on Reynolds numbers $Re_t = 3500, 5000$. The case $Re_t = 6,500$ is carefully analyzed in another paper [6].

In Table I, we report the throat Reynolds number Re_t, the corresponding inlet Reynolds number Re_i, flow rate Q, and the Kolmogorov length scale η for the considered flow regimes. The value of η was found by plugging into (12) the value of Re_t and the diameter of the expansion channel D_i as characteristic length. Notice that for both flow regimes the flow in the entrance region is laminar, Re_i being below the critical Reynolds number for transitional flow in a straight pipe ($Re \simeq 2000$ [43]).

On the lateral surface of the computational domain we prescribe a no-slip boundary condition. For the two flow regimes in Table I, at the inlet section we prescribe a Poiseuille velocity profile to get the desired flow rate, a choice which is justified by the considered values of Re_i. The length of the inlet chamber L_i was set to four times its diameter. At the outlet section, we prescribe a stress-free (natural) boundary condition. The actual experimental set up of the FDA benchmark is a closed loop [27] and it is not reflected by our condition. However, homogeneous Neumann conditions are expected to introduce a minimal error localized only in the close neighborhood of the outlet section [28]. The results of the flow analysis are not affected, provided that the expansion channel is long enough. For all the simulations, the length of the expansion channel (L_o in Fig. 1) was set to 12 times its diameter and we checked that in all the cases the velocity components reach a plateau before the outlet.

As for the initial condition, we start our simulations with fluid at rest, i.e., $\boldsymbol{u} = \mathbf{0}$ everywhere in Ω. We use a smooth transition of the velocity profile at the inlet from rest to the regime flow conditions.

For both flow regimes in Table I, we considered several meshes with different levels of refinement. The selection of the time step was driven by accuracy considerations solely. In fact, even though the semi-implicit treatment of the convective term in eq. (36) does not guarantee the unconditional stability in time, we encounter no stability issues in the numerical experiments.

mesh name	h_{min}	h_{avg}	h_{max}	# nodes	# tetrahedra	Δt
$1900k$	1.06e-4	5.15e-4	1.49e-3	3.7e5	1.9e6	1e-4
$1200k$	1.08e-4	5.46e-4	1.63e-3	2.3e5	1.2e6	1e-4
$900k$	1.09e-4	5.16e-4	1.87e-3	1.8e5	9e5	1e-4
$330k$	2.23e-4	9.48e-4	1.93e-3	6.5e4	3.3e5	2e-4
$140k$	3.39e-4	1.11e-3	3.09e-3	3.1e4	1.4e5	3e-4

Table II. Case $Re_t = 3500$: meshes used for the simulations, with their minimum diameter h_{min}, average diameter h_{avg}, maximum diameter h_{max}, and number of nodes and tetrahedra. The units for the diameters are meter. We also report the time step Δt (in s) used for the simulations with each mesh.

We compare the experimental data provided by the FDA with our numerical simulations for the flow regimes listed in Table I. The experimental data were acquired by three independent laboratories and one of the laboratory ran three trials, so that for each case we have five sets of data. The comparison is made in terms of normalized axial component of the velocity and normalized pressure difference along the centerline. The axial component of the velocity u_z is normalized with respect to the average axial velocity at the inlet $\bar{u}_i$

$$u_z^{norm} = \frac{u_z}{\bar{u}_i}, \quad \text{with} \quad \bar{u}_i = \frac{Q}{\pi D_i^2/4}, \tag{70}$$

where Q is the volumetric flow rate calculated from the throat Reynolds number (see Table I). The pressure difference data are normalized with respect to the dynamic pressure in the throat

$$\Delta p^{norm} = \frac{p_z - p_{z=0}}{1/2\rho\bar{u}_t^2}, \quad \text{with} \quad \bar{u}_t = \frac{Q}{\pi D_t^2/4}, \tag{71}$$

where p_z denotes the wall pressure along the z axis and $p_{z=0}$ is the wall pressure at $z = 0$. To compute the value of Δp^{norm}, we probed the pressure value at the corresponding location on the axis of the domain, since we observed pressure values being approximately uniform on axial cross-sections.

The graphs with the above comparisons are reported in Sec. 6.1 and 6.2 for $Re_t = 3500$ and $Re_t = 5000$, respectively.

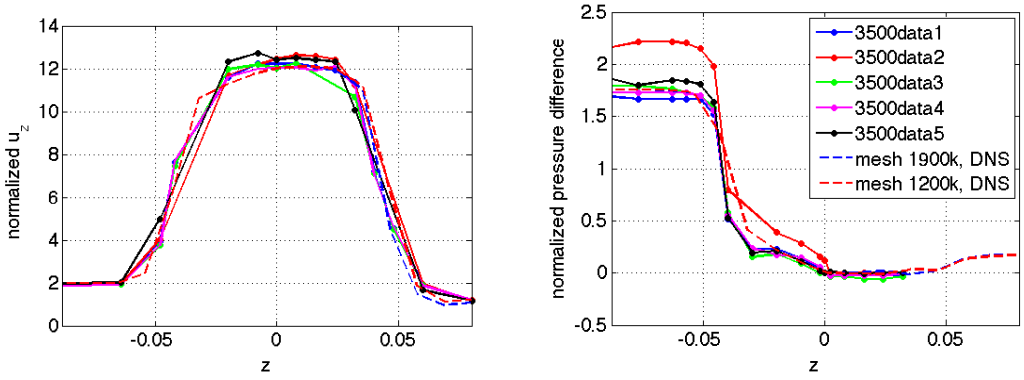
6.1. Case $Re_t = 3500$

Among the Reynolds numbers considered by the FDA benchmark, $Re_t = 3500$ is the lowest above the critical Reynolds number for transitional flow in a straight pipe. As mentioned earlier, we have dealt with this case in [37], where we showed that DNS with a properly refined mesh is able to capture with precision the jet breakdown observed in the experiments. Here, we start from the mesh used in [37] for the DNS at $Re_t = 3500$ and make it progressively coarser to understand the performances of the EFR algorithm described in Sec. 4.

The meshes we considered and the associated time step used in the simulations are reported in Table II. The name of each mesh reflects the number of elements. After several numerical experiments, in [37] we managed to identify a time step value and a mesh sufficiently refined in the different regions of the domain such that the results obtained with DNS were in excellent agreement with the experimental data. Mesh $1900k$ features that level of refinement and is associated with the same Δt used for the simulations in [37]. The time step associated with all the other meshes in Table II was chosen such that the ratio $h_{min}/\Delta t$ is kept approximatively constant.

Starting from fluid at rest, the turbulent regime is fully developed already at $t \simeq 0.3$ s. We let the simulations run till past that time and then take around 10 solutions whose average is compared to the experimental data. In fact, since the measurements of a turbulent flow are averaged over time [27], we average also the numerical results for a fair comparison. We noticed that averaging over more than 10 solutions does not change the average value.

A DNS is possible only with meshes $1900k$ and $1200k$. In fact, a DNS with mesh $900k$ does not reach regime conditions for the instabilities in the computed velocity due to mesh under-resolution



(a) normalized axial velocity along z

(b) normalized pressure difference along z

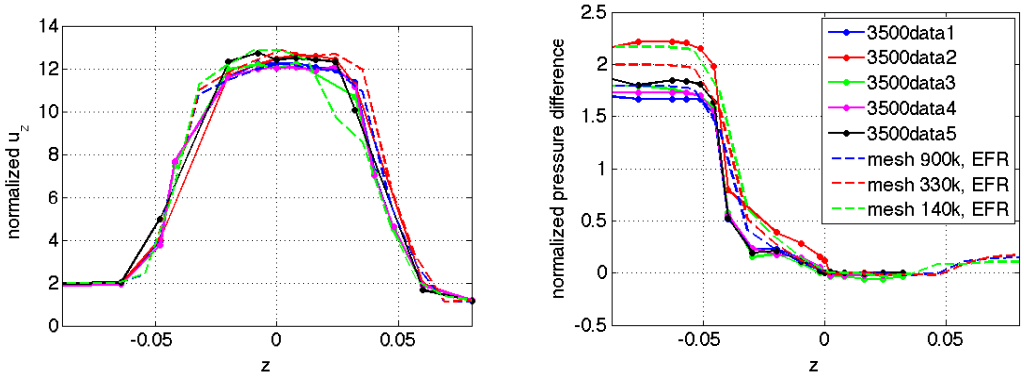
Figure 2. Case $Re_t = 3500$, DNS with two different meshes: comparison between experimental data (solid lines) and numerical results (dashed lines) for (a) normalized axial velocity (70) along the z axis and (b) normalized pressure difference (71) along the z axis. The legend in (b) is common to both subfigures.

causing the simulation to crash. We report the comparison for the normalized axial velocity (70) along the z axis (Fig. 2(a)) and the normalized pressure drop (71) along the z axis (Fig. 2(b)). In Fig. 2, we plotted a dot for every measurement and a solid line to linearly interpolate the five sets of measurements, while we used a dashed line for the numerical results obtained with meshes $1900k$ and $1200k$. We notice that the axial velocities computed with both meshes match the measurements all along the portion of the z axis under consideration ($-0.088 < z < 0.08$), capturing accurately the jet breakdown point observed in experiments. As shown in Fig. 2(b), also the simulated pressure drop on both meshes is in very good agreement with the experimental data, except in the convergent where the simulated pressure difference overestimates almost all the measurements. The reason for this overestimation is explained in [37]. Fig. 2 shows that even mesh $1200k$ has a sufficient level of refinement to obtain numerical results in excellent agreement with the measurements in terms of average quantities, despite its h_{avg} is roughly 20 times larger than the Kolmogorov length scale at $Re_t = 3500$ (see Tables I and II).

We test the EFR algorithm described in Sections 4 and 5 with deconvolution of order $N = 0$ on all the meshes in Table II coarser than mesh $1200k$. We report the comparison between computed and measured normalized axial velocity (70) and pressure difference (71) in Fig. 3(a) and 3(b), respectively. Remarkably, the EFR algorithm succeeds in curing the convective term instabilities even on mesh $140k$, which has 88% less elements than mesh $1200k$ (the coarsest mesh that allowed for a DNS) and an average diameter more than 42 times larger than η at $Re_t = 3500$. From Fig. 3(a), we see that the axial velocity computed on meshes $900k$ and $330k$ are in agreement with the measurements all along the z axis. With mesh $140k$, the jet starts to break down closer to the sudden expansion than the jets obtained with the other two meshes. Nonetheless, the total jet length computed with mesh $140k$ agrees very well with the measured total jet length. As for the pressure drop, Fig. 3(b) shows that the computations on the three meshes fall within the measurements. In particular, the pressure drop simulated with mesh $900k$ is in very good agreement with the experimental data, again with the exception of the convergent region.

In [45], none of the presented CFD results was able to reproduce the correct jet breakdown point, because DNS predicted a longer jet (likely due to an incorrect simulation setup) while simulations with turbulence models under-predicted the jet length. Thus, we may state that all the computed axial velocities in Fig. 3(a) are reasonable, regardless of the mesh.

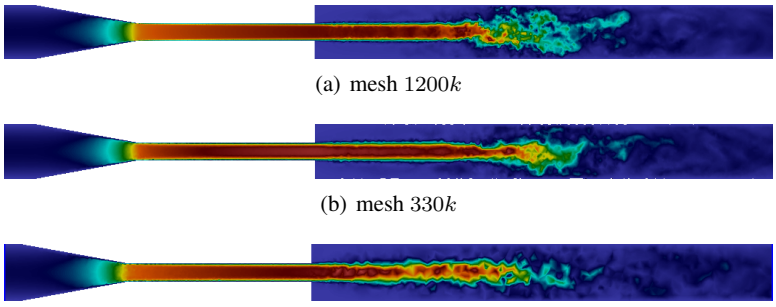
For a qualitative comparison, we report in Fig. 4 the velocity magnitude computed with meshes $1200k$, $330k$, and $140k$ on a section of the domain after the turbulent regime is fully established. Again, the results with mesh $1200k$ (in Fig. 4(a)) have been obtained with DNS and therefore show



(a) normalized axial velocity along z

(b) normalized pressure difference along z

Figure 3. Case $Re_t = 3500$, EFR with three different meshes, $N = 0$: comparison between experimental data (solid lines) and numerical results (dashed lines) for (a) normalized axial velocity (70) along the z axis and (b) normalized pressure difference (71) along the z axis. The legend in (b) is common to both subfigures.



(a) mesh $1200k$

(b) mesh $330k$

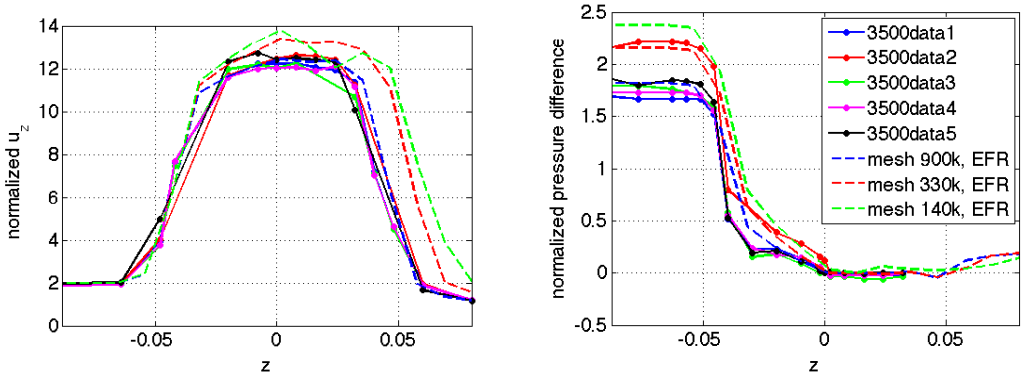
(c) mesh $140k$

Figure 4. Case $Re_t = 3500$: velocity magnitude computed with meshes (a) $1200k$, (b) $330k$, and (c) $140k$ on a section of the domain after the turbulent regime is fully established. The results with mesh $1200k$ have been obtained with DNS, while the results with meshes $330k$ and $140k$ have been obtained with the EFR algorithm and $N = 0$.

a high level of detail. With meshes $330k$ and $140k$, the finer details of the smaller turbulent structures are lost, yet thanks to the EFR algorithm the average behavior of the flow is well captured (see Fig. 4(b) and 4(c)) at a fraction of the computational cost. In fact, a time step of the DNS with mesh $1200k$ takes around 240 s on 80 CPUs, while a time step of the EFR algorithm with mesh $330k$ takes around 80 s (50 s for the evolve step plus 30 s for the filter step) on 48 CPUs and with mesh $140k$ around 65 s (50 s for the evolve step plus 15 s for the filter step) on 24 CPUs. These computational times refer to simulations run on Maxwell, a cluster of the Research Computing Center at the University of Houston.

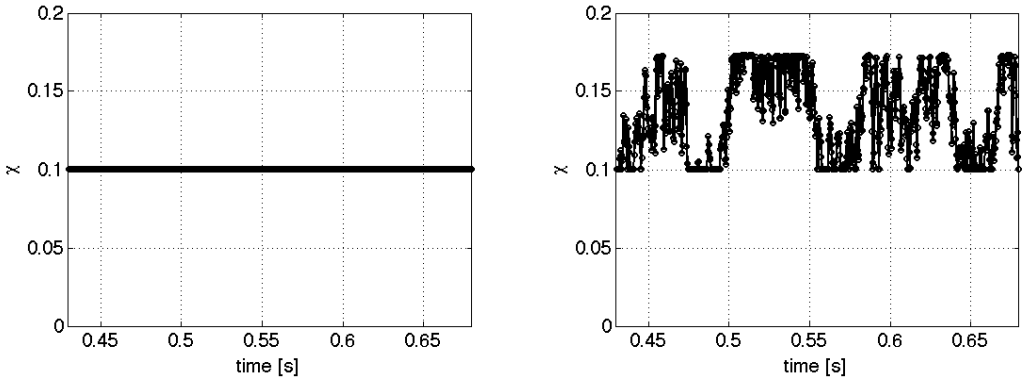
Next, we set the deconvolution order N to 1 and repeat the simulations on meshes $900k$, $330k$, and $140k$. We report the comparison for the normalized axial velocity (70) along the z axis in Fig. 5(a) and the normalized pressure difference (71) along the z axis in Fig. 5(b). From Fig. 5(a), we see that, while the axial velocity computed on mesh $900k$ is still in excellent agreement with the experimental data, the jet obtained with the coarser meshes is too long.

At $Re_t = 3500$ the choice $N = 1$ is more diffusive than the corresponding case $N = 0$ with the same relaxation parameter and on the same mesh. We speculate that this is related to (32) and to Property 2 listed at the end of Sect. 3.3.1. In fact, if we assume that the velocity is regular enough so that $a_{D_N}(\cdot)$ is bounded independently of δ (as stated in the Property mentioned above), the amount



(a) normalized axial velocity along z (b) normalized pressure difference along z

Figure 5. Case $Re_t = 3500$, EFR with three different meshes, $N = 1$: comparison between experimental data (solid lines) and numerical results (dashed lines) for (a) normalized axial velocity (70) along the z axis and (b) normalized pressure difference (71) along the z axis. The legend in (b) is common to both subfigures.

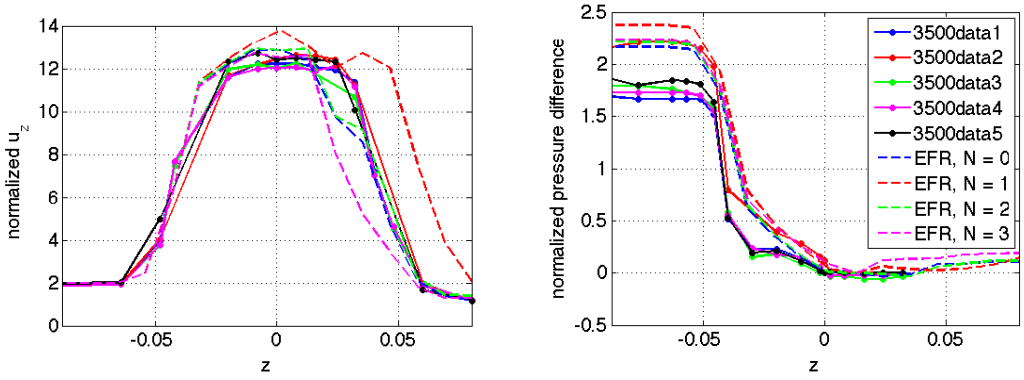


(a) $N = 0$ (b) $N = 1$

Figure 6. Case $Re_t = 3500$, EFR with mesh $140k$: value of χ over time interval $[0.43, 0.68]$ s for (a) $N = 0$ and (b) $N = 1$.

of viscosity introduced by our EFR scheme is proportional to δ^2 for $N = 0$ and to δ^4 for $N = 1$. For $h \approx \delta$ this implies that coarsening the mesh of a factor of 2 increases the extra-diffusion of a factor of 4 for $N = 0$ and of a factor of 16 for $N = 1$. More in general, the behavior of the artificial viscosity (33) as N increases is the result of a complex interplay of the effect of the Yosida regularization (see Sec. 3.3.1) and the fact that as $N \to \infty$ operator D_N approaches F^{-1} (an unbounded operator, see Sec. 3.3). This will be investigated in future works. In Fig. 5(b), we see that the pressure drop is progressively more overestimated in the entrance region as the mesh gets coarser (up to 25% overestimation on mesh $140k$ with respect to the average measured pressure difference).

This is reflected by the the value of χ over the time interval $[0.43, 0.68]$ s for the mesh $140k$. As a matter of fact, the parameter χ selected following (57) changes in time because of variations of $\|a\|_\infty$. In Fig. 6(a), we see that for $N = 0$ the value of χ over interval $[0.43, 0.68]$ s takes constantly the value 0.1, which corresponds to $||a||_\infty = 1$. When $N = 1$, the value of χ oscillates between 0.1 and 0.17 (see Fig. 6(b)). A larger value of χ corresponds to a more regular velocity detected by the indicator function that gets smaller at the denominator of (57). This is reflected by the longer jets in Fig. 5 that we get with meshes $330k$ and $140k$ for $N = 1$.



(a) normalized axial velocity along z

(b) normalized pressure difference along z

Figure 7. Case $Re_t = 3500$, EFR with mesh $140k$ and four different values of the deconvolution order $N = 0, 1, 2, 3$: comparison between experimental data (solid lines) and numerical results (dashed lines) for (a) normalized axial velocity (70) along the z axis and (b) normalized pressure drop (71) along the z axis. The legend in (b) is common to both figures.

Notice that the values of χ computed with eq. (57) and shown in Fig. 6 are two orders of magnitude larger than Δt used for mesh $140k$ (see Table II). For the problem under consideration, the choice $\chi \simeq \Delta t$ is not appropriate, since it would lead to an under-diffused flow that would not match the experimental data.

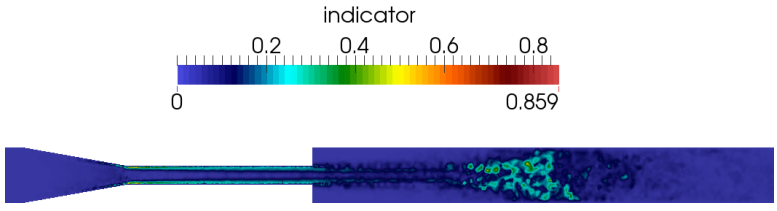
Remark 6.1
We see from Fig. 6 that the value of χ for mesh $140k$ is around 0.1 for $N = 0$ and less that 0.2 for $N = 1$. In the proof of Proposition 4.1, we assumed $\chi \in [0, 1]$. On one hand, χ defined in (57) is clearly positive. In fact, it would be negative when $h < \eta$, which means that the mesh is refined enough for DNS and there is no need for the filtering step. On the other hand, to have $\chi > 1$ for mesh $140k$ (the mesh with the largest ratio h/η among those used at $Re_t = 3500$) one would have to take a time step 10 times larger than the one we used. Such a large time step would be inadequate to follow the physics of the problem. Another possibility to have $\chi > 1$ would be to set $\delta = h_{min}/\sqrt{10}$, which is not appropriate as explained in Remark 4.3

As previously pointed out, the dependence of the solution and the overall smoothing effects on N is not completely clear. In fact, the jet length reduces when we compare the case $N = 1$ to $N = 3$, while the cases $N = 0$ and $N = 2$ produces similar results - see Fig. 7(a) and 7(b). As for the pressure drop, the only outlier is the case $N = 1$, while for $N = 0, 2, 3$ we get similar results. This non-trivial sensitivity to N deserves deeper investigations and it is subject of an ongoing work [6].

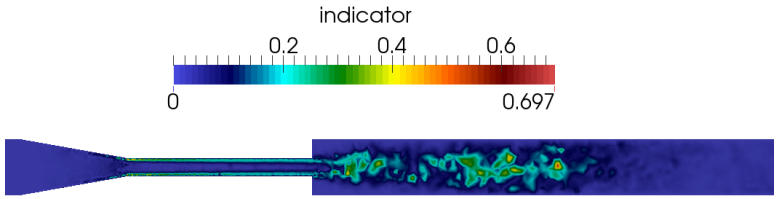
On the other hand, for $N = 0$ and a coarse mesh with $7.2 \cdot 10^4$ elements and $h_{avg} = 1.25 \cdot 10^{-3}$m the simulation crashes less than 0.1 s after reaching regime conditions regardless of the deconvolution order. In this case, overdiffusion is not enough for stabilization purposes.

A key role in the EFR algorithm is played by the *indicator function*. We show in Fig. 8 the indicator function a_{D_0} (see (28) and Remark 3.1 for the definition), computed with meshes $330k$, and $140k$ at the same time step as the velocity magnitudes reported in Fig. 4(b) and 4(c), respectively. The color bars refer to the value of the indicator function on a particular section of the domain, and not on the whole domain. For both cases in Fig. 8, the indicator function takes its largest value in the boundary layer at the entrance of the throat. Moreover, on mesh $140k$ it takes fairly large values all along the jet, while on mesh $330k$ larger values are taken only where the jet breaks down. Fig. 8 shows that a_{D_0} is a suitable indicator function since it correctly selects the regions of the domain where the velocity does need regularization.

Remark 6.2
When $||a||_\infty$ is very small, the value of χ becomes large. In order to avoid dealing with a large

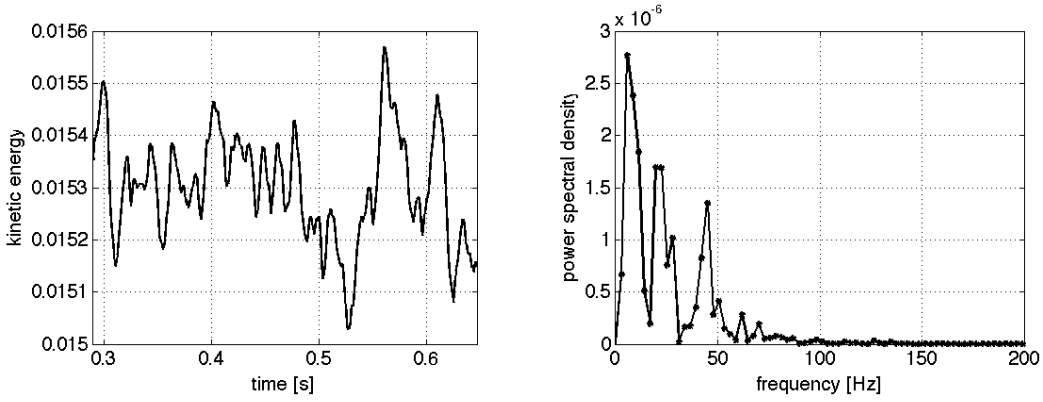


(a) mesh $330k$



(b) mesh $140k$

Figure 8. Case $Re_t = 3500$, EFR with $N = 0$: indicator function a_{D_0} computed with meshes (a) $330k$ and (b) $140k$ at the same time step as the velocity magnitudes reported in Fig. 4(b) and 4(c), respectively.



(a) kinetic energy

(b) power spectral density

Figure 9. $Re = 3500$, DNS with mesh $1900k$: (a) kinetic energy over time interval $[0.29, 0.64]$ s and its (b) power spectral density.

value of χ, in our solver the filter is turned on only when the velocity is sufficiently large to make the current Kolmogorov length scale - computed with the current Reynolds number - smaller than h_{min}.

Finally, let us analyze the evolution of the kinetic energy of the system. We let the DNS with mesh $1900k$ run for several tenths of seconds. The computed kinetic energy is shown in Fig. 9(a): it oscillates around the mean value 0.0153. The periodic character of the solution can be figured out by means of a Fourier analysis of the kinetic energy in time (see, e.g., [4]). The power spectral density of the signal in Fig. 9(a) is reported in Fig. 9(b). Significant frequencies reach up to 50 Hz. In Fig. 10, we see the power spectral density of the kinetic energy computed with the EFR algorithm on meshes $900k$ and $140k$. The comparison pinpoints that when we resort to coarser meshes the amplitude damping at different frequencies is larger. However, the frequency of 50 Hz does not significantly change also with the coarsest meshes. We conclude that the EFR does not affect the dispersion - see Fig. 9(b).

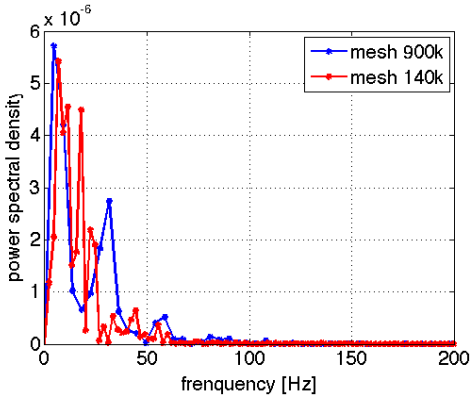


Figure 10. $Re = 3500$, EFR with meshes $900k$ and $140k$: power spectral density of the kinetic energy.

mesh name	h_{min}	h_{avg}	h_{max}	# nodes	# tetrahedra	Δt
$3000k$	1.17e-4	4.76e-4	9.64e-4	5.5e5	3e6	1e-4
$1900k$	1.06e-4	5.15e-4	1.49e-3	3.7e5	1.9e6	1e-4
$900k$	1.09e-4	5.16e-4	1.87e-3	1.8e5	9e5	1e-4
$330k$	2.23e-4	9.48e-4	1.93e-3	6.5e4	3.3e5	2e-4

Table III. Case $Re_t = 5000$: meshes used for the simulations, with their minimum diameter h_{min}, average diameter h_{avg}, maximum diameter h_{max}, and number of nodes and tetrahedra. The units for the diameters are meter. We also report the time step Δt (in s) used for the simulations with each mesh.

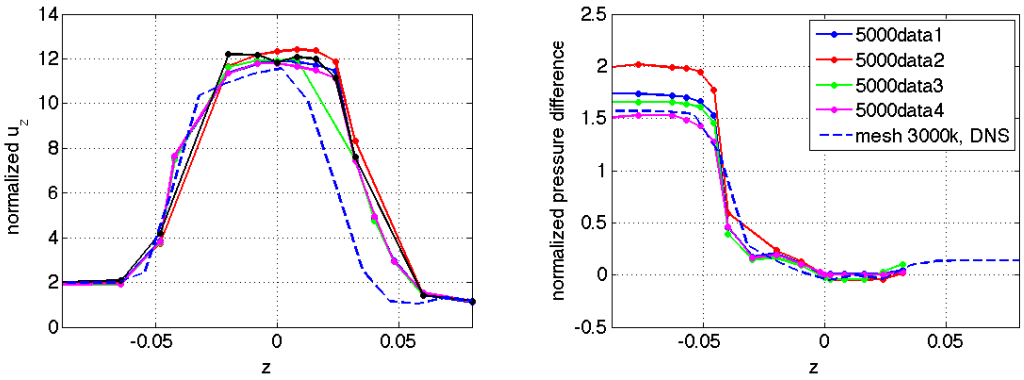
6.2. *Case* $Re_t = 5000$

The second flow regime we consider features a throat Reynolds number $Re_t = 5000$. As for $Re_t = 3500$, turbulence downstream of the sudden expansion was observed in all the experiments with a reproducible jet breakdown point.

We consider several meshes with different levels of refinement. Table III contains the details of all the meshes under consideration, together with the associated time step used in the simulations. Meshes $1900k$, $900k$, $330k$ are the same used for the simulations at $Re_t = 3500$, while the finest mesh was generated for this specific case. Despite the large number of nodes and tetrahedra, mesh $3000k$ has an average diameter roughly 24 times larger than the Kolmogorov scale at $Re_t = 5000$ (see Tables I and III) and is not refined enough for a DNS to give results in good agreement with the experimental data. In fact, we see in Fig. 11(a) that the computed velocity underestimates the measured velocity in the throat and the computed jet length is too small due to insufficient diffusion. A better agreement is found between computed and measured pressure drop (see Fig. 11(b)). We remark that the lab whose data are labeled as "data5" provided measurements for for the axial velocity but not of the pressure at $Re_t = 5000$.

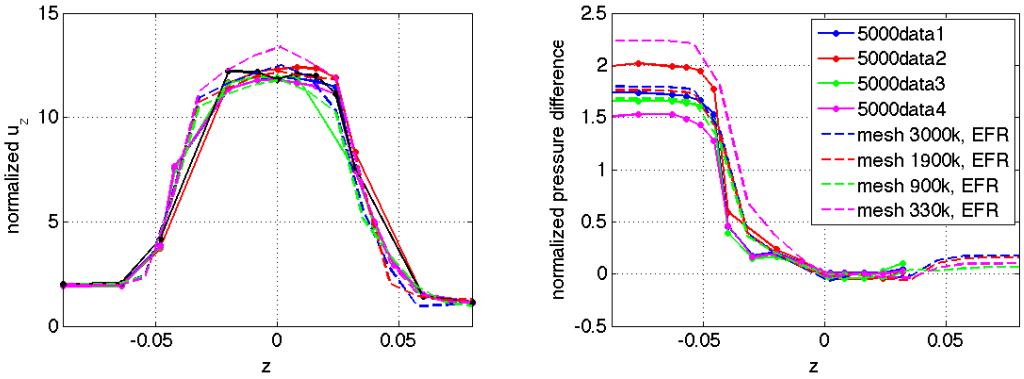
While in the results of Sec. 6.1 the indicator function with $N = 1$ was too overdiffusive, in this case for $Re_t = 5000$ the case $N = 1$ introduces an appropriate amount of viscosity. We report the comparison between computed and measured normalized axial velocity (70) and pressure drop (71) for all the meshes in Table III in Fig. 12(a) and 12(b), respectively. Indeed, from those figures we see that the jet length is very well captured with all the meshes, and the axial velocity and pressure difference computed on meshes $3000k$, $1900k$, and $900k$ are in excellent agreement with the respective measured quantities. For the results on mesh $330k$, we observe that the axial velocity gets overestimated in the throat and the pressure difference is overestimated in the entrance region. A similar behavior for the solution on coarser meshes was observed at $Re_t = 3500$ for $N = 1$ (see Fig. 5).

From Fig. 12, we notice that there is little difference between the results obtained with mesh $3000k$ and those obtained with mesh $900k$, as they both compare well with the experimental data. See also Fig. 13 for a qualitative comparison of the velocity magnitude computed in the two cases



(a) normalized axial velocity along z (b) normalized pressure difference along z

Figure 11. Case $Re_t = 5000$, DNS with mesh $3000k$: comparison between experimental data (solid lines) and numerical results (dashed line) for (a) normalized axial velocity (70) along the z axis and (b) normalized pressure difference (71) along the z axis. The legend in (b) is common to both subfigures.

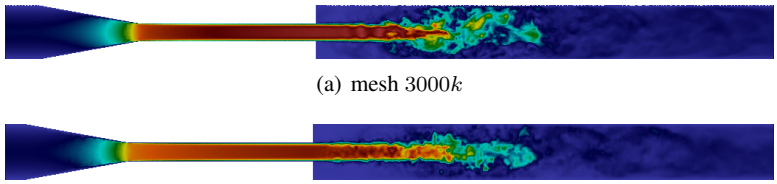


(a) normalized axial velocity along z (b) normalized pressure difference along z

Figure 12. Case $Re_t = 5000$, EFR with four different meshes, $N = 1$: comparison between experimental data (solid lines) and numerical results (dashed lines) for (a) normalized axial velocity (70) along the z axis and (b) normalized pressure difference (71) along the z axis. The legend in (b) is common to both subfigures.

on a section of the domain after the turbulent regime is fully established. The computational time needed for the two simulations is very different: a time step of the EFR algorithm with mesh $3000k$ takes around 280 s (220 s for the evolve step plus 60 s for the filter step) on 208 CPUs and with mesh $900k$ around 220 s (165 s for the evolve step plus 55 s for the filter step) on 96 CPUs. Again the computational times refer to simulations run on Maxwell. Better computational times were achieved on Stampede, a cluster of the XSEDE consortium (around 52 s per iteration on 256 CPUs for mesh $3000k$).

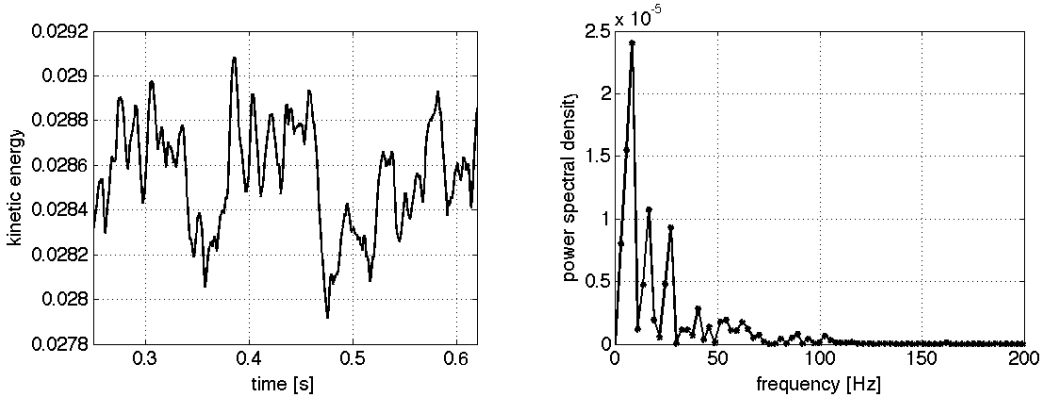
In order to analyze the evolution of the kinetic energy of the system, we let the EFR algorithm with mesh $1900k$ run for several tenths of seconds. The computed kinetic energy is shown in Fig. 14(a) and the power spectral density of the signal in Fig. 14(a) is reported in Fig. 14(b). At $Re_t = 5000$, the kinetic energy oscillates around the mean value 0.0285. Moreover, as expected, more peaks appear in Fig. 14(b) than in Fig. 9(b), which has been obtained with the same mesh at $Re_t = 3500$: at $Re_t = 5000$ ($Re_t = 3500$, resp.) frequencies up to 100 Hz (50 Hz, resp.) have power peaks higher than $5 \cdot 10^{-7}$.



(a) mesh $3000k$

(b) mesh $900k$

Figure 13. Case $Re_t = 5000$, EFR with $N = 1$: velocity magnitude computed with (a) mesh $3000k$ and (b) mesh $900k$ on a section of the domain after the turbulent regime is fully established.



(a) kinetic energy

(b) power spectral density

Figure 14. $Re = 5000$, EFR with mesh $1900k$: (a) kinetic energy over time interval $[0.25, 0.62]$ s and its (b) power spectral density.

7. CONCLUSIONS AND PERSPECTIVES

In this paper, we demonstrated the effectiveness of a nonlinear filtering technique for a realistic 3D flow problem at moderately large Reynolds numbers. Nonlinear filtering stabilizes marginally resolved scales without over-diffusing, thereby allowing to use less degrees of freedom than required by a DNS. To select the regions of the domain where filtering is needed, we employ a deconvolution-based indicator function.

The numerical implementation of this filtering technique is a three-step algorithm that can be realized by a legacy (black-box) Navier-Stokes solver. In fact, step i) is a Navier-Stokes problem that can be handled by the existing solver, step ii) is the filtering problem, which is a Stokes-like problem and can be handled by the same solver without major modification of the code, and step iii) is a simple relaxation step. Such an algorithm can also be interpreted as an operator splitting method for the time discretization of a perturbed version of the Navier-Stokes equations. This interpretation outlines the impact of the filter as an extra-diffusion. In addition, it allows to identify the impact of non-Dirichlet boundary conditions.

We performed a computational study of flow at two different Reynolds numbers in a nozzle geometry and compared the numerical results with experimental measurements acquired by the FDA. The comparison with the measurements demonstrated that the classical choice of the relaxation parameter $\chi \simeq \Delta t$ does not provide enough numerical diffusion. We proposed a new formula based on physical and numerical arguments to tune a value of χ that leads to very good agreement with the experimental measurements. Through the numerical tests, we investigated the role of the deconvolution order. Several meshes were considered to understand how under-refined the mesh can be while still capturing the physical average quantities.

As a follow-up of the present work, inspired by the reinterpretation of the EFR algorithm as an operator splitting method for a perturbed version of the NavierStokes equations, we intend to investigate whether or not the EFR algorithm can be extended to high order methods in time. We also plan to perform an extensive sensitivity analysis for the systematic study of the influence of each parameter in the model on the solution. Extensive testing at higher Reynolds number will be carried out too.

ACKNOWLEDGMENTS

The research presented in this work was carried out during Dr. Quaini's visit at Department of Mathematics and Computer Science at Emory University in the spring semester 2014, supported by the Institute for Quantitative Theory and Methods (QuanTM). This support is gratefully acknowledged. This research has been supported in part by the NSF under grants DMS-1262385 and DMS-1109189 (Quaini).

REFERENCES

1. Computational Fluid Dynamics: An FDA's Critical Path Initiative. https://fdacfd.nci.nih.gov/.
2. FDA's Critical Path Initiative. http://www.fda.gov/ScienceResearch/SpecialTopics/CriticalPathInitiative/ucm076689.htm.
3. LifeV - A parallel Finite Element library. www.lifev.org.
4. F. Autieri, N. Parolini, and L. Quartapelle. Numerical investigation on the stability of singular driven cavity flow. *Journal of Computational Physics*, 183:1–25, 2002.
5. Y. Bazilevs, V.M. Calo, J.A. Cottrell, T.J.R. 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(14):173 – 201, 2007.
6. L. Bertagna and A. Quaini. Applications of a LES-EFR scheme to moderately high reynolds numbers. in preparation, 2015.
7. J. Borggaard, T. Iliescu, and J.P. Roop. A bounded artificial viscosity large eddy simulation model. *SIAM J. Numer. Anal.*, 47:622–645, 2009.
8. A. L. Bowers, L. G. Rebholz, A. Takhirov, and C. Trenchea. Improved accuracy in regularization models of incompressible flow via adaptive nonlinear filtering. *International Journal for Numerical Methods in Fluids*, 70(7):805–828, 2012.
9. A.L. Bowers and L.G. Rebholz. Numerical study of a regularization model for incompressible flow with deconvolution-based adaptive nonlinear filtering. *Comput. Methods Appl. Mech. Eng.*, 258:1–12, 2013.
10. J.P. Boyd. Two comments on filtering (artificial viscosity) for Chebyshev and Legendre spectral and spectral element methods: Preserving boundary conditions and interpretation of the filter as a diffusion. *Journal of Computational Physics*, 143(1):283 – 288, 1998.
11. H. Brezis. *Functional Analysis, Sobolev Spaces and Partial Differential Equations*. Springer New York, 2011.
12. F. Brezzi. On the existence, uniqueness and approximation of saddle point problems arising from Lagrange multipliers. *RAIRO Anal. Numer.*, 8:129–151, 1974.
13. A.N. Brooks and T.J.R. Hughes. Streamline upwind/Petrov-Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equation. *Comput. Methods Appl. Mech. Engrg.*, 32:199–259, 1982.
14. S. Deparis, M. A. Fernandez, and L. Formaggia. Acceleration of a fixed point algorithm for fluid-structure interaction using transpiration conditions. *ESAIM: Mathematical Modelling and Numerical Analysis*, 37:601–616, 2003.
15. A. Dunca and Y. Epshteyn. On the Stolz-Adams deconvolution model for the large-eddy simulation of turbulent flows. *SIAM J. Math. Anal.*, 37(6):1980–1902, 2005.
16. Howard C Elman, David Silvester, and Andy Wathen. *Finite elements and fast iterative solvers: with applications in incompressible fluid dynamics*. Oxford University Press, 2014.
17. J.H. Ferziger and M. Peric. *Computational Methods for Fluid Dynamics*. Springer-Verlag, Berlin, 2002.
18. P. Fischer and J. Mullen. Filter-based stabilization of spectral element methods. *Comptes Rendus de l'Academie des Sciences - Series I - Mathematics*, 332(3):265 – 270, 2001.
19. L. Formaggia, A. Quarteroni, and A. Veneziani. *Cardiovascular Mathematics. Modeling and Simulation of the Cyrculatory system*, volume 1 of *Modeling, Simulation and Applications*. Springer, 2009.
20. U. Frisch. *Turbulence: The Legacy of A.N. Kolmogorov*. Cambridge University Press, 1995.
21. E. Garnier, N. Adams, and P. Sagaut. *Large Eddy Simulation for Compressible Flows*. Springer, Berlin, 2009.
22. A. Gauthier, F. Saleri, and A. Veneziani. A Fast Preconditioner for the Incompressible Navier Stokes Equations. *Computing and Visualization in Science*, 6(2):105–112, 2004.
23. M. Germano. Differential filters of elliptic type. *Phys. of Fluids*, 29:1757–1758, 1986.
24. P. Gervasio, F. Saleri, and A. Veneziani. Algebraic fractional-step schemes with spectral methods for the incompressible Navier-Stokes equations. *J. Comp. Phys*, 214(1):347–365, 2006.

25. R. Glowinski. *Finite element methods for incompressible viscous flow, in: P.G.Ciarlet, J.-L.Lions (Eds), Handbook of numerical analysis*, volume 9. North-Holland, Amsterdam, 2003.
26. Jean-Luc Guermond, Richard Pasquetti, and Bojan Popov. From suitable weak solutions to entropy viscosity. *Journal of Scientific Computing*, 49(1):35–50, 2011.
27. P. Hariharan, M. Giarra, V. Reddy, S.W. Day, K.B. Manning, S. Deutsch, S.F.C. Stewart, M.R. Myers, M.R. Berman, G.W. Burgreen, E.G. Paterson, and R.A. Malinauskas. Multilaboratory Particle Image Velocimetry analysis of the FDA benchmark nozzle model to support validation of Computational Fluid Dynamics simulations. *J. Biomech. Engrg.*, 133:041002, 2011.
28. J Heywood, R Rannacher, and S Turek. Artificial Boundaries and Flux and Pressure Conditions for the Incompressible Navier-Stokes Equations. *International Journal for Numerical Methods in Fluids*, 22:325–352, January 1996.
29. J.C. Hunt, A.A. Wray, and P. Moin. Eddies stream and convergence zones in turbulent flows. Technical Report CTR-S88, CTR report, 1988.
30. A. N. Kolmogorov. Dissipation of energy in isotropic turbulence. *Dokl. Akad. Nauk. SSSR*, 32:19–21, 1941.
31. A. N. Kolmogorov. The local structure of turbulence in incompressible viscous fluids at very large Reynolds numbers. *Dokl. Akad. Nauk. SSSR*, 30:299–303, 1941.
32. W. Layton, L.G. Rebholz, and C. Trenchea. Modular nonlinear filter stabilization of methods for higher Reynolds numbers flow. *J. Math. Fluid Mech.*, 14:325–354, 2012.
33. J. Leray. Essai sur le mouvement d'un fluide visqueux emplissant l'espace. *Acta. Math*, 63:193–248, 1934.
34. J. Mathew, R. Lechner, H. Foysi, J. Sesterhenn, and R. Friedrich. An explicit filtering method for large eddy simulation of compressible flows. *Physics of Fluids*, 15(8):2279–2289, 2003.
35. J. Mullen and P. Fischer. Filtering techniques for complex geometry fluid flows. *Communications in Numerical Methods in Engineering*, 15(1):9–18, 1999.
36. M.A. Olshanskii and X. Xiong. A connection between filter stabilization and eddy viscosity models. *Numerical Methods for Partial Differential Equations*, 29(6):2061–2080, 2013.
37. T. Passerini, A. Quaini, U. Villa, A. Veneziani, and S. Canic. Validation of an open source framework for the simulation of blood flow in rigid and deformable vessels. *International Journal for Numerical Methods in Biomedical Engineering*, 29(11):1192–1213, 2013.
38. S.B Pope. *Turbulent flows*. Cambridge University Press, Cambridge, 2000.
39. A. Quarteroni, R. Sacco, and F. Saleri. *Numerical Mathematics*. Springer Verlag, 2007.
40. A. Quarteroni, F. Saleri, and A. Veneziani. Analysis of the Yosida method for the incompressible Navier-Stokes equations. *J. Math. Pures Appl.*, 78:473–503, 1999.
41. A. Quarteroni, F. Saleri, and A. Veneziani. Factorization methods for the numerical approximation of Navier-Stokes equations. *Comput. Methods Appl. Mech. Engrg.*, 188:505–526, 2000.
42. A. Quarteroni and A. Valli. *Numerical Approximation of Partial Differential Equations*. Springer-Verlag, 1994.
43. O. Reynolds. An experimental investigation of the circumstances which determine whether the motion of water shall be direct or sinuous and of the law of resistance in parallel channels. *Proc. R. Soc. Lond. Ser.*, 35:84–99, 1883.
44. F. Saleri and A. Veneziani. Pressure Correction Algebraic Splitting Methods for the Incompressible Navier-Stokes Equations. *SIAM Journal on Numerical Analysis*, 43(1):174–194, 2006.
45. S.F.C. Stewart, E.G. Paterson, G.W. Burgreen, P. Hariharan, M. Giarra, V. Reddy, S.W. Day, K.B. Manning, S. Deutsch, M.R. Berman, M.R. Myers, and R.A. Malinauskas. Assessment of CFD performance in simulations of an idealized medical device: Results of FDA's first computational inter laboratory study. *Cardiovascular Engineering and Technology*, 3(2):139–160, 2012.
46. S. Stolz and N.A. Adams. An approximate deconvolution procedure for large-eddy simulation. *Physics of Fluids*, 11(7):1699–1701, 1999.
47. S. Stolz, N.A. Adams, and L. Kleiser. An approximate deconvolution model for large-eddy simulation with application to incompressible wall-bounded flows. *Physics of Fluids*, 13(4):997–1015, 2001.
48. R. Temam. *Navier-Stokes Equations*. North-Holland, 1984.
49. H. Tennekes and J.L. Lumley. *A first course in turbulence*. MIT Press, Cambridge, 1972.
50. M.R. Visbal and D.P. Rizzetta. Large eddy simulation on curvilinear grids using compact differencing and filtering schemes. *J. Fluids Eng.*, 124:836–847, 2002.
51. A.W. Vreman. An eddy-viscosity subgrid-scale model for turbulent shear flow: Algebraic theory and applications. *Physics of Fluids*, 16(10):3670–3681, 2004.