Partially Reduced SQP Methods for Optimal Turbine and Compressor Blade Design

Wolfgang Egartner  
Interdisciplinary Center for Scientific Computing  
University of Heidelberg  
Im Neuenheimer Feld 368  
D-69120 Heidelberg, Germany  
E-mail: Wolfgang.Egartner@iwr.uni-heidelberg.de

Volker H. Schulz  
Institute for Computer Applications  
University of Stuttgart  
Pfaffenwaldring 27  
D-70569 Stuttgart, Germany  
E-mail: volker@ica3.uni-stuttgart.de

In this paper we present an algorithm for turbomachinery optimal blade-to-blade (S1-streamsurface) design over a full working range. We formulate the design task as a constrained boundary control multiple setpoint optimization problem in partial differential equations and develop a partially reduced SQP (PRSQP) algorithm that makes way for an efficient parallel implementation. We present numerical results based on a 2D coupled Euler/boundary-layer solver that is widely used in engineering practice.

1 Introduction

Optimal turbomachinery design not only requires aerodynamic, heat transfer, aeromechanical, mechanical, manufacturing, time and budget constraints to be satisfied simultaneously. In order to move to truly integrated multidisciplinary design, also speed is of the essence as any step performed in a trial and error manner is consuming days of highly expensive working power [20, 9].

Therefore we present a new optimization approach to the $S1$-streamsurface (blade-to-blade) design problem that will help to reduce turnaround times. It is based on promising experiences in a pilot project where the same approach has been taken aiming at a desired Laval number distribution for the inviscid flow regime [16, 17]. With the algorithm presented here we are driving at a fully automized tool for coupled viscous/inviscid $S1$-streamsurface flow that helps the designer to minimize total pressure loss or to maximize efficiency for the 3D-blade while preserving solidness, manufacturing and heat engineering constraints.

To reach this goal via the $S1$-streamsurface approach, solutions are needed
that are optimal for a prescribed working range, e. g. over a variety of inlet flow angles or Mach numbers, and over a family of streamsurfaces that is large enough to prescribe the 3D-blade.

For that purpose, we first set up a single operating point problem for a single streamsurface which already incorporates 2D-versions of the essential constraints (like blade profile curvature or area). Then we develop a PRSQP optimization algorithm for that problem that efficiently expands to the more complex quasi-3D working range problem via a multiple setpoint formulation.

In this paper, we present the kernel of the algorithm and show how it expands to the working range optimization problem.

2 NLP formulation (single/multiple operating point)

We state the single operating point problem as

$$\min_{x,p} f(x,p), \quad f : \mathbb{R}^n \to \mathbb{R},$$

$$\text{s. t. } c(x,p) = 0, \quad c : \mathbb{R}^n \to \mathbb{R}^{m_c},$$

$$g(p) \leq 0, \quad g : \mathbb{R}^{m_p} \to \mathbb{R}^{m_g},$$

where $x \in \mathbb{R}^{m_c}$, $p \in \mathbb{R}^{m_p}$, $n = m_c + m_p$, $m_p \ll m_c$ and $\nabla_x c$ has full rank.

In this setup, $x$ represents the flow variables (i. e. density, node positions of the grid and boundary layer variables) and $p$ the controls (geometry parameters), which are quintic Bezier spline parameters. The cost functional $f$ is an appropriate total pressure loss coefficient. The coupled viscous/inviscid system of discretized flow equations is denoted by $c(x,p) = 0$, and the inequality $g(p) \leq 0$ represents geometry constraints like upper bounds on blade profile area and momentum of inertia, upper bounds on curvature at the leading edge and lower bounds on leading and trailing edge thicknesses.

3 Flow solution

Where the inviscid computations in the pilot project have been based on a conservative streamfunction solver [5] we now use the coupled viscous/inviscid solver ISES [2, 4]. It uses a conservative streamline curvature discretization of the Euler equations, which is coupled to the boundary layer models via the displacement thickness approach [11].

The coupled nonlinear system of discretized equations is solved by a Newton method [7, 3] and the linear system to be solved at each Newton step is attacked with a direct sparse solver.

The grid refinement one needs is case dependent. It ranges from about $150 \times 20$ to $300 \times 40$ so that the dimension of $x$ is between 10,000 and 40,000.
The number of profile parameters $p$ depends on the parameters used in the industrial design system. As these design parameters would yield very badly conditioned or even ill-posed optimization problems (different parameterizations may give the same blade!) we use special B-Spline parameters which span a subspace of the design parameter space. In our case, the number of parameters usually lies between 20 and 40.

4 Single operating point algorithm

SQP methods, which date back to [8, 14, 19] solve the nonlinear system of equations evolving from the nonlinear Karush-Kuhn-Tucker equations (first order necessary conditions for constrained nonlinear programming)

$$
\nabla f^T = \nabla c^T \lambda + \nabla g^T \mu \\
diag(\mu)g = 0 , \quad \mu \leq 0 \\
c = 0
$$

by means of a slightly modified Newton or Quasi-Newton iteration. Thus a sequence of quadratic programs (QPs) is generated with solutions $\Delta x$ and $\Delta p$ defining increments to be added to the current iterate.

As the number of controls (the B-spline parameters of the profile) is small compared to the number of flow variables (see above) we split the step into two components,

$$
\begin{pmatrix}
\Delta x \\
\Delta p
\end{pmatrix} = \begin{pmatrix}
-c_x^{-1}c_p \Delta p \\
\Delta p
\end{pmatrix} + \begin{pmatrix}
-c_x^{-1}c \\
0
\end{pmatrix},
$$

of which the first one lies in the nullspace of the flow equations and the other in its complement. So we use update formulas in order to produce approximations $B$ directly of the Hessian’s projection onto the nullspace of the linearized constraints,

$$B \sim \nabla_{pp}L - c_p^T c_x^{-T} \nabla_{xx}L c_x^{-1} c_p,$$

with $L$ denoting the Lagrangian of the optimization problem. Early discussions about this subject can be found in [6, 12, 13]. A selection of recent works providing theoretical as well as numerical aspects is [1, 10, 18, 15].

The system resulting from our approach looks like

$$
\begin{align*}
\min_{\Delta p} & \quad \frac{1}{2} \Delta p^T B \Delta p + \gamma^T \Delta p \\
\text{s. t.} & \quad \nabla g \Delta p + g \leq 0,
\end{align*}
$$
where $\gamma = \nabla_p f - \nabla_p c^T (\nabla_a c^T)^{-1} \nabla_a f$ is the reduced gradient. With the QP solution $\Delta p$, and $\Delta x$ according to (5), we iterate

$$x_{k+1} = x_k + \alpha_k \Delta x , \quad (7)$$

$$p_{k+1} = p_k + \alpha_k \Delta p , \quad (8)$$

where $\alpha_k$ is chosen in a way that some merit function is reduced. In order to guarantee locally superlinear convergence of the algorithm it is important that $B$ approaches the Hessian of the Lagrange function in the solution point of problem (1).

The constraints on the control variables are stated explicitly within the reduced problem so that in contrast to ordinary reduced SQP methods a small quadratic program remains. So we denote the resulting algorithm as partially reduced SQP or, in short, PRSQP.

5 Working range expansion

One major benefit of the approach described above is that the main cost of the computation of each step is packed into the computation of the step in flow variables (Newton step for the flow equations) and the evaluation of the reduced gradient (adjoint system). Consequently, whenever one has to optimize several flow problems at a time, the corresponding forward and adjoint solution steps can be carried out simultaneously in a parallel computing environment.

We want to explain this benefit looking at the working range validation problem, which writes

$$\min_{x_i \in \mathbb{R}^n, \ p \in \mathbb{R}^m} \sum_{i=1}^{N} \omega_i f(x_i, p) , \quad f : \mathbb{R}^n \to \mathbb{R} ,$$

s. t. $c_1(x_1, p) = 0 , \quad c_1 : \mathbb{R}^n \to \mathbb{R}^m$,

$$\vdots$$

$c_N(x_N, p) = 0 , \quad c_N : \mathbb{R}^n \to \mathbb{R}^m$,

$g(p) \leq 0 , \quad g : \mathbb{R}^m \to \mathbb{R}^m$, \quad (9)

where $c_i(x_i, p) := c(x_i, p, q_i)$ with the $q_i$ representing the operating points (e. g. inlet Mach numbers or flow angles). Again, $\nabla_a c$ is supposed to have full rank for all operating points.

From the above structure, the linearized system of flow equations has the
so that the forward and adjoint systems of diverse operating points can be computed independently. In order to compute the reduced gradient $\gamma$ for the working range problem, the master process has just to combine the reduced gradients

$$
\gamma_i = \nabla_p f_i - \nabla_p c_i^T (\nabla_x c_i^T)^{-1} \nabla_x f_i
$$

(11)

to

$$
\gamma = \sum_{i=1}^n \omega_i \gamma_i
$$

(12)

where $f_i = f(x_i, p)$.

### 6 Numerical results

In the first step of the project (single operating point, single blade) the goal has been to make sure that the basic algorithm converges smoothly for a variety of test cases. Moreover, it has to automatically produce practically useful results.

As an example, we want to show a computational result for a turbine rotor blade. While in that case the initial profile is already considered quite satisfactory, we can drive the total pressure loss coefficient from 0.024 to 0.019.

This is considered a significant reduction by our industry partners, and, what is more, the imposed geometry constraints $g$ guarantee that this solution is still practical.

Looking at figure 1, one can see, for instance, that the leading edge curvature is kept within reasonable bounds, which is an important manufacturing constraint.

In addition, the trailing edge thickness is maintained. As we have observed very often, the thickness of the trailing edge tends to be reduced in order to minimize the total pressure loss. This, however, would be impractical for various reasons.

Another important bound is a minimal area (to be able to preserve solidness constraints). As smaller blades in general give smaller losses, also this constraint is active at the solution point of our example.
Figure 1: ISES blade profile (blunt trailing edges are left open, the Kutta condition is applied between the endpoints while losses are computed with a special model), optimized with respect to total pressure loss preserving leading edge curvature, trailing edge thickness and profile area bounds.

Given the initial blade and starting from scratch with a $157 \times 20$ grid, 20 PRSQP iterations are needed to achieve this result. The whole run takes about 5 minutes on a Macintosh PPC 604e with 180 MHz (RAM requirement is about 30 MB), which is about four times the computational effort for a single forward solution.

Future developments include further implementations with respect to working range and quasi-3D extensions. The inherently parallel structure of the corresponding algorithms should yield a new, very efficient system for preliminary quasi-3D working range prototyping.

Acknowledgements

This work has been supported by ABB Power Generation Ltd, MTU Motoren-und Turbinen-Union München GmbH and the Federal Ministry of Education, Science, Research and Technology (BMBF) under grant number 032741A within the German research effort AG Turbo.
References


