Journal of Computer Engineering
ISSN 10003428

Original Research Article
Vol 12, Issue 6, 2023 | ISSN: 1000-3428

Adjoint Multidisciplinary Optimization of Radial Turbines Under Structural Constraints: A CAD-Based Approach

Dr. Sigrid Adriaenssens
Tel Aviv University, Israel.

Abstract

Most adjoint-based optimization frameworks only consider aerodynamic performance and constraints, leading to designs that need to pass through revisions by structural requirements. Only in recent years, adjoint optimization frameworks have been extended to include structural constraints. These frameworks also make use of CAD-based parametrizations to maintain a connection to the master CAD geometry and to serve as the connection between the fluid and solid domains.

Indexed keywords: Computer Engineering, Advanced Computing, Technology, Open Access

Article History: Received: 17 March 2023 | Accepted: 20 May 2023 | Published: 13 June 2023

In this work, a CAD-based adjoint multidisciplinary optimization framework for turbomachinery components is presented. A CAD-based parametrization is used for defining the shape freedom, from which the fluid and solid grids are generated, and a Reynolds-Averaged Navier-Stokes solver is used to compute the efficiency. The maximum von Mises stress is computed using a linear stress solver based on the Finite Element Method. The CFD and stress solvers each have adjoint capabilities, permitting an efficient computation of gradients at a cost independent of the size of the design space.

An adjoint optimization of a radial turbine is performed with the objective of maximizing the aerodynamic efficiency while adhering to the structural constraints. Results show that within a reduced design time an aerodynamic optimal design can be achieved whilst keeping the mechanical stresses within range of the prescribed tolerance.

Introduction

In the field of turbomachinery, multidisciplinary optimizations (MDO's) have been widely applied using gradient-free optimization methods. These methods are straightforward to implement due to their non-intrusive nature of not requiring any source code access. However, gradient-free methods require a high number of iterations to converge towards an optimum and the design space is limited by the curse of dimensionality, i.e., the computational effort increases exponentially with respect to the number of design parameters. Alternatively, gradient-based optimization methods use gradient information to converge towards a local optimum, typically with less iterations while allowing larger degrees of freedom. However, this requires computing the gradient of the objective with respect to the design parameters.

The required gradient can traditionally be computed using a non-invasive approach such as finite differences (FD). However, the cost of using FD is also proportional to the number of design parameters n: $n + 1$ evaluations for 1st order FD and $2n$ evaluations for 2nd order FD. The adjoint method (Pironneau, 1974; Jameson, 1988) allows a gradient calculation at a cost proportional to the number of objectives, rather than the number of design parameters. Since the number of objectives is generally much less than the number of design parameters, this significantly speeds up the gradient computation, and as a result, the optimization.

State of the art adjoint optimizations in turbomachinery focus on aerodynamic cost functions and constraints (Wang and He, 2010; Walther and Nadarajah, 2013; Luo et al., 2014). Only recently have adjoint MDO's been extended to include structural constraints (Verstraete et al., 2017). Including these constraints within a CAD-based adjoint MDO framework enables the efficient design of geometries that are not only aerodynamically optimal, but also structurally feasible. In this work, the adjoint MDO framework CADO (Verstraete, 2010) of the von Karman Institute for Fluid Dynamics is used to optimize the efficiency of a radial turbine under structural constraints.

How to cite: Dr. Sigrid Adriaenssens Tel Aviv University, Israel. (2023). Adjoint Multidisciplinary Optimization of Radial Turbines Under Structural Constraints: A CAD-Based Approach. Journal of Computer Engineering, 12(6), 61–67. DOI: https://doi.org/10.5281/zenodo.19349239

First, the adjoint MDO framework will be briefly presented. This will be followed by a discussion of the numerical setup, including the flow and structural solvers and the optimization method. Finally, the optimization results are discussed.

Adjoint Multidisciplinary Optimization Framework

The optimization begins with CAD parameters $\alpha \in \mathbb{R}^n$ that are used to describe the geometry (figure 1). The number of CAD parameters is defined by n. A CAD-based approach is chosen to give users the possibility of imposing geometrical constraints for an optimization. Geometrical constraints can be used to define e.g. constraints on a shape's curvature for manufacturing purposes. Additionally, the CAD surface serves as the interface between the structured mesh of the fluid domain and the unstructured mesh of the solid domain. The CAD parameters α are used as inputs to the CAD kernel which generates the geometry.

Based on the CAD geometry, a structured mesh is generated for the CFD calculation and an unstructured mesh is generated for the structural solver. Following the mesh generation, CFD and CSM analyses are carried out to compute the performance parameters of interest y.These include the efficiency η and the maximum von Mises stress σ_{max} for the fluid and structural disciplines, respectively. The performance parameters are then used to define an objective J, which is to be minimized.

Discrete adjoint implementations of the CFD and CSM solvers, combined with forward differentiated implementations of the CAD kernel and mesh generation, allows an efficient calculation of the sensitivities $J_\alpha \in \mathbb{R}^n$ of the objective function with respect to the CAD design parameters. A more detailed discussion of this framework can be found in (Verstraete et al., 2017).

Previous work used the open-source structural solver Calculix (Dhondt and Wittig, 1998) and an inverse distance interpolation for the structural grid generation (Verstraete et al., 2017). In this paper, an in-house adjoint structural solver is used, which is also used to morph the unstructured grid using a linear elastic analogy. Compared to the inverse distance method, the linear elastic analogy has shown to require less remeshing for this radial turbine geometry.

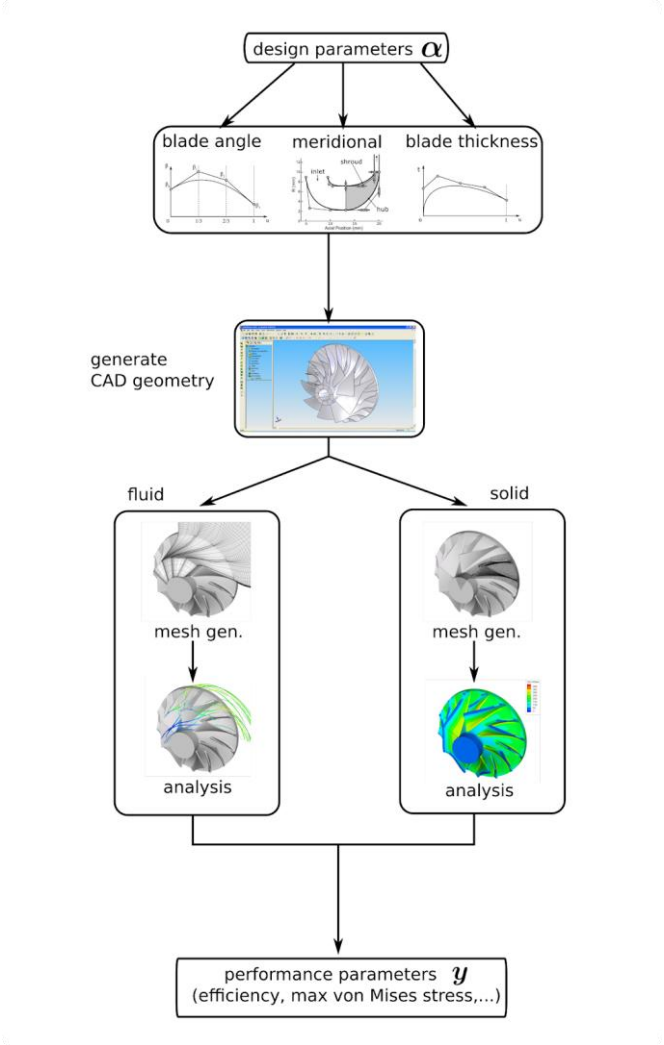

Figure 1 Flowchart of Multidisciplinary Framework in CADO (CAD-based Optimization), in-house optimization code of the von Karman Institute for Fluid Dynamics

Numerical Setup

The numerical setup for this optimization is based on the same setup as in (Verstraete et al., 2017), which will be briefly summarized in this section.

An adjoint MDO of a radial turbine is performed, using CAD design parameters to modify the geometry. In total, n = 24 design parameters are defined for the optimization. 11 design parameters are used to define the shape of the meridional passage (figure 2), 12 parameters define the blade angle distribution from leading to trailing edge of the hub and shroud (figure 3), and one parameter is used for the trailing edge cut back definition (figure 4).

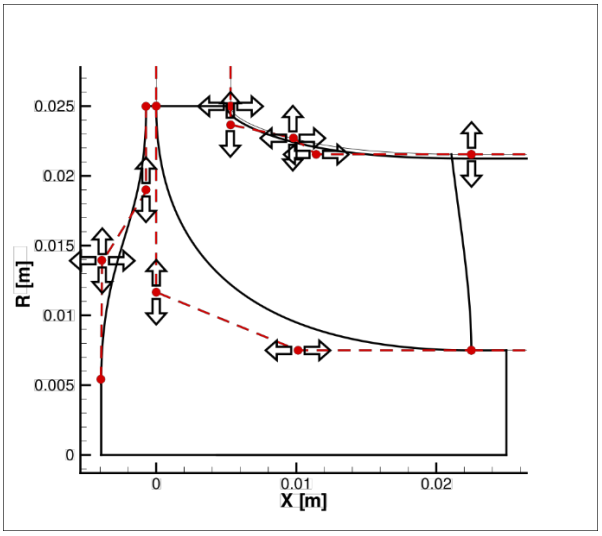

Figure 2 Parametrization for Meridional Shape of Turbine Wheel

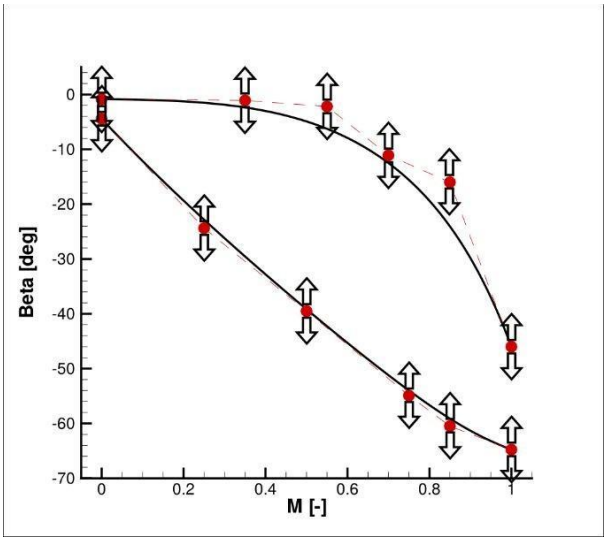

Figure 3 Parametrization for Blade Angle Distribution from Leading to Trailing Edge. Top: Hub. Bottom: Shroud

The master CAD geometry is generated based on the CAD design parameters and serves as the reference for the generation of the structured grid for the fluid domain and the unstructured grid for the solid domain (figure 5). The fluid domain is analyzed by solving the Reynolds-Averaged Navier-Stokes (RANS) equations on a multi-block structured grid and the solid domain is analyzed by solving the linear elastic equations using the finite element method (FEM) on an unstructured grid. Instead of using the open-source solver Calculix (Dhondt and Wittig, 1998) and an inverse-distance interpolation for the unstructured mesh deformation, an in-house structural solver is used for the stress analysis, as well as unstructured mesh deformation.

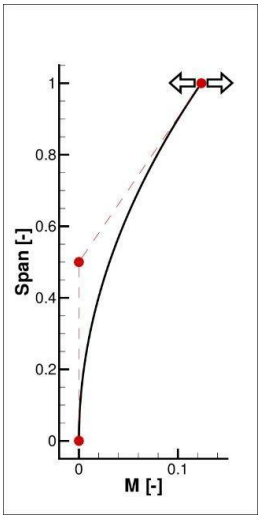

Figure 4 Parametrization for Trailing Edge Cut Back

Figure 5 Orange: Structured Grid for Fluid Analysis. Green: Unstructured Grid for Solid Analysis.

Flow Solver

The RANS flow solver used in this optimization is based on the cell-centered finite volume method, using a multi-block structured grid. An implicit time integration scheme with local time-stepping and geometric multigrid is used to converge towards a steady-state solution. Roe's approximate Riemann solver (Roe, 1981) is used to compute the inviscid fluxes. Second order accuracy is achieved using a MUSCL reconstruction (Van Leer, 1979) of the primitive variables. A van-Albada type limiter (Venkatakrishnan, 1993) is used to reduce shock oscillations and numerical dissipation is handled by the entropy correction of Harten and Hyman (Harten and Hyman, 1983). A central discretization scheme is used to compute the viscous fluxes. The negative Spalart-Allmaras turbulence model (Allmaras et al., 2012),

assuming a fully turbulent inflow is used for the turbulence closure problem. More details on the flow solver and its discrete adjoint implementation can be found in (Mueller and Verstraete, 2017).

Structural Solver

The structural solver used in this optimization is an in-house structural FEM solver, written in C++. The linear elastic equations are discretized on an unstructured grid of quadratic tetrahedral elements. The resulting linear system of linear elasticity is solved by iterative solvers provided by the Eigen library (Guennebaud et al., 2010), while the generalized eigenvalue problem of the vibration analysis is solved using the SLEPc library (Hernandez et al., 2005).

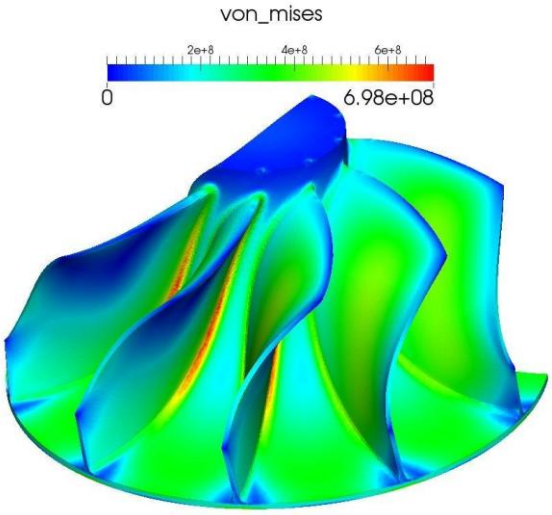

Figure 6 Distribution of von Mises Stress σ

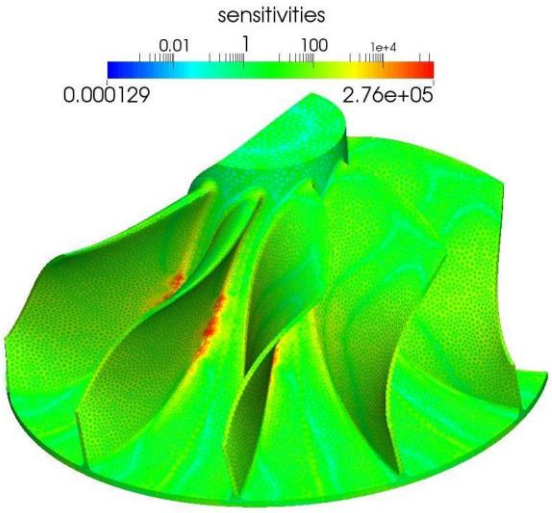

Figure 7 Sensitivities of Maximum von Mises stress with Respect to Node Coordinates $\left|\left|\left|\frac{\partial \sigma}{\partial^m x_i^{dx}}\right|\right|\right|$

The discrete adjoint model (Schwalbach et al., 2016) was implemented using the open-source algorithmic differentiation (AD) tool CoDiPack (Sagebaum et al., 2017). The differentiation of the linear system assembly and cost function evaluation are straightforward. However, the linear system solver requires special treatment. A black-box approach would cause its adjoint to use the same number of iterations and the same Krylov subspace. This may lead to wrong results, which will propagate through the adjoint model and influence the gradients. To avoid this, the adjoint model is treated in such a way that another linear system is setup to solve to for the adjoint variables (Tadjouddine et al., 2006). Due to the symmetric positive definiteness of the stiffness matrix, the same system matrix is used to solve the primal and adjoint linear systems (Schwalbach et al., 2017).

A forward and reverse (adjoint) run of the structural solver delivers the maximum von Mises stress (figure 6), as well as the required structural gradients (figure 7). This requires a computational cost of approximately two linear system solves using the same stiffness matrix, which corresponds to a cost of 2.1 times the primal solver run time. The solver has previously been used to successfully perform an adjoint structural optimization of a radial turbine geometry using a linear elastic mesh deformation.

Solid Mesh Deformation

The mesh deformation uses a hierarchical approach based on the CAD geometry to ensure conformity between the unstructured mesh and the updated CAD. First, the nodes found along the edges of the outer mesh are displaced according to the morphed CAD edges. Since the CAD edges are described by a B-spline curve, this is solved using a linear spring analogy in parametric space (figure 8). The remaining outer nodes are then displaced according to the morphed CAD surfaces. Analogously to the first step, the CAD surface is defined by B-spline surfaces, allowing a solution in parametric space using an inverse distance interpolation (figure 9). Using the outer node displacements as boundary conditions, the inner mesh nodes are then solved for using the linear elastic FEM solver (figure 10).

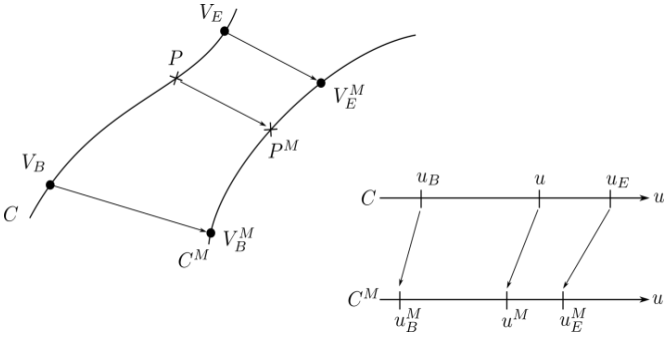

Figure 8 Morphing of Edge Nodes in Parametric Space

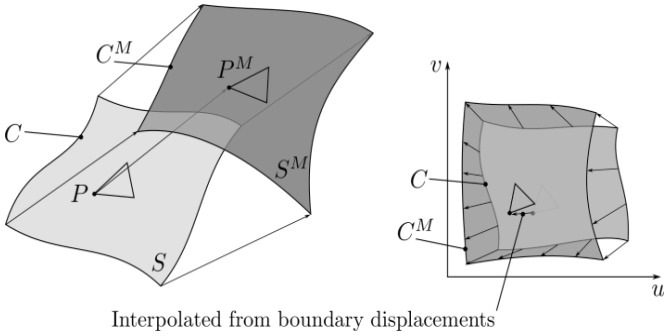

Figure 9 Morphing of Surface Nodes in Parametric Space

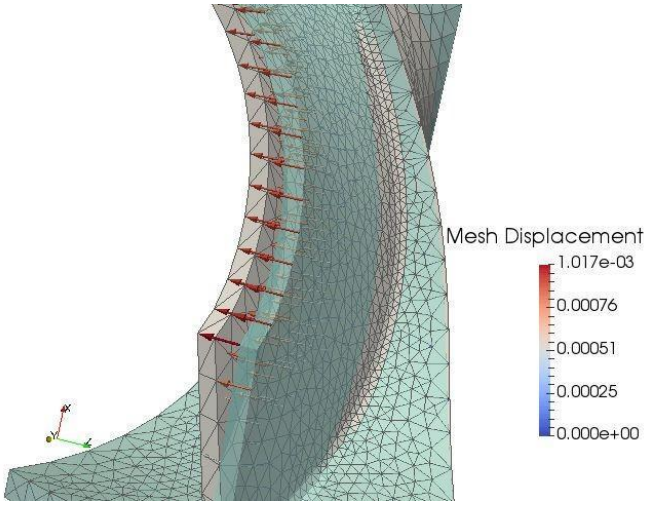

Figure 10 Morphing of Inner Mesh Points Based on Surface Node Displacements

Optimization Method

A steepest descent optimization method of the form

$$\alpha_{i+1} = \alpha_i - \lambda \circ \underline{dd}J\alpha \quad (1)$$

is used, where the CAD parameters αare used as the design parameters that modify the shape of the radial turbine. The step sizes for the different parameters are defined in the vector $\lambda \in \mathbb{R}^n$ and the $\circ$ operator represents the Hadamard
product such that

$$\lambda \circ \underline{dd}J\alpha = \left(\lambda_0 \cdot \underline{dd}_\alpha J_0, \ldots, \lambda_{n-1} \cdot \underline{dd}_{\alpha_{n-1}} J_{-1}\right)^T. \quad (2)$$

Usually, a scalar step size is used in a steepest descent algorithm. However, a vector of different step sizes is required due to different units being used for the individual CAD parameters α.

The total to static efficiency

$$\eta_{TS} = \frac{\frac{0,2}{0,1}_{TT} - 1}{\left(_{pp_{02,1}}\right)_{\gamma} - 1} \quad (3)$$

is selected as the performance parameter from the fluid domain. The subscript 1 denotes the inlet and 2 denotes the outlet of the fluid domain. Total pressure and total temperature are referred to by subscript 0. As a structural constraint from the solid domain, the maximum von Mises stress σ_{max} is required to be under a defined threshold σ_{req}. To ensure a smooth and differentiable maximum von Mises stress function, it is computed as the p- norm

$$\sigma_{max} = \sqrt[p]{\sum_{i=0}^{m-1} \sigma^p} \quad (4)$$

over the m FEM nodes. In this optimization, $p = 50$. The design goal and constraint are combined to formulate a single objective function which must be minimized. The objective
function is defined as

$$J = (1 - \eta_{TS}) + \omega \cdot S(\sigma_{max}) \cdot (\sigma_{max} - \sigma_{req})_2, \quad (5)$$

where ω is a penalty weight and the magnitude of the right-hand penalty term is controlled by the sigmoid-like function

$$S(\sigma_{max}) = \frac{}{1+e^{-(_{\sigma}m^1 ax - \sigma_{req})}}, \quad (6)$$

such that the penalty term smoothly reduces if the constraint is satisfied ($\sigma_{max} < \sigma_{req}$) and increases otherwise. The left-hand term of the objective function represents the goal of increasing efficiency.

In this work, a two-step optimization is performed. First, a coarse CFD mesh with around 750,000 mesh points is used. This is done to achieve an improved design within a short time frame. Starting at the 25th iteration, a finer mesh with around 1.4 million points is used to further improve the design with a higher numerical accuracy.

Optimization Results

Figure 11 shows the evolution of the objective function J, efficiency η_{TS}, and maximum von Mises stress throughout the optimization. The first 24 design iterations were performed using a coarser CFD mesh with around 750,000 mesh points. Wall clock run time for the first 24 iterations is around 3.5 days (84 hours), resulting in a 2.77% efficiency improvement.

	Baseline	Optimal
Efficiency [−]	0.746	0.776
Power [W]	1253.485	1377.786
Inlet Mass Flow [$^{g}_{s}$]	101.505	101.505
Outlet Mass Flow[$^{g}_{s}$]	101.495	101.495

Table 1 Value Comparison Between Baseline and Optimal Geometries

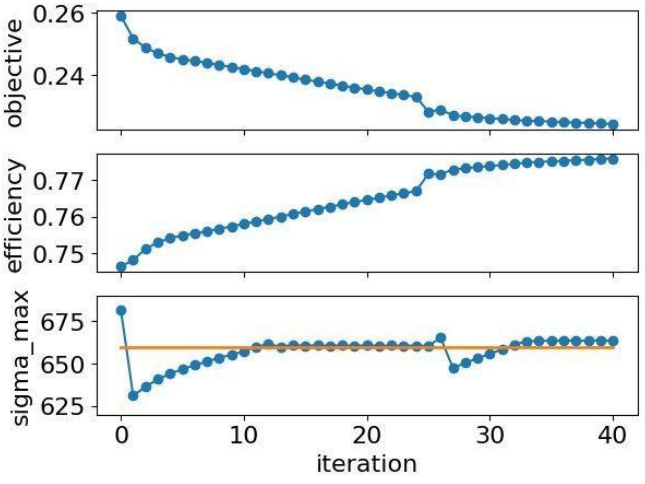

Figure 11 Blue: Evolution of Objective Function J, Efficiency η_{TS}, and Maximum von Mises Stress σ_{max}. Orange: Stress Threshold σ_{req}

At iteration 25, a finer CFD mesh with approximately 1.4 million mesh points is used until iteration 40, taking an additional 4 days (96 hours) to increase the efficiency further, resulting in a total improvement of 3.97%. Due to the higher accuracy, there is a sudden jump at this iteration in figure 11. The resulting maximum von Mises stress is within range of the defined constraint $\sigma_{req} = 660$ *MPa*.

Changes in the parameters that describe the blade angle distribution and meridional shape are exhibited in figures 12 and 13, where the control points are shown as well. Especially in the meridional plot, the differences between the baseline and optimal shapes are evident. The shape of the hub remains virtually unchanged while the shroud has been adjusted significantly. The inlet is slightly reduced, while the outlet is expanded as the blade is straightened in order to improve the total-to-static efficiency. The resulting initial and optimal geometries are shown in figures 14 and 15, respectively.

Additionally, the power has increased by 9.92%, because no aerodynamic constraint was imposed in the present optimization (table 1). A prescribed mass flow boundary condition is used to ensure that the mass flow at the inlet and outlets remain constant throughout the optimization. No remeshing was required for the solid mesh during the entire optimization.

The primal and adjoint CFD simulations are run in parallel on 6 Intel i7-4790K cores for optimal load-balancing. The forward-differentiated mesh generation and the adjoint structural solver are both executed on 8 cores.

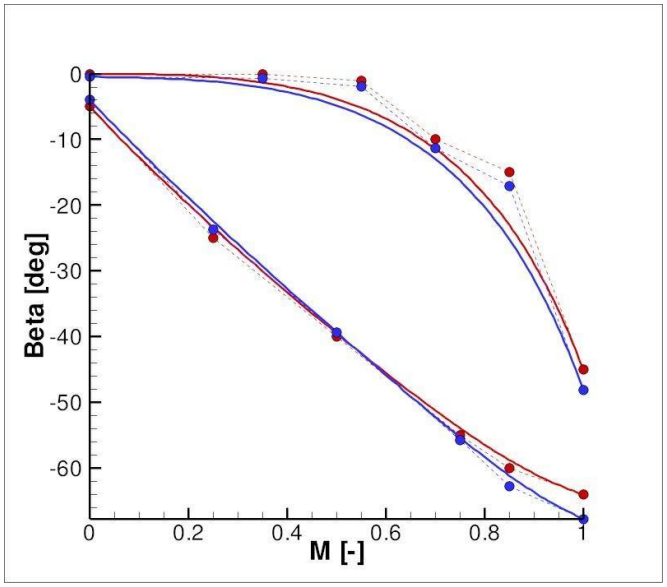

Figure 12 Blade Angle Distribution and Parametrization. Red: Baseline. Blue: Optimal

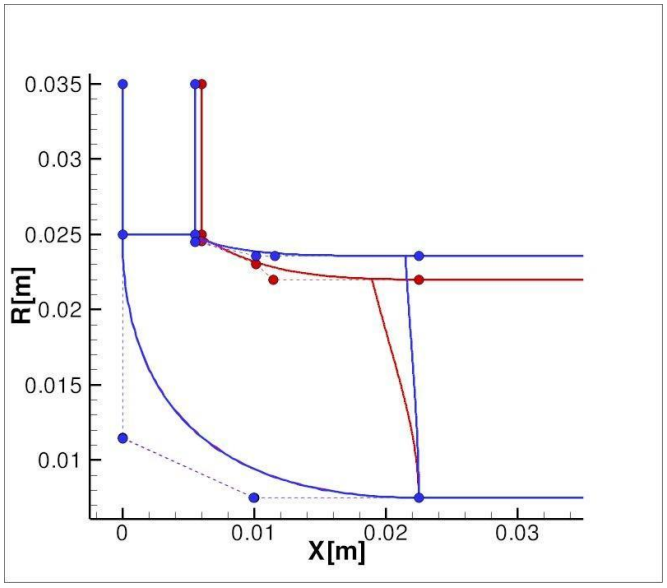

Figure 13 Meridional Shape and Parametrization.
Red: Baseline. Blue: Optimal

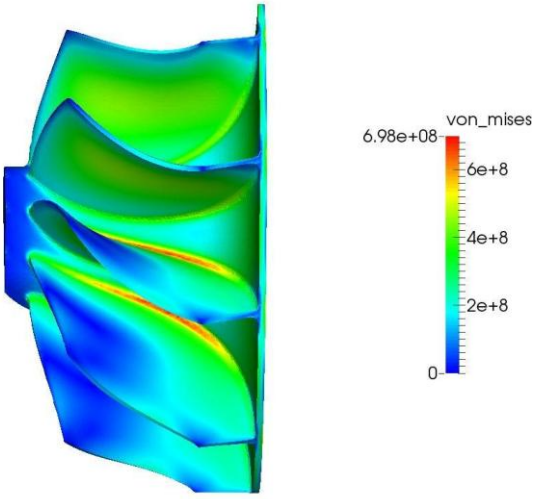

Figure 14 von Mises Stress Distribution of Initial Geometry

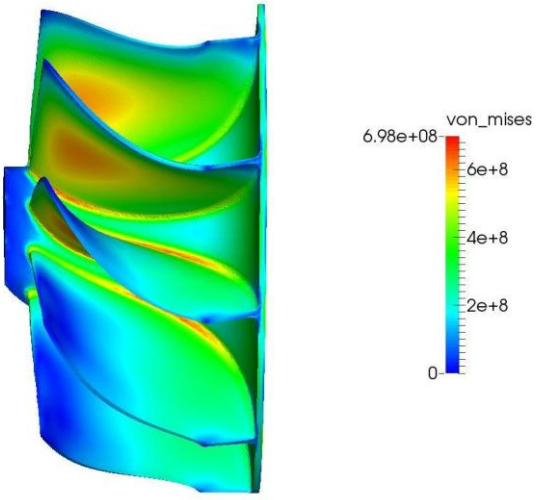

Figure 15 von Mises Stress Distribution of Optimized Geometry

Conclusion

This paper has introduced a CAD-based adjoint multidisciplinary optimization framework for turbomachinery components. An adjoint optimization of a radial turbine geometry was performed in two phases. The first phase used a coarser CFD mesh to achieve an efficiency improvement of 2.77% within 3.5 days. The second phase used a finer CFD mesh to further drive the improvement of efficiency η_{TS} up to 3.97% within an additional 4 days. The optimal geometry satisfies the structural constraints of keeping the maximum von Mises stress within the region of the threshold σ_{req} and the power increased by 9.92%.

Future work will integrate an adjoint vibration analysis into the MDO framework to perform an adjoint MDO under the structural constraints of stress, as well as vibration.

NOMENCLATURE

m number of FEM nodes
n number of design parameters
p pressure
y performance parameters
J objective
S sigmoid function
T temperature
α design parameters
γ heat capacity ratio
η efficiency
λ step sizes
σ von Mises stress
ω penalty weight

References

Allmaras S. R., Johnson F. T., and Spalart P.R. (2012). Modifications and clarifications for the implementation of the Spalart-Allmaras turbulence model. In Proceedings of the 7th International Conference on Computational Fluid Dynamics (ICCFD7-1902).

Dhondt G. and Wittig K. (1998). Calculix: a free software three-dimensional structural finite element program. MTU Aero Engines GmbH, Munich

Guennebaud G., Jacob B., et al. (2010). Eigen v3. http://eigen.tuxfamily.org

Harten A., Hyman J. M. (1983). Self-adjusting grid methods for one-dimensional hyperbolic conservation laws. Journal of Computational Physics 50(2), 235-269
Hernandez V., Roman J. E., and Vidal V. (2005). SLEPc: A scalable and flexible toolkit for the solution of eigenvalue problems. ACM Trans. Math. Software 31(3), 351-362

Jameson A. (1988). Aerodynamic design via control theory. Journal of Scientific Computing 3(3), 233-260

Luo J., Zhou C., and Liu F. (2014). Multipoint design optimization of a transonic compressor blade by using an adjoint method. Journal of Turbomachinery 136(5), 051005

Mueller L. and Verstraete T. (2017). CAD integrated multipoint adjoint-based optimization of a turbocharger radial turbine. International Journal of Turbomachinery Propulsion and Power 2(3), 14

Pironneau O. (1974). On optimum design in fluid mechanics. Journal of Fluid Mechanics 64 (01), 97-110

Roe P.L. (1981). Approximate Riemann solvers, parameter vectors, and difference schemes. Journal of Computational Physics 43(2), 357-372

Sagebaum M., Albring T., and Gauger, N. R. (2017). High-performance derivative computations using codipack. arXiv preprint arXiv:1709.07229

Schwalbach M., Verstraete T., and Gauger N. R. (2016). Developments of a discrete adjoint structural solver for shape and composite material optimization. In The 7th International Conference on Algorithmic Differentiation.

Schwalbach M., Verstraete T., and Gauger N. R. (2017). Adjoint optimization of turbomachinery components under mechanical constraints. In The 8th VKI PhD Symposium.

Tadjouddine M., Forth S. A., and Keane A. J. (2006). Adjoint differentiation of a structural dynamics solver. Automatic Differentiation: Applications, Theory, and Implementations, Springer, 309-319

Van Leer B. (1979). Towards the ultimate conservative difference scheme. V. A second order sequel to Godunov's method. Journal of Computational Physics 32(1), 101-136

Venkatakrishnan V. (1993). On the accuracy of limiters and convergence to steady state solutions. In Proceedings of the 31st AIAA Aerospace Sciences Meeting.

Verstraete T. (2010). Cado: a computer aided design and optimization tool for turbomachinery applications. In 2nd Int. Conf. on Engineering Optimization, 6-9

Verstraete T., Müller L., and Müller J.-D. (2017). Multidisciplinary adjoint optimization of turbomachinery components including aerodynamic and stress performance. In 35th AIAA Applied Aerodynamics Conference, 4083

Walther B. and Nadarajah S. (2013). Constrained adjoint-based aerodynamic shape optimization of a single-stage transonic compressor. Journal of Turbomachinery 135(2), 021017

Wang D. X. and He L. (2010). Adjoint aerodynamic design optimization for blades in multistage turbomachines - part I: methodology and verification. Journal of

Turbomachinery 132(2), 021011