

Optimal Shape Design for Open Rotor Blades

Thomas D. Economou*, Francisco Palacios†, and Juan J. Alonso‡
Stanford University, Stanford, CA 94305, U.S.A.

A continuous adjoint formulation for optimal shape design of rotating surfaces, including open rotor blades, is developed, analyzed, and applied. The compressible Euler equations are expressed in a rotating reference frame, and from these governing flow equations, an adjoint formulation centered around finding surface sensitivities using differential geometry is derived. The surface formulation provides the gradient information necessary for performing gradient-based aerodynamic shape optimization. A two-dimensional test case consisting of a rotating airfoil is used to verify the accuracy of the gradient information obtained via the adjoint method against finite differencing, and a gradient accuracy study is also performed. The shape of the airfoil is then optimized for drag minimization in the presence of transonic shocks. In three-dimensions, the formulation is verified against finite differencing for a classic, two-bladed rotor, which is then redesigned for minimum inviscid torque using a Free-Form Deformation approach to geometry parameterization. Optimal shape design for open rotor blades is presented as a final application of the new continuous adjoint formulation.

Nomenclature

Variable Definition

c	Airfoil chord length
\vec{d}	Force projection vector
js	Scalar function defined at each point on S
\vec{n}	Unit normal vector
p	Static pressure
p_∞	Freestream pressure
\vec{r}	Position vector from the frame rotation center to a point in the flow domain
\vec{r}_o	Specified frame rotation center
\vec{u}_r	Velocity due to rotation at a point, $\vec{\omega} \times \vec{r}$
\vec{v}	Flow velocity vector in the inertial frame
\vec{v}_r	Relative flow velocity vector, $\vec{v} - \vec{u}_r$
v_∞	Freestream velocity
\vec{A}	Euler flux Jacobian matrices
A_z	Projected area in the z -direction
C_d	Coefficient of drag
C_l	Coefficient of lift
C_p	Coefficient of pressure
C_Q	Coefficient of torque, $Q/(0.5\rho_\infty\pi R^3(\omega R)^2)$
C_T	Coefficient of thrust, $T/(0.5\rho_\infty\pi R^2(\omega R)^2)$
E	Total energy per unit mass
\vec{F}	Euler convective fluxes
\vec{F}_{rot}	Rotating Euler convective fluxes
H	Stagnation enthalpy

*Ph.D. Candidate, Department of Aeronautics & Astronautics, AIAA Student Member.

†Engineering Research Associate, Department of Aeronautics & Astronautics, AIAA Member.

‡Associate Professor, Department of Aeronautics & Astronautics, AIAA Senior Member.

H_m	Mean curvature of a surface
\bar{I}	Identity matrix
J	Cost function defined as an integral over S
M_∞	Freestream Mach number
Q	Rotor torque
\mathcal{Q}	Vector of source terms
R	Rotor radius
$\mathcal{R}(U)$	System of governing flow equations
S	Solid wall domain boundary (design surface)
T	Rotor thrust
\mathcal{T}	Rotation matrix for transforming data between periodic boundaries
U	Vector of conservative variables
W	Vector of characteristic variables
α	Angle of attack
β	Sideslip angle
γ	Ratio of specific heats, $\gamma = 1.4$ for air
ρ	Fluid density
ρ_∞	Freestream density
$\vec{\phi}$	Adjoint velocity vector
$\vec{\omega}$	Specified angular velocity vector of the rotating frame
ω	Angular velocity magnitude
Γ	Far-field domain boundary
Ψ	Vector of adjoint variables
Ω	Flow domain

Mathematical Notation

\vec{b}	Spatial vector $b \in \mathbb{R}^n$, where n is the dimension of the physical cartesian space (in general, 2 or 3)
B	Column vector or matrix B , unless capitalized symbol clearly defined otherwise
\vec{B}	$\vec{B} = (B_x, B_y)$ in two dimensions or $\vec{B} = (B_x, B_y, B_z)$ in three dimensions
$\nabla(\cdot)$	Gradient operator
$\nabla \cdot (\cdot)$	Divergence operator
$\partial_n(\cdot)$	Normal gradient operator at a surface point, $\vec{n}_S \cdot \nabla(\cdot)$
$\nabla_S(\cdot)$	Tangential gradient operator at a surface point, $\nabla(\cdot) - \partial_n(\cdot)$
\cdot	Vector inner product
\times	Vector cross product
\otimes	Vector outer product
B^T	Transpose operation on column vector or matrix B
$\delta(\cdot)$	Denotes first variation of a quantity

I. Introduction and Motivation

ENVIRONMENTAL pressures to decrease both fuel burn and emissions, coupled with fuel price volatility, continue to drive the need for more efficient aircraft propulsion technology. Open rotor propulsion systems have long been studied due to their potential for game-changing advances in propulsive efficiency. Previous flight testing by General Electric, Pratt & Whitney, and NASA [1] has shown that open rotor engines lowered specific fuel consumption for an MD-80 and a B727 between 20-40 % depending on the type of engine being replaced and the choice of flight cruise Mach number. Furthermore, this fuel burn reduction was considered conservative, as it was based on a demonstrator engine alone, and not a finalized, production engine. Other work has estimated that open rotors could save approximately 30 % of the fuel cost for a medium range transport, and about 12 % of the total direct operating cost of the aircraft [2]. These results were based on dated fuel prices which were lower than current values, and emphasis should be placed on the likelihood of fuel price volatility continuing in the future.

Although the technology is promising, significant challenges must be addressed before a wide-spread adoption of open rotors occurs: assessing possible increases in noise, the potential aircraft structural weight penalties of noise insulation or protection from blade-out, and potential weight and/or cost penalties of

the open rotors as compared to turbofans [3]. These same challenges also offer an opportunity for the aircraft designer to take advantage of synergistic interactions between the configuration design and open rotor installation. New proposals for unconventional aircraft configurations or engine placement may target enhanced aerodynamic performance, noise shielding, or provide safety in the event of blade-out. These complex systems will require high fidelity analysis and system-level integration studies in order to assess the viability of the open rotor as a next generation propulsion system. Interaction effects during installation and the potential for large increases in noise have recently garnered much interest in the research community [4–8].

As a first step toward the multidisciplinary design of open rotors, this paper describes a new adjoint-based methodology to be used for the efficient optimal shape design of rotating aerodynamic surfaces, such as rotor blades. By focusing on an axisymmetric, single rotor configuration, the governing Euler flow equations can be recast into a rotating frame of reference moving with the body, and this transformation allows for the steady solution of a problem which was unsteady in the inertial frame. Adjoint-based formulations for optimal shape design of steady problems have a rich history in aeronautics, and their effectiveness is well established [9–11].

Rotor design has long been pursued using various techniques, but to our knowledge, only several publications have addressed adjoint-based shape design using the non-inertial governing flow equations. Lee and Kwon [12] presented a continuous adjoint formulation for inviscid, hovering rotor flows on unstructured meshes. More recently, discrete adjoint formulations for the Reynolds-averaged Navier-Stokes (RANS) equations in a rotating frame have been shown by Nielsen et al. [13] with the Spalart-Allmaras turbulence model on unstructured meshes and by Dumont et al. [14] with the $k - \omega$ turbulence model and the shear stress transport correction on structured meshes.

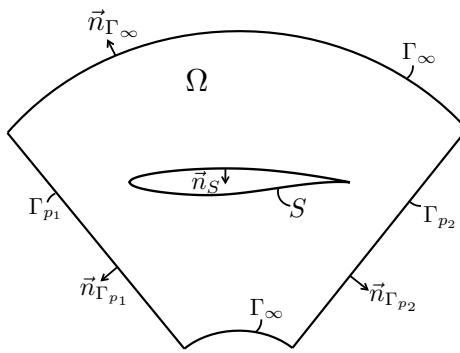


Figure 1. Notional schematic of the flow domain, Ω , and the disconnected boundaries with their corresponding surface normals: S , Γ_∞ , as well as new periodic boundaries, Γ_p .

While a discrete adjoint approach can often be more straightforward to implement, especially if automatic differentiation is available, we pursue in this article a continuous approach. The continuous formulation can offer the advantage of physical insight into the character of the governing flow equations and their adjoint system, and this insight can aid in composing well-behaved numerical solution methods. More specifically, the treatment given in this article is a systematic methodology for the compressible Euler equations centered around finding *surface sensitivities* with the use of differential geometry formulas. The surface sensitivities show the designer exactly where shape changes will have the most effect on a chosen objective function. This type of surface formulation has no dependence on volume mesh sensitivities and has been successfully applied to full aircraft configurations and even extended to the RANS equations [15, 16]. It is here extended for the optimal shape design of steadily rotating surfaces. Note that this methodology is general and could be used for the design of other rotating bodies such as propellers, turbomachinery, wind turbines, etc.

As there are few applications of these techniques and existing examples have been applied mostly within the rotorcraft research community, a novel application to optimal shape design of open rotor blades is presented. A Free-Form Deformation (FFD) approach to geometry parameterization allows for advanced design variable definitions during shape optimization. The combination of the adjoint formulation, unstructured meshes, and the FFD approach give the designer more freedom to explore non-intuitive design spaces involving complex geometries such as highly twisted, swept open rotor blades.

The contributions of this article are the following: a detailed derivation of the continuous adjoint formulation emphasizing surface sensitivities, practical numerical implementation details for central schemes including a new type of dissipation switch, two- and three-dimensional gradient verification for the adjoint against finite differencing, a gradient accuracy study, and three optimal shape design examples including open rotor blades.

The paper begins with a description of the physical problem in Section II, including the governing flow equations with corresponding boundary conditions. Section III contains a detailed derivation of the continuous adjoint formulation for the compressible Euler equations in a rotating reference frame. This

derivation is generalized to two or three dimensions. Numerical implementation strategies such as numerical methods, design variable definition, and mesh deformation appear in Section IV. Lastly, Section V presents numerical results and discussion for both two- and three-dimensional numerical experiments, including a final example for the optimal shape design of open rotor blades.

II. Description of the Physical Problem

Ideal fluids are governed by the Euler equations. In our particular problem, these equations are considered in a domain, Ω , bounded by a disconnected boundary which is divided into a far-field component, Γ_∞ , a solid wall boundary, S , and periodic boundary faces, Γ_p , as seen in Fig. 1. The surface S will also be referred to as the design surface, and it is considered continuously differentiable (C^1). In practical shape design applications, the assumption of differentiability does not hold for sharp corners or edges that might appear along trailing edges or the tips of wings or rotor blades. Special considerations must be made at these locations during the design process, and these will be discussed later with other numerical implementation details. Normal vectors to the boundary surfaces are directed out of the domain by convention.

The governing flow equations in the limit of vanishing viscosity are the compressible Euler equations. When simulating fluid flow about certain aerodynamic bodies that operate under an imposed steady rotation, including many turbomachinery, propeller, and rotor applications, it can be advantageous to transform the system of Euler equations into a reference frame that rotates with the body of interest. With this transformation, a flow field that is unsteady when viewed from the inertial frame can be solved for in a steady manner, and thus more efficiently, without the need for grid motion. For conciseness, this formulation of the governing system will be referred to as the rotating Euler equations.

Considering the flow domain of Fig. 1 after performing the appropriate transformation into a reference frame that rotates with a steady angular velocity, $\vec{\omega} = \{\omega_x, \omega_y, \omega_z\}^T$, and a specified rotation center, $\vec{r}_o = \{x_o, y_o, z_o\}^T$, the absolute velocity formulation [17] of the steady, rotating Euler equations in conservation form is

$$\begin{cases} \mathcal{R}(U) = \nabla \cdot \vec{F}_{rot} - \mathcal{Q} = 0 & \text{in } \Omega, \\ (\vec{v} - \vec{u}_r) \cdot \vec{n}_S = 0 & \text{on } S, \\ W_+ = W_\infty & \text{on } \Gamma_\infty, \\ U_1 = \mathcal{T}U_2 & \text{on } \Gamma_{p_1}, \\ U_2 = \mathcal{T}^{-1}U_1 & \text{on } \Gamma_{p_2}, \end{cases} \quad (1)$$

where

$$U = \begin{Bmatrix} \rho \\ \rho\vec{v} \\ \rho E \end{Bmatrix}, \quad \vec{F}_{rot} = \begin{Bmatrix} \rho(\vec{v} - \vec{u}_r) \\ \rho\vec{v} \otimes (\vec{v} - \vec{u}_r) + \bar{I}\bar{p} \\ \rho H(\vec{v} - \vec{u}_r) + \vec{u}_r p \end{Bmatrix}, \quad \mathcal{Q} = \begin{Bmatrix} 0 \\ -\rho(\vec{\omega} \times \vec{v}) \\ 0 \end{Bmatrix},$$

where ρ is the fluid density, $\vec{v} = \{u, v, w\}^T$ is the absolute flow velocity, E is the total energy per unit mass, H is the total enthalpy per unit mass, p is the static pressure, and \vec{u}_r is the velocity due to rotation ($\vec{u}_r = \vec{\omega} \times \vec{r}$). Here, \vec{r} is the position vector pointing from the rotation center to a point (x, y, z) in the flow domain, or $\vec{r} = \{(x - x_o), (y - y_o), (z - z_o)\}^T$. The velocity due to rotation is also sometimes called the whirl velocity. The second and third lines of Eq. 1 represent the solid wall and characteristic-based far-field boundary conditions, respectively, with an adjustment for rotation. Rotor simulations often take advantage of rotational periodicity by solving for the flow around a single blade with periodic boundaries rather than the entire set of blades, and this can decrease computational cost greatly. With rotational periodicity (fourth and fifth lines of Eq. 1), vector flow quantities at each point, such as the momentum, must be rotated using a transformation matrix, \mathcal{T} , so that the angle of incidence between the vector and the domain boundary remains the same when transforming the state to the corresponding periodic face. The transformation matrix, \mathcal{T} , can be found in the appendix. In order to close the system of equations after assuming a perfect gas, the pressure is determined from

$$p = (\gamma - 1)\rho \left[E - \frac{1}{2}(\vec{v} \cdot \vec{v}) \right], \quad (2)$$

and the stagnation enthalpy is given by

$$H = E + \frac{p}{\rho}. \quad (3)$$

It is important to note that not all simulations of rotating bodies can benefit from this solution approach. The flow field must be steady in the rotating frame, and some conditions or geometric features, such as relative surface motion, can cause unsteadiness for rotating bodies. If the incoming flow velocity is not parallel to the axis of rotation, the conditions are no longer axisymmetric, and the blades would not see a steady field during rotation. In this case, the rotating Euler equations would not provide an efficient, steady solution method.

III. Surface Sensitivities Using a Continuous Adjoint Methodology

The objective of this section is to describe the way in which we quantify the influence of geometric modifications on the pressure distribution at a solid surface in the flow domain. Again, the motivation for pursuing a continuous formulation based on surface sensitivities includes the following important considerations: manipulation of the continuous equations offers direct insight into the mathematical character of the governing equations, there is no dependence on volume mesh sensitivities when computing the first variation of a functional (only a surface integral remains), and the effect of local surface shape modifications can be directly visualized using the surface sensitivities to further designer intuition. It is here that the continuous adjoint formulation is systematically derived following a procedure similar to that of Bueno-Orovio et al. [15].

A typical shape optimization problem seeks the minimization of a certain cost function, J , with respect to changes in the shape of the boundary, S . For example, for the designer concerned with aerodynamic performance, an obvious choice for J might be the coefficient of drag on S . Therefore, we will concentrate on functionals defined as integrals over the solid surface S ,

$$J = \int_S j_S ds, \quad (4)$$

where j_S is a scalar function defined at each point on S . As the designer choosing the shape of S , the question we would like to answer is the following: what effect does a change in the shape of S have on the value of J ? In the case of drag minimization mentioned above, shape deformations that reduce the drag on the surface are desired.

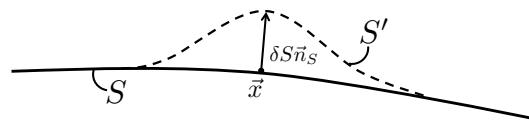
Therefore, the goal is to compute the variation of Eq. 4 caused by arbitrary but small (and multiple) deformations of S and to use this information to drive our geometric changes in order to find an optimal shape for the design surface, S . This leads directly to a gradient-based optimization framework. The shape deformations applied to S will be infinitesimal in nature and can be described mathematically by

$$S' = \{\vec{x} + \delta S(\vec{x})\vec{n}_S(\vec{x}), \vec{x} \in S\}, \quad (5)$$

Figure 2. An infinitesimal shape deformation in the local surface normal direction.

where S has been deformed to a new surface S' by applying an infinitesimal profile deformation, δS , in the local normal direction, \vec{n}_S , at a point, \vec{x} , on the surface, as shown in Fig. 2. Upon application of the surface deformation, the cost function varies due to the changes in the solution induced by the deformation:

$$\delta J = \int_{\delta S} j_S ds + \int_S \delta j_S ds. \quad (6)$$



Note that taking the variation results in two separate terms. The first term depends on the variation of the geometry and the value of the scalar function in the original state, while the second term depends on the original geometry and the variation of the scalar function caused by the deformation.

With the basic outline of the shape design problem described, let us more specifically define the scalar function as

$$j_S = \vec{d} \cdot (p \vec{n}_S). \quad (7)$$

The vector \vec{d} is the force projection vector, and it is an arbitrary, constant vector which can be chosen to relate the pressure, p , at the surface to a desired quantity of interest. For aerodynamic applications, likely candidates are

$$\vec{d} = \begin{cases} \left(\frac{1}{C_\infty}\right) (\cos \alpha \cos \beta, \sin \alpha \cos \beta, \sin \beta), & C_D \quad \text{Drag coefficient,} \\ \left(\frac{1}{C_\infty}\right) (-\sin \alpha, \cos \alpha, 0), & C_L \quad \text{Lift coefficient,} \\ \left(\frac{1}{C_\infty}\right) (-\sin \beta \cos \alpha, -\sin \beta \sin \alpha, \cos \beta), & C_{SF} \quad \text{Side-force coefficient,} \\ \left(\frac{1}{C_\infty C_D}\right) (-\sin \alpha - \frac{C_L}{C_D} \cos \alpha \cos \beta, -\frac{C_L}{C_D} \sin \beta, \cos \alpha - \frac{C_L}{C_D} \sin \alpha \cos \beta), & \frac{C_L}{C_D} \quad \text{L/D,} \\ \left(\frac{1}{C_\infty}\right) (-1, 0, 0), & C_T \quad \text{Thrust coefficient,} \\ \left(\frac{1}{C_\infty L_{ref}}\right) (0, (z - z_o), -(y - y_o)), & C_Q \quad \text{Torque coefficient,} \end{cases} \quad (8)$$

where $C_\infty = \frac{1}{2} v_\infty^2 \rho_\infty A_z$, v_∞ is the freestream velocity, ρ_∞ is the freestream density, L_{ref} is a reference length for computing moments, and A_z is the reference area. In practice for a three-dimensional surface, we often sum up all positive components of the normal surface vectors in the z -direction in order to calculate the projection A_z . A pre-specified reference area can also be used in a similar fashion, and this is an established procedure in applied aerodynamics. For the thrust and torque coefficient projection vectors in three dimensions, it is assumed that the axis of rotation is aligned with the positive x -axis, such that thrust is in the negative x -direction.

Starting from Eq. 6, Eq. 7 can be introduced, and further manipulation can be performed:

$$\begin{aligned} \delta J &= \int_{\delta S} \vec{d} \cdot (p \vec{n}_S) ds + \int_S \delta[\vec{d} \cdot (p \vec{n}_S)] ds \\ &= \int_S \vec{d} \cdot [\partial_n(p \vec{n}_S) - 2H_m(p \vec{n}_S)] \delta S ds + \int_S \vec{d} \cdot (\delta p \vec{n}_S + p \delta \vec{n}_S) ds \\ &= \int_S \vec{d} \cdot [\partial_n(p \vec{n}_S) - 2H_m(p \vec{n}_S)] \delta S ds + \int_S (\vec{d} \cdot \vec{n}_S) \delta p ds - \int_S \vec{d} \cdot [p \nabla_S(\delta S)] ds \\ &= \int_S \vec{d} \cdot [\partial_n(p \vec{n}_S) - 2H_m(p \vec{n}_S)] \delta S ds + \int_S (\vec{d} \cdot \vec{n}_S) \delta p ds - \int_S [\nabla_S \cdot (p \vec{d} \delta S) - \vec{d} \cdot \nabla_S(p) \delta S] ds \\ &= \int_S \vec{d} \cdot [\partial_n(p \vec{n}_S) + \nabla_S(p) - 2H_m(p \vec{n}_S)] \delta S ds + \int_S (\vec{d} \cdot \vec{n}_S) \delta p ds \\ &= \int_S (\vec{d} \cdot \nabla p) \delta S ds + \int_S (\vec{d} \cdot \vec{n}_S) \delta p ds, \end{aligned} \quad (9)$$

where we have used relationships from differential geometry in the following order

$$\begin{cases} \int_{\delta S} q ds = \int_S [\partial_n(q) - 2H_m q] \delta S ds, \\ \delta \vec{n}_S = -\nabla_S(\delta S), \\ \int_S \nabla_S \cdot (\vec{q}) ds = 0, \\ \nabla \cdot \vec{q} = \partial_n(\vec{q} \cdot \vec{n}_S) + \nabla_S(\vec{q}) - 2H_m(\vec{q} \cdot \vec{n}_S). \end{cases} \quad (10)$$

The second relationship in Eq. 10 holds for small deformations [18], q is an arbitrary scalar function, \vec{q} is an arbitrary vector, and H_m is the mean curvature of S computed as $(\kappa_1 + \kappa_2)/2$, where (κ_1, κ_2) are curvatures in two orthogonal directions on the surface. Also note that we have used integration by parts to expand the final term in going from the third to fourth line of Eq. 9.

Eq. 9 states that evaluating the variation of the cost function requires information about the geometry and its variation, as well as the pressure and its variation at the design surface. While the pressure at all points on the surface can be determined from a single flow solution in the domain (i.e., solving the governing flow equations within Ω with suitable boundary conditions at Γ and S), obtaining the variation of the pressure for multiple, arbitrary surface deformations is not as straightforward. In fact, as expressed in Eq. 9, calculating δJ for a number, N , of shape deformations would require N linearized flow solutions in order to compute the value of δp corresponding to each deformation. Ideally, the explicit dependence on δp in the variation of the functional would be removed, so that the variation due to an arbitrary number of deformations can be computed in a much more efficient manner.

In order to accomplish this, we note that any variations of the flow variables are constrained to satisfy the system of governing flow equations, $\mathcal{R}(U) = 0$. Therefore, our optimal shape problem can be considered a constrained optimization problem, and we can build a Lagrangian using the original cost function, Eq. 4, and the governing equations to transform it into an unconstrained optimization problem:

$$J = \int_S j_S ds + \int_{\Omega} \Psi^T \mathcal{R}(U) d\Omega, \quad (11)$$

where we have introduced the adjoint variables, which operate as Lagrange multipliers and are defined as

$$\Psi = \begin{Bmatrix} \psi_{\rho} \\ \psi_{\rho u} \\ \psi_{\rho v} \\ \psi_{\rho w} \\ \psi_{\rho E} \end{Bmatrix} = \begin{Bmatrix} \psi_{\rho} \\ \vec{\varphi} \\ \psi_{\rho E} \end{Bmatrix}. \quad (12)$$

Now, reconsider the process of finding the first variation of J when applying the surface deformation. The variation of the governing flow equations due to the change in the surface shape also appears:

$$\delta J = \int_S (\vec{d} \cdot \nabla p) \delta S ds + \int_S (\vec{d} \cdot \vec{n}_S) \delta p ds + \int_{\Omega} \Psi^T \delta \mathcal{R}(U) d\Omega. \quad (13)$$

The third term of Eq. 13 will provide the new information necessary for removing the dependence on δp , and thus, we must linearize the governing equations with respect to small perturbations of the design surface to find $\delta \mathcal{R}(U)$. First, consider the governing equations and, more specifically, the convective fluxes for the rotating frame formulation. The contributions due to rotation can be separated from the traditional fluxes for the Euler equations, \vec{F} , as

$$\vec{F}_{rot} = \begin{Bmatrix} \rho(\vec{v} - \vec{u}_r) \\ \rho \vec{v} \otimes (\vec{v} - \vec{u}_r) + \bar{I}p \\ \rho H(\vec{v} - \vec{u}_r) + \vec{u}_r p \end{Bmatrix} = \begin{Bmatrix} \rho \vec{v} \\ \rho \vec{v} \otimes \vec{v} + \bar{I}p \\ \rho H \vec{v} \end{Bmatrix} - \begin{Bmatrix} \rho \vec{u}_r \\ \rho \vec{v} \otimes \vec{u}_r \\ \rho E \vec{u}_r \end{Bmatrix} = \vec{F} - (U \otimes \vec{u}_r), \quad (14)$$

where Eq. 3 has been used to express the rotational contributions purely in terms of the conservative variables, U . By substituting this result into Eq. 1, the following form of the governing equations is retrieved,

$$\mathcal{R}(U) = \nabla \cdot \vec{F} - \nabla \cdot (U \otimes \vec{u}_r) - \mathcal{Q} = 0 \text{ in } \Omega, \quad (15)$$

which retain the same boundary conditions as described above for the original rotating frame formulation. The linearization of Eq. 15 results in

$$\begin{aligned} \delta \mathcal{R}(U) &= \nabla \cdot \delta \vec{F} - \nabla \cdot \delta(U \otimes \vec{u}_r) - \delta \mathcal{Q} \\ &= \nabla \cdot \left(\frac{\partial \vec{F}}{\partial U} \delta U \right) - \nabla \cdot \left[\frac{\partial(U \otimes \vec{u}_r)}{\partial U} \delta U \right] - \frac{\partial \mathcal{Q}}{\partial U} \delta U \\ &= \nabla \cdot \left(\vec{A} - \bar{I} \vec{u}_r \right) \delta U - \frac{\partial \mathcal{Q}}{\partial U} \delta U, \end{aligned} \quad (16)$$

with the linearized form of the boundary condition at the surface,

$$\delta \vec{v} \cdot \vec{n}_S = -(\vec{v} - \vec{u}_r) \cdot \delta \vec{n}_S - \partial_n(\vec{v} - \vec{u}_r) \cdot \vec{n}_S \delta S, \quad (17)$$

where \vec{A} is the Jacobian of \vec{F} using conservative variables, and the Jacobian of the source term using conservative variables, $\frac{\partial \mathcal{Q}}{\partial U}$, is given in the appendix. Details on the linearization of the solid wall boundary condition are also contained in the appendix. Eq. 16 can now be introduced into Eq. 13 to produce

$$\delta J = \int_S (\vec{d} \cdot \nabla p) \delta S ds + \int_S (\vec{d} \cdot \vec{n}_S) \delta p ds + \int_{\Omega} \Psi^T \nabla \cdot \left(\vec{A} - \bar{I} \vec{u}_r \right) \delta U d\Omega - \int_{\Omega} \Psi^T \frac{\partial \mathcal{Q}}{\partial U} \delta U d\Omega. \quad (18)$$

To remove the dependence on δp , we now perform manipulations such that the domain integrals can either be eliminated or transformed into surface integrals. Integrating the first domain integral by parts gives

$$\int_{\Omega} \nabla \cdot [\Psi^T (\vec{A} - \bar{\vec{I}}\vec{u}_r) \delta U] d\Omega - \int_{\Omega} \nabla \Psi^T \cdot (\vec{A} - \bar{\vec{I}}\vec{u}_r) \delta U d\Omega - \int_{\Omega} \Psi^T \frac{\partial \mathcal{Q}}{\partial U} \delta U d\Omega, \quad (19)$$

and applying the divergence theorem to the first term of Eq. 19, assuming a smooth solution, gives

$$\int_S \Psi^T (\vec{A} - \bar{\vec{I}}\vec{u}_r) \cdot \vec{n}_S \delta U ds + \int_{\Gamma_{\infty}} \Psi^T (\vec{A} - \bar{\vec{I}}\vec{u}_r) \cdot \vec{n}_S \delta U ds - \int_{\Omega} \left[\nabla \Psi^T \cdot (\vec{A} - \bar{\vec{I}}\vec{u}_r) + \Psi^T \frac{\partial \mathcal{Q}}{\partial U} \right] \delta U d\Omega, \quad (20)$$

where the final two terms in Eq. 19 have been combined into a single domain integral. While we have assumed a smooth solution here with the use of the divergence theorem, formulations supporting discontinuities, such as shocks, have also been developed [19]. With the appropriate choice of boundary conditions, the integral over the far-field boundary can be forced to vanish. The domain integral can also be made to vanish, if its integrand is zero at every point in the domain. When set equal to zero, the terms within the brackets constitute the set of partial differential equations which are commonly referred to as the adjoint equations. Therefore, the domain integral will vanish provided that the adjoint equations are satisfied as

$$\nabla \Psi^T \cdot (\vec{A} - \bar{\vec{I}}\vec{u}_r) + \Psi^T \frac{\partial \mathcal{Q}}{\partial U} = 0 \text{ in } \Omega, \quad (21)$$

or after taking the transpose

$$(\vec{A} - \bar{\vec{I}}\vec{u}_r)^T \cdot \nabla \Psi + \frac{\partial \mathcal{Q}^T}{\partial U} \Psi = 0 \text{ in } \Omega. \quad (22)$$

The accompanying boundary condition on S for the adjoint equations will be presented below.

At this point, only the first term in Eq. 20 remains. The surface integral can be evaluated by hand given our knowledge of the governing equations to give

$$\int_S \Psi^T (\vec{A} - \bar{\vec{I}}\vec{u}_r) \cdot \vec{n}_S \delta U ds = \int_S (\delta \vec{v} \cdot \vec{n}_S) (\rho \psi_{\rho} + \rho \vec{v} \cdot \vec{\varphi} + \rho H \psi_{\rho E}) ds + \int_S [\vec{n}_S \cdot \vec{\varphi} + \psi_{\rho E} (\vec{v} \cdot \vec{n}_S)] \delta p ds, \quad (23)$$

where we have used the flow boundary condition, $(\vec{v} - \vec{u}_r) \cdot \vec{n}_S = 0$, in evaluating \vec{A} at the surface, S . To eliminate the dependence on $\delta \vec{v} \cdot \vec{n}_S$, we will use the linearized boundary condition on S , Eq. 17, to obtain

$$\begin{aligned} & \int_S \Psi^T (\vec{A} - \bar{\vec{I}}\vec{u}_r) \cdot \vec{n}_S \delta U ds \\ &= - \int_S [\partial_n (\vec{v} - \vec{u}_r) \cdot \vec{n}_S \delta S + (\vec{v} - \vec{u}_r) \cdot \delta \vec{n}_S] \vartheta ds + \int_S [\vec{n}_S \cdot \vec{\varphi} + \psi_{\rho E} (\vec{v} \cdot \vec{n}_S)] \delta p ds \\ &= - \int_S [\partial_n (\vec{v} - \vec{u}_r) \cdot \vec{n}_S \delta S - \nabla_S (\delta S) \cdot (\vec{v} - \vec{u}_r)] \vartheta ds + \int_S [\vec{n}_S \cdot \vec{\varphi} + \psi_{\rho E} (\vec{v} \cdot \vec{n}_S)] \delta p ds \\ &= - \int_S \{\partial_n (\vec{v} - \vec{u}_r) \cdot \vec{n}_S \vartheta + \nabla_S \cdot [(\vec{v} - \vec{u}_r) \vartheta]\} \delta S ds + \int_S [\vec{n}_S \cdot \vec{\varphi} + \psi_{\rho E} (\vec{v} \cdot \vec{n}_S)] \delta p ds, \end{aligned} \quad (24)$$

where as a shorthand, $\vartheta = \rho \psi_{\rho} + \rho \vec{v} \cdot \vec{\varphi} + \rho H \psi_{\rho E}$. To obtain this last expression we have used the linearized boundary condition, geometric relations from the second and third lines of Eq. 10, and we have also used the product rule in going from the second to the third line. After performing these manipulations to the integrals, we update δJ from Eq. 18:

$$\begin{aligned} \delta J &= \int_S (\vec{d} \cdot \nabla p) \delta S ds + \int_S (\vec{d} \cdot \vec{n}_S) \delta p ds + \int_S \{\partial_n (\vec{v} - \vec{u}_r) \cdot \vec{n}_S \vartheta + \nabla_S \cdot [(\vec{v} - \vec{u}_r) \vartheta]\} \delta S ds \\ &\quad - \int_S [\vec{n}_S \cdot \vec{\varphi} + \psi_{\rho E} (\vec{v} \cdot \vec{n}_S)] \delta p ds. \end{aligned} \quad (25)$$

The last step is to rearrange the variation of the functional as

$$\delta J = \int_S \left\{ \vec{d} \cdot \nabla p + \partial_n(\vec{v} - \vec{u}_r) \cdot \vec{n}_S \vartheta + \nabla_S \cdot [(\vec{v} - \vec{u}_r) \vartheta] \right\} \delta S ds + \int_S \left[\vec{d} \cdot \vec{n}_S - \vec{n}_S \cdot \vec{\varphi} - \psi_{\rho E}(\vec{v} \cdot \vec{n}_S) \right] \delta p ds, \quad (26)$$

where we have separated the different terms to have a clear view of those depending on the flow variations (first integral) and those depending on the current flow state, adjoint variables, or geometry, which are all considered known quantities (second integral).

Eq. 26 is the key to evaluating the functional sensitivity with respect to deformations on the design surface, S . Recall that the original goal of the preceding manipulations was to eliminate the explicit dependence on δp appearing in Eq. 9. In comparing Eqs. 9 & 26, we see that our manipulations have introduced new terms into both integrals, and we can make use of this development to eliminate the first integral. As the integral is over the design surface, the new terms provide the mechanism for forcing the integrand to zero in the form of a boundary condition for the adjoint equations at the solid surface. Therefore, the adjoint equations with the admissible adjoint boundary condition that eliminates the dependence on the fluid flow variations (δp) can be written as

$$\begin{cases} \left(\vec{A} - \bar{\vec{I}} \vec{u}_r \right)^T \cdot \nabla \Psi + \frac{\partial \mathcal{Q}}{\partial U}^T \Psi = 0 & \text{in } \Omega, \\ \vec{n}_S \cdot \vec{\varphi} = \vec{d} \cdot \vec{n}_S - \psi_{\rho E}(\vec{v} \cdot \vec{n}_S) & \text{on } S, \end{cases} \quad (27)$$

with the appropriate far-field boundary conditions, and the variation of the objective function becomes

$$\delta J = \int_S \left\{ \vec{d} \cdot \nabla p + \partial_n(\vec{v} - \vec{u}_r) \cdot \vec{n}_S \vartheta + \nabla_S \cdot [(\vec{v} - \vec{u}_r) \vartheta] \right\} \delta S ds = \int_S \frac{\partial J}{\partial S} \delta S ds, \quad (28)$$

where $\frac{\partial J}{\partial S} = \vec{d} \cdot \nabla p + \partial_n(\vec{v} - \vec{u}_r) \cdot \vec{n}_S \vartheta + \nabla_S \cdot [(\vec{v} - \vec{u}_r) \vartheta]$ is what we call the *surface sensitivity*. The surface sensitivity provides a measure of the variation of the objective function with respect to infinitesimal variations of the surface shape in the direction of the local surface normal. It can be further simplified for ease of computation:

$$\begin{aligned} \frac{\partial J}{\partial S} &= \vec{d} \cdot \nabla p + \partial_n(\vec{v} - \vec{u}_r) \cdot \vec{n}_S \vartheta + \nabla_S \cdot [(\vec{v} - \vec{u}_r) \vartheta] \\ &= \vec{d} \cdot \nabla p + (\nabla \cdot \vec{v}) \vartheta + (\vec{v} - \vec{u}_r) \cdot \nabla(\vartheta), \end{aligned} \quad (29)$$

where the product rule, the relationship $\nabla \cdot \vec{u}_r = \nabla \cdot (\vec{\omega} \times \vec{r}) = 0$, and the solid wall boundary condition, $(\vec{v} - \vec{u}_r) \cdot \vec{n}_S = 0$, were used in simplifying to retrieve the final line of Eq. 29. This value is computed at each surface node of the numerical grid with negligible computational cost.

IV. Numerical Implementation

In this section, we explain several numerical implementation details that were critical components of a robust solution methodology. For combating stability issues during the solution of the adjoint equations, we present a new type of dissipation switch for central schemes. Without these special considerations, achieving high quality shape optimization results would be difficult for realistic applications.

A. Numerical Methods

Solution procedures for both the rotating Euler equations and the corresponding adjoint equations were implemented within the *SU²* software suite (Stanford University Unstructured). This collection of C++ codes is built specifically for PDE analysis and PDE-constrained optimization on unstructured meshes, and it is particularly well suited for aerodynamic shape design. Modules for performing direct and adjoint flow solutions, acquiring gradient information by projecting surface sensitivities into the design space, and deforming meshes are included in the suite, amongst others. Scripts written in the Python programming language are also used to automate execution of the *SU²* suite components, especially for performing shape optimizations.

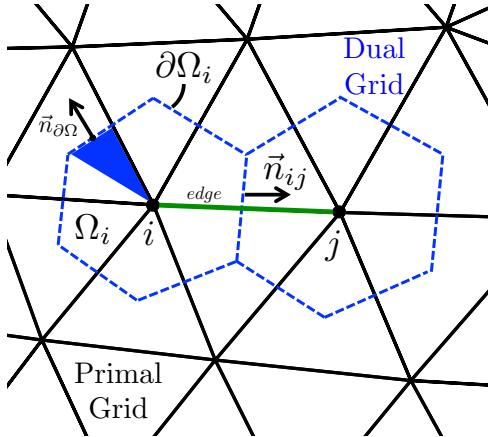


Figure 3. Schematic of the primal and dual mesh structure including the control volume, control volume faces, edges, and normals.

grid approach for convergence acceleration. The rotating Euler equations are spatially discretized using a central scheme with JST-type artificial dissipation [20], and the adjoint equations use a slightly modified JST scheme. Source terms are approximated using piecewise constant reconstruction within each of the finite volume cells. Further detail on the numerical solution procedures is given below.

1. Direct Problem

We highlight here several practical details for implementing the rotating Euler equations within an existing, edge-based solver using central schemes.

TIME STEP LIMIT Local time stepping can be used, as the solution is being marched in pseudo-time to the steady state in the rotating frame. The only modification required in order to compute the local time step for an element involves the procedure for finding the maximum eigenvalue. A contribution to the flow velocity through each face of the control volume due to the implied rotation must be included: $\vec{u}_r \cdot \vec{n}_{ij}$, where \vec{u}_r has been computed at the mid-point of the current edge.

ARTIFICIAL DISSIPATION For the JST scheme on unstructured grids in the context of an edge-based solver, artificial dissipation is computed using the differences in the undivided Laplacians (higher order) and the conserved variables (lower order) between the nodes on either end of the current edge [21]. If the level of dissipation is scaled based upon the maximum eigenvalue for arbitrarily shaped control volumes, then again, the contribution to the local velocity through the control volume face due to rotation must be included. The two levels of dissipation are blended by using the typical pressure switch for triggering lower-order dissipation in the vicinity of shock waves.

EXACT INTEGRATION OF THE VOLUME FLUX DUE TO ROTATION Assuming a steadily rotating frame and a non-deforming mesh, the adjustments to the convective fluxes along each edge due to rotation are entirely geometrical and can be exactly integrated and stored as a preprocessing step. Following the Finite Volume Method, the governing equations are integrated over a control volume, Ω_i ,

$$\int_{\Omega_i} \frac{\partial U}{\partial t} d\Omega_i + \int_{\Omega_i} \nabla \cdot \vec{F}_{rot} d\Omega_i = \int_{\Omega_i} \mathcal{Q} d\Omega_i, \quad (30)$$

which after integrating, using the Divergence theorem, and separating the traditional Euler convective flux from the rotating contribution (Eq. 15) becomes

$$|\Omega_i| \frac{\partial U}{\partial t} + \int_{\partial\Omega_i} \vec{F} \cdot \vec{n} dS_i - \int_{\partial\Omega_i} (U \otimes \vec{u}_r) \cdot \vec{n} dS_i = |\Omega_i| \mathcal{Q}. \quad (31)$$

The optimization results presented in this work make use of the *SciPy* library (<http://www.scipy.org>), a well-established, open-source software package for mathematics, science, and engineering. The *SciPy* library provides many user-friendly and efficient numerical routines for the solution of non-linear constrained optimization problems, such as conjugate gradient, Quasi-Newton, or sequential least-squares programming algorithms. At each design iteration, the *SciPy* routines only require the values and gradients of the objective functions as inputs, computed by means of our continuous adjoint approach, as well as the set of any chosen constraints.

Both the direct and adjoint problems are solved numerically using a Finite Volume Method formulation with an edge-based structure. The median-dual vertex-based scheme stores instances of the solution at the nodes of the primal grid and constructs the dual mesh around these nodes by connecting the surrounding cell centers and the mid-points of the edges between the primal grid nodes, as seen in Fig. 3. The solver is capable of both explicit and implicit psuedo-time integration for relaxing the solution to the steady-state. The code is fully parallel and takes advantage of an agglomeration multi-grid approach for convergence acceleration.

The rotating Euler equations are spatially discretized using a central scheme with JST-type artificial dissipation [20], and the adjoint equations use a slightly modified JST scheme. Source terms are approximated using piecewise constant reconstruction within each of the finite volume cells. Further detail on the numerical solution procedures is given below.

Discretizing the equations results in a system of coupled ODEs to be time-marched toward a steady state,

$$|\Omega_i| \frac{dU_i}{dt} + \sum_{j \in \mathcal{N}_i} \vec{F}_{ij} \cdot \vec{n}_{ij} dS_{ij} - \int_{\partial\Omega_i} (U \otimes \vec{u}_r) \cdot \vec{n} dS_i = |\Omega_i| \mathcal{Q}, \quad (32)$$

where \mathcal{N}_i is the set of edges connecting node i to the neighboring nodes, $|\Omega_i|$ is the cell volume, and \vec{F}_{ij} is the numerical flux. The integral in the third term on the left hand side of Eq. 32 can be integrated exactly. In practice, the integral is computed by finding the surface normal for each sub-face of a control volume (shaded wedge inside Ω_i and accompanying surface normal, $\vec{n}_{\partial\Omega}$ in Fig. 3) and taking the inner product of this value with the rotational velocity, \vec{u}_r , computed at the centroid of the sub-face. For each edge of the control volume, this constant integrated volume flux is stored from the adjacent sub-faces as a preprocessing step. After computing the convective flux along each edge in a control volume during the time integration scheme, the average value of the conserved variables is multiplied by the rotating volume flux and the result is subtracted from the residual. A different implementation involves computing both the average of the rotational velocity and conserved variables at the mid-point of each edge and using the edge normal to compute the rotating volume flux. This method is less efficient, but results in solutions within approximately 1 % of the exact integration method depending on the mesh.

2. Adjoint Problem

The adjoint equations are similarly discretized in space and marched in pseudo-time to their steady solution, and some practical issues dealing with their solution are given here. These strategies were needed to obtain high quality shape design results.

DISSIPATION SWITCH The rotating frame version of the adjoint equations suffered from stability issues, and special attention was required when computing the adequate artificial dissipation for second-order accurate schemes. In regions of the flow near sonic or stagnation points, the adjoint equations would suffer from an instability often leading to divergence. This was particularly apparent at the leading edge of airfoils and rotor blades.

In order to restore stability, we propose a modified JST-type scheme with lower and higher order dissipation that is blended through a new switch based on the adjoint variables rather than the traditional pressure switch from the direct problem. More specifically, the dissipation switch is constructed by applying the limiter of Venkatakrishnan [22] to the adjoint density variable. This technique, borrowed from the formulation of slope limiters for upwind schemes, requires the gradient of the adjoint variables and can be used to locate the regions of largest solution variability in space. In these regions of highest variability, additional lower order dissipation is added, just as additional dissipation is added near shocks in the traditional JST scheme.

It is important to note that this dissipation strategy was made possible by our continuous adjoint approach. By handling the continuous equations and time-marching them to a steady state with a central scheme, stability issues could be easily identified and fixed by focusing on solution methods for the adjoint PDEs rather than modifying the numerical grid. The modified JST scheme described above relieved convergence issues, especially for three-dimensional problems.

SURFACE SENSITIVITY AT SHARP EDGES The continuous adjoint derivation in this article specifically assumes a continuously differentiable design surface. With realistic, complex geometries, such as rotor blades or complete aircraft configurations, sharp corners or edges can be quite common along trailing edges or wing tips. At the nodes on these sharp edges, the local surface normal is not well defined, and this leads to incorrect values of the surface sensitivity at these nodes and errors in the gradient if changes in the design variables cause a movement of the sharp edge.

Additional terms arising from integration by parts when corners are present can be added to the computational geometry relationships used in Eqns. 9 & 10 in order to correct errors in the gradient, and this is currently being pursued. However, as will be shown with the current formulation, accurate values for the gradient will result as long as the edge nodes remain fixed in space. This is easily accomplished by selecting the appropriate design variables. For example, FFD boxes for deforming the design surface shape should exclude sharp edges with the current adjoint formulation, and this strategy will be used for the three-dimensional rotor results presented in this paper.

B. Design Variable Definition and Mesh Deformation

The above adjoint derivation presented a method for computing the variation of an objective function with respect to infinitesimal surface shape deformations in the direction of the local surface normal at points on the design surface. While it is possible to use each surface node in the computational mesh as a design variable capable of deformation, this approach is not often pursued. A more practical choice is to compute the surface sensitivities at each mesh node on the design surface and then project this information into a design space made up of a smaller set (possibly a complete basis) of design variables. The procedure for computing the surface sensitivities is used repeatedly in a gradient-based optimization framework in order to march the design surface shape toward an optimum through gradient projection and mesh deformation.

For the numerical results presented in this paper, two methodologies for design variable definition were used. In two-dimensional airfoil calculations, Hicks-Henne bump functions were employed [23] which can be added to the original airfoil geometry to modify the shape. The Hicks-Henne function with maximum at point x_n is given by

$$f_n(x) = \sin^3(\pi x^{e_n}), \quad e_n = \frac{\log(0.5)}{\log(x_n)}, \quad x \in [0, 1], \quad (33)$$

so that the total deformation of the surface can be computed as $\Delta y = \sum_{n=1}^N \delta_n f_n(x)$, with N being the number of bump functions and δ_n the design variable step. These functions are applied separately to the upper and lower surfaces. After applying the bump functions to recover a new surface shape with each design iteration, a spring analogy method is used to deform the volume mesh around the airfoil.

In three dimensions, a Free-Form Deformation (FFD) [24] strategy has been adopted. In FFD, an initial box encapsulating the object (rotor blade, wing, fuselage, etc.) to be redesigned is parameterized as a Bézier solid. A set of control points are defined on the surface of the box, the number of which depends on the order of the chosen Bernstein polynomials. The solid box is parameterized by the following expression

$$X(u, v, w) = \sum_{i,j,k=0}^{l,m,n} P_{i,j,k} B_j^l(u) B_j^m(v) B_k^n(w), \quad (34)$$

where $u, v, w \in [0, 1]$, and B^i is the Bernstein polynomial of order i . The Cartesian coordinates of the points on the surface of the object are then transformed into parametric coordinates within the Bézier box. Control points of the box become design variables, as they control the shape of the solid, and thus the shape of the surface grid inside. The box enclosing the geometry is then deformed by modifying its control points, with all the points inside the box inheriting a smooth deformation. Once the deformation has been applied, the new Cartesian coordinates of the object of interest can be recovered by simply evaluating the mapping inherent in Eq. 34.

Once the boundary displacements have been computed using the FFD strategy, a classical spring method is used in order to deform the rest of vertices of the unstructured mesh. [25] The method is based on the definition of a stiffness matrix, k_{ij} , that connects the two ends of a single bar (mesh edge). Equilibrium of forces is then imposed at each mesh node

$$\left(\sum_{j \in \mathcal{N}_i} k_{ij} \vec{e}_{ij} \vec{e}_{ij}^T \right) \vec{u}_i = \sum_{j \in \mathcal{N}_i} k_{ij} \vec{e}_{ij} \vec{e}_{ij}^T \vec{u}_j, \quad (35)$$

where the displacement \vec{u}_i is unknown and is computed as a function of the known surface displacements \vec{u}_j , \mathcal{N}_i is the set of neighboring points to node i , and \vec{e}_{ij} the unit vector in the direction connecting both points. The system of equations is solved iteratively by a conjugate gradient algorithm with Jacobi preconditioning.

V. Numerical Results

A. Shape Design of a Rotating Airfoil

As a verification test for the gradient information obtained by the continuous adjoint formulation, a numerical experiment was devised for a NACA 0012 airfoil rotating in still air ($M_\infty = 0$) which can be solved using the rotating Euler equations. The flow is two-dimensional in the x - y plane with rotation out of the page in

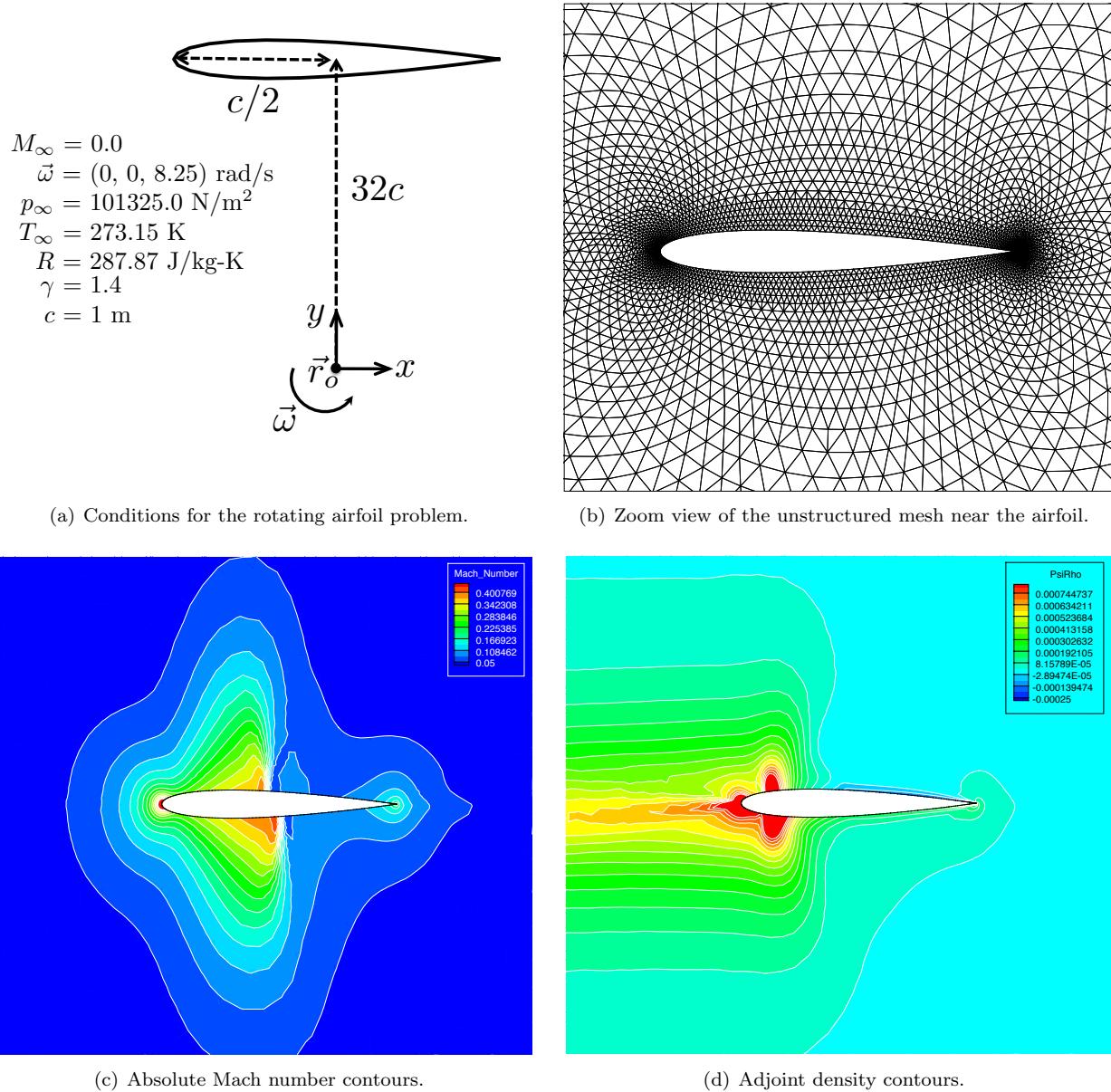
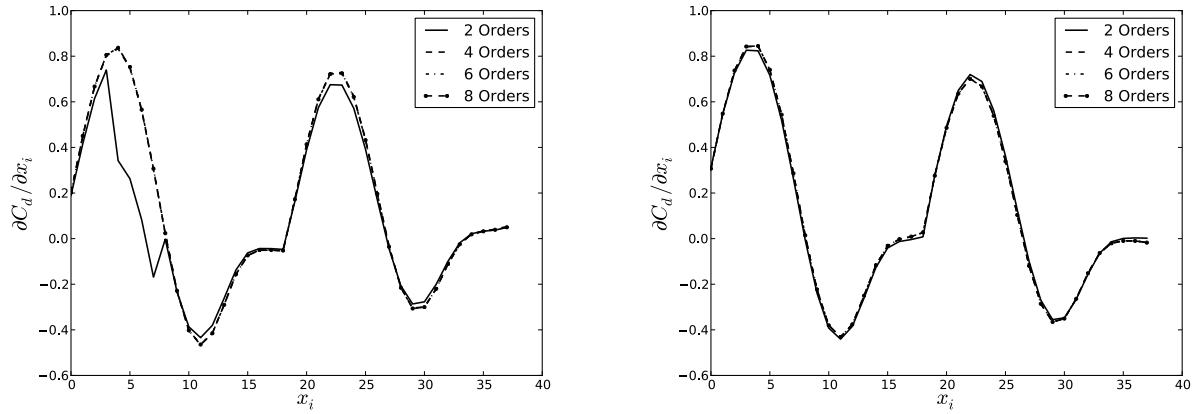


Figure 4. Details for the two-dimensional numerical experiment, the computational mesh, and solutions for the initial geometry.

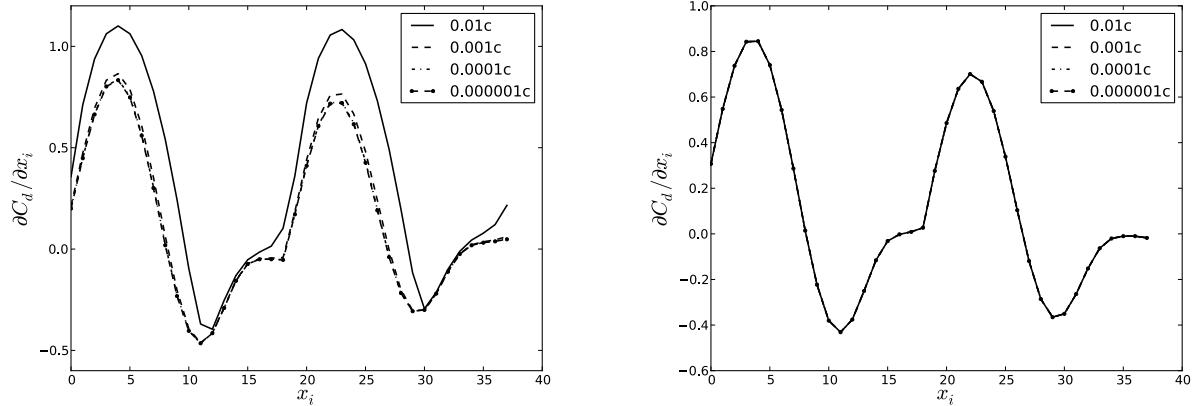
the z -direction. The specific flow conditions for the problem, and in particular the angular velocity of the airfoil rotation, were chosen such that transonic shocks appeared on the upper and lower airfoil surfaces. The goals of the test case are two-fold: to verify the gradient of C_d with respect to a set of Hicks-Henne design variables obtained from the continuous adjoint formulation against finite differencing, and to perform an unconstrained airfoil shape optimization for minimizing C_d . The details for the numerical experiment and the unstructured mesh appear in Fig. 4. The mesh consisted of 10,216 triangular elements, 5,233 nodes, 200 edges along the airfoil, and 50 edges along the far-field boundary.

Fig. 4 shows the absolute Mach number contours around the airfoil. In the inertial frame, the Mach numbers appear entirely subsonic as the air is pushed out of the path of the rotating airfoil. However, there are clear shock structures on both the upper and lower surface, and when the velocity due to rotation is taken into account to form the relative velocity, the local Mach number near the airfoil surface exceeds one. Fig. 4 also presents contours for ψ_ρ near the surface. Note the strong features near the nose and sonic points in the adjoint solution. Convergence issues often originated in these regions, but the modified dissipation



(a) C_d gradients from finite differencing based upon various levels of convergence in the density residual (order of magnitude reduction). Very low levels of convergence affect the gradient accuracy. The step size for each case was 0.0001c.

(b) C_d gradients for the continuous adjoint based upon various levels of convergence in the density and adjoint density residuals (order of magnitude reduction). The results here show that the adjoint gradients are fairly insensitive to convergence level. The gradient projection step size for each case was 0.0001c.



(c) C_d gradients from finite differencing with various step sizes. It is clear that the step size impacts the accuracy of the gradient information, and that a sufficiently small step must be taken. Little difference is apparent between 0.0001c and 0.000001c. All cases were converged 8 orders of magnitude in the density residual.

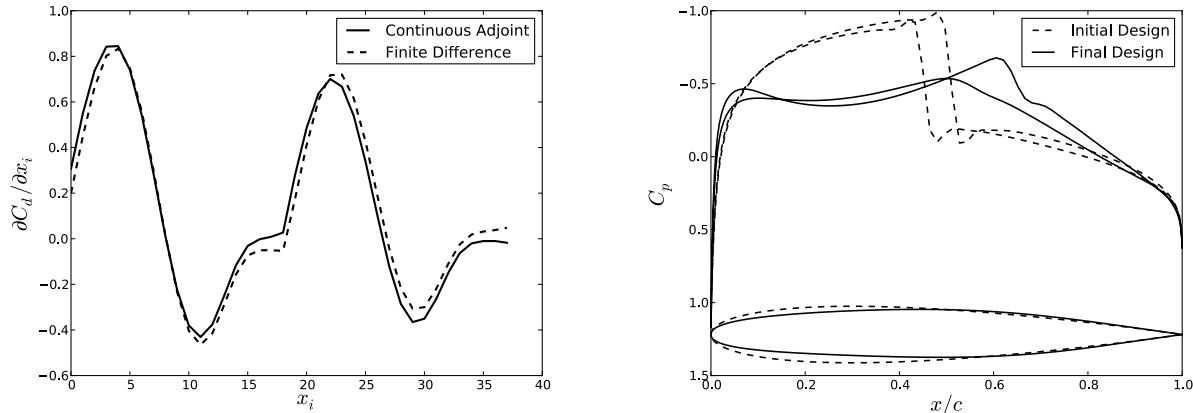
(d) C_d gradients for the continuous adjoint with different gradient projection step sizes for the Hicks-Henne bump deformations. As expected, there is no dependence on the step size for the adjoint, as the surface sensitivities are computed independently of the gradient projection into the design space. The solutions were converged 8 orders of magnitude in the density and adjoint density residuals.

Figure 5. Comparison studies between the continuous adjoint and finite differencing for the gradient of C_d . The set of Hicks-Henne bump function design variables, x_i , are along the x -axis.

switch successfully located and alleviated the issues by adding extra dissipation only where necessary.

In order to verify the accuracy of the gradient information obtained by the continuous adjoint formulation, 37 Hicks-Henne bump functions were chosen as design variables along the upper and lower surfaces of the NACA 0012. A comparison was then made between the gradient of C_d with respect to the design variables resulting from the surface sensitivities found using the continuous adjoint approach and a finite differencing approach using small step sizes for the bump deformations. For this problem, the force coefficients (C_l , C_d , and C_p) were computed using ρ_∞ , p_∞ , and the velocity due to rotation at the nose of the airfoil. The gradients compare favorably, although there are slight differences between the adjoint and finite differencing, as seen in Fig. 6. Further studies were performed to explore the sensitivity of the gradients to both the step size of the bump deformations and the level of convergence attained by the flow solver for both the direct and adjoint problems. These comparisons are given in Fig. 5. Similar to gradient accuracy results shown by

Kim et al. [26], the finite difference gradients are quite sensitive to the chosen step size and level of solver convergence, whereas the adjoint gradients are largely insensitive to these parameters.



(a) Direct comparison of the gradients obtained by the continuous adjoint and finite differencing. Direct and adjoint solutions were converged 8 orders of magnitude in the density residual and adjoint density residual, respectively. The step size for the bump deformations was 0.000001c.

(b) C_p and profile shape comparison for the initial rotating NACA 0012 and the minimum drag airfoil. The optimizer has made the airfoil thinner, especially the forward half, and it is no longer symmetric.

Figure 6. Direct comparison of the gradients and a comparison of the initial and final airfoil designs.

Finally, a redesign of the rotating airfoil was performed using the gradient information obtained from the adjoint formulation. The specific shape optimization problem was for unconstrained drag minimization with respect to the Hicks-Henne design variables. Upon completion, the C_d was successfully reduced from 0.006887 down to 0.000317, which is a 95.4 % reduction. The value of C_l began at -0.0666 for the initial NACA 0012 design and was -0.0448 for the final design. Note that the lift is in the negative y -direction. C_p distributions as well as the profile shapes of the initial and final designs are compared in Fig. 6, and pressure contours around the initial and final designs are shown in Fig. 7. It is clear from the results that the optimization process has eliminated the two transonic shocks on the upper and lower surfaces that originally appeared on the rotating NACA 0012.

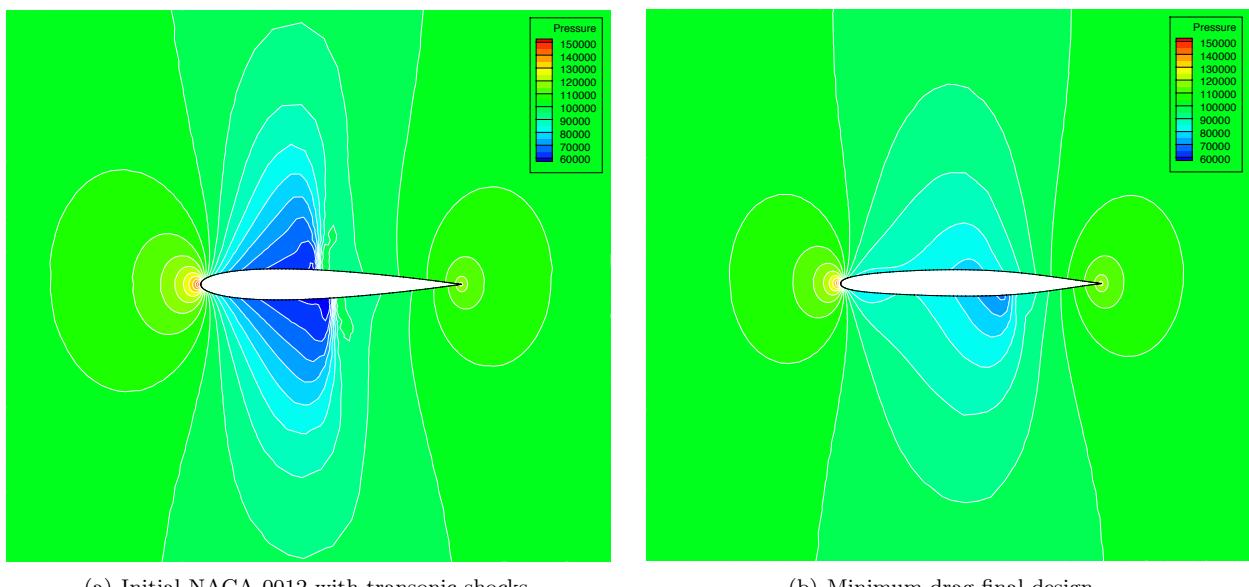
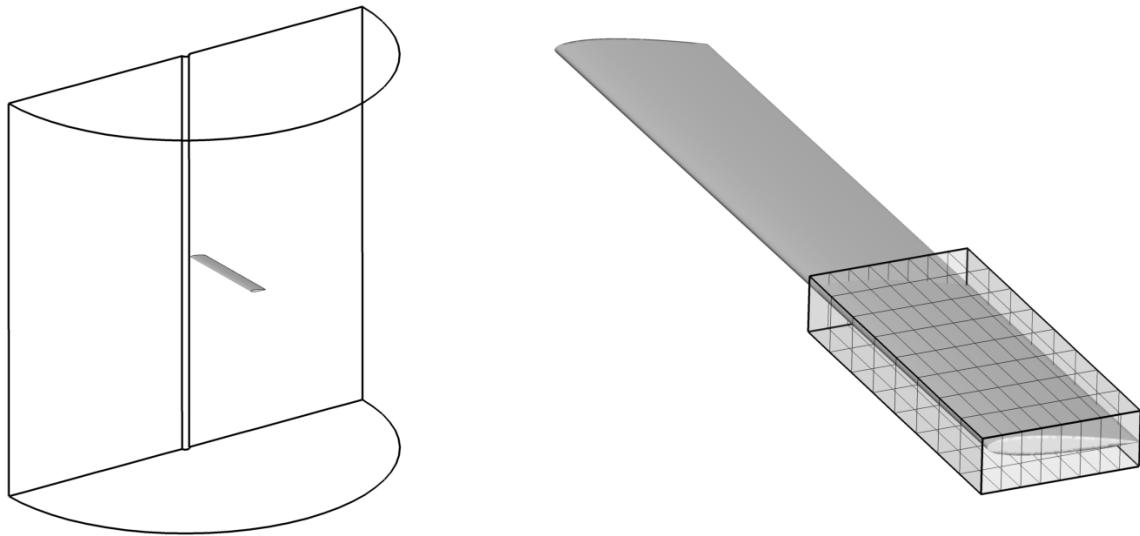


Figure 7. Pressure contours (N/m^2) for the initial and final rotating airfoil designs.

B. Redesign of a Rotor in Hover

For the first design case in three dimensions, a simple rotor geometry was chosen such that comparisons with the experimental data of Caradonna and Tung [27] could be made. The rotor geometry consists of two untwisted, untapered blades with an aspect ratio of 6 and a constant NACA 0012 airfoil section along the entire span. For comparison purposes, a lifting case was chosen with a collective pitch angle of 8 degrees and a pre-cone angle of 0.5 degrees. The flow conditions are that of hover at 2500 RPM which results in a tip Mach number of 0.877.



(a) Mesh topology showing the rotor blade surface, far-field boundaries, and periodic faces making up the half-cylinder. (b) Rotor geometry with the FFD box surrounding the blade tip.

Figure 8. Mesh and FFD box details for the Caradonna and Tung numerical experiment.

The computational mesh, seen in Fig. 8, takes advantage of rotational periodicity by simulating flow over a single blade in a half-cylinder. The hybrid-element mesh uses 3.36 million tetrahedra and 19,422 pyramids with a total of 588,572 nodes. The outer faces of the domain are close enough to the rotor that care must be taken to allow for subsonic induced velocities, including the rotor wake, to pass through them. Therefore, characteristic-based inlet and outlet boundary conditions were chosen. Flow tangency is satisfied at the blade surface and the small central hub.

Fig. 9 contains the C_p contours on the upper blade surface along with C_p distributions at several radial stations compared to experiment. Note the shock near the blade tip where the flow is transonic, and as seen in the C_p comparisons with experiment, the shock is crisply captured with the current inviscid scheme. Also shown in Fig. 9 are surface sensitivity contours from the rotating adjoint solution based on a torque objective function, C_Q . It should be noted that the most sensitive locations on the blade surface are in the vicinity of the shock and the expansion region upstream of the shock. Visualizing the surface sensitivities in this manner can offer designer intuition and can also aid the designer in defining appropriate design variables for automatic shape design.

In order to verify the gradient information and optimize the rotor geometry, design variables were defined using a FFD parameterization. First, a box was generated around the tip region of the blade where shape changes are to be made with the design variables becoming the displacement of the individual control points that define the FFD box. Note again that the surface sensitivities have guided the design variable definition by locating the FFD box around the more sensitive tip region. Movements in the vertical direction were allowed for 84 control points in total on the upper and lower surfaces of the FFD box. In order to maintain a smooth surface during deformation, control points near the trailing edge and inboard side of the FFD box were held fixed. The trailing edge of the rotor blade was not included in the FFD box, which is in line with the discussion above concerning surface sensitivities at sharp edges. Before attempting shape design, the gradients of C_Q with respect to a subset of the FFD control point variables on the upper surface given by both the continuous adjoint and finite differencing were compared. Fig. 8 contains a view of the FFD

box around the blade tip, and the gradient comparison appears in Fig. 10. Again, while the adjoint and finite differencing gradients are not identical, they exhibit similar character and differences in line with expectations for the two approaches.

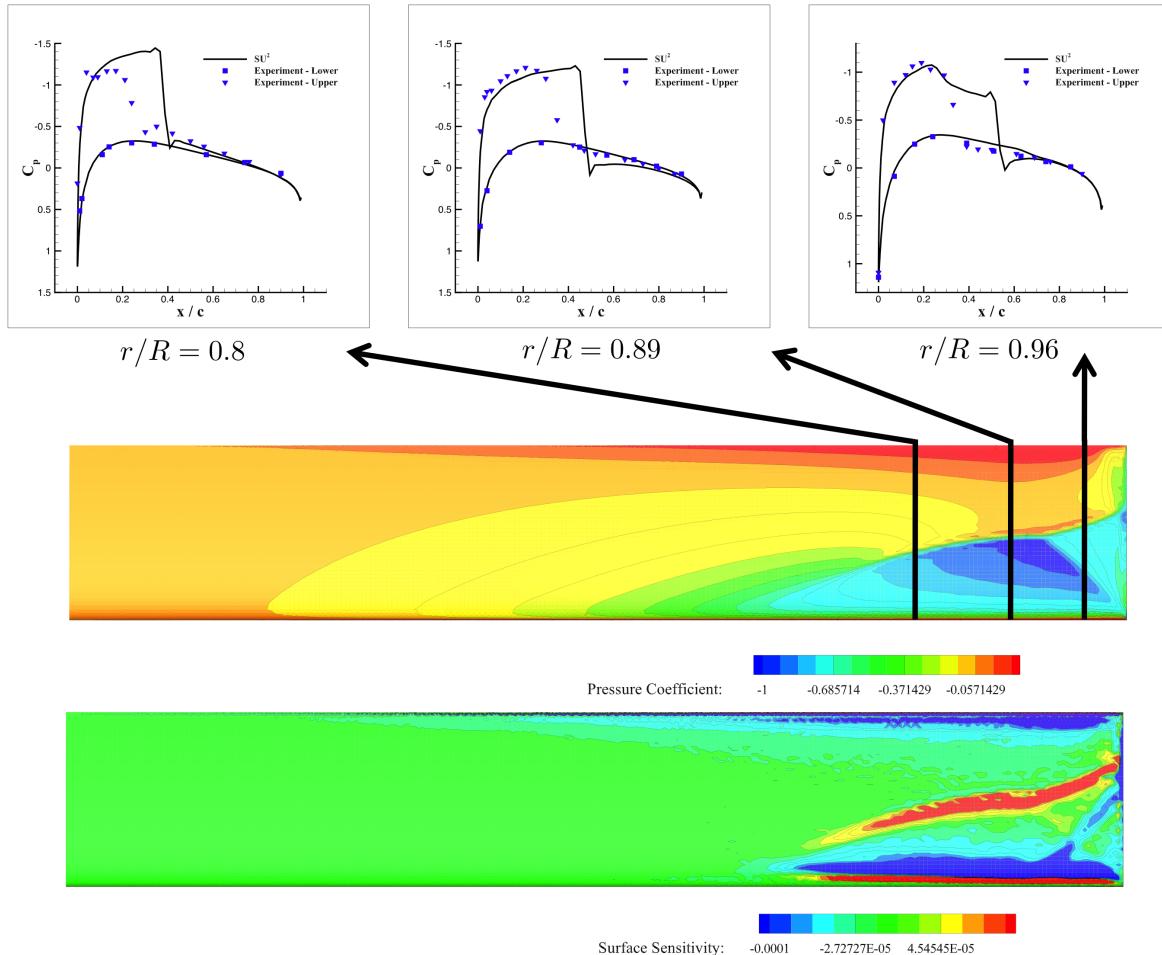
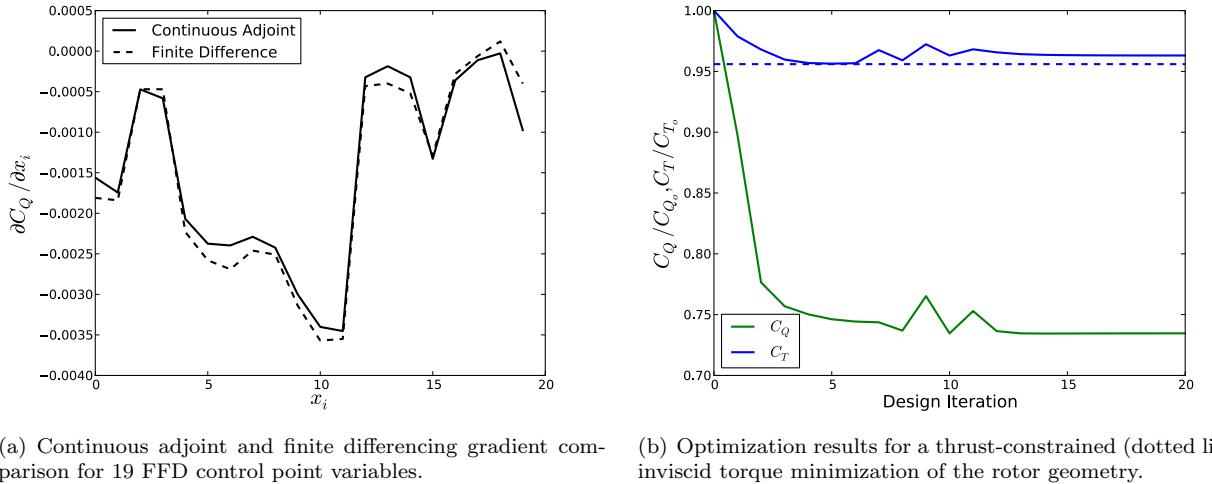


Figure 9. C_p contours on the upper surface of the initial rotor geometry and comparison to experiment at multiple span locations. The right end is the blade tip which is rotating toward the bottom of the page. The surface sensitivity contours for a torque objective function are also shown. Note the high sensitivity to shape deformations in the vicinity of the shock.

Lastly, a redesign of the rotor blade shape for minimizing torque with a minimum thrust constraint of $C_T = 0.0055$ was performed using gradient information obtained via the continuous adjoint approach. After 20 design cycles, C_Q was reduced by 26.9 % from 0.0006098 to 0.0004458 while maintaining a C_T value of 0.00553 from a starting value of 0.00575. These optimization results are presented in Fig. 10. The initial and final surface shapes with C_p contours are compared in Fig. 11. The strong shock on the upper surface has been removed due to a pronounced change in the shape near the tip. The optimized design features a blade tip with a sharper, downturned leading edge and a thinner, asymmetric section shape.

C. Open Rotor Blade Shape Design

Open rotor blades will operate in a transonic flow regime due to a desire for high effective bypass ratios (high propulsive efficiency) at or slightly below current commercial aircraft cruise speeds. This has led to relatively large diameter rotors with swept sections to delay the development of shocks near the blade tips. As an example for using the continuous adjoint method to design complex geometries, a generic open rotor blade shape has been optimized for minimum inviscid torque at low-speed (take-off) conditions. These conditions were chosen because the acoustic signature, another design consideration of current research interest, during



(a) Continuous adjoint and finite differencing gradient comparison for 19 FFD control point variables.

(b) Optimization results for a thrust-constrained (dotted line) inviscid torque minimization of the rotor geometry.

Figure 10. Gradient verification using the FFD control point variables and optimization results.

take-off directly affects community noise. When designing the full system, both aerodynamics and acoustics should be considered at multiple points in the flight envelope, such as during take-off and cruise, and this will be addressed in future work.

A generic, 8-bladed, single rotor configuration was created, and for simplicity, the rotor is installed on an “infinite” hub. All blades are identical and were designed for a sea level thrust requirement of 10,000 lbs. The thrust requirement was chosen so that if coupled with a second counter-rotating set of blades, the engine would produce approximately 20,000 lbs. of thrust and be appropriate for a typical single-aisle commercial aircraft. The blades use a NACA 65-series airfoil, and the starting rotor configuration had a 4.27 m diameter with a hub-to-tip ratio of 0.36, similar to a generic configuration by Stuermer and Yin [7]. The initial blade shape was designed for minimum induced losses based on blade element momentum theory considerations [28] with a small modification for considering blade sweep.

The simulation again took advantage of rotational periodicity by solving for the flow about only one of the eight blades using periodic boundary conditions. The computational mesh uses the same topology and boundary condition specification as the Caradonna and Tung rotor case, and it consists of 387,791 tetrahedra and 14,652 pyramids with a total of 79,446 nodes. The baseline geometry can be seen in Fig. 12. For performing shape design, a FFD box with the same number of control point design variables as the previous rotor case was placed around the tip of the blade. This is also shown in Fig. 12.

A redesign of the rotor blade shape for minimizing inviscid torque with a thrust constraint was performed. Initial operating conditions were set for take-off at sea level with $M_\infty = 0.23$ and an RPM of 1337, and at these conditions, a shock appears near the tip of the rotor blade. A thrust constraint of $C_T = 0.0067$ was imposed during the redesign in order to maintain the original design thrust of 10,000 lbs. at sea level. After 20 design cycles, C_Q was reduced by 5.7 % from 0.002501 to 0.002357 while maintaining a C_T value of 0.0067. These optimization results are presented in Fig. 13. The initial and final surface shapes with C_p contours are also compared in Fig. 13. The initial shock near the trailing edge of the blade tip has been removed and replaced with regions of more gradual pressure variations. This example exhibits the viability of the method for the design of advanced geometries. Even when considering changes to only a small region of the geometry, the designer can realize substantial performance improvements in an automatic way.

VI. Conclusions

A continuous adjoint formulation for the rotating Euler equations has been present, verified, and applied. This formulation allows for the design of rotating aerodynamic bodies in a gradient-based optimization framework. More specifically, the treatment given in this article is a systematic methodology for the compressible Euler equations centered around finding *surface sensitivities* with the use of differential geometry formulas which has no dependence on volume mesh sensitivities when computing the first variation of a functional

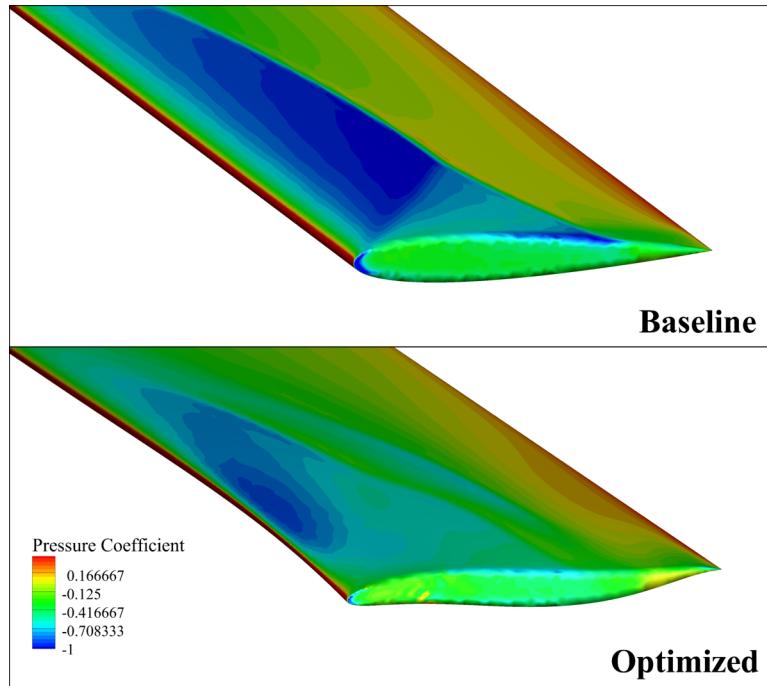
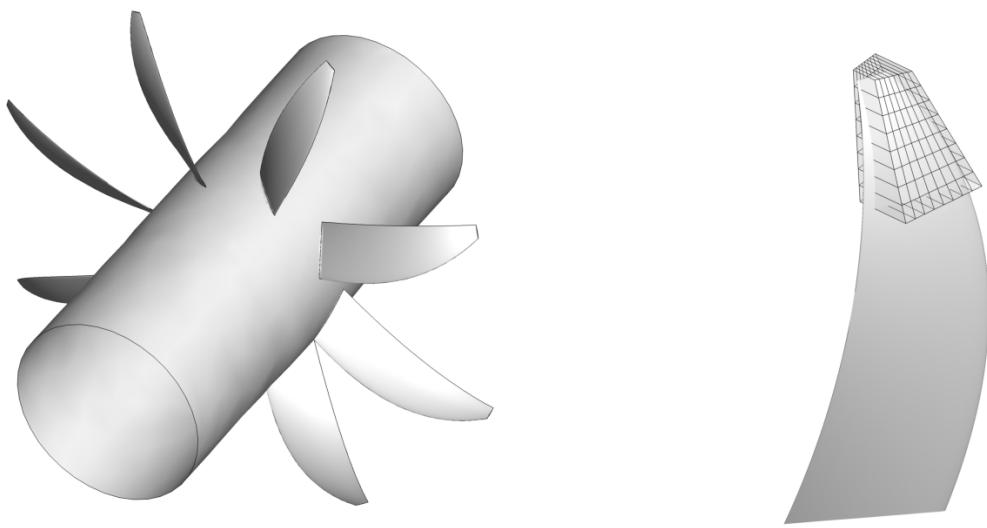


Figure 11. Comparison of the baseline and optimized rotor geometries with C_p contours. The strong shock has been removed due to a distinct change in the tip shape.



(a) The generic open rotor geometry installed on an “infinite” hub. (b) FFD box placement near the tip of the open rotor blade.

Figure 12. Details for the open rotor blade numerical experiment.

(only a surface integral remains). To further designer intuition, the surface sensitivities can be visualized to clearly locate regions on the surface where shape changes will have the most effect on a chosen objective function. The methodology is general and could be used for the design of other rotating bodies such as propellers, compressors, or turbines.

The continuous formulation offers the advantage of physical insight into the character of the governing

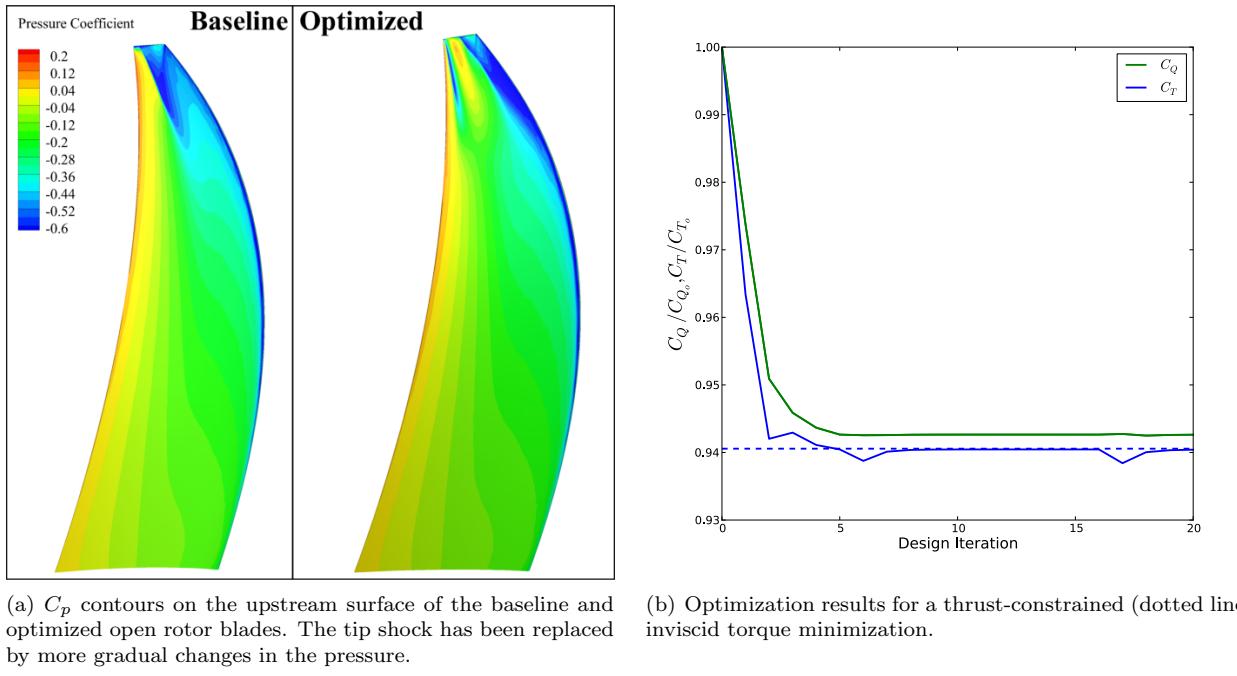


Figure 13. Results for the open rotor blade shape design.

flow equations and their adjoint system, and in this article, the insight was used to form a modified central scheme to alleviate convergence issues arising during the solution of the adjoint system. Along with strategies for defining Free-Form Deformation (FFD) variables, the new dissipation switch proposed in the modified central scheme was essential to obtaining quality solution results when performing optimal shape design.

The gradient information provided by the surface formulation has been verified for design variables in two and three dimensions. In both situations, the gradients compared favorably with those obtained via finite differencing. In a gradient accuracy study, it was found that while the gradients obtained by finite differencing were very sensitive to design variable step sizes and solver convergence, the continuous adjoint gradient information was largely insensitive to these parameters.

Optimal shape design involving the new adjoint formulation was demonstrated for three separate cases: a two-dimensional airfoil, a classic, three-dimensional rotor geometry, and open rotor blades. In each case, improvements in performance were realized, and in three dimensions, realistic thrust constraints were applied during optimization. Since there are few current applications of these techniques, open rotor blades represent a novel application. The combination of the adjoint formulation, modified numerical solution methods on unstructured meshes, and the FFD approach offers a powerful optimal shape design procedure and gives the designer more freedom to explore non-intuitive design spaces involving complex geometries such as highly twisted, swept open rotor blades.

VII. Acknowledgements

T. Economou would like to acknowledge U.S. government support under and awarded by DoD, Air Force Office of Scientific Research, National Defense Science and Engineering Graduate (NDSEG) Fellowship, 32 CFR 168a.

References

- ¹Reid, C., "Overview of Flight Testing of GE Aircraft Engines UDF Engine," *AIAA Paper 1988-3082*, AIAA, ASME, SAE, and ASEE, Joint Propulsion Conference, 24th, Boston, Massachusetts, July 11-13, 1988.
- ²Sadler, J. H. R., and Hodges, G. S., "Turboprop and Open Rotor Propulsion for the Future," *AIAA Paper 1986-1472*, AIAA, ASME, SAE, and ASEE, Joint Propulsion Conference, 22nd, Huntsville, Alabama, June 16-18, 1986.

- ³Fischer, B., and Klug, H., "Configuration Studies for a Regional Airliner Using Open-Rotor Ultra-High-Bypass-Ratio Engines," *AIAA Paper 1989-2580*, AIAA/ASME/SAE, and ASEE, Joint Propulsion Conference, 25th, Monterey, California, July 10-12, 1989.
- ⁴Ricouard, J., Julliard, E., Omais, M., Regnier, V., Parry, A. B., Baralon, S., "Installation effects on contra-rotating open rotor noise," *AIAA 2010-3795*, 16th AIAA/CEAS Aeroacoustics Conference, Stockholm, Sweden, 2010.
- ⁵Stuermer, A., "Unsteady CFD Simulations of Contra-Rotating Propeller Propulsion Systems," *AIAA 2008-5218*, 44th AIAA/ASME/SAE Joint Propulsion Conference & Exhibit, Hartford, CT, 2008.
- ⁶Deconinck, T., Hoffer, P., Hirsch, C., De Muelenaere, A., Bonaccorsi, J., Ghorbaniasl, G., "Prediction of Near- and Far-Field Noise Generated by Contra-Rotating Open Rotors," *AIAA 2010-3794*, 16th AIAA/CEAS Aeroacoustics Conference, Stockholm, Sweden, 2010.
- ⁷Stuermer, A., Yin, J., "Low-Speed Aerodynamics and Aeroacoustics of CROR Propulsion Systems," *AIAA 2009-3134*, 15th AIAA/CEAS Aeroacoustics Conference, Miami, FL, 2009.
- ⁸Stuermer, A., Yin, J., "Aerodynamic and Aeroacoustic Installation Effects for Pusher-Configuration CROR Propulsion Systems," *AIAA 2010-4235*, 28th AIAA Applied Aerodynamics Conference, Chicago, IL, 2010.
- ⁹Jameson, A., "Aerodynamic Design Via Control Theory," *AIAA 81-1259*, 1981.
- ¹⁰Jameson, A., Alonso, J. J., Reuther, J., Martinelli, L., Vassberg, J. C., "Aerodynamic Shape Optimization Techniques Based On Control Theory," *AIAA-1998-2538*, 29th Fluid Dynamics Conference, Albuquerque, NM, June 15-18, 1998.
- ¹¹Anderson, W. K. and Venkatakrishnan, V., "Aerodynamic Design Optimization on Unstructured Grids with a Continuous Adjoint Formulation," *Journal of Scientific Computing*, Vol. 3, 1988, pp. 233-260.
- ¹²Lee, S. W., Kwon, O. J., "Aerodynamic Shape Optimization of Hovering Rotor Blades in Transonic Flow Using Unstructured Meshes," *AIAA Journal*, Vol. 44, No. 8, pp. 1816-1825, August, 2006.
- ¹³Nielsen, E. J., Lee-Rausch, E. M., Jones, W. T., "Adjoint-Based Design of Rotors using the Navier-Stokes Equations in a Noninertial Reference Frame," AHS International 65th Forum and Technology Display, Grapevine, TX, May 27-29, 2009.
- ¹⁴Dumont, A., Le Pape, A., Peter, J., Huberson, S., "Aerodynamic Shape Optimization of Hovering Rotors Using a Discrete Adjoint of the Reynolds-Averaged Navier-Stokes Equations," *Journal of the American Helicopter Society*, Vol. 56, No. 3, pp. 1-11, July, 2011.
- ¹⁵Bueno-Orovio, A., Castro, C., Palacios, F., and Zuazua, E., "Continuous Adjoint Approach for the Spalart-Allmaras Model in Aerodynamic Optimization," *AIAA Journal*, Vol. 50, No. 3, pp. 631-646, March 2012.
- ¹⁶Palacios, F., Alonso, J. J., Colombo, M., Hicken, J., and Lukaczyk, T., "Adjoint-Based method for supersonic aircraft design using equivalent area distribution," *AIAA-2012-269*, 50th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition, Nashville, Tennessee, Jan. 9-12, 2012.
- ¹⁷Holmes, D., Tong, S., S., "A Three-Dimensional Euler Solver for Turbomachinery Blade Rows," *Journal of Engineering for Gas Turbines and Power*, Vol. 107, April, 1985.
- ¹⁸Sokolowski, J., Zolesio, J.-P., *Introduction to Shape Optimization*, Springer Verlag, New York, 1991.
- ¹⁹Baeza, A., Castro, C., Palacios, F., and Zuazua, E., "2-D Euler Shape Design on Nonregular Flows Using Adjoint Rankine-Hugoniot Relations," *AIAA Journal*, Vol. 47, No. 3, pp. 552-562, 2009.
- ²⁰Jameson, A., Schmidt, W., and Turkel, E., "Numerical Solution of the Euler Equations by Finite Volume Methods Using Runge-Kutta Time-Stepping Schemes," *AIAA 81-1259*, 1981.
- ²¹Mavriplis, D., "Accurate Multigrid Solution of the Euler Equations on Unstructured and Adaptive Meshes," *AIAA Journal*, Vol. 28, No. 2, pp. 213-221, 1990.
- ²²Venkatakrishnan, V., "Convergence to Steady State Solutions of the Euler Equations on Unstructured Grids with Limiters," *Journal of Computational Physics*, Vol. 118, No. 1, pp. 120-130, 1995.
- ²³Hicks, R. and Henne, P., "Wing design by numerical optimization," *Journal of Aircraft*, Vol. 15, pp. 407-412, 1978.
- ²⁴Samareh, J. A., "Aerodynamic shape optimization based on Free-Form deformation," *AIAA-2004-4630*, 10th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, Albany, New York, Aug. 2004.
- ²⁵Degand, C. and Farhat, C., "A three-dimensional torsional spring analogy method for unstructured dynamic meshes," *Computers & Structures*, Vol. 80, pp. 305-316, 2002.
- ²⁶Kim, S., Alonso, J. J., Jameson, A., "A Gradient Accuracy Study for the Adjoint-Based Navier-Stokes Design Method," *AIAA-99-0299*, 37th Aerospace Sciences Meeting and Exhibit, Reno, NV, Jan. 11-14, 1999.
- ²⁷Caradonna, F. X., Tung, C., "Experimental and Analytical Studies of a Model Helicopter Rotor in Hover," *NASA Technical Memorandum 81232*, NASA Ames Research Center, Moffett Field, CA, Sept., 1981.
- ²⁸Adkins, C. N., and Liebeck, R. H., "Design of Optimum Propellers," *Journal of Propulsion and Power*, Vol. 10, No. 5, pp. 676-682, 1994.

A. Transformation Matrix for Rotationally Periodic Boundary Conditions

The rotation matrix given by

$$\mathcal{T} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & \cos \theta_y \cos \theta_z & \sin \theta_x \sin \theta_y \cos \theta_z - \cos \theta_x \sin \theta_z & \cos \theta_x \sin \theta_y \cos \theta_z + \sin \theta_x \sin \theta_z & 0 \\ 0 & \cos \theta_y \sin \theta_z & \sin \theta_x \sin \theta_y \sin \theta_z + \cos \theta_x \cos \theta_z & \cos \theta_x \sin \theta_y \sin \theta_z - \sin \theta_x \cos \theta_z & 0 \\ 0 & -\sin \theta_y & \sin \theta_x \cos \theta_y & \cos \theta_x \cos \theta_y & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (36)$$

performs a periodic transformation when multiplied by the conservative variable vector, U , at one periodic face into the correct state to be applied at the matching periodic face. Note that this matrix is formed by assuming positive, right-handed rotation first about the x -axis (θ_x), then the y -axis (θ_y), and finally the z -axis (θ_z). It is then extended with ones in the first and last positions on the diagonal, because point quantities such as density and energy do not require rotation in space. The same transformation matrix is used for periodic boundary conditions with the adjoint problem.

B. Linearized Solid Wall Boundary Condition

The linearized flow equations and boundary conditions are introduced during the continuous adjoint derivation in order to remove any dependence on flow variations in the expression for the variation of the objective function. The details for linearizing the solid wall boundary condition for the rotating Euler equations are given here. We start with the flow tangency boundary condition:

$$(\vec{v} - \vec{u}_r) \cdot \vec{n}_S = 0 \text{ on } S, \quad (37)$$

where \vec{v} is the absolute flow velocity, \vec{u}_r is the velocity due to rotation at the point on the wall, and \vec{n}_S is the local wall unit normal. Consider linearization with respect to small perturbations in the surface, δS , for the both velocity and the normal terms separately,

$$(\vec{v} - \vec{u}_r)' = (\vec{v} - \vec{u}_r) + \delta(\vec{v} - \vec{u}_r) + \partial_n(\vec{v} - \vec{u}_r)\delta S, \quad (38)$$

$$(\vec{n}_S)' = \vec{n}_S + \delta\vec{n}_S, \quad (39)$$

where the second term on the right hand side of Eq. 38 represents the change in the flow solution caused by the deformation and the third term represents the change due solely to the geometry of the deformation. The normal of Eq. 39 does not involve the flow equations, so the only change is due to the deformation. The complete linearized boundary condition can be obtained by taking the dot product of the two linearized components,

$$\begin{aligned} (\vec{v} - \vec{u}_r)' \cdot (\vec{n}_S)' &= \{(\vec{v} - \vec{u}_r) + \delta\vec{v} - \delta\vec{u}_r + \partial_n(\vec{v} - \vec{u}_r)\delta S\} \cdot (\vec{n}_S + \delta\vec{n}_S) \\ &= (\vec{v} - \vec{u}_r) \cdot \delta\vec{n}_S + \delta\vec{v} \cdot \vec{n}_S + \partial_n(\vec{v} - \vec{u}_r)\delta S \cdot \vec{n}_S, \end{aligned} \quad (40)$$

where in order to simplify we have used the original boundary condition (Eq. 37), $\delta\vec{u}_r = 0$, and the approximation that any products of variations are negligible. Keeping in mind that the linearized boundary condition still must equal zero, Eq. 40 can be rearranged to give a useful result for the continuous adjoint derivation:

$$\delta\vec{v} \cdot \vec{n}_S = -(\vec{v} - \vec{u}_r) \cdot \delta\vec{n}_S - \partial_n(\vec{v} - \vec{u}_r) \cdot \vec{n}_S \delta S, \quad (41)$$

C. Source Term Jacobian

The Jacobian of the source term appearing during the linearization of the rotating Euler equations is given here in both two and three dimensions. It is assumed that the reference frame is under a steady rotation, $\vec{\omega} = \{\omega_x, \omega_y, \omega_z\}^T$, where the three components are in the cartesian x -, y -, and z -directions. In two dimensions, it will be assumed that the fluid exists in the x - y plane with an angular velocity in the z -direction. The Jacobian in two dimensions is then

$$\frac{\partial \mathcal{Q}}{\partial U} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & -\omega_z & 0 \\ 0 & \omega_z & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}. \quad (42)$$

In three-dimensions, the steady rotation of the frame along an arbitrary axis results in the following

Jacobian:

$$\frac{\partial \mathcal{Q}}{\partial U} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\omega_z & \omega_y & 0 \\ 0 & \omega_z & 0 & -\omega_x & 0 \\ 0 & -\omega_y & \omega_x & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}. \quad (43)$$

Note that the transpose of this matrix generally appears during the solution procedure.