AIAA SciTech Forum
23-27 January 2023, National Harbor, MD & Online
AIAA SCITECH 2023 Forum

10.2514/6.2023-0531

High-fidelity Aerodynamic and Aerostructural Optimization of UAV Propellers Using the Adjoint Method

Ping He*, Heyecan U. Koyuncuoglu†
Iowa State University, Ames, Iowa, 50011, USA

Helen Hu‡
Johns Hopkins University, Baltimore, MD, 21218, USA

Anvesh Dhulipalla§, Haiyang Hu¶, Hui Hu‖
Iowa State University, Ames, Iowa, 50011, USA

Downloaded by Iowa State University on October 26, 2023 | http://arc.aiaa.org | DOI: 10.2514/6.2023-0531

Unmanned aerial vehicles (UAVs) have seen widespread usage in commercial and military applications because they can significantly reduce the cost and human risk in a mission. The propeller is a crucial component for UAVs and its aerodynamic efficiency impacts the vehicle's overall performance. Aerodynamic optimization is a powerful technique that can use computer simulations to find the best possible design for UAV propellers automatically. However, existing propeller aerodynamic optimization commonly uses low-fidelity models, e.g., blade element momentum, and it is not clear to what extent high-fidelity optimization can benefit the UAV propeller design. In this paper, we develop the capability to conduct high-fidelity aerodynamic and aerostructural optimization for UAV propellers. We use finite-volume computational fluid dynamics and finite-element structural dynamics solvers to simulate the fluid and solid domains, respectively. We then use the discrete adjoint approach to compute the derivatives, which allows us to conduct gradient-based optimization with a large number of design variables. To consider fluid-structure interaction and its derivative computation, we utilize OpenMDAO/MPhys, an open-source framework that facilitates high-fidelity multidisciplinary design optimization. We conduct aerodynamic and aerostructural optimizations with shape and planform variables (e.g., span and chord). The objective function is the propeller shaft power, and the constraints include propeller thrust, mass, von-mises stress, and propeller geometry (e.g., thickness and curvature). Compared with the baseline design, the shape-only aerodynamic, shape+planform aerodynamic, and shape+planform aerostructural optimizations exhibit 7.4%, 12.8%, and 11.8% power reduction, respectively, and all the constraints are satisfied. This study has the potential to significantly reduce the time period for designing high-performance UAV propellers.

I. Introduction

Unmanned aerial vehicles (UAVs) have been widely used for commercial and military applications, such as delivery, surveillance, reconnaissance, and scientific research because they can significantly reduce the cost and human risk in a mission [1]. UAVs are typically powered by propellers, and the propeller aerodynamic performance directly impacts the UAV's power consumption and flight range. Multidisciplinary design optimization (MDO) [2] is a promising technique for maximizing the UAV propeller performance. It uses multiphysics computer simulations to find the best possible design automatically and has the potential to significantly reduce the design period. To allow large design freedom, we need a large number of design variables to fully describe the complex UAV propeller design surface geometry. Therefore, a gradient-based optimization algorithm coupled with the adjoint gradient computation approach [3–6] forms a powerful combination. The gradient-based optimization approach has been widely used for optimizing various aerospace engineering problems, including aircraft [7–9], wind turbine [10, 11], and gas turbines [12–14]. It has also

*Assistant Professor, Department of Aerospace Engineering, AIAA Senior Member. Email: phe@iastate.edu
†PhD Student, Department of Aerospace Engineering, AIAA Student Member.
‡Undergraduate Student, Department of Mechanical Engineering.
§PhD Student, Department of Aerospace Engineering.
¶Postdoc Research Associate, Department of Aerospace Engineering.
‖Martin C. Jischke Professor, Department of Aerospace Engineering, AIAA Associate Fellow.

Copyright © 2023 by the authors. Published by the American Institute of Aeronautics and Astronautics, Inc., with permission.

been used in propeller designs; however, existing gradient-based propeller designs typically replied on low-fidelity models [15, 16], e.g., blade element momentum (BEM). It is not clear to what extent high-fidelity MDO can benefit the UAV propeller designs.

In this study, we demonstrate a high-fidelity MDO framework for optimizing UAV propeller performance based on the OpenMDAO/MPhys framework. OpenMDAO [17] is an open-source multidisciplinary design, analysis, and optimization framework developed by NASA. OpenMDAO has been widely used for large-scale design optimization in aerospace engineering applications. MPhys [18] is a recently developed derivative of OpenMDAO. MPhys facilitates the coupling of high-fidelity solvers with various disciplines. We have recently coupled our open-source discrete adjoint solver called DAFoam [19, 20] into the OpenMDAO/MPhys framework. We will demonstrate high-fidelity aerodynamic and aerostructural optimizations for UAV propeller blades. The aerodynamic and structural analysis will be conducted by using computational fluid dynamics and finite-element solvers.

The rest of this paper is organized as follows. In Section II, we elaborate on the aerodynamic and aerostructural analysis and optimization framework, followed by the detailed aerodynamic and optimization results in Section III. Finally in Section IV conclusions are drawn and perspectives and future improvements are provided.

Downloaded by Iowa State University on October 26, 2023 | http://arc.aiaa.org | DOI: 10.2514/6.2023-0531

II. Method

In this section, we elaborate on the mathematical background of our aerodynamic and aerostructural analysis and optimization framework.

A. Aerodynamic Analysis using CFD

We use OpenFOAM's rhoSimpleFoam solver [21] to simulate the flow over UAV propeller blades. It solves 3D, steady-state turbulent flows governed by the compressible Navier-Stokes (NS) equations written in the multiple-reference-frame (MRF) format.

$$\nabla \cdot (\rho \boldsymbol{U}_a) = 0, \tag{1}$$

$$\nabla \cdot (\rho \boldsymbol{U}_r \boldsymbol{U}_a) + \boldsymbol{\omega} \times \boldsymbol{U}_a + \nabla p - \nabla \cdot \mu_{\text{eff}} (\nabla \boldsymbol{U}_a + \nabla \boldsymbol{U}_a^T) = 0, \tag{2}$$

$$\nabla \cdot (\rho e \boldsymbol{U}_a) + \nabla \cdot (0.5 \rho |U|_a^2 \boldsymbol{U}_a + p \boldsymbol{U}_a) - \alpha_{\text{eff}} \nabla \cdot (\nabla e) = 0, \tag{3}$$

where $\boldsymbol{U}_a$ and $\boldsymbol{U}_r$ are the absolute and relative velocities, respectively, and they are related through $\boldsymbol{U}_a = \boldsymbol{U}_r + \boldsymbol{\omega} \times \boldsymbol{x_c}$ with $\boldsymbol{\omega}$ being the rotational speed vector and $\boldsymbol{x_c}$ being the cell-center coordinate vector. The above governing equations are solved using the compressible form of the SIMPLE algorithm based on the absolute velocity in the stationary frame; however, the flux for the convective term in the momentum equation (2) is computed using the relative velocity in the rotating frame.

The Spalart-Allmaras model is used to connect the turbulent eddy viscosity to the mean flow variables, the details of this formulation and description of the terms can be found in their paper [22]:

$$\nabla \cdot (\boldsymbol{U}_a \widetilde{\nu}) + \frac{1}{\sigma} \{ \nabla \cdot [(\nu + \widetilde{\nu}) \nabla \widetilde{\nu}] + C_{b1} |\nabla \widetilde{\nu}|^2 \} - C_{b1} \widetilde{S} \widetilde{\nu} + C_{w1} f_w \left(\frac{\widetilde{\nu}}{d} \right)^2 = 0 \tag{4}$$

B. Structural Analysis using FEM

We use TACS [23], an open-source finite-element solver, to simulate the propeller blade structure. TACS has linear and geometrically nonlinear simulation capabilities; however, only linear analysis is used in this paper. The linear elastic governing equation is as follows:

$$Ku - F = 0 \tag{5}$$

where K is the structural stiffness matrix, u is the structural displacement vector, and F is the load vector.

C. Aerostructural Analysis with Coupled CFD and FEM

The aerostructural coupling is implemented in the OpenMDAO/MPhys framework. As mentioned above, OpenMDAO [17] is an open-source multidisciplinary design, analysis, and optimization framework developed by NASA. MPhys [18] is a recently developed derivative of OpenMDAO with a flexible interface that allows the multidisciplinary

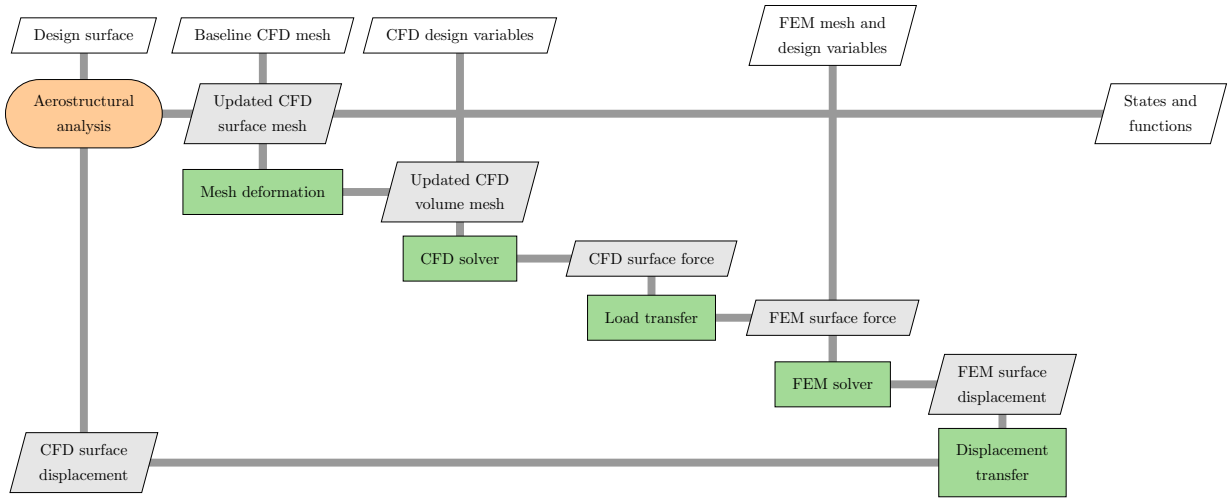

Fig. 1 Schematic of aerostructural analysis using OpenMDAO/MPhys [18]. Here we use the XDSM representation. The diagonal blocks are the components and the off-diagonal blocks are data transfer. The CFD and FEM solvers are OpenFOAM and TACS, respectively.

coupling of various high-fidelity solvers, such as finite-volume computational fluid dynamics and finite-element computational structural dynamics.

Figure 1 shows the aerostructural analysis using the OpenFOAM and TACS solvers. The inputs are the latest design surface geometry (i.e., propeller blade geometry) in the optimization loop, baseline CFD and FEM meshes, and CFD and FEM design variables (e.g., blade rotation speed). The outputs are the converged aerostructural state variables (e.g., velocity, pressure, and structural displacement) and objective and constraint functions (e.g., propeller power and thrust). The "Aerostructural analysis" (main driver) starts by computing the "Updated CFD surface mesh" based on the "Design surface" geometry and the "CFD surface mesh displacement" computed from the "Displacement transfer" component (in the first iteration, the displacement will be zero). Then, the "Mesh deformation" component will use the "Updated CFD surface mesh" to compute the "Updated CFD volume mesh". Next, the "CFD solver" component receives the "Updated CFD volume mesh", simulates the flow fields, extracts the "CFD surface force", and passes it to the "Load transfer" component. The "Load transfer" component will interpolate the "CFD surface force" to the "FEM surface force" and pass the force to the "FEM solver". Then, the "FEM solver" component will compute the structural displacement and extract the "FEM surface displacement" and pass it to the "Displacement transfer" component. The "Displacement transfer" component will then interpolate the "FEM surface displacement" to the "CFD surface displacement", and then pass it to the "Aerostructural analysis" main driver to start the next iteration. The above process will repeat until the residuals for both the CFD and FEM solvers are small than the prescribed tolerances.

MPhys provides a general aerostructural template called "Scenario" to facilitate the above coupling between CFD and FEM solvers. MPhys currently supports solvers such as ADflow [24], FUN3D [25, 26], OpenAeroStruct [27], and TACS. We have developed a Python interface called mphys_dafoam to extend the aerostructural scenario so it can use OpenFOAM and DAFoam as the CFD and adjoint solvers (the adjoint will be elaborated on in the next section). The mesh deformation component uses IDWarp [28], an open-source inverse distance weighted mesh deformation code. The load and displacement transfer components are based on the Meld approach from a generic aeroelastic analysis and adjoint-based gradient evaluation tool called FUNtoFEM [29]. TACS is used as the FEM solver. The CFD and FEM iteration is conducted using the nonlinear block Gauss–Seidel solver in OpenMDAO with an Aitken relaxation approach.

D. Discrete Adjoint Computation

We use the discrete adjoint method to compute the derivatives for optimization. The function of interest is the function of both the design and the state variables:

Downloaded by Iowa State University on October 26, 2023 | http://arc.aiaa.org | DOI: 10.2514/6.2023-0531

$$f = f(\boldsymbol{x}, \boldsymbol{w}) \tag{6}$$

where $\boldsymbol{x} \in \mathbb{R}^{n_x}$ is the design vector, and $\boldsymbol{w}^n \in \mathbb{R}^{n_w}$ is the flow state variable vector. n_x and n_w are the numbers of design and state variables, respectively. The chain rule is applied to compute $\mathrm{d}f/\mathrm{d}\boldsymbol{x}$:

$$\left[\frac{\mathrm{d}f}{\mathrm{d}\boldsymbol{x}}\right]_{1\times n_x} = \left[\frac{\partial f}{\partial \boldsymbol{x}}\right]_{1\times n_x} + \left[\frac{\partial f}{\partial \boldsymbol{w}}\right]_{1\times n_w} \left[\frac{\mathrm{d}\boldsymbol{w}}{\mathrm{d}\boldsymbol{x}}\right]_{n_w\times n_x} \tag{7}$$

The partial derivatives in Eq. (7) are relatively cheap to evaluate because they only involve explicit computations. However, the total derivative $\mathrm{d}\boldsymbol{w}/\mathrm{d}\boldsymbol{x}$ is expensive because both terms are determined implicitly.
Similarly, the chain rule is applied to the flow residual vector, $\boldsymbol{R} \in \mathbb{R}^{n_w}$, to solve the $\mathrm{d}\boldsymbol{w}/\mathrm{d}\boldsymbol{x}$ term. Because the governing equations have to be satisfied regardless of the values of design variables x, the total derivative $\mathrm{d}\boldsymbol{R}/\mathrm{d}\boldsymbol{x}$ must be zero:

$$\frac{\mathrm{d}\boldsymbol{R}}{\mathrm{d}\boldsymbol{x}} = \frac{\partial \boldsymbol{R}}{\partial \boldsymbol{x}} + \frac{\partial \boldsymbol{R}}{\partial \boldsymbol{w}} \frac{\mathrm{d}\boldsymbol{w}}{\mathrm{d}\boldsymbol{x}} = 0 \tag{8}$$

Substituting Eq. (8) into Eq. (7):

$$\left[\frac{\mathrm{d}f}{\mathrm{d}\boldsymbol{x}}\right]_{1\times n_x} = \left[\frac{\partial f}{\partial \boldsymbol{x}}\right]_{1\times n_x} - \left[\frac{\partial f}{\partial \boldsymbol{w}}\right]_{1\times n_w} \left[\frac{\partial \boldsymbol{R}}{\partial \boldsymbol{w}}\right]^{-1}_{n_w\times n_w} \left[\frac{\mathrm{d}\boldsymbol{R}}{\mathrm{d}\boldsymbol{x}}\right]_{n_w\times n_x} = \left[\frac{\partial f}{\partial \boldsymbol{x}}\right]_{1\times n_x} - \left[\boldsymbol{\psi}^T\right]_{1\times n_w} \left[\frac{\mathrm{d}\boldsymbol{R}}{\mathrm{d}\boldsymbol{x}}\right]_{n_w\times n_x} \tag{9}$$

where $\boldsymbol{\psi}$ is the adjoint vector. Transposing the Jacobian and solving with $[\mathrm{d}f/\mathrm{d}\boldsymbol{w}]^T$ as the right-hand side yields the adjoint equations:

$$\left[\frac{\partial \boldsymbol{R}}{\partial \boldsymbol{w}}\right]^T_{n_w\times n_w} \cdot [\boldsymbol{\psi}]_{n_w\times 1} = \left[\frac{\partial f}{\partial \boldsymbol{w}}\right]^T_{n_w\times 1} \tag{10}$$

Substituting Eq. (10) into Eq. (7), we can compute the total derivatives:

$$\frac{\mathrm{d}f}{\mathrm{d}\boldsymbol{x}} = \frac{\partial f}{\partial \boldsymbol{x}} - \boldsymbol{\psi}^T \frac{\partial \boldsymbol{R}}{\partial \boldsymbol{x}} \tag{11}$$

Since the design variable is not explicitly present in Eq. (11), the adjoint equations are needed to solve only once for each function of interest. Therefore, its computational cost is independent of the number of design variables but proportional to the number of objective functions. This approach is also known as the adjoint method, and it is advantageous for aerodynamic design. Because typically, there are only a few functions of interest, but several hundred design variables can be used.

E. Aerostructural Adjoint Coupling

The above adjoint formulation assumes the CFD and FEM state variables are combined into one and solved in a coupled manner. However, this will increase the size of the Jacobian matrix and the corresponding memory cost. We use a segregated method (block Gauss-Seidel) to solve the coupled aerostructural adjoint, as shown below.

$$\begin{bmatrix} \dfrac{\partial \boldsymbol{R}_{\mathrm{CFD}}}{\partial \boldsymbol{w}_{\mathrm{CFD}}}^T & \dfrac{\partial \boldsymbol{R}_{\mathrm{FEM}}}{\partial \boldsymbol{w}_{\mathrm{CFD}}}^T \\ \dfrac{\partial \boldsymbol{R}_{\mathrm{CFD}}}{\partial \boldsymbol{w}_{\mathrm{FEM}}}^T & \dfrac{\partial \boldsymbol{R}_{\mathrm{FEM}}}{\partial \boldsymbol{w}_{\mathrm{FEM}}}^T \end{bmatrix} \begin{bmatrix} \boldsymbol{\Psi}_{\mathrm{CFD}} \\ \boldsymbol{\Psi}_{\mathrm{FEM}} \end{bmatrix} = \begin{bmatrix} \dfrac{\partial f}{\partial \boldsymbol{w}_{\mathrm{CFD}}}^T \\ \dfrac{\partial f}{\partial \boldsymbol{w}_{\mathrm{FEM}}}^T \end{bmatrix} \tag{12}$$

where the subscript CFD and FEM denote the residual and state variables for the CFD and FEM solvers, respectively. We use DAFoam [19, 20, 30] to solve the CFD adjoint equation:

$$\frac{\partial \boldsymbol{R}_{\mathrm{CFD}}}{\partial \boldsymbol{w}_{\mathrm{CFD}}}^T \boldsymbol{\Psi}_{\mathrm{CFD}} = \frac{\partial f}{\partial \boldsymbol{w}_{\mathrm{CFD}}}^T \tag{13}$$

DAFoam is an open-source discrete adjoint implementation for OpenFOAM. DAFoam uses a Jacobian-free adjoint approach, in which the partial derivatives and matrix-vector products are computed using the automatic differentiation method, as detailed in Kenway et al. [31]. DAFoam uses the generalized minimal residual (GMRES), an iterative linear equation solver in the PETSc [32] library to solve the adjoint equation. A nested preconditioning strategy with the

Downloaded by Iowa State University on October 26, 2023 | http://arc.aiaa.org | DOI: 10.2514/6.2023-0531

additive Schwartz method is used as the global preconditioner and the incomplete lower and upper (ILU) factorization approach with one level of fill-in for the local preconditioning. The preconditioner matrix $[\partial \boldsymbol{R}/\partial \boldsymbol{w}]^T_{RC}$ is constructed by approximating the residuals and their linearizations [19] to improve convergence. The construction of $[\partial \boldsymbol{R}/\partial \boldsymbol{w}]^T_{RC}$ is only done for the first time instance and then is reused for the adjoint equation. This treatment significantly reduces the adjoint runtime because the constructing $[\partial \boldsymbol{R}/\partial \boldsymbol{w}]^T_{RC}$ consists of about 30% of the adjoint runtime.
We use TACS to solve the adjoint equation for the FEM part:

$$\frac{\partial \boldsymbol{R}_{\text{FEM}}}{\partial \boldsymbol{w}_{\text{FEM}}}^T \boldsymbol{\Psi}_{\text{FEM}} = \frac{\partial f}{\partial \boldsymbol{w}_{\text{FEM}}}^T, \tag{14}$$

The matrix-vector product for the off-diagonal components is computed by using a matrix-free manner through automatic differentiation.
Traditionally, the above aerostructural coupling is implemented in a solver-specific manner and is hard to extend. By leveraging the flexibility of OpenMDAO and MPhys, the aerostructural is implemented in a modular manner. In MPhys, we need to implement the methods to compute the output based on the input for the components shown in Fig. 1, as well as the product of the state Jacobian matrix with a given vector. OpenMADO will then use the MAUD algorithm [33] to unify the adjoint total derivative computation. The CFD and FEM coupled adjoint is solved using the linear block Gauss–Seidel solver with Aitken relaxation in OpenMDAO.

F. Aerodynamic and Aerostructural Optimization Framework

Figure 2 shows the schematic of the aerodynamic optimization framework under the OpenMDAO/MPhys [18] framework. In the "Preprocessing" step, we generate the "CFD mesh" for the "Baseline design surface" geometry. We also generate the free-form deformation (FFD) control points for geometry parameterization. Then, the "Optimizer" will pass the initial "Geometric design variables" (FFD points displacement) to the "Geometry parameterization" module pyGeo. pyGeo [34] is an open-source FFD tool to parameterize the design surface geometry. pyGeo can embed a set of point clouds into the prescribed FFD box and can then deform the point cloud by moving the FFD coordinates. pyGeo works for structured and unstructured meshes. The FFD box generated in the preprocessing step will fully contain the CFD design surface and surface mesh. Next, the "Updated design surface" is passed to the "Mesh deformation" module IDWarp. IDWarp is an open-source mesh deformation tool that uses an inverse-distance weighting approach [28] to deform high-quality volume mesh based on the new design surface. We use IDWarp to deform the volume mesh because it is smooth and can avoid numerical noise compared with re-generating the volume mesh. IDWarp will pass the "Updated volume mesh" to the "CFD solver" (OpenFOAM). OpenFOAM will then solve the flow and send the "Converged CFD state variables" to the "Adjoint solver" component. OpenFOAM also computes the aerodynamic "Objective and constraint functions" (e.g., power, thrust). Finally, the "Adjoint solver" computes the total derivatives of the objective and constraint functions and passes them back to the "Optimizer". The "Optimizer" will use the function values and derivatives to update the design variables for the next optimization iteration. The above process will repeat until the optimization converges.
Figure 3 shows the schematic of the aerostructural optimization framework using OpenMDAO/MPhys [18]. The aerostructural optimization framework is similar to the aerodynamic-only one, except that: (1). The FFD box will fully contain the CFD design surface and the entire FEM mesh. Therefore, pyGeo will deform the CFD surface mesh and the FEM mesh to ensure a consistent fluid-structure interface. (2). The mesh deformation module is embedded in the aerostructural analysis module (Fig. 1). (3). Instead of single disciplinary analysis, the aerostructural optimization performs coupled aerostructural analysis and adjoint, each requiring solving the CFD and FEM systems using block Gauss–Seidel methods. (4). The objective and constraint functions passed back to the optimizer contain both aerodynamic and structural variables, e.g., power, thrust, aggregated von-mises stress, and structural mass.

III. Results and Discussion

In this section, we show the aerodynamic and aerostructural optimization results. We conduct three optimization cases: (1) aerodynamic optimization with only the shape variable, (2) aerodynamic optimization with the shape and planform variables, and (3) aerostructural optimization.

Downloaded by Iowa State University on October 26, 2023 | http://arc.aiaa.org | DOI: 10.2514/6.2023-0531

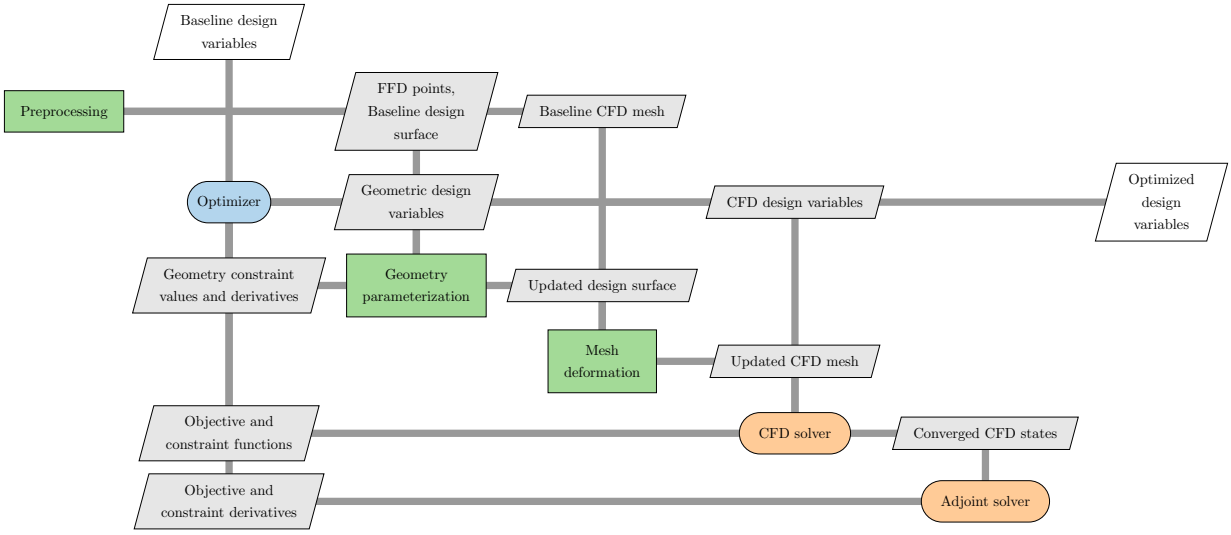

Fig. 2 Schematic of aerodynamic optimization using OpenMDAO/MPhys [18]. Here we use the XDSM representation. The diagonal blocks are the components and the off-diagonal blocks are data transfer. The CFD and adjoint solvers are OpenFOAM and DAFoam, respectively.

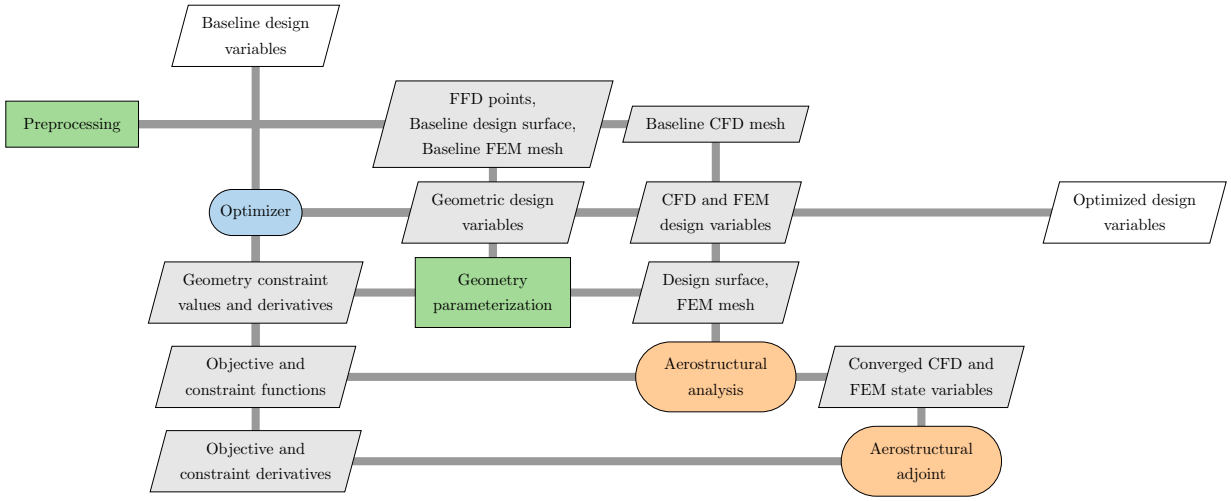

Fig. 3 Schematic of aerostructural optimization using OpenMDAO/MPhys [18]. Here we use the XDSM representation. The diagonal blocks are the components and the off-diagonal blocks are data transfer. The aerostructural analysis component is detailed in Fig. 1, and the aerostructural coupled adjoint is elaborated on in Sec. II.E.

A. Model Setup and Validation

We use a UAV propeller from our previous studies [35, 36] as the benchmark for aerostructural analysis and optimization. Figure 4 shows the CFD and FEM meshes for the propeller blades. We use Pointwise to generate an unstructured mesh with 850,000 cells (Fig. 4 right and mid). For structural consideration, we mesh both the propeller blade and the spinner. In addition, we add a rounded fillet at the root of the blade to avoid stress concentration. The propeller has two blades; however, we simulate only one of them with periodic boundary faces. For structural simulations, we generate an unstructured triangulated mesh with 239,000 cells (Fig. 4 right) using ICEM CFD. We use the CTETRA linear solid

Downloaded by Iowa State University on October 26, 2023 | http://arc.aiaa.org | DOI: 10.2514/6.2023-0531

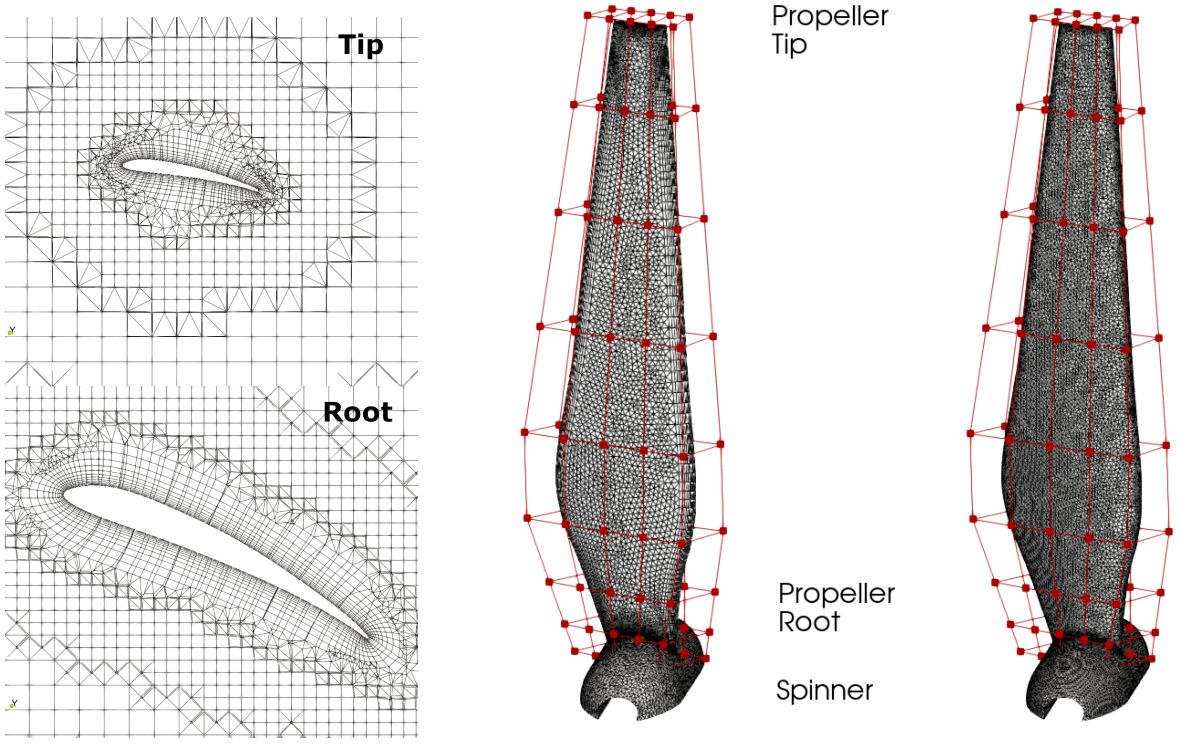

Fig. 4 Propeller meshes and FFD points. Left: CFD meshes at the root and tip sections. Mid: CFD surface mesh for the propeller blade. Right: FEM mesh for the propeller blade. The CFD and FEM meshes are generated using Pointwise and ICEM-CFD, respectively. The red squares are the FFD points.

element type. We plan to 3D print the designs for experimental validation and use the VeroWhite as the blade material, and its properties are density 1170 kg/m^3, elastic modulus 2.5×10^9 Pa, Poisson's ratio 0.3, and yield stress 7×10^7 Pa. We impose zero displacement and rotation conditions on the inner and periodic faces of the spinner.

The far-field velocity is 5 m/s and the propeller is rotating at 4000 RPM. The propeller has a radius of 0.12 m and the tip speed is 52.8 m/s. The Reynolds number, which is based on the chord and blade rotation speed at 75% span, is 44,000. As mentioned above, we use OpenFOAM's rhoSimpleFoam solver with the MRF approach to simulate the steady-state turbulent flow with rotation. The turbulence model is Spalart–Allmaras. The simulation domain is a cylinder with a radius of 0.25 m, which mimics a wind tunnel flow setting. We select mesh cells that are close to the propeller as the rotation domain, while the rest of the cells in the simulation domain is stationary ($\omega = 0$). We find that this setting improves the convergence of the flow simulations.

We measure the aerodynamic performance of the baseline propeller design in the Iowa State University low-speed wind tunnel (Fig. 5). We 3D print the baseline propeller design and install it in the close-looped wind tunnel. The propeller is driven by a brushless motor (920Kv) powered by a DC power supply (VOLTEQ HY3050EX). The rotational speed of the propeller is measured by a tachometer (Monarch PLT200), which generates a pulse signal for each propeller's rotation. To mitigate the disturbance (changes in RPM), an automatic rotational speed correction is formulated into the system using a proportional-integral-derivative algorithm (PID). A force measurement (ATI-IA Mini 45) analysis has been performed on the 3D-printed baseline design. The measured thrust at 4000 RPM is 1.18 N. The simulated thrusts with aerodynamic-only (rigid propeller) and aerostructural settings are 1.37 and 1.41 N, respectively. The simulations overpredict the thrust by 16% to 19%. We suspect this may be caused by the idealized working conditions in CFD simulations, e.g., the nacelle was not included in CFD and we assume the electric motor has an efficiency of 100%. We will report a more detailed comparison between the CFD simulations and experimental measurement for the baseline and optimized designs in future work.

Downloaded by Iowa State University on October 26, 2023 | http://arc.aiaa.org | DOI: 10.2514/6.2023-0531

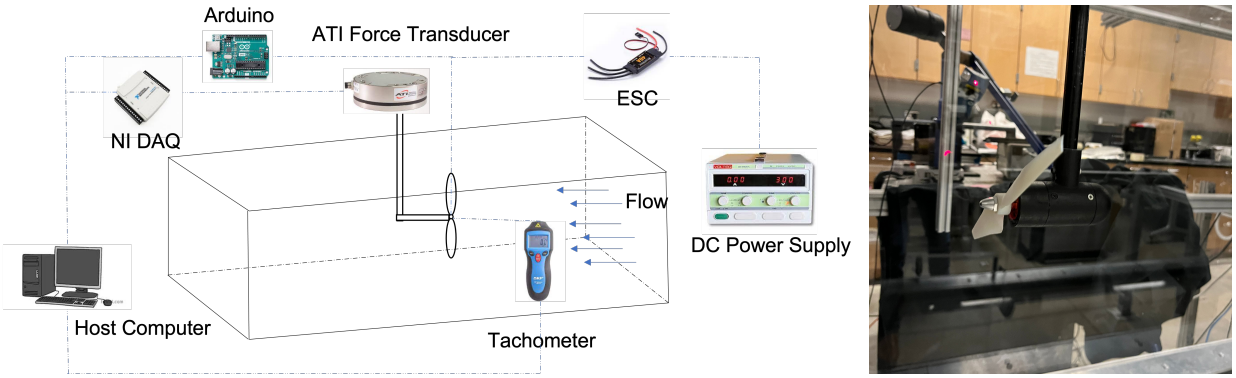

Fig. 5 Schematic of the Iowa State low-speed wind tunnel and experimental setup for the propeller blades.

Table 1 Aerodynamic optimization problem configuration. We use 60 and 74 design variables for the (a) shape-only and (b) shape and planform optimizations, respectively. We use 246 constraints. The subscript bl denotes the baseline design.

	Function/Variable	Description	Quantity
Minimize	P	Propeller shaft power	
with respect to	$-0.01\text{ m} \le \Delta x \le 0.01\text{ m}$	Propeller shape change	60
	$-10^\circ \le \gamma \le 10^\circ$	Propeller twist for each FFD section	6
	$-20\% \le \Delta s \le 20\%$	Propeller span change	1
	$-20\% \le \Delta c \le 20\%$	Propeller chord change for each FFD section	6
	$-20\% \le \Delta\omega \le 20\%$	Propeller rotation speed change	1
		Total Design Variables	**60**^{a}, **74**^{b}
subject to	$V_{\text{bl}} \le V \le 3 \cdot V_{\text{bl}}$	Minimum volume constraint	1
	$0.8 \cdot t_{\text{bl}} \le t \le 3 \cdot t_{\text{bl}}$	Minimum thickness constraint	240
	$C \le 1.5 C_{\text{bl}}$	Blade spanwise curvature constraint	2
	$T = T_{\text{target}}$	Thrust constraint	1
	$Q_{\text{non-ortho}} \le 78^\circ$	Mesh quality constraint (non-orthogonality)	1
	$Q_{\text{skewness}} \le 5.8$	Mesh quality constraint (skewness)	1
		Total Constraint Functions	**246**

B. Aerodynamic Optimization

Table 2 summarizes the aerodynamic optimization configuration. The objective function is the propeller shaft power P. We set a body-fitted FFD box for the propeller blade, as shown in Fig. 4. To avoid poor mesh quality at the propeller-spinner intersection in the optimization, we fix the FFD points near the root and allow only the top 6 layers of FFD points to move in the axial (x) direction. In total, we have 60 shape variables for the blade. In addition to the local shape, we set global design variables that allow a set of FFD points to move simultaneously. To be more specific, we allow the FFD points to change the twist and chord of the blade at the top 6 FFD spanwise sections. We also allow the FFD points to scale in the radial (z) direction to change the span. The chord and span change are within 20% of their initial values. Finally, we allow the blade rotation speed to change ±20% with respect to its initial value during the optimization. We have 60 and 74 design variables for the shape-only and shape+planform cases.

To ensure a practical design, we impose various geometric and physical constraints. To be more specific, we impose a minimal blade volume constraint so the blade volume does not decrease and cannot exceed three times its baseline

Downloaded by Iowa State University on October 26, 2023 | http://arc.aiaa.org | DOI: 10.2514/6.2023-0531

Table 2 Summary of aerodynamic optimization cases. We obtain 7.4% and 12.8% power reduction for the shape-only and shape+planform cases, respectively. ω and s are the rotation speed and span, respectively. The subscript bl denotes the baseline design.

	Baseline Design	Optimized Design (Shape-Only)	Optimized Design (Shape+Planform)
P, W	12.23	11.32 (↓7.4%)	10.67 (↓12.8%)
T, N	1.400	1.400	1.400
ω, RPM	4042	4042	3233
s/s_{bl}	1.0	1.0	1.2

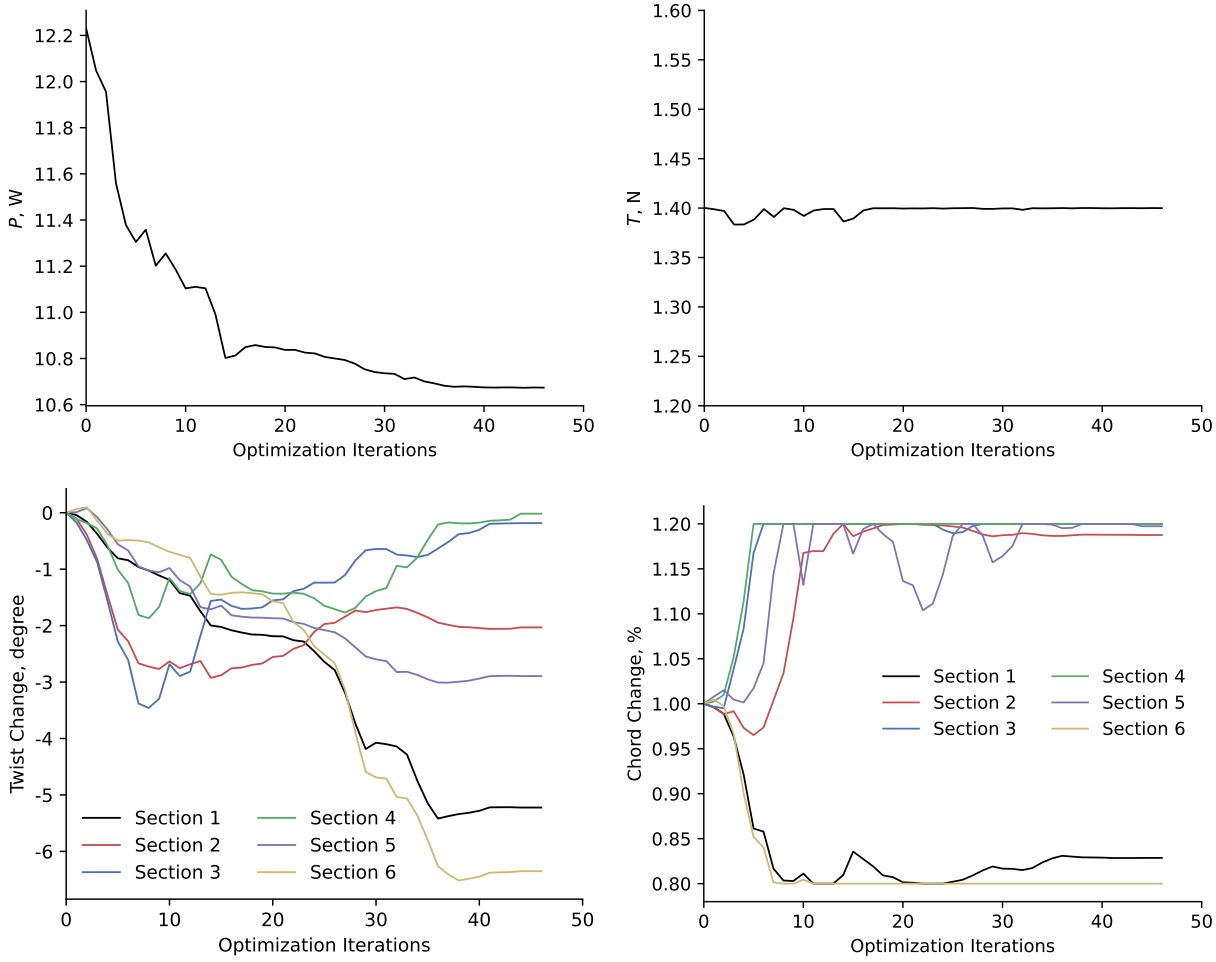

Fig. 6 Aerodynamic optimization convergence history for the shape+planform setup. We obtain 12.8% power reduction while the thrust constraint is satisfied.

volume. We also impose blade thickness constraints to prevent the blade thickness from being too thin. To this end, we enforce the thickness at 240 blade locations to be within 0.8 to 3 of its baseline thickness. We impose an equality constraint to fix the propeller thrust at 1.4 Newtons with the corresponding thrust coefficient of 0.078.

Our experience shows that the mesh deformation for the propeller blade is challenging and often results in poor mesh quality, especially for the first few optimization iterations. Therefore, we impose two mesh quality constraints for the volume mesh deformation. The maximal non-orthogonality and skewness are constrained to be less than 78° and 5.8,

Downloaded by Iowa State University on October 26, 2023 | http://arc.aiaa.org | DOI: 10.2514/6.2023-0531

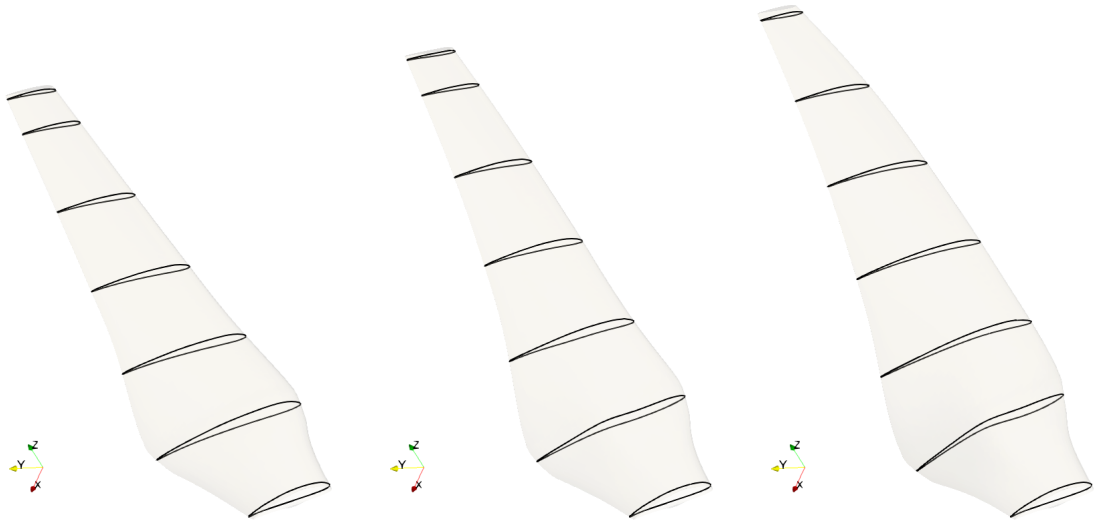

Fig. 7 Comparison of airfoil profiles among the baseline (left), optimized shape-only (mid), and shape+planform (right) designs at different spanwise sections.

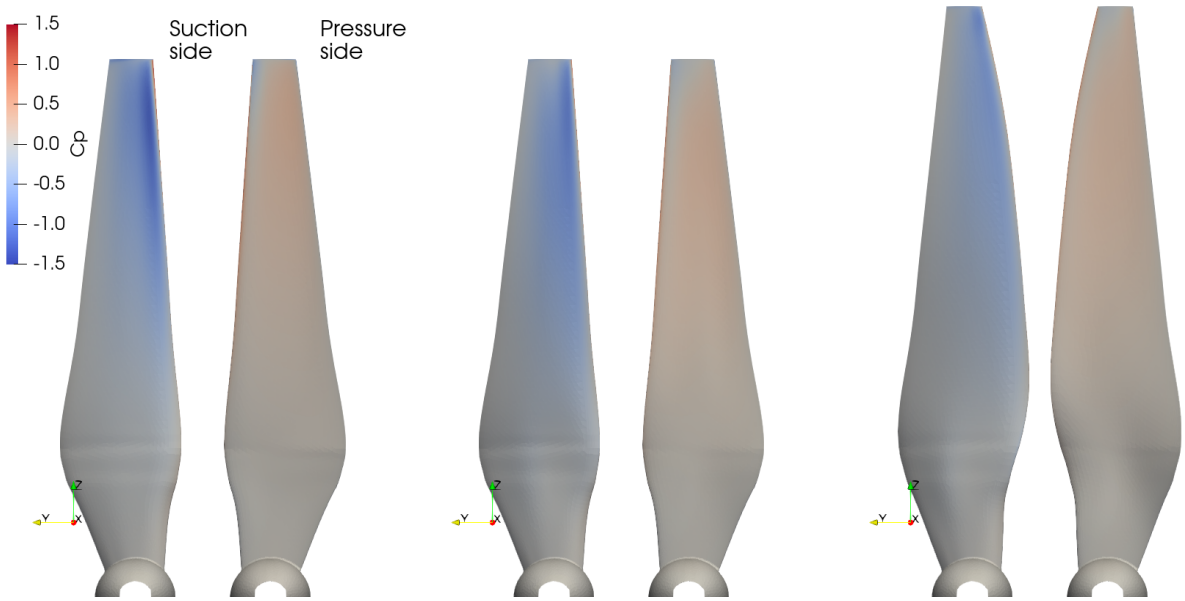

Fig. 8 Comparison of pressure contours among the baseline (left), optimized shape-only (mid), and shape+planform (right) cases.

respectively. The mesh quality values are computed using OpenFOAM's built-in checkMesh utility and DAFoam has a flexible interface to compute the constraints' derivatives with respect to the design variables through automatic differentiation. Moreover, we impose a spanwise curvature constraint to prevent a wavy shape in the spanwise, similar to what we used in the previous wing aerodynamic optimization study [37]. The spanwise curvature of the blade is constrained to no more than 1.5 times of the baseline curvature.

Table 2 summarizes the aerodynamic optimization results. We obtain 7.4% and 12.8% power reduction for the shape-only and shape+planform cases, respectively, and the thrust constraints are satisfied. We find that having the planform variables, especially allowing the rotation speed and span to change, helps improve the propeller performance by

Downloaded by Iowa State University on October 26, 2023 | http://arc.aiaa.org | DOI: 10.2514/6.2023-0531

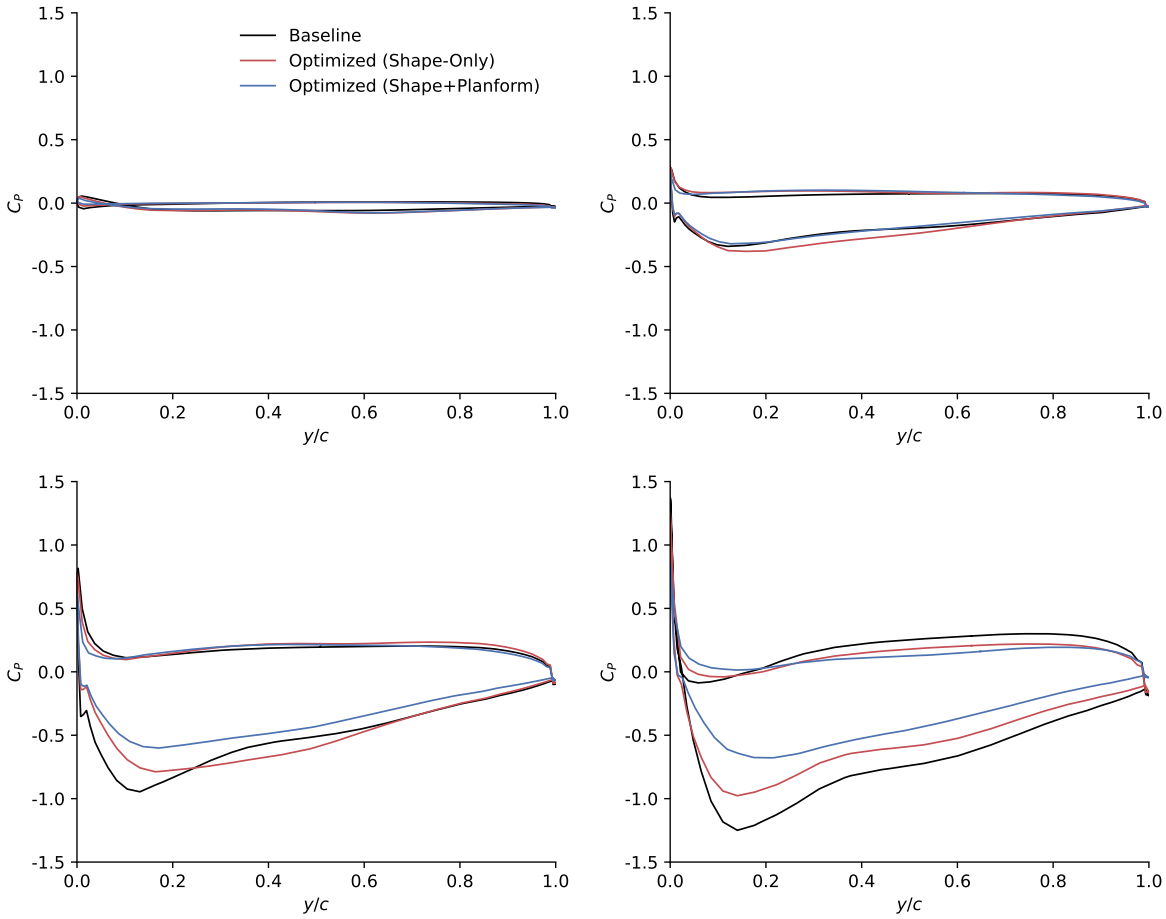

Fig. 9 Comparison of pressure profiles at 10% (top left), 40% (top right), 70% (bot left), 90% (bot right) spanwise locations.

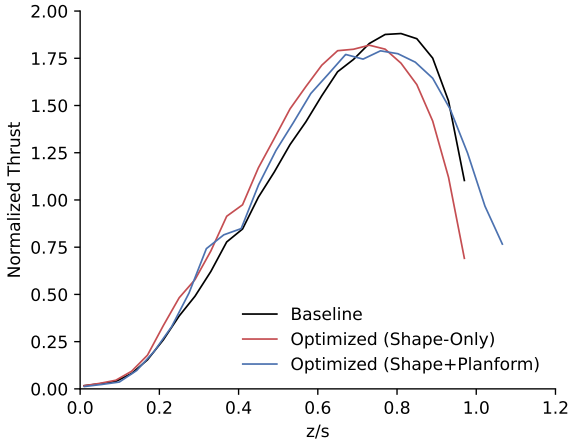

Fig. 10 Comparison of thrust distribution along the span among the baseline, optimized shape-only, and shape+planform designs.

Downloaded by Iowa State University on October 26, 2023 | http://arc.aiaa.org | DOI: 10.2514/6.2023-0531

Table 3 Aerostructural optimization problem configuration. We used 74 design variables and 245 constraints.

	Function/Variable	Description	Quantity
Minimize	P	Propeller shaft power	
with respect to	$-10^\circ \leq \gamma \leq 10^\circ$	Propeller twist for each FFD section	6
	-0.01 m $\leq \Delta x \leq 0.01$ m	Propeller shape change	60
	$-20\% \leq \Delta s \leq 20\%$	Propeller span change	1
	$-20\% \leq \Delta c \leq 20\%$	Propeller chord change for each FFD section	6
	$-20\% \leq \Delta\omega \leq 20\%$	Propeller rotation speed change	1
		Total Design Variables	**74**
subject to	$0.8 \cdot t_{\text{bl}} \leq t \leq 3 \cdot t_{\text{bl}}$	Minimum thickness constraint	240
	$C \leq 1.5 C_{\text{bl}}$	Blade spanwise curvature constraint	2
	$T = T_{\text{target}}$	Thrust constraint	1
	$m \leq m_{\text{bl}}$	Propeller mass constraint	1
	$\sigma/\sigma_{\text{yield}} \leq 1$	Max stress constraint	1
		Total Constraint Functions	**245**

Downloaded by Iowa State University on October 26, 2023 | http://arc.aiaa.org | DOI: 10.2514/6.2023-0531

additionally 5.4%. In the shape+planform case, the optimizer chooses to reduce the rotation speed and increase the span to reduce the propeller power.

Figure 6 shows the aerodynamic optimization convergence history for the shape+planform setup. The optimization runs for 49 iterations. Generally, the power decreases in a monotonic manner with oscillation between the 5th to 15th iterations. The oscillation is caused by the violation of thrust constraint (Fig. 6 top right). In addition, we observe that the chord decreases at the 1st and 6th spanwise FFD sections, while the rest of the chord variables increase. Moreover, the 1st and 6th spanwise FFD sections have the largest twists, compared with the other sections.

Figure 7 shows the comparison of airfoil profiles among the baseline, optimized shape-only, and shape+planform (right) designs at different spanwise sections. Compared with the baseline design, the shape-only optimized design slightly tilts the blade toward the leading edge, while the main blade planform remains unchanged. This can be seen more clearly in Fig. 8, where the comparison of pressure contours among the baseline, optimized shape-only, and shape+planform cases are plotted. For the shape+planform optimized design, the optimizer decreases the chord at the root and tip and increases the chord at other spanwise locations, consistent with what we observed in Fig. 6.

Figure 9 shows the comparison of pressure profiles at 10%, 40%, 70%, and 90% spanwise locations. We find that the blade loading is negligible at the root and increases with increasing the spanwise location. Also, we observe that the shape+planform optimized design decreases the blade loading at all spanwise locations. So, to maintain the desired thrust, the shape+planform design increases the span.

Figure 10 shows the comparison of thrust distribution along the span among the baseline, optimized shape-only, and shape+planform designs. Compared with the baseline design, the shape-only optimized design decreases the peak thrust and shifts it towards the root, while maintaining the same integrated thrust to satisfy the thrust constraint. The shape+planform optimization's peak thrust is also reduced compared with the baseline design, and it increases its span to satisfy the thrust constraint.

C. Aerostructural Optimization

Table 3 shows the summary of the aerostructural optimization problem configuration. The aerostructural optimization setup is similar to that used in the shape+planform aerodynamic optimization except that: (1) We do not impose a volume constraint for the blade from the CFD side, instead we impose a mass constraint to the FEM side such that the blade mass cannot increase during the optimization. (2) We do not impose mesh quality constraints because the extreme mesh deformation is automatically filtered out by the structural solver. (3) We impose an aggregated von-mises stress constraint to avoid structural failure. The aggregated stress constraint value is computed using the Kreisselmeier–Steinhauser (KS) function and is an approximated maximal stress value. The stress constraint is normalized by the material yield stress.

Table 4 Summary of aerostructural optimization cases. We obtain 11.8% power reduction, and the thrust, mass, and stress constraints are satisfied. ω and s are the rotation speed and span, respectively. The subscript bl denotes the baseline design.

	Baseline Design	Optimized Design
P, W	12.00	10.58 ($\downarrow$11.8%)
T, N	1.400	1.400
m/m_{bl}	1.0	0.97
$\sigma/\sigma_{\text{yield}}$	0.064	0.068
ω, RPM	4042	3286
s/s_{bl}	1.0	1.2

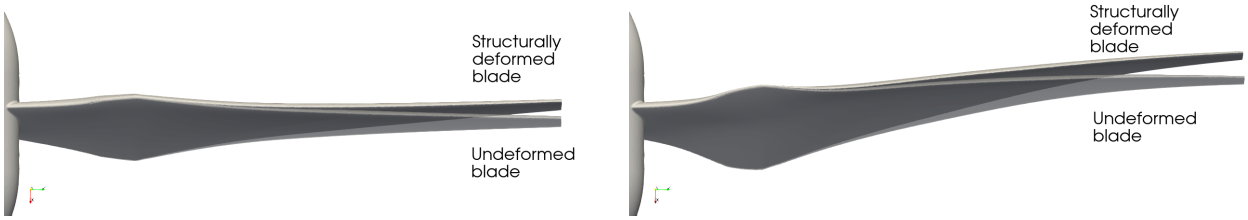

Fig. 11 Comparison between the undeformed and structurally deformed blade geometries for the (left) baseline and (right) optimized designs. The tip deformations are about 3% and 4% of the blade span for the baseline and aerostructurally optimized designs, respectively.

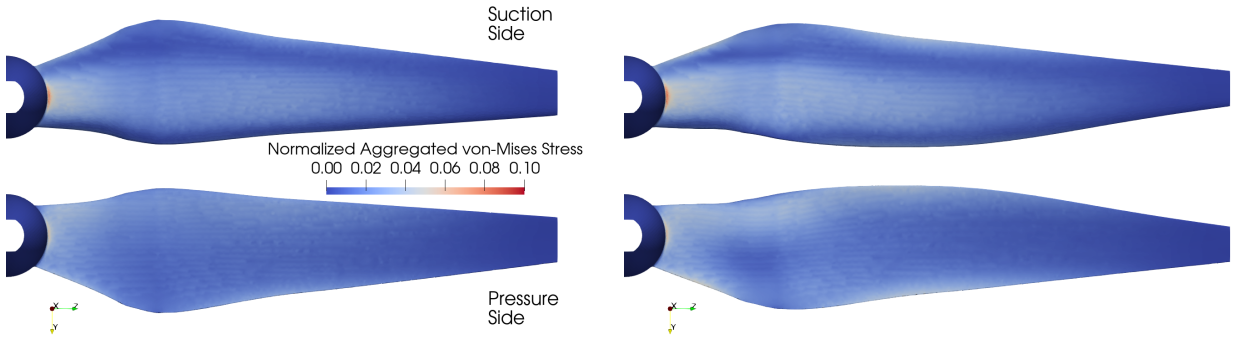

Fig. 12 Comparison of normalized aggregated von-mises stress between the (left) baseline and (right) aerostructurally optimized designs.

In total, we have 246 constraints.

Table 4 shows the summary of the aerostructural optimization. We obtain 11.8% power reduction and the thrust, mass, and stress constraints are satisfied. The thrust constraint is active while the mass and stress constraints are inactive. Similar to what we found for the aerodynamic case, the optimized design decreases the rotation speed and increases the span to reduce the power.

Figure 11 shows the comparison between the undeformed and structurally deformed blade geometries for the baseline and optimized designs. As expected, the structural deformation increases along the spanwise direction. The tip deformations are about 3% and 4% of the blade span for the baseline and optimized designs, respectively The blade deforms mostly in the axial direction due to the pressure load and the corresponding axial force (thrust). We also observe that structural deformation from the baseline design is smaller than that from the optimized design, especially at the blade tip. This is primarily because the optimized design has a larger span, although the total axial force (thrust) is the same between the

Downloaded by Iowa State University on October 26, 2023 | http://arc.aiaa.org | DOI: 10.2514/6.2023-0531

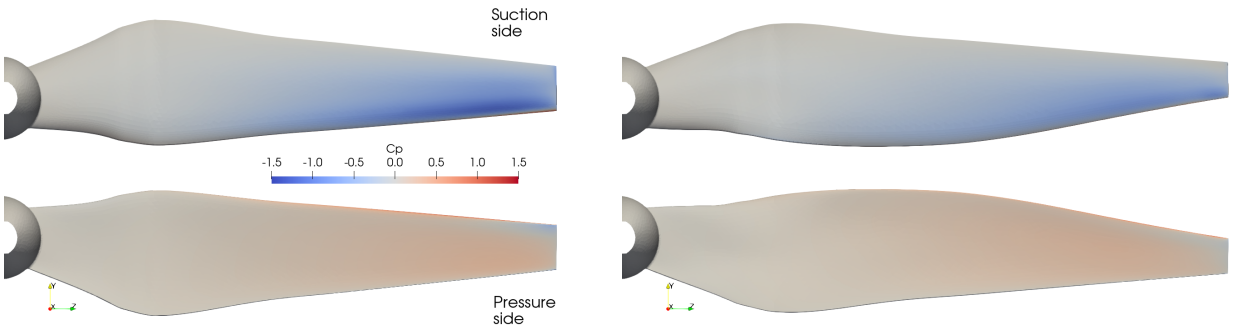

Fig. 13 Comparison of pressure contours between the (left) baseline and (right) aerostructurally optimized designs.

Downloaded by Iowa State University on October 26, 2023 | http://arc.aiaa.org | DOI: 10.2514/6.2023-0531

baseline and optimized designs.

Figures 12 and 13 show the comparison of normalized aggregated von-mises stress and pressure between the baseline and optimized designs. As expected, we observe a relatively high-stress concentration region near the blade root for both baseline and optimized designs. However, the maximal stress is within 10% of the material's yield stress. Similar to the aerodynamic case, the optimized design's pressure difference (load) is lower than the baseline design, as shown in Fig. 13, which can be seen more clearly in Fig. 14. Figure 15 shows the comparison of thrust distribution along the span between the baseline and aerostructurally optimized designs. Overall, we observe a similar trend as in the aerodynamic case. The optimized design decreases the peak thrust near the tip but increases the span to maintain the total integrated thrust.

IV. Conclusion

In this paper, we conduct aerodynamic and aerostructural optimizations for a UAV propeller running at a relatively low Reynolds number (40,000). We use a high-fidelity CFD solver (OpenFOAM) to simulate the flow and a FEM solver (TACS) to simulate the structure. We then use the discrete adjoint approach to compute the derivatives of objective and constraint functions with respect to a large number of design variables and constraints using DAFoam (fluid) and TACS (structure). The adjoint method allows us to have large design freedom for performance improvement. The fluid-structure interaction and its derivative computation are conducted building on NASA's OpenMDAO/MPhys framework.

The objective function is the propeller shaft power, the design variables are the propeller shape, twist, span, chord, and rotation speed, and the constraints are the thrust and geometry (e.g., blade thickness and surface curvature). Our aerodynamic optimization results show that allowing both the shape, platform (i.e., span, and chord), and rotation speed to change (shape+planform case) during the optimization provides 12.8% power reduction, and all constraints are satisfied. The shape+planform case exhibits 5.4% more power reduction, compared with allowing only the shape variables to change (shape-only case). Compared with the baseline design, the shape+planform optimized design increases the span and decreases the rotation speed. The pressure load near the tip is lower in the shape+planform optimized design; however, the span increases to ensure the constant integrated thrust force, enforced by the thrust constraint in the optimization. The aerostructural optimization exhibits slightly lower power reduction (11.8%) compared with the aerodynamic optimization. However, the overall optimized designs are similar between the aerodynamic and aerostructural optimizations. We observe clear structural deformation near the blade tip, which justifies the need for considering fluid-structure interaction in the optimization.

This study has the potential to significantly reduce the time period for designing high-performance UAV propellers. We are currently testing the performance of the baseline and optimized designs in the Iowa State University low-speed wind tunnel and will report more detailed results in future work.

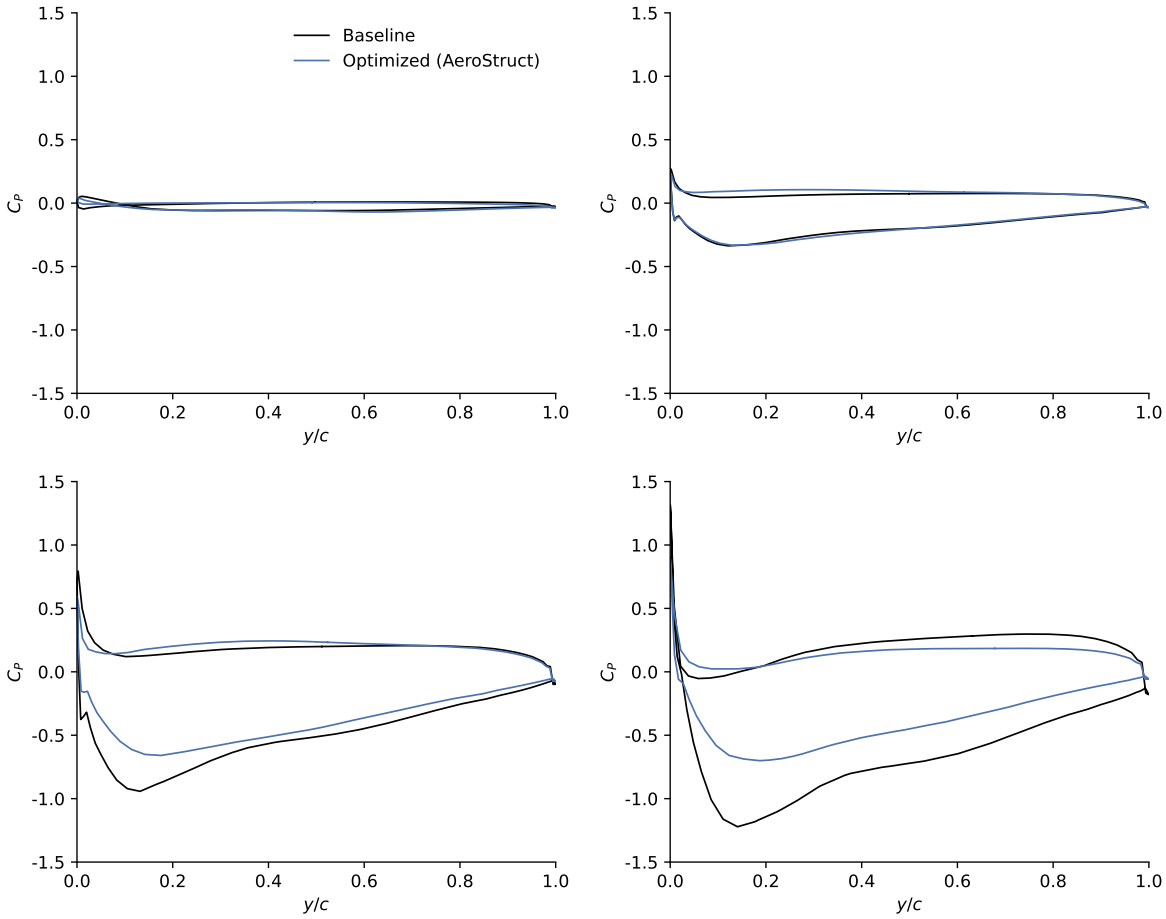

Fig. 14 Comparison of pressure profiles at 10% (top left), 40% (top right), 70% (bot left), 90% (bot right) spanwise locations between the baseline and aerostructurally optimized designs.

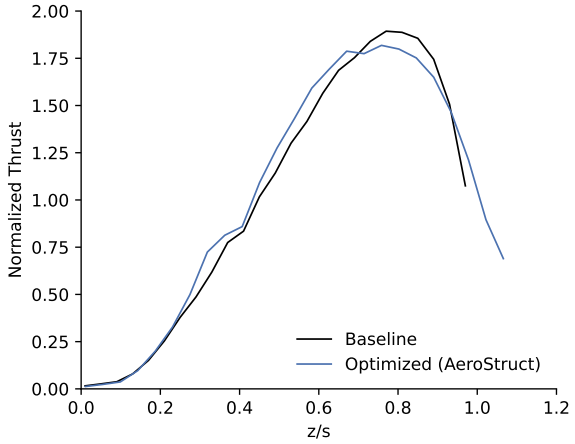

Fig. 15 Comparison of thrust distribution along the span between the baseline and aerostructurally optimized designs.

Downloaded by Iowa State University on October 26, 2023 | http://arc.aiaa.org | DOI: 10.2514/6.2023-0531

Acknowledgments

Ping He acknowledges the funding supported by the National Science Foundation under Grant Number 2223676. The computations were done in the Extreme Science and Engineering Discovery Environment (XSEDE), which is supported by National Science Foundation Grant number ACI-1548562.

References

[1] Hassanalian, M., and Abdelkefi, A., "Classifications, applications, and design challenges of drones: A review," *Progress in Aerospace Sciences*, Vol. 91, 2017, pp. 99–131.

[2] Martins, J. R., and Lambe, A. B., "Multidisciplinary design optimization: a survey of architectures," *AIAA journal*, Vol. 51, No. 9, 2013, pp. 2049–2075.

[3] Jameson, A., "Aerodynamic design via control theory," *Journal of scientific computing*, Vol. 3, No. 3, 1988, pp. 233–260.

[4] Giles, M. B., Duta, M. C., Muller, J.-D., and Pierce, N. A., "Algorithm developments for discrete adjoint methods," *AIAA journal*, Vol. 41, No. 2, 2003, pp. 198–205.

[5] Peter, J. E., and Dwight, R. P., "Numerical sensitivity analysis for aerodynamic optimization: A survey of approaches," *Computers & Fluids*, Vol. 39, No. 3, 2010, pp. 373–391.

[6] Kenway, G. K., Mader, C. A., He, P., and Martins, J. R., "Effective adjoint approaches for computational fluid dynamics," *Progress in Aerospace Sciences*, Vol. 110, 2019, p. 100542.

[7] Reuther, J., Jameson, A., Farmer, J., Martinelli, L., and Saunders, D., "Aerodynamic shape optimization of complex aircraft configurations via an adjoint formulation," *34th aerospace sciences meeting and exhibit*, 1996, p. 94.

[8] Kenway, G. K., and Martins, J. R., "Multipoint high-fidelity aerostructural optimization of a transport aircraft configuration," *Journal of Aircraft*, Vol. 51, No. 1, 2014, pp. 144–160.

[9] Lyu, Z., Kenway, G. K., and Martins, J. R., "Aerodynamic shape optimization investigations of the common research model wing benchmark," *AIAA journal*, Vol. 53, No. 4, 2015, pp. 968–985.

[10] Dhert, T., Ashuri, T., and Martins, J. R., "Aerodynamic shape optimization of WIND TURBINE blades using a Reynolds-averaged Navier–Stokes model and an adjoint method," *Wind Energy*, Vol. 20, No. 5, 2017, pp. 909–926.

[11] Madsen, M. H. A., Zahle, F., Sørensen, N. N., and Martins, J. R., "Multipoint high-fidelity CFD-based aerodynamic shape optimization of a 10 MW wind turbine," *Wind Energy Science*, Vol. 4, No. 2, 2019, pp. 163–192.

[12] Wang, D., and He, L., "Adjoint aerodynamic design optimization for blades in multistage turbomachines—Part I: Methodology and verification," *Journal of Turbomachinery*, Vol. 132, No. 2, 2010.

[13] Ma, C., Su, X., and Yuan, X., "An efficient unsteady adjoint optimization system for multistage turbomachinery," *Journal of Turbomachinery*, Vol. 139, No. 1, 2017, p. 011003.

[14] He, P., Mader, C. A., Martins, J. R., and Maki, K. J., "Aerothermal optimization of a ribbed U-bend cooling channel using the adjoint method," *International Journal of Heat and Mass Transfer*, Vol. 140, 2019, pp. 152–172.

[15] Hwang, J. T., and Ning, A., "Large-scale multidisciplinary optimization of an electric aircraft for on-demand mobility," *2018 AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, 2018, p. 1384.

[16] Moore, K. R., and Ning, A., "Takeoff and performance trade-offs of retrofit distributed electric propulsion for urban transport," *Journal of Aircraft*, Vol. 56, No. 5, 2019, pp. 1880–1892.

[17] Gray, J. S., Hwang, J. T., Martins, J. R. R. A., Moore, K. T., and Naylor, B. A., "OpenMDAO: An open-source framework for multidisciplinary design, analysis, and optimization," *Structural and Multidisciplinary Optimization*, Vol. 59, No. 4, 2019, pp. 1075–1104. https://doi.org/10.1007/s00158-019-02211-z.

[18] "Mphys code," https://github.com/OpenMDAO/mphys, 2022. Accessed: 2023-01-01.

[19] He, P., Mader, C. A., Martins, J. R. R. A., and Maki, K. J., "An Aerodynamic Design Optimization Framework Using a Discrete Adjoint Approach with OpenFOAM," *Computers & Fluids*, Vol. 168, 2018, pp. 285–303. https://doi.org/10.1016/j.compfluid.2018.04.012.

Downloaded by Iowa State University on October 26, 2023 | http://arc.aiaa.org | DOI: 10.2514/6.2023-0531

[20] He, P., Mader, C. A., Martins, J. R. R. A., and Maki, K. J., "DAFoam: An Open-Source Adjoint Framework for Multidisciplinary Design Optimization with OpenFOAM," *AIAA Journal*, Vol. 58, No. 3, 2020, pp. 1304–1319. https://doi.org/10.2514/1.J058853.

[21] Weller, H. G., Tabor, G., Jasak, H., and Fureby, C., "A tensorial approach to computational continuum mechanics using object-oriented techniques," *Computers in physics*, Vol. 12, No. 6, 1998, pp. 620–631.

[22] Spalart, P., and Allmaras, S., "A One-Equation Turbulence Model for Aerodynamic Flows," *30th Aerospace Sciences Meetingand Exhibit*, AIAA Paper 1992-439, 1992. https://doi.org/org/10.2514/6.1992-439.

[23] Kennedy, G. J., and Martins, J. R., "A parallel finite-element framework for large-scale gradient-based design optimization of high-performance structures," *Finite Elements in Analysis and Design*, Vol. 87, 2014, pp. 56–73.

[24] Mader, C. A., Kenway, G. K., Yildirim, A., and Martins, J. R., "ADflow: An open-source computational fluid dynamics solver for aerodynamic and multidisciplinary optimization," *Journal of Aerospace Information Systems*, Vol. 17, No. 9, 2020, pp. 508–527.

[25] Nielsen, E. J., and Anderson, W. K., "Aerodynamic design optimization on unstructured meshes using the Navier-Stokes equations," *AIAA journal*, Vol. 37, No. 11, 1999, pp. 1411–1419.

[26] Nielsen, E. J., Lu, J., Park, M. A., and Darmofal, D. L., "An implicit, exact dual adjoint solution method for turbulent flows on unstructured grids," *Computers & Fluids*, Vol. 33, No. 9, 2004, pp. 1131–1155.

[27] Jasa, J. P., Hwang, J. T., and Martins, J. R. R. A., "Open-source coupled aerostructural optimization using Python," *Structural and Multidisciplinary Optimization*, Vol. 57, No. 4, 2018, pp. 1815–1827. https://doi.org/10.1007/s00158-018-1912-8.

[28] Secco, N. R., Kenway, G. K., He, P., Mader, C., and Martins, J. R., "Efficient mesh generation and deformation for aerodynamic shape optimization," *AIAA Journal*, Vol. 59, No. 4, 2021, pp. 1151–1168. https://doi.org/10.2514/1.J059491.

[29] "FUNtoFEM code," https://github.com/smdogroup/funtofem, 2022. Accessed: 2023-01-01.

[30] "DAFoam Documentation," https://dafoam.github.io, April 20, 2022. Accessed: 2022-04-21.

[31] Kenway, G. K. W., Mader, C. A., He, P., and Martins, J. R. R. A., "Effective adjoint approaches for computational fluid dynamics," *Progress in Aerospace Sciences*, Vol. 110, 2019, p. 100542. https://doi.org/10.1016/j.paerosci.2019.05.002.

[32] Balay, S., Abhyankar, S., Adams, M. F., Brown, J., Brune, P., Buschelman, K., Dalcin, L., Dener, A., Eijkhout, V., Gropp, W. D., Kaushik, D., Knepley, M. G., May, D. A., McInnes, L. C., Mills, R. T., Munson, T., Rupp, K., Sanan, P., Smith, B. F., Zampini, S., Zhang, H., and Zhang, H., "PETSc Users Manual," *Tech. Rep. ANL-95/11 - Revision 3.10*, Argonne National Laboratory, 2018. https://doi.org/10.2172/1483828.

[33] Martins, J. R., and Hwang, J. T., "Review and unification of methods for computing derivatives of multidisciplinary computational models," *AIAA journal*, Vol. 51, No. 11, 2013, pp. 2582–2599.

[34] Kenway, G. K. W., Kennedy, G. J., and Martins, J. R. R. A., "A CAD-Free Approach to High-Fidelity Aerostructural Optimization," *Proceedings of the 13th AIAA/ISSMO Multidisciplinary Analysis Optimization Conference*, Fort Worth, TX, 2010. AIAA 2010-9231.

[35] Zhou, W., Ning, Z., Li, H., and Hu, H., "An experimental investigation on rotor-to-rotor interactions of small UAV propellers," *35th AIAA applied aerodynamics conference*, 2017, p. 3744.

[36] Han, N., Zhang, Z., and Hu, H., "An Experimental Investigation on the Aerodynamic and Aeroacoustic Characteristics of Dual-Rotor Propellers for UAV Applications," *AIAA AVIATION 2022 Forum*, 2022, p. 3892.

[37] Koyuncuoglu, H. U., and He, P., "Simultaneous wing shape and actuator parameter optimization using the adjoint method," *Aerospace Science and Technology*, Vol. 130, 2022, p. 107876.

Downloaded by Iowa State University on October 26, 2023 | http://arc.aiaa.org | DOI: 10.2514/6.2023-0531