ISABE2009-1225

COMPUTATION OF SECOND-ORDER SENSITIVITIES OF THE TOTAL PRESSURE LOSSES FUNCTIONAL IN CASCADE FLOWS

T. Zervogiannis, D.I. Papadimitriou and K.C. Giannakoglou
National Technical University of Athens
School of Mechanical Engineering
Lab. of Thermal Turbomachines
P.O. Box 64069, 15710 Athens, Greece
kgianna@central.ntua.gr

Abstract

The minimization of total pressure losses in a compressor cascade flow, using the SQP method and the exact Hessians of both the objective function and constraints, is presented. The direct differentiation of the governing equations computes first-order sensitivities and, then, the second-order sensitivities are derived using the adjoint method. This is, in fact, much more efficient compared to all other combinations of the aforementioned techniques yielding also exact Hessians. In our problem, the objective function corresponds to the difference in total pressure between the cascade inlet and outlet. The flow turning through the cascade airfoil is constrained. Apart from the exact Newton method, a quasi Newton method, in which the Hessian matrix is computed exactly at the first cycle and then approximately updated, is also implemented.

Nomenclature

b	Design variable
F	Objective function
c	Constraint
p_t	Total pressure
a	Flow angle
t	Blade thickness
z	SQP slack variable
M	Mach number
R	Residual operator
Re	Reynolds number based on chord
λ, μ	Lagrange multipliers
σ	Centering parameter
γ	Cascade stagger angle
s	Solidity
x, y	Cartesian coordinates
$AVDR$	Axial velocity density ratio

Subscripts

in, out	Flow domain inlet/outlet
is	Isentropic
tar	Target

Superscripts

t	thickness constraint
a	flow angle constraint

Introduction

The adjoint method has been widely used in aerodynamic shape optimization problems to support gradient-based optimization algorithms. Its major advantage is that the CPU cost is independent of the number of the design variables. However, the efficiency of the optimization algorithm itself depends on the conditioning of the problem, which can be roughly assessed by the properties of the Hessian matrix. It is well known that gradient-based methods perform poorly in ill-conditioned problems. This can be alleviated by either introducing information from the exact Hessian or switching entirely to a Newton method. Below, the term exact Newton method will be used to denote a Newton method relying on exactly computed Hessian matrices. Note that formulas for the Hessian matrix which do not take into account some terms which (in this paper) proved to be negligible, are referred to as "exact" Hessian matrices and the corresponding descent method will be referred to as "exact" Newton method. In contrast, optimization methods that make use of approximations to the exact Hessian matrix will be referred to as "quasi Newton" methods.

In the CFD community, works addressing the issue of computing exact Hessians in aerodynamics are limited. [12], [14], and [4] rely on automatic differentiation (AD) tools and various combinations of discrete adjoint techniques and direct differentiation of the flow equations to compute second-order sensitivities of a functional related to geometric and flow input variables. In [6], this was also extended to the computation of second-order sensitivities on a parallel computer. [1], [2] and [3] proposed a preconditioning technique that introduces second-order information by applying Fourier analysis and uses the so-obtained Hessian symbol.

In previous works by the same group, [7], [8], [9], [10] the authors have computed the exact Hessian for the objective function associated with the inverse design of 2D airfoil shapes; this was achieved through the combined use of direct differentiation of the state equations and adjoint techniques, in both continuous and discrete form. These studies concluded that the direct-adjoint method is the one with the lowest CPU cost. This approach is employed herein for the minimization of total pressure losses in a 2D compressor cascade subject to geometric and flow constraints. The direct differentiation and the discrete adjoint method are implemented using exclusively hand differentiation. Compared to AD techniques which could be used instead, hand differentiation, maximizes code efficiency and economy in memory. Furthermore, the method provides a mathematical framework to support robust design applications, by computing the exact Hessian with respect to flow input variables.

Optimization Problem – Mathematical Background

The scope of this study is to minimize the total pressure losses in turbomachinery cascade flows, maintaining the average flow turning at a desired value and the airfoil thickness (at several chordwise positions) greater than some user-defined minimum values. This problem can be interpreted as

$$\begin{gathered}\text{minimize } F=\int_{in} p_t dS-\int_{out} p_t dS \\ \text{subject to} \\ c^a=\frac{1}{2}\int_{out}\left(a_{out}-a_{tar,out}\right)^2 dS=0 \\ \text{and } c_r^t=t_r-t_{tar,r}\geq 0\end{gathered}$$

Here b_i, i=1,…,N is the vector of Bézier control points parameterizing the cascade airfoil, U_m, m=1,…,M are the flow variables over the grid points and $t_{tar,r}$, r=1,…,R are the minimum allowed blade thicknesses at certain chordwise positions (subscript r).

The Lagrangian function of the design problem is defined as

$$\mathcal{L} = F - \lambda_r^t c_r^t - \lambda^a c^a$$

and is used to define the Karush-Kuhn-Tucker (KKT) optimality conditions, [5], for the above problem

$$\frac{d\mathcal{L}}{db_i} = 0$$
$$c_r^t - z_r^t = 0$$
$$c^a = 0$$
$$\lambda_r^t z_r^t - \sigma\mu^d \geq 0$$
$$\mu^d = \sum_r \frac{1}{R} z_r^t \mu_r^t$$

Slack variables z_r^t are introduced as extra optimization parameters in order to convert the thickness inequality constraints to equality ones. The Newton method can be applied to the KKT conditions. This is proved, [5], to be equivalent to defining a quadratic optimization subproblem in each step, to be solved for the subproblem variables d_i and Lagrange multipliers μ_r^t and μ^a; such a problem reads

$$\begin{aligned} \text{minimize } & \tfrac{1}{2} d_i \tfrac{d^2\mathcal{L}}{db_i db_j} d_i + \tfrac{dF}{db_i} d_i \\ \text{subject to } & \tfrac{dc^a}{db_i} d_i = 0 \\ \text{and } & \tfrac{dc_r^t}{db_i} d_i - z_r^t = 0 \end{aligned}$$

This is the so-called Sequential Quadratic Programming (SQP) method. in which all subproblems are solved using the Newton method for d_i, μ_r^t and μ^a

$$\begin{bmatrix} \frac{d^2\mathcal{L}}{db_i db_j} & -\frac{dc_l^t}{db_i} & -\frac{dc^a}{db_i} & 0 \\ \frac{dc_r^t}{db_j} & 0 & 0 & -\delta_{rl} \\ \frac{dc^a}{db_j} & 0 & 0 & 0 \\ 0 & z_r^t\delta_{rl} & 0 & \mu_r^t\delta_{rl} \end{bmatrix} \begin{bmatrix} \Delta d_j \\ \Delta\mu_l^t \\ \Delta\mu^a \\ \Delta z_l \end{bmatrix} = \begin{bmatrix} \frac{d^2\mathcal{L}}{db_i db_j} d_j + \frac{dF}{db_i} - \frac{dc_r^t}{db_i}\mu_r^t - \frac{dc^a}{db_i}\mu^a \\ \frac{dc_r^t}{db_i} d_i - z_r^t \\ \frac{dc^a}{db_i} d_i \\ z_r^t \mu_r^t - \sigma\mu^d \end{bmatrix}$$

with μ_r^t and z_r^t be strictly positive and R be the number of airfoil thickness constraints. The solution to the subproblem is used to update b_i, λ_i^t and λ^a as follows:

$$d_i - (b_i)^{n+1} - (b_i)^n$$
$$\mu_i^t = \left(\lambda_i^t\right)^{n+1}$$
$$\mu^a = (\lambda^a)^{n+1}$$

The Flow Solver

The flow variables U_m are derived from the solution of the Reynolds-Averaged Navier-Stokes (*RANS*) equations for steady flows, R_n=$0, n = 1, \ldots, M$. These are numerically solved on 2D unstructured grids with triangular-quadrilateral elements, using the finite volume technique and a vertex-centered, finite volume upwind scheme with Roe's approximate Riemann solver for the convection fluxes, [11]. The state equations are coupled with the one-equation Spalart-Allmaras turbulence model, [13].

Computation of First–Order Sensitivities

The gradient of F with respect to b_i can be computed either as

$$\frac{dF}{db_i} = \frac{\partial F}{\partial b_i} + \frac{\partial F}{\partial U_k}\frac{dU_k}{db_i} \tag{1}$$

where $\frac{dU_k}{db_i}$ are the outcome of the numerical solution of the differentiated flow equations

$$\frac{dR_n}{db_i} = \frac{\partial R_n}{\partial b_i} + \frac{\partial R_n}{\partial U_k}\frac{dU_k}{db_i} = 0 \tag{2}$$

or, equivalently, as

$$\frac{dF}{db_i} = \frac{\partial F}{\partial b_i} + \Psi_n \frac{\partial R_n}{\partial b_i} \tag{3}$$

where Ψ_n result from solving the adjoint to the flow equations

$$\frac{\partial F}{\partial U_k} + \Psi_n \frac{\partial R_n}{\partial U_k} = 0 \tag{4}$$

Following the adjoint approach (eqs. 3 and 4), the computation of $\frac{dU_k}{db_i}$ can be overcome, reducing the overall CPU cost for the gradient computation to only one additional equivalent flow solution (over and above that required to solve the flow equations). However, the computation of $\frac{dU_k}{db_i}$ cannot be avoided if the second-order sensitivities are to be computed by means of the direct-adjoint approach (hereafter, direct differentiation is merely abbreviated to direct).

Computation of Second–Order Sensitivities

Similar to the gradient computation, the Hessian matrix can be computed by re-differentiating the functional with respect to b_i to get

$$\frac{d^2F}{db_idb_j} = \frac{\partial^2 F}{\partial b_i \partial b_j} + \frac{\partial^2 F}{\partial b_i \partial U_k}\frac{dU_k}{db_j} + \frac{\partial^2 F}{\partial U_k \partial b_j}\frac{dU_k}{db_i} + \frac{\partial^2 F}{\partial U_k \partial U_m}\frac{dU_k}{db_i}\frac{dU_m}{db_j} + \frac{\partial F}{\partial U_k}\frac{d^2U_k}{db_idb_j} \quad (5)$$

The second-order sensitivities of the flow variables, $\frac{d^2U_k}{db_idb_j}$ are obtained by differentiating eq. 2, leading to

$$\frac{d^2R_n}{db_idb_j} = \frac{\partial^2 R_n}{\partial b_i \partial b_j} + \frac{\partial^2 R_n}{\partial b_i \partial U_k}\frac{dU_k}{db_j} + \frac{\partial^2 R_n}{\partial U_k \partial b_j}\frac{dU_k}{db_i} + \frac{\partial^2 R_n}{\partial U_k \partial U_m}\frac{dU_k}{db_i}\frac{dU_m}{db_j} + \frac{\partial R_n}{\partial U_k}\frac{d^2U_k}{db_idb_j} = 0 \quad (6)$$

Multiplying $\frac{d^2R_n}{db_idb_j}$ with the adjoint variables and adding it to eq. 5 gives

$$\begin{aligned}\frac{d^2F}{db_idb_j} =& \underbrace{\frac{\partial^2 F}{\partial b_i \partial b_j}}_{h1} + \underbrace{\frac{\partial^2 F}{\partial b_i \partial U_k}\frac{dU_k}{db_j} + \frac{\partial^2 F}{\partial U_k \partial b_j}\frac{dU_k}{db_i}}_{h2} + \\ &+ \underbrace{\frac{\partial^2 F}{\partial U_k \partial U_m}\frac{dU_k}{db_i}\frac{dU_m}{db_j}}_{h3} + \underbrace{\Psi_n\frac{\partial^2 R_n}{\partial b_i \partial b_j}}_{h4} + \\ &+ \underbrace{\Psi_n\frac{\partial^2 R_n}{\partial b_i \partial U_k}\frac{dU_k}{db_j} + \Psi_n\frac{\partial^2 R_n}{\partial U_k \partial b_j}\frac{dU_k}{db_i}}_{h5} + \\ &+ \underbrace{\Psi_n\frac{\partial^2 R_n}{\partial U_k \partial U_m}\frac{dU_k}{db_i}\frac{dU_m}{db_j}}_{h6} + \\ &+ \underbrace{\left(\frac{\partial F}{\partial U_k} + \Psi_n\frac{\partial R_n}{\partial U_k}\right)\frac{d^2U_k}{db_idb_j}}_{h7} \end{aligned} \quad (7)$$

By satisfying the same adjoint equations that were used for the first-order sensitivities, eq. 4, term "$h7$" can be eliminated. Furthermore, since F is an integral over the inlet and outlet boundaries and b_i control the airfoil shape, terms "$h1$" and "$h2$" also vanish. So, the Hessian becomes

$$\begin{aligned}\frac{d^2F}{db_idb_j} =& \underbrace{\frac{\partial^2 F}{\partial U_k \partial U_m}\frac{dU_k}{db_i}\frac{dU_m}{db_j}}_{h3} + \underbrace{\Psi_n\frac{\partial^2 R_n}{\partial b_i \partial b_j}}_{h4} + \\ &+ \underbrace{\Psi_n\frac{\partial^2 R_n}{\partial b_i \partial U_k}\frac{dU_k}{db_j} + \Psi_n\frac{\partial^2 R_n}{\partial U_k \partial b_j}\frac{dU_k}{db_i}}_{h5} + \\ &+ \underbrace{\Psi_n\frac{\partial^2 R_n}{\partial U_k \partial U_m}\frac{dU_k}{db_i}\frac{dU_m}{db_j}}_{h6} \end{aligned} \quad (8)$$

To compute the gradient $\frac{dc^a}{db_i}$ and Hessian $\frac{d^2c^a}{db_idb_j}$ of the flow turning constraint, a similar development to that already presented for F is to be made. Hence

$$\frac{dc^a}{db_i} = \frac{\partial c^a}{\partial b_i} + \Psi_n^a\frac{\partial R_n}{\partial b_i} \quad (9)$$

and

$$\frac{d^2c^a}{db_idb_j} = \frac{\partial^2 c^a}{\partial U_k \partial U_m}\frac{dU_k}{db_i}\frac{dU_m}{db_j} + \Psi_n^a \frac{\partial^2 R_n}{\partial b_i \partial b_j} + $$
$$+ \Psi_n^a \frac{\partial^2 R_n}{\partial b_i \partial U_k}\frac{dU_k}{db_j} + \Psi_n^a \frac{\partial^2 R_n}{\partial U_k \partial b_j}\frac{dU_k}{db_i}$$
$$+ \Psi_n^a \frac{\partial^2 R_n}{\partial U_k \partial U_m}\frac{dU_k}{db_i}\frac{dU_m}{db_j} \quad (10)$$

where the new adjoint variables Ψ_n^a result from

$$\frac{\partial c^a}{\partial U_k} + \Psi_n^a \frac{\partial R_n}{\partial U_k} = 0 \quad (11)$$

and $\frac{dU_k}{db_i}$ are computed by eq. 2. The gradient and Hessian of the thickness constraint function are computed with a simple, straightforward closed-form derivation.

Case Study – The "exact" Newton method

The proposed method based on eqs. 8 and 10 was applied to the total pressure losses minimization of a 2D compressor cascade with $\gamma = 30^o$, s=0.65 and $AVDR = 0.93$. All terms comprising the Hessian matrix expression (eq. 8) were programmed and only the terms depending on the sensitivities of the turbulence model were omitted. The outlet isentropic Mach number was set to $M_{out,is}$=0.45, the inlet angle to a_{in}=47^o and the desirable flow turning was $a_{in} - a_{out}$=22^o. Each airfoil side was parameterized using eight Bézier control points. The ordinates of three of them for each side were chosen as design variables, yielding six design variables in total. The airfoil thickness distribution was constrained not to decrease below 95% of its initial value. Examining the Hessian for the functional and constraint (fig. 1) in all optimization cycles (results from the first cycle are displayed here), we conclude that the term marked with "h6" is negligible. So, this term will be dropped hereafter, for the sake of CPU cost reduction. In view of the above, the final expression of the Hessian of F, which will be used during the optimization process, becomes

$$\frac{d^2F}{db_idb_j} = \Psi_n \frac{\partial^2 R_n}{\partial b_i \partial b_j} + \frac{\partial^2 F}{\partial U_k \partial U_e}\frac{dU_k}{db_i}\frac{dU_m}{db_j} + $$
$$+ \Psi_n \frac{\partial^2 R_n}{\partial b_i \partial U_k}\frac{dU_k}{db_j} + \Psi_n \frac{\partial^2 R_n}{\partial U_k \partial b_j}\frac{dU_k}{db_i} \quad (12)$$

Eq. 12 gives the so-called "exact" Hessian, to make a distinction from eq. 8 which gives the exact Hessian.

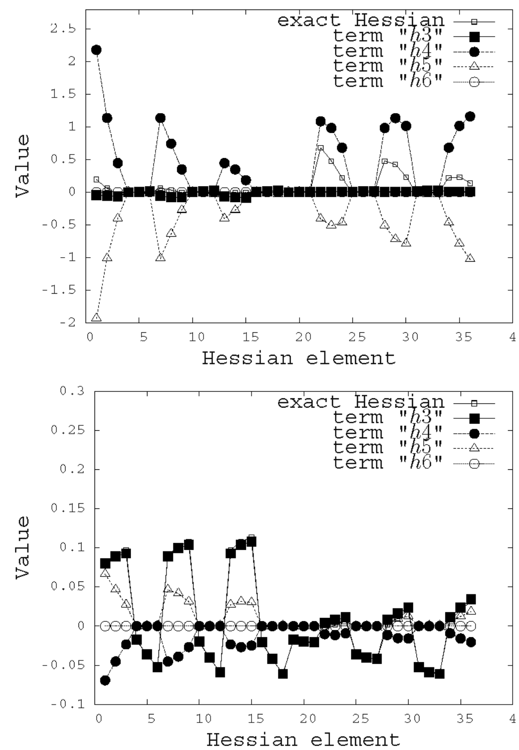

Figure 1: Comparison of terms appearing in the expression for the Hessian matrix in eq. 8 (for the functional, top) and eq. 10 (for the flow turning constraint function, bottom).

"Exact" Newton vs. Quasi Newton Optimization

Plots of the decrease in total pressure losses during the optimization are shown in fig. 2 with respect to the optimization cycles and the total CPU cost. The optimal geometry, which conforms to the set of constraints, is demonstrated in fig. 3. The flow separation region over the optimized airfoil is reduced, as can be seen in fig. 4. Further reduction without violation of either the angle or thickness constraints was not possible. A close-up view of the Mach field near the airfoil can be seen in fig. 5. The Newton algorithm converges slightly faster than the "exactly initialised" quasi Newton algorithm and conforms slightly better to the angle constraint, fig. 6, due to the more accurate computation of the Hessian at all cycles, even though the cost of the Newton method per optimization cycle is notably higher. However, for a higher number of design variables, the quasi Newton method, is expected to perform much better. "Pure" quasi Newton runs (i.e. with the unit matrix used to initialize the Hessian) were made for the case examined above. Their performance depended heavily on the initial steepest-descent step and was, in any case, worse than both the Newton and the quasi Newton methods presented herein.

Conclusions–Discussion

A method for the design of 2D compressor cascades for minimum total pressure losses subject to a number of geometric and flow constraints was presented. The method presented was based on the combined use of direct

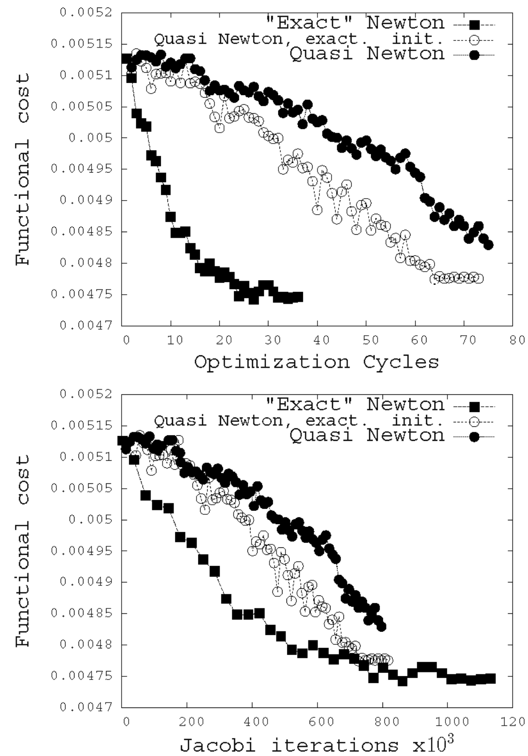

Figure 2: Total pressure losses reduction using the "exact" and quasi Newton algorithms, plotted in terms of optimization cycles (top) and total CPU cost, measured by the number of Jacobi iterations (bottom).

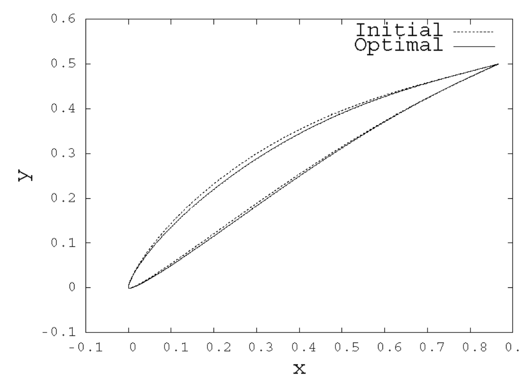

Figure 3: Initial and optimized geometry. The x and y axes are not in scale.

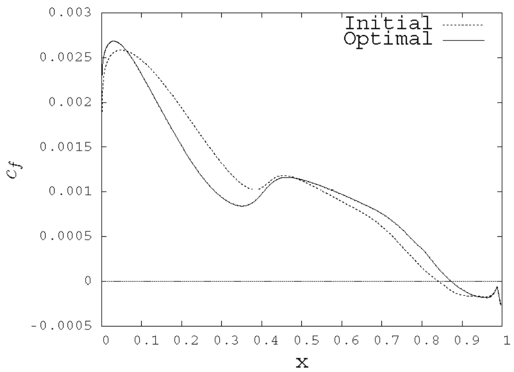

Figure 4: Friction coefficient distribution over the initial and optimal airfoil.

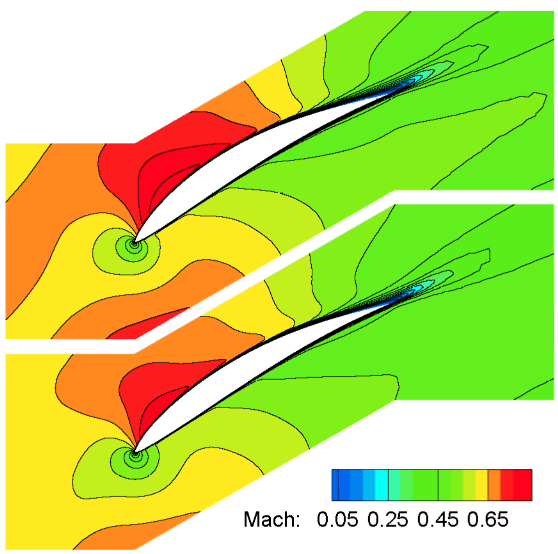

Figure 5: Close-up view of the Mach number field in the region close to the initial (up) and optimal (down) airfoil.

differentiation and discrete adjoint method to compute first- and second-order sensitivities of the objective and constraint functions. To overcome the high CPU cost caused by the

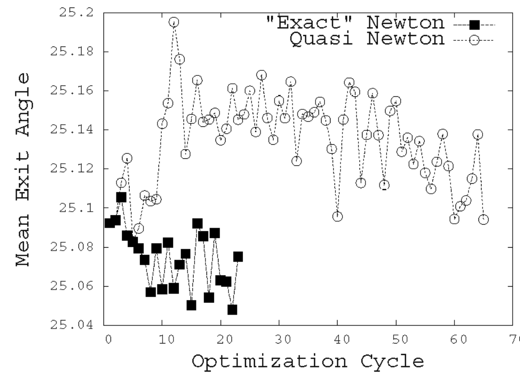

Figure 6: Convergence plots of the flow turning constraint using the “exact” Newton and quasi Newton algorithms.

N+2 equivalent flow solutions per optimization cycle, which were required to compute exact gradients and Hessians, a quasi Newton method was devised, in which the computation of the exact Hessian is made once, in the beginning of the algorithm; in all subsequent cycles, the Hessian is updated using a quasi-Newton method and the exactly computed gradient (which is obtained through the discrete adjoint method). A contribution of this paper is a proposal for a simpler Hessian formula (referred to as the “exact” Hessian within the paper) which resulted from the elimination of non-important terms as indicated by numerical experiments. The “exact” Hessian matrix is also useful for the computation of the robustness of aerodynamic shapes, when robust design methods are in use.

References

[1] E. Arian. Analysis of the Hessian for aeroelastic optimization. Technical Report NASA CR-198245,

Institute for Computer Applications in Science and Engineering, 1995.

[2] E. Arian and S. Ta'asan. Analysis of the Hessian for aerodynamic optimization: Inviscid flow. *Computers & Fluids*, 28(7):853-877, 1999.

[3] E. Arian and V. N. Vatsa. A preconditioning method for shape optimization governed by the Euler equations. Technical Report TR-98-14, Institute for Computer Applications in Science and Engineering, 1998.

[4] M. Martinelli and R. Duvigneau. Comparison of second-order derivatives and metamodel-based Monte-Carlo approaches to estimate statistics for robust design of a transonic wing. AIAA Paper 2008-2071, 2008.

[5] J. Nocedal and S. J. Wright. *Numerical Optimization*. Springer Series in Operations Research. Springer, New York, 1999.

[6] A. O. Oloso and A. C. Taylor III. Aerodynamic shape-sensitivity analysis and design optimization on the IBM-SP2. AIAA Paper 1997-2273, 1997.

[7] D. I. Papadimitriou and K. C. Giannakoglou. Computation of the Hessian matrix in aerodynamic inverse design using continuous adjoint formulations. *Computers & Fluids*, 37(8):1029-1039, 2008.

[8] D. I. Papadimitriou and K. C. Giannakoglou. The continuous direct-adjoint approach for second order sensitivities in viscous aerodynamic inverse design problems. *Computers & Fluids*, doi:10.1016/j.compfluid.2008.12.007, 2008.

[9] D. I. Papadimitriou and K. C. Giannakoglou. Direct, adjoint and mixed approaches for the computation of Hessian in airfoil design problems. *International Journal for Numerical Methods in Fluids*, 56:1929-1943, Apr. 2008.

[10] D. I. Papadimitriou and K. C. Giannakoglou. Aerodynamic shape optimization using adjoint and direct approaches. *Archives of Computational Methods in Engineering*, 15(4):447-488, 2008.

[11] P. L. Roe. Approximate Riemann solvers, parameter vectors, and difference schemes. *Journal of Computational Physics*, 135(2):250-258, Aug. 1997.

[12] L. Sherman., A. C. Taylor III, L. L. Green, P. A. Newman, G. W. Hou and V. M. Korivi. First- and second-order aerodynamic sensitivity derivatives via automatic differentiation with incremental iterative methods. *Journal of Computational Physics*, 129:307-331, Dec. 1996.

[13] P. R. Spalart and S. R. Allmaras. A one-equation turbulence model for aerodynamic flows. AIAA Paper 92-0439, 1992.

[14] A. C. Taylor III, L. L. Green, P. A. Newman, and M. M. Putko. Some advanced concepts in discrete aerodynamic sensitivity analysis. AIAA Paper 2001-2529, 2001.