Applications of Mathematics

Bohumír Bastl; Marek Brandner; Kristýna Slabá; Eva Turnerová

Consistent streamline residual-based artificial viscosity stabilization for numerical simulation of incompressible turbulent flow by isogeometric analysis

Applications of Mathematics, Vol. 67 (2022), No. 6, 805–829

Persistent URL: http://dml.cz/dmlcz/151057

67 (2022) APPLICATIONS OF MATHEMATICS No. 6, 805–829

CONSISTENT STREAMLINE RESIDUAL-BASED ARTIFICIAL VISCOSITY STABILIZATION FOR NUMERICAL SIMULATION OF INCOMPRESSIBLE TURBULENT FLOW BY ISOGEOMETRIC ANALYSIS

Bohumír Bastl, Marek Brandner, Kristýna Slabá,
Eva Turnerová, Plzeň

Received May 31, 2021. Published online June 6, 2022.

Abstract. In this paper, we propose a new stabilization technique for numerical simulation of incompressible turbulent flow by solving Reynolds-averaged Navier-Stokes equations closed by the SST k-ω turbulence model. The stabilization scheme is constructed such that it is consistent in the sense used in the finite element method, artificial diffusion is added only in the direction of convection and it is based on a purely nonlinear approach. We present numerical results obtained by our in-house incompressible fluid flow solver based on isogeometric analysis (IgA) for the benchmark problem of a wall bounded turbulent fluid flow simulation over a backward-facing step. Pressure coefficient and reattachment length are compared to experimental data acquired by Driver and Seegmiller, to the computational results obtained by open source software OpenFOAM and to the NASA numerical results.

Keywords: isogeometric analysis; turbulence modeling; spurious oscillations; stabilization techniques; B-splines; backward-facing step

MSC 2020: 35Q35, 65M12, 65M60

1. Introduction

Fluid mechanics belongs among the most challenging areas of research (in both theory and numerical simulation) and its development is driven by the requirement of the modern technologies for deeper understanding of behaviour of real fluids and as accurate as possible numerical simulations. Generally, the motion of the fluid is described

The research has been supported by the Czech Science Foundation (grant No. 19-04006S).

DOI: 10.21136/AM.2022.0131-21

by the Navier-Stokes equations (see, e.g., [41]), which are used to model a wide range of problems such as weather, water and some air flow or, e.g., study of blood flow.

The turbulent flow can be simulated by solving directly the Navier-Stokes equations for the given boundary and initial conditions, which is known as Direct Numerical Simulation (DNS). However, it means that the whole range of the scales (sizes of the eddies) must be resolved up to the smallest scales of the flow (cf., e.g., [41]), which substantially increases computational and memory requirements for higher Reynolds numbers. Thus, DNS is typically applied only for low Reynolds number flows.

The RANS (Reynolds-Averaged Navier-Stokes) approach is probably the most widely used method for numerical simulation of turbulent flows in the industry. RANS equations are time-averaged (ensemble-averaged) equations of fluid flow motion such that the variables are decomposed into a mean (time-averaged) component and a fluctuating component. The Reynolds stresses resulting from the time-averaging, see, e.g., [41], [31], are typically approximated using the Boussinesq assumptions, which gives rise to turbulence models. Various turbulence models have been proposed to satisfy different types and conditions of a fluid flow. In modern engineering applications, the commonly used turbulence models are, e.g., the one-equation Spalart-Allmaras and two-equation SST k-ω turbulence models, see [41], [24].

Isogeometric analysis (IgA) is a relatively recently introduced numerical method for solving partial differential equations (cf., e.g., [18]). The main motivation of isogeometric analysis is to bridge the gap between geometric modelling (Computer Aided Design—CAD) and numerical simulation (Finite Element Analysis—FEA). Isogeometric analysis is based on an isoparametric framework, i.e., the same shape functions are used for both the representation of geometry and the solution space of the problem unknowns. The method has been successfully applied in a wide range of practical problems, like, e.g., fluid flow simulation, heat transfer analysis, shape optimization, linear elasticity, plate and shell analysis, etc., see, e.g., [38], [8], [3], [2]. Exact representation of complex computational domains, and higher smoothness of basis functions, and higher accuracy of the numerical solution with respect to degrees of freedom (cf. [19]) are several advantages of the isogeometric analysis.

On the other hand, one of the disadvantages of isogeometric analysis is that true local refinement is not possible for the NURBS/B-spline objects, but it can be overcome by various generalizations, e.g., T-, THB- or LR-splines (cf., e.g., [15]). Further, among other drawbacks belong the time-consuming matrix assembly in the construction of the system of linear equations (see [16]) or degraded performance of direct solvers because of the wide support of B-spline/NURBS basis functions (see [10]).

Solving RANS equations closed with a turbulence model for numerical simulation of a turbulent flow is closely connected to the treatment of dominated convection or reaction terms. It is well known that the continuous Galerkin discretization produces unstable discretizations in the convection-dominated regime. In order to overcome these deficiencies, many stabilization techniques have been proposed with the aim to remove (or to diminish) spurious oscillations without leading to excessive smearing of discontinuities or layers.

Stabilization techniques used within the continuous Galerkin method can be divided into those that are based on the upwinding, on the modification of a variational formulation, and those based on the enrichment of a set of basis functions. They can also be distinguished as consistent or inconsistent (cf., e.g., [22], [1]). Consistency of a finite element method means that a sufficiently smooth solution satisfies also the discrete equation (see [36]). It is easy to show that a consistent finite element method has the property of Galerkin orthogonality. This property plays an important role in the proof of convergence and error estimation of stabilized methods. Some techniques only add numerical diffusion in the direction along the streamlines (e.g., streamline upwind/Petrov-Galerkin (SUPG) or streamline diffusion finite element method (SDFEM)) while others add it in multiple directions (isotropic diffusion), see [22], [20], [21]. From another perspective, linear and nonlinear schemes are distinguished. There are many linear approaches to stabilize Galerkin-based methods, like SUPG, GLS (generalized least squares), CIP (continuous interior penalty), LPS (local projection stabilization), etc., see, e.g., [7], [20], [21], [22]. It is well known that a solution computed with SUPG does not prevent the presence of spurious oscillations of small magnitudes in the vicinity of sharp gradients. Small overshoots and undershoots resulting from the SUPG method are caused by the fact that the SUPG method as one of the linear methods is neither monotone nor monotonicity preserving.

Nonlinear methods have been proposed to reduce the drawbacks of SUPG-type methods and the use of nonlinear methods is the only feasible way to obtain a high order of accuracy with strongly reduced local oscillations. These techniques are called spurious oscillations at layers diminishing schemes (SOLD), see, e.g., [20], [21]. The aim of these approaches is to reduce or remove spurious oscillations in the multidimensional case by adding artificial diffusion in directions where solutions are rapidly changing. Let us note that regardless the nonlinear stabilization technique is applied to a linear or nonlinear problem, the resulting stabilized problem is always nonlinear. This gives rise to questions concerning the existence and uniqueness of solutions of such problems and convergence of appropriate iterative methods. Unfortunately, very few results are known in this area.

Bazilevs et al. show in [5] advantages of the isogeometric analysis in conjunction with linear and nonlinear stabilization techniques. Especially, they show on a linear convection-dominated problem on one patch that IgA-based approximations are suitable for problems with internal and boundary layers. There are not many papers dealing with modeling incompressible turbulent fluid flow using IgA. Most of them are focused on LES (Large Eddy Simulation) or VMS (Variational MultiScale) techniques (cf., e.g., [4], [39]) and only few of them are devoted to RANS equations completed with the two-equation turbulence model solved by IgA based on a continuous Galerkin approach, see, e.g., [32].

The main goal of this paper is to propose a new stabilization technique suitable for RANS equations completed with a turbulence model in combination with IgA. This stabilization technique is not based on an additional term to the SUPG-type linear stabilization. On the contrary, it is based on a purely nonlinear approach (it is nonlinear even in the case of a 1D linear problem). This approach, which we call the *streamline residual-based artificial viscosity stabilization*, is consistent and nonlinear artificial viscosity is added only in the direction of convection. In our study of the stabilization techniques, we focus on a standard benchmark problem for the wall bounded turbulent fluid flow simulation—flow over a backward-facing step—which is frequently studied in the literature. The turbulent flow is simulated by solving stabilized RANS equations with the SST k-ω turbulence model. We compare the solutions obtained by our in-house IgA-based fluid flow solver (see [3], [2]) for different stabilization techniques and B-spline discretization spaces to experimental and computational data and we especially focus on the comparison with our newly proposed stabilization techniques. Moreover, we have to deal with a multi-patch geometry of the backward-facing step for numerical simulation of the turbulent flow which causes another instabilities. Papers dealing with stabilization techniques for IgA typically focus on single-patch geometries or multi-patch geometries with regular meshes (see, e.g., [26], [39], [18]), but only few of them are devoted to multi-patch geometries with irregular meshes (see [34] for convection-diffusion equations and [32] for RANS equations).

The paper is organized as follows. Section 2 introduces the RANS equations closed with the SST k-ω turbulence model and their weak formulation of the semi-discrete form. In Section 3, we give an overview of fundamentals of B-spline objects, present the main ideas of isogeometric analysis and formulate the Galerkin discretization. Section 4 is devoted to standard stabilization schemes. Then, we propose our new stabilization method in Section 5. In Section 6, we discuss the choice of the stabilization parameter and the element length. In Section 7, we formulate the benchmark problem studied in the paper, and we present the results obtained by our in-house IgA-based fluid flow solver and their comparison with the experimental data and reference solutions in Section 8. In Section 9, we conclude the paper.

2. Reynolds-averaged Navier-Stokes equations

Consider a bounded domain $\Omega \subset \mathbb{R}^d$, d being the number of spatial dimensions, with the boundary $\partial\Omega$ consisting of two parts, Dirichlet $\partial\Omega_D$ and Neumann $\partial\Omega_N$. The Reynolds-averaged Navier-Stokes problem is given as a system of $d+1$ ensemble-averaged (or time-averaged, see, e.g., [11], [41]) differential equations together with boundary and initial conditions

$$
\begin{aligned}
\frac{\partial \bar{\mathbf{u}}}{\partial t} + \bar{\mathbf{u}} \cdot \nabla \bar{\mathbf{u}} &= -\nabla \bar{p} + \nabla \cdot [(\nu + \nu_{\mathrm{T}})(\nabla \bar{\mathbf{u}} + \nabla \bar{\mathbf{u}}^{\top})] && \text{in } \Omega \times (0, \bar{t}), \\
\nabla \cdot \bar{\mathbf{u}} &= 0 && \text{in } \Omega \times (0, \bar{t}), \\
\bar{\mathbf{u}}(\mathbf{x}, 0) &= \bar{\mathbf{u}}_0(\mathbf{x}) && \text{in } \Omega, \\
\bar{\mathbf{u}} &= \mathbf{g} && \text{in } \partial\Omega_D \times [0, \bar{t}], \\
(\nu + \nu_{\mathrm{T}})(\nabla \bar{\mathbf{u}} + \nabla \bar{\mathbf{u}}^{\top}) \cdot \mathbf{n} - \mathbf{n}\bar{p} &= \mathbf{0} && \text{in } \partial\Omega_N \times [0, \bar{t}],
\end{aligned} \tag{2.1}
$$

where $\bar{\mathbf{u}} = \bar{\mathbf{u}}(\mathbf{x}, t)$ is the mean flow velocity, $\bar{p} = \bar{p}(\mathbf{x}, t)$ is the mean kinematic pressure, ν is the given kinematic viscosity of the fluid, ν_{T} is the eddy (turbulent) viscosity introduced below and $\mathbf{g}$ and $\mathbf{u}_0(\mathbf{x})$ are given functions. In essentially all practical formulations of the RANS equations, the time derivative term is included, despite the fact that $\bar{\mathbf{u}}$ is independent of time. This formulation is only auxiliary and the solution of the RANS equations is understood as stationary. Note that the so-called Boussinesq approximation (cf., e.g., [31]) is applied to obtain (2.1), which generally introduces also the term $\frac{2}{3}\nabla k$ on the right-hand side of the first equation in (2.1), where k is the turbulent kinetic energy. However, similarly to [42], [43], [41], the contribution of this term is neglected in this paper.

A wide range of approaches has been developed to approximate the eddy viscosity ν_{T} in the Boussinesq assumptions, which lead to different so-called turbulence models also known as eddy viscosity models. The turbulence models vary from relatively simple algebraic models to more complex models, e.g., one-equation models and two-equation models.

The SST (shear stress transport) k-ω turbulence model for unknown variables, the turbulent kinetic energy k and turbulent specific dissipation ω, is one of the most commonly used two-equation models and it can be formulated as (cf., e.g., [31])

$$
\begin{aligned}
\frac{\partial k}{\partial t} + \bar{\mathbf{u}} \cdot \nabla k &= P_k + \nabla \cdot [(\sigma_k \nu_{\mathrm{T}} + \nu)\nabla k] - \beta^* k\omega && \text{in } \Omega \times (0, \bar{t}), \\
\frac{\partial \omega}{\partial t} + \bar{\mathbf{u}} \cdot \nabla \omega &= \frac{\gamma}{\nu_{\mathrm{T}}} P_k + \nabla \cdot [(\sigma_\omega \nu_{\mathrm{T}} + \nu)\nabla \omega] - \beta \omega^2 \\
&\quad + 2(1 - F_1)\sigma_{\omega 2} \frac{1}{\omega} \nabla k \cdot \nabla \omega && \text{in } \Omega \times (0, \bar{t}),
\end{aligned} \tag{2.2}
$$

$$
\begin{aligned}
k(\mathbf{x},0) &= k_0(\mathbf{x}) && \text{in } \Omega,\\
\omega(\mathbf{x},0) &= \omega_0(\mathbf{x}) && \text{in } \Omega,\\
k &= g^k && \text{in } \partial\Omega_D \times [0,\bar{t}],\\
\omega &= g^\omega && \text{in } \partial\Omega_D \times [0,\bar{t}],\\
\nabla k \cdot \mathbf{n} &= 0 && \text{in } \partial\Omega_N \times [0,\bar{t}],\\
\nabla \omega \cdot \mathbf{n} &= 0 && \text{in } \partial\Omega_N \times [0,\bar{t}],
\end{aligned}
$$

where g^k, g^ω are given functions and

$$
\begin{aligned}
F_1 &= \tanh\Big(\Big[\min\Big[\max\Big(\frac{\sqrt{k}}{\beta^*\omega y}, \frac{500\nu}{y^2\omega}\Big), \frac{4\sigma_{\omega 2}k}{CD_{k\omega}y^2}\Big]\Big]^4\Big),\\
CD_{k\omega} &= \max\Big(2\sigma_{\omega 2}\frac{1}{\omega}\nabla k \cdot \nabla\omega, 10^{-10}\Big),\\
P_k &= \min\Big(2\nu_{\mathrm{T}}\sum_{i,j=1}^{d} S_{ij}\frac{\partial \bar{u}_i}{\partial \bar{x}_j}, 10\beta^* k\omega\Big),\\
S_{ij} &= \frac{1}{2}\Big(\frac{\partial \bar{u}_i}{\partial \bar{x}_j} + \frac{\partial \bar{u}_j}{\partial \bar{x}_i}\Big),
\end{aligned}
\tag{2.3}
$$

$\beta^* = \frac{9}{100}$, $\sigma_{\omega 2} = 0.856$. The values of the remaining parameters σ_k, σ_ω, α, β, and γ are dependent on the wall distance y, which is the distance of a point in the domain Ω to the nearest wall. Let φ_1 and φ_2 be two given parameters. Then, let us define a parameter φ, whose value depends on the wall distance y, such that it varies between the given parameters φ_1, φ_2 as

$$
\varphi = \varphi_1 F_1 + \varphi_2(1 - F_1). \tag{2.4}
$$

This relation is applied to calculate appropriate values of the parameters σ_k, σ_ω, α, and β using

$$
\begin{aligned}
\sigma_{k1} = 0.85, \quad \sigma_{\omega 1} = 0.5, \quad \alpha_1 = \frac{5}{9}, \quad \beta_1 = \frac{3}{40},\\
\sigma_{k2} = 1, \quad \sigma_{\omega 2} = 0.856, \quad \alpha_2 = 0.44, \quad \beta_2 = 0.0828,
\end{aligned}
\tag{2.5}
$$

and parameter γ using

$$
\gamma_1 = \frac{\beta_1}{\beta^*} - \frac{\sigma_{\omega 1}\kappa^2}{\sqrt{\beta^*}}, \quad \gamma_2 = \frac{\beta_2}{\beta^*} - \frac{\sigma_{\omega 2}\kappa^2}{\sqrt{\beta^*}}, \tag{2.6}
$$

where $\kappa = 0.41$. Since the two-equation model switches according to the wall distance y, the eddy viscosity has to be also dependent on the wall distance, i.e.,

$$
\nu_{\mathrm{T}} = \frac{a_1 k}{\max(a_1\omega, SF_2)}, \tag{2.7}
$$

where $a_1 = 0.31$, $S = \sum_{i,j=1}^{d} 2S_{ij}S_{ij}$ is magnitude of the strain rate tensor S_{ij} and

$$F_2 = \tanh\Big(\Big[\max\Big(\frac{2\sqrt{k}}{\beta^*\omega y}, \frac{500\nu}{y^2\omega}\Big)\Big]^2\Big). \tag{2.8}$$

Then, the problems (2.1) and (2.2) are fully coupled because ν_{T} in (2.1) depends on the solution of (2.2) and the solution of (2.2) depends on the solution $\bar{\mathbf{u}}$ of (2.1). Further, presence of the time derivative of $\bar{\mathbf{u}}$ is helpful for the convergence of numerical methods.

In the following, a weak formulation of the RANS equations is presented. To treat the time dependence of the RANS problem, we apply a semi-discrete method such that we start with the time discretization. The time discretization is used as an iterative approach to determine a steady-state solution. Let superscripts n and $n+1$ denote the values at the time layers t_n and t_{n+1}. Then, applying the semi-implicit time discretization we look for $\bar{\mathbf{u}}^{n+1}$ and $\bar{p}^{n+1}$ such that

$$\begin{aligned}\frac{\bar{\mathbf{u}}^{n+1} - \bar{\mathbf{u}}^n}{\Delta t} - \nabla\cdot[(\nu + \nu_{\mathrm{T}}^n)(\nabla\bar{\mathbf{u}}^{n+1} + (\nabla\bar{\mathbf{u}}^{n+1})^\top)] + \bar{\mathbf{u}}^{n+1}\cdot\nabla\bar{\mathbf{u}}^{n+1} + \nabla\bar{p}^{n+1} &= \mathbf{0},\\ \nabla\cdot\bar{\mathbf{u}}^{n+1} &= 0.\end{aligned} \tag{2.9}$$

In order to formulate the weak formulation of the semi-discrete problem, we define solution and test function spaces as

$$\begin{aligned}V &= \{\mathbf{u}\in H^1(\Omega)^d;\ \mathbf{u} = \mathbf{g} \text{ on } \partial\Omega_D\},\\ V_0 &= \{\mathbf{v}\in H^1(\Omega)^d;\ \mathbf{v} = \mathbf{0} \text{ on } \partial\Omega_D\},\end{aligned} \tag{2.10}$$

where the boundary equalities are understood in the sense of traces and we assume that $\mathbf{g}\in H^{1/2}(\partial\Omega)^d$. The weak formulation of the RANS problem in semi-discrete form is to find $\bar{\mathbf{u}}^{n+1}\in V$ and $\bar{p}^{n+1}\in L^2(\Omega)$, $n = 0, 1, 2, \ldots$, satisfying $\bar{\mathbf{u}}^0 = \bar{\mathbf{u}}_0(\mathbf{x})$ and

$$\begin{aligned}\Big(\frac{\bar{\mathbf{u}}^{n+1} - \bar{\mathbf{u}}^n}{\Delta t}, \mathbf{v}\Big) + a(\bar{\mathbf{u}}^{n+1}, \mathbf{v}, \nu_{\mathrm{T}}^n) + c(\bar{\mathbf{u}}^{n+1}, \bar{\mathbf{u}}^{n+1}, \mathbf{v}) - b(\mathbf{v}, \bar{p}^{n+1}) &= 0,\\ b(\bar{\mathbf{u}}^{n+1}, q) &= 0\end{aligned} \tag{2.11}$$

for all $\mathbf{v}\in V_0$ and $q\in L^2(\Omega)$, where

$$\begin{aligned}a(\mathbf{u}, \mathbf{v}, \nu_{\mathrm{T}}) &= ((\nu + \nu_{\mathrm{T}})(\nabla\mathbf{u} + (\nabla\mathbf{u})^\top), \nabla\mathbf{v}),\\ c(\mathbf{w}, \mathbf{u}, \mathbf{v}) &= (\mathbf{w}\cdot\nabla\mathbf{u}, \mathbf{v}),\\ b(\mathbf{v}, p) &= (\nabla\cdot\mathbf{v}, p)\end{aligned} \tag{2.12}$$

and the L^2 scalar products in Ω are defined as

$$(\mathbf{u},\mathbf{v}) = \int_\Omega \mathbf{u}\cdot\mathbf{v} \quad \text{for vector functions,} \tag{2.13}$$

$$(\sigma,\tau) = \int_\Omega \sigma : \tau \quad \text{for second-order tensors,} \tag{2.14}$$

$$(p,q) = \int_\Omega pq \quad \text{for scalar functions.} \tag{2.15}$$

The weak formulation of the turbulence model is derived similarly to the RANS equations, i.e., we use the semi-implicit time discretization to discretize the problem in time first and then we look for a weak form of the semi-discrete problem. For this purpose, we define solution and test function spaces as

$$\begin{aligned} \widehat{V} &= \{k \in H^1(\Omega);\ k = g^k \text{ on } \partial\Omega_D\},\\ \widetilde{V} &= \{\omega \in H^1(\Omega);\ \omega = g^\omega \text{ on } \partial\Omega_D\},\\ \widehat{V}_0 &= \{w \in H^1(\Omega);\ w = 0 \text{ on } \partial\Omega_D\}, \end{aligned} \tag{2.16}$$

where the boundary equalities are understood in the sense of traces and we assume that $g^k, g^\omega \in H^{1/2}(\partial\Omega)$. Then, the weak formulation is: find $k^{n+1} \in \widehat{V}$ and $\omega^{n+1} \in \widetilde{V}$, $n = 0, 1, 2, \ldots$, such that $k^0 = k_0(\mathbf{x})$, $\omega^0 = \omega_0(\mathbf{x})$ and

$$\begin{aligned} &\Big(\frac{k^{n+1}-k^n}{\Delta t}, w\Big) + a_k(k^{n+1}, w, \nu_{\mathrm{T}}^{n+1}) + c_k(\bar{\mathbf{u}}^{n+1}, k^{n+1}, w)\\ &\qquad + (\beta^*\omega^{n+1}k^{n+1}, w) = (P_k, w),\\ &\Big(\frac{\omega^{n+1}-\omega^n}{\Delta t}, w\Big) + a_\omega(\omega^{n+1}, w, \nu_{\mathrm{T}}^{n+1}) + c_\omega(\bar{\mathbf{u}}^{n+1}, \omega^{n+1}, w)\\ &\qquad + (\beta(\omega^{n+1})^2, w) - d_\omega(\omega^{n+1}, k^{n+1}, w) = \Big(\frac{\gamma}{\nu_{\mathrm{T}}^n}P_k, w\Big) \end{aligned} \tag{2.17}$$

for all $w \in \widehat{V}_0$, where

$$\begin{aligned} a_k(k,w,\nu_{\mathrm{T}}) &= ((\sigma_k\nu_{\mathrm{T}}+\nu)\nabla k, \nabla w), \quad c_k(\bar{\mathbf{u}},k,w) = (\bar{\mathbf{u}}\cdot\nabla k, w),\\ a_\omega(\omega,w,\nu_{\mathrm{T}}) &= ((\sigma_\omega\nu_{\mathrm{T}}+\nu)\nabla\omega, \nabla w), \quad c_\omega(\bar{\mathbf{u}},\omega,w) = (\bar{\mathbf{u}}\cdot\nabla\omega, w),\\ d_\omega(\omega,k,w) &= \Big(2(1-F_1)\sigma_{\omega 2}\frac{1}{\omega}\nabla k\cdot\nabla\omega, w\Big). \end{aligned} \tag{2.18}$$

Due to the explicit discretization of ν_{T} in (2.11), the problems (2.11) and (2.17) can be separated.

3. Isogeometric analysis and Galerkin discretization of RANS equations

Isogeometric analysis (IgA) shares a lot of features with the Finite Element Method, but it is closely connected to the description of geometry, i.e., it directly uses the geometric description from the CAD system also for numerical analysis. Indeed, the computational domain with a boundary represented as B-spline/NURBS objects (cf. [35], [18]) is exactly discretized and then an isoparametric approach is applied. This is the main advantage of IgA, which cannot be achieved by a FEM polynomial description of the boundaries.

A B-spline surface of degree q is a vector function of two parameters (ξ and ψ) determined by a control net of control points $\mathbf{P}_{i,j}$, $i = 1, 2, \ldots, n$, $j = 1, 2, \ldots, m$, and two knot vectors $\Xi = (\xi_1, \ldots, \xi_{n+q+1})$, $\Psi = (\psi_1, \ldots, \psi_{m+q+1})$, where $\xi_i \leqslant \xi_{i+1}$ and $\psi_i \leqslant \psi_{i+1}$. Then its parametrization is given as

$$f(\xi, \psi) = \sum_{i=1}^{n} \sum_{j=1}^{m} \mathbf{P}_{i,j} N_i^q(\xi) M_j^q(\psi) = \sum_{i=1}^{n} \sum_{j=1}^{m} \mathbf{P}_{i,j} B_{ij}(\xi, \psi). \tag{3.1}$$

Let us note that f can generally be of degree p, q in parameters ξ, ψ, but for the simplicity of notation we restrict ourselves to the case $p = q$ in this paper. The B-spline basis functions $N_i^q(\xi)$ and $M_j^q(\psi)$ are determined by the knot vectors Ξ and Ψ and degree q recursively by the Cox-de Boor formula as

$$\begin{aligned} N_i^0(\xi) &= \begin{cases} 1, & \xi_i \leqslant \xi < \xi_{i+1}, \\ 0 & \text{otherwise}, \end{cases} \\ N_i^k(\xi) &= s_i^k(\xi) N_i^{k-1}(\xi) + (1 - s_{i+1}^k(\xi)) N_{i+1}^{k-1}(\xi), \quad k = 1, \ldots, q, \end{aligned} \tag{3.2}$$

where

$$s_i^k(\xi) := \begin{cases} \dfrac{\xi - \xi_i}{\xi_{i+k} - \xi_i}, & \xi_i < \xi_{i+1}, \\ \text{arbitrary} & \text{otherwise}. \end{cases} \tag{3.3}$$

$B_{ij}(\xi, \psi)$ in (3.1) is the corresponding tensor product B-spline basis function $N_i^q(\xi) M_j^q(\psi)$. One of the fundamental properties of B-spline basis functions is that they are C^{q-m_i} continuous at the knot ξ_i, where m_i is the multiplicity of the knot ξ_i. Other properties include local support, non-negativity and partition of unity. More details about the B-spline basis can be found in [35], [18], [3].

Let us denote the tensor product B-spline space spanned by the basis functions $B_{ij}(\xi, \psi)$, $i = 1, \ldots, n$, $j = 1, \ldots, m$, by

$$S_r^q = \operatorname{span}\{B_{ij}\}_{i=1,j=1}^{n,m}, \tag{3.4}$$

where r is the order of continuity of the basis functions, and the Cartesian product of these tensor product B-spline spaces by

$$\mathbf{S}_r^q = S_r^q \times S_r^q. \tag{3.5}$$

A B-spline mesh used in isogeometric analysis is defined by the knot vectors Ξ, Ψ. If the subsequent knots are different from each other, i.e., $\xi_i \neq \xi_{i+1}$ and $\psi_j \neq \psi_{j+1}$, then $[\xi_i, \xi_{i+1}] \times [\psi_j, \psi_{j+1}]$ defines an element of the computational mesh.

The idea of the Galerkin discretization method in IgA applied to RANS equations is to introduce finite dimensional subspaces $V^h \subset V$, $V_0^h \subset V_0$, $W^h \subset L_2(\Omega)$, $\widehat{V}^h \subset \widehat{V}$, $\widetilde{V}^h \subset \widetilde{V}$, $\widehat{V}_0^h \subset \widehat{V}_0$ together with their basis functions which define also the geometry of Ω and solve the RANS problem projected into these subspaces. Moreover, to get a stable Galerkin discretization of a saddle-point problem, the velocity and pressure spaces V_0^h and W^h must satisfy the so-called LBB (Ladyženskaja-Babuška-Brezzi) condition

$$\inf_{q \in W^h \setminus \{0\}} \sup_{\mathbf{v} \in V_0^h \setminus \{\mathbf{0}\}} \frac{\int_\Omega q \nabla \cdot \mathbf{v}}{\|\mathbf{v}\|_{H^1(\Omega)} \|q\|_{L^2(\Omega)}} \geqslant \gamma > 0, \tag{3.6}$$

where γ is a constant independent of the mesh. Here, we consider only the case of so-called inf-sup stable combinations of velocity and pressure spaces ensuring that the inf-sup condition is satisfied. The reader is referred, e.g., to [14], [41] for more details, discussion and basic theory of the saddle point problems and their numerical analysis.

We apply an extension of the classical Taylor-Hood finite elements for the isogeometric analysis which is proved to satisfy the inf-sup condition (see, e.g., [6]), i.e., the pressure basis functions are taken to be equal to the geometry basis functions ($B_l^p(\xi, \psi) = B_l$ for all l) and the velocity basis functions $\{B_j^u(\xi, \psi)\}$ as the B-spline basis functions obtained by degree elevation of the pressure basis functions $\{B_l^p(\xi, \psi)\}$. Hence, the velocity basis functions are of bi-degree $(q+1, q+1)$, and both velocity and pressure bases are defined on the same mesh and have the same order of continuity. This yields

$$\begin{aligned}
V^h &= \{\mathbf{u} : \mathbf{u} \circ \mathbf{F} \in \mathbf{S}_r^{q+1} \wedge \mathbf{u} = \mathbf{g} \text{ on } \partial\Omega_D\},\\
W^h &= \{q : q \circ \mathbf{F} \in S_r^q\},\\
V_0^h &= \{\mathbf{v} : \mathbf{v} \circ \mathbf{F} \in \mathbf{S}_r^{q+1} \wedge \mathbf{u} = \mathbf{0} \text{ on } \partial\Omega_D\},\\
\widehat{V}^h &= \{k : k \circ \mathbf{F} \in S_r^q \wedge k = g^k \text{ on } \partial\Omega_D\},\\
\widetilde{V}^h &= \{\omega : \omega \circ \mathbf{F} \in S_r^q \wedge k = g^\omega \text{ on } \partial\Omega_D\},\\
\widehat{V}_0^h &= \{w : w \circ \mathbf{F} \in S_r^q \wedge w = 0 \text{ on } \partial\Omega_D\},
\end{aligned}$$

where $\mathbf{F}$ is the parameterization representing the mapping from the parameteric domain $\widehat{\Omega}$ to the physical space Ω. We assume that the parameterization $\mathbf{F}$ is invertible with smooth inverse on each element of the computational mesh.

To treat the nonlinearity in the convective term of the RANS problem, we employ Picard's method, i.e., the problem is solved iteratively at each time step and the nonlinear term is linearized using the solution from the previous Picard's iteration step [14]. Then, we look for discrete solutions $\bar{\mathbf{u}}_h^{n+1,m+1} \in V^h$, $\bar{p}_h^{n+1,m+1} \in W^h(\Omega)$, $m = 0, 1, \ldots, M$, such that for all $\mathbf{v}_h \in V_0^h$ and $q_h \in W^h(\Omega)$

$$\begin{aligned}(3.7)\qquad &\frac{1}{\Delta t}(\bar{\mathbf{u}}_h^{n+1,m+1}, \mathbf{v}_h) + a(\bar{\mathbf{u}}_h^{n+1,m+1}, \mathbf{v}_h, \nu_{\mathrm{T}}^n) + c(\bar{\mathbf{u}}_h^{n+1,m}, \bar{\mathbf{u}}_h^{n+1,m+1}, \mathbf{v}_h)\\ &\qquad - b(\mathbf{v}_h, \bar{p}_h^{n+1,m+1}) = \frac{1}{\Delta t}(\bar{\mathbf{u}}_h^n, \mathbf{v}_h),\\ &\qquad b(\bar{\mathbf{u}}_h^{n+1,m+1}, q_h) = 0.\end{aligned}$$

In the first time step, $\bar{\mathbf{u}}^{1,0} = \bar{\mathbf{u}}_0$ is considered as the given initial condition from (2.1). Then, $\bar{\mathbf{u}}^{n+1,0} = \bar{\mathbf{u}}^{n,M}$ is considered in all other time steps. Considering the basis of V_0^h formed by vector functions

$$(\underbrace{B_i^u(\mathbf{x}), 0, \ldots, 0}_{d}), \ldots, (\underbrace{0, \ldots, B_i^u(\mathbf{x})}_{d}), \quad i = 1, \ldots, n_u,$$

and the basis of W^h formed by functions $B_i^p(\mathbf{x})$, $i = 1, \ldots, n_p$, we eventually arrive at a non-symmetric saddle-point linear system, which is not mentioned here, but can be found, e.g., in [3]. The system matrix is sparse thanks to local supports of the B-spline basis functions.

When the iteration process (3.7) converges or the maximum number of Picard's iterations is reached, the obtained solution is used to iterate analogously linearized turbulence model, i.e., we look for $k_h^{n+1,\widetilde{m}+1} \in \widehat{V}^h$, $\omega_h^{n+1,\widetilde{m}+1} \in \widetilde{V}^h$, $\widetilde{m} = 0, 1, \ldots, \widetilde{M}$, such that for all $w_h \in \widehat{V}_0^h$ we have

$$\begin{aligned}(3.8)\qquad &\Big(\frac{k_h^{n+1,\widetilde{m}+1} - k_h^n}{\Delta t}, w_h\Big) + a_k(k_h^{n+1,\widetilde{m}+1}, w_h, \nu_{\mathrm{T}}^{n+1,\widetilde{m}})\\ &\qquad + c_k(\bar{\mathbf{u}}_h^{n+1}, k_h^{n+1,\widetilde{m}+1}, w_h) + (\beta^* \omega_h^{n+1,\widetilde{m}} k_h^{n+1,\widetilde{m}+1}, w_h) = (P_k, w_h),\\ &\Big(\frac{\omega_h^{n+1,\widetilde{m}+1} - \omega_h^n}{\Delta t}, w_h\Big) + a_\omega(\omega_h^{n+1,\widetilde{m}+1}, w_h, \nu_{\mathrm{T}}^{n+1,\widetilde{m}}) + c_\omega(\bar{\mathbf{u}}_h^{n+1}, \omega_h^{n+1,\widetilde{m}+1}, w_h)\\ &\qquad + (\beta \omega_h^{n+1,\widetilde{m}} \omega_h^{n+1,\widetilde{m}+1}, w_h) - d_\omega(\omega_h^{n+1,\widetilde{m}}, k_h^{n+1,\widetilde{m}}, w_h) = \Big(\frac{\gamma}{\nu_{\mathrm{T}}^n} P_k, w_h\Big).\end{aligned}$$

Note that all the parameters (like, e.g., σ_k or P_k etc.) of the turbulence model are functions of both space and time and these parameters are evaluated from the last obtained solution (e.g., $\sigma_k^{n+1,\widetilde{m}}$ etc.).

To achieve a stable iterative process, we add stabilization terms introduced below to the formulation of the discrete problem (3.7)–(3.8). Also, a special treatment of the last blending term in the ω-equation in (3.8) is implemented according to [28].

It is well-known that the tensor-product structure of B-spline/NURBS objects prevents the possibility to describe an arbitrary computational domain by one B-spline/NURBS object. Thus, so-called multi-patch B-spline/NURBS meshes are typically used for the description of computational domains in isogeometric analysis. Then, any solver based on isogeometric analysis working on multipatch domains has to be able to join B-spline/NURBS patches along their interfaces into one computational domain. Only the conforming connection of meshes of patches is considered in this paper, i.e., patches have along a common interface the same elements (discretization) and the same control nets describing this interface. Conforming meshes can be handled easily—it is enough to identify the corresponding control points determining the common interface on both patches.

4. Stabilization techniques for IgA

In this section, we discuss only stabilization techniques which can be used together with IgA and which are then employed in numerical experiments in Section 7. The stabilization schemes are formulated only for the RANS equations, but their application to the turbulence model is analogous.

4.1. Classical streamline diffusion (CSD). The classical streamline diffusion method is one of the oldest stabilization techniques. Employing the CSD stabilization, the following artificial diffusion term is added to the formulation of a discrete problem (3.7), i.e.,

$$\sum_{i_{\mathrm{e}}=1}^{n_{\mathrm{el}}} \int_{Q_{i_{\mathrm{e}}}} \tau_{\mathrm{S}} (\mathbf{b} \cdot \nabla \bar{\mathbf{u}}_h^{n+1,m+1})(\mathbf{b} \cdot \nabla \mathbf{v}_h), \tag{4.1}$$

where $\mathbf{b} = \bar{\mathbf{u}}_h^{n+1,m}$ is the convection coefficient, n_{el} is the number of elements $Q_{i_{\mathrm{e}}}$ and $\tau_{\mathrm{S}} \in L^\infty(\Omega)$ is a suitable non-negative stabilization parameter, which is given in Section 6. Although this type of linear CSD stabilization leads to an over-diffusive solution and makes the whole approximation a first-order scheme, this approach can be advantageous as an initialization technique in the search for a steady state solution.

For space-time formulation, we arrive at T-CSD stabilization

$$\sum_{i_{\mathrm{e}}=1}^{n_{\mathrm{el}}} \int_{Q_{i_{\mathrm{e}}}} \tau_{\mathrm{S}} \Big(\frac{\bar{\mathbf{u}}_h^{n+1,m+1} - \bar{\mathbf{u}}_h^{n}}{\Delta t} + \mathbf{b} \cdot \nabla \bar{\mathbf{u}}_h^{n+1,m+1} \Big) (\mathbf{b} \cdot \nabla \mathbf{v}_h). \tag{4.2}$$

4.2. Streamline Upwind/Petrov-Galerkin (SUPG) stabilization. The SUPG method proposed in [7] has become one of the most popular stabilization method. The weighted element residual

$$\sum_{i_e=1}^{n_{el}} \int_{Q_{i_e}} \tau_S R(\bar{\mathbf{u}}_h^{n+1,m+1}, \bar{p}_h^{n+1,m+1}) \mathbf{b} \cdot \nabla \mathbf{v}_h \tag{4.3}$$

is added to the Galerkin problem (3.7), where τ_S is a suitably chosen non-negative SUPG stabilization parameter which is explained in Section 6, $\mathbf{b} = \bar{\mathbf{u}}_h^{n+1,m}$ is the convection coefficient and $R(\bar{\mathbf{u}}_h^{n+1,m+1}, \bar{p}_h^{n+1,m+1})$ is the residual given as

$$R(\bar{\mathbf{u}}, \bar{p}) = \frac{\bar{\mathbf{u}} - \bar{\mathbf{u}}_h^n}{\Delta t} - \nabla \cdot [(\nu + \nu_T^n)(\nabla \bar{\mathbf{u}} + (\nabla \bar{\mathbf{u}})^\top)] + \bar{\mathbf{u}}_h^{n+1,m} \cdot \nabla \bar{\mathbf{u}} + \nabla \bar{p}. \tag{4.4}$$

For linear finite elements and B-splines of the first degree, the second derivative vanishes within each element. However, this term cannot be neglected if higher order basis functions are used (which are typical for IgA).

4.3. Crosswind stabilization. As mentioned above, the SUPG stabilization can be assumed to add numerical diffusion in a streamline direction. The crosswind diffusion is employed to reduce spurious oscillations in a crosswind direction. The crosswind diffusion term is constructed such that the resulting stabilization is consistent and can be written as (cf., e.g., [20])

$$\sum_{i_e=1}^{n_{el}} \int_{Q_{i_e}} \tau_{cw} P^\perp \nabla \bar{\mathbf{u}}_h^{n+1,m+1} \cdot \nabla \mathbf{v}_h, \tag{4.5}$$

where τ_{cw} is the crosswind stabilization parameter and $P^\perp$ is an orthogonal projector defined by

$$P^\perp = \mathbf{I} - \frac{\mathbf{b} \otimes \mathbf{b}}{\|\mathbf{b}\|^2}. \tag{4.6}$$

If the magnitude of the convection coefficient equals zero, then $P^\perp = \mathbf{0}$. Many formulas have been proposed for τ_{cw}, which is highly non-trivial to define, but the variant

$$\tau_{cw} = \frac{\tau_S \|\mathbf{b}\|^2 \|R(\bar{\mathbf{u}}_h^{n+1,m+1}, \bar{p}_h^{n+1,m+1})\|}{\|\mathbf{b}\| \|\nabla \bar{\mathbf{u}}_h^{n+1,m}\| + \|R(\bar{\mathbf{u}}_h^{n+1,m+1}, \bar{p}_h^{n+1,m+1})\|} \tag{4.7}$$

(cf. [20]) is used in our numerical experiments, where τ_S is the stabilization parameter discussed in Section 6.

5. Streamline residual-based artificial viscosity stabilization

In this section, we propose a new stabilization technique, where the streamline artificial viscosity term is added to the discrete weak form (3.7) as

$$\sum_{i_{\mathrm{e}}=1}^{n_{\mathrm{el}}} \int_{Q_{i_{\mathrm{e}}}} \tau(\mathbf{b}\cdot\nabla\bar{\mathbf{u}}_h^{n+1,m+1})(\mathbf{b}\cdot\nabla\mathbf{v}_h), \tag{5.1}$$

where $\mathbf{b} = \bar{\mathbf{u}}_h^{n+1,m}$ is the convection coefficient and τ is a suitable stabilization parameter which we propose in the form

$$\tau = C_1\tau_{\mathrm{S}}h^{\alpha}\tanh^2(C_2R(\bar{\mathbf{u}}_h^{n+1,m},\bar{p}_h^{n+1,m})), \tag{5.2}$$

where $\alpha \geqslant 0$, τ_{S} is a non-negative parameter defined in Section 6, h is an element length also discussed in Section 6, R is the element residual defined by (4.4) and C_1 and C_2 are suitable scale factors. We call this stabilization method *Streamline Residual-Based Artificial Viscosity (SRBAV) method*.

This new stabilization technique acts in the streamline direction, has the character of a viscous (diffusion) term and uses artificial viscosity, which, among other things, ensures the consistency of the scheme. Thus, this stabilization technique is a modification of the CSD method (only the diffusion term is added, which acts in the direction of the flow) and the procedure proposed by Nazarov for scalar conservation law in [29], [30]. However, the technique proposed by Nazarov is isotropic and based on the residual norm. In our case, we use the residual only as a switch activating the stabilization. At the same time, we use the stabilization parameter τ_{S} originating from the SUPG method. We also emphasize that this stabilization is genuinely nonlinear (i.e., it is nonlinear even for the linear 1D case) and it is not based on the idea of supplementing the SUPG-type linear stabilization with a nonlinear term in the case of multiple dimensions.

Let us emphasize that for large values of the residuals, the value of $\tanh^2$ approaches the value of one. Thus, the stabilization (specifically for $\alpha = 0$) is close to the CSD stabilization. This prevents the artificial viscosity values from being too large for large values of residuals and the numerical solution from being inappropriately smeared. At the same time, however, the stabilization is consistent because the stabilization term vanishes for residuals going to zero.

The factors C_1 and C_2 are chosen such that $C_1 \propto 1/L^{\alpha}$ and $C_2 \propto L/U^2$ for the RANS equations, where L is the characteristic length scale and U is the characteristic velocity of the flow that appear in the definition of the Reynolds number. This ensures invariance to data scaling unless we use a dimensionless formulation of the problem. The factors C_1 and C_2 are chosen analogously for the equations of the turbulence model.

A more detailed analysis of how to appropriately choose the parameter α and how to modify the switching function, if necessary, will be a part of a future work.

6. Choice of the stabilization parameter τ_{S} and element length

The choice of the stabilization parameter τ_{S} is a challenging part of the stabilization methods. The parameter usually involves a characteristic size of an element that plays also a significant role. Most of the literature deals with the linear steady convection-diffusion equation, thus the stabilization parameters are only convection dependent or convection and diffusion dependent. Only a small number of papers focuses on the stabilization parameter which takes into account also reaction, see [9], [22], [26], [39]. There are some texts dealing with higher order basis functions, see, e.g., [26], [18], [39], [34]. However, formulas for the stabilization parameter are not optimal for general problems because they are typically constructed for a particular discrete problem with no clear way for generalization (cf., e.g., [20]).

If the coefficients of the problem solved are not constant or the problem also involves reaction, the following stabilization parameter is recommended for the general unsteady convection-diffusion-reaction problem (cf., e.g., [12])

$$\tau_{\mathrm{S}} = \Big(\Big(\frac{2\|\mathbf{b}\|}{h}\Big)^2 + 9\Big(\frac{4\varepsilon}{h^2}\Big)^2 + \Big(\frac{2}{\Delta t} + |r|\Big)^2\Big)^{-1/2}, \tag{6.1}$$

where $\mathbf{b}$ is the convection coefficient, ε is the diffusion coefficient, r is the reaction coefficient (i.e., $\mathbf{b} = \bar{\mathbf{u}}$, $\varepsilon = \nu + \nu_{\mathrm{T}}$ and $r = 0$ for RANS equations) and h is the characteristic size of an element. This stabilization parameter was proposed by Shakib and Hughes (cf. [37]) and this formula is fourth-order accurate. An asymptotic analysis near the different limits shows the superior convergence of (6.1) to the theoretical values compared with

$$\tau_{\mathrm{S}} = \Big(\frac{2\|\mathbf{b}\|}{h} + \frac{4\varepsilon}{h^2} + \frac{2}{\Delta t} + |r|\Big)^{-1}. \tag{6.2}$$

However, the fourth-order accuracy does not extend to higher dimensions or general systems. Nevertheless, numerical experiments indicate that (6.1) presents slightly lower errors (in the C^2 norm) compared with (6.2). Alternative formulations for the stabilization parameter τ_{S} are discussed, e.g., in [22].

One should be aware of the fact that τ_{S} defined in (6.1) is not robust for small time steps. Indeed, $\tau_{\mathrm{S}} \to 0$ as $\Delta t \to 0$, i.e., the stabilization effect vanishes for small time steps and the spurious oscillations may be expected in the vicinity of the sharp layers, see [17], [22]. As we are interested in a stationary solution of the RANS problem, the term with the time step size is omitted in (6.1) in our computations.

The computation of the characteristic size of an element h is another key component of the stabilization parameter τ_{S} in (6.1) and τ in (5.2). The appropriate measure of the element h is usually chosen as the element length in the specific direction, such as the direction of the convection or solution gradient, see, e.g., [34], [39], [20]. Therefore, we consider the convection-dependent element length in the form (cf., e.g., [40])

$$h = 2\|\mathbf{b}\| \left(\sum_{a=1}^{n_{\mathrm{en}}} |\mathbf{b} \cdot \nabla N_{\mathrm{a}}| \right)^{-1} \tag{6.3}$$

in our computations (usually denoted h_{UGN}), where N_{a} and n_{en} are the active basis function and the number of active basis functions associated with an evaluation point.

Note that the appropriate choice of the direction-dependent element length h is still not clear for reaction dominated problems such as the turbulence model, see, e.g., [22]. Alternative formulations for the computation of h in isogeometric analysis and a convection-reaction dominated problem will be studied in a future work.

7. Flow over a backward-facing step

In this section, we introduce the benchmark wall bounded problem for turbulent fluid flow simulation—flow over a backward-facing step—which is a classical problem used to study separation, recirculation and reattachment of the flow and thus to validate the turbulence models. We investigate numerical results obtained by our in-house IgA-based fluid flow solver using stabilization techniques mentioned in Section 4 and our new stabilization scheme proposed in Section 5, and we compare them to the experimental data acquired by Driver and Seegmiller [13], to the numerical results obtained by open source software OpenFOAM in [33] and to the NASA computational SSTm results from CFL3D (Computational Fluid Laboratory 3-Dimensional) CFD flow solver which are available in [25]. Especially, we focus on validation of the pressure coefficient and reattachment length.

7.1. Problem description and setting. The problem description is shown in illustrative Figure 1, where $H = 0.0127\,\mathrm{m}$ is the step height, the height of the inlet channel is $8H$ and its length is $110H$, outlet channel length is $50H$ in our computations. The backward-facing step (BFS) problem represents a flow of fluid which enters the computational domain through the inlet and develops fully into a turbulent flow before reaching the step. Because of the sudden change in geometry, the flow separates at the step corner, a recirculation region occurs next to the step and the flow again reattaches with the bottom wall behind the step at the point of reattachment. The distance from the point of separation to the point of reattachment is the reattachment length.

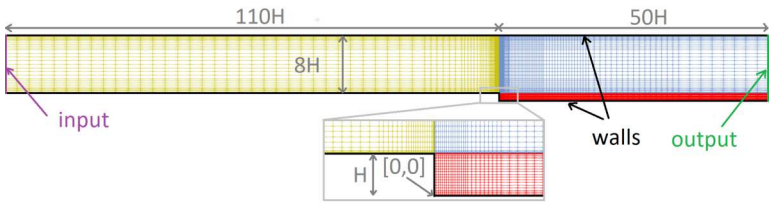

Figure 1. Flow over a backward-facing step—the illustrative problem description and computational mesh.

The experiment performed by Driver and Seegmiller [13] was conducted with such a test configuration that their data can be compared to the numerical results of the incompressible fluid flow in 2D, see, e.g., [23]. Although the unsteady problem (2.1) is solved in our computations, we assume time discretization as an iterative approach to determine a steady-state solution, which is used for comparison with the experimental data in [13], NASA computational SSTm results [25] and numerical results obtained by OpenFOAM [33].

The Dirichlet boundary conditions are set according to Table 1, the outflow conditions are defined in (2.1) and (2.2). The inlet boundary conditions for the velocity $\bar{\mathbf{u}}$, turbulent kinetic energy k and specific turbulent dissipation ω are specified by profiles. The velocity profile is expressed as

$$\bar{u}_2(x,y) = \bar{u}_{\max}\Big(1 - \Big(\frac{|y-\overline{y}|}{Y}\Big)^{1/\bar{n}}\Big), \tag{7.1}$$

where $\bar{u}_{\max}$ is the maximal value of the velocity profile which equals to $\bar{u}_\infty$ (given in Table 1) in our experiment, $\overline{y} = 5H$ is the y-coordinate of the midpoint of the inlet channel, $Y = 4H$ is half of the inlet channel height and $\bar{n}$ is set such that the velocity profile determined by (7.1) corresponds to the velocity profile from the OpenFOAM validation experiment, i.e., $\bar{n} = 1000$ is used in the experiments. The expression (7.1) is used analogously for k and ω profiles with the maximum (free-stream) values specified in Table 1 to avoid the discontinuous boundary conditions of k and ω at the corners where the Dirichlet boundary of the inlet and walls meet and thus the corresponding functions g^k and g^ω in (2.2) lie in $H^{1/2}(\partial\Omega_D)$. The values k_∞ and ω_∞ are again set according to the OpenFOAM validation experiment and its parameter setting. The wall boundary condition for ω is approximated using the expression (cf., e.g., [41], [11])

$$\omega = \frac{6\nu}{\beta y_1^2}, \tag{7.2}$$

where y_1 is the normal distance of the grid point nearest to the wall.

	Inlet boundary	Walls	Initial condition
Velocity	$\bar{u}_\infty = 44.2$	$[0, 0]$	$\bar{\mathbf{u}}_0(\mathbf{x}) = [0, 0]$
k	$k_\infty = 0.00109$	0	$k_0(\mathbf{x}) = 0.00109$
ω	$\omega_\infty = 181728$	$3.44 \cdot 10^6$	$\omega_0(\mathbf{x}) = 181728$

Table 1. Initial and boundary conditions of the backward-facing step flow problem.

	SRBAV1-SRBAV0	TCSD-SRBAV0
RANS	SRBAV with $\alpha = 1$	T-CSD
turbulence model	SRBAV with $\alpha = 0$	SRBAV with $\alpha = 0$
	SRBAV1-SUPG	SRBAV1-SOLD
RANS	SRBAV with $\alpha = 1$	SRBAV with $\alpha = 1$
turbulence model	SUPG	SUPG with additional crosswind

Table 2. Examples of chosen stabilization schemes for RANS and turbulence model equations and their abbreviated notation.

8. Results and discussion

In this section, we present the numerical results obtained by our in-house IgA-based fluid flow solver for the model problem of flow over a backward-facing step for stabilization techniques mentioned in Section 4 and our new stabilization scheme proposed in Section 5, and we compare them with the experimental data from Driver and Seegmiller in [13], NASA reference solution provided in [25] and numerical results obtained by the prepared validation backward-facing step example in OpenFOAM using SimpleFOAM solver according to [33].

Our in-house IgA-based fluid flow solver is implemented in C++ in the framework of G+Smo (Geometry and Simulation modules) library, see [27]. We tested several different inf-sup stable pairs of B-spline discretization spaces, $\mathbf{S}_0^2 \times S_0^1$ (i.e., linear basis functions for pressure and quadratic for velocity, which is close to the standard FEM), $\mathbf{S}_1^3 \times S_1^2$ (i.e., quadratic basis functions for pressure and cubic for velocity, with C^1 continuity across the elements) and $\mathbf{S}_2^4 \times S_2^3$ (i.e., cubic basis functions for pressure and quartic for velocity, with C^2 continuity across the elements).

For all these pairs of B-spline discretization spaces, we run the simulations on the B-spline mesh shown in illustrative Figure 1 satisfying $y^+ \approx 0.85$, where $y^+ = u_\tau y_1/\nu$ is the dimensionless wall distance of a grid point nearest to the wall and u_τ is the friction velocity. The mesh is composed of three conforming patches with 12439 B-spline elements in total. The corresponding numbers of degrees of freedom are 111403 for $\mathbf{S}_0^2 \times S_0^1$, 113552 for $\mathbf{S}_1^3 \times S_1^2$ and 115719 for $\mathbf{S}_2^4 \times S_2^3$ discretizations.

The discrete RANS problem is solved decoupled from the discrete turbulence model. In the first time step, the linearized RANS problem is solved until iterations of the numerical solution of $\bar{\mathbf{u}}$ and $\bar{p}$ converge (or until the maximum number of the Picard iterations is achieved). Then, we continue solving the discrete turbulence model such that the computed velocity and pressure solutions are used for the evaluation of turbulence model terms. This sequence is repeated for each time step. As already mentioned, time steps serve only as an iterative process converging to a steady-state solution.

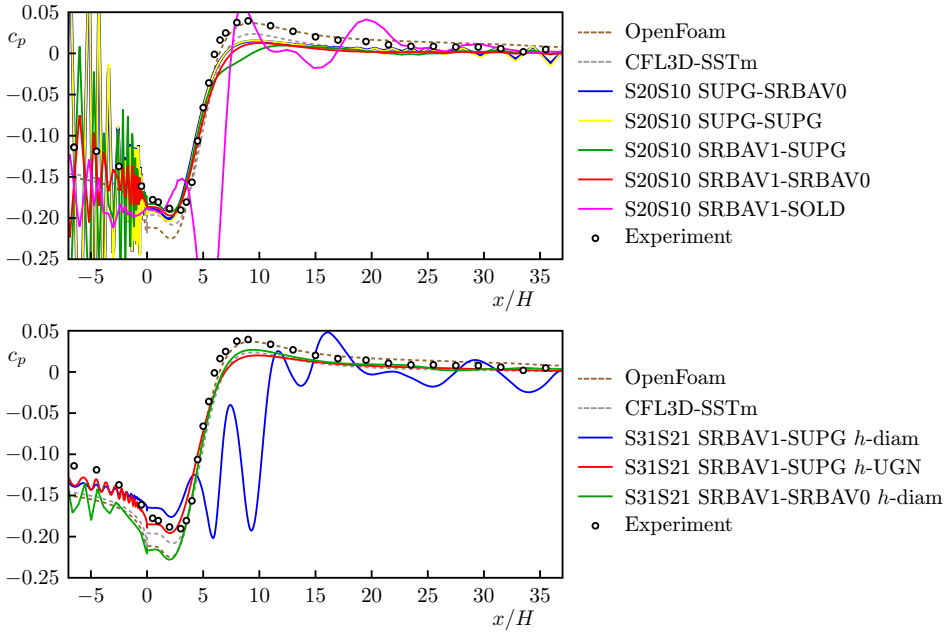

Figure 2. Comparison of the pressure coefficient c_p at the lower wall obtained for various stabilization methods, where S20S10 stands for $\mathbf{S}_0^2 \times S_0^1$ (top), and comparison of using different element length h (bottom), where S31S21 stands for $\mathbf{S}_1^3 \times S_1^2$.

In the experiments in this section, we choose the same stabilization parameter τ_{S} from (6.1) without dependence on a time step, i.e., $\tau_{\mathrm{S}} = ((2|\mathbf{b}|/h)^2 + 9(4\varepsilon/h^2)^2 + |r|^2)^{-1/2}$ for all stabilizations, where h is the direction-dependent element length (6.3). Let us note that we use abbreviations for employed stabilization schemes for RANS and turbulence model equations in the following, such as SRBAV1-SRBAV0 if SRBAV stabilization scheme with the parameter $\alpha = 1$ is used for RANS and $\alpha = 0$ is used for turbulence model equations, see Table 2 for other examples of used abbreviations. The factors C_1 and C_2 in (5.2) are chosen so that $C_1 = 1/H^\alpha$ for both discrete RANS and turbulence model equations and $C_2 = H/\bar{u}_\infty^2$ for the RANS equations, $C_2 = H/\bar{u}_\infty^3$ for the k-equation and $C_2 = (H/\bar{u}_\infty)^2$ for the ω-equation.

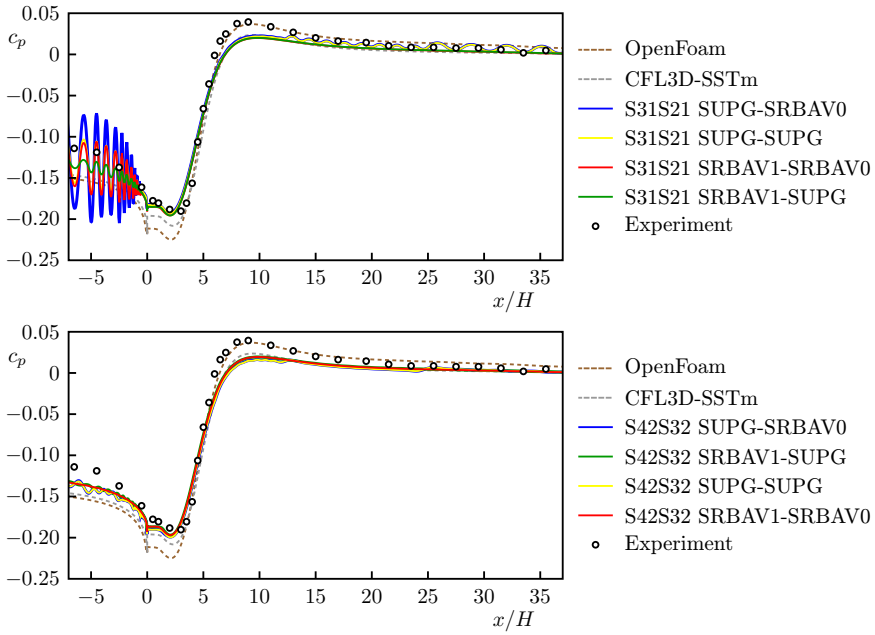

Figure 3. Comparison of the pressure coefficient c_p at the lower wall obtained for various stabilization methods, where S31S21 stands for $\mathbf{S}_1^3 \times S_1^2$ (top) and S42S32 for $\mathbf{S}_2^4 \times S_2^3$ (bottom).

To achieve a stable iterative process, we first use the T-CSD method for the stabilization of RANS equations and the SRBAV scheme with $\alpha = 0$ for the turbulence model, i.e., TCSD-SRBAV0, and we choose the time step size $\Delta t = 10^{-5}$. The numerical simulation is stopped in $t = 0.01\,\mathrm{s}$ and the result is used as the initial condition for the subsequent computation for the study of various stabilization methods and comparison to experimental and reference data. First, we continue solving the RANS closed problem using $\Delta t = 10^{-5}$ for the next 1000 time steps and then we increase the time step to $\Delta t = 10^{-4}$ and let the iterative process converge to the steady state (if the sequence of iterates is convergent). In the following experiments, we mention only the stabilization schemes, which are used after the initialization step, i.e., the stabilization methods used from the time moment $t = 0.01\,\mathrm{s}$.

In Figures 2 and 3, we present the pressure coefficient c_p at the lower wall obtained for different discretization spaces and various stabilization methods. The pressure coefficient is defined by

$$c_p = \frac{\bar{p} - \bar{p}_\infty}{\frac{1}{2}\varrho \bar{u}_\infty^2}, \tag{8.1}$$

where $\bar{p}_\infty$ and $\bar{u}_\infty$ are free stream values of the computed mean pressure and velocity at $x/H = -4$ and ϱ is the fluid density.

We compare the stabilization schemes proposed in Section 4 and the newly proposed scheme introduced in Section 5 for both RANS and turbulence model equations employing the inf-sup stable B-spline discretization space pair $\mathbf{S}_0^2 \times S_0^1$ (which is close to the standard FEM) in Figure 2 (top) and the discretization spaces $\mathbf{S}_1^3 \times S_1^2$ and $\mathbf{S}_2^4 \times S_2^3$ in Figure 3.

The inner nonlinear iteration process stagnates in some cases, i.e., the relative error of the solution does not decrease for the consecutive steps and the maximum number of inner iterations is reached in every time step. Note that we set maximum of 5 inner iterations, but the stagnation of the nonlinear iteration method does not change even if we increase the maximum number of the inner iterations to 1000. For this reason, we show the numerical results only in $t = 0.12\,\mathrm{s}$ in Figure 2. In Figure 3, we present the results for the discretization spaces $\mathbf{S}_1^3 \times S_1^2$ and $\mathbf{S}_2^4 \times S_2^3$ in the steady state.

The following conclusions can be drawn from the results obtained. Approaches based on low-order methods or high-resolution methods equipped with nonlinear techniques for limiting oscillations (e.g., finite volume TVD flux limiter-based schemes) give the least oscillatory results. These are the methods used in the OpenFOAM and NASA CFL3D packages and the T-CSD scheme. The non-consistent T-CSD scheme is only used to start the calculation (as described above) and is not included in the comparison of results, because we focus on consistent stabilization schemes. Based on the results shown in Figure 2 (top), variants with the SOLD stabilization are excluded from further comparison (they contain significant artificial oscillations even for higher orders of approximation). The results using the stabilization method SRBAV0-SRBAV0 are less satisfactory than for other stabilization methods and also the convergence of the nonlinear iterations is slow in this case. The graphs in Figure 3 suggest that in the case of higher order methods, it can be observed that oscillations decrease with the degree of approximation and higher smoothness of the approximation. This is an advantage of using isogeometric analysis. Regarding the type of stabilization, it appears that in the case of the turbulence model approximation, the results are roughly the same and the type of stabilization does not have a significant effect. In the case of the approximation of the RANS equations, SRBAV stabilizations seem to be more robust than the SUPG stabilization. In the case of our problem, we obtained the best results using the SRBAV1-SUPG and SRBAV1-SRBAV0 stabilizations. From Figure 2 (bottom), it can be concluded that in the case of the problem we have solved the stabilization SRBAV appears to be more robust also with respect to the choice of the stabilization parameter h, e.g., if h is chosen as the element diameter (h-diam in Figure 2 (bottom)) instead of the direction-dependent h (h-UGN in Figure 2 (bottom)).

Finally, we present a comparison of reattachment lengths (in steady-states) for all the discretization spaces and for the best stabilization approaches mentioned in the previous paragraphs, i.e., SRBAV1-SRBAV0, SUPG-SUPG, SRBAV1-SUPG, SUPG-SRBAV0, in Table 3. The point where the skin friction coefficient equals to zero is set as the reattachment point. The skin friction coefficient is defined by

$$c_{\mathrm{f}} = \frac{\tau_{\mathrm{w}}}{\frac{1}{2}\varrho \bar{u}_{\infty}^2}, \tag{8.2}$$

where $\bar{u}_{\infty}$ is again the free stream value of the computed velocity at $x/H = -4$ and τ_{w} is the wall shear stress. It can be concluded that all stabilizations combined with higher degree and continuity basis functions and isogeometric analysis give better estimates of the reattachment length than the OpenFOAM and NASA CFL3D packages.

Together with the previously mentioned observations, SRBAV1-SRBAV0 or SRBAV1-SUPG can be considered as favourable options for the numerical simulation of the turbulent flow over a backward-facing step by solving the RANS equations with turbulence model with the help of isogeometric analysis.

	$\mathbf{S}_0^2 \times S_0^1$	$\mathbf{S}_1^3 \times S_1^2$	$\mathbf{S}_2^4 \times S_2^3$		
	x/H	x/H	x/H		x/H
SRBAV1-SRBAV0	6.26	6.27	6.33	Experiment [13]	6.26 ± 0.1
SUPG-SUPG	6.02	6.25	6.18	NASA [25]	6.5
SRBAV1-SUPG	6.15	6.15	6.19	OpenFOAM [33]	6.39
SUPG-SRBAV0	5.96	6.27	6.26		

Table 3. Comparison of the reattachment length.

9. Summary and conclusions

In this paper, we have provided a study of linear and nonlinear stabilization methods for RANS equations and SST k-ω turbulence model and high Reynolds numbers in combination with isogeometric analysis.

We have proposed a new stabilization method, which we call the streamline residual based artificial viscosity. It is consistent, genuinely nonlinear and the numerical diffusion is added only in the direction of the flow.

The classical flow over a backward-facing step was considered in two dimensions as a model problem in order to investigate the behaviour of the stabilized isogeometric discretization. We have presented numerical solutions obtained by our in-house incompressible fluid flow solver based on isogeometric analysis and we have compared our results with experimental data and reference numerical solutions.

The test computations we have presented show that the flow over a backward-facing step problem was successfully solved using the new SRBAV stabilization. Also, our experiments have shown that isogeometric analysis based on the inf-sup stable B-spline discretization space pairs of higher degrees and continuities stabilized by SRBAV1-SRBAV0, SRBAV1-SUPG can provide results which match the experimental data and reference solutions available in the literature very well. Moreover, the results indicate that oscilations of the numerical solution are reduced using the SRBAV stabilization for discretization spaces of higher degree and continuity, which are typical for IgA.

A c k n o w l e d g e m e n t. We thank to all referees for their valuable comments which helped us to improve the paper.

References

[1] *G. R. Barrenechea, V. John, P. Knobloch*: A local projection stabilization finite element method with nonlinear crosswind diffusion for convection-diffusion-reaction equations. ESAIM, Math. Model. Numer. Anal. *47* (2013), 1335–1366. zbl MR doi

[2] *B. Bastl, M. Brandner, J. Egermaier, H. Horníková, K. Michálková, E. Turnerová*: Numerical simulation of lid-driven cavity flow by isogeometric analysis. Acta Polytech., Pr. ČVUT Praha *61* (2021), 33–48. doi

[3] *B. Bastl, M. Brandner, J. Egermaier, K. Michálková, E. Turnerová*: IgA-Based Solver for turbulence modelling on multipatch geometries. Adv. Eng. Softw. *113* (2017), 7–18. doi

[4] *Y. Bazilevs, V. M. Calo, J. A. Cottrell, T. J. R. Hughes, A. Reali, G. Scovazzi*: Variational multiscale residual-based turbulence modeling for large eddy simulation of incompressible flows. Comput. Methods Appl. Mech. Eng. *197* (2007), 173–201. zbl MR doi

[5] *Y. Bazilevs, V. M. Calo, T. E. Tezduyar, T. J. R. Hughes*: $YZ\beta$ discontinuity capturing for advection-dominated processes with application to arterial drug delivery. Int. J. Numer. Methods Fluids *54* (2007), 593–608. zbl MR doi

[6] *A. Bressan, G. Sangalli*: Isogeometric discretizations of the Stokes problem: Stability analysis by the macroelement technique. IMA J. Numer. Anal. *33* (2013), 629–651. zbl MR doi

[7] *N. A. Brooks, T. J. R. Hughes*: Streamline upwind/Petrov-Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equations. Comput. Methods Appl. Mech. Eng. *32* (1982), 199–259. zbl MR doi

[8] *A. Buffa, G. Sangalli, R. Vázquez*: Isogeometric analysis in electromagnetics: B-splines approximation. Comput. Methods Appl. Mech. Eng. *199* (2010), 1143–1152. zbl MR doi

[9] *E. Burman, A. Ern*: Nonlinear diffusion and discrete maximum principle for stabilized Galerkin approximations of the convection-diffusion-reaction equation. Comput. Methods Appl. Mech. Eng. *191* (2002), 3833–3855. zbl MR doi

[10] *N. Collier, D. Pardo, L. Dalcin, M. Paszynski, V. M. Calo*: The cost of continuity: A study of the performance of isogeometric finite elements using direct solvers. Comput. Methods Appl. Mech. Eng. *213-216* (2012), 353–361. zbl MR doi

[11] *P. A. Davidson*: Turbulence: An Introduction for Scientists and Engineers. Oxford University Press, Oxford, 2004. zbl MR doi

[12] *J. Donea, A. Huerta*: Finite Element Methods for Flow Problems. John Wiley & Sons, Chichester, 2003. doi

[13] *D. M. Driver, H. L. Seegmiller*: Features of a reattaching turbulent shear layer in divergent channel flow. AIAA J. *23* (1985), 163–171. doi

[14] *H. C. Elman, D. J. Silvester, A. J. Wathen*: Finite Elements and Fast Iterative Solvers: With Applications in Incompressible Fluid Dynamics. Numerical Mathematics and Scientific Computation. Oxford University Press, Oxford, 2014. zbl MR doi

[15] *A. Falini, J. Špeh, B. Jüttler*: Planar domain parameterization with THB-splines. Comput. Aided Geom. Des. *35-36* (2015), 95–108. zbl MR doi

[16] *R. R. Hiemstra, G. Sangalli, M. Tani, F. Calabrò, T. J. R. Hughes*: Fast formation and assembly of finite element matrices with application to isogeometric linear elasticity. Comput. Methods Appl. Mech. Eng. *355* (2019), 234–260. zbl MR doi

[17] *M.-C. Hsu, Y. Bazilevs, V. M. Calo, T. E. Tezduyar, T. J. R. Hughes*: Improving stability of stabilized and multiscale formulations in flow simulations at small time steps. Comput. Methods Appl. Mech. Eng. *199* (2010), 828–840. zbl MR doi

[18] *T. J. R. Hughes, J. A. Cottrell, Y. Bazilevs*: Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement. Comput. Methods Appl. Mech. Eng. *194* (2005), 4135–4195. zbl MR doi

[19] *T. J. R. Hughes, A. Reali, G. Sangalli*: Duality and unified analysis of discrete approximations in structural dynamics and wave propagation: Comparison of p-method finite elements with k-method NURBS. Comput. Methods Appl. Mech. Eng. *197* (2008), 4104–4124. zbl MR doi

[20] *V. John, P. Knobloch*: On spurious oscillations at layers diminishing (SOLD) methods for convection-diffusion equations. I. A review. Comput. Methods Appl. Mech. Eng. *196* (2007), 2197–2215. zbl MR doi

[21] *V. John, P. Knobloch*: On spurious oscillations at layers diminishing (SOLD) methods for convection-diffusion equations. II. Analysis for P_1 and Q_1 finite elements. Comput. Methods Appl. Mech. Eng. *197* (2008), 1997–2014. zbl MR doi

[22] *V. John, E. Schmeyer*: Finite element methods for time-dependent convection-diffusion-reaction equations with small diffusion. Comput. Methods Appl. Mech. Eng. *198* (2008), 475–494. zbl MR doi

[23] *J.-Y. Kim, A. J. Ghajar, C. Tang, G. L. Foutch*: Comparison of near-wall treatment methods for high Reynolds number backward-facing step flow. Int. J. Comput. Fluid Dyn. *19* (2005), 493–500. zbl doi

[24] *D. Kuzmin, O. Mierka, S. Turek*: On the implementation of the $k-\varepsilon$ turbulence model in incompressible flow solvers based on a finite element discretization. Int. J. Comput. Sci. Math. *1* (2007), 193–206. zbl MR doi

[25] *Langley Research Center*: NASA Turbulence Modeling Resource: 2DBFS: 2D Backward Facing Step. Available at `https://turbmodels.larc.nasa.gov/backstep_val.html` (2021).

[26] *R. Li, Q. Wu, S. Zhu*: Proper orthogonal decomposition with SUPG-stabilized isogeometric analysis for reduced order modelling of unsteady convection-dominated convection-diffusion-reaction problems. J. Comput. Phys. *387* (2019), 280–302. zbl MR doi

[27] *A. Mantzaflaris, B. Jüttler*: G+Smo (Geometry plus Simulation Modules) v0.8.1. Available at `https://github.com/gismo` (2018). sw

[28] *F. Moukalled, L. Mangani, M. Darwish*: The Finite Volume Method in Computational Fluid Dynamics: An Advanced Introduction with OpenFOAM and Matlab. Fluid Mechanics and Its Applications 113. Springer, Cham, 2016. zbl MR doi

[29] *M. Nazarov*: Convergence of a residual based artificial viscosity finite element method. Comput. Math. Appl. *65* (2013), 616–626. zbl MR doi

[30] *M. Nazarov, J. Hoffman*: Residual-based artificial viscosity for simulation of turbulent compressible flow using adaptive finite element methods. Int. J. Numer. Methods Fluids *71* (2013), 339–357. zbl MR doi
[31] *R. H. Nichols*: Turbulence Models and Their Application to Complex Flows: Revision 4.01. University of Alabama at Birmingham, Birmingham, 2014.
[32] *K. Nordanger, R. Holdahl, A. M. Kvarving, A. Rasheed, T. Kvamsdal*: Implementation and comparison of three isogeometric Navier-Stokes solvers applied to simulation of flow past a fixed 2D NACA0012 airfoil at high Reynolds number. Comput. Methods Appl. Mech. Eng. *284* (2015), 664–688. zbl MR doi
[33] *OpenCFD (ESI Group)*: OpenFOAM: User Guide v2112: The open source CFD toolbox. Available at `https://www.openfoam.com/documentation/guides/latest/doc/index.html` (2019). sw
[34] *Y. Otoguro, K. Takizawa, T. E. Tezduyar*: Element length calculation in B-spline meshes for complex geometries. Comput. Mech. *65* (2020), 1085–1103. zbl MR doi
[35] *L. Piegl, W. Tiller*: The *NURBS* Books. Monographs in Visual Communication. Springer, Berlin, 1997. zbl doi
[36] *H.-G. Roos, M. Stynes, L. Tobiska*: Robust Numerical Methods for Singularly Perturbed Differential Equations: Convection-diffusion-reaction and Flow Problems. Springer Series in Computational Mathematics. Springer, Berlin, 2008. zbl MR doi
[37] *F. Shakib, T. J. R. Hughes*: A new finite element formulation for computational fluid dynamics. IX. Fourier analysis of space-time Galerkin/least-squares algorithms. Comput. Methods Appl. Mech. Eng. *87* (1991), 35–58. zbl MR doi
[38] *A. Tagliabue, L. Dedè, A. Quarteroni*: Isogeometric analysis and error estimates for high order partial differential equations in fluid dynamics. Comput. Fluids *102* (2014), 277–303. zbl MR doi
[39] *K. Takizawa, T. E. Tezduyar, Y. Otoguro*: Stabilization and discontinuity-capturing parameters for space-time flow computations with finite element and isogeometric discretizations. Comput. Mech. *62* (2018), 1169–1186. zbl MR doi
[40] *T. E. Tezduyar*: Finite element in fluids: Stabilized formulations and moving boundaries and interfaces. Comput. Fluids *36* (2007), 191–206. zbl MR doi
[41] *H. K. Versteeg, W. Malalasekera*: An Introduction to Computational Fluid Dynamics: The Finite Volume Method. Pearson Education, Harlow, 2007.
[42] *D. C. Wilcox*: Turbulence Modeling for CFD. DCW Industries, La Canada, 2006.
[43] *H. Zhang, T. Craft, H. Iacovides*: The formulation of the RANS equations for hypersonic turbulent flows. Proceedings of the 5th World Congress on Mechanical, Chemical, and Material Engineering (MCM'19). Avestia Publishing, Orléans, 2019, pp. Article ID 172, 9 pages. doi

Authors' address: *Bohumír Bastl*, *Marek Brandner*, *Kristýna Slabá*, *Eva Turnerová* (corresponding author), University of West Bohemia, Univerzitní 2732/8, 301 00 Plzeň, Czech Republic, e-mail: `bastl@kma.zcu.cz`, `brandner@kma.zcu.cz`, `kslaba@ntis.zcu.cz`, `turnerov@kma.zcu.cz`.