Applied and Computational Mechanics 16 (2022) 5–22

Applied and Computational Mechanics

Curvilinear element of the discontinuous Galerkin method designed to capture the labyrinth seal geometry exactly

O. Bublík[a,*], A. Pecka[a], J. Vimmr[a]

[a] *Faculty of Applied Sciences, University of West Bohemia, Univerzitní 8, 301 00 Plzeň, Czech Republic*

Received 17 December 2021; accepted 6 June 2022

Abstract

The present study applies the discontinuous Galerkin finite element method to a numerical simulation of a compressible fluid flow through a labyrinth seal. This paper is proposes a curvilinear hexahedral element, which is deformed in such a way that it matches the rotated walls of the labyrinth seal exactly. A numerical study is performed on the staggered labyrinth seal with two teeth on the rotor and one tooth on the stator. For numerical simulation, three computational meshes with different refinement are considered. All of the numerical simulations are performed for both stationary rotor and for rotor rotating at 50 Hz. The obtained numerical results are compared with results computed by the commercial CFD software Ansys Fluent.

Keywords: labyrinth seal, discontinuous Galerkin method, curvilinear element

1. Introduction

The design of steam turbines with the basic subsystems has not changed substantially for over a century. The research and development in this area has been focused mainly on improving safety and efficiency and on enhancing the overall performance by working with higher and higher pressure ratios. These improvements have been made partly thanks to computer simulations and CFD simulations in particular. The turbine performance is heavily influenced by the design of seals. The amount of steam leaking around the edge of the blades instead of passing through them is critical to performance and thermodynamic efficiency. An overview of sealing in turbomachinery can be found for instance in [5].

Labyrinth seals are a common type of sealing in turbomachinery. Such a seal creates a complicated path for the fluid to take in order to reduce leakage. A labyrinth seal consists of a number of cavities wherein eddies are formed. The primary purpose of these eddies is to dissipate energy thereby increasing the resistance of the flow. Labyrinth seals are ideal for sealing the tip of turbine blades because of a high pressure ratios between the stages and high rotational speed of the rotor. Other types of seals might not withstand such conditions. Although labyrinth seals do not prevent leakage completely because of the clearance between the stator and rotor, for the same reason they are durable and reliable and resist high temperatures and pressure ratios.

There are a number of geometries of labyrinth seals, including straight-through, staggered, stepped and combinations. Each type of labyrinth seal has multiple parameters that can be adjusted, such as clearance size and height, width, pitch and angle of inclination of the tooth. These parameters have a drastic effect on the efficiency, lifespan and stability of the turbine. For instance, large clearance leads to loses in power output and may cause instabilities. On the other hand, clearance which is too small may, under some conditions, result in rubbing and

*Corresponding author. Tel.: +420 377 632 398, e-mail: obublik@kme.zcu.cz.
https://doi.org/10.24132/acm.2022.732

cause damage. For this reason, flow in labyrinth seals and its influence on the turbine has been extensively studied.

The oldest method for estimating the leakage through a labyrinth seal was developed by Martin [10] in 1908. He came up with an expression for quantifying the leakage of an ideal straight seal. His finding was generalised by many authors, see literature survey [14]. Another analytical approach for modelling labyrinth seals was developed by Iwatsubo [8] and improved by Childs and Scharrer [4]. This empirical model is called the bulk-flow model and is based on a control volume approach. It involves solving the continuity and circumferential momentum equations for each cavity. It is used to analyse instabilities that may arise due to flow inside labyrinth seals.

Plenty of experimental and numerical studies that are concerned with fine-tuning of labyrinth geometries have been published throughout the years. Stocker [16], for example, designed several step seals in an attempt to increase turbulence in cavities and performed numerous experiments to evaluate their efficiency for gas turbines with high pressure ratios. Wittig et al. [19] conducted one of the earliest studies which explored possibilities of CFD computing for the labyrinth seal simulation. In particular, they performed CFD simulations for various pressure ratios and geometrical parameters of both straight-through and stepped seals with a finite-difference based compressible solver with k-ε turbulent model and validated the solver with their experimental setup. To name more recent examples, Hur et al. [7], Kang et al. [9] and Zhang et al. [21] all investigated the dependence of the clearance size and other tooth parameters on the leakage characteristics in stepped labyrinth seal both numerically and experimentally. All of the three studies considered only a 2D flow in a cross section of a radially symmetric labyrinth seal, with they computed using commercial CFD solvers based on finite volume approach. There are also studies, which model the flow in the labyrinth seal as fully three dimensional, see for example [11] or [20] and references therein.

The majority of numerical studies on fluid flow in labyrinth seals make use of traditional discretization methods, such as finite volume method, and, in most cases, they use commercial CFD software. The aim of this study is to apply a less traditional high-order method that, also thanks to the use of curvilinear elements, gives similar results as the traditional methods when using much coarser mesh. In particular, the discontinuous Galerkin (DG) finite element method is used for the numerical simulation of a fluid flow in labyrinth seals. The advantage of the DG method is its stability, robustness, low artificial damping and the ability to achieve a high-order of spatial accuracy, see e.g. [3, 13, 18]. The key part of this study is the transformation of the reference element onto the physical element in such a way, that it matches the geometry of the labyrinth seal exactly. Because of this, considerably fewer elements can be used in the circumferential direction without a significant loss of accuracy. The motivation for the development of this particular curvilinear element are applications where the rotor is undergoing a precession. These problems can be solved in two different ways. The first option is to prescribe a motion to the rotor, in which case, the governing equations are solved using the arbitrary Lagrangian-Eulerian method [2, 12]. The second option is to change the perspective from a stationary frame of reference to the non-inertial rotating reference frame that follows the precession of the rotor. The presented curvilinear element is especially suitable for the latter approach, where the labyrinth seal is not rotationally symmetric. This paper presents a use of the curvilinear element only for rotationally symmetric labyrinth seals for simplicity of the numerical study.

In this paper, six simulations on a chosen labyrinth seal geometry with three different meshes and two rotor speeds is performed. The results are compared with the results obtained by the

commercial CFD software Ansys Fluent. A smaller element size in the circumferential direction is used for Ansys Fluent, since its solver uses a traditional finite-volume approach without curvilinear elements. Due to numerical difficulties with the k-ε and k-ω turbulence models in combination with the discontinuous Galerkin method, Spalart–Allmaras turbulence model was used instead. Although this turbulence model in not suitable for this particular problem, the authors believe that the Spalart–Allmaras turbulence model is sufficient for demonstrating the idea of the presented numerical approach with curvilinear elements. The same turbulence model was also used in the case of Ansys Fluent so that the comparison between the two solvers is valid.

2. Mathematical model

The system of Favre-averaged Navier-Stokes equations is considered as the underlying mathematical model. The system can be written in dimensionless formulation as follows

$$\begin{aligned} \frac{\partial \bar{\varrho}}{\partial t} + \frac{\partial(\bar{\varrho}\tilde{u}_i)}{\partial x_i} &= 0, \\ \frac{\partial}{\partial t}(\bar{\varrho}\tilde{u}_j) + \frac{\partial}{\partial x_i}(\bar{\varrho}\tilde{u}_i\tilde{u}_j + \bar{p}\delta_{ij}) &= \frac{\partial}{\partial x_i}(\tilde{\tau}_{ij} + t_{ij}), \qquad j = 1,2,3, \\ \frac{\partial}{\partial t}(\bar{\varrho}\tilde{e}) + \frac{\partial}{\partial x_i}(\bar{\varrho}\tilde{e}\tilde{u}_i + \bar{p}\tilde{u}_i) &= \frac{\partial}{\partial x_i}\left[(\tilde{\tau}_{ij} + t_{ij})\tilde{u}_j - \tilde{q}_i - q_{ti}\right], \end{aligned} \tag{1}$$

where $i = 1,2,3$ is the summation index, $\bar{\varrho}$ and $\bar{p}$ are dimensionless time-averaged values of density and pressure, $\tilde{u}_i$ and $\tilde{e}$ are dimensionless mass-averaged velocity components and energy. The mass-averaged heat flux $\tilde{q}_i$ and viscous stress tensor $\tilde{\tau}_{ij}$ and the turbulent heat flux q_{ti} and the Reynolds stress tensor t_{ij} are given by the following relations

$$\begin{aligned} \tilde{q}_i &= -\frac{\kappa}{\kappa-1}\frac{\mu}{\mathrm{Pr}}\frac{\partial}{\partial x_i}\left(\frac{\bar{p}}{\bar{\varrho}}\right), \qquad \tilde{\tau}_{ij} = \frac{2\mu}{\mathrm{Re}}\bar{S}_{ij}, \\ q_{ti} &= -\frac{\kappa}{\kappa-1}\frac{\mu_t}{\mathrm{Pr}_t}\frac{\partial}{\partial x_i}\left(\frac{\bar{p}}{\bar{\varrho}}\right), \qquad t_{ij} = \frac{2\mu_t}{\mathrm{Re}}\bar{S}_{ij}, \end{aligned} \tag{2}$$

with

$$\bar{S}_{ij} = \frac{1}{2}\left(\frac{\partial\tilde{u}_i}{\partial x_j} + \frac{\partial\tilde{u}_j}{\partial x_i}\right) - \frac{1}{3}\delta_{ij}\frac{\partial\tilde{u}_k}{\partial x_k}. \tag{3}$$

Here, Re is the Reynolds number, Pr is the Prandtl number, Pr_t is the turbulent Prandtl number, which is chosen as $\mathrm{Pr}_t = 0.89$, and μ and μ_t are regular and turbulent dynamic viscosities. The equation of state for ideal gas is considered as

$$\bar{p} = (\kappa-1)\left(\bar{\varrho}\tilde{e} - \frac{1}{2}\bar{\varrho}\tilde{u}_i\tilde{u}_i\right). \tag{4}$$

To include the influence of turbulent fluctuations on the mean flow the one-equation turbulence model of Spalart and Allmaras [15] is used. Note that this model neglects turbulent kinetic energy, therefore this quantity was set to zero in the above equations. The equation governing the transportation, dissipation and generation of viscosity-like quantity $\tilde{\nu}$ is in the dimensionless

form given by

$$\frac{\partial(\bar{\varrho}\tilde{\nu})}{\partial t}+\frac{\partial(\bar{\varrho}\tilde{\nu}\tilde{u}_j)}{\partial x_j}=\frac{1}{\sigma\,\mathrm{Re}}\left[\frac{\partial}{\partial x_j}\left((\mu+\bar{\varrho}\tilde{\nu})\frac{\partial\tilde{\nu}}{\partial x_j}\right)+\bar{\varrho}c_{b2}\frac{\partial\tilde{\nu}}{\partial x_j}\frac{\partial\tilde{\nu}}{\partial x_j}\right]+ c_{b1}\bar{\varrho}\tilde{S}\tilde{\nu}-\frac{1}{\mathrm{Re}}c_{w1}f_w\frac{1}{\bar{\varrho}}\left(\frac{\bar{\varrho}\tilde{\nu}}{D}\right)^2 \tag{5}$$

and completed by the following relations

$$\mu_t=\bar{\varrho}\tilde{\nu}f_{v1},\quad \tilde{S}=\bar{\Omega}+\frac{1}{\mathrm{Re}}\frac{\tilde{\nu}}{\kappa^2D^2}f_{v2},\quad f_w=g\left(\frac{1+c_{w3}^6}{g^6+c_{w3}^6}\right)^{\frac{1}{6}},\quad g=r+c_{w2}\left(r^6-r\right),$$

$$r=\min\left(\frac{\tilde{\nu}}{\mathrm{Re}\,\tilde{S}\kappa^2D^2},10\right),\quad f_{v1}=\frac{\chi^3}{\chi^3+c_{v1}^3},\quad \chi=\frac{\tilde{\nu}}{\nu},\quad f_{v2}=1-\frac{\chi}{1+\chi f_{v1}}$$

and constants $c_{b1}=0.135\,5$, $c_{b2}=0.622$, $\sigma=\frac{2}{3}$, $\kappa=0.41$, $c_{w1}=\frac{c_{b1}}{\kappa^2}+\frac{1+c_{b2}}{\sigma}$, $c_{w2}=0.3$, $c_{w3}=2.0$, $c_{v1}=7.1$. Here $\nu=\mu/\bar{\varrho}$ is the kinematic viscosity, D is the distance to the nearest wall and $\bar{\Omega}$ is the vorticity magnitude.

The system of Favre-averaged Navier-Stokes equations (1) and the transport equation (5) constituted one coupled system, which can be written in the following compact vector form

$$\frac{\partial\boldsymbol{w}}{\partial t}+\frac{\partial}{\partial x_i}\boldsymbol{f}_i(\boldsymbol{w},\nabla\boldsymbol{w})=\boldsymbol{p}(\boldsymbol{w},\nabla\boldsymbol{w}), \tag{6}$$

where $\boldsymbol{w}$ is the vector of conservative variables, $\boldsymbol{p}$ is the production term and $\boldsymbol{f}_i$ is the total flux, which consist of the inviscid (Euler) flux $\boldsymbol{f}_i^E$ and the viscous flux $\boldsymbol{f}_i^V$ as follows

$$\boldsymbol{f}_i(\boldsymbol{w},\nabla\boldsymbol{w})=\boldsymbol{f}_i^E(\boldsymbol{w})+\boldsymbol{f}_i^V(\boldsymbol{w},\nabla\boldsymbol{w}). \tag{7}$$

In three dimensions the vectors read

$$\boldsymbol{w}=\begin{bmatrix}\bar{\varrho}\\ \bar{\varrho}\tilde{u}_1\\ \bar{\varrho}\tilde{u}_2\\ \bar{\varrho}\tilde{u}_3\\ \bar{\varrho}\tilde{e}\\ \bar{\varrho}\tilde{\nu}\end{bmatrix},\quad \boldsymbol{f}_i^E(\boldsymbol{w})=\begin{bmatrix}\bar{\varrho}\tilde{u}_i\\ \bar{\varrho}\tilde{u}_i\tilde{u}_1+\delta_{i1}\bar{p}\\ \bar{\varrho}\tilde{u}_i\tilde{u}_2+\delta_{i2}\bar{p}\\ \bar{\varrho}\tilde{u}_i\tilde{u}_3+\delta_{i3}\bar{p}\\ \tilde{u}_i(\bar{\varrho}\tilde{e}+\bar{p})\\ \bar{\varrho}\tilde{\nu}\tilde{u}_i\end{bmatrix},\quad \boldsymbol{f}_i^V(\boldsymbol{w},\nabla\boldsymbol{w})=\begin{bmatrix}0\\ \tilde{\tau}_{i1}+t_{i1}\\ \tilde{\tau}_{i2}+t_{i2}\\ \tilde{\tau}_{i3}+t_{i3}\\ (\tilde{\tau}_{ij}+t_{ij})\tilde{u}_j-\tilde{q}_i-q_{ti}\\ \frac{\mu+\bar{\varrho}\tilde{\nu}}{\sigma\,\mathrm{Re}}\frac{\partial\tilde{\nu}}{\partial x_j}\end{bmatrix},$$

$$\boldsymbol{p}(\boldsymbol{w},\nabla\boldsymbol{w})=\begin{bmatrix}0\\ 0\\ 0\\ 0\\ 0\\ \frac{1}{\sigma\,\mathrm{Re}}\bar{\varrho}c_{b2}\frac{\partial\tilde{\nu}}{\partial x_i}\frac{\partial\tilde{\nu}}{\partial x_i}+c_{b1}\bar{\varrho}\tilde{S}\tilde{\nu}-\frac{1}{\mathrm{Re}}c_{w1}f_w\frac{1}{\bar{\varrho}}\left(\frac{\bar{\varrho}\tilde{\nu}}{D}\right)^2\end{bmatrix}.$$

3. Spatial discretization

The discontinuous Galerkin (DG) method [3, 13, 18] is a very stable and robust solution for a compressible fluid flow modelling. In the field of CFD, it is the main competitor for the finite volume method – still the most commonly used method in the industry. The DG method

naturally offers an arbitrary order of accuracy by choosing basis polynomial of appropriate order. In this section, spatial discretization of the system of compressible Navier-Stokes equations (6) is performed using the DG method.

Let $\mathcal{T} = \{\Omega_1, \Omega_2, \ldots, \Omega_K\}$ be the partition of the computational domain $\Omega \subset \mathcal{R}^3$, that is $\bigcup_k \overline{\Omega}_k = \overline{\Omega}$ and $\bigcap_k \Omega_k = \emptyset$. The solution of the system (6) is considered to be an element of the space of vector functions $\boldsymbol{S}_h = [S_h]^6$, where

$$S_h = \{w \in L^2(\Omega) \colon w|_{\Omega_k} \in P^q(\Omega_k) \quad \forall \Omega_k \in \mathcal{T}\}, \tag{8}$$

and $P^q(\Omega_k)$ is the space of polynomial of degree up to q on Ω_k. Note that discontinuities on the boundary of elements are allowed. Taking the dot product of the system of equations (6) with a test function $\boldsymbol{v} \in \boldsymbol{S}_h$, integrating it over an element $\Omega_k \in \mathcal{T}$, applying the divergence theorem and substituting the physical flux f by a numerical flux $\mathcal{F}$ in the surface integral, the following integral identity is obtained

$$\int_{\Omega_k} \frac{\partial \boldsymbol{w}}{\partial t} \cdot \boldsymbol{v} \,\mathrm{d}\Omega - \int_{\Omega_k} \boldsymbol{f}_i(\boldsymbol{w}, \nabla \boldsymbol{w}) \cdot \frac{\partial \boldsymbol{v}}{\partial x_i} \,\mathrm{d}\Omega + \oint_{\partial\Omega_k} \mathcal{F}(\boldsymbol{w}^{\pm}, \nabla \boldsymbol{w}^{\pm}, \boldsymbol{n}) \cdot \boldsymbol{v}^{-} \,\mathrm{d}\Gamma = \int_{\Omega_k} \boldsymbol{p}(\boldsymbol{w}, \nabla \boldsymbol{w}) \cdot \boldsymbol{v} \,\mathrm{d}\Omega. \tag{9}$$

Here $\boldsymbol{n}$ is the unit outward normal to the boundary $\partial\Omega_k$. The left- and right-hand side limit values of a function say ψ are denoted by ψ^- and ψ^+, respectively. In the case of the DG method, discontinuities are allowed on the boundary of each element Ω_k. Therefore, only the limit values of any test function $\boldsymbol{v}$ and the solution $\boldsymbol{w}$ are defined on $\partial\Omega_k$. For this reason, the inviscid and viscous normal physical flux is approximated by an inviscid and viscous numerical flux on $\partial\Omega_k$, respectively, that is

$$\begin{aligned} \boldsymbol{f}_i^E(\boldsymbol{u})\, n_i &\approx \mathcal{F}_{\text{in}}^E(\boldsymbol{u}^{\pm}, \boldsymbol{n}), \\ \boldsymbol{f}_i^V(\boldsymbol{u}, \nabla \boldsymbol{u})\, n_i &\approx \mathcal{F}_{\text{in}}^V(\boldsymbol{u}^{\pm}, \nabla \boldsymbol{u}^{\pm}, \boldsymbol{n}). \end{aligned}$$

The Lax-Friedrichs flux [17] is chosen as the inviscid numerical flux $\mathcal{F}_E$. In order to approximate the normal viscous flux, the interior penalty method is used [6]. Analogically to the physical flux, we define the total numerical flux as $\mathcal{F}_{\text{in}} = \mathcal{F}_{\text{in}}^E - \mathcal{F}_{\text{in}}^V$. In order to account for the boundary conditions, we define the numerical flux $\mathcal{F}$ as

$$\mathcal{F}(\boldsymbol{u}^{\pm}, \nabla \boldsymbol{u}^{\pm}, \boldsymbol{n}) = \begin{cases} \mathcal{F}_{\text{in}}(\boldsymbol{u}^{\pm}, \nabla \boldsymbol{u}^{\pm}, \boldsymbol{n}) & \text{if } \boldsymbol{x} \in \Omega, \\ \mathcal{F}_{\text{b}}(\boldsymbol{u}^{-}, \nabla \boldsymbol{u}^{-}, \boldsymbol{n}) & \text{if } \boldsymbol{x} \in \partial\Omega, \end{cases} \tag{10}$$

where $\mathcal{F}_{\text{b}}$ is the boundary flux which is given by the boundary conditions.

Each component w_m ($m = 1, 2, \ldots, 6$) of the solution vector $\boldsymbol{w}$ can be expanded as a linear combination of local basis functions $\varphi_1^{(k)}, \varphi_2^{(k)}, \ldots, \varphi_{N_k}^{(k)}$ on each element Ω_k as follows

$$w_m(\boldsymbol{x}, t)\Big|_{\Omega_k} = \sum_{s=1}^{N_k} W_{ms}^{(k)}(t)\, \varphi_s^{(k)}(\boldsymbol{x}). \tag{11}$$

Here N_k denotes the number of basis functions supported on Ω_k. The total number of basis functions is $N = \sum_k N_k$. $W_{m1}^{(k)}, W_{m2}^{(k)}, \ldots, W_{mN_k}^{(k)}$ are coefficients of $\boldsymbol{w}$ on Ω_k corresponding to

the m-th equation. Plugging the expansion (11) into (9), substituting the basis functions for the test function $\boldsymbol{v}$ and splitting the surface integral, the following semi-discrete scheme is obtained

$$\sum_{s=1}^{N_k} \frac{\partial W_{ms}^{(k)}}{\partial t} \int_{\Omega_k} \varphi_r^{(k)} \varphi_s^{(k)} \,\mathrm{d}\Omega - \int_{\Omega_k} f_{mi}(\boldsymbol{w}, \nabla \boldsymbol{w}) \frac{\partial \varphi_r^{(k)}}{\partial x_i} \,\mathrm{d}\Omega + \sum_{l=1}^{L_k} \int_{\Gamma_{kl}} \mathcal{F}_m(\boldsymbol{w}^{\pm}, \nabla \boldsymbol{w}^{\pm}, \boldsymbol{n}) \, \varphi_r^{(k)-} \,\mathrm{d}\Gamma = \int_{\Omega_k} p_m(\boldsymbol{w}, \nabla \boldsymbol{w}) \, \varphi_r^{(k)} \,\mathrm{d}\Omega, \quad (12)$$

where $\Gamma_{k1}, \Gamma_{k2}, \ldots, \Gamma_{kL_k}$ are faces of the element Ω_k and L_k denotes the number of faces of Ω_k.

4. Element transformation

This section is focused on the evaluation of integrals in the semi-discrete scheme (12), which will be performed on the reference element. Let

$$x_i = x_i^{(k)}(\boldsymbol{\varepsilon}), \quad i = 1, 2, 3, \quad (13)$$

be the transformation from the reference element Ω_0 with coordinates $\boldsymbol{\varepsilon} = [\varepsilon_1, \varepsilon_2, \varepsilon_3]$ onto the physical element Ω_k with coordinates $\boldsymbol{x} = [x_1, x_2, x_3]$. Note that the transformation depends on the location of the physical element, therefore it is different for each element. All integrals are evaluated on the reference element. In general, a volume integral is transformed as follows

$$\int_{\Omega_k} \psi(\boldsymbol{x}) \,\mathrm{d}\Omega = \int_{\Omega_0} \hat{\psi}(\boldsymbol{\varepsilon}) \, J_k \,\mathrm{d}\Omega_0, \quad (14)$$

here J_k is the determinant of the Jacobian $\partial x_i^{(k)} / \partial \varepsilon_j$ of the transformation (13) and $\hat{\psi}$ refers to the function ψ transformed onto the reference element. For convenience, the basis functions are defined on the reference element, and so they can be integrated on the reference elements directly. We denote the basis function defined on the reference element by $\hat{\varphi}_1, \hat{\varphi}_2, \ldots, \hat{\varphi}_N$. The physical flux, on the other hand, is defined on the physical element. Since the physical flux is a contravariant tensor, it transforms as follows

$$\hat{f}_{mj}(\hat{\boldsymbol{w}}, \nabla \hat{\boldsymbol{w}}) = f_{mi}(\boldsymbol{w}, \nabla \boldsymbol{w}) \, \frac{\partial \varepsilon_j^{(k)}}{\partial x_i}, \quad (15)$$

where $\partial \varepsilon_j^{(k)} / \partial x_i$ is the inverse Jacobian. The normal numerical flux $\mathcal{F}$ and the production term p are also defined on the physical element. However, they are 0-order tensors, which means that their transformation is trivial.

The remaining integral that needs to be dealt with is the surface integral. Analogically to the reference element, a reference face Γ_0 is defined. Moreover, let

$$s_i = s_i^{(kl)}(\boldsymbol{\zeta}), \qquad i = 1, 2 \quad (16)$$

be the transformation from the reference face Γ_0 with coordinates $\boldsymbol{\zeta} = [\zeta_1, \zeta_2]$ onto the physical face Γ_{kl} with coordinates $\boldsymbol{s} = [s_1, s_2]$. In general, surface integrals are transformed as follows

$$\int_{\Gamma_{kl}} \psi(\boldsymbol{s}) \,\mathrm{d}\Gamma = \int_{\Gamma_0} \psi(\boldsymbol{\zeta}) \, J_{kl} \,\mathrm{d}\Gamma_0, \quad (17)$$

where J_{kl} is the determinant of the Jacobian of the transformation (16). Putting all that together, the semi-discrete scheme (12) becomes

$$\sum_{s=1}^{N_k} \frac{\partial W_{ms}^{(k)}}{\partial t} \int_{\Omega_0} \hat{\varphi}_r \, \hat{\varphi}_s \, J_k \, \mathrm{d}\Omega_0 - \int_{\Omega_0} f_{mi}(\boldsymbol{w}, \nabla \boldsymbol{w}) \frac{\partial \varepsilon_j^{(k)}}{\partial x_i} \frac{\partial \hat{\varphi}_r}{\partial \varepsilon_j} \, J_k \, \mathrm{d}\Omega_0 + \sum_{l=1}^{L_k} \int_{\Gamma_0} \mathcal{F}_m(\boldsymbol{w}^{\pm}, \nabla \boldsymbol{w}^{\pm}, \boldsymbol{n}) \, \hat{\varphi}_r^{-} \, J_{kl} \, \mathrm{d}\Gamma_0 = \int_{\Omega_0} p_m(\boldsymbol{w}, \nabla \boldsymbol{w}) \, \hat{\varphi}_r \, J_k \, \mathrm{d}\Omega_0 \tag{18}$$

with summation indices i and j. The vector of conservative variables $\boldsymbol{w}$ is a linear combination of the basis function, which are defined on the reference element. This means, that only $\hat{\boldsymbol{w}}$ and $\nabla \hat{\boldsymbol{w}}$ are known directly, given the coefficient of the linear combination. However, $\boldsymbol{w}$ and $\nabla \boldsymbol{w}$ appear in (18). Since $\boldsymbol{w}$ is a 0th-order tensor, its transformation is trivial. The gradient $\nabla \boldsymbol{w}$ is a covariant tensor and so the partial derivatives transform as follows

$$\frac{\partial \boldsymbol{w}}{\partial x_i} = \frac{\partial \hat{\boldsymbol{w}}}{\partial \varepsilon_j} \frac{\partial \varepsilon_j}{\partial x_i}. \tag{19}$$

5. Temporal discretization

The semi-discrete scheme (12) or (18) can be written as the following system of ordinary differential equations

$$\boldsymbol{M} \frac{\mathrm{d}\boldsymbol{W}(t)}{\mathrm{d}t} = \boldsymbol{R}(\boldsymbol{W}(t)), \tag{20}$$

here $\boldsymbol{W}$ is the vector of the unknown coefficients $W_{ms}^{(k)}$ of the linear combination (11). The mass matrix $\boldsymbol{M}$ and the right-hand side vector $\boldsymbol{R}$ are obtained by evaluating both the volume and surface integrals in (18) by the Gaussian quadrature rule of sufficient order.

In this study, only problems of finding the steady state are solved. Therefore, a first order of accuracy is sufficient for the time integration. In particular, the backward Euler method is used for time integration of the nonlinear system of ordinary differential equations (20) and the resulting nonlinear system of algebraic equations is linearized with the aid of the Jacobian $\partial \boldsymbol{R}/\partial \boldsymbol{W}$ to obtain the following system of linear algebraic equations

$$\left[\frac{1}{\Delta t_n} \boldsymbol{M} - \frac{\partial \boldsymbol{R}}{\partial \boldsymbol{W}}(\boldsymbol{W}_n) \right] \Delta \boldsymbol{W}_n = \boldsymbol{R}(\boldsymbol{W}_n). \tag{21}$$

The solution in the next time level is $\boldsymbol{W}_{n+1} = \boldsymbol{W}_n + \Delta \boldsymbol{W}_n$. The linearization can also be though of as performing one iteration of the Newton–Raphson method. The Jacobian $\partial \boldsymbol{R}/\partial \boldsymbol{W}$ is evaluated numerically, since it is tedious to derive it analytically. The entry (i, j) of the Jacobian can be approximated by the difference quotient

$$\left. \frac{\partial \boldsymbol{R}}{\partial \boldsymbol{W}} \right|_{ij} \approx \frac{\boldsymbol{R}_i(\boldsymbol{W} + h\boldsymbol{e}_j) - \boldsymbol{R}_i(\boldsymbol{W})}{h}, \tag{22}$$

where $\boldsymbol{e}_j$ is a vector with 1 in the j-th entry and zero elsewhere and h is a sufficiently small number, e.g., $h = 10^{-8}$. The system of linear equations (21) is solved iteratively using the GMRES method with the ILU(0) preconditioner.

6. Curvilinear element

The target of this section is to define a transformation that matches the geometry of the seal exactly. This transformation should map each hexahedral element into a ring sector as shown in Fig. 1. To this end, the transformation is divided into two steps. The first step is the mapping Φ that maps the reference element onto elements in an auxiliary mesh, which is essentially the unfolded final mesh. Any cross section in the $\hat{x}\hat{y}$ plane of the auxiliary mesh corresponds to a cross section in the final mesh. The axis θ corresponds to the angle in the final mesh. The transformation Φ has following form

$$\begin{aligned}
\hat{x} &= a_0 + a_1\varepsilon + a_2\xi + a_3\eta + a_4\varepsilon\xi + a_5\varepsilon\eta + a_6\xi\eta + a_7\varepsilon\xi\eta, \\
\hat{y} &= b_0 + b_1\varepsilon + b_2\xi + b_3\eta + b_4\varepsilon\xi + b_5\varepsilon\eta + b_6\xi\eta + b_7\varepsilon\xi\eta, \\
\theta &= c_0 + c_1\varepsilon + c_2\xi + c_3\eta + c_4\varepsilon\xi + c_5\varepsilon\eta + c_6\xi\eta + c_7\varepsilon\xi\eta,
\end{aligned} \tag{23}$$

where $\varepsilon \in [0,1]$, $\xi \in [0,1]$, $\eta \in [0,1]$. By plugging the coordinates of each of the eight vertices of the reference element into (23), each equation becomes a linear system of eight equations and eight unknowns. By solving for a_i, b_i and c_i ($i = 1, 2, \ldots, 7$), we obtain Φ explicitly. The second step is the mapping Ψ given by

$$\begin{aligned}
x &= \hat{x}, \\
y &= \hat{y}\cos(\theta), \\
z &= \hat{y}\sin(\theta),
\end{aligned} \tag{24}$$

which folds the auxiliary mesh into a circle. The overall transformation χ is the composition of the two transformations $\chi = \Psi \circ \Phi$.

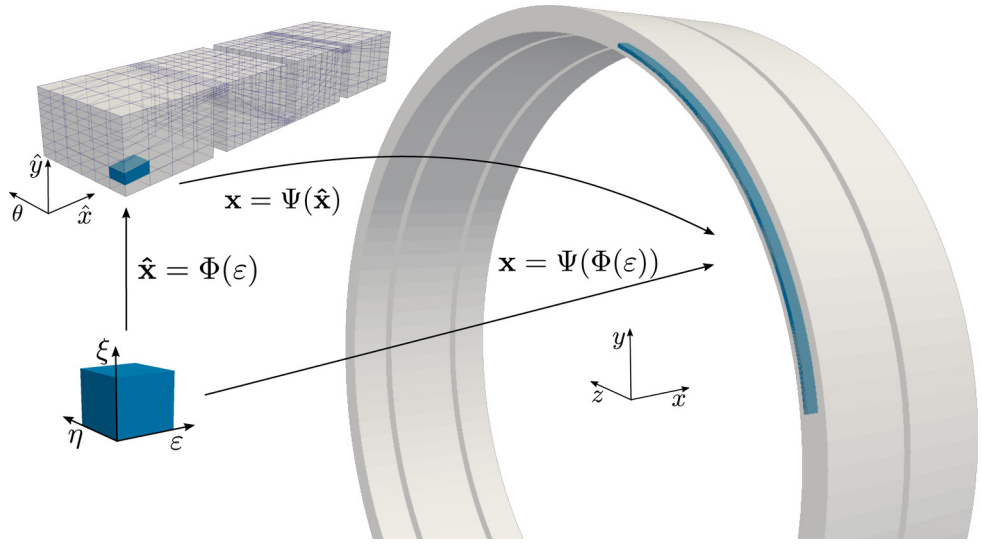


Fig. 1. Transformation of the reference element into a physical element

7. Seal geometry and computational meshes

As the test geometry, a staggered labyrinth seal depicted in Fig. 2 was chosen. There are two cavities in the labyrinth seal, which are created by three teeth each with the clearance of 0.4 mm. Two of the teeth are placed on the rotor and one tooth on the stator.

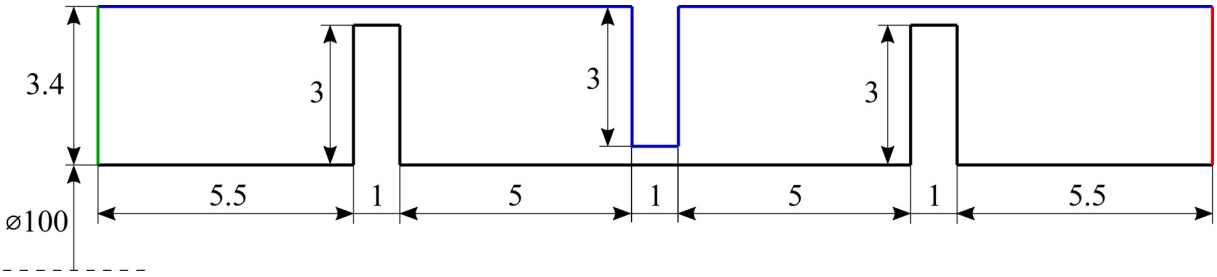


Fig. 2. Geometry of the test seal. Inlet (green), outlet (red), stator (blue) and rotor (black) are on the left, right, top and bottom, respectively. Units are millimetres

The total of four different meshes were considered. Three of the meshes were used for the present solver and one for Ansys Fluent. The meshes used for the present solver are labelled by numbers 1, 2 and 3 and have 1 008, 2 268 and 4 032 elements in the cross section, respectively. All three of the meshes are divided into 12 sections around the circumference of the seal. The total number of elements is obtained by multiplying the number of elements in the cross section by the number of circumferential divisions. The mesh that is used for simulations by Ansys Fluent is divided into 72 sections around the circumference of the seal and has 4 032 elements in the cross section. The total number of elements therefore is 290 304. This mesh has the exact same cross section as mesh 3 but the total number of elements is higher due to the higher number of circumferential divisions. The number of elements and number of divisions for each of the meshes is summarized in Table 1.

Table 1. Number of elements for each of the four meshes. Meshes 1, 2 and 3 are used for the present solver and the remaining mesh is used to compute the solution using Ansys Fluent. The total number of elements is the number of divisions multiplied by the number of elements in a cross section

mesh labels	circumfer. divisions	number of elements	
		cross section	total
1	12	1 008	12 096
2	12	2 268	27 216
3	12	4 032	48 384
Fluent	72	4 032	290 304

The reason why it is sufficient to use 12 divisions around the circumference of the seal is that the present solver uses special transformation onto circumferentially-curved elements, which capture the curvature of the seal. Ansys Fluent on the other hand does not implement curved elements, after all it uses the finite volume method, therefore, it requires more elements along the circumference of the seal. Moore [11] conducted a numerical study to find the right number of circumferential divisions and concluded that the mesh independence is achieved for about 61 divisions. To be on the safe side, 72 circumferential divisions were used in this paper, which corresponds to 5° for each division.

8. Numerical results

In this section, numerical results obtained using the discontinuous Galerkin method with three different meshes is compared with results obtained by the commercial CFD software Ansys Fluent. Linear basis functions were used in the case the discontinuous Galerkin method, which corresponds to the second order of accuracy. A compressible turbulent fluid flow is considered in all of the cases. In the simulation, air close to standard conditions was used as the fluid. The

boundary conditions are given by the pressure ratio $p_0^{\text{in}}/p^{\text{out}} = 1.267$. The fluid parameters along with the boundary conditions are summarized in Table 2.

Table 2. Fluid parameters and boundary conditions

name	symbol	value	units
fluid			
heat capacity	c_p	1 005	$\mathrm{J\,kg^{-1}\,K^{-1}}$
heat capacity ratio	κ	1.40	–
dynamic viscosity	η	$1.879 \cdot 10^{-5}$	Pa s
thermal conductivity	λ	$2.620 \cdot 10^{-2}$	$\mathrm{W\,m^{-1}\,K^{-1}}$
inlet			
total pressure	p_0^{in}	126.7	kPa
temperature	T_0^{in}	293	K
outlet			
static pressure	p^{out}	100	kPa

Three different meshes were used to obtain result by the present solver, see Section 7. Two simulations were performed with each computational mesh – one simulation with stationary rotor and one simulation with rotor rotating at the frequency of 50Hz. The results obtained by the present solver are compared with the results obtained by Ansys Fluent. A mesh with the same number of elements in the cross section as the finest mesh for the present solver is used for Ansys Fluent. The only difference is that the mesh used for Ansys fluent has more circumferential division and therefore a higher total number of elements.

In order to visualise the pressure profile along the labyrinth seal, a line through the seal was chosen as shown in Fig. 3 in a cross section. Static pressure along this line is plotted in Fig. 4 for both the stationary rotor (left) and rotating rotor (right). The blue, red and yellow lines corresponds to pressure obtained by the DG solver for three different meshes and the black dashed line corresponds to pressure obtained by Ansys Fluent.

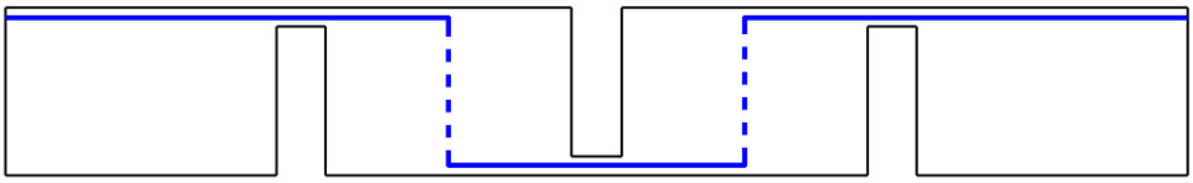


Fig. 3. Pressure line

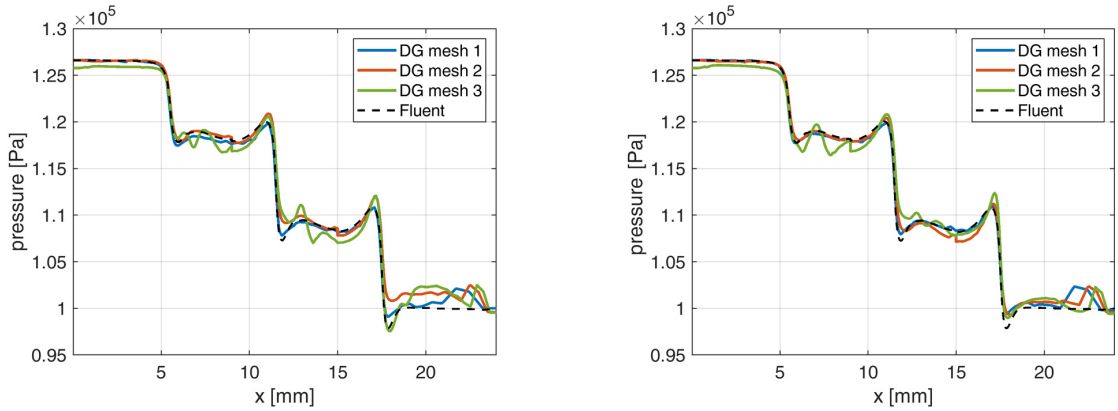


Fig. 4. Pressure along the line shown in Fig. 3 for stationary rotor (left) and rotating rotor (right)

Six lines labelled by letters A–F were chosen in a cross section of the labyrinth seal as shown in Fig. 5 to visualise the velocity in the cavities and in the clearances. Figs. 6–8 show velocity along these lines for stationary rotor and Figs. 9–11 show velocity for rotating rotor. Figs. 6 and 9 shows the x-component of velocity in the clearances A and F. Figs. 7 and 10 and Figs. 8 and 11 show appropriate components of velocity in the first and second cavity, respectively. Here, x-component of velocity is shown along the vertical lines B and D and y-component of velocity is shown along the horizontal lines C and E. In addition, Fig. 12 shows z-component of velocity along the vertical lines B and D, but only for the case of rotating rotor. In all of those cases, results computed by the present DG solver for three different meshes and by Ansys Fluent are shown.

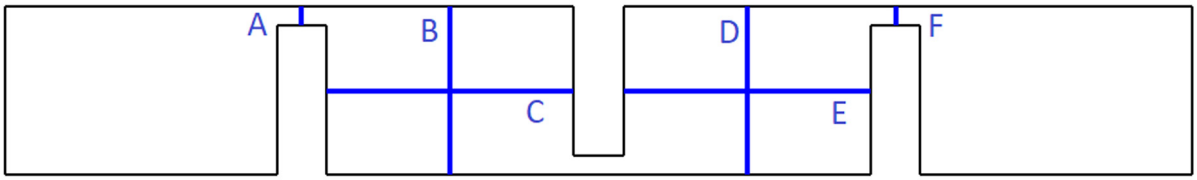


Fig. 5. Velocity lines

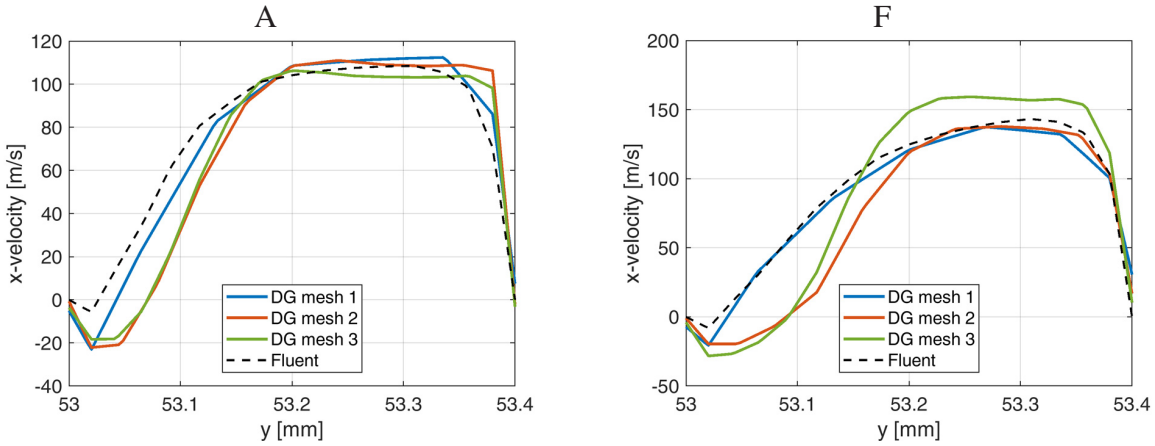


Fig. 6. Comparison of x component of velocities along the lines specified in Fig. 5 with rotor at rest

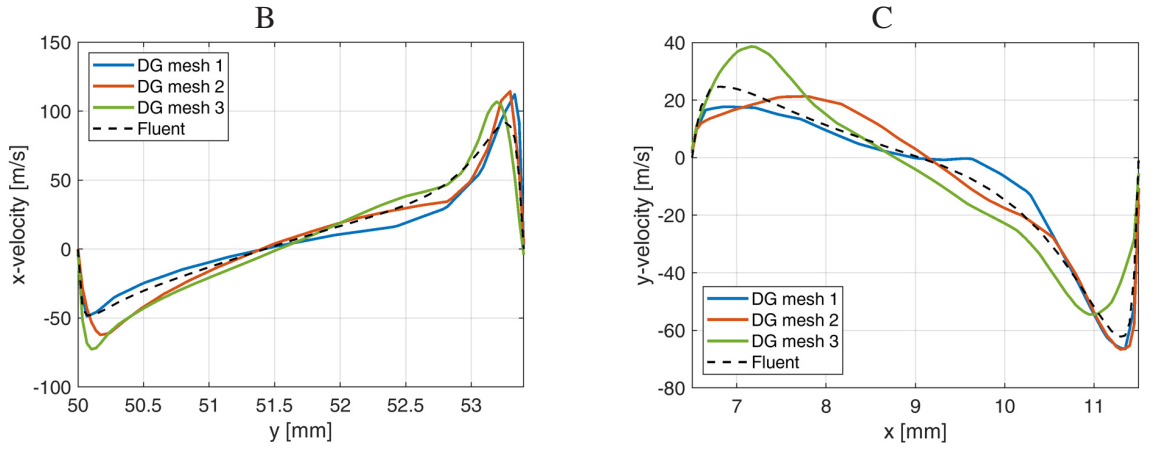


Fig. 7. Comparison of x and y components of velocities along the lines specified in Fig. 5 with rotor at rest

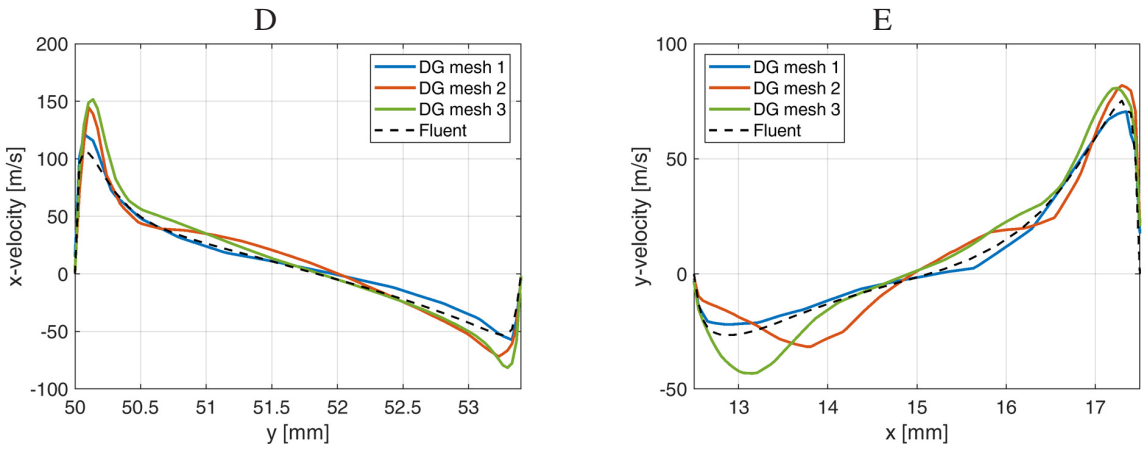


Fig. 8. Comparison of x and y components of velocities along the lines specified in Fig. 5 with rotor at rest

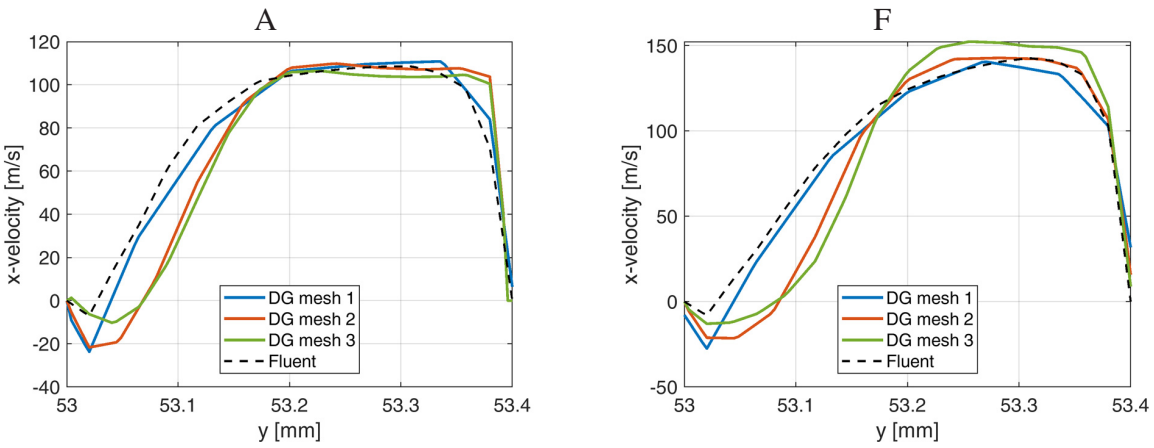


Fig. 9. Comparison of x component of velocities along the lines specified in Fig. 5 with rotor rotating with a frequency 50 Hz

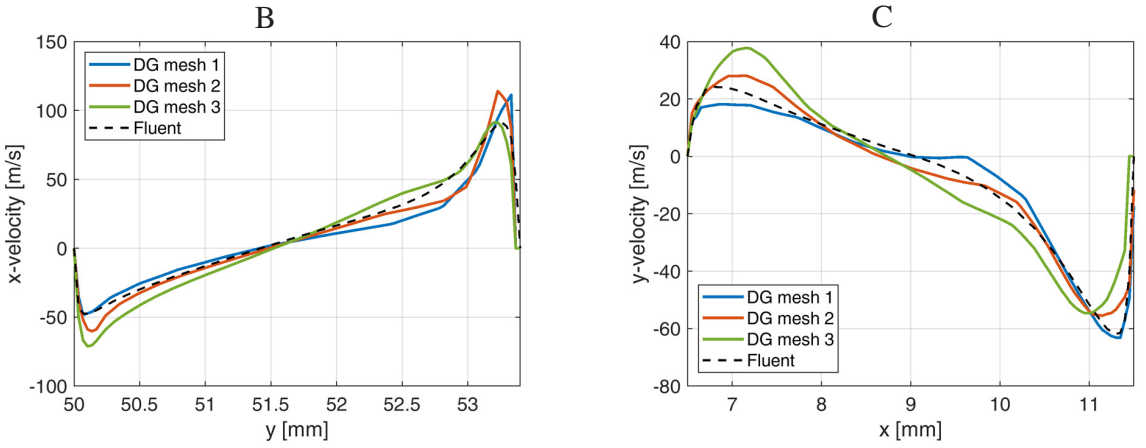


Fig. 10. Comparison of x and y components of velocities along the lines specified in Fig. 5 with rotor rotating with a frequency 50 Hz

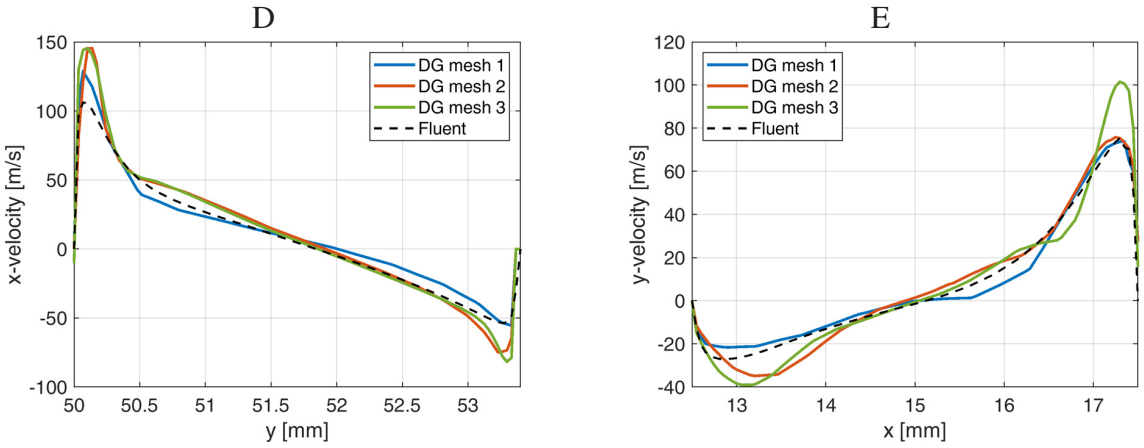


Fig. 11. Comparison of x and y components of velocities along the lines specified in Fig. 5 with rotor rotating with a frequency 50 Hz

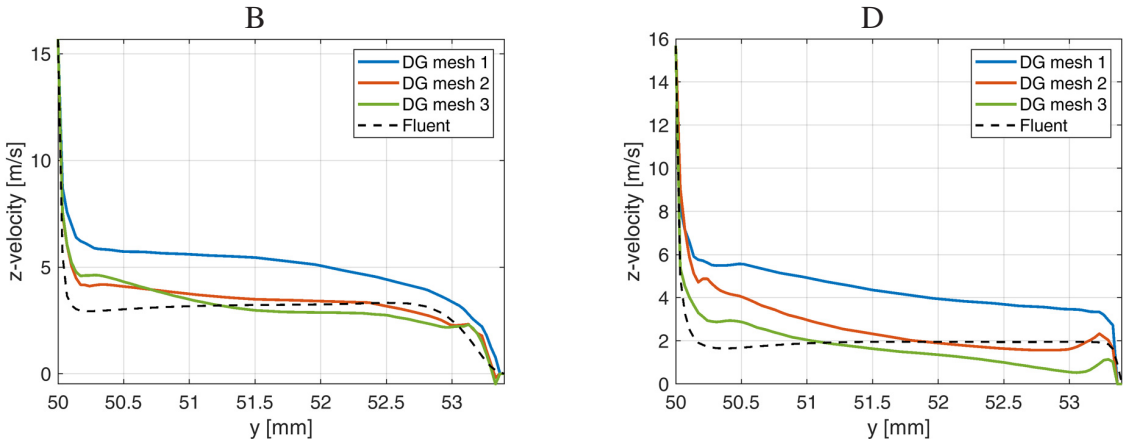
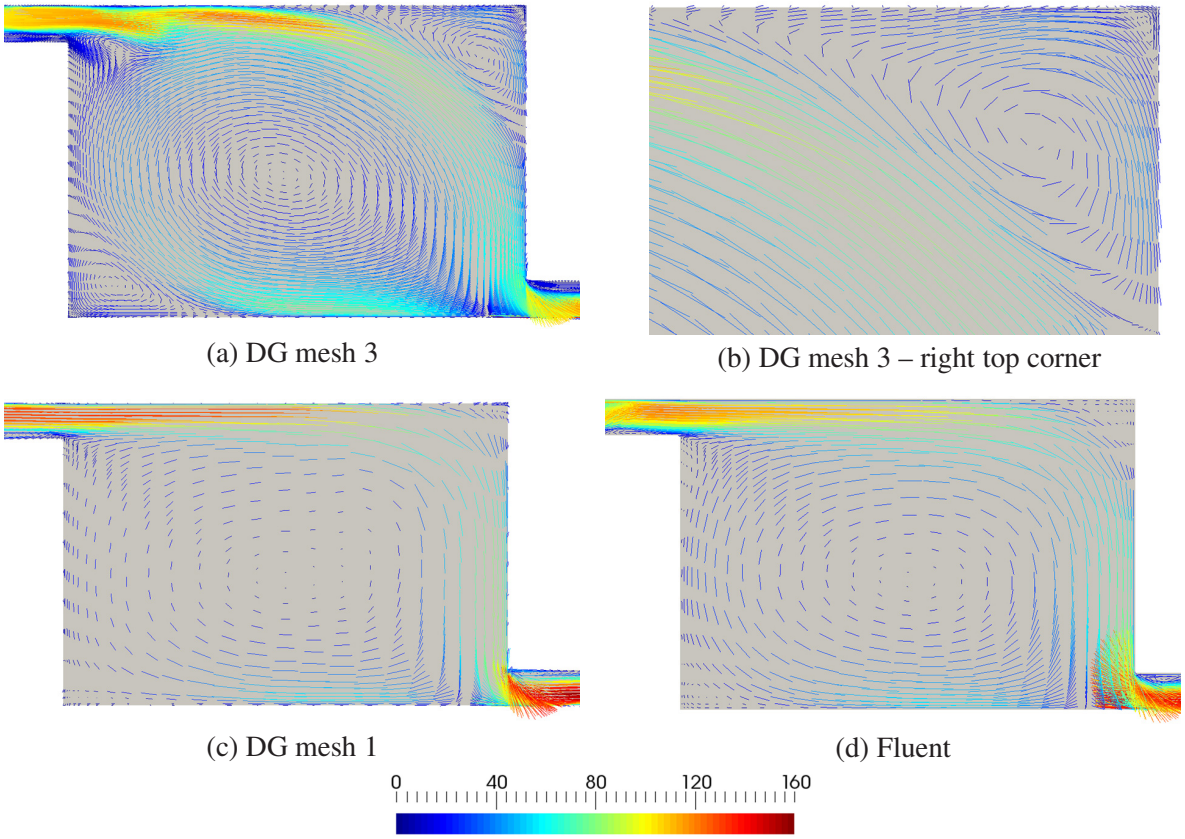


Fig. 12. Comparison of z component of velocity along the lines specified in Fig. 5 with rotor rotating with a frequency 50 Hz

Figs. 13 and 14 show the comparison of velocity vectors inside the first and second cavities computed by present DG solver and by Ansys Fluent. In both cases, solution using mesh 1 and mesh 3 are shown. The solution obtained by Fluent resembles to the solution obtained by the present solver with mesh 1 much more closely than the solution with obtained with the finest mesh. This fact indicates that the discontinuous Galerkin method is able to achieve higher accuracy on the same computational mesh. Notice that secondary vortices appear in both cavities in the solution obtained using mesh 3. A detailed view of chosen secondary vortices is shown in Figs. 13 and 14.

The distribution of Mach number in a chosen cross section is visualised in Fig. 15. A comparison between the results computed by the present solver with all of the three meshes and results obtained by Ansys Fluent is shown. Finally, the last figure, Fig. 16, shows the z-component of velocity in a chosen cross section. Here a comparison between the results computed by the present solver with the finest mesh, mesh 3, and result obtained by the Ansys Fluent is shown.



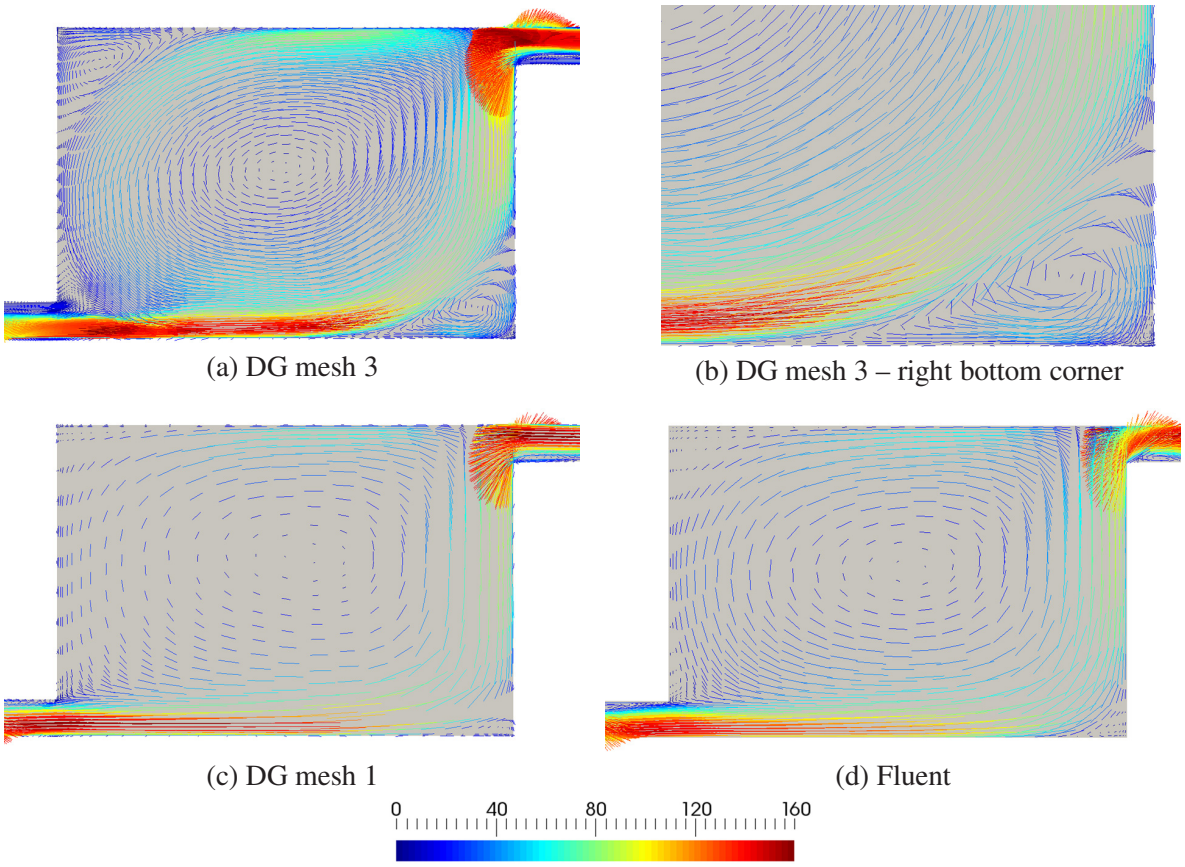
(a) DG mesh 3

(b) DG mesh 3 – right top corner

(c) DG mesh 1

(d) Fluent

Fig. 13. Velocity vectors inside the first cavity



(a) DG mesh 3

(b) DG mesh 3 – right bottom corner

(c) DG mesh 1

(d) Fluent

Fig. 14. Velocity vectors inside the second cavity

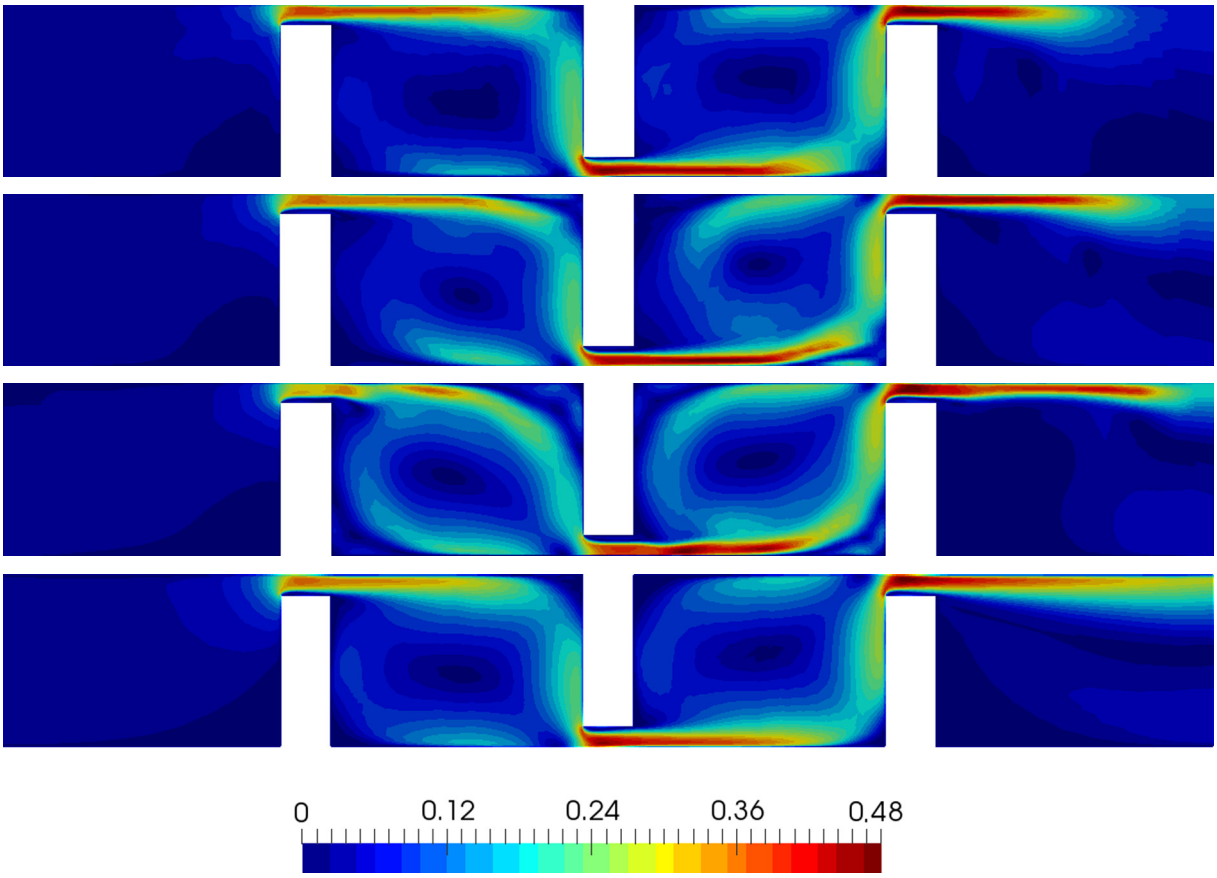


Fig. 15. Mach number contours in a cross section computed by the present solver with mesh 1, 2 and 3 and by Fluent, respectively from top to bottom

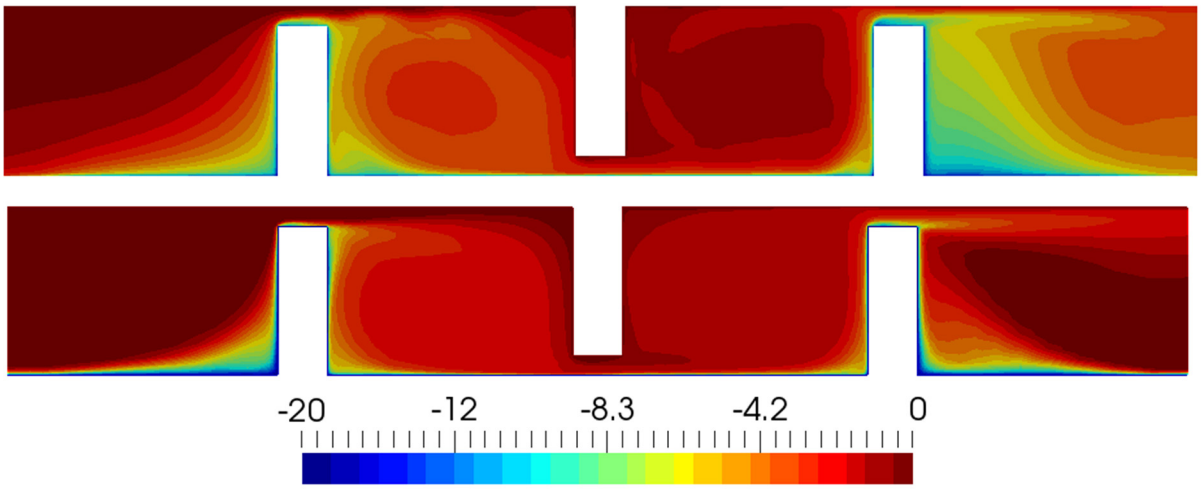


Fig. 16. Contours of the z-component of velocity in a cross section computed by the present solver with mesh 3 (top) and by Fluent (bottom) for rotor rotating at 50 Hz

9. Conclusion

In the present paper, the DG method was implemented and applied to the numerical simulation of a turbulent compressible fluid flow in a staggered labyrinth seal. Two different cases were consider – one case with a stationary rotor and the other with rotor rotating at 50 Hz. Each case was computed three times using three different meshes. To validate the DG method, a numerical simulation using the commercial CFD software Ansys Fluent was performed. In order to visualise the results, pressure was plotted across a cross section of the labyrinth seal and velocity was plotted along chosen lines in the clearances and inside both cavities. Moreover, the Mach number contours and the contours of the z-component of velocity were plotted inside a chosen cross section of the labyrinth seal.

The numerical results, i.e. pressure and velocity fields, are in the quantitative agreement with the solution obtained by Ansys Fluent. The comparison of the pressure distribution across the labyrinth seal, see Fig. 4, is the most important indicator of weather the two solvers agree with each other. The results show that the pressure field in the individual cavities takes on similar values in the case of all simulations by the present solver and in the case of Ansys Fluent. In the velocity plots, i.e., Figs. 6–12, the greatest agreement between the present solver and Ansys Fluent is seen for the coarsest mesh, that is for mesh 1. The reason for this is that the DG method has a high accuracy and, with a finer mesh, it starts to capture fine structures, such as small eddies, which change the character of the solution. This affect can be seen in Figs. 13 and 14. In particular, notice the small eddies in the case of the results computed by the finest mesh, i.e., mesh 3, in Figs. 13a and 14a. The eddies that from right on top of the rotor and stator teeth have a significant effect on the solution. The presence of eddies in the clearances can be seen in Figs. 6 and 9, where the dip in velocity close to the teeth is apparent when mesh 2 or mesh 3 is used. The effect of the rotor rotation on the z-component of velocity is shown in Fig. 12. The present solver gives a similar value z-component of velocity as Ansys Fluent when mesh 2 or mesh 3 is used. However, Ansys Fluent shows somewhat sharper boundary layer due to the wall treatment with wall functions, see [1].

In conclusion, the results suggest that the presented DG solver is able to capture the flow field more precisely than Ansys Fluent on a mesh with similar element size. Using the presented curvilinear element, a smaller number of elements can be placed around the circumference of the seal without a substantial loss of accuracy. The curvilinear element can be extended for cases where rotor undergoes precession. A fluid flow in a labyrinth seal with rotor undergoing precession can be solved in two different ways. The first option is to prescribe a motion to the rotor, in which case, the governing equations are solved using the arbitrary Lagrangian-Eulerian method. The second option is to change the perspective from a stationary frame of reference to the non-inertial rotating reference frame that follows the precession of the rotor. The presented curvilinear element is especially suitable for the latter approach, where the labyrinth seal is not rotationally symmetric.

9.1. Limitations of the study

The implementation of the discontinuous Galerkin method in combination with the $k-\varepsilon$ or $k-\omega$ turbulence model in a robust way is still a challenging task. To avoid numerical difficulties, the one-equation Spalart–Allmaras turbulence model was used in this study instead. A study that implements one of the aforementioned two-equation turbulence models is needed. Although the results of the present study suggest that the curvilinear element allows substantially large element size to be used in the circumferential direction, more research needs to be done to

determine the dependency of the number of circumferential division on the accuracy of the solution.

Acknowledgements

This work was supported by the ERDF as part of the project "Research Cooperation for Higher Efficiency and Reliability of Blade Machines (LoStr)" No. CZ.02.1.01/0.0/0.0/16_026/0008389.

References

[1] Ansys Fluent theory guide, 2021R2, Section 4.2.7.

[2] Bublík, O., Vimmr, J., Jonášová, A., Comparison of discontinuous Galerkin time integration schemes for the solution of flow problems with deformable domains, Applied Mathematics and Computation 267 (2015) 329–340. https://doi.org/10.1016/j.amc.2015.03.063

[3] Cockburn, B., Shu C.-W., Runge-Kutta discontinuous Galerkin methods for convection-dominated problems, Journal of Scientific Computing (16) (2001) 173-261. https://doi.org/10.1023/A:1012873910884

[4] Childs, D. W., Scharrer, J. K., An Iwatsubo-based solution for labyrinth seals: Comparison to experimental results, Journal of Engineering for Gas Turbines and Power 108 (2) (1986) 325–331. https://doi.org/10.1115/1.3239907

[5] Chupp, R. E., Hendricks, R. C., Lattime, S. B., Steinetz, B. M. Sealing in turbomachinery, Journal of Propulsion and Power 22 (2) (2006) 313–349. https://doi.org/10.2514/1.17778

[6] Dolejší, V., Holík, M., Hozman, J., Efficient solution strategy for the semi-implicit discontinuous Galerkin discretization of the Navier-Stokes equations, Journal of Computational Physics 230 (11) (2011) 4176–4200. https://doi.org/10.1016/j.jcp.2010.10.029

[7] Hur, M. S., Lee, S. I., Moon, S. W., Kim, T. S., Kwak, J. S., Kim, D. H., Jung, I. Y., Effect of clearance and cavity geometries on leakage performance of a stepped labyrinth seal, Processes 8 (11) (2020) 1–17. https://doi.org/10.3390/pr8111496

[8] Iwatsubo, T., Evaluation of instability forces of labyrinth seals in turbines or compressors in NASA CP 2133, Proceedings of a Workshop at Texas A & M University, Rotordynamic Instability Problems in High Performance Turbomachinery, 1980, pp. 205–222. https://ntrs.nasa.gov/citations/19800021214

[9] Kang, Y., Kim, T. S., Kang, S. Y., Moon, H. K., Aerodynamic performance of stepped labyrinth seals for gas turbine applications, Proceedings of the ASME Turbo Expo 2010: Power for Land, Sea, and Air, Volume 4: Heat Transfer, Parts A and B, Glasgow, UK, 2010, pp. 1191–1199. https://doi.org/10.1115/GT2010-23256

[10] Martin, H. M., Labyrinth packings, Engineering (10) (1908) 35–36.

[11] Moore, J. J., Three-dimensional CFD rotordynamic analysis of gas labyrinth seals, ASME Journal of Vibration and Acoustics 125 (4) (2003) 427–433. https://doi.org/10.1115/1.1615248

[12] Persson, P.-O., Bonet, J., Peraire, J., Discontinuous Galerkin solution of the Navier-Stokes equations on deformable domains, Computer Methods in Applied Mechanics and Engineering 198 (17–20) (2009) 1585–1595. https://doi.org/10.1016/j.cma.2009.01.012

[13] Reed, W. H., Hill, T. R., Los Alamos scientific laboratory report LA-UR-73-479, 1973.

[14] Sneck, H. J., Labyrinth seal literature survey, ASME Journal of Lubrication Technology (96) (1974) 579–581. https://doi.org/10.1115/1.3452498

[15] Spalart, P. R., Allmaras, S. R., A one-equation turbulence model for aerodynamic flows, AIAA Paper (1992) 92–439. https://doi.org/10.2514/6.1992-439

[16] Stocker, H. L., Determining and improving labyrinth seal performance in current and advanced high performance gas turbines, Procceedings of the AGARD Conference, 1978, pp. 13.1–13.22.

[17] Toro, E. F., Riemann solvers and numerical methods for fluid dynamics, Heidelberg, Springer, 1999.

[18] Vimmr, J., Bublík, O., Pecka, A., A parallel implementation of an implicit discontinuous Galerkin finite element scheme for fluid flow problems, Advances in Engineering Software (113) (2017) 108–119. https://doi.org/10.1016/j.advengsoft.2016.11.007

[19] Wittig, S., Schelling, U., Kim, S., Jacobsen, K., Numerical predictions and measurements of discharge coefficients in labyrinth seals, Proceedings of the ASME International Gas Turbine Conference and Exhibition, 1987. https://doi.org/10.1115/87-GT-188

[20] Xingyun, J., Zhang, H., Zheng, Q., Fan, S., Zhitao, T., Investigation on rotor-labyrinth seal system with variable rotating speed, International Journal of Turbo and Jet-Engines 36 (1) (2019) 19–29. https://doi.org/10.1515/tjj-2016-0066

[21] Zhang, L., Zhu, H., Liu, C., Tong, F, Experimental and numerical investigation on leakage characteristic of stepped labyrinth seal, Proceedings of the ASME Turbo Expo 2016: Power for Land, Sea, and Air, Seoul, Korea, 2016. https://doi.org/10.1115/GT2016-56743