VII International Conference on Computational Methods for Coupled Problems in Science and Engineering
COUPLED PROBLEMS 2017
M. Papadrakakis, E. Oñate and B. Schrefler (Eds)

DEVELOPMENT OF AN AUTOMATED PROCESS FOR TURBINE BLADE OPTIMISATION

TESSA UROIĆ*, BORNA ŠOJAT* AND HRVOJE JASAK†,*

*Faculty of Mechanical Engineering and Naval Architecture (FSB)
University of Zagreb
Ivana Lučića 5, 10000 Zagreb, Croatia
e-mail: tessa.uroic@fsb.hr, borna.sojat@stud.fsb.hr, hrvoje.jasak@fsb.hr
web page: http://www.fsb.unizg.hr/cfd

†Wikki Ltd.
Unit 459, Southbank House, Black Prince Road, London SE1 7SJ, United Kingdom
e-mail: h.jasak@wikki.co.uk, web page: http://wikki.gridcore.se/wikkiweb/company

Key words: Turbomachinery, Optimisation, Genetic Algorithm, `OpenFOAM`

Abstract. In this paper we present a fully automated procedure for turbine blade optimisation. Optimisation process consists of geometry parametrisation using B-splines, mesh deformation using the dynamic mesh library in `OpenFOAM`, numerical simulation of transonic flow through the blade passage and finding the feasible solutions with the Multi-Objective Genetic Algorithm (MOGA). The process proved to be robust whether starting the optimisation from unfeasible geometry or a conventional blade profile.

1 INTRODUCTION

The motivation for this work comes from industrial demands for faster and more efficient design cycles. Turbomachinery components are regularly found in many energy conversion processes where the work load is predetermined and constant. Thus, it is possible to optimise the components of a machine for a single operating point. The process of turbomachinery optimisation usually consists of several steps: geometry description in a mathematical sense (parametrisation), calculation of objective (fitness) functions, e.g. via computational fluid dynamics (CFD) simulation, and evaluation of the obtained solution(s). There are many optimisation approaches for different applications, from the method of trial-and-error to advanced adjoint optimisation algorithms [1]. The most common is the genetic or evolutionary algorithm which is based on Darwin's theory of natural selection: only the fittest individuals survive and provide their genetic code to the following generation. There are many examples of turbomachinery optimisation using

the genetic algorithm, sometimes combined with surrogate methods to decrease time-to-solution.

Sieverding et al. [2] have developed a two-dimensional optimisation process for industrial axial compressors, where they used a genetic optimisation algorithm and gave special attention to definition of an appropriate fitness function. The separately optimised blade sections are later stacked to create a three-dimensional blade and validated using a commercial CFD code. The parametrisation of the blade section is done using the concept of superposition of camber line and thickness distribution, and both are described by Bezier curves. The optimisation is single objective with a fitness function which takes into account non-dimensional loss coefficients, exit flow angle, etc. The calculation of the function is done by simulating the flow around the blade.

Zheng et al. [3] conducted optimisation of a transonic rotor of an axial compressor. The geometry is parametrised by B-spline and the design parameters were blade sweep and lean angle at four different blade heights. Response surface approximation was used for the objective function, with a polynomial and basis-function response methods. Genetic optimisation algorithms were used for obtaining the optimal solutions for both polynomial and basis-function methods. There were no constraints included, and variation of the blade sections was not considered.

Optimisation of the same transonic rotor was done by Wang et al. [4] also using a surrogate model and genetic algorithm, with an additional uncertainty analysis by probabilistic collocation method. The uncertainty is introduced by varying the outlet static pressure at the design operating point, and the trade-off result is compared to a deterministic single objective optimisation result.

Siddique et al. [5] presented an optimisation process of a centrifugal pump impeller consisting of a CFD simulation and a surrogate-based optimisation method. Inlet and exit angles of the impeller were chosen as the design variables and maximum head and efficiency were the two objective functions.

Ennil et al. [6] used a fully automated CFD simulation coupled with response surface analysis and multi objective genetic algorithm to optimise a small scale axial air driven turbine with respect to energy losses.

Before-mentioned researches do not elaborate the bottleneck of the optimisation process, i.e. the generation of the computational mesh for the CFD simulation.

Page et al. [7] stressed the importance of meshing in the automatic CFD design optimisation processes. They proposed an automatic mesh generation and an adjoint based error analysis to select the optimal mesh configuration for a given geometry and the objective function. The estimation of the error induced by space discretisation was carried out and was used as an indicator to improve mesh quality to obtain a more accurate solution.

However, this approach does not take into account the time needed for generation of a complex mesh and a large number of evaluations which have to be conducted by the genetic algorithm.

Our paper presents a robust, fully automated fast optimisation process for turbomachinery, applied to a single two-dimensional blade pasagge. The process consists of four steps:

1. geometry parametrisation using B-spline curves,
2. deformation of a starting, fully structured computational mesh,
3. CFD simulation of the compressible flow,
4. estimation of the obtained solutions and generation of new geometry by the optimisation algorithm.

Each of these steps will be described in the next section.

2 METHODOLOGY

In this section, each of the four steps of the optimisation loop will be described. The flowchart of the optimisation loop is shown in Fig. 1.

2.1 Geometry parametrisation

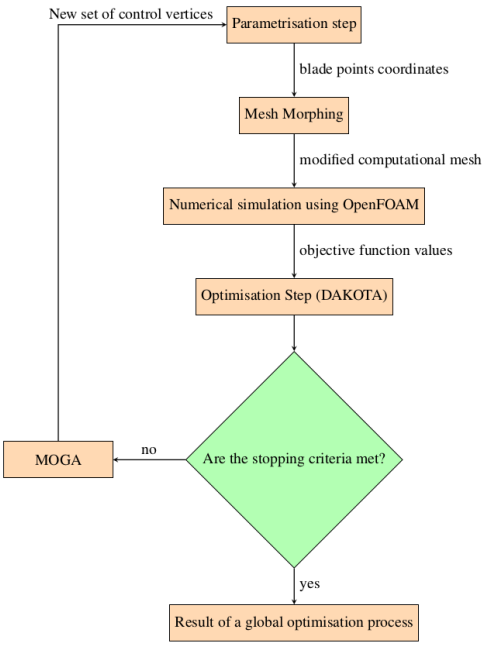

Figure 1: Optimisation loop

To achieve communication between the optimisation algorithm and the CFD simulation, i.e. the mesh deformation utility, it was necessary to describe the blade geometry in a straightforward way. The easiest and most convenient, but maybe not the most practical way for blade profiles, is to represent the geometry with control points. These control points are defined by the optimisation algorithm as the design parameters. However, the following step of the optimisation process, the mesh deformation, requires many more points than the defined number of control points. To *connect* the control points and extract more points in–between which will describe the geometry more precisely, we chose to parametrise the blade using periodic B-spline curves [8]. B-spline approximates the control points and is easily deformed by changing the position of the control points. The result is a smooth and continuous blade surface without any sudden and sharp jumps. Each segment

of the curve between the two control points is of the 5^{th} degree and 40 parametrisation points were extracted from each segment. An example of a parametrised NACA 4421 profile can be seen in Fig. 2.

2.2 Mesh deformation

In [9] an overview of numerical errors resulting from the discretisation procedure (treating convection and temporal terms) and mesh-induced errors is given. Consistent use of discretisation schemes in blade passage simulations can be ensured, but mesh-induced errors are more difficult to measure and control. A large number of geometries are tested by the optimisation algorithm and it is time demanding to create a new computational mesh for each geometry. In order to compare the results for different geometries, similar mesh resolution and quality (orthogonality, skewness) must be ensured for each case. Creating the computational mesh by hand for every single geometry to have more control over the resulting mesh properties is impossible. Automatic meshing algorithms cannot provide the same or even similar mesh configuration for different geometries, no matter how small the shape variation is. This is the reason a mesh morphing procedure was developed for blade geometry optimisation, using the dynamic mesh library implemented in `OpenFOAM` [10].

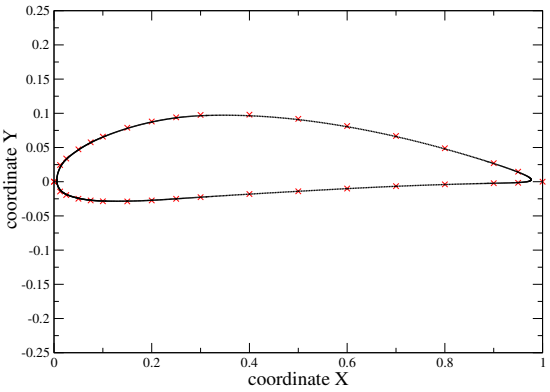

Figure 2: NACA 4421 blade with control points in red and B-spline in black

The morphing procedure is based on the assumption that the computational mesh behaves as an elastic body in a state of equilibrium, as described in [11]. If some force acts on the boundary of the body, it will cause the motion of the boundary which will influence the interior of the body. Influence of boundary motion on interior points of the mesh can be described with a Laplace equation:

$$\nabla \cdot (\gamma \nabla \vec{u}) = 0 \tag{1}$$

where $\vec{u}$ is the displacement vector, and γ is the diffusion coefficient. By applying the Laplace displacement equation with a constant coefficient γ, the largest deformation of the mesh elements occurs near the mesh boundary. It usually leads to extreme mesh distortion and negative cell volumes. This is why a space dependent coefficient of diffusivity is used. Notice that increase of the diffusivity coefficient results in lower deformation of the mesh element. Thus, it is beneficial to define the diffusivity coefficient to be inversely proportional to distance of the mesh element from the mesh boundary. The function can

be linear, quadratic or exponential [11]. In this study, a quadratic function is used:

$$\gamma(l) = l^{-2}, \tag{2}$$

where l is the minimum distance of the mesh element from the mesh boundary.

The Laplace displacement equation (1) is discretised into a linear system of equations:

$$[A] \cdot \vec{u}_k = r_k, \quad k = 1, 3, \tag{3}$$

where $\vec{A}$ is the coefficient matrix, $\vec{u}$ is the unknown vector of displacement and $\vec{r}$ is the right hand side vector. The new geometry configuration is defined with a set of points obtained from the parametrisation which are introduced into the system (3) as a fixed value (Dirichlet) boundary condition.

2.3 Numerical simulation

After obtaining the new mesh from the deformation procedure, simulation of transonic flow through the blade passage is performed using `foam-extend`, a community driven fork of the open source software `OpenFOAM`. The governing equations are the continuty equation (4), momentum equation (5) and energy equation (6).

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \vec{u}) = 0 \tag{4}$$

$$\frac{\partial \vec{u}}{\partial t} + \nabla \cdot (\rho \vec{u} \otimes \vec{u}) - \nabla \cdot (\nu \nabla \vec{u}) = -\nabla p, \tag{5}$$

$$\frac{\partial \rho e}{\partial t} + \nabla \cdot (\rho e \vec{u}) = \rho \vec{g} \cdot \vec{u} + \nabla \cdot (\sigma \cdot \vec{u}) - \nabla \cdot \vec{q} + \rho Q \tag{6}$$

The energy equation is weakly coupled with the rest of the system because e (specific internal energy), and temperature T influence ρ and convective velocity $\vec{u}$ through the equation of state, written here for the ideal gas (7).

$$\rho = \frac{P}{RT} = \psi P, \tag{7}$$

where ψ is the compressibility,

$$\psi = \frac{1}{RT}. \tag{8}$$

Compressibility effects can be observed at higher fluid velocities, and Mach number is commonly used to determine the critical velocity at which the flow can be considered compressible.

In this study $k - \omega$ SST [12, 13] turbulence model was used.

The results of the numerical simulation were processed, objective function values were calculated from corresponding flow variables and delivered to the optimisation algorithm.

In turbines, stator blades are vital parts of the geometry. They are used to accelerate the fluid flow as much as possible with the smallest possible pressure drop, while directing the fluid flow at optimal attack angle onto rotors. Having this in mind, objective functions were defined as follows. The target was to produce a geometry that would give the largest increase in velocity while keeping the pressure drop as small as possible. Thus, Δu and Δp were defined as:

$$\Delta u = u_{outlet} - u_{inlet} \longrightarrow \text{maximise}, \tag{9}$$

and

$$\Delta p = p_{inlet} - p_{outlet} \longrightarrow \text{minimise}. \tag{10}$$

2.4 Multi-objective optimisation

Within this work genetic optimisation algorithm was used. Genetic algorithm is a derivative free global method. Derivative free methods are usually more robust than the gradient based approaches and they are applied when the problem is non-smooth, multi modal or poorly behaved. On the other hand, this approach is usually more computationally demanding because of its slower convergence rates for finding an optimal solution(s). Multi-objective Genetic Algorithm, MOGA [14], an algorithm specially designed for multi-objective problems was used, since there were two objective functions which are conflicting with each other. This means a unique optimal solution doesn't exist, but the algorithm calculates a set of feasible Pareto solutions. Each solution is a trade-off since the objective functions cannot achieve their optimal values simultaneously.
Settings of the MOGA algorithm were as follows:

- size of a generation: 50
- number of children: 40
- crossover rate: 0.75
- mutation rate: 1
- maximal number of generations: 10

The latter was used as the stopping criterion for the optimisation loop.

To achieve convergence of the optimal solution in a reasonable amount of time, strong geometric constraints were prescribed for the design variables, Fig. 3.

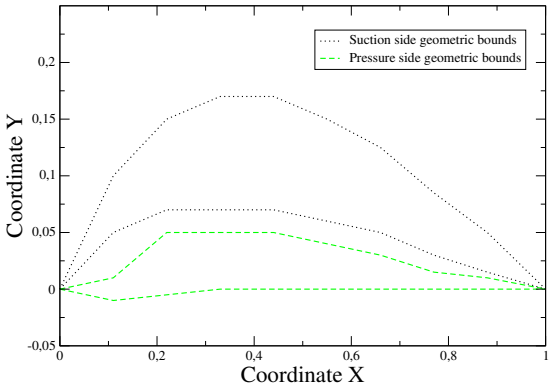

Figure 3: Geometric constraints

3 RESULTS AND DISCUSSION

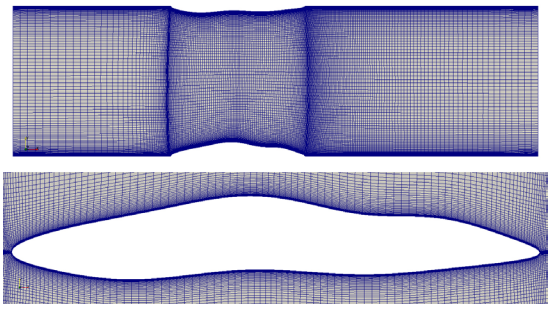

Figure 4: Initial geometry and computational mesh

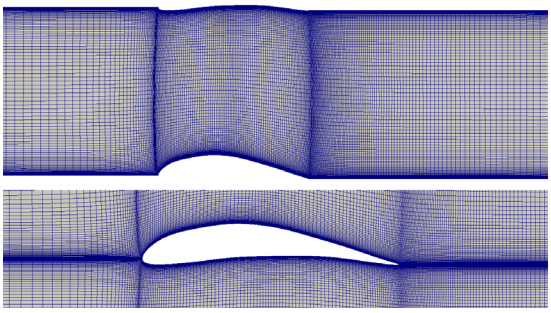

Figure 5: Optimised geometry and computational mesh after second MOGA pass

We decided to start the optimisation process with a *dummy* geometry, a shape which is not a stator blade in any sense, Fig. 4. The reason for such initial geomery is that we wanted to see if the deformation algorithm is able to deform the mesh starting from a distorted and very different shape, while keeping the mesh quality, and to see whether the optimisation process can produce a good result starting from a far from ideal initial geometry configuration. The blade was split into suction and pressure side to form a passage. It is possible to expand the width of the passage in the mesh deformation algorithm to accomodate thicker blades, but this was avoided by prescribing geometric constraints. The computational mesh is fully structured and has approximately 28000 cells. Mesh refinement study was conducted and current mesh density was chosen because it provided satisfactory results while being computationally less expensive than the finest mesh tested. In case of errors appearing in the mesh deformation procedure (negative volume cells, highly skewed or non-orthogonal cells), these cases would be rejected as suboptimal and would not be evaluated, but this happened in none of the cases. The patches parallel to the blade walls were treated as periodic, using General Grid Interface (GGI) [15] for interpolation of flow variables. Value of total pressure was defined at the inlet, and value of static pressure at the outlet. Several attack angles were tested and here the results for attack angle $\alpha = 5°$ are presented.

One iteration of the process, including parametrisation, mesh deformation, numerical simulation and objective function evaluation lasted 13 minutes, on a single Intel i5-4570 (3.2 GHz) CPU core. Numerical simulation took the longest, 12 minutes (92,3%), mesh deformation took less than a minute (7,7%) while geometry parametrisation and objective function evaluation took less than 1 second. In comparison, the same iteration with an automatic meshing algorithm, [16], would take 15 minutes, and it wouldn't be possible to control local mesh quality. The process is easily parallelizable, as multiple evaluations can be done on different processor cores at the same time.

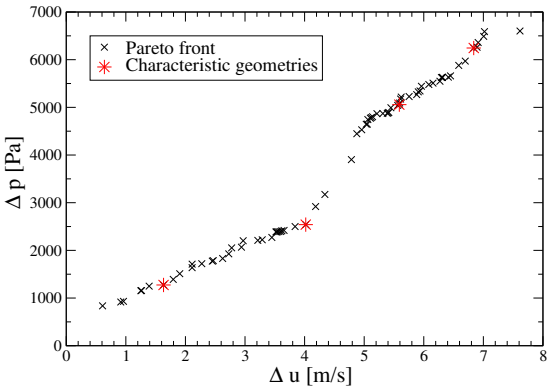

Figure 6: Pareto front

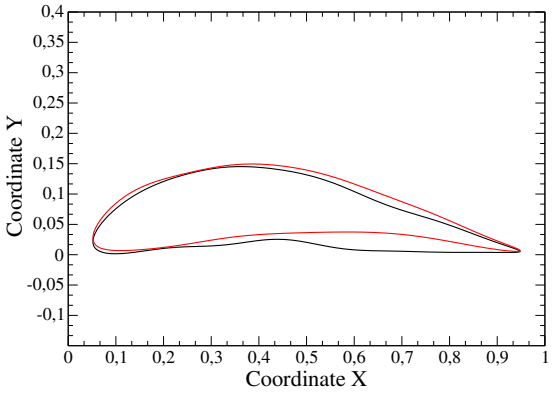

Figure 7: Pareto optimum after second MOGA optimisation (red) compared to the initial optimum (black)

The process ran 887 evaluations before reaching the stopping criterion of 10 generations and the resulting Pareto front is shown in Fig. 6. Red stars represent four chosen characteristic geometries, for which objective functions' values are shown in Tab. 1, as well as the values for the initial geometry. The corresponding geometries are shown in Figs. 8, 9, 10, 11, going from left to right in the Pareto front, Fig. 6.

Table 1: Parameters of four characteristic Pareto front geometries.

Point on the Pareto front	**Objective function values**
Initial geometry	$\Delta u = 2.695$ m/s $\Delta p = 2251.37$ Pa $\Delta p/\Delta u = 835.39$ Pa/ms^{-1}
1	$\Delta u = 1.633$ m/s $\Delta p = 1274.37$ Pa $\Delta p/\Delta u = 780.39$ Pa/ms^{-1}
2	$\Delta u = 4.185$ m/s $\Delta p = 2919.64$ Pa $\Delta p/\Delta u = 697.64$ Pa/ms^{-1}
3	$\Delta u = 5.589$ m/s $\Delta p = 5057.05$ Pa $\Delta p/\Delta u = 904.82$ Pa/ms^{-1}
4	$\Delta u = 6.839$ m/s $\Delta p = 6245.39$ Pa $\Delta p/\Delta u = 913.20$ Pa/ms^{-1}

Solution 2 had the smallest pressure drop for the given velocity increase, and we have

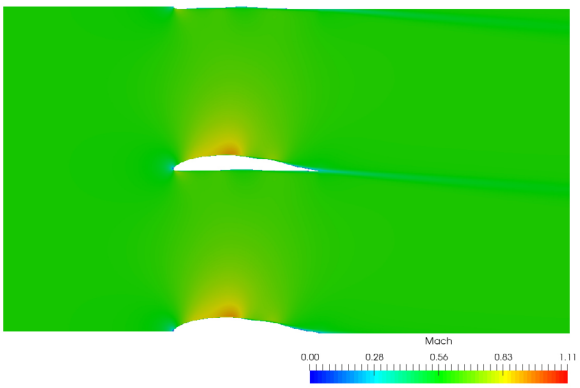

Figure 8: Optimal solution 1

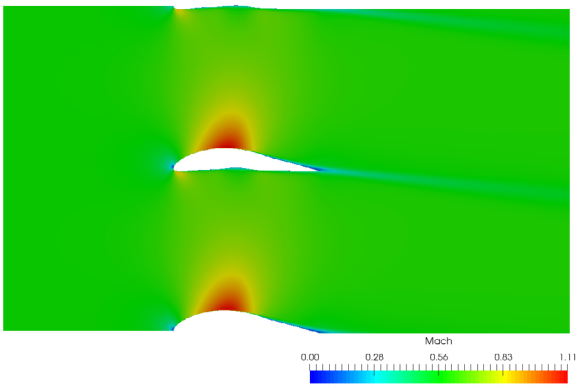

Figure 9: Optimal solution 2

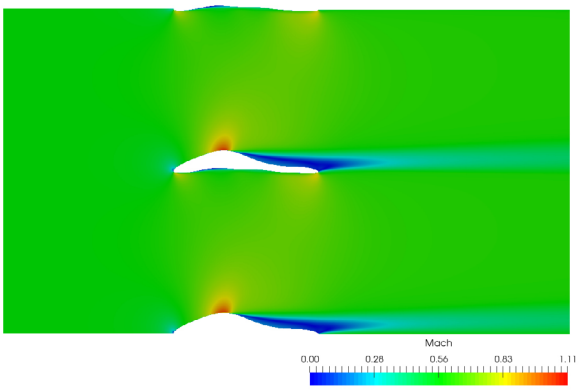

Figure 10: Optimal solution 3

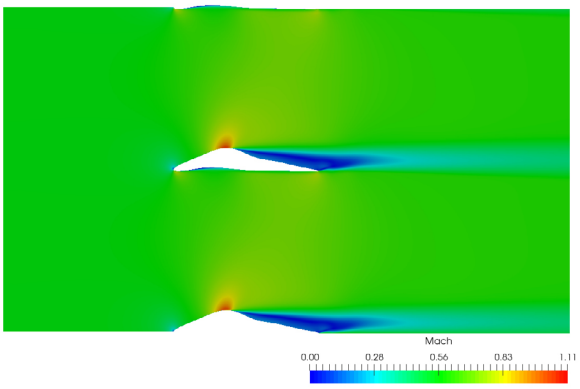

Figure 11: Optimal solution 4

chosen it to conduct further optimisation using MOGA to see whether starting from a feasible solution would produce different results. After 702 evaluations, a final shape of the stator blade was obtained, Figs. 5 and 7. The values of the objective functions for this blade were $\Delta u = 5.576$ m/s and $\Delta p = 3131.40$ Pa, which is 561.59 Pa of pressure drop per 1 m/s velocity increase, while for the first Pareto optimum, this ratio was 697.64 Pa per 1 m/s. Thus, second pass of MOGA, or starting from a feasible initial geometry, produced a better result.

4 CONCLUSIONS

We developed a robust optimisation procedure in which we included a mesh deformation algorithm which enabled us to localise numerical errors induced by the spatial discretisation in the Finite Volume Method. Additionaly, the mesh deformation utility contributed to the efficiency of the overall procedure because it is considerably faster than the automatic meshing algorithm we tested. The procedure could be made even more robust by carefully defining the design variables, e.g. parametrising the camber line with a set of control points and defining a blade thickness distribution. 16 parametrisation control points were chosen as the design parameters. Genetic algorithm which was used has to perform a large number of evaluations to converge to an optimal set of solutions, and strong geometric constraints were prescribed to narrow the design space. Starting the optimisation process from an unfeasible geometry produced a set of feasible solutions without any failures in any of the steps of the process. Starting the optimisation process from *the best* of those solutions yielded an even better solution, according to the objective functions. This optimisation process can be modified for three-dimensional cases and more complex geometries, e.g. for applications in naval architecture, aeronautical engineering, etc.

REFERENCES

[1] Zhang, P., Lu, J., Song, L., Feng, Z. Study on continuous adjoint optimization with turbulence models for aerodynamic performance and heat transfer in turbomachinery cascades *International Journal of Heat and Mass Transfer.* (2017) **104**:1069–1082.

[2] Sieverding, F., Ribi, B., Casey, M., Meyer, M. Design of Induatrial Axial Compressor Blade Sections for Optimal Range and Performance *Journal of Turbomachinery.* (2004) **126**:323–331.

[3] Zheng, R., Xiang, J., Sun, J. Blade geometry optimization for axial flow compressor *Proceedings of ASME Turbo Expo 2010.* (2010)

[4] Wang, X., Hirsch, C., Liu, Z., Kang, S., Lacor, C. Uncertainty-based robust aerodynamic optimization of rotor blades *International Journal for Numerical Methods in Enginnering.* (2013) **94**:111–127.

[5] Siddique, H., Mrinal, K.R., Samad A. Optimization of a Centrifugal Pump Impeller by Controlling Blade Profile Parameters *Proceedings of ASME Turbo Expo 2016.* (2016)

[6] Ennil, A.B., Al-Dadah, R., Mahmoud, S., Rahbar, K., Aljubori, A. Minimization of loss in small scale axial air turbine using CFD modeling and evolutionary algorithm optimization *Applied Thermal Engineering.* (2016) **102**:841–848.

[7] Page, J.H., Watson, R., Ali, Z. Advances of Turbomachinery Design Optimization *Proceedings of 53rd AIAA Aerospace Sciences Meeting.* (2015)

[8] Rogers, D.F. *An introduction to NURBS with historical perspective* The Morgan Kaufmann Series in Computer Graphics (2001)

[9] Jasak, H. *Error analysis and estimation for the Finite Volume Method with applications to fluid flows* PhD thesis, Imperial College of Science, Technology and Medicine, London (1996)

[10] Jasak, H. Open source CFD in research and industry *International Journal of Naval Architecture and Ocean Engineering.* (2009) **1**:89–94.

[11] Tuković, Ž. *Metoda kontolnih volumena na domenama promjenjivog oblika* PhD thesis, Faculty of Mechanical Engineering and Naval Architecture, University of Zagreb (2005)

[12] Menter, F.R. Two-equation eddy-viscosity turbulence models for engineering applications *AIAA Journal.* (1994) **32**:1598–1605.

[13] Menter, F.R., Kuntz, M., Langtry, R. Ten years of industrial experience with the SST turbulence model *Turbulence, Heat and Mass Transfer.* (2003) **4**:625–632.

[14] Adams, B.M. et al. *Dakota, a multilevel parallel object-oriented framework for design optimization, parameter estimation, uncertainty quantification and sensitivity analysis: Version 6.4 User's manual* (2016)

[15] Beaudoin, M., Jasak, H. Development of a General Grid Interface for turbomachinery simulations with OpenFOAM *Proceedings of Open Source CFD International Conference.* (2008)

[16] Retrieved from *http://cfmesh.com/cfmesh/* Last access: 13th March 2018.