



Title:

**Word Template for CADxx Extended Abstracts**

Authors:

John C. Gotti, gotti@cityuny.edu, City University of New York

Jessie A. James, Jessie@tsu.edu, Texas State University

Donald L. Corleone, don@dlc.com, 3DLC and Associates

Keywords:

Rapid Prototyping, FDM, Z-Printer, Surface Finish, Dimensional Accuracy, Process Optimization

DOI: 10.14733/cadconfP.2024.xxx-yyy

Introduction:

In engineering workflows it is common practice to maintain master CAD geometries, which then serve as a foundation for further design and development. Preserving CAD parametrisations during aerodynamic shape optimisation becomes challenging since existing CAD tools do not provide derivatives necessary for gradient-based design methods. Hence, gradients are obtained by approximations or with simplifications, precluding full CAD integration into the design optimisation loop.

In the present work, we obtain exact CAD sensitivities (derivatives) with respect to CAD design parametrisation using our algorithmically differentiated Open Cascade Technology (OCCT) CAD-kernel software. The extension of this software for optimisation of parametric models and BRep (NURBS) compose the tool for explorations of the optimal shape in different by size and nature design spaces. In addition, we demonstrate the imposition of geometric constraints for both approaches, a salient part of industrial design, and an intuitive method of storing them in standard CAD format.

CAD-based Shape Optimisation:

To assess the aerodynamic performance of a given CAD geometry with design parameters  $\alpha$ , one usually computes a scalar cost function  $J$  (drag, lift, total pressure loss, etc.) on the corresponding computational mesh. In order to obtain a shape with improved characteristics we consider an optimisation problem [2]:

$$\min_{\alpha} J(U(X(\alpha)), X(\alpha), \alpha) \quad (2.1)$$

$$R(U(X(\alpha)), X(\alpha)) = 0. \quad (2.2)$$

Equation (2.2) describes the flow field within the domain of interest by system of Reynolds-Averaged Navier-Stokes equations, with the state variable  $U$  and computational mesh coordinates  $X$ , which depend on design parameters  $\alpha$ . In case of large amount of design parameters (usually the case in industrial applications) the adjoint method proves to be computationally efficient and could be derived by application of a chain rule to the system (2.1)-(2.2) yielding:

$$\frac{dJ}{d\alpha} = \left[ \frac{dJ}{dX} + \nu^T f \right] \frac{\partial X}{\partial \alpha}, \quad (2.3)$$

where

$$f = -\frac{\partial R}{\partial X}. \quad (2.4)$$

Here  $\nu$  represents the solution of adjoint equations:

$$\left(\frac{\partial R}{\partial U}\right)^T \nu = \frac{\partial J}{\partial U}. \quad (2.5)$$

After computing the solution of primal and adjoint equations (2.2),(2.5), one can rewrite cost function gradient in terms of surface grid points derivatives:

$$\frac{dJ}{d\alpha} = \frac{dJ}{dX_S} \frac{dX_S}{d\alpha}. \quad (2.6)$$

Here, the relation (spring analogy, inverse distance weighting) between volume and surface grid points displacement is used  $X = X(X_S)$ . The first term in (2.6), usually called *CFD sensitivity*, corresponds to the flow sensitivity in the surface grid points  $X_S$ . These derivatives could be calculated by several available CFD solvers that have implemented the adjoint method. In this work we use our in-house discrete adjoint solver STAMPS (previously *mgopt*) [3].

The second term (*CAD sensitivity*) represents the derivative of the surface grid points  $X_S$  with respect to the CAD model design parameters. This part is calculated in the automatically differentiated version of OCCT [1]. Although the process of differentiation of the complete CAD system was time-consuming and involved comprehensive code modifications (due to the size and complexity of the software), the final result is extremely beneficial for the CAD-based optimisation. The differentiated OCCT provides the derivatives for almost every possible CAD parametrisation and geometrical manipulation.

Equipped with these derivatives, we compose them in the total gradient, which is then used in iterative gradient-based optimisation loop:

$$\alpha^{(n+1)} = A(\alpha^{(n)}, \frac{dJ}{d\alpha}(\alpha^{(n)})), \quad (2.7)$$

with  $A$  as an optimisation algorithm. Next sections describe two cases of the above mentioned method, depending on the nature of CAD design parameter  $\alpha$ : as design variable in parametric CAD model or BRep/NURBS parametrisation.

#### Parametric CAD-model Optimisation:

Parametric models are extremely popular in industry since they allow designers and engineers to map their intuition and experience on a set of familiar and conventional variables: e.g. blade thickness, fillet radius, wing span, etc. The OCCT kernel proposes large amount of methods for typical parametric model construction: starting from sketches with pre-defined sizes, construction of 2D curves to further 3D shapes manipulations, boolean operations, etc. The software architecture of the differentiated OCCT allows to build parametric models with no difference to the original version, but additionally is capable of providing the derivatives with respect to the used design parameters.

Existence of a parametric CAD model creates a natural way to incorporate various geometrical constraints, since most of design variables are explicitly controlled by user-defined parametrisation. In the final paper, an example of parametric CAD model in differentiated OCCT will be presented (TU Berlin Stator testcase, see description below), as well as the results of aerodynamic shape optimisation subject to several geometrical constraints.

#### BRep (NURBS-based) Optimisation and Constraints:

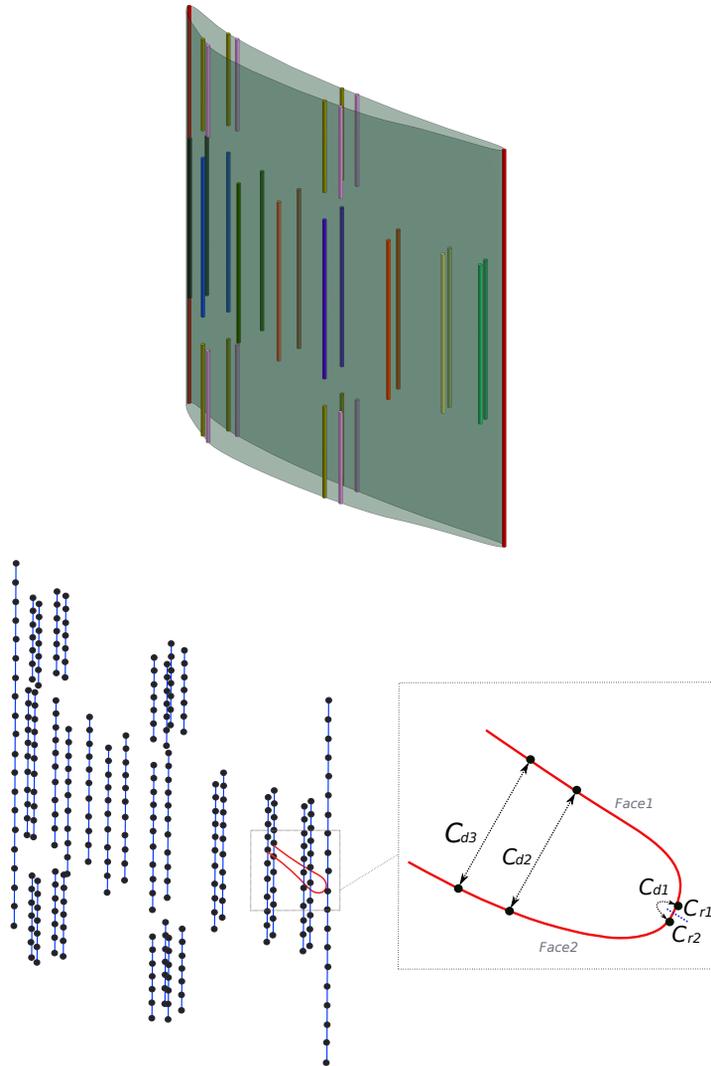


Fig. 1: Left: Constraint visualisation from the STEP file; Right: Constraints computation on the test-points

Alternatively to the previous section, instead of changing the parameters of the model's construction algorithm, one can directly modify the geometry of the resulting shape, so-called BRep (Boundary Representation). Changes to this BRep data (control points positions and weights of corresponding NURBS) eliminate the initial parametrisation, but propose richer design spaces compared to sometimes limited parametric models. Hence, this method serves better the ultimate goal of optimisation, potentially exploring non-conventional designs. The method is CAD-vendor independent, and requires only a generic CAD-file (STEP, IGES, etc.), eluding problems with parametrisation tree and making the optimisation more automatic. With method implementation in OCCT, one can easily and intuitively refine design space by adding extra control points.

CAD models are usually constructed from multiple adjacent patches. Therefore, modifications of control points individually on patches can violate (i) patch-continuity (holes between the CAD faces, non-smooth shapes) or (ii) other geometrical constraints. We alleviate this problem by filtering out the shape modes with undesired constraints violations using discrete spaces constructed using test-points (NSPCC approach) [5],[4]. Conceptually, the approach requires that the constraints are satisfied on the particular set of points defined on the surface (test-points).

In this work, we propose further development of NSPCC for user-friendly constraints definition. Several methods were devised to accelerate and automate the process of test-point distribution and visualise them. In Fig. 1. the constraints are deployed in pairs and stored in a coloured STEP format. On each coloured pair, a number of test-points are distributed by means of OCCT, and then composed in the constraint matrix as in [4]. Making use of the OCCT functionality and its differentiated version, the following types of constraints are implemented and also presented in Fig.1.:

- Cross-patch continuity ( $C_{d1}$ )
- Distance ( $C_{d2}, C_{d3}$ )
- Distance in X, Y, or Z direction
- Radius of curvature in some point ( $C_{r1}, C_{r2}$ )

In the next section, the results of NURBS-based optimisation with above-mentioned constraints are shown.

#### Results for the TU Berlin Stator Testcase:

The complete description of TU Berlin Stator testcase could be found via <http://aboutflow.sems.gmul.ac.uk/events/munich2016/benchmark/testcase3/>. This case represents a typical turbomachinery optimisation problem, subject to a number of geometrical constraints. The corresponding constraint file is shown in Fig. 1. and includes minimal thickness constraint in the middle of the blade, spaces for four bolts (two at the hub and two at the shroud sides), axial chord constraint (Distance in X) and minimal radius for the trailing and the leading edge (Radius of curvature constraint).

With all constraints set, the optimisation is conducted with NSPCC approach to minimise the total pressure loss in the stator at the operating point with 42 degrees of swirl. The results are provided in Fig. 2. and were obtained on the coarse mesh with the STAMPS solver and differentiated OCCT. The changes of NURBS control point net resulted in partial shrinking of the blade, while NSPCC methodology prevented an infraction of the constraints, in general providing 18% improvement to the cost function.

#### Conclusions:

The use of differentiated CAD-kernel OCCT in combination with adjoint CFD method is showcased for the aerodynamic shape optimisation. The developed CAD tool allows to find optimised designs and impose geometrical constraints for both parametric models and generally richer NURBS parametrisation spaces with result directly in the CAD format. In the final paper NURBS-based optimisation will be enhanced by automatic design space refinement and it will be compared to a parametric optimisation where the design space consists of conventional turbomachinery parameters.

#### Acknowledgement:

This research is a part of the IODA project - Industrial Optimal Design using Adjoint CFD. IODA is *Marie Skłodowska-Curie Innovative Training Network* funded by European Commission under Grant Agreement No. 642959.

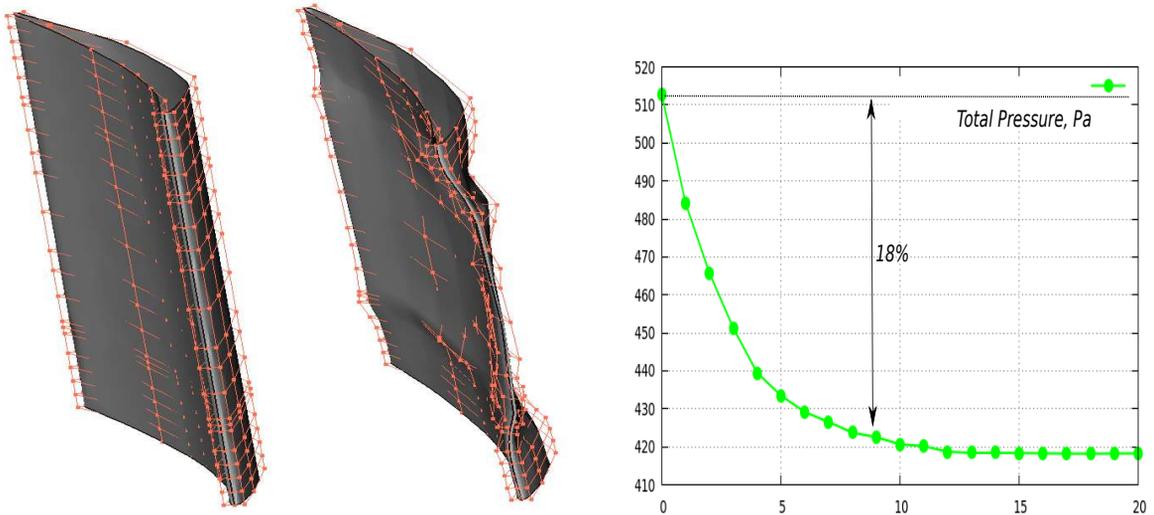


Fig. 2: Left: Original and Optimised TUB stator blade with NURBS Control Point Net; Right: 20 iterations of Optimisation loop

John C. Gotti, <https://orcid.org/aaaa-bbbb-cccc-dddd>

Jessie A. James, <https://orcid.org/aaaa-bbbb-cccc-dddd>

Donald L. Corleone, <https://orcid.org/aaaa-bbbb-cccc-dddd>

#### IMPORTANT NOTES:

WE ARE REQUIRED BY CROSSREF TO INCLUDE HYPERLINK INTO EVERY REFERENCE THAT HAS DOI LINKING. PLEASE GO TO: <http://www.crossref.org/SimpleTextQuery/>, REGISTER YOUR E-MAIL, CUT AND PASTE THE LIST OF REFERENCES INTO THE BOX, GET THE DOI LINKS AND PASTE THEM INTO YOUR PAPER! THE LINKS SHOULD LOOK LIKE THIS, I.E. USE HTTPS AND DROP THE DX: <https://doi.org/10.1080/16864360.2014.881190>

#### References:

- [1] Auriemma, S.; Banovic, M.; Mykhaskiv, O.; Legrand, H.; Müller, J.-D.; Walther, A.: Optimisation of a U-bend using CAD-based adjoint method with differentiated CAD kernel, ECCOMAS Congress, 2016.
- [2] Jameson, A.: Aerodynamic Design via Control Theory, Journal of Scientific Computing, 3, 1988, 233–260. <https://doi.org/10.1007/BF01061285>
- [3] Xu, S.: CAD-based CFD shape optimisation using discrete adjoint solvers, Ph.D. Thesis, Queen Mary University of London, 2015.
- [4] Xu, S.; Radford, D.; Müller, J.-D.: CAD-based adjoint shape optimisation of a one-stage turbine with geometric constraints, ASME Turbo Expo, 2015.
- [5] Xu, S.; Wolfram, J.; Müller J.-D.: CAD-based shape optimisation with CFD using a discrete adjoint, International Journal for Numerical Methods in Fluids, 74(3), 2013, 153-168. <https://doi.org/10.1002/flid.3844>