Continuous adjoint based optimization using a pseudo-compressibility implicit solver in OpenFOAM©

Christos K. Vezyris*, Evangelos M. Papoutsis-Kiachagias, Ioannis S. Kavvadias and Kyriakos C. Giannakoglou
Parallel CFD & Optimization Unit, School of Mechanical Engineering
National Technical University of Athens, Greece
Email: cvezyris@gmail.com, vaggelisp@gmail.com, kavvadiasj@hotmail.com, kgianna@central.ntua.gr

Summary

This paper presents the development of the continuous adjoint method, including the differentiated Spalart–Allmaras turbulence model, for shape optimization. For the solution of the flow and adjoint equations, a pseudo-compressibility based implicit solver was programmed in the OpenFOAM© environment. In foam-3.1-ext, the coupling of the incompressible flow equations relies upon the Rhie-Chow interpolation. Here, the pseudo-compressibility approach is applied, by transforming the elliptic system of incompressible equations into a hyperbolic one. This allows the usage of solvers suited for this type of PDEs, which are potentially faster. The shape optimization process is applied in two cases, a U-bent and the Ahmed body, aiming at minimum total pressure losses and drag, respectively.
Keywords: *Shape optimization, continuous adjoint method, pseudo-compressibility, adjoint turbulence model*

1 Introduction

Deterministic optimization algorithms require the computation of the gradient of the selected objective function w.r.t a set of design variables. The most efficient way to compute this gradient is the adjoint method[1], in either its discrete or continuous variant.

In this paper, the continuous adjoint method[2], where the adjoint PDEs are firstly derived and then discretized, is used. The state equations are the mean-flow Navier-Stokes equations for incompressible fluids and the Spalart–Allmaras turbulence model[3] PDE is used to effect closure. In the derivation of the adjoint PDEs and boundary conditions, the turbulence model PDE is differentiated[4], thus, overcoming the common assumption of frozen turbulence.

When incompressible flows are numerically predicted, the main difficulty is found in the physical decoupling of the continuity and momentum equations since there is no pressure term in the continuity equation. In segregated algorithms (e.g. SIMPLE), when the numerical solution for the velocity precedes the one for the pressure, a Poisson equation is derived and solved for the pressure[5], in order to impose the divergence-free velocity constraint. The main reason that segregated solution algorithms, like SIMPLE, have been adopted by the CFD community (and implemented in the CFD package OpenFOAM©) is the low memory requirement.

On the other hand, implicit solvers treat a single matrix containing linear couplings for all flow variables. Implicit solvers could increase convergence rates, induce potentially greater stability and, in some cases, might really be needed to ensure convergence.

By manipulating the continuity equation, pseudo-compressibility aims at formulating a hyperbolic system of equations. The system transformation is achieved by adding a pseudo-time derivative for the pressure, multiplied with a coefficient $^1/_\beta$, to the continuity equation. The choice of the β coefficient is crucial in the stability and convergence of the system.

In this paper, the formulation of the flow solver, based on the pseudo-compressibility approach, is presented at first. Then, the objective functions used in internal (i.e. total pressure losses) and external (i.e. drag) aerodynamics are given, followed by the development of the continuous adjoint formulation and the differentiation of the Spalart–Allmaras turbulence model. Programming issues in OpenFOAM© are discussed. Finally, results obtained during the optimization of the U-bend and Ahmed body are presented.

2 The flow solver

In the pseudo-compressibility approach[6], the continuity equation is augmented by a pseudo-time marching term for the pressure, multiplied by a coefficient ($^1/_\beta$). By doing so, the system of equations for the incompressible flow is transformed from elliptical to hyperbolic. The resultant system is well posed and existing algorithms for the numerical solution of compressible flows can be used to march the system in time and reach a steady state solution[7].

The pseudo-compressibility method can be seen as a preconditioning method applied to the compressible flow solution methods. After the addition of the pseudo-time derivative term(s), the mean-flow PDEs, read

$$R_p = \frac{1}{\beta}\frac{\partial p}{\partial t} + \frac{\partial \upsilon_j}{\partial x_j} = 0 \quad (1)$$

$$R_{\upsilon_i} = \frac{\partial \upsilon_i}{\partial t} + \upsilon_j \frac{\partial \upsilon_i}{\partial x_j} + \frac{\partial p}{\partial x_i} + \frac{\partial}{\partial x_j}\left[(\nu+\nu_t)\left(\frac{\partial \upsilon_i}{\partial x_j} + \frac{\partial \upsilon_j}{\partial x_i}\right)\right] = 0 \quad (2)$$

where p and $\upsilon_i, i = 1,2(,3)$ is the kinematic pressure and velocity (i.e. first state variables), respectively, β is the artificial compressibility coefficient and $x_i, i = 1,2(,3)$ the Cartesian coordinates. ν and ν_t are the bulk and turbulent viscosity, respectively. Based on the Spalart–Allmaras turbulence model[3], the viscosity coefficient is given by $\nu_t = \tilde{\nu} f_{v_1}$, where $\tilde{\nu}$ (i.e. last state variable) is the solution in the corresponding state equation, $R_{\tilde{\nu}} = 0$, where

$$R_{\tilde{\nu}} = \frac{\partial \tilde{\nu}}{\partial t} + \frac{\partial (\upsilon_i \tilde{\nu})}{\partial x_i} - \frac{\partial}{\partial x_i}\left[\left(\nu + \frac{\tilde{\nu}}{\sigma}\right)\frac{\partial \tilde{\nu}}{\partial x_i}\right] - \frac{c_{b_2}}{\sigma}\left(\frac{\partial \tilde{\nu}}{\partial x_i}\right)^2 - \tilde{\nu}P(\tilde{\nu}) + \tilde{\nu}D(\tilde{\nu}) = 0 \quad (3)$$

The production $P(\tilde{\nu})$ and destruction $D(\tilde{\nu})$ terms are given by

$$P(\tilde{\nu}) = c_{b1}\tilde{S}, \quad D(\tilde{\nu}) = c_{w1} f_w(\tilde{S}) \frac{\tilde{\nu}}{\Delta^2} \quad (4)$$

where, the definition of the remaining terms f_{v_1}, f_w, $\tilde{S}$, and constants c_{b_1}, c_{b_2}, c_{w_1} and σ can be found in[3]. Δ stands for the distance from the nearest wall boundary face.

For the discretization of the inviscid fluxes, the approximate Riemann solver proposed by Roe[8] is used. In the Roe flux scheme, a Jacobian matrix based on averaged face-sharing neighboring cells must be computed.

The pseudo-compressibility approach has been implemented in foam-3.1-ext using the existing block matrix infrastructure. Each element of the diagonal and off-diagonal left-hand-side matrices (*diag,l,u* in the OpenFOAM© terminology) is a rank-two tensor (*tensor3* (*tensor4*) for 2D (3D) simulations in OpenFOAM© terminology). This, leads to a strong numerical coupling of the mean-flow PDEs. For the discretization of the diffusion terms, the standard OpenFOAM© approach was used (*fvm::laplacian()*). Contributions of the inviscid fluxes were accounted for without using existing OpenFOAM© operators. For this purpose the developed Roe scheme was used. It should be mentioned that foam-3.1-ext uses cell-centered, finite-volume discretization schemes with the ability to support arbitrary convex polyhedral meshes. The U-bend case uses a structured mesh while the Ahmed body an unstructured one.

3 The Continuous Adjoint Method

Depending on the application, two objective functions F_i are considered. The first refers to the volume averaged total pressure losses (applied to internal aerodynamics problems) between the inlet and outlet domains while the second corresponds to the drag force (in external aerodynamics). These functions are

$$F_1 = -\int_{S_{I,O}} \left(p + \frac{1}{2}\upsilon_k^2\right)\upsilon_i n_i dS \quad (5)$$

$$F_2 = \int_{S_w} \left[pn_i - (\nu+\nu_t)\left(\frac{\partial \upsilon_i}{\partial x_j} + \frac{\partial \upsilon_j}{\partial x_i}\right)n_j\right] r_i dS \quad (6)$$

where n_i is the outwards pointing normal unit vector, r_i are the components of the unit vector aligned with the farfield velocity, S_I, S_O and S_w stand for the inlet, outlet and solid wall boundaries, respectively.

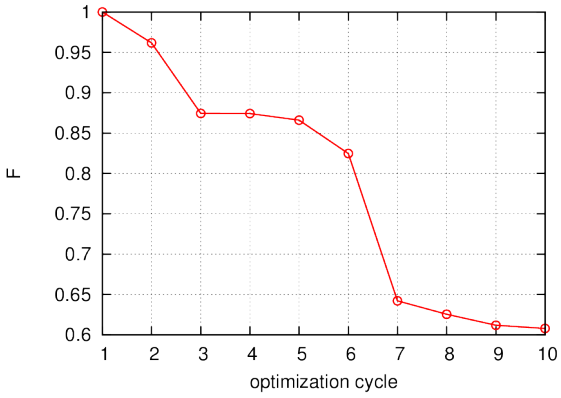

Figure 1: Evolution of the total pressure losses objective function (non-dimensionalized by its initial value) during the optimization loop for the U-bend case. The steepest descent approach was used.

In the continuous adjoint method, the augmented objective function F_{aug} is defined as the sum of the objective function F and the field (Ω) integral of the residual of the state equations $R_{\vec{U}} = 0$, $\vec{U} = (p, \upsilon_i, \tilde{\nu})$ multiplied by the adjoint fields $\vec{\Psi} = (q, u_i, \tilde{\nu}_a)$, $F_{aug} = F + \int_\Omega \vec{\Psi} R_{\vec{U}} d\Omega$. Its variation with respect to the design variable array, $b_n\,(n = 1\ldots,N)$, after applying the Leibniz theorem, becomes[4,9],

$$\frac{\delta F_{aug}}{\delta b_n} = \frac{\delta F_i}{\delta b_n} + \int_\Omega q\frac{\partial R_p}{\partial b_n}d\Omega + \int_\Omega u_i \frac{\partial R_{\upsilon_i}}{\partial b_n}d\Omega + \int_\Omega \tilde{\nu}_a \frac{\partial R_{\tilde{\nu}}}{\partial b_n}d\Omega + \int_S (qR_p + u_i R_{\upsilon_i} + \tilde{\nu}_a R_{\tilde{\nu}})\frac{\delta x_k}{\delta b_n} n_k dS \quad (7)$$

The adjoint field equations[4] and their boundary conditions are derived by eliminating field integrals depending on $\frac{\partial p}{\partial b_n}$, $\frac{\partial \upsilon_i}{\partial b_n}$, $\frac{\partial \tilde{\nu}}{\partial b_n}$. The field adjoint to the

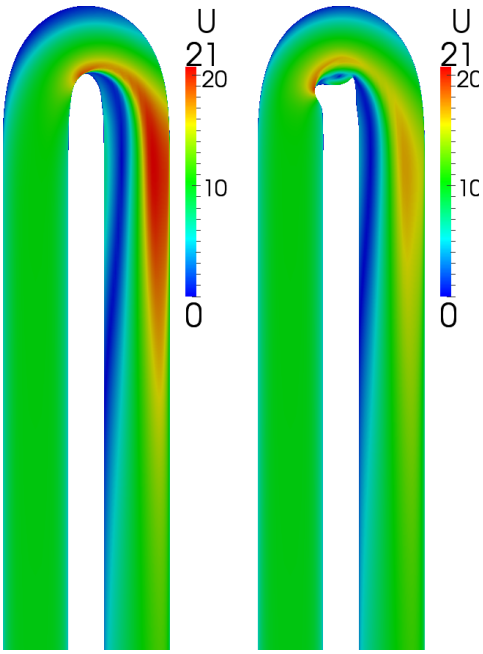

Figure 2: Velocity field for the U-bend case before and after the optimization.

mean-flow and turbulence equations are given by

$$R_q = \frac{\partial q}{\partial t} + \frac{\partial u_j}{\partial x_j} = 0 \tag{8a}$$

$$\begin{aligned} R_{u_i} = {} & \frac{\partial u_i}{\partial t} - \upsilon_j \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \\ & - \frac{\partial}{\partial x_j} \left[(\nu + \nu_t) \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \right] \\ & - \beta \frac{\partial q}{\partial x_i} - \tilde{\nu} \frac{\partial \tilde{\nu}_a}{\partial x_i} \\ & - \frac{\partial}{\partial x_l} \left(e_{jli} e_{jmq} \frac{C_S}{S} \frac{\partial \upsilon_q}{\partial x_m} \tilde{\nu} \tilde{\nu}_a \right) = 0 \end{aligned} \tag{8b}$$

$$\begin{aligned} R_{\tilde{\nu}_a} = {} & \frac{\partial \tilde{\nu}_a}{\partial t} - \upsilon_j \frac{\partial \tilde{\nu}_a}{\partial x_j} - \frac{\partial}{\partial x_j} \left[\left(\nu + \frac{\tilde{\nu}}{\sigma} \right) \frac{\partial \tilde{\nu}_a}{\partial x_j} \right] + \frac{1}{\sigma} \frac{\partial \tilde{\nu}_a}{\partial x_j} \frac{\partial \tilde{\nu}}{\partial x_j} \\ & + 2 \frac{c_{b2}}{\sigma} \frac{\partial}{\partial x_j} \left(\tilde{\nu}_a \frac{\partial \tilde{\nu}}{\partial x_j} \right) + \tilde{\nu}_a \tilde{\nu} \, C_{\tilde{\nu}}(\tilde{\nu}, \vec{\upsilon}) \\ & + \frac{\partial \nu_t}{\partial \tilde{\nu}} \frac{\partial u_i}{\partial x_j} \left(\frac{\partial \upsilon_i}{\partial x_j} + \frac{\partial \upsilon_j}{\partial x_i} \right) + (-P + D) \, \tilde{\nu}_a = 0 \end{aligned} \tag{8c}$$

where e_{ijk} is the permutation symbol. Terms C_S, S and $C_{\tilde{\nu}}$ can be found in a recent publication[4] by the same group.

After having computed the adjoint fields using the same pseudo-compressibility approach, by numerically satisfying the adjoint equations and their boundary conditions, the variation of the F_{aug} becomes independent of variations in the state variables, leading to the expressions of the sensitivity derivatives in terms of $\vec{U}$ and $\vec{\Psi}$ as well as variations in geometrical quantities. The sensitivity

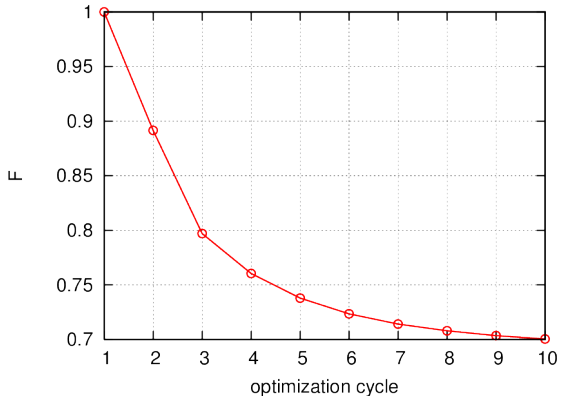

Figure 3: Evolution of the objective function (non-dimensionalized by its initial value) during the optimization loop for the Ahmed body case. In each cycle, the design variables are updated using steepest descent with a user-defined step value.

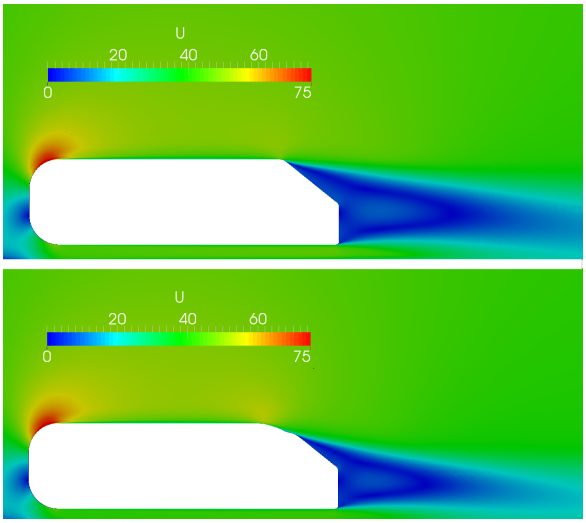

Figure 4: Velocity field for the Ahmed body. Top: Velocity field for the initial shape of the body. Bottom: Optimized geometry. The decrease in the drag force exerted on the body reflects upon the smaller recirculation region developed in the wake.

derivatives of F_{aug} are given by

$$\begin{aligned} \frac{\delta F_{aug}}{\delta b_n} = {} & A_{F_i} - \int_{S_W} (\nu + \nu_t) \left(\frac{\partial u_i}{\partial x_j} \right. \\ & \left. + \frac{\partial u_j}{\partial x_i} \right) n_j \frac{\partial \upsilon_i}{\partial x_k} \frac{\delta x_k}{\delta b_n} dS \\ & - \int_{S_W} (\nu + \nu_t) \frac{\partial \tilde{\nu}_a}{\partial x_j} n_j \frac{\partial \tilde{\nu}}{\partial x_k} \frac{\delta x_k}{\delta b_n} dS \\ & + \int_{\Omega} \tilde{\nu}_a \tilde{\nu} \, C_d(\tilde{\nu}, \vec{\upsilon}) \frac{\partial \Delta}{\partial b_n} d\Omega \end{aligned} \tag{9}$$

where, depending on the objective function used,

$$A_{F_1}=0 \tag{10}$$

$$A_{F_2}=\int_{S_W}\left[-(\nu+\nu_t)\frac{\partial}{\partial x_k}\left(\frac{\partial v_i}{\partial x_j}+\frac{\partial u_j}{\partial x_i}\right)+\frac{\partial p}{\partial x_k}\delta_j^j\right]n_j r_i\frac{\delta x_k}{\delta b_n}dS+\int_{S_W}\left[-(\nu+\nu_t)\left(\frac{\partial v_i}{\partial x_j}+\frac{\partial u_j}{\partial x_i}\right)+p\delta_i^j\right]r_i\frac{\delta}{\delta b_n}(n_j dS) \tag{11}$$

The wall boundary (S_W) and field integrals in equation (9) depend on the state and adjoint variables. The term $\frac{\partial\Delta}{\partial b_n}$ is treated as in a previous work[4], by formulating and solving the adjoint to the eikonal equation for distance computations.

4 Applications

The developed code was applied in two shape-optimization problems. Both are parameterized by using volumetric B-splines. First, a U-bend case, illustrated in figure 2, was optimized. Here, 216 design variables are used. These are the coordinates of the volumetric B-spline control points. The Reynolds number is $\sim 7.3\times10^4$, the number of cells is 1.7×10^4, the mean non-dimensional wall distance is $y^+ = 0.8$ and the magnitude of the inlet velocity is $v_{in} = 10^m/_s$. The objective function F_1 is considered. In figure 1, the convergence for the optimization process, is presented. A total drop in the objective function of $\sim 40\%$ after 10 optimization cycles is observed. The velocity field for the optimized duct is illustrated in figure 2.

The second case is the Ahmed body. Depending on the rear slant angle, various configurations for this test case exist. Here, a body with a slant angle of 30^o is to be optimized. This case is parameterized with 210 design variables. As in the previous case, the design variables are the coordinates of the volumetric B-spline control points. Considering the length of the car as characteristic length and $U_\infty = 40^m/_s$, the Reynolds number is equal to 2.9×10^6. The number of cells is $\sim 2\times10^5$ and the mean non-dimensional wall distance is $y^+ = 1.0$. The objective function used is the drag force (i.e. F_2). The convergence history of the steepest descent algorithm, used to minimize F_2, can be found in figure 3. A $\sim 30\%$ reduction in the objective function after 10 optimization cycles is achieved. Figure 4 illustrates the computed velocity magnitude isoareas, before and after the optimization.

Details on the convergence characteristics of the pseudo-compressibility method applied to the flow and adjoint solver will be presented in the full paper.

Acknowledgment

This research was funded from the People Programme (ITN Marie Curie Actions) of the European Union's 7^{th} Framework Programme (FP7/2007-2013) under REA grant agreement n^o 317006 (AboutFLOW project). The first author is an AboutFLOW Early Stage Researcher.

References

[1] Jameson, A. Aerodynamic design via control theory. *Journal of Scientific Computing* **3**, 233–260 (1988).

[2] Papadimitriou, D. and Giannakoglou, K. Aerodynamic shape optimization using first and second order adjoint and direct approaches. *Archives of Computational Methods in Engineering (State of the Art Reviews)* **15**(4), 447–488 (2008).

[3] Spalart, P. R. and Allmaras, S. R. A one–equation turbulence model for aerodynamic flows. *La Recherche Aérospatiale* **43**(1), 5 – 21 (1994).

[4] Papoutsis-Kiachagias, E. and Giannakoglou, K. Continuous adjoint methods for turbulent flows, applied to shape and topology optimization: Industrial applications. *Archives of Computational Methods in Engineering* , 1–45 (2014).

[5] Darwish, M., Sraj, I., and Moukalled, F. A coupled finite volume solver for the solution of incompressible flows on unstructured grids. *Journal of Computational Physics* **228**(1), 180 – 201 (2009).

[6] Chorin, A. J. A numerical method for solving incompressible viscous flow problems. *Journal of Computational Physics* **135**(2), 118 – 125 (1997).

[7] Rahman, M. M. and Siikonen, T. An artificial compressibility method for viscous incompressible and low Mach number flows. *International Journal for Numerical Methods in Engineering* **75**(11), 1320–1340 (2008).

[8] Roe, P. Approximate Riemann solvers, parameter vectors, and difference schemes. *Journal of Computational Physics* **43**(2), 357 – 372 (1981).

[9] Papadimitriou, D. and Giannakoglou, K. A continuous adjoint method with objective function derivatives based on boundary integrals for inviscid and viscous flows. *Computers & Fluids* **36**(2), 325–341 (2007).