J. Math. Fluid Mech. (2026) 28:30
© 2026 The Author(s)
https://doi.org/10.1007/s00021-026-01009-6

Journal of Mathematical Fluid Mechanics

Revisiting the Slip Boundary Condition: Surface Roughness as a Hidden Tuning Parameter

Matthias Maier, Peter Munch and Murtazo Nazarov

Communicated by S. Turek

Abstract. In this paper, we investigate the effect of boundary surface roughness on numerical simulations of incompressible fluid flow past a cylinder in two and three spatial dimensions furnished with slip boundary conditions. The governing equations are approximated using a continuous finite element method, stabilized with a Galerkin least-squares approach. Through a series of numerical experiments, we demonstrate that: (*i*) the introduction of surface roughness through numerical discretization error, or mesh distortion, makes the potential flow solution unstable; (*ii*) when numerical surface roughness and mesh distortion are minimized by using high-order isoparametric geometry mappings, a stable potential flow is obtained in both two and three dimensions; (*iii*) numerical surface roughness, mesh distortion and refinement level can be used as control parameters to manipulate drag and lift forces resulting in numerical values spanning more than an order of magnitude. Our results cast some doubt on the predictive capability of the slip boundary condition for wall modeling in turbulent simulations of incompressible flow.

Mathematics subject classification. 65M30, 65M60.

Keywords. Incompressible Navier-Stokes equations, Finite element approximation, Galerkin least-squares stabilization, G2, Slip boundary conditions.

1. Introduction

We investigate the effect of boundary surface roughness on numerical simulation of incompressible fluid flow past a cylinder in two and three spatial dimensions equipped with slip boundary conditions. We focus on the classical bluff body problem, specifically, flow past a cylinder in both two and three spatial dimensions. Our main objective is to investigate how the surface roughness of the cylinder—introduced by numerical discretization error and mesh distortion—influences the qualitative nature and quantitative behavior of the simulated flow. Understanding such an influence on the qualitative and quantitative behavior of the flow is of particular importance as using slip boundary conditions for wall modeling in turbulent simulations has been used extensively [3,10,12,18,19,27].

1.1. Incompressible Fluid Flow, the Invsicid limit, and Boundary Conditions

Incompressible fluid flow is modeled mathematically by the following Navier-Stokes equations: Let Ω be an open domain in $\mathbb{R}^d$, $d = 2, 3$, and let $T > 0$ be the final time. We seek a velocity field $\boldsymbol{u} := (u_1, \dots, u_d)$ and a pressure p solving the partial differential equations

$$\begin{cases} \partial_t \boldsymbol{u} + \boldsymbol{u}\cdot\nabla\boldsymbol{u} + \nabla p - 2\nu\nabla\cdot\boldsymbol{\varepsilon}(\boldsymbol{u}) = 0, \\ \nabla\cdot\boldsymbol{u} = 0, \end{cases} \tag{1.1}$$

Published online: 04 March 2026

for every $(\boldsymbol{x}, t) \in \Omega \times (0, T]$, in some week sense [36] and furnished with appropriate boundary and initial conditions. Here, $\boldsymbol{\varepsilon}(\boldsymbol{u}) := \frac{1}{2}(\nabla \boldsymbol{u} + \nabla \boldsymbol{u}^\top)$ is the rate-of-strain tensor with $\nu \geq 0$ being the kinematic viscosity of the fluid.

We are interested in the case where the kinematic viscosity vanishes, i.e., $\nu \to 0$, which corresponds to the inviscid flow regime. It is usually believed [5,23] that the inviscid limit of the Navier–Stokes equations is given by the Euler equations,

$$\begin{cases} \partial_t \boldsymbol{u} + \boldsymbol{u} \cdot \nabla \boldsymbol{u} + \nabla p = 0, \\ \nabla \cdot \boldsymbol{u} = 0, \end{cases} \tag{1.2}$$

at least away from boundaries. However, establishing this limit for bounded domains with, for example, Dirichlet boundary conditions remains a challenging problem; see, for instance, [23,26] and references therein.

For the Navier–Stokes equations, the natural boundary condition on solid walls is the *no-slip* condition, $\boldsymbol{u} = 0$, which enforces both zero normal and tangential velocity at the boundary. At high Reynolds numbers this creates a sharp boundary layer [35]. In contrast, the natural boundary condition for the Euler equations on solid walls is the no-penetration condition, $\boldsymbol{u} \cdot \boldsymbol{n} = 0$, also known as the *slip* boundary condition, which prevents flow through the boundary while allowing tangential motion. Here $\boldsymbol{n}$ is the unit normal vector to the boundary. Consequently, no boundary layer emerges. This discrepancy between the viscous and inviscid solutions at the boundary is difficult to reconcile and the reason why the analytical investigation of the inviscid limit of Navier–Stokes solutions remains challenging [23].

The so-called Navier-slip boundary condition [29] combines both slip and no-slip conditions:

$$\begin{cases} \boldsymbol{u} \cdot \boldsymbol{n} = 0, \\ \beta \boldsymbol{u} \cdot \boldsymbol{t}_k + 2\nu(\boldsymbol{\varepsilon}(\boldsymbol{u}) \cdot \boldsymbol{n}) \cdot \boldsymbol{t}_k = 0, \quad k = 1, 2, \end{cases} \tag{1.3}$$

where $\boldsymbol{t}_k$ are orthogonal unit tangent vectors to $\boldsymbol{n}$ on the boundary, and β is a friction coefficient. Note that $\beta \to 0$ recovers the slip boundary condition, while $\beta \to \infty$ corresponds to enforcing a no-slip boundary condition. Iftimie and Planas [17] proved that the inviscid limit of the Navier–Stokes equations with slip boundary conditions indeed converges to solutions of the Euler equations. This is a significant result with practical implications as slip conditions eliminate the formation of boundary layers, which is beneficial for numerical simulations.

Navier-slip boundary conditions (1.3) with various choices of friction coefficient β is used for *wall-modeled* large-eddy simulation for channel flows, see for instance [3,27]. However, it is not clear whether using Navier-slip boundary conditions is advantageous for bluff body problems, since avoiding boundary layers may lead to a *potential* flow, i.e., an inviscid, incompressible, steady, and irrotational flow, which yields zero net drag force on the body [6]. This phenomenon is famously known as d'Alembert's zero-drag paradox.

1.2. Resolution of d'Alembert's Zero-Drag Paradox

d'Alembert's paradox remained unresolved for almost 150 years. Prandtl [31] famously proposed a resolution to the paradox, arguing that viscous effects near solid boundaries cannot be neglected and that the no-slip boundary condition must be imposed. In other words, this resolution of the paradox asserts that inviscid models, such as the Euler equations, cannot be used to accurately calculate the drag or lift forces on a body [31].

Recently, Hoffman and Johnson [12] proposed a different point of view for resolving d'Alembert's paradox by arguing that potential flow is unstable at the separation points in three space dimensions. This instability then naturally leads to the development of a turbulent flow with substantial drag caused by streaks of low-pressure streamwise vorticity generated by the breakdown of the potential flow near the rear separation region [12]. This new perspective has opened a novel direction for turbulent flow simulation, as it eliminates the need to resolve the computationally expensive Prandtl boundary layer: When numerically approximating the flow past a body using the inviscid Euler equations with slip boundary conditions,

while initializing the solution as potential flow, the flow should become unstable. Computational results reported in [10,15,16,20,37] demonstrated that simulating flow past various objects–including a full car, an airplane, and landing gear–using slip boundary conditions produced turbulent solutions with drag and lift coefficients in close agreement with experimental data.

An interesting observation is that these two resolutions appear to fundamentally contradict each other as they assume an opposite viewpoint on the role of the boundary layer. Prandtl's resolution on the one hand is supported by extensive experimental evidence of flows studied in the laboratory [31], whereas the viewpoint assumed by Hoffman and Johnson [12] is corroborated by computational evidence [10,15,16,20,37]. This raises a fundamental question: which of these two viewpoints should be regarded as the correct resolution of d'Alembert's paradox?

1.3. Surface Roughness as a Hidden Tuning Parameter

To the best of our knowledge, all of the numerical evidence [10,12,15,16,20,37] in support of Hoffman and Johnson [12] has been obtained from simulations using the `Unicorn` solver [13,14], which is built upon the `FEniCS` project [24]. `Unicorn` uses an inconsistent Galerkin least-squares (GLS) stabilized finite element discretization on unstructured meshes [4,21]. We note that a GLS stabilized finite element approximation—as is the case for any other discretization technique—contains a number of hidden, or implicit, modeling parameters, which can significantly affect the computational results. For simulating flow past a cylinder, which is the subject of our paper, we note the following hidden tuning parameters:

(i) Nonuniform artificial viscosity: the GLS stabilization introduces an artificial viscosity that depends on the local element size. For unstructured meshes, this leads to a nonuniform dissipation, potentially affecting the instability of the potential solution.
(ii) Geometry approximation error: the curvature of the cylinder must be resolved accurately. This requires high-order isoparametric mappings to reduce the geometric error.
(iii) Choice of finite element spaces: when using piecewise linear elements, the curved boundary of the cylinder is represented as a polygonal approximation, introducing additional errors. To reduce both geometry and discretization errors, higher-order polynomial spaces are needed.

At the time when `Unicorn` was conceptualized and implemented, `FEniCS` supported only unstructured tetrahedral meshes with piecewise linear geometric approximations. As a result, none of these points have been addressed.

It is the purpose of the present paper to investigate numerically whether and how these hidden tuning parameters can affect the stability of flow past a cylinder with slip boundary conditions.

1.4. Our Contribution

The objective of this paper is to reproduce the numerical experiments originally reported by Hoffman and Johnson [12] for a three dimensional cylinder geometry while addressing the three items outlined above. For the sake of comparison we choose to use a similar discretization technique that was outlined in [12], namely a Galerkin least-squares (GLS) stabilized finite element discretization on unstructured meshes [4,21]. We want to emphasize that it is not our objective to develop, or improve on the GLS discretization and stabilization approach, or judge the merits of it.

In [30, Sec. 5.2], subsonic flow past a circular cylinder in 2D was investigated using an Euler solver with a slip boundary condition. Even at extreme mesh resolutions (down to element sizes of $\approx 10^{-7}$), a significant deviation from potential flow was observed, indicating the need for more accurate geometric representation and numerical resolution. We thus base the solver for the current investigation on the open source finite element toolkit `deal.II` [1,2], that supports quadrilateral and hexahedral meshes along with high-order isoparametric mappings for curved geometries. Our solver addresses the three key limitations outlined above: it enables perfectly symmetric, structured grids, avoiding artificial dissipation

from mesh nonuniformity; it supports high-order manifold descriptions, significantly reducing geometric approximation errors; it allows for arbitrary high-order finite element spaces, improving accuracy in both velocity and pressure approximations. We have made our solver publicly available for verification and further experimentation [25].

Our main observations that are discussed in detail in Sections 3 and 4 can be summarized as follows:

(i) GLS simulations of incompressible, high Reynolds number flow past a 2D cylinder using slip boundary conditions—while minimizing the geometry error (by using a quasi-uniform mesh, high order boundary approximation, and isoparametric discretization)—results in a stable potential flow profile with vanishing drag and lift; see Section 3.3.

(ii) The flow profile and resulting drag and lift values can be manipulated dramatically by introducing surface roughness; see Section 3.3. We observe quantitative changes for flow statistics of 3 to 5 orders of magnitude.

(iii) We confirm that this behavior carries over into 3D: When minimizing the geometry error (by using quasi-uniform mesh, high order boundary approximation, and isoparametric discretization) we observe that GLS simulations of incompressible, high Reynolds number flow past a 3D cylinder using slip boundary conditions results again in a stable potential flow profile with vanishing drag and lift; see Section 3.4.

(iv) We document quantitatively how the stability and resulting drag and lift values of 3D potential flow is affected again by the introduction of (a) surface roughness (Section 3.4) and (b) mesh distortion (Section 3.5).

This indiccates that numerical surface roughness and mesh distortion can be used as control parameters to manipulate drag and lift forces resulting in numerical values spanning various orders of magnitude. This casts some doubt on the predictive capability of the slip boundary condition for wall modeling in turbulent simulations of incompressible flow.

1.5. Paper Organization

The remainder of the paper is organized as follows. In Section 2 we summarize the full and inconsistent Galerkin least-squares (GLS) stabilized finite element discretizations, our strategy for boundary approximations, and aspects of our solver implementation. We conduct a series of validation tests, as well as the aforementioned parameter studies in Section 3. We summarize our findings, discuss implications and conclusions in Section 4.

2. Finite Element Approximation

For our numerical parameter studies reported in Section 3 we use the same discretization technique that was outlined in [12], a Galerkin least-squares (GLS) stabilized finite element discretization on unstructured meshes. For the sake of completeness we now summarize the discretization technique. We have made our implementation publicly available for verification and further experimentation [25].

We denote by $\mathcal{T}_h$ a subdivision of Ω into a finite number of disjoint elements K such that $\overline{\Omega} = \cup_{k\in\mathcal{T}_h}\overline{K}$, where $\overline{\Omega}$ and $\overline{K}$ denotes the closure of Ω and K, respectively. Let us consider a family of shape-regular and conforming meshes $\{\mathcal{T}_h\}_{h>0}$, where h denotes the smallest diameter of all triangles of $\mathcal{T}_h$. Next, we denote by $\boldsymbol{g}_K : \widehat{K} \mapsto K$ the diffeomorphism that maps the reference element $\widehat{K}$ to the real cell K. With each mesh $\mathcal{T}_h$ we associate the following continuous approximation space:

$$\mathcal{X}_h := \{v_h : v_h \in \mathcal{C}^0(\overline{\Omega});\ \forall K \in \mathcal{T}_h,\ v_h|_K \circ \boldsymbol{g}_K \in \mathbb{Q}_k(\widehat{K})\}, \tag{2.1}$$

where $\mathbb{Q}_k$ is the set of multivariate polynomials of total degree at most $k \geq 1$ defined over $\widehat{K}$. For every element K, we define a mesh-size h_K as the minimum distance between any pair of its vertices.

2.1. Space and Time Discretization of the Navier-Stokes Equations

For high Reynolds numbers, the Navier-Stokes equations (1.1) are generally convection dominiated. Consequently, finite element discretizations have to be stabilized in some form. We follow the approach in [12] and stabilized the system using the so-called General Galerkin method. Next, we split the time interval $[0,T]$ into N intervals of variable size: $0 = t_0 < t_1 < \ldots < t_N = T$, and let $\tau^n = t_{n+1} - t_n$ is the local time step. Given previous approximations $(\boldsymbol{u}_h^k, p_h^k) \in \boldsymbol{\mathcal{V}}_h \times \mathcal{X}_h$ for time t_k, $k < n$, and where $\boldsymbol{\mathcal{V}}_h = [\mathcal{X}_h]^d$, we construct a finite element approximation $(\boldsymbol{u}_h^n, p_h^n) \in \boldsymbol{\mathcal{V}}_h \times \mathcal{X}_h$ for time t_n by solving

$$F(\boldsymbol{u}_h^n, p_h^n; \boldsymbol{v}, q) + S(\boldsymbol{u}_h^n, p_h^n; \boldsymbol{v}, q) = 0, \quad \forall (\boldsymbol{v}, q) \in \boldsymbol{\mathcal{V}}_h \times \mathcal{X}_h. \tag{2.2}$$

Here, $F(\boldsymbol{u}_h^n, p_h^n)$ is a fully discrete, weak form corresponding to (1.1):

$$\begin{aligned} F(\boldsymbol{u}_h^n, p_h^n; \boldsymbol{v}, q) := \big(D_\tau \boldsymbol{u}_h^n, \boldsymbol{v}\big) + \big(\boldsymbol{u}_h^n \cdot \nabla \boldsymbol{u}_h^n, \boldsymbol{v}\big) - \big(p_h^n, \nabla \cdot \boldsymbol{v}\big) + \big(2\nu \boldsymbol{\varepsilon}(\boldsymbol{u}_h^n), \boldsymbol{\varepsilon}(\boldsymbol{v})\big) \\ + \big(\nabla \cdot \boldsymbol{u}_h^n, q\big) - (2\nu \boldsymbol{n}_h \cdot \boldsymbol{\varepsilon}(\boldsymbol{u}_h^n), \boldsymbol{v})_{\partial\Omega} + (p_h^n, \boldsymbol{n}_h \cdot \boldsymbol{v})_{\partial\Omega}, \end{aligned} \tag{2.3}$$

where $D_\tau \boldsymbol{u}_h^n$ is the usual backward differentiation formula of order two (BDF2) to approximate the time derivative of $\boldsymbol{u}_h$:

$$D_\tau \boldsymbol{u}_h^n := \frac{2\tau^n + \tau^{n-1}}{\tau^n(\tau^n + \tau^{n-1})} \boldsymbol{u}_h^n - \frac{\tau^n + \tau^{n-1}}{\tau^n \cdot \tau^{n-1}} \boldsymbol{u}_h^{n-1} + \frac{\tau^n}{\tau^{n-1} \cdot (\tau^n + \tau^{n-1})} \boldsymbol{u}_h^{n-2}.$$

Here, we compute the local time step using the following CFL-type condition: $\tau_n := \text{cfl}\ h_{\min,\Omega}/|\boldsymbol{u}_h^n|$, where the constant $\text{cfl} = 1$, unless otherwise stated, is used throughout all numerical simulations presented in this paper.

For convection dominated problems, standard Galerkin methods may become unstable and require stabilization. To address this, various mesh-dependent stabilization techniques such as the well-known *Galerkin least squares* (GLS) stabilization [4,21] have been introduced. Such stabilization approaches augment the partial differential equations with mesh-dependent stabilization terms, that help to stabilize the numerical solution. Following [4,12,21], we introduce a GLS stabilization term $S_{\text{GLS}}(\boldsymbol{u}_h^n, p_h^n; \boldsymbol{v}, q)$:

$$\begin{aligned} S_{\text{GLS}}(\boldsymbol{u}_h^n, p_h^n; \boldsymbol{v}, q) := \sum_{K \in \mathcal{T}_h} \Big(\delta_1^K \big(D_\tau \boldsymbol{u}_h^n + \boldsymbol{u}_h^n \cdot \nabla \boldsymbol{u}_h^n + \nabla p_h^n\big),\ \boldsymbol{u}_h^n \cdot \nabla \boldsymbol{v} + \nabla q\Big)_K \\ + \sum_{K \in \mathcal{T}_h} \big(\delta_2^K \nabla \cdot \boldsymbol{u}_h^n,\ \nabla \cdot \boldsymbol{v}\big)_K, \end{aligned} \tag{2.4}$$

where the stabilization coefficients are given by

$$\delta_1^K = c_1 \left(\frac{1}{\Delta t_n^2} + \frac{\max_K |\boldsymbol{u}_h^{n-1}|^2}{h_K^2} \right)^{-\frac{1}{2}}, \quad \delta_2^K = c_2 h_K,$$

when $\nu < |\boldsymbol{u}_h^{n-1}| h_K$ and

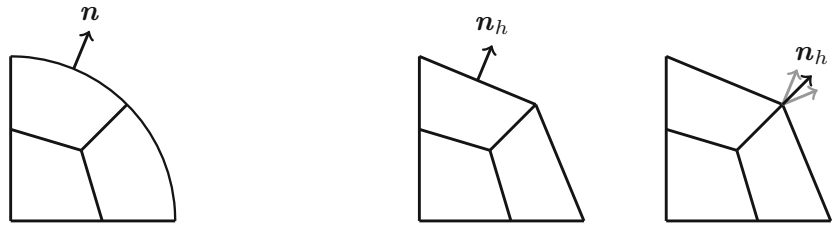
$$\delta_1^K = c_1 h_K^2, \quad \delta_2^K = c_2 h_K^2,$$

if $\nu \geq |\boldsymbol{u}_h^{n-1}| h_K$. We have set the stabilization parameters to $c_1 = 1$, or 2, and $c_2 = 1$ in our numerical simulations.

Note that the stabilization term (2.4) can be simplified by dropping the terms for the time derivative leading to a simplified stabilization term [11]:

$$\begin{aligned} S_{\text{G2}}(\boldsymbol{u}_h^n, p_h^n; \boldsymbol{v}, q) := \sum_{K \in \mathcal{T}_h} \Big(\delta_1^K \big(\boldsymbol{u}_h^n \cdot \nabla \boldsymbol{u}_h^n + \nabla p_h^n\big),\ \boldsymbol{u}_h^n \cdot \nabla \boldsymbol{v} + \nabla q\Big)_K \\ + \sum_{K \in \mathcal{T}_h} \big(\delta_2^K \nabla \cdot \boldsymbol{u}_h^n,\ \nabla \cdot \boldsymbol{v}\big)_K. \end{aligned} \tag{2.5}$$

This form of inconsistent stabilization— the stabilization term without time derivatives and without viscous terms— has been criticized in the literature, primarily due to its lack of full consistency with the underlying differential equations. However, it offers a significant simplification in implementation,



(a) Idealized, curved boundary (b) Approximation with bilinear mapping

FIG. 1. (a) Idealized, curved boundary with outward facing boundary normal $\boldsymbol{n}(\boldsymbol{x})$; (b) corresponding approximation $\boldsymbol{n}_h(\boldsymbol{x})$ for the case of a boundary approximated with a bilinear mapping. Here, the normal field on each face differs. In order to reconstruct a normal $\boldsymbol{n}$ in the vertex we average the normals coming from all adjacent faces

particularly by avoiding time-dependent test spaces, which are often challenging to handle in practice. Despite its inconsistency, this approach has shown promising performance in turbulent flow simulations, as demonstrated in [11].

We have observed largely similar results for flow past a cylinder when using S_{GLS} and S_{G2}. We have thus opted to use the consistent stabilization term S_{GLS} for the majority of numerical experiments. We comment on some selected computational results using S_{G2} as well.

2.2. Boundary Conditions and Boundary Approximation

In the numerical experiments outlined below the discretized system (2.2) is closed by enforcing various boundary conditions. We use

- strongly enforced Dirichlet conditions at the inflow prescribing the velocity field as

$$\boldsymbol{u}_h^n(\boldsymbol{x}) = \boldsymbol{u}_{\mathrm{D}}(\boldsymbol{x}), \quad \text{for } \boldsymbol{x} \in \Gamma_{\mathrm{inlet}},$$

  where $\boldsymbol{u}_{\mathrm{D}}(\boldsymbol{x})$ denotes the imposed inflow profile;
- weekly enforced Dirichlet conditions on the outflow controlling the velocity field via a Nitsche method described below,
- and on the cylinder surface and channel walls, we impose either no-slip, $\boldsymbol{u}_h^n(\boldsymbol{x}) = 0$, or slip boundary conditions, $\boldsymbol{u}_h^n(\boldsymbol{x}) \cdot \boldsymbol{n}_h = 0$.

Here, $\boldsymbol{n}_h$ denotes a discrete reconstruction of the outward facing unit normal field on the boundary $\Gamma = \partial\Omega$.

On the outlet Γ_{outlet} we enforce weakly—via Nitsche's method—a velocity field to prevent recirculation at high Reynolds numbers. This is done by augmenting the stabilized equation (2.2) with a boundary term:

$$\begin{aligned} N(\boldsymbol{u}_h^n, p_h^n; \boldsymbol{v}, q) = &-\nu(\partial_n \boldsymbol{u}_h^n, \boldsymbol{v})_{\Gamma_{\mathrm{outlet}}} - \nu(\boldsymbol{u}_h^n - \boldsymbol{u}_{\mathrm{D}}^n, \partial_n \boldsymbol{v})_{\Gamma_{\mathrm{outlet}}} \\ &+ \left(\frac{\beta_{\mathrm{D}}}{h} (\boldsymbol{u}_h^n - \boldsymbol{u}_{\mathrm{D}}^n), \boldsymbol{v} \right)_{\Gamma_{\mathrm{outlet}}}, \end{aligned} \tag{2.6}$$

with $\beta_{\mathrm{D}} = 1$. Whereas the Dirichlet, slip and no slip boundary conditions are enforced strongly, meaning they are directly enforced in the finite element function space rather than through weak imposition or penalty methods. For the slip boundary condition, we accomplish this by first reconstructing the normal vector at each collocation point on the boundary, as described in the next section. Then, one component of the velocity is expressed as a linear combination of the others. For example, in 2D, the condition $\boldsymbol{u}_h^n \cdot \boldsymbol{n}_h = 0$ expands to $u_{h,0}^n n_0 + u_{h,1}^n n_1 = 0$, which implies $u_{h,0}^n = -\frac{n_1}{n_0} u_{h,1}^n$. This is done in `deal.II` by constructing an additional constrained matrix similar to the process outlined in [7, Sec. 3.4].

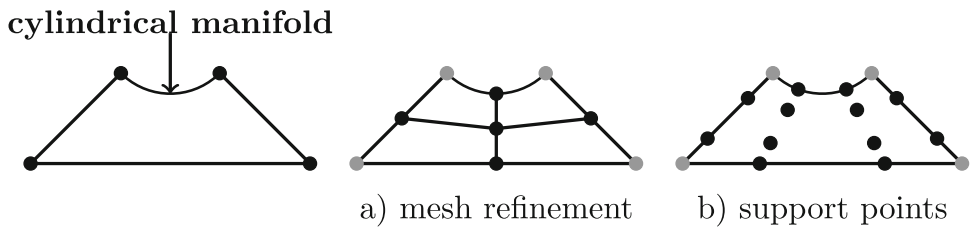


FIG. 2. Usage of a high-order manifold and mapping in `deal.II`: The curvature information is used for (a) the generation of new mesh points during mesh refinement, and (b) for the placement of support points. Here, exemplified for a cubic finite element with corresponding cubic bilinear mapping

Reconstruction of normal vectors For imposing the boundary conditions outlined above we need to reconstruct an outward facing unit normal field $\boldsymbol{n}_h(\boldsymbol{x})$ approximating the normal field $\boldsymbol{n}$ on the boundary $\Gamma = \partial\Omega$. In our numerical experiments we will use bilinear (or trilinear), as well as higher-order isoparametric mappings that approximate the (true) curvature of the curved boundary Γ to various polynomial degrees; see Figure 1. On the interior of a face the discrete normal $\boldsymbol{n}_h(\boldsymbol{x})$ is well defined and we use the exact discrete normal imposed by the mapping. On collocation points falling on face boundaries (vertices, or edges), however, the discrete normal field is multi valued. In order to reconstruct a normal $\boldsymbol{n}_h$ in such points we average the various normals coming from all adjacent faces; see Figure 1(b).
Low-order bilinear (trilinear) and high-order isoparametric mappings `deal.II` supports various *mapping* strategies for constructing the approximation space $\mathcal{X}_h$ via (2.1). In addition a *manifold* object can be attached to the mesh that describes the curvature of the geometry that is used for mesh refinement, as well as for constructing the mapping and placement of support points; see Figure 2. We refer the reader to [7] for a detailed description of the manifold concept and its implementation in `deal.II`. Attaching such high-order manifold allows us to position exactly the new points generated during mesh refinement and also needed for high-order mapping, as indicated in Figure 2.

For the sake of completeness we describe briefly how the normal is computed for a given face. Recall that $\boldsymbol{g}_K : \widehat{K} \mapsto K$ denotes the mapping between the reference element $\widehat{K}$ and the real cell K. Let $(\mathbb{J}_K(\widehat{\boldsymbol{x}}))_{ij} = \partial_{\widehat{x}_j}(\boldsymbol{g}_K)_i$ denote the Jacobian of $\boldsymbol{g}_K(\widehat{x})$. The normal $\boldsymbol{n}_h(\boldsymbol{x})$ on a face is now given by

$$\boldsymbol{n}_h(\boldsymbol{x}) = \frac{\mathbb{J}_K^{-T}(\widehat{\boldsymbol{x}})\widehat{\boldsymbol{n}}(\widehat{\boldsymbol{x}})}{\left\|\mathbb{J}_K^{-T}(\widehat{\boldsymbol{x}})\widehat{\boldsymbol{n}}(\widehat{\boldsymbol{x}})\right\|},$$

where $\widehat{\boldsymbol{x}} = \boldsymbol{g}_K^{-1}(\boldsymbol{x})$, and $\widehat{\boldsymbol{n}}$ is the corresponding normal of $\partial\widehat{K}$ in $\widehat{\boldsymbol{x}}$.

2.3. Solution Procedure and Linear Solver

Given previous approximations $(\boldsymbol{u}_h^k, p_h^k) \in \boldsymbol{\mathcal{V}}_h \times \mathcal{X}_h$ for time t_k, $k < n$, we need to find the solution $(\boldsymbol{u}_h^n, p_h^n)$ to the nonlinear equation (2.2), viz., $\forall (\boldsymbol{v}, q) \in \boldsymbol{\mathcal{V}}_h \times \mathcal{X}_h$:

$$B(\boldsymbol{u}_h^n, p_h^n; \boldsymbol{v}, q) := F(\boldsymbol{u}_h^n, p_h^n; \boldsymbol{v}, q) + S(\boldsymbol{u}_h^n, p_h^n; \boldsymbol{v}, q) + N(\boldsymbol{u}_h^n, p_h^n; \boldsymbol{v}, q) \;=\; 0.$$

Starting from an initial guess $(\boldsymbol{u}_h^{(0)}, p_h^{(0)})$ for $(\boldsymbol{u}_h^n, p_h^n)$ we construct a sequence of approximations with a Newton iteration:

$$\begin{aligned} B'_{\boldsymbol{u},p}(\boldsymbol{u}^{(k)}, p^{(k)}; \boldsymbol{v}, q)[\delta\boldsymbol{u}_h^n, \delta p_h^n] &= -B(\boldsymbol{u}^{(k)}, p^{(k)}; \boldsymbol{v}, q), \\ \boldsymbol{u}^{(k)} &= \boldsymbol{u}^{(k)} + \delta\boldsymbol{u}^{(k)}, \\ p^{(k)} &= p^{(k)} + \delta p^{(k)}. \end{aligned} \tag{2.7}$$

TABLE 1. *Comparison of runtime of the "flow past a 2D cylinder with $Re = 100$" configuration (Section 3.1): all variants are run with $r = 5$, $cfl = 1$, and isoparametric mapping, except when otherwise indicated: the second $\mathbb{Q}_1$ computation on refinement level $r = 5$ is computed with $cfl = 2$ for direct comparison with $\mathbb{Q}_2$; the second $\mathbb{Q}_2$ variant is computed with linear mapping*

	$\mathbb{Q}_1$ ($r = 4$)	$\mathbb{Q}_1$ (cfl $= 2$)	$\mathbb{Q}_2$	$\mathbb{Q}_2$ (lin.)
number of cells	22,528	90,112	22,528	22,528
number of DoFs	68,976	273,120	273,120	273,120
tot. simulation time [s]	340.5	1170	960.1	944.5
number of time steps	10806	10926	10831	10830
avg. nonlinear iterations	2.18	2.00	2.94	2.94
avg. linear iterations	1.63	2.00	2.05	2.10
norm. sim. time [ms]	8.87	26.70	14.73	14.16

Here, $B'_{\boldsymbol{u},p}(\boldsymbol{u}^*, p^*; \boldsymbol{v}, q)[\delta\boldsymbol{u}, \delta p]$ denotes the Gâteaux derivative of the bilinear form $B(.;.)$ at point $(\boldsymbol{u}^*, p^*)$ in direction $(\delta\boldsymbol{u}, \delta p)$ with respect to the first argument. We obtain:

$$\begin{aligned} F'_{\boldsymbol{u},p}(\boldsymbol{u}^*, p^*; \boldsymbol{v}, q)[\delta\boldsymbol{u}, \delta p] = {} & (D'_\tau \delta\boldsymbol{u} + \boldsymbol{u}^*\cdot\nabla\delta\boldsymbol{u} + \delta\boldsymbol{u}\cdot\nabla\boldsymbol{u}^*, \boldsymbol{v}) \\ & -\big(\delta p, \nabla\cdot\boldsymbol{v}\big) + \big(2\nu\boldsymbol{\varepsilon}(\delta\boldsymbol{u}), \boldsymbol{\varepsilon}(\boldsymbol{v})\big) + \big(\nabla\cdot\delta\boldsymbol{u}, q\big), \end{aligned}$$

where

$$D'_\tau \boldsymbol{u} = \frac{2\tau^n + \tau^{n-1}}{\tau^n(\tau^n + \tau^{n-1})}\,\boldsymbol{u}.$$

Correspondingly, the derivative of the GLS stabilization term reads

$$\begin{aligned} S'_{\boldsymbol{u},p}(\boldsymbol{u}^*, p^*; \boldsymbol{v}, q)[\delta\boldsymbol{u}, \delta p] = {} & \\ & \sum_{K\in\mathcal{T}_h} \Big(\delta_1^K \big(D'_\tau\delta\boldsymbol{u} + \boldsymbol{u}^*\cdot\nabla\delta\boldsymbol{u} + \delta\boldsymbol{u}\cdot\nabla\boldsymbol{u}^* + \nabla\delta p\big),\, \boldsymbol{u}^*\cdot\nabla\boldsymbol{v} + \nabla q\Big)_K \\ & + \sum_{K\in\mathcal{T}_h} \Big(\delta_1^K \big(D'_\tau\boldsymbol{u}^* + \boldsymbol{u}^*\cdot\nabla\boldsymbol{u}^* + \nabla p^*\big),\, \delta\boldsymbol{u}\cdot\nabla\boldsymbol{v}\Big)_K + \sum_{K\in\mathcal{T}_h} \big(\delta_2^K \nabla\cdot\delta\boldsymbol{u},\, \nabla\cdot\boldsymbol{v}\big)_K. \end{aligned}$$

The linear subsystem (2.7) is solved with an iterative GMRES method. We precondition the subsystems with a monolithic geometric multigrid using point-Jacobi iterations as smoother accelerated with a relaxation scheme. On the coarsest level a direct solver from the `Amesos` package from the `Trilinos` library is used [8]). For more informations on the linear solver, we refer interested readers to [32]. As stopping criterion, we use an absolute tolerance of 10^{-7} for the nonlinear solver and an relative tolerance of 10^{-2} for the linear solver. To keep the computational efficiency high, we evaluate the linear subsystem (2.7) with a matrix-free approach [22]. Finally, the entire system is set up fully MPI parallelized [1,2].

2.4. Computational Cost Analysis

We briefly analyze various computational costs associated with our choice of discretization. We compare using $\mathbb{Q}_2$ and $\mathbb{Q}_1$ ansatz spaces, as well as using an isoparametric mapping instead of a linear mapping. As a benchmark, we run the "flow past a 2D cylinder with $Re = 100$" (see Section 3.1 configuration with refinement levels $r = 4, 5$ and cfl $= 1, 2$. We specify the number of cells, number of degrees of freedom, total simulation time, number of time steps, average number of non-linear iterations, average number of linear iterations, and simulation time normalized by the total number of linear iterations. All simulations have been run with 12 MPI ranks on a workstation featuring an Intel Core i9-14900.

The results are summarized in Table 1. They indicate that running $\mathbb{Q}_2$ is about 20% faster that running $\mathbb{Q}_1$ with the same number of unknowns,, which is mainly related to the fact that the throughput of matrix-free operator evaluation increases with the polynomial degree [22]. The timing differences

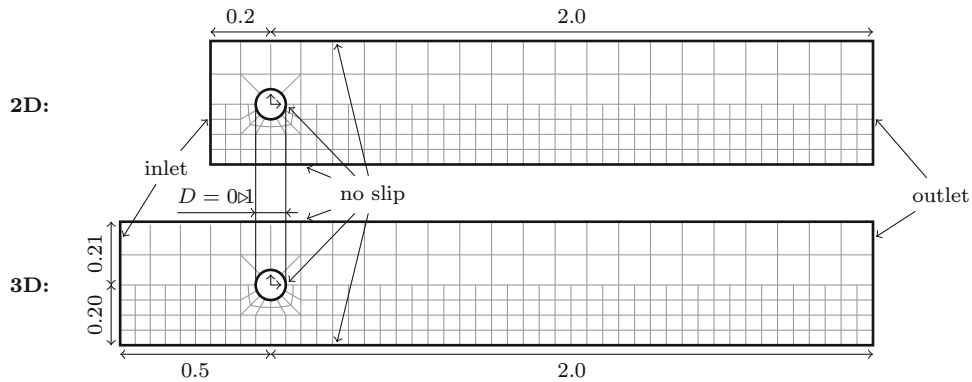


FIG. 3. 2D and 3D computational domains and chosen boundary conditions for the validation tests. A cylindrical cutout is centered at the origin with a diameter of $D = 0.1$. In the background, computational meshes are shown: the coarse mesh is shown on the top half and a single-refined mesh is shown on the bottom half. For 3D, only a 2D xy cut plane is shown; the actual 3D mesh is obtained by extruding the 2D mesh in the third dimension to a depth of with $L = 0.41$ with a total of 4 subdivisions on the coarsest level

between isoparametric and linear mappings are minor with a deviation of less than 1 %). This is due to the implementation of boundary mappings found in `deal.II`, where geometric metrics are precomputed at quadrature-point level so that the same amount of data needs to be loaded from main memory.

3. Numerical Validation and Experiments

We now present a small number of verification and validation tests for our solver that are all based on various established "flow past a cylinder" benchmarks with different Reynolds numbers that have been introduced over the last decades [34]. To this end we consider two geometries, one for 2D and one for 3D simulations. Figure 3 shows the geometries together with the coarsest mesh level and the first refinement level. Unless noted otherwise, we perform all numerical experiments with one of the two geometries and (coarse) meshes shown in Figure 3. Finer meshes are obtained from the two coarse meshes by globally refining the mesh with an appropriate curvature approximation; see Section 2.2. We choose to use the GLS stabilization term $S_{\mathrm{GLS}}(\boldsymbol{u}_h^n, p_h^n; \boldsymbol{v}, q)$ given by (2.4) throughout.

Quantities of interest. Let Γ_{cyl} denote the boundary of the cylindrical cutout for the 2D or 3D domain; see Figure 3. In analogy to [34] we introduce drag and lift coefficients as follows:

$$c_{\mathrm{d}} = \frac{2}{A\bar{u}^2} F_x, \quad c_{\mathrm{l}} = \frac{2}{A\bar{u}^2} F_y, \quad \text{where } \boldsymbol{F} = \int_{\Gamma_{\mathrm{cyl}}} \big(-p\boldsymbol{I} + 2\nu\boldsymbol{\varepsilon}(\boldsymbol{u}) \big) \cdot \boldsymbol{n} \, \mathrm{d}S. \tag{3.1}$$

Here, A denotes the cross section of the cylinder, which is $A = D = 0.1$ in 2D, and $A = DL = 0.1 \cdot 0.41$ in 3D. The quantity $\bar{u}$ denotes the mean inflow velocity. As a last quantity of interest we introduce the pressure difference Δp between the front and back of the cylinder [34].

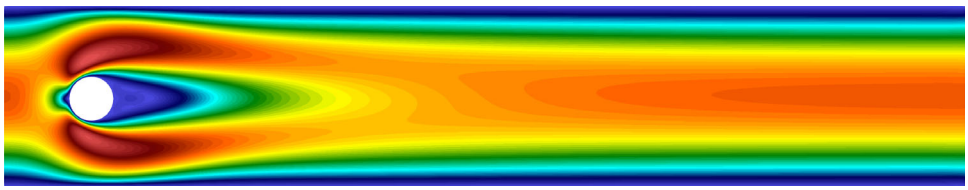
3.1. Validation: Flow Past a 2D Cylinder with $Re = 20,\ 100$

We start by computing a well-known benchmark configuration introduced in [33]. The geometry and boundary conditions are the same as the 2D mesh depicted in Figure 3. We use no slip boundary conditions on cylinder and walls with a parabolic inflow profile,

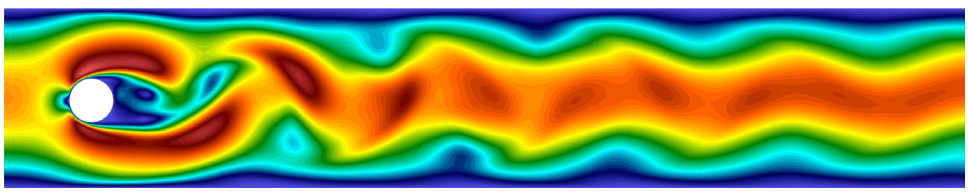
$$\boldsymbol{u}_{\mathrm{D}}(\boldsymbol{x}) = \big(4\, u_{\max}\, x_2 (H - x_2)/H^2,\ 0 \big),$$

TABLE 2. *Validation: flow past a 2D cylinder at $Re = 100$ with no-slip boundary conditions. Minimum, maximum, temporal average and standard deviation of drag and lift coefficients, c_d, c_l, and pressure difference Δp are computed on the time interval $[5, 20]$ for refinement level $r = 7$*

	Minimum	Maximum	Average μ	Deviation σ
c_d	3.1522	3.2155	3.1840	0.0220
c_l	-1.0173	0.9841	-0.0187	0.7055
Δp	2.4201	2.5069	2.4637	0.0299



(a) $Re = 20$



(b) $Re = 100$

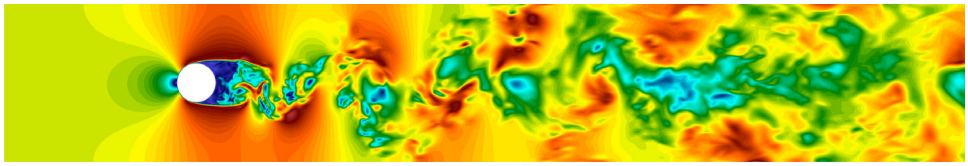
FIG. 4. Validation: flow past a 2D cylinder at $Re = 20$ and 100 with no-slip boundary conditions. Temporal snapshot of the velocity magnitudes at time $T = 20$ are shown for both steady and unsteady simulations with $Re = 20$, and $Re = 100$, respectively. A no-slip boundary condition is applied on both the cylinder surface and the channel walls. The shown snapshots are for a simulation using $\mathbb{Q}_1$ finite elements on a coarse grid with refinement level $r = 4$, totaling $69k$ degrees of freedom

where $H = 0.41$. We set the kinematic viscosity to $\nu = 0.001$ and $u_{\max} = 0.3$, and $u_{\max} = 1.5$, respectively. This translates to Reynolds numbers $Re = 20$ with $\bar{u} = 0.2$, and $Re = 100$ with $\bar{u} = 1.0$, respectively. The simulation is run from time $t = 0$ to $t = 20$ with isoparametric, continuous $\mathbb{Q}_1$ finite elements; see Section 2.2. We perform a convergence study on a sequence of increasingly globally refined grids with refinement levels $r = 4$ to $r = 7$ amounting to a total of $69k$ to $4.34M$ degrees of freedom. We set the stabilization parameters to $c_1 = 2$, and $c_2 = 1$.

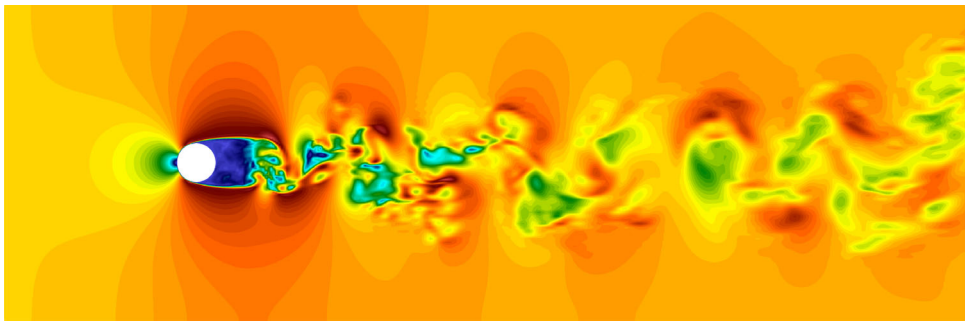
Temporal snapshots of velocity magnitudes for time $T = 20$ for both Reynolds numbers are shown in Figure 4. For the $Re = 20$ case we observe a steady flow with $c_d = 5.57819(63)$, $c_l = 0.010603(24)$, and $\Delta p = 0.1175220(10)$. The values and uncertainty intervals have been obtained by extrapolating the values for the four refinement levels to the limit. The results are in good agreement with [33, Table 3] and [28]. The corresponding results for the unsteady case with $Re = 100$ are summarized in Table 2 for refinement level $r = 7$. Here, for a quantity of interest q we introduce a temporal average μ_q and standard deviation σ_q:

$$\mu_q := \frac{1}{t_2 - t_1} \int_{t_1}^{t_2} q(t)\,\mathrm{d}t, \qquad \sigma_q^2 := \frac{1}{t_2 - t_1} \int_{t_1}^{t_2} \big(q(t) - \mu_q\big)^2\,\mathrm{d}t. \tag{3.2}$$

We again observe that our results are in good agreement with the values reported in [33, Table 4].



(a) $Re = 3900$, narrow domain



(b) $Re = 3900$, wide domain

FIG. 5. Validation: flow past a 3D cylinder at $Re = 3900$ with no-slip boundary conditions. 2D cutout (with $z = 0$) of a temporal snapshot of the velocity magnitudes at time $T = 0.5$ are shown. A no-slip boundary condition is applied on the cylinder surface and slip boundary conditions are applied on the channel walls. The simulation is performed using $\mathbb{Q}_2$ finite elements with a total of $53M$ degrees of freedom

3.2. Validation: Flow Past a 3D Cylinder with $Re = 3900$

Next we perform a 3D validation test with the 3D geometry outlined in Figure 3, again with no slip conditions on the cylinder, but with slip boundary conditions at the walls of the channel [9]. Furthermore, the cylinder is now inscribed symmetrically with equal distance to the top and bottom wall. The kinematic viscosity is set to $\nu = 0.001$ and the input profile is chosen to be a constant inflow at $\bar{u} = u_{\max} = 39$. This translates to a Reynolds number of $Re = 3900$. The simulation is run from time $t = 0$ until we hit an asymptotic statistical regime at $t = 0.5$. Zero initial data is used for the flow combined with a linear ramp-up of the inflow from 0 to $u_{\max}$ for $t = 0$ to $t = 0.02$. We choose an isoparametric, continuous $\mathbb{Q}_2$ finite elements on a grid with refinement level $r = 4$ amounting to a total of $53M$ degrees of freedom; see Section 2.2. In addition we run the simulation on a second modified grid where the distances to the upper, lower and side walls has been enlarged by a factor two. We again set the stabilization parameters to $c_1 = 2$, and $c_2 = 1$.

A temporal snapshot of velocity magnitudes at time $t = 0.5$ is shown in Figure 5. The drag coefficient c_d has been computed by averaging from on the time interval $[0.2, 0.5]$:

$$\begin{aligned} \text{Narrow domain:} \quad & \mu_{c_d} = 1.3421, \qquad \sigma_{c_d} = 0.0177, \\ \text{Wide domain:} \quad & \mu_{c_d} = 1.0558, \qquad \sigma_{c_d} = 0.0106. \end{aligned}$$

The drag value we obtained for the larger domain is in good agreement with those reported from computational and experimental results [9]. The drag coefficient for the smaller domain is slightly larger due to a higher influence of wall effects.

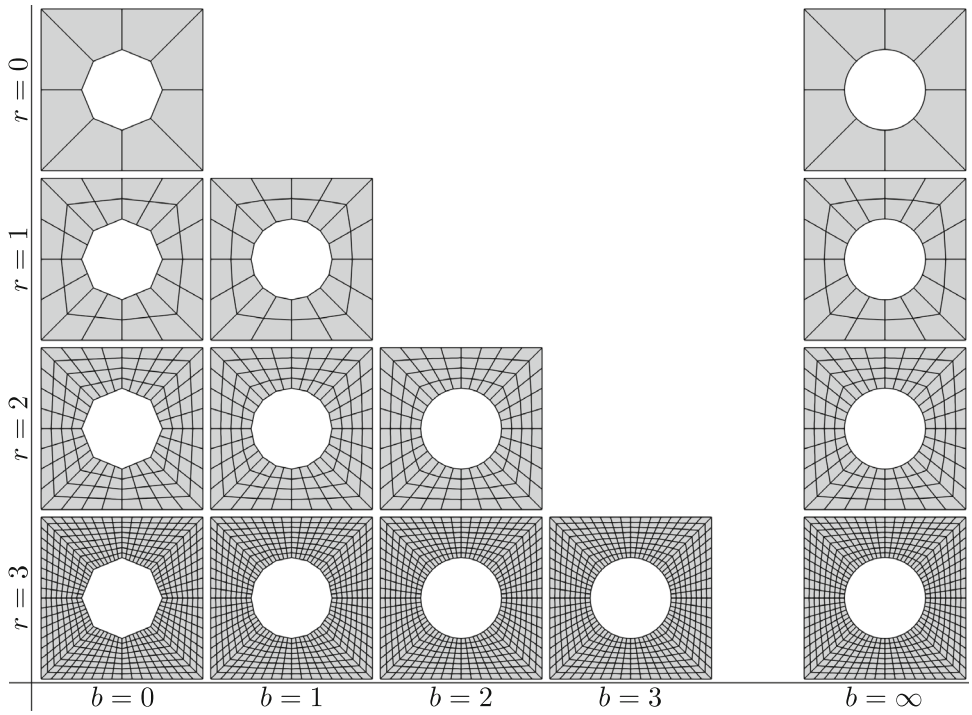
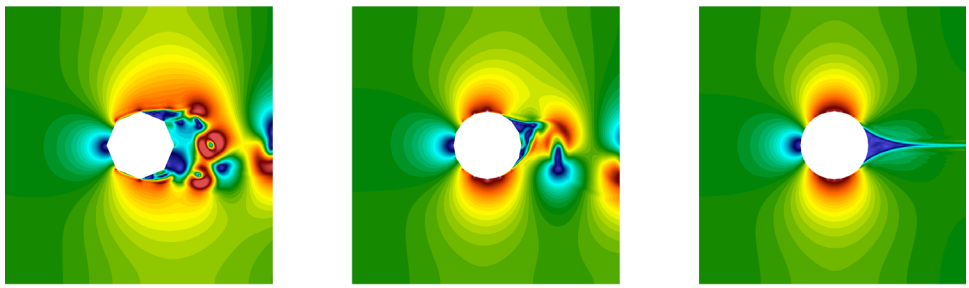


FIG. 6. Visualization of the *numerical surface roughness* for meshes introduced into our computations by refining the coarse mesh r times. The high-order manifold at the cylinder is removed after b refinement steps, resulting in a linear $\mathbb{Q}_1$ mapping. For case $b = \infty$ the high-order manifold has not been removed and an isoparametric $\mathbb{Q}_2$ mapping is used

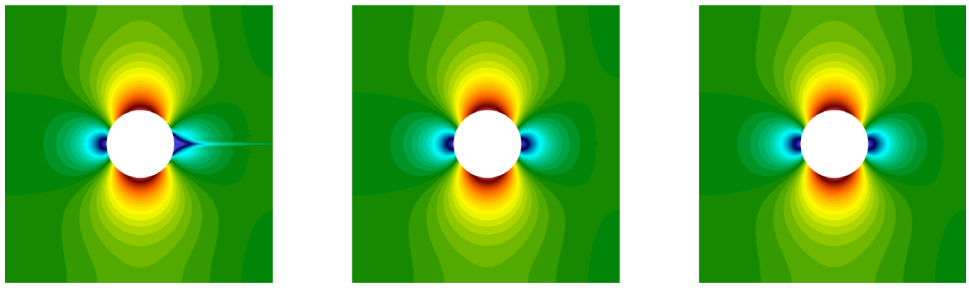
3.3. Parameter Study: Surface Roughness and Flow Past a 2D Cylinder

We now modify the previous flow configuration as follows. First, as suggested in [12, Sec. 9] we set $\nu = 0$ and set *slip boundary conditions* also on the cylinder. We keep the uniform inflow profile with $\bar{u} = u_{\max} = 39$, as well as zero initial conditions and the linear ramp-up of inflow conditions from $t = 0$ to $t = 0.05$. We again use continuous $\mathbb{Q}_2$ finite elements for the 2D mesh at refinement level $r = 4$ ($273k$ total degrees of freedom). In addition to the isoparametric approximation ($b = \infty$) various degrees of surface roughness are simulated ($b = 0, \ldots, 4$). This is achieved by first refining the mesh with a high-order manifold description on the cylinder b times, then removing the manifold description for the remaining $r - b$ refinements, resulting in a linear $\mathbb{Q}_1$ mapping; see Section 2.2. The process is visualized in Figure 6. As a convenient metric for the surface roughness, we also report the maximal relative radial deviation d_{r} of the discretized boundary from the true circle. The deviation starts at 14.6 % for $b = 0$, and halfs with every boundary approximation step.

A temporal snapshot showing velocity magnitudes in a zoom around the cylinder for time $t = 0.5$ are reported in Figure 7. For the simulations we varied the degree of surface roughness from $b = 0$ to $b = 4$ and show an isoparametric approximation with $b = \infty$. We perform statistics for drag and lift coefficients and the pressure drop on the time interval $[0.2, 0.5]$. We have verified that all simulations are past the transient startup and have reached a fully developed flow regime at $t = 0.2$. The results are reported in Table 3. Judging from Figure 7 it is evident that the flow becomes stable for decreasing surface roughness while rapidly approaching a potential flow profile. This is corroborated by the statistics summarized in Table 3, where, depending on the level of surface roughness ($b = 0, \ldots 4, \infty$), the resulting drag and lift coefficients vary by 3 to 5 orders of magnitude.

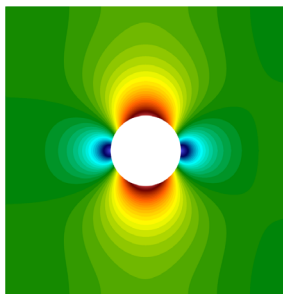


(a) $b = 0$ ($d_r = 14.6\,\%$) (b) $b = 1$ ($d_r = 7.32\,\%$) (c) $b = 2$ ($d_r = 3.66\,\%$)



(d) $b = 3$ ($d_r = 1.83\,\%$) (e) $b = 4$ ($d_r = 0.915\,\%$) (f) $b = \infty$ ($d_r \approx 0\,\%$)

FIG. 7. Parameter study with flow past 2D cylinder with continuous $\mathbb{Q}_2$ finite elements on the 2D mesh configuration with refinement level $r = 4$ amounting to a total number of degrees of freedom of $273k$. The high-order $\mathbb{Q}_2$ manifold on the cylinder is removed after $b = 0$ to 4 refinement steps resulting in various different degrees of surface roughness. The magnitude of the velocity is shown on a rainbow scale in a zoom around the cylinder



(a) $r = 5$, $b = \infty$ ($d_r \approx 0\,\%$)

FIG. 8. Parameter study with flow past 2D cylinder with continuous $\mathbb{Q}_2$ finite enements on the 2D mesh configuration with refinement level $r = 5$ amounting to a total number of degrees of freedom of $1.092M$. isoparametric $\mathbb{Q}_2$ approximation ($b = \infty$) of the boundary. The magnitude of the velocity is shown on a rainbow scale in a zoom around the cylinder

The snapshot of the solution at the final time $t = 0.5$ an isoparametric discretization with a refinement level of $r = 5$ ($1.092M$ total degrees of freedom) are shown in Figure 8. For this resolution, we observe that geometric errors from the cylinder boundary are almost entirely eliminated. As a result, no boundary error propagates downstream, and the computed drag force is nearly zero.

TABLE 3. *2D parameter study: quantities of interest as a function of surface roughness b for refinement level $r = 4$. Temporal average and standard deviation of drag and lift coefficients, c_d, c_l, and pressure difference Δp are computed on the time interval* $[0.2, 0.5]$

	μ_{c_d}	(σ_{c_d})	μ_{c_l}	(σ_{c_l})
$b = 0$	1.947×10^{0}	(3.866×10^{-1})	3.672×10^{-3}	(9.443×10^{-1})
$b = 1$	6.316×10^{-1}	(1.625×10^{-1})	1.718×10^{-2}	(6.177×10^{-1})
$b = 2$	1.752×10^{-2}	(4.041×10^{-3})	-1.884×10^{-3}	(4.131×10^{-2})
$b = 3$	9.875×10^{-3}	(3.825×10^{-07})	5.476×10^{-11}	(7.624×10^{-11})
$b = 4$	7.589×10^{-3}	(2.668×10^{-16})	2.864×10^{-8}	(2.667×10^{-22})
$b = \infty$	7.622×10^{-3}	(2.374×10^{-16})	2.656×10^{-8}	(6.762×10^{-22})
	$\mu_{\Delta p}$	$(\sigma_{\Delta p})$		d_{r}
$b = 0$	5.143×10^{3}	(2.361×10^{3})	$b = 0$	14.6 %
$b = 1$	2.438×10^{3}	(9.104×10^{2})	$b = 1$	7.32 %
$b = 2$	5.168×10^{2}	(2.168×10^{1})	$b = 2$	3.66 %
$b = 3$	2.046×10^{2}	(2.026×10^{-3})	$b = 3$	1.83 %
$b = 4$	7.352×10^{0}	(7.352×10^{-15})	$b = 4$	0.915 %
$b = \infty$	7.455×10^{0}	(1.074×10^{-13})	$b = \infty$	≈ 0 %

3.4. Parameter Study: Surface Roughness and Flow Past a 3D Cylinder

In light of the stability analysis conducted in [12] and [11, Chap. 12], that establish that the potential flow solution is unstable at separation points in three spatial dimensions, resulting to strong vorticity generations in the streamwise direction, the main question is of course whether the numerical results presented in Section 3.3 carry over to three space dimensions. To this end we repeat the parameter study presented previously but for three spatial dimensions. Except for using the 3D mesh depicted in Figure 3 we keep all other parameters unchanged. We choose a refinement level of $r = 4$ with $\mathbb{Q}_2$ finite elements amounting to a total of $53M$ degrees of freedom. We set the stabilization parameters to $c_1 = 1$, and $c_2 = 1$.

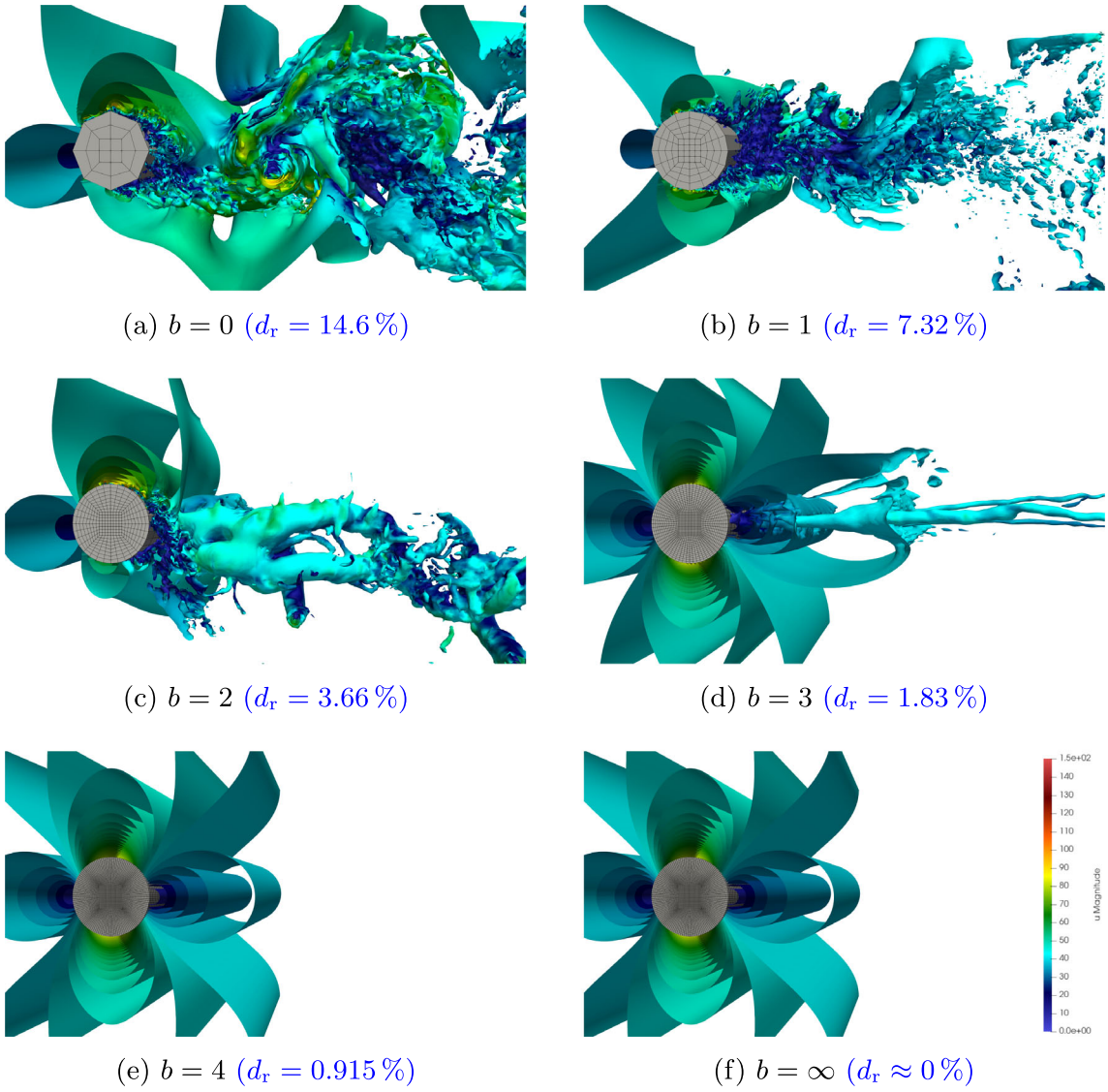
Note that the mesh is symmetric along the x_2 and x_3 directions and is uniformly structured. As a result, the GLS stabilization term is distributed symmetrically and uniformly throughout the domain, avoiding the introduction of additional numerical artifacts.

Temporal snapshots for time $t = 0.5$ depicting pressure iso-surfaces around the cylinder are shown in Figure 9. For the simulations we varied the degree of surface roughness from $b = 0$ to $b = 4$ and also report an isoparametric approximation with $b = \infty$. In Figure 9 we visualize the surface roughness with a gray cylindrical inset showing the corresponding coarse grid that defines the obstacle. We note that all simulations are run at the same global refinement level of $r = 4$.

Similarly to the 2D parameter study, we perform statistics for drag and lift coefficients and the pressure drop on the time interval $[0.2, 0.5]$. We have verified that all simulations are past the transient startup and have reached a fully developed flow regime at $t = 0.2$. The results are reported in Table 4. We make the observation that the flow behaves qualitatively and quantitatively similar to the 2D parameter study. It is evident that the flow profile becomes stable for decreasing surface roughness while rapidly approaching a potential flow profile. This is again confirmed by the statistics summarized in Table 4, where, depending on the level of surface roughness ($b = 0$ to 4, ∞), the resulting drag and lift coefficients vary by 3 to 5 orders of magnitude.

Comparison with Hoffman and Johnson [11] We conclude the parameter study with a short comparison of our computational results to [11] that originally introduced this benchmark. While studying the benchmark configuration, the authors of [11] reported the following observations [11, p. 88]:

(i) Only one separation point is on the downstream of the cylinder;
(ii) No boundary layer is prior to the separation point;
(iii) Strong generation of streamwise vortices at the separation point.



(a) $b = 0$ ($d_r = 14.6\,\%$) (b) $b = 1$ ($d_r = 7.32\,\%$)

(c) $b = 2$ ($d_r = 3.66\,\%$) (d) $b = 3$ ($d_r = 1.83\,\%$)

(e) $b = 4$ ($d_r = 0.915\,\%$) (f) $b = \infty$ ($d_r \approx 0\,\%$)

FIG. 9. Parameter study with flow past 3D cylinder with continuous $\mathbb{Q}_2$ finite elements on the 3D mesh configuration with refinement level is $r = 4$ amounting to a total number of degrees of freedom of $53M$. The high-order $\mathbb{Q}_2$ manifold on the cylinder is removed after $b = 0$ to 4 refinement steps resulting in various different degrees of surface roughness. This is visualized with a gray cylindrical inset showing the corresponding coarse grid; see the discussion in Section 3.3 and Figure 6. The snapshots show 30 equidistant pressure iso-surfaces between minimal and maximal pressure and are colorized with the velocity magnitude on a rainbow color scale between $[0, 150]$

Based on our numerical simulations, we confirm that the first two observations also hold in the isoparametric case with $b = \infty$. However, strong streamwise vorticity is observed only when the geometry error is sufficiently large. For example, in Figure 9(d), where the surface roughness is set to $b = 3$, significant streamwise vorticity is present. As the surface roughness decreases, for instance, for $b = 4$ and $b = \infty$ (cf. Figures 9(e) and (f)), no streamwise vortices or streamwise instabilities are observed.

Finally, we remark in passing that—while not reported in this manuscript—we observe the same qualitative and quantitative behavior of the flow for differing refinement levels ($r = 3$, 5, etc.), i.e., for a given refinement level r we are reliable able to tune the resulting flow characteristics (drag, lift, pressure

TABLE 4. *3D parameter study: quantities of interest as a function of surface roughness b for refinement level $r = 4$. Temporal average and standard deviation of drag and lift coefficients, c_d, c_l, and pressure difference Δp are computed on the time interval* $[0.2, 0.5]$

	μ_{c_d}	(σ_{c_d})	μ_{c_l}	(σ_{c_l})
$b = 0$	1.444×10^{0}	(5.693×10^{-2})	1.272×10^{-2}	(5.337×10^{-1})
$b = 1$	4.211×10^{-1}	(2.563×10^{-2})	-3.108×10^{-3}	(1.111×10^{-1})
$b = 2$	1.445×10^{-1}	(6.159×10^{-2})	-6.938×10^{-2}	(2.022×10^{-1})
$b = 3$	1.343×10^{-2}	(9.575×10^{-3})	-1.860×10^{-2}	(2.769×10^{-2})
$b = 4$	-1.168×10^{-4}	(1.516×10^{-17})	6.533×10^{-10}	(5.367×10^{-23})
$b = \infty$	-9.960×10^{-5}	(1.057×10^{-17})	-2.669×10^{-10}	(2.182×10^{-23})
	$\mu_{\Delta p}$	$(\sigma_{\Delta p})$		d_{r}
$b = 0$	2.617×10^{3}	(4.877×10^{2})	$b = 0$	14.6 %
$b = 1$	1.166×10^{3}	(6.484×10^{1})	$b = 1$	7.32 %
$b = 2$	4.364×10^{2}	(3.009×10^{1})	$b = 2$	3.66 %
$b = 3$	1.690×10^{2}	(7.266×10^{0})	$b = 3$	1.83 %
$b = 4$	5.177×10^{0}	(4.001×10^{-13})	$b = 4$	0.915 %
$b = \infty$	5.246×10^{0}	(6.500×10^{-13})	$b = \infty$	≈ 0 %

TABLE 5. *3D parameter study: temporal average of the drag coefficient, $\mu_{\Delta p}$, and standard deviation $\sigma_{\Delta p}$ as a function of refinement level $r = 2, 3, 4$ and percentage of applied random mesh distortion of interior vertices. Temporal average and standard deviation are computed on the time interval* $[0.5, 1.5]$. *For comparison, we repeat the values obtained for the case of no mesh distortion in the last column. Here, statistics are computed over the shorter time interval* $[0.2, 0.5]$

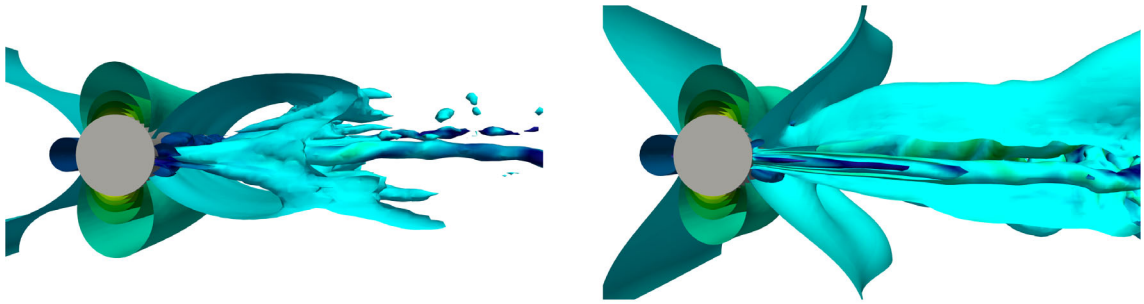
	10 % pert.	5 % pert.	2.5 % pert.	0 % pert.
r=2	5.997×10^{-1}	5.9123×10^{-1}	5.733×10^{-1}	-2.870×10^{-3}
	(1.624×10^{-1})	(1.464×10^{-1})	(1.511×10^{-1})	(3.067×10^{-18})
r=3	2.696×10^{-1}	4.474×10^{-1}	2.807×10^{-1}	-6.844×10^{-4}
	(7.107×10^{-2})	(1.671×10^{-1})	(6.313×10^{-2})	(7.590×10^{-18})
r=4	1.301×10^{-1}	7.078×10^{-2}	1.105×10^{-1}	-9.960×10^{-5}
	(4.668×10^{-2})	(1.516×10^{-2})	(1.291×10^{-2})	(1.057×10^{-17})

drop) 3 to 5 orders of magnitude by choosing a different boundary roughness b. Similary, we have also verified the behavior of using the inconsistent stabilization term $S_{\mathrm{G2}}(\boldsymbol{u}_h^n, p_h^n; \boldsymbol{v}, q)$ given by (2.5) instead of the consistent GLS stabilization term given by (2.4). The inconsistent stabilization gives very similar results which we omit to present.

3.5. Parameter Study: Mesh Distortion and Flow Past a 3D Cylinder

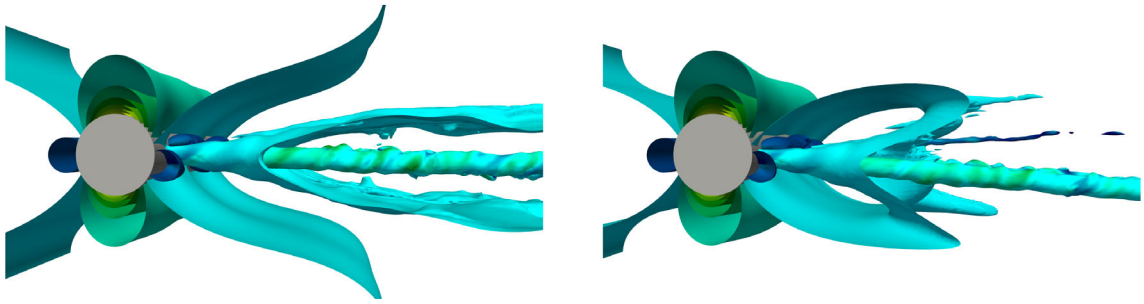
We now turn to the final numerical experiment of this study. The goal of this test is to investigate whether and how random mesh distortion influences the flow dynamics of incompressible fluid past a cylinder in three spatial dimensions. Starting from the 3D mesh with refinement levels $r = 2$, 3, and 4, we use again a $\mathbb{Q}_2$ isoparametric finite element approximation, resulting in approximately $0.8M$, $6.6M$, $53M$ degrees of freedom, respectively. We again set the stabilization parameters to $c_1 = 1$, and $c_2 = 1$.

To introduce distortion, every interior vertex of the mesh is perturbed randomly by up to $p = 10\%$, 5%, and 2.5% of the minimal distance to the nearest neighboring vertex. This produces an unstructured mesh in which the elements exhibit varying volumes. Consequently, the resulting GLS stabilization terms become nonuniform, breaking the mirror symmetry of the mesh. We furthermore reduce the influence of boundary effects on the channel walls by prescribing periodic boundary conditions on both side walls and the top and bottom of the channel. We run all simulations on the larger time interval $[0.0, 1.5]$. We have verified that all simulations are past the transient startup and have reached a fully developed flow regime at $t = 0.5$. We take statistical averages on the time interval $[0.5, 1.5]$. The results are reported



(a) refinement $r = 3$, 10 % perturbation

(b) refinement $r = 3$, 5 % perturbation



(c) refinement $r = 4$, 10 % perturbation

(d) refinement $r = 4$, 5 % perturbation

FIG. 10. Parameter study with flow past 3D cylinder with continuous $\mathbb{Q}_2$ isoparametric finite elements on the 3D mesh configuration with refinement levels $r = 3$, 4 at time $t = 1.5$, amounting to a total number of degrees of freedom of $6.3M$, and $53M$ respectively. A mesh distortion of 10% and 5% is introduced, respectively. The snapshots show 30 equidistant pressure iso-surfaces between minimal and maximal pressure and are colorized with the velocity magnitude on a rainbow color scale between $[0, 150]$

in Table 5. It is striking that even a miniscule amount of *nonuniform* mesh distortion is sufficient to fully trigger the three dimensional instability leading to streamwise vorticity. In particular there exists a bifurcation point between 2.5% mesh distortion and the fully symmetric mesh with 0% distortion at which the flow becomes potential; cf. Tables 4, and 5. However, we again observe a strong dependence of the quantitative values on the level of mesh refinement controlled by the refinement parameter r.

For completeness, a series of temporal snapshots of the simulations are shown in Figure 10, where, consistent with previous visualizations, 30 equidistant pressure iso-surfaces are plotted between the minimum and maximum pressure values. We verify that the simulations do indeed reproduce the third observation from [11, p. 88], namely, the strong generation of streamwise vorticity. Our numerical findings thus support the conclusion that the addition of nonuniform discretization errors through mesh distortion destabilizes an otherwise potential-like flow, leading to significant drag production [11, p. 88]. We note however, that we are able again to tune quantitative results by about an order of magnitude by controlling the refinement level r.

4. Discussion and Conclusion

The objective of this work was to investigate the influence of numerical discretization errors on incompressible flow past a cylinder in two and three spatial dimensions when equipped with the slip boundary condition. The two main sources of discretization errors we examined were: (a) the influence of the geometry approximation error at the curved boundary; and (b) the influence of nonuniform artificial viscosity.

The finite element discretization of the Navier–Stokes equations was stabilized using a Galerkin–Least-Squares (GLS) formulation.

To first minimize the influence of such discretization errors we employed carefully constructed quasi-uniform quadrilateral and hexahedral meshes, combined with a high-order isoparametric mapping to accurately represent the cylinder boundary. Through a series of systematic numerical validations in both two and three dimensions, we found that using a slip boundary condition leads to a *stable potential flow profile*, provided that numerical surface roughness and mesh distortion are minimized. However, the introduction of numerical discretization errors—whether through inaccurate geometry representation or mesh distortion—destabilizes the flow, resulting in nonzero drag and lift coefficients. Most importantly, we have been able to demonstrate that numerical surface roughness and mesh distortion can be used as a control parameter in this regard: By carefully choosing the degree of boundary approximation error (or mesh distortion) one can manipulate the drag and lift forces over a large interval of numerical values spanning 3 to 5 orders of magnitude.

As a second numerical test we analyzed numerically the stability of the potential flow simulation under random, nonuniform mesh distortion while minimizing the influence of discretization errors on the boundary (by using an isoparametric mapping and periodic wall boundary conditions for the channel). It is striking that even a miniscule amount of nonuniform mesh distortion is sufficient to fully trigger the three dimensional instability leading to streamwise vorticity. We observe in particular that there exists a bifurcation point between 2.5% mesh distortion and the fully symmetric mesh with 0% distortion at which the flow becomes potential. However, we again note that we observe a strong dependence of the quantitative values on the level of mesh refinement.

Based on these findings, we conclude that slip boundary conditions— at least in the form suggested by [11] with an under resolved, coarse mesh and using GLS stabilization— are not suitable for *wall-modeled* large-eddy simulation of realistic fluid flows at low viscosity. Slip boundary conditions fail to accurately predict key aerodynamic quantities such as drag and lift, and do not correctly capture flow separation behavior.

Acknowledgements. This work has been partially supported by the National Science Foundation under grant DMS-2045636 (MM), by the Air Force Office of Scientific Research, USAF, under grant/contract number FA9550-23-1-0007 (MM), and Swedish Research Council under grant number 2021-04620 (PM & MN). The authors acknowledge fruitful discussions and collaborations with Bruno Blais and Laura Prieto Saavedra as well as the larger `deal.II` community.

Funding Open access funding provided by Uppsala University.

Conflict of interest The authors declare that they have no conflict of interest.

Open Access. This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

References

[1] Africa, P.C., Arndt, D., Bangerth, W., Blais, B., Fehling, M., Gassmöller, R., Heister, T., Heltai, L., Kinnewig, S., Kronbichler, M., Maier, M., Munch, P., Schreter-Fleischhacker, M., Thiele, J.P., Turcksin, B., Wells, D., Yushutin, V.: The deal.II Library, Version 9.6. J. Numer. Math. **32**(4), 369–380 (2024). https://doi.org/10.1515/jnma-2024-0137

[2] Arndt, D., Bangerth, W., Davydov, D., Heister, T., Heltai, L., Kronbichler, M., Maier, M., Pelteret, J.-P., Turcksin, B., Wells, D.: The deal.II finite element library: design, features, and insights. Computers & Mathematics with Applications **81**(1), 407–422 (2021). https://doi.org/10.1016/j.camwa.2020.02.022

[3] Bose, S.T., Moin, P.: A dynamic slip boundary condition for wall-modeled large-eddy simulation. Phys. Fluids **26**(1), 015104 (2014). https://doi.org/10.1063/1.4849535

[4] Brooks, A.N., Hughes, T.J.R.: Streamline upwind/Petrov-Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equations. Comput. Methods Appl. Mech. Engrg. **32**(1–3), 199–259 (1982). https://doi.org/10.1016/0045-7825(82)90071-8. (**FENOMECH ''81, Part I (Stuttgart, 1981)**)

[5] Constantin, P., Kukavica, I., Vicol, V.: On the inviscid limit of the Navier-Stokes equations. Proc. Amer. Math. Soc. **143**(7), 3075–3090 (2015). https://doi.org/10.1090/S0002-9939-2015-12638-X

[6] d'Alembert, J.L.R.: Essai d'une nouvelle théorie de la résistance des fluides. David l'aîné, (1752)

[7] Heltai, L., Bangerth, W., Kronbichler, M., Mola, A.: Propagating geometry information to finite element computations. ACM Transactions on Mathematical Software (TOMS) **47**(4), 1–30 (2021)

[8] Heroux, M., Bartlett, R., Hoekstra, V.H.R., Hu, J., Kolda, T., Lehoucq, R., Long, K., Pawlowski, R., Phipps, E., Salinger, A. et al.: An overview of Trilinos. Citeseer (2003)

[9] Hoffman, J.: Efficient computation of mean drag for the subcritical flow past a circular cylinder using general galerkin g2. Int. J. Numer. Meth. Fluids **59**(11), 1241–1258 (2009). https://doi.org/10.1002/fld.1865

[10] Hoffman, J., Jansson, N.: A computational study of turbulent flow separation for a circular cylinder using skin friction boundary conditions. In Salvetti, M.V., Geurts, B., Meyers, J., Sagaut, P. editors, Quality and Reliability of Large-Eddy Simulations II, pages 57–68, Dordrecht, (2011). Springer Netherlands

[11] Hoffman, J., Johnson, C.: Computational turbulent incompressible flow, volume 4 of Applied Mathematics: Body and Soul. Springer, Berlin, (2007). https://doi.org/10.1007/978-3-540-46533-1

[12] Hoffman, J., Johnson, C.: Resolution of d'Alembert's paradox. J. Math. Fluid Mech. **12**(3), 321–334 (2010). https://doi.org/10.1007/s00021-008-0290-1

[13] Hoffman, J., Jansson, J., Degirmenci, C., Jansson, N., Nazarov, M.: Unicorn: a unified continuum mechanics solver, chapter 18. Automated solution of differential equations by the finite element method (Eds. A.Logg, K-A.Mardal, G.N.Wells), Springer, (2012)

[14] Hoffman, J., Jansson, J., Vilela de Abreu, R., Degirmenci, N.C., Jansson, N., Müller, K., Nazarov, M., Spühler, J.H.: Unicorn: parallel adaptive finite element simulation of turbulent flow and fluid-structure interaction for deforming domains and complex geometry. Comput. & Fluids **80**, 310–319 (2013). https://doi.org/10.1016/j.compfluid.2012.02.003

[15] Hoffman, J., Jansson, J., Jansson, N., De Abreu, R.V.: Towards a parameter-free method for high Reynolds number turbulent flow simulation based on adaptive finite element approximation. Comput. Methods Appl. Mech. Engrg. **288**, 60–74 (2015). https://doi.org/10.1016/j.cma.2014.12.004

[16] Hoffman, J., Jansson, J., Johnson, C.: New theory of flight. J. Math. Fluid Mech. **18**(2), 219–241 (2016). https://doi.org/10.1007/s00021-015-0220-y

[17] Iftimie, D.S., Planas, G.: Inviscid limits for the Navier-Stokes equations with Navier friction boundary conditions. Nonlinearity **19**(4), 899–918 (2006). https://doi.org/10.1088/0951-7715/19/4/007

[18] Jansson, J., Wingstedt, K.: Adaptive Euler – First Principles Fast Prediction of High-Lift Aerodynamics. https://doi.org/10.2514/6.2025-1081

[19] Jansson, J., Johnson, C., Scott, R.: Predictive Euler CFD - Resolution of NASA Vision (2030). https://doi.org/10.2514/6.2022-3589

[20] Jansson, N., Hoffman, J., Nazarov, M.: Adaptive simulation of turbulent flow past a full car model. In State of the Practice Reports, SC '11, pages 20:1–20:8, New York, NY, USA, (2011). ACM. https://doi.org/10.1145/2063348.2063375

[21] Johnson, C., Saranen, J.: Streamline diffusion methods for the incompressible Euler and Navier-Stokes equations. Math. Comp. **47**(175), 1–18 (1986). https://doi.org/10.2307/2008079

[22] Kronbichler, M., Kormann, K.: A generic interface for parallel cell-based finite element operator application. Computers & Fluids **63**, 135–147 (2012)

[23] Kukavica, I., Vicol, V., Wang, F.: Remarks on the inviscid limit problem for the Navier-Stokes equations. Pure Appl. Funct. Anal. **7**(1), 283–306 (2022)

[24] Logg, A., Mardal, K.-A., Wells, G.N.: Automated Solution of Differential Equations by the Finite Element Method. Springer-Verlag, (2012)

[25] Maier, M., Munch, P., Nazarov, M.: dealii-ns-gls: an experimental, stabilized incompressible navier-stokes solver using deal.II, May (2025). https://doi.org/10.5281/zenodo.15455164

[26] Masmoudi, N.: Remarks about the inviscid limit of the Navier-Stokes system. Comm. Math. Phys. **270**(3), 777–788 (2007). https://doi.org/10.1007/s00220-006-0171-5

[27] Min, T., Kim, J.: Effects of hydrophobic surface on skin-friction drag. Phys. Fluids **16**(7), L55–L58 (2004). https://doi.org/10.1063/1.1755723

[28] Nabh, G.: On high order methods for the stationary incompressible Navier-Stokes equations. Interdisziplinäres Zentrum für Wiss. Rechnen der Univ, Heidelberg (1998)

[29] Navier, C.: Mémoire sur les lois du mouvement des fluides. Mémoires de l'Académie Royale des Sciences de l'Institut de France **6**(1823), 389–440 (1823)

[30] Nazarov, M., Hoffman, J.: On the stability of the dual problem for high Reynolds number flow past a circular cylinder in two dimensions. SIAM J. Sci. Comput. **34**(4), A1905–A1924 (2012). https://doi.org/10.1137/110836213

[31] Prandtl, L.: On the motion of fluids with very little viscosity. Proceedings of the Third International Congress of Mathematicians, (1904)
[32] Prieto Saavedra, L., Munch, P., Blais, B.: A matrix-free stabilized solver for the incompressible Navier-Stokes equations, (2024)
[33] Schäfer, M., Turek, S., Durst, F., Krause, E., Rannacher, R.: Benchmark Computations of Laminar Flow Around a Cylinder, pages 547–566. Vieweg+Teubner Verlag, Wiesbaden, (1996a.) https://doi.org/10.1007/978-3-322-89849-4_39
[34] Schäfer, M., Turek, S., Durst, F., Krause, E., Rannacher, R.: Benchmark Computations of Laminar Flow Around a Cylinder, pages 547–566. Vieweg+Teubner Verlag, Wiesbaden, (1996b). https://doi.org/10.1007/978-3-322-89849-4_39
[35] Schlichting, H., Gersten, K.: Boundary-layer theory. Springer-Verlag, Berlin, ninth edition, (2017). https://doi.org/10.1007/978-3-662-52919-5. With contributions from Egon Krause and Herbert Oertel Jr., Translated from the German by Katherine Mayes
[36] Temam, R.: Navier-Stokes Equations: Theory and Numerical Analysis. AMS Chelsea Publishing, (2001)
[37] Vilela de Abreu, R., Jansson, N., Hoffman, J.: Computation of aeroacoustic sources for a Gulfstream G550 nose landing gear model using adaptive FEM. Comput. & Fluids **124**, 136–146 (2016). https://doi.org/10.1016/j.compfluid.2015.10.017

Matthias Maier
Department of Mathematics
Texas A&M University
3368 TAMU
College Station TX 77843
USA
e-mail: maier@tamu.edu

Peter Munch
Institute of Mathematics
Technical University of Berlin
Straße des 17. Juni 136
10623 Berlin
Germany
e-mail: muench@math.tu-berlin.de

Murtazo Nazarov
Division of Scientific Computing,
Department of Information Technology
Uppsala University
Lägerhyddsvägen 1
752 37 Uppsala
Sweden
e-mail: murtazo.nazarov@uu.se

(accepted: January 29, 2026)