11$^{\text{th}}$ ANKARA INTERNATIONAL AEROSPACE CONFERENCE **AIAC-2021-098**
8-10 September 2021 - METU, Ankara TURKEY

DEVELOPMENT OF A SUBSONIC DISCRETE-ADJOINT SOLVER USING FDOT

Berk Sarıkaya*
Middle East Technical University
Ankara, Turkey

İsmail H. Tuncer†
Middle East Technical University
Ankara, Turkey

ABSTRACT

The automatic differentiation tool, FDOT, is employed to perform sensitivity analyses using a panel code. Aerodynamic shape optimization of NACA0012 airfoil profile is done by the computed sensitivity derivatives. The baseline airfoil is optimized for the target lift and moment coefficients. The airfoil profiles are parametrized by the Class Shape Transformation. Optimization is performed by an in-house steepest descent algorithm. Validation and case studies are presented.

INTRODUCTION

Computational fluid dynamics (CFD) generally allows to evaluate larger design spaces in much shorter time span than wind tunnel tests, or flight tests. However, this case may not be necessarily true. For example, gradient-based optimization process requires the gradient information which is either obtained by adjoint solvers or by finite difference (FD) approaches where the latter means running the CFD solver over and over again by perturbing each design variable one by one. Even though clever parametrization/deformation techniques such as Free-Form Deformation (FFD) reduce the design variables significantly, FD gradients are still not feasible for large scale optimization cases. Gradient-free methods, in a similar fashion, may require a lot of flowfield computations, due to the optimization algorithm. The problem has been overcome with the introduction of the adjoint methods which are pioneered by Jameson [Jameson, 1989]. Adjoint method allows one to calculate the gradient information with cost similar to one flowfield calculation, irrespective of the number of the design variables present. However, adjoint methods require the partial derivative information of various variables present in the CFD code. To compute the required partial derivatives, automatic/algorithmic differentiation (AD) tools are widely used, since these tools are less prone to user induced error in the case of hand-differentiation. Therefore, to automatically differentiate (AD) the code, source-code transformation or operator-overloading tools can be used. For example, a source-code transformation tool, has been used in [Kaya *et al.*, 2019]. This paper explores the usage of operator-overloading where some hassles of the source-code transformations are avoided and everything is handled intrinsically where discrete-adjoints of a flowfield can be obtained rapidly.

*MSc. Student, Aerospace Engineering Department, Email: berk.sarikaya@metu.edu.tr
†Professor, Aerospace Engineering Department, Email: ismail.h.tuncer@ae.metu.edu.tr

METHOD

Aerodynamic shape optimization procedure generally consists of shape parametrization, flowfield calculation, gradient calculation, projection of gradients and mesh deformation in accordance with gradient information. In Figure 1, the chain of optimization is shown. Each subdivision of the chain can be automatically differentiated and can be incorporated below a wrapper code. In fact, the entire design chain can be automatically differentiated. Such an example is given in [Gauger *et al.*, 2012]. First, airfoil shapes are parametrized with Class Shape Transformations. Then, sensitivity derivatives are computed. An adjoint solver is required to compute the partial derivatives in the sensitivity analysis. Using FDOT, which stands for ***F****ast Automatic* ***D****ifferentiation Based on* ***O****perator-Overloading* ***T****echnique*, a discrete adjoint solver is created. [Djeddi and Ekici, 2019]

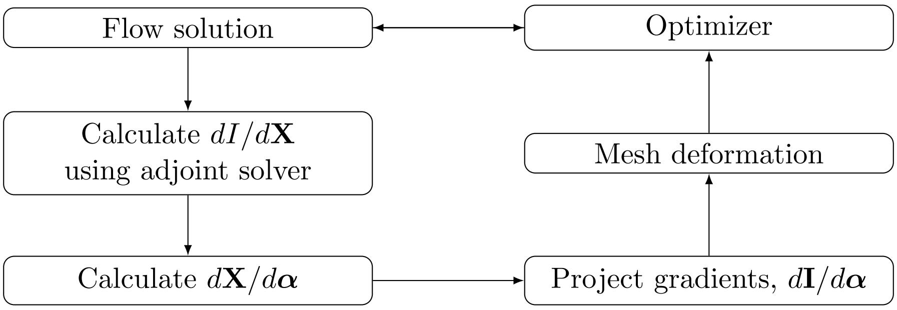
Flow solution
Optimizer
Calculate $dI/d\mathbf{X}$ using adjoint solver
Mesh deformation
Calculate $d\mathbf{X}/d\boldsymbol{\alpha}$
Project gradients, $d\mathbf{I}/d\boldsymbol{\alpha}$

Figure 1: Example chain of optimization.

Flow Solver

The flow solver is a panel code, which solves a system of M equations with M unknowns, obtained by M-1 number of panels and 1 equation from Kutta condition. After the system of influence coefficients are solved, then local tangential velocities can be computed to obtain pressure distribution over the airfoil.

Adjoint Method

The total derivative of an objective function, I, is given by [Mavriplis, 2007],

$$\frac{dI}{d\boldsymbol{\alpha}} = \left(\frac{\partial I}{\partial \mathbf{X}} + \frac{\partial I}{\partial \mathbf{q}}\frac{\partial \mathbf{q}}{\partial \mathbf{x}}\right)\frac{\partial \mathbf{x}}{\partial \boldsymbol{\alpha}} = \frac{dI}{d\mathbf{X}}\frac{d\mathbf{X}}{d\boldsymbol{\alpha}} \tag{1}$$

In equation 1, the vector of design variables is denoted with $\boldsymbol{\alpha}$, and the mesh/panel nodes are denoted with $\mathbf{X}$. Also, terms with bold font denote vectors. Necessary partial derivatives in this equation can be computed via the adjoint method. Any **converged** solution can be written in the following form,

$$\frac{d\mathbf{R}}{d\boldsymbol{\alpha}} = \frac{\partial \mathbf{R}}{\partial \boldsymbol{\alpha}} + \frac{\partial \mathbf{R}}{\partial \mathbf{q}}\frac{\partial \mathbf{q}}{\partial \boldsymbol{\alpha}} = 0 \tag{2}$$

where $\mathbf{R}$ is the residual vector and total derivative of the residual vector with respect to design variables should vanish as convergence is reached. [Djeddi and Ekici, 2021] Solving for $\partial\mathbf{q}/\partial\boldsymbol{\alpha}$ in equation 2, and inserting it into equation 1 yields,

$$\frac{d\mathbf{I}}{d\boldsymbol{\alpha}} = \frac{\partial \mathbf{I}}{\partial \boldsymbol{\alpha}} + \psi^T\frac{\partial \mathbf{R}}{\partial \boldsymbol{\alpha}} \tag{3}$$

where ψ^T given in equation 3 is the adjoint vector,

$$\psi^T = -\frac{\partial I}{\partial q}\left[\frac{\partial R}{\partial q}\right]^{-1} \tag{4}$$

Thus, FDOT uses the saved expression tree to solve for the adjoint vector at a computational cost proportional to one flow solution [Djeddi and Ekici, 2019]. $dI/d\mathbf{X}$ values which are the sensitivities of the objective function to the panel nodes are the output of this routine.

Discrete Adjoint Solver

The flow solver is transformed into the adjoint solver by the FDOT, as mentioned in the preceding part. FDOT is a toolbox which allows automatic differentiation of Fortran codes. With minor modifications to the source code, an adjoint solver can be created rapidly. Also, the gradient information is accurate to machine precision. The code requires a fully-converged flowfield solution, which is an output of the flow solver. The solver has the same properties as the flow solver, such as spatial accuracy. The fully-converged CFD solution is an input of the adjoint solver. When the adjoint code runs, an expression tree is formed. To calculate adjoints, the expression tree/tape is rewound (reverse mode) and fixed-point Newton iterations are carried out until convergence is reached.

Parametrization with CST

Class Shape Transformations are used to parametrize the airfoils [Kulfan, 2008]. CST allow for smooth behaving airfoils whereas using airfoil y-coordinates directly in the optimization may yield wavy airfoils. Additionally, CST makes it possible to obtain realistic airfoils using any number of Bernstein polynomials. A code is developed that which minimizes the difference between the baseline airfoil and the CST representation of the profile. This way, it is possible to obtain the weights of the baseline airfoil with desired number of Bernstein polynomials. In Figure 2, 6th order CST representation of NACA0012 airfoil is shown. The RMS is on the order of 10^{-5}. In the optimization process, d$\mathbf{X}$/d$\boldsymbol{\alpha}$ terms are calculated using finite-differencing CST weights. [Mavriplis, 2007].

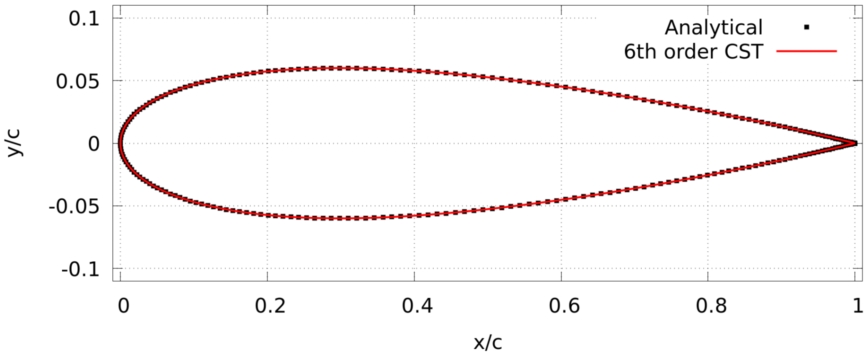

Figure 2: 6th order CST representation of NACA0012 airfoil

RESULTS AND DISCUSSION

The goal is to design an airfoil with target lift and pitching moment coefficient and the minimal adverse pressure gradient, using the low computational cost of the panel solver. Three cases are presented, with increasing complexity in objective functions. Constraints are added case-by-case. Initially, NACA0012 airfoil is chosen to be the baseline airfoil at zero degrees of angle of attack. 6th order CST representation of the airfoil is used. Target lift and moment coefficients are prescribed as 0.3 and 0.005, respectively. In the first case, the target lift coefficient is satisfied. Then, the pitching moment coefficient is added as a secondary constraint. Third, the adverse pressure gradient term is added as an additional constraint to move the location of minimum pressure location as far back as possible. The results of the optimization are presented in the following parts.

Validation of the Sensitivity Derivatives

Finite Differences can be used to validate sensitivity derivatives. Any objective function I and design variable ξ can be used to obtain a first-order accurate gradient with a step-size of h,

$$\frac{\partial I}{\partial \xi} = \frac{I(\xi + h) - I(\xi)}{h} + O(h) \tag{5}$$

Equation 5 is still prone to truncation and cancellation errors unless complex stepping is introduced, while FDOT predictions are accurate to machine precision [Djeddi and Ekici, 2019]. However, with small enough stepsize even a first-order accurate gradient information is enough to validate the gradients obtained by adjoint method. In Figure 3, gradients calculated using adjoint method and finite differences are shown. Clearly the agreement is excellent between adjoint and finite differences. Additionally, using finite differences, 14 flow solutions are required for gradient information, which is costly. However, with introduction of the adjoint method outlined in the preceding chapter, the cost associated with gradient computation is in the order of one flow solution.

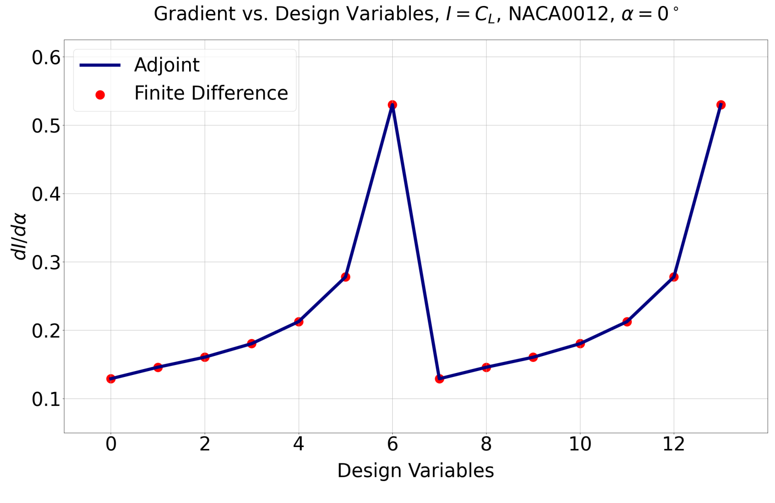

Figure 3: Gradient vs. design variables for lift coefficient as the objective function

Table 1: Aerodynamic Coefficients

	C_L	C_M
Baseline	0.0	0.0
Case I	0.301	-0.069
Case II	0.297	-0.004

Case I: Target lift coefficient

For the first case, the objective function to be minimized is,

$$I = |C_{L,target} - C_L| \tag{6}$$

Using Eqn. 6, unconstrained optimization problem outlined in the preceding parts is now a constrained optimization problem. The target lift coefficient, $C_{L,target}$ is set to be 0.3. Figure 4 shows the results of the optimization under a target lift constarint. The optimized airfoil profile has thickness distribution similar to that of the baseline. The difference to note is the addition of camber. Target lift coefficient is realized by addition of camber, as shown in Table 1. As a final remark, a cusp near trailing edge has formed, which is undesired in this case due to high pitching moment associated with it.

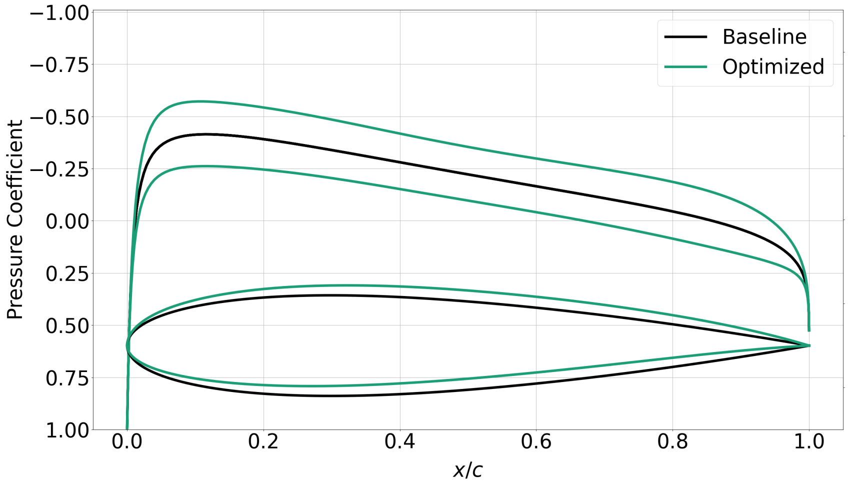

Figure 4: Airfoil profiles and pressure distributions for case I

Case II: Target lift and pitching moment coefficients

For the second case, the objective function to be minimized is,

$$I = |C_{L,target} - C_L| + \beta \, |C_{M,target} - C_M| \tag{7}$$

where β is a weighting term that is used to give equal importance for lift and moment terms. In Eqn. 7, target C_L is set to be 0.3 and the target C_M is set to be 0.005. The weight coefficient, β, is set to 5, to have a similar order of magnitude in the pitching moment term. In Fig. 5 optimized and the baseline profiles are shown. The pressure coefficient changes a lot when compared with the previous case. Much of the camber is now focused near $0.25c$ to decrease the C_M. The airfoil resembles a reflexed airfoil, and the cusp at the trailing edge is eliminated. Both target coefficients are realized as shown in Table 1. By having near zero moment, it is possible to incorporate the optimized airfoil on tailless aircraft configurations. Additionally, on a conventional aircraft trim would require less horizontal stab movement.

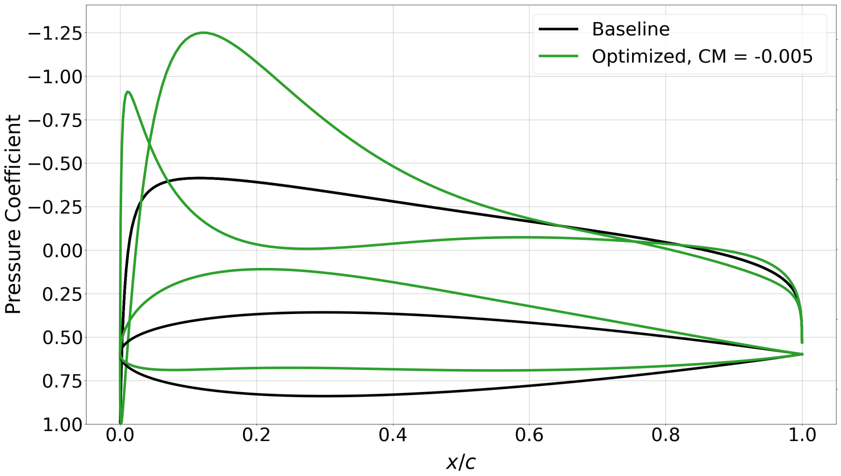

Figure 5: Airfoil profiles and pressure distributions for case II

Finally, all of the optimized airfoil profiles and the baseline airfoil is plotted in Figure 6. A simple camber addition is observed in for the case with no pitching moment criterion, with cusp formation near trailing edge. Near-zero moment airfoil obtained in case II has reflexed trailing edge, and most of the camber is focused near $0.25c$. Also, the cusp has been eliminated at the trailing edge, which is a major source of pitching moment.

In the Figure 7, objective function change with minor iterations are shown. Case I converges easily with only 1 gradient call and case II converges with 20 gradient evalutions. It should be kept in mind that due to low pitching moment target this is a highly challenging case.

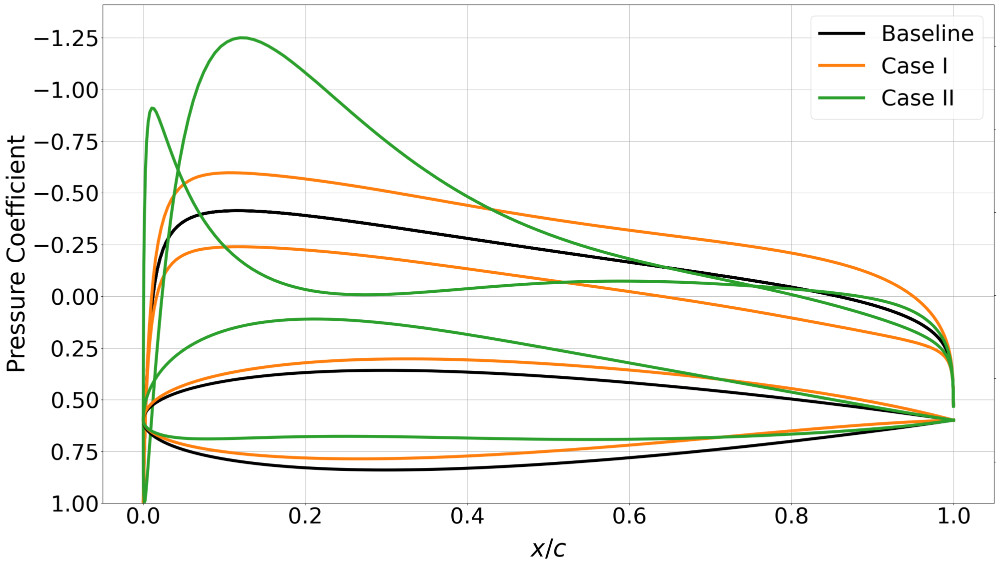

Figure 6: Airfoil profiles and pressure distributions for all the cases

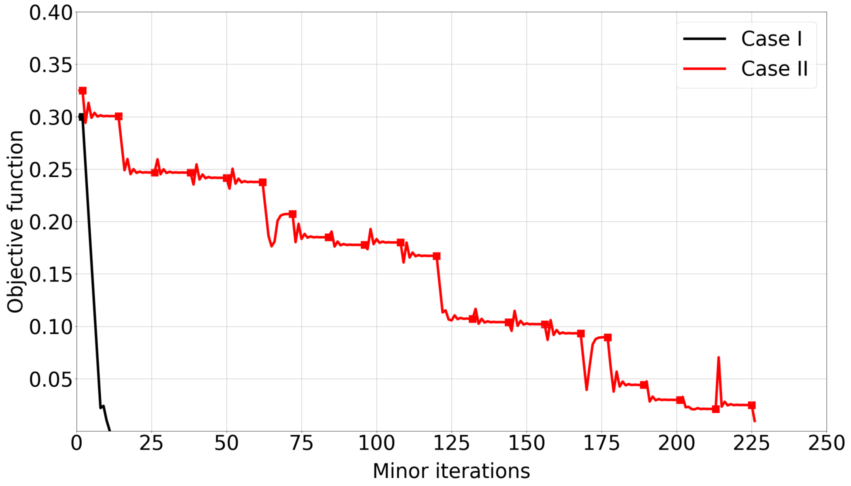

Figure 7: Optimization histories

CONCLUSIONS

Automatic differentiation tool, FDOT, is used to automatically differentiate a panel code written in Fortran. The panel code uses Class Shape Transformations to parametrize the airfoil shapes. The computed sensitivity derivates are compared with finite differences and the agreement is excellent. Then, a case study is presented with two parts using an in-house steepest descent optimizer. In the first part, target lift coefficient is prescribed and the target is realized. Second part builds upon on the complexity in the objective function, in the form of pitching moment. Both of the targets are realized in the second case, aswell. The shape change is significant and much of the camber is focused near quarter-chord. The optimized airfoil has near zero pitching moment. By adding a pitching moment criteron alongside the lift criterion, it is shown that designing airfoils with arbitrary lift and moment coefficients is possible.

References

Djeddi, R. and Ekici, K. (2019) *FDOT: A Fast, memory-efficient and automated approach for Discrete Adjoint Sensitivity Analysis using the Operator Overloading Technique*, Elsevier, Aerospace Science and Technology, Vol 91, pp: 159-174, Aug 2019

Djeddi, R. and Ekici, K. (2021) *Helicopter Rotor Optimization via Operator Overloading-Based Discrete Adjoint Approach*, American Institute of Aeronautics and Astronautics SciTech Conference 2021 - AIAA-2021-0170, Jan 2021

Gauger, N. R., Walther, A., Moldenhauer, C. and Widhalm, M. (2007) *Automatic Differentiation of an Entire Design Chain for Aerodynamic Shape Optimization*, Springer-Verlag, New Results in Numerical and Experimental Fluid Mechanics VI, pp: 454-461, 2007

Hicks, R. M. and Vanderplaats, G. N. (1975) *Design of Low-Speed Airfoils by Numerical Optimization*, SAE International, SAE Technical Paper Series, February 1975

Jameson, A. (1989) *Computational Aerodynamics for Aircraft Design*, Science Vol. 245, Issue 4916, pp. 361-371, Jul 1989

Kaya, H., Tuncer, I. H. and Tiftikci, H. (2019) *Development of a 2D Discrete Turbulent Adjoint Solver Using Automatic Differentiation*, Ankara International Aerospace Conference 2019, AIAC-2019-205, 2019

Kulfan, B. M. (2008) *Universal Parametric Geometry Representation Method*, Journal of Aircraft, Vol 45, No:1, Jan 2008

Mavriplis, D. J. (2007) *Discrete Adjoint-based Approach for Optimization Problems on Three-Dimensional Unstructured Meshes*, AIAA Journal, Vol 45, No:4, Apr 2007