Towards Low-Noise Design of a Proprotor with the Vortex Lattice Method and Gradient-Based Optimization

Daniel Ingraham*
NASA Glenn Research Center, Cleveland, Ohio, 44135

David Kinney†
NASA Ames Research Center, Moffett Field, California, 94035

This work aims to combine an aerodynamic model based on the unsteady vortex lattice method with an acoustic model provided by Farassat's formulation 1A to perform gradient-based optimizations of a proprotor with aerodynamic and acoustic constraints. The resulting combination of tools is applied to the problem of designing a single proprotor operating at a cruise condition, with and without an acoustic constraint and wing placed downstream of the proprotor rotation plane. Results are compared to a baseline design studied previously, and to similar optimizations performed with a simpler blade element momentum theory aerodynamic model. Each optimization case achieved feasibility and made significant improvements in the objective function, but the optimality criterion was not satisfied. Overall the designs the optimizer found roughly comported with our previous experience with similar problems, with some discrepancies that are discussed.

I. Nomenclature

C	=	proprotor chord spline control points
c	=	proprotor chord distribution
D	=	proprotor diameter
$\vec{f}$	=	proprotor loading distribution
P	=	proprotor pitch
R_{tip}	=	proprotor tip radius, i.e. $D/2$
r	=	radial location on the proprotor blade
T	=	proprotor thrust
$\vec{x}_{\text{VLM}}$	=	blade surface coordinate locations for the Vortex Lattice Method
$\vec{x}_{1/4c}$	=	blade quarter-chord locations
α	=	local angle of attack
η	=	proprotor propulsive efficiency
ϕ	=	local velocity angle
Θ	=	proprotor twist spline control points
θ	=	proprotor twist distribution
Ω	=	proprotor rotation rate

II. Introduction

Noise, especially rotor/propeller noise, is anticipated to be a critical factor in the design of urban air mobility (UAM) and advanced air mobility (AAM) vehicles. As the name suggests, UAM vehicles will operate in densely-populated areas, exposing the public to the noise they produce. If UAM/AAM vehicles are to be acceptable to the general public, then, designers need tools that allow them to quickly determine the effect design changes have on vehicle aerodynamic and acoustic performance[1]. Such tools need to be fast and accurate enough to qualitatively capture impacts from these changes, and ideally would be able to determine optimal configurations for a given objective and set of design constraints.

*Research Aerospace Engineer, Acoustics Branch, MS 54-3, Member AIAA
†Research Aerospace Engineer

Computational fluid dynamics (CFD), while quite general and powerful, is as of yet still too computationally intensive to be used in routine optimizations, especially when acoustic predictions are needed (though there is promising work in this area, e.g., Içke et al. [2]). Lower-order propeller aerodynamic prediction tools like blade element momentum theory (BEMT) work quite well in simplified situations and can be used for low-noise design (see Ingraham et al. [3]), but are limited in the types of problems they can reasonably attack.

When considering multi-disciplinary optimization, gradient-based techniques are attractive, as their computational cost scales much better than gradient-free approaches as the number of design variables increases (c.f. [4], chapter 7), and should exhibit faster convergence rates than gradient-free methods (c.f. [4], chapter 3). While the number of design variables considered in this work would not preclude using gradient-free optimization algorithms, the ultimate goal of this effort involves embedding the models described here into larger multi-disciplinary optimizations that would significantly increase the number of design variables. Additionally, despite the careful attention paid to the computational efficiency of the software used in this work, the optimizations we will describe can take a significant amount of wall clock time, and thus the accelerated convergence provided by gradient-based approaches is welcome.

The vortex lattice method (VLM, c.f. [5, 6]) is an attractive approach to predicting aerodynamic forces on lifting surfaces. More capable than pseudo-2D approaches like BEMT, it has considerably more modest computational requirements than CFD. Others have investigated using the vortex lattice method to design optimal propellers. Chang and Sullivan[7] optimized the twist distribution of a propeller blade subject to a shaft power constraint via a VLM implementation and gradient-based optimization. Miller[8] used VLM and gradient-based optimization to show the effect the twist and chord distribution, propeller sweep, diameter, rotation rate, and blade count have on aerodynamic efficiency and noise. Cho and Lee[9] found optimal propeller blade twist and chord distributions, again using gradient-based optimization and a VLM. Finally, Burger et al.[10, 11] combined a VLM implementation and a genetic algorithm to perform single and multi-objective propeller aerodynamic optimizations, and included noise constraints[11].

The preceding references, to our knowledge, used some form of a prescribed wake when modeling the propeller wake with the vortex lattice method. More recently, Ingraham[12] showed how an unsteady, free-wake vortex lattice method could be combined with an acoustic model provided by Farassat's formulation 1A [13] to design a low-noise propeller with gradient-based optimization. In that work, two cases were attempted: an isolated propeller and a propeller immediately upstream of a wing, both operating in a Mach 0.11 free stream, representative of a cruise condition. Most of that effort was focused on modifying the VLM implementation employed there to be amenable to gradient-based optimization. This work will explore using VSPAERO, a different and in some ways more advanced VLM implementation, in conjunction with other tools, to perform similar proprotor design problems. Specifically, the isolated propeller and propeller-wing optimizations will be repeated, and will include both aerodynamic and acoustic constraints. Comparisons between similar optimizations with a lower-fidelity aerodynamic model using Blade Element Momentum Theory will also be made, and the effect of adding constraints on the chord and twist distributions' concavity will be shown.

III. Models and Approach

The optimization problem will be constructed using OpenMDAO[14, 15], a multi-disciplinary optimization framework focused on gradient-based optimization techniques. OpenMDAO allows a user to define units of computation called `Component`s, and then arrange and connect these `Component`s in arbitrarily complex ways. Each `Component` is responsible for calculating the outputs associated with its inputs and, optionally, the derivatives of each output with respect to the inputs (the "partial derivatives"). OpenMDAO is then able to propagate these partial derivatives though the model to compute "total derivatives," i.e., derivatives of the responses (objective and constraints) with respect to each design variable, and then pass this information to a gradient-based optimizer to solve the optimization problem. The sparse nonlinear optimizer SNOPT[16] will be used here, called via the pyOptSparse software package[17].

An Extended Design Structure Matrix (XDSM[18]) of the optimization problem that will be the focus of this work is shown in Figure 1. As shown in the Figure, the first step in constructing the design optimization problem is parametrizing the geometry—the optimizer must be able to change the geometry in an automated fashion before being analyzed by the aerodynamic and acoustic models. In this work, the proprotor geometry will be parameterized via OpenVSP[19]*. Specifically, at each optimization iteration, the chord and twist distribution along the blade will be defined by PCHIP splines available in OpenVSP, using 6 evenly-spaced control points from hub to tip. After the chord and twist control points are set, OpenVSP updates the model geometry and constructs $\vec{x}_{\text{VLM}}$, the mesh input for VSPAERO's VLM implementation. The geometry parameterization will be integrated into the optimization problem

*`https://openvsp.org/`

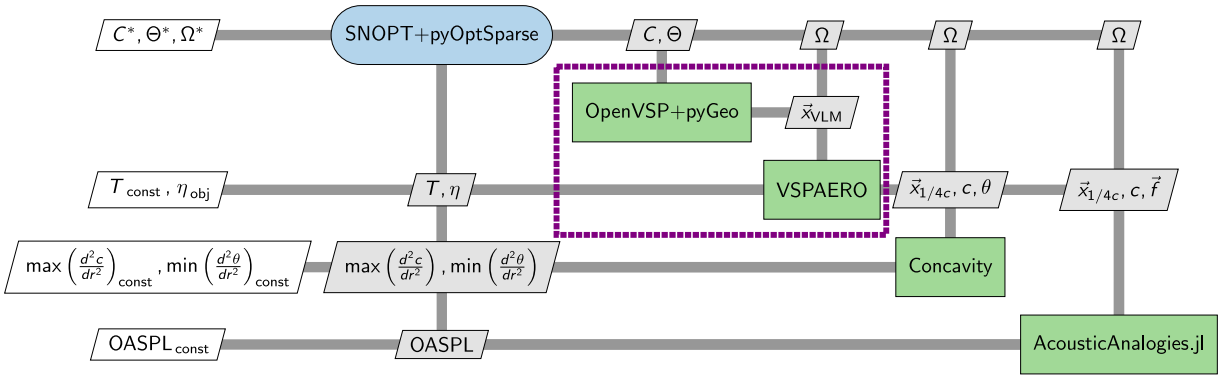

Fig. 1 XDSM of the complete optimization problem in this work, including chord concavity, twist concavity, and acoustic constraints.The dotted purple rectangle indicates the group of OpenMDAO `Components` differentiated using the finite difference method.

using the pyGeo software package[20].

The VSPAERO code [21], part of the OpenVSP software package, will provide the VLM implementation for this work. In particular, we will use the unsteady form of the vortex lattice method implemented in VSPAERO, a "free wake" approach closely following that described in Chapter 13 of Katz and Plotkin [5]. (VSPAERO also includes a steady VLM and both steady and unsteady panel methods, among many other features.) In addition to the free-wake unsteady VLM, VSPAERO also approximates compressibility effects through the usual Prandtl-Glauert correction, and includes models to capture span-local viscous loading and stall effects. Here, at each optimization iteration, VSPAERO will take in the mesh constructed by OpenVSP $\vec{x}_{\text{VLM}}$ and Ω, the rotation rate of the proprotor, and calculate the thrust T and propulsive efficiency η. Additionally, VSPAERO will calculate c, θ, and $\vec{x}_{1/4c}$, the chord, twist, and quarter-chord locations for the 20 spanwise stations present in the VLM mesh, and the loading per unit span $\vec{f}$ along the blade's span. VSPAERO is run for three revolutions worth of time, with twenty time steps per revolution. As shown in Figure 1's XDSM, these additional output quantities are used in the "Concavity" and acoustic models, to be discussed below.

As will be discussed in the results section, constraining the chord and twist concavity had a beneficial effect on the optimization results. To calculate the chord and twist concavity, standard second-order second derivative finite differences were used, dropping to first-order at the hub and tip (c.f. [22], page 51), to differentiate c and θ with respect to the radial or spanwise location r. The interest here is in the maximum value of $\frac{d^2c}{dr^2}$ and minimum value of $\frac{d^2\theta}{dr^2}$, which is calculated in a differentiable manner via the Kreisselmeier-Steinhauser constraint aggregation function (c.f. [23]), as implemented in the FLOWMath.jl software package*. The derivatives of the "Concavity" `Component` were calculated using ForwardDiff.jl[24], an automatic differentiation library.

We use the compact form of Farassat's formulation 1A [13] (see also [25]), including the compact monopole approximation of Lopes [26] to calculate the tonal noise produced by the propeller, as implemented in the open-source software package AcousticAnalogies.jl†. Farassat's formulation 1A (F1A) is an acoustic analogy, i.e, a rearrangement of the Navier-Stokes equations into a left hand "propagation" and right hand "source" sides. The acoustic analogy equation takes the form of an inhomogeneous wave equation, and is solved using a free-space Green's function. The application of the Green's function to the acoustic analogy equation results in two surface integrals and one volume integral. The two surface integrals represent the tonal noise due to the thickness and loading on the blade's surface, while the volume integral captures nonlinear sources of sound off the surface (e.g., shocks and turbulence). The volume integral will be neglected in this work, as the blade tip speeds in the target problem are low. Broadband noise sources, though significant in these types of problems[27], will not be considered in this work. The "compactness" of the approach refers to the simplification of the source's geometry (here, the propeller blade). As discussed by Lopes, if the acoustic source is elongated in one direction and the observer is sufficiently far from the source, the surface integrals associated with the Green's function can be replaced by line integrals along the blade's span. This results in considerable computational cost savings in the acoustic model.

*`https://github.com/byuflowlab/FLOWMath.jl`

†`https://github.com/OpenMDAO/AcousticAnalogies.jl`

Table 1 Parameters for the baseline propeller design

number of blades	3
diameter	24 inches
hub diameter	4.8 inches
chord	1.5 inches
pitch	16 inches
design RPM	≈ 7100
airfoil cross-section	NACA 0012

A few comments on the particulars of the acoustic model are necessary to fully defined the `Component`. As shown in Figure 1, the acoustic model requires the chord and loading distribution along the blade's span, and the rotation rate of the propeller. The chord is used to calculate cross-sectional area associated with each spanwise station, an input to the thickness part of the F1A calculation. The loading $\vec{f}$ from the VLM exists on a time discretization that is too coarse for an accurate acoustic prediction, and is thus interpolated onto a denser time grid of 32 time steps over two blade passes using Akima splines[28], again from FLOWMath.jl. Like the "Concavity" `Component`, the derivatives of the acoustic `Component` are calculated via the ForwardDiff.jl package.

Results will be compared to a baseline propeller design [29] throughout this work. Parameters of the baseline design are displayed in Table 1. The propeller twist is defined by

$$\theta(r) = \arctan\left(\frac{P}{\pi D \frac{r}{R_{\text{tip}}}}\right), \tag{1}$$

where r is the radial location along the blade's span, D and P are the diameter and pitch defined in Table 1, and R is the propeller tip radius. We will also compare the VSPAERO results to identical optimizations with CCBlade.jl[30, 31], a blade element momentum theory code, replacing the VSPAERO-based aerodynamic model.

To investigate installation effects, optimizations with and without a wing directly downstream of the propeller wing will be compared. The leading edge of the wing's tip will lie on the propeller rotation axis, $1/4R_{\text{tip}}$ downstream from the propeller rotation plane. The wing will have a span of $2R_{\text{tip}}$ and chord of $0.919R_{\text{tip}}$, and angle of attack of 5° relative to a Mach 0.11 free stream aligned with the proprotor rotation axis.

For the cases including an acoustic constraint, the overall sound pressure level (OASPL) at a single downstream "microphone" location will be calculating using the VSPAERO-AcousticAnalogies.jl model. The location is specified in the same coordinate system used by the OpenVSP model of the isolated proprotor and proprotor-wing geometry:

- The origin is in the proprotor rotation plane, on the proprotor rotation axis.
- The proprotor rotation axis is aligned with the x axis, and the proprotor rotates about the negative-x axis.
- The "first" proprotor blade is initially aligned with the y axis.
- The Mach 0.11 freestream velocity acts in the positive x direction.

A screenshot of the OpenVSP proprotor-wing geometry is shown in Figure 2. In this coordinate system, the microphone location is set to $(14.2572R_{\text{tip}}, 7.4590R_{\text{tip}}, 8.8894R_{\text{tip}})$.

Table 2 summarizes the objective, design variables, and constraints included in the complete optimization problem, i.e., the optimization including the chord and twist concavity constraints and the acoustic constraint. For the "complete" optimization problem, there are a total of 13 design variables, four constraints, and a single objective. Optimization without the concavity and acoustic constraints were also attempted and will be presented in next section—a listing of each optimization case is found in Table 3.

IV. Results

A. Case 1: CCBlade.jl

Figure 3 summarizes the results of an initial optimization using the CCBlade.jl-based proprotor aerodynamic model. The blue markers indicate the initial condition of the optimization, which is the baseline propeller described in the

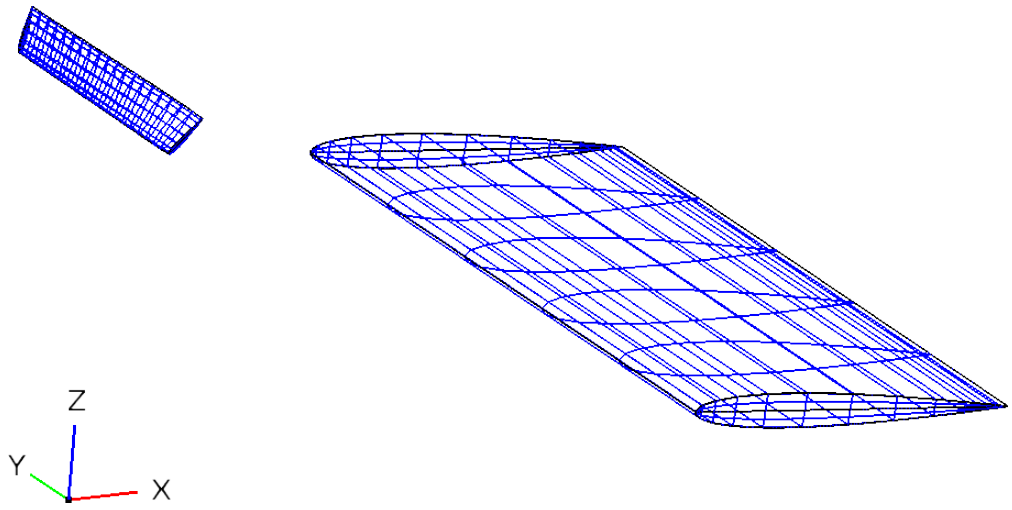

Fig. 2 Screenshot of OpenVSP model of the proprotor-wing geometry, with axes indicating coordinate system orientation. The proprotor blade is the object on the left, the wing on the right.

Table 2 Summary of the complete optimization problem corresponding to the XDSM in Figure 1

	Variable/Function	**Size**
maximize	propulsive efficiency	
with respect to	proprotor rotation rate	1
	chord distribution control points	6
	twist distribution control points	6
subject to	thrust	1
	maximum chord distribution concavity	1
	minimum twist distribution concavity	1
	overall sound pressure level	1

Table 3 Listing of optimization cases

Case Name	**aero model**	**with wing?**	**with concavity constraints?**	**with acoustic constraint?**	**note**
Case 1	CCBlade.jl	no	no	no	
Case 2	VSPAERO	no	no	no	
Case 3	VSPAERO	no	no	no	starting from CCBlade.jl-optimal design
Case 4	VSPAERO	no	yes	no	
Case 5	VSPAERO	no	yes	yes	
Case 6	VSPAERO	yes	yes	yes	

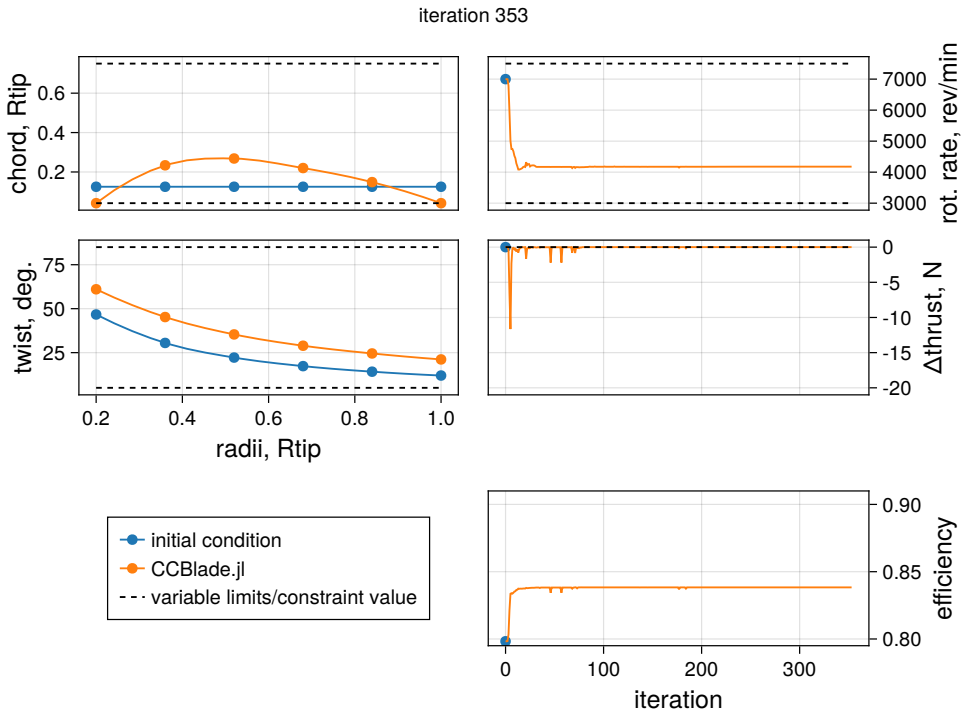

Fig. 3 CCBlade.jl-based optimization results, isolated proprotor, no acoustic constraints

previous section and summarized in Table 1. The orange markers are the CCBlade.jl optimization results. The left column shows the chord and twist distribution at the beginning and end of the optimization, with the circular markers indicating the control points of the PCHIP splines that were used by OpenVSP's Blade Element Analysis feature to construct the chord and twist distribution passed to the CCBlade.jl `Component`, and the solid lines indicating that interpolated chord and twist. The right column shows the convergence history of the propeller rotation rate design variable, thrust constraint, and efficiency objective. The thrust is expressed as Δthrust, the difference between the actual thrust for a given iteration and the target value—so the desired value of the Δthrust quantity is 0.

As seen in Figure 3, the optimizer decides to decrease the proprotor rotation rate while nearly uniformly increasing the twist, increasing efficiency and maintaining the same thrust as the baseline design. The chord is increased slightly throughout the middle of the blade, but is set to the lower limit at the hub and tip, likely to minimize hub and tip losses associated with those parts of the blade. Both the chord and twist distributions look smooth and reminiscent of actual propeller designs. SNOPT was able to converge the optimization successfully after about 350 iterations.

B. Case 2: VSPAERO, isolated proprotor, no concavity or acoustic constraints

Figure 4 shows the results of an optimization identical to that summarized in Figure 3, but with VSPAERO providing the aerodynamic model. Qualitatively the results are somewhat similar to the CCBlade.jl-based optimization in Figure 3: the optimizer also increases the twist and decreases the rotation rate, which again maintains the same thrust value as the baseline design while increasing efficiency. Like the previous CCBlade.jl results, the optimizer minimized the chord at the hub and tip locations, but also nearly did the same at the second chord control point from the tip. The chord distribution also shows a strange "dip" at the third control point from the hub, and the twist distribution is not as smooth as the CCBlade.jl results. SNOPT was unable to converge this optimization—Figure 4 actually shows only a portion of the optimization history. After making good progress at the beginning, the optimization "stalled," i.e. the optimizer continued to run, but made very small changes to any of the design variables after the 1000 iterations shown it the figure. (This was true of all of the VSPAERO-based optimizations presented in this work: the optimizer was able to achieve feasibility relatively quickly and make significant improvements in the objective function, but satisfying the optimality criterion eluded it.)

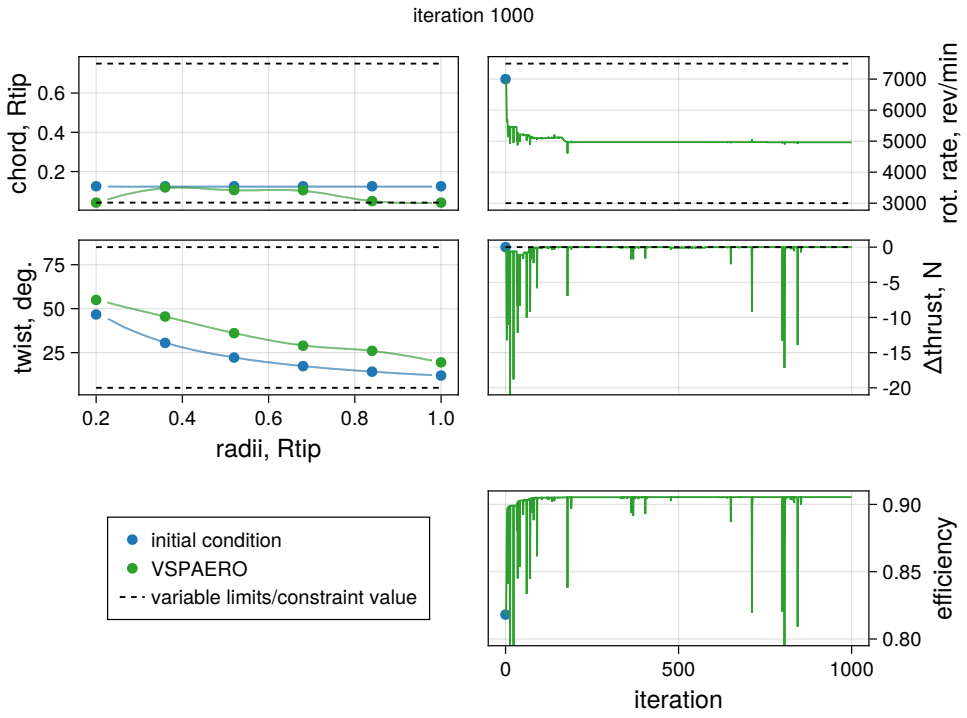

Fig. 4 VSPAERO-based optimization results, isolated proprotor, no acoustic constraints

C. Case 3: VSPAERO, isolated proprotor, no concavity or acoustic constraints, starting from CCBlade.jl-optimal

To investigate the discrepancy between the CCBlade.jl results in Figure 3 and VSPAERO results in Figure 4, the VSPAERO optimization was repeated, but starting from a condition very similar to the optimal design found by CCBlade.jl and shown in Figure 3. The results of this new optimization are shown in Figure 5. The thrust constraint value for the VSPAERO optimization shown in Figure 5 was set to the VSPAERO-predicted value for the CCBlade.jl-optimal design used for the initial condition (i.e., the blue markers in Figure 5), and thus is slightly different from the previous optimization of Figure 4. (This was done to make this latest optimization qualitatively similar to the previous attempts, where the optimization began with a design that satisfied the thrust constraint.) Additionally, the convergence history from this new VSPAERO optimization (i.e. the value of the chord, twist, and rotation rate design variables for each iteration) was fed to the CCBlade.jl-based model and the resulting output plotted alongside the VSPAERO results in Figure 5. The intent of this latest optimization was twofold: first, starting the VSPAERO optimization from a different initial condition should reveal if the optimizer is becoming "stuck" in a local minima—if the optimizer is pulled to a similar design point as previously, a local minima would likely not be the cause of the strange chord and twist results in Figure 4. Second, observing differences in the VSPAERO and CCBlade.jl predictions for the same optimization history may reveal why the results obtained from the two codes are so different, and hopefully provide inspiration as to the source of the discrepancy.

Comparison between the VSPAERO results in figures 5 and 4 shows that, when starting from the CCBlade.jl-optimal result, the optimizer did indeed land on a design very similar to the initial VSPAERO optimization. Relative to the first VSPAERO optimization shown in Figure 4, the optimizer opted to

- decrease the rotation rate from 5000 revolutions/minute to about 4100,
- increase the chord slightly at the fifth chord control point from the hub,
- uniformly increase the twist throughout the blade by a small amount.

As seen in the figures, the changes in the chord and twist were quite minor. The rotation rate was the most significant difference between the two runs, and was likely due to the differing thrust constraint value used for the two optimizations, as discussed previously. Overall, the fact that that optimizer chose a design similar to the initial VSPAERO optimization in figure 4 provides evidence that the optimizer is not becoming trapped in a local minimum, but instead is finding a design that the OpenVSP-VSPAERO model is inherently incentivised to prefer.

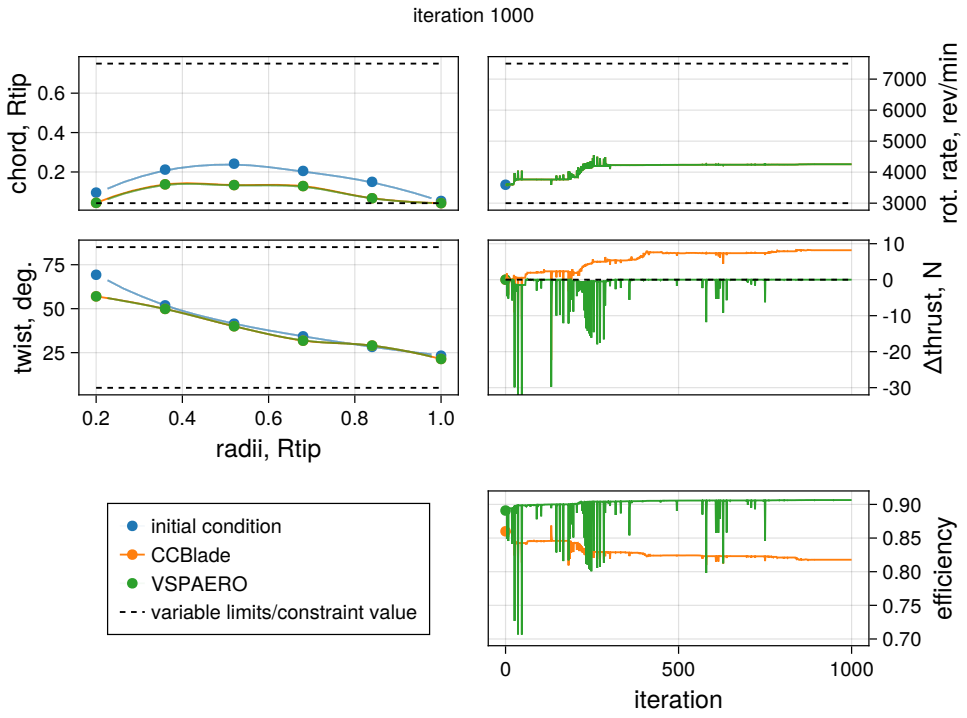

Fig. 5 VSPAERO-based optimization results, isolated proprotor, no acoustic constraints, starting from CCBlade.jl-optimal design

A comparison between the VSPAERO and CCBlade outputs in Figure 5 is also interesting. We notice that both codes show a significant change in efficiency from the start of the optimization to iteration 58, but in opposite directions (the VSPAERO-predicted efficiency increased slightly, while the CCBlade.jl-predicted efficiency decreased significantly). A reproduction of Figure 5 showing the convergence history of the rotation rate, thrust, and efficiency from iteration 1 to 58, and the chord and twist distribution for both iterations 1 and 58, is shown in Figure 6. As in the previous figures, the convergence history of the rotation rate, thrust, and efficiency is shown in orange and green for the CCBlade.jl and VSPAERO codes, respectively. For the chord and twist distributions, the dashed lines show the chord and twist for iteration 1, the solid lines iteration 58. (The chord and twist each iteration is identical for either code, as the VSPAERO convergence history was saved and passed to the CCBlade.jl model, as mentioned previously.) From the figure, we see that the rotation rate changed little from iteration 1 to 58, and the predicted thrust from both codes also changed very slightly. The chord and twist distributions, of course, is where we see the most significant changes. The chord for iteration 56 at the fifth control point from the hub has been reduced significantly relative to iteration 1. Iteration 56's twist, however, has increased at the fifth and sixth control points from the hub (and slightly at the hub control point), and slightly decreased elsewhere.

We see the effect the changes in chord and twist from iteration 1 to 58 have on the blade loading in Figure 7. Rather than showing the rotation rate and thrust convergence history, the plots in the left-hand column of Figure 7 are the normal loading (the loading responsible for the thrust) and circumferential loading (the loading responsible for the torque) along the blade for iteration 1 and 58, and both aerodynamic models. Looking at the normal loading plots, we see that, though the thrust has changed little from iteration 1 to 58, the normal loading distribution has shifted from the middle part of the blade to the tip for both models. This shift is especially pronounced for the CCBlade.jl results, appears to be caused by the increased twist near the tip and decreased twist mid-span (and perhaps by the slight increase in rotation rate), and occurs in spite of iteration 58's decreased chord in the tip region. Similarly, the circumferential loading was also decreased in the middle part of the blade and increased near the tip. But the magnitude of this shift differed between the two codes, with the circumferential loading only slightly increasing with the VSPAERO model near the tip, but more sharply increasing with the CCBlade.jl. Both codes show significant decreases in circumferential loading in the middle of the blade, with again larger changes in the CCBlade.jl model. Overall, these results show that

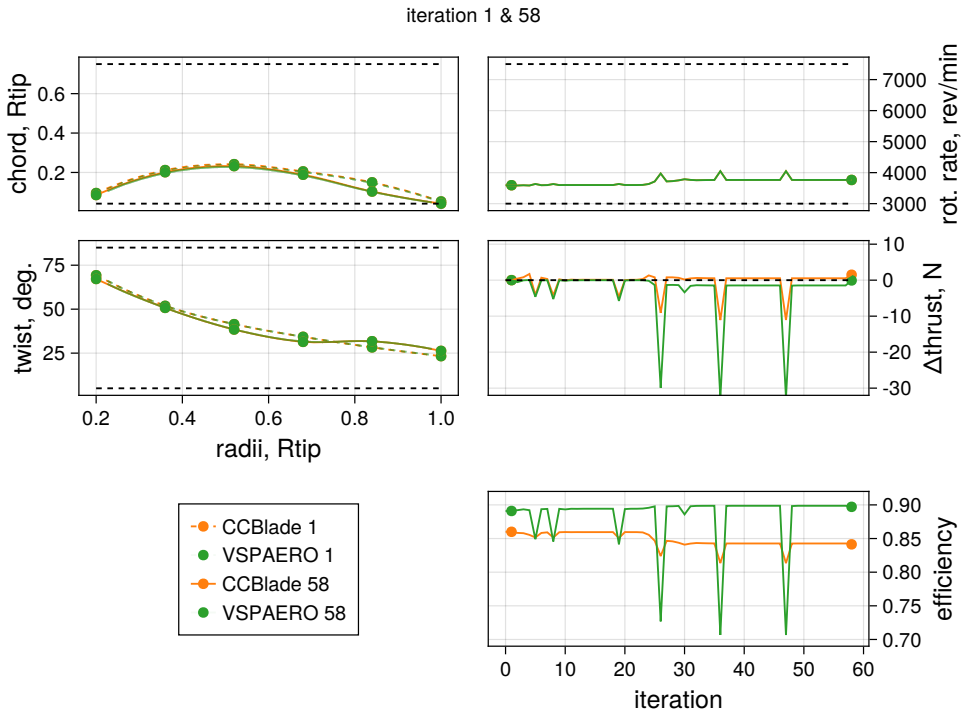

Fig. 6 VSPAERO-based optimization results, isolated proprotor, no acoustic constraints, comparing iterations 1 and 58.The CCBlade.jl results are those obtained by passing the VSPAERO design variable convergence history to the CCBlade.jl-based model.

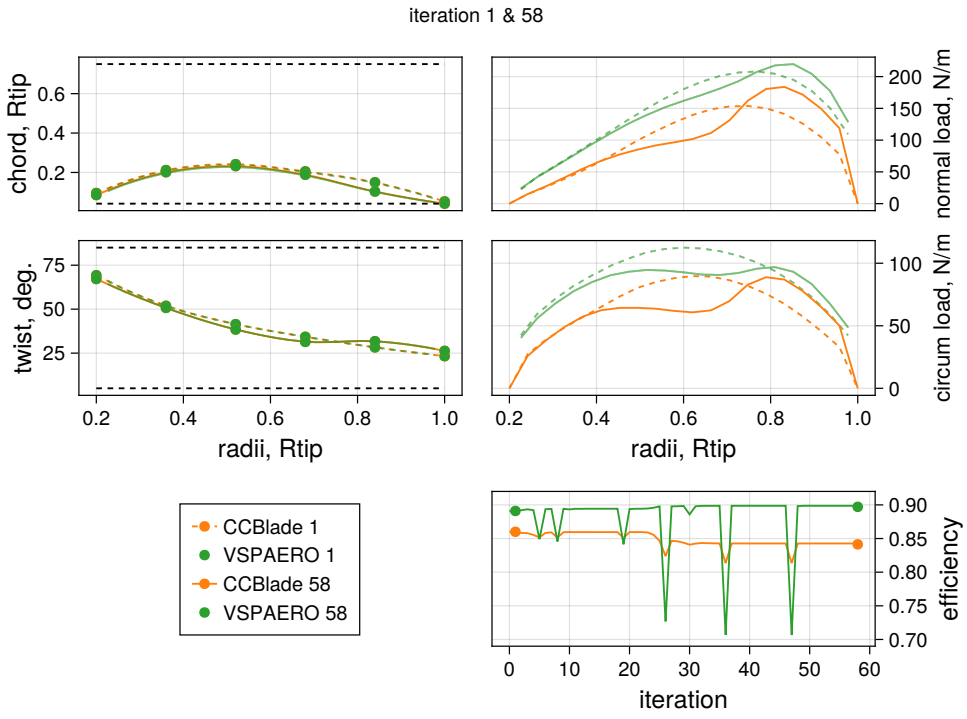

Fig. 7 VSPAERO-based optimization results, isolated proprotor, no acoustic constraints, comparing iterations 1 and 58, including loading distribution. The CCBlade.jl results are those obtained by passing the VSPAERO design variable convergence history to the CCBlade.jl-based model.

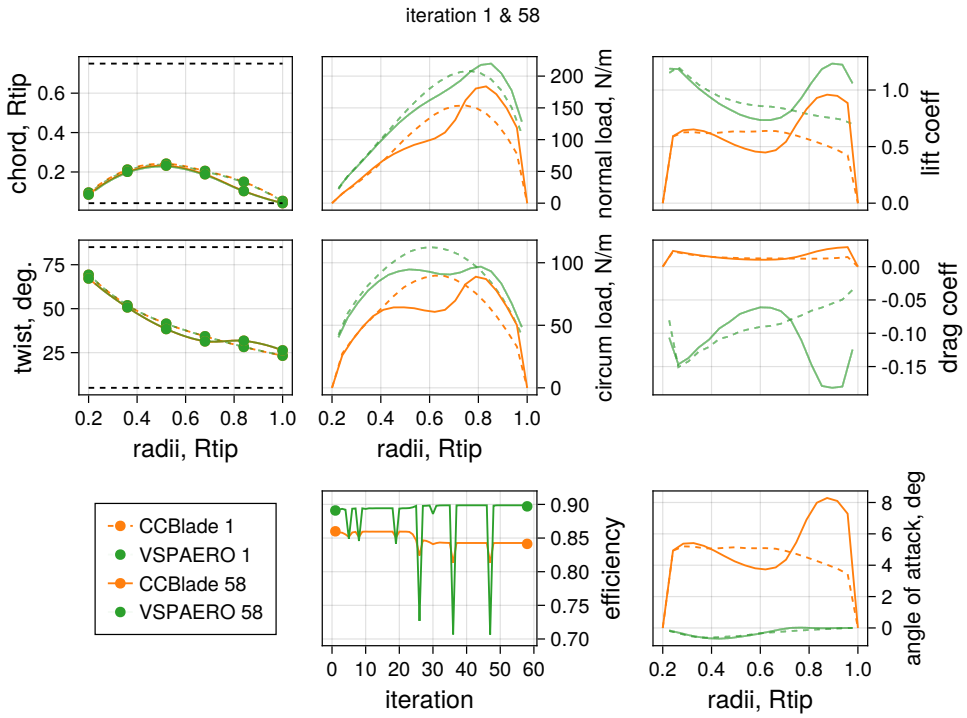

Fig. 8 VSPAERO-based optimization results, isolated proprotor, no acoustic constraints, comparing iterations 1 and 58, including local lift coefficient, drag coefficient, and angle of attack. The CCBlade.jl results are those obtained by passing the VSPAERO design variable convergence history to the CCBlade.jl-based model.

CCBlade.jl is much more sensitive to the twist than VSPAERO. Interestingly, the VSPAERO loading distribution in both directions for iteration 58 is quite smooth—much more so than CCBlade.jl, despite the non-smooth chord and twist.

Figure 8 is meant to provide more insight into the differences between the two aerodynamic models. The figure shows the same data as in the previous Figure 7, but includes an additional column at the right hand side with the local lift coefficient, drag coefficient, and angle of attack along the blade radius. For both aerodynamic models, the twist changes from iteration 1 to 58 have a strong effect on the local lift coefficient distribution. With the CCBlade.jl results, the effect of this change is also clearly seen in the normal and circumferential loading (as discussed previously). The VSPAERO normal loading seems relatively unaffected, despite the large lift coefficient spike in the tip region, and somewhat smaller dip in the middle of the blade. The explanation for this apparent discrepancy in the behavior of the two codes lies in Figure 8's angle of attack plot. First, though, we must remember that the loading associated with the lift coefficient is applied perpendicular to the direction of the local velocity at each radial location. The direction of this local velocity is $\phi = \theta - \alpha$, where θ is the blade twist angle and α the angle of attack at the radial station. As ϕ increases, the loading due to lift is progressively oriented more toward the circumferential (torque) loading direction and away from the normal (thrust) loading direction—all things equal, increasing ϕ will reduce the lift force's contribution to the normal loading and increase its contribution to the circumferential loading. Close inspection of the tip region (say, $r/R_{\text{tip}} \approx 0.9$) of the CCBlade.jl data shows that the twist and angle of attack for iteration 1 is approximately 25° and 4°, respectively, giving a local velocity angle of about $25° - 4° = 21°$. For iteration 58, the CCBlade.jl angle of attack in the same $r/R_{\text{tip}} \approx 0.9$ region increased significantly to 8°, with the twist also increasing to about 30°. But despite these changes, the local velocity angle only increased slightly to $30° - 8° = 22°$. The situation is much different for the VSPAERO model. The twist angles are the same, but the angle of attack is essentially zero throughout the blade for both iterations 1 and 58, but especially at the r/R_{tip} location. Thus the increase in the local flow angle ϕ from iteration 1 to 58 is essentially equal to the increase in twist, which, for the location of interest near the tip, is about 5°. This has the effect of rotating the loading due to lift away from the normal loading direction, and thus explains why the VSPAERO model shows a sharp increase in lift coefficient, but a very small increase in normal loading at $r/r_{\text{tip}} \approx 0.9$.

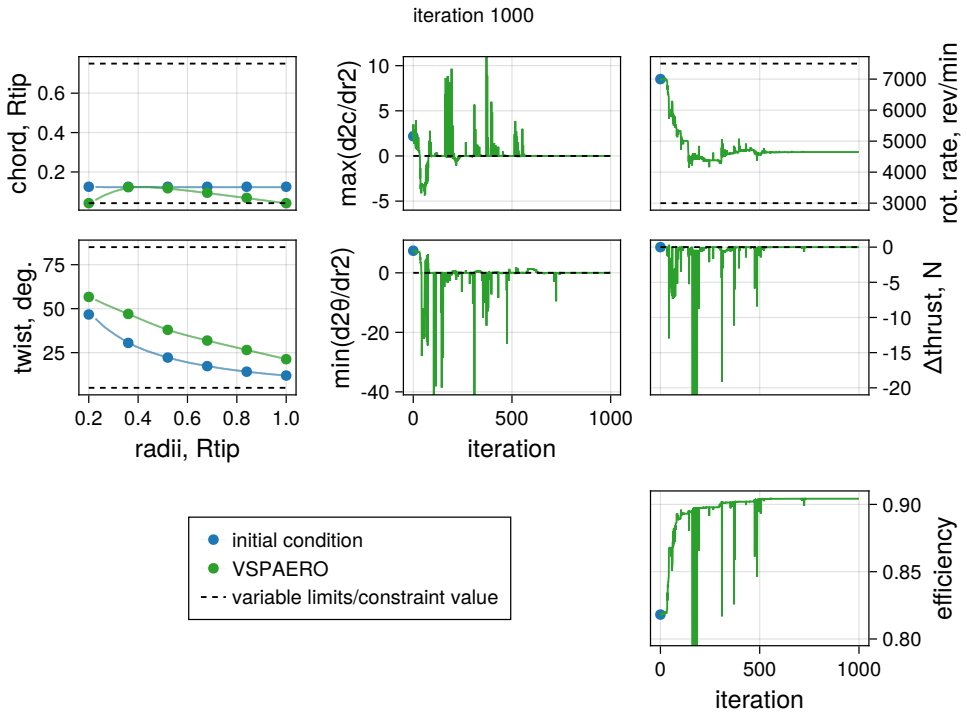

Fig. 9 VSPAERO-optimization results, isolated proprotor, no acoustic constraint, with chord and twist concavity constraints

D. Case 4: VSPAERO, isolated proprotor, with concavity constraints, no acoustic constraint

The comparisons between the VSPAERO and CCBlade.jl results in Figures 6,7,8 show that the VSPAERO model appears to be less sensitive to twist changes in general, and is able to take advantage of small, non-smooth tweaks in both the chord and twist to produce relatively smooth normal and circumferential loading distributions. But these small-scale "tweaks" are not typical of actually propeller blades (c.f. [32]). Very frequently, a propeller blade's chord distribution will be concave down, i.e. the second derivative of the chord with respect to radial position r will be less than zero throughout the span of the blade. Similarly, a propeller blade's twist distribution is usually concave up, meaning the second derivative of twist with respect to r is greater than zero. None of the VSPAERO results presented so far in this work satisfy either relationship, but the CCBlade.jl-optimal design in Figure 3 satisfies both. This motivated adding two inequality constraints to the optimization problem to enforce the concave-down chord and concave-up twist profile. (This was also done in one of our previous efforts [12], and was found to be useful recently in a similar work [33].) As described in the previous section, the second derivative of the chord and twist distribution was calculated via standard finite differences, and the maximum/minimum value for the chord/twist (respectively) was found using the Kreisselmeier-Steinhauser constraint aggregation function. The results of this new optimization are shown in Figure 9. Like the previous figures, Figure 9 includes the chord and twist distribution at the "end" of the optimization in the left-most column, and also the convergence history of the rotation rate, thrust, and efficiency in the right-most column. The middle column shows the convergence history of the maximum value of $\frac{dc}{dr}$ and the minimum value of $\frac{d\theta}{dr}$. Clearly both concavity constraints are active during most of the history, including at the end of the optimization. The results are somewhat similar to the original VSPAERO optimization in Figure 4: the optimizer again decided to decrease the rotation rate and increase the twist, satisfying the thrust constraint and increasing the efficiency. The optimizer also decided to set the chord at the huh and tip to the lower limit, likely to minimize losses associated with those regions, as in the original VSPAERO case. But the small "wiggles" present in Figure 4's chord and twist distributions are removed by the concavity constraints.

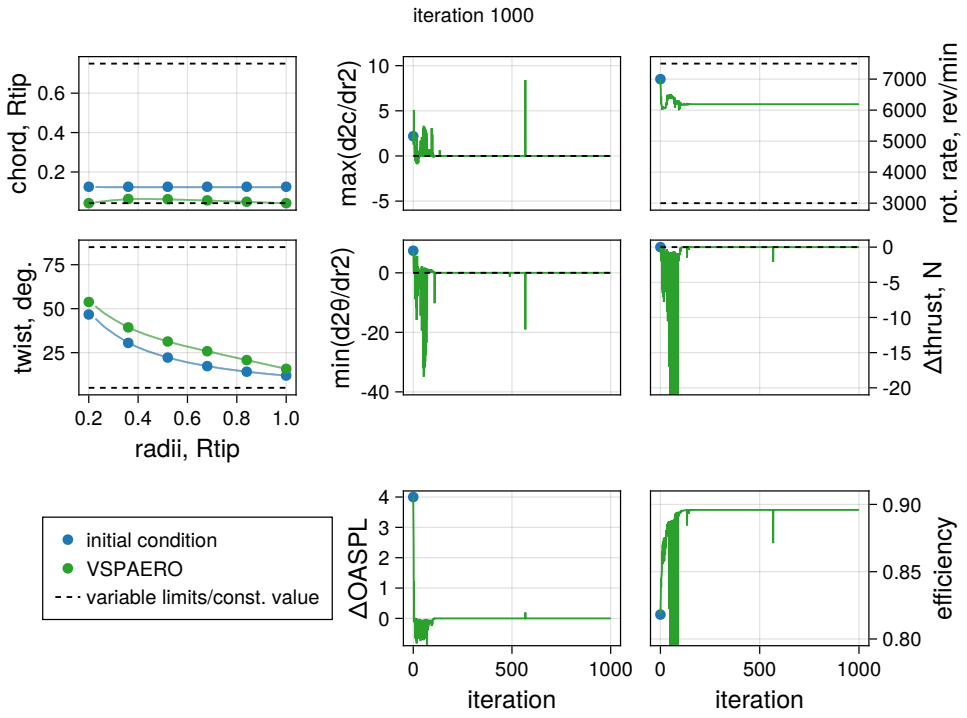

Fig. 10 VSPAERO-optimization results, isolated proprotor, with chord and twist concavity constraints, with acoustic constraint

E. Case 5: VSPAERO, isolated proprotor, with concavity constraints, with acoustic constraint

Figure 10 shows the results of a VSPAERO-based optimization identical to that of Figure 9, including the chord and twist concavity constraints, but also includes an acoustic constraint. Specifically, the overall sound pressure level (OASPL) at a location (described in the Model and Approach section) downstream of the proprotor rotation plane is constrained to be four decibels lower than what is predicted by the VSPAERO-AcousticAnalogies.jl model for the initial condition (again, the C24ND operating at 7000 RPM). As expected, the optimizer again decides to reduce the rotation rate and increase the twist, increasing the efficiency while maintaining the thrust required by the thrust constraint. However, the rotation rate reduction for this case was not nearly as significant as that shown in the previous VSPAERO cases (Figures 4, 5, and 9). Similarly, the increase in twist was also less significant compared to the previous VSPAERO results, and the chord was decreased throughout the middle part of the blade. From previous experience[3, 12] we would expect adding a noise constraint would result in the optimizer choosing to decrease the rotation rate, increase the twist, and increase the chord even more significantly than an equivalent optimization without the constraint. The reason for this deviation is not clear at this time, but we are tempted to blame inaccuracies associated with the finite difference method used to differentiate the OpenVSP-VSPAERO model.

F. Case 5: VSPAERO, proprotor with wing, with concavity constraints, with acoustic constraint

The results of a final test case are shown in Figure 11, which includes a wing immediately downstream of the proprotor rotation plane, as described in the Models and Approach section and shown in Figure 2, but is otherwise identical to the previous shown in Figure 10. Interestingly, the resulting blade design for this case is more similar to that shown in Figure 9 (isolated proprotor, with chord and twist concavity constraints, without the acoustic constraint) than Figure 10 (isolated proprotor, with chord and twist concavity constraints, with the acoustic constraint). Like the design in Figure 9, the optimizer reduced the rotation rate of the proprotor significantly to approximately 4200 RPM, and increased the twist relative to the baseline design by about the same amount. The chord distribution is where the most significant differences are seen: while minimized at the hub and tip like all the previous cases, the optimizer decided to increase the chord significantly compared to the baseline design and Figure 9. The increase in chord must make up for

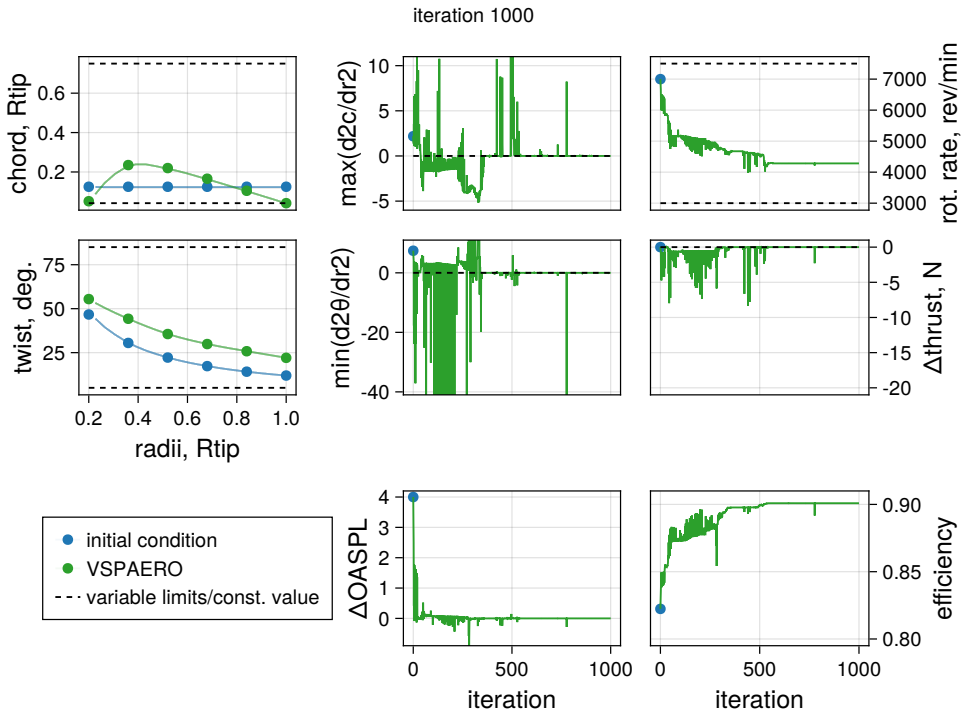

Fig. 11 VSPAERO-optimization results, isolated proprotor, with chord and twist concavity constraints, with acoustic constraint, with wing

the lower rotation rate found in this case, and perhaps losses associated with the presence of the wing.

V. Conclusions

This work showed the results of combining an aerodynamic model based on the unsteady vortex lattice method with an acoustic model provided by Farassat's formulation 1A, and applying this combination of tools to a proprotor design problem. A gradient-based optimizer was used to solve the optimization problem, with the combination of the geometry parameterization and aerodynamic models being differentiated by the finite difference method, and the acoustic model using an automatic differentiation library. The test case consisted of a single proprotor operating in a Mach 0.11 freestream, representative of a cruise condition, with and without an acoustic constraint. A wing was eventually added immediately downstream of the proprotor rotation plane, to mimic installation effects. Each optimization case achieved feasibility and made significant improvements in the objective function, but the optimality criterion was not satisfied. The addition of constraints on the chord and twist concavity was found to be necessary to produce realistic-looking blade designs—the reason for this is unknown and will be the subject of further study.

Acknowledgments

We thank Tim Brooks of NASA Glenn/BQMI for help with VSPAERO and pygeo, and Eliot Aretskin-Hariton of NASA Glenn for reviewing an early form of this paper. We are grateful for NASA's Transformational Tools and Technologies project, part of the Transformative Aeronautics Concepts Program, for funding this work.

References

[1] Rizzi, S. A., Huff, D. L., Boyd, D. D., Jr., Bent, P., Henderson, B. S., Pascioni, K. A., Sargent, D. C., Josephson, D. L., Marson, M., He, H. B., and Snider, R., "Urban Air Mobility Noise: Current Practice, Gaps, and Recommendations," Tech. Rep. NASA/TP-2020-5007433, NASA Langley Research Center, Hampton, VA, United States, October 2020.

[2] Içke, R. Ö., Baysal, O., Lopes, L. V., and Diskin, B., "Optimizing Proprotor Blades Using Coupled Aeroacoustic and Aerodynamic Sensitivities," *AIAA AVIATION 2021 FORUM*, American Institute of Aeronautics and Astronautics, 2021. https://doi.org/10.2514/6.2021-3037, URL http://dx.doi.org/10.2514/6.2021-3037.

[3] Ingraham, D., Gray, J. S., and Lopes, L. V., "Gradient-Based Propeller Optimization with Acoustic Constraints," *AIAA Scitech 2019 Forum*, American Institute of Aeronautics and Astronautics, 2019. https://doi.org/10.2514/6.2019-1219.

[4] Martins, J. R. R. A., and Ning, A., *Engineering Design Optimization*, Cambridge University Press, 2022.

[5] Katz, J., and Plotkin, A., *Low-speed aerodynamics*, 2nd ed., Cambridge university press, 2001.

[6] Drela, M., *Flight vehicle aerodynamics*, The MIT Press, Cambridge, Massachusetts, 2014.

[7] Chang, L. K., and Sullivan, J. P., "Optimization of Propeller Blade Twist by an Analytical Method," *AIAA Journal*, Vol. 22, No. 2, 1984, p. 252–255. https://doi.org/10.2514/3.48441, URL http://dx.doi.org/10.2514/3.48441.

[8] Miller, C. J., "Optimally Designed Propellers Constrained By Noise," Ph.D. thesis, Purdue University, 1984.

[9] Cho, J., and Lee, S.-c., "Propeller blade shape optimization for efficiency improvement," *Computers and Fluids*, Vol. 27, No. 3, 1998, p. 407–419. https://doi.org/10.1016/s0045-7930(97)00035-2, URL http://dx.doi.org/10.1016/S0045-7930(97)00035-2.

[10] Burger, C., Hartfield, R., and Burkhalter, J., "Propeller Performance Optimization Using Vortex Lattice Theory and a Genetic Algorithm," *44th AIAA Aerospace Sciences Meeting and Exhibit*, 2006. https://doi.org/10.2514/6.2006-1067, URL http://dx.doi.org/10.2514/6.2006-1067.

[11] Burger, C., Hartfield, R., and Burkhalter, J., "Performance and Noise Optimization of a Propeller using the Vortex Lattice Method and a Genetic Algorithm," *48th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, American Institute of Aeronautics and Astronautics, 2007. https://doi.org/10.2514/6.2007-1883, URL http://dx.doi.org/10.2514/6.2007-1883.

[12] Ingraham, D., "Low-Noise Propeller Design with the Vortex Lattice Method and Gradient-Based Optimization," *AIAA SCITECH 2023 Forum*, American Institute of Aeronautics and Astronautics, 2023. https://doi.org/10.2514/6.2023-2039, URL http://dx.doi.org/10.2514/6.2023-2039.

[13] Farassat, F., "Linear Acoustic Formulas for Calculation of Rotating Blade Noise," *AIAA Journal*, Vol. 19, No. 9, 1981, pp. 1122–1130. https://doi.org/10.2514/3.60051.

[14] Gray, J. S., Hearn, T. A., Moore, K. T., Hwang, J., Martins, J., and Ning, A., "Automatic Evaluation of Multidisciplinary Derivatives Using a Graph-Based Problem Formulation in OpenMDAO," *15th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, American Institute of Aeronautics and Astronautics, 2014. https://doi.org/doi:10.2514/6.2014-2042.

[15] 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.

[16] Gill, P., Murray, W., and Saunders, M., "SNOPT: An SQP Algorithm for Large-Scale Constrained Optimization," *SIAM Review*, Vol. 47, No. 1, 2005, pp. 99–131. https://doi.org/10.1137/S0036144504446096.

[17] Wu, N., Kenway, G., Mader, C. A., Jasa, J., and Martins, J. R. R. A., "pyOptSparse: A Python framework for large-scale constrained nonlinear optimization of sparse systems," *Journal of Open Source Software*, Vol. 5, No. 54, 2020, p. 2564. https://doi.org/10.21105/joss.02564.

[18] Lambe, A. B., and Martins, J. R. R. A., "Extensions to the Design Structure Matrix for the Description of Multidisciplinary Design, Analysis, and Optimization Processes," *Structural and Multidisciplinary Optimization*, Vol. 46, 2012, pp. 273–284. https://doi.org/10.1007/s00158-012-0763-y.

[19] McDonald, R. A., and Gloudemans, J. R., "Open Vehicle Sketch Pad: An Open Source Parametric Geometry and Analysis Tool for Conceptual Aircraft Design," *AIAA SCITECH 2022 Forum*, American Institute of Aeronautics and Astronautics, 2022. https://doi.org/10.2514/6.2022-0004, URL http://dx.doi.org/10.2514/6.2022-0004.

[20] Hajdik, H. M., Yildirim, A., Wu, N., Brelje, B. J., Seraj, S., Mangano, M., Anibal, J. L., Jonsson, E., Adler, E. J., Mader, C. A., Kenway, G. K. W., and Martins, J. R. R. A., "pyGeo: A geometry package for multidisciplinary design optimization," *Journal of Open Source Software*, Vol. 8, No. 87, 2023, p. 5319. https://doi.org/10.21105/joss.05319.

[21] Kinney, D. J., 2015. URL https://openvsp.org/wiki/doku.php?id=vspaerotutorial.

[22] Tannehill, J. C., Anderson, D. A., and Pletcher, R. H., *Computational Fluid Mechanics and Heat Transfer*, Taylor and Francis, 1997.

[23] Lambe, A. B., Kennedy, G. J., and Martins, J. R. R. A., "An evaluation of constraint aggregation strategies for wing box mass minimization," *Structural and Multidisciplinary Optimization*, Vol. 55, No. 1, 2016, p. 257–277. https://doi.org/10.1007/s00158-016-1495-1, URL http://dx.doi.org/10.1007/s00158-016-1495-1.

[24] Revels, J., Lubin, M., and Papamarkou, T., "Forward-Mode Automatic Differentiation in Julia," *arXiv:1607.07892 [cs.MS]*, 2016. URL https://arxiv.org/abs/1607.07892.

[25] Farassat, F., "Derivation of Formulations 1 and 1A of Farassat," Tech. Rep. NASA/TM 2007-214853, NASA Langley Research Center, Hampton, VA, United States, Mar 2007.

[26] Lopes, L. V., "Compact Assumption Applied to Monopole Term of Farassat's Formulations," *Journal of Aircraft*, Vol. 54, No. 5, 2017, p. 1649–1663. https://doi.org/10.2514/1.c034048.

[27] Lopes, L. V., and Ingraham, D. J., "Influence of the Perception Constraint, Observer Position, and Broadband Self-Noise on Low-Fidelity UAM Vehicle Perception-Influenced-Design (PID) Optimization," *Vertical Flight Society's 79th Annual Forum and Technology Display*, Vertical Flight Society, 2023.

[28] Akima, H., "A New Method of Interpolation and Smooth Curve Fitting Based on Local Procedures," *Journal of the ACM*, Vol. 17, No. 4, 1970, p. 589–602. https://doi.org/10.1145/321607.321609, URL http://dx.doi.org/10.1145/321607.321609.

[29] Zawodny, N. S., Pettingill, N. A., Lopes, L. V., and Ingraham, D. J., "Experimental Validation of an Acoustically and Aerodynamically Optimized UAM Proprotor Part 1: Test Setup and Results," Tech. Rep. NASA/TM–20220015637, NASA Langley Research Center, Hampton, VA, United States, February 2023.

[30] Andrew Ning, S., "A simple solution method for the blade element momentum equations with guaranteed convergence," *Wind Energy*, 2013, p. n/a–n/a. https://doi.org/10.1002/we.1636.

[31] Ning, A., "Using blade element momentum methods with gradient-based design optimization," *Structural and Multidisciplinary Optimization*, 2021. https://doi.org/10.1007/s00158-021-02883-6, URL http://dx.doi.org/10.1007/s00158-021-02883-6.

[32] "APC Downloads Page," , 2023. URL https://www.apcprop.com/technical-information/file-downloads/, accessed 2023-10-24.

[33] Pacini, B., Prajapati, M., Duraisamy, K., Martins, J. R., and He, P., "Towards Mixed-Fidelity Aero-Structural-Acoustic Optimization for Urban Air Mobility Vehicle Design," *AIAA AVIATION 2023 Forum*, American Institute of Aeronautics and Astronautics, 2023. https://doi.org/10.2514/6.2023-3905, URL http://dx.doi.org/10.2514/6.2023-3905.