



HAL
open science

Assessment of Methods for Propeller Performance Calculation at High Incidence Angles

Luiz F. T. Fernandez, Murat Bronz, Nathalie Bartoli, Thierry Lefebvre

► **To cite this version:**

Luiz F. T. Fernandez, Murat Bronz, Nathalie Bartoli, Thierry Lefebvre. Assessment of Methods for Propeller Performance Calculation at High Incidence Angles. AIAA SCITECH 2023, Jan 2023, NATIONAL HARBOR, United States. 10.2514/6.2023-2283 . hal-04006384

HAL Id: hal-04006384

<https://hal.science/hal-04006384>

Submitted on 27 Feb 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Assessment of Methods for Propeller Performance Calculation at High Incidence Angles

Luiz F. T. Fernandez¹

*ONERA/DTIS, Université de Toulouse, Toulouse, France
ENAC, Université de Toulouse, F-31400 Toulouse, France*

Murat Bronz²

ENAC, Université de Toulouse, F-31400 Toulouse, France

Thierry Lefebvre³ and Nathalie Bartoli⁴

ONERA/DTIS, Université de Toulouse, Toulouse, France

Accurately modelling propulsion systems forces and moments is of paramount importance for both design and control of aerial vehicles. With the increasing interest on hybrid vertical takeoff and landing UAVs, methods for propellers forces and moments calculation must be reliable throughout the entire flight envelope, including high incidence angles. In this work we evaluate six different existing methodologies for propeller thrust calculation in two different ways: wind tunnel tests are conducted to evaluate numerical differences, and we also use closed loop dynamical simulations in order to compare the obtained control commands, allowing for an estimation of the impact of such differences in a simulated flight. The methodologies are also compared with respect to applicability considering different steps of the design process. We obtained similar results for all methodologies as they show an offset but yet good prediction capabilities for high incidence angles.

Contents

I Introduction	2
II Methodologies	3
II.A Gill and D’Andrea method	4
II.B Davoudi and Duraisamy HBEM method	4
II.C Leng et al. method	5
II.D Rubin and Zhao	5
II.E CCBlade by Ning	6
II.F Summary of methodologies	6
III Propeller definition and test procedure	6
III.A Test bench	6
III.B Test procedure	8
III.B.1 No-load test and post processing	8
III.B.2 Propeller test and post processing	8
IV Implementation aspects	9
IV.A Overall strategy	9
IV.A.1 Gill and D’Andrea method 1	10
IV.A.2 Gill and D’Andrea method 2	11
IV.A.3 Davoudi and Duraisamy HBEM method	11

¹PhD Candidate, DTIS, luiz.tiberio@onera.fr, AIAA Student Member

²Assistant Professor, murat.bronz@enac.fr, AIAA Member

³Research Engineer, DTIS, thierry.lefebvre@onera.fr, AIAA Member

⁴Senior researcher, DTIS, nathalie.bartoli@onera.fr, AIAA MDO TC Member.

IV.A.4 Leng et al. method	11
IV.A.5 CCBlade by Ning	11
V Thrust results	11
VI Quadrotor vehicle simulation and flight experiments	12
VII Conclusions	13

Nomenclature			
Acronyms		C_Y	Force coefficient in y axis
EVTOL	Electric Vertical takeoff and landing vehicle	C_{d_0}	Propeller airfoil drag coefficient for zero α
MAV	Micro air vehicle	C_{d_α}	Propeller airfoil drag curve slope
MDO	Multidisciplinary design and optimization	C_{l_0}	Propeller airfoil lift coefficient for zero α
UAM	Urban air mobility	C_{l_α}	Propeller airfoil lift curve slope
UAV	unmanned aerial vehicle	C_{m_α}	Propeller airfoil moment curve slope
VTOL	Vertical takeoff and landing vehicle	C_{m_0}	Propeller airfoil moment coefficient for zero α
Symbols		c_{tip}	Propeller tip chord
β	Incidence Angle	F_t	Thrust Force
μ	advance ratio	F_y	Force in y axis
ω	Propeller angular velocity	M_x	Moment in x axis
ψ	Azimuth Angle	M_y	Moment in y axis
ρ	Air density	M_z	Moment in z axis
σ	Propeller solidity	N_b	Propeller number of blades
θ_{tip}	Twist angle at propeller tip	δ	Propeller span efficiency factor
C_M	Moment coefficient	λ	inflow ratio
C_P	Power coefficient	R	Propeller Radius
C_T	Thrust coefficient	V	Flight speed

I. Introduction

IN the last decade, unmanned aerial vehicles (UAVs) gained attention as they stopped being restricted to military applications and started to be used for several purposes. If from one side such kind of vehicle is now well established for leisure and cinematography, the number of possibilities regarding their application for the benefit of the environment and the mankind increases constantly. Recently, there has been studies regarding the use of drones to detect forest pathogens [1], to manage wildfires operating as swarms [2] and in a blockchain framework [3], to help in epidemic situations [4], and even to evaluate the potential of selecting an umbrella specie to conserve the marine Megafauna [5]. Winged vertical takeoff and landing (VTOL) UAVs stands out as they feature longer flight ranges, speeds and payload capacity, while maintaining the capability of taking off and landing without needing a long runway. Tail-sitters and tilt-wings are examples of VTOL configurations, similar in the sense that both wing and propulsion system must operate at high angles of incidence. In such condition, the well known relation $T = k_t \times \omega^2$ to calculate thrust as a function of a coefficient and the angular velocity is no longer valid. They et al [6] conducted wind tunnel tests to show how the incidence angle affects propeller forces and moments. Serrano et al [7] also tested with four different propellers, varying

advance ratio and incidence angles. From the aerodynamic modeling perspective, there has been a considerable effort to accurately account for this phenomenon. Gill and D’Andrea [8] proposed two low-order first-principle-based parametric approaches. Davoudi and Duraisamy [9] and Leng et al [10] presented blade element momentum based methodologies. Rubin and Zhao [11] expanded the actuator disk model thrust formula to obtain a methodology that uses axial thrust to obtain the force at different incidence angles, without using any geometric information. Bauersfeld et al [12] proposed an hybrid approach, using both blade element theory and a learning based method to predict aerodynamic thrust as torques, as well as parasitic effects, to achieve high modeling accuracy suitable even for very aggressive drone flights. Other set of contributions can be found with the UAM and wind turbine design and operations communities, as they are also interested in similar problems. Simmons [13, 14] conducted wind tunnel tests and used system identification techniques to model propellers at incidence for a subscale electric vertical takeoff and landing (EVTOL) vehicle in the context of urban air mobility (UAM). Ning [15] presented a blade element momentum method formulation that is specially suited for gradient based optimization, whose initial version [16] was used by Moore and Ning [17] to evaluate the effect of distributed electric propulsion on traditional aircraft and by Hendricks et al. [18] to design a Turboelectric Tiltwing UAM vehicle using multidisciplinary design and optimization (MDO), and by Ning and Petch [19] to design downwind land-based wind turbines.

Our goal in this paper is to understand the pros and cons of the available methods for the characterization of propellers at incidence. We compare the methodologies proposed by Gill and D’Andrea [8], Leng et al [10], Davoudi and Duraisamy [9], Rubin and Zhao [11], and the CCBlade by Ning [15]. Each method uses a different formulation, needs different inputs and is therefore more suitable to different design and operation stages or objectives. Gill and D’Andrea’s [8] original methods is based on wind tunnel measurements, making it more attractive for advanced design or operation phases and specially control law development, when the propeller is already defined and available for tests. The methods proposed by Rubin and Zhao [11] and Leng’s [10] relies on the axial performance data, whereas the latter also needs chord and pitch angle as inputs. Davoudi and Duraisamy HBEM method [9] and the CCBlade by Ning [15] are fully computational and need propeller geometry information, as the latter also uses airfoil polar as inputs. We tackle this problem in two different ways:

- Pure numerical approach: we directly compare the values of thrust obtained for each one of the methods with wind tunnel test results.
- Applied analysis: we set a simulation environment for a quadcopter vehicle and evaluate the impact of changing thrust and torque calculation on the final simulated trajectory and control commands. This reveals the physical significance of each method’s numerical differences in realistic simulation.

Throughout this paper, we are mostly interested in the level of accordance of each methodology with experimental results, as well as the possible effect of differences in a real system. Pure experimental approaches were not taken into account, as we plan to use our study for design and optimization purposes.

This paper is structured as follows. Section II briefly describes the methodologies under comparison and in Section III we discuss the propeller used for this evaluation. Section III.A presents the wind tunnel testing procedure and some implementation aspects are discussed in Section IV. In Section V we show the results obtained using the wind tunnel evaluations, and then we discuss the methods for dynamic simulation and obtained results in Section VI. We show our conclusions and perspectives in Section VII.

II. Methodologies

Throughout this section we consider that forces and moments generated by a propeller with radius R are functions of the incoming wind speed V , angular velocity ω , and angle between the rotor plane and incoming wind β . Forces and moments can be calculated using dimensionless coefficients and dynamic pressure:

$$\begin{bmatrix} F_t \\ F_y \\ M_x \\ M_y \\ M_z \end{bmatrix} = \begin{bmatrix} C_t(V, \omega, \beta)0.5\rho\pi R^4\omega^2 \\ C_y(V, \omega, \beta)0.5\rho\pi R^4\omega^2 \\ C_{M_x}(V, \omega, \beta)0.5\rho\pi R^5\omega^2 \\ C_{M_y}(V, \omega, \beta)0.5\rho\pi R^5\omega^2 \\ C_{M_z}(V, \omega, \beta)0.5\rho\pi R^5\omega^2 \end{bmatrix} \quad (1)$$

The advance ratio μ , inflow ratio λ , and propeller solidity σ are defined as:

$$\mu = \frac{V \sin(\beta)}{\omega R} \quad \lambda = \frac{V \cos(\beta)}{\omega R} \quad \sigma = \frac{N_b c_{tip}}{\pi R} \quad (2)$$

A. Gill and D'Andrea method

With the objective of reducing computational cost and required inputs for control applications, the authors [8] reduced the calculation of dimensionless coefficients to:

$$C_T = \frac{\sigma}{2\delta} \left[(1 - \delta)(C_{l_0}\delta(1 + \delta) - 2C_{l_\alpha}\delta(\lambda - \theta_{tip}) + C_{l_\alpha}\theta_{tip}\mu^2) - C_{l_0}\delta\mu^2 \log(\delta) \right] \quad (3)$$

$$C_y = \frac{\mu\sigma}{2\delta} \left[(1 - \delta)(2C_{d_0}\delta + \theta_{tip}((C_{l_\alpha} - 2C_{d_\alpha})\lambda + 2C_{d_\alpha}\theta_{tip})) - C_{l_0}\delta\lambda \log(\delta) \right] \quad (4)$$

$$C_{M_x} = \left[2C_{d_0}(1 + \delta + \delta^2) + 3C_{l_0}(\delta + 1)\lambda + 6(C_{d_\alpha}(\lambda - \theta_{tip}) - C_{l_\alpha}\lambda)(\lambda - \theta_{tip}) + \frac{3\mu^2(C_{d_0}\delta + C_{d_\alpha}\theta_{tip}^2)}{\delta} \right] \frac{\sigma(1 - \delta)}{6} \quad (5)$$

$$C_{M_y} = \frac{\sigma\mu(1 - \delta)}{2} \left[C_{l_0}(1 + \delta) - C_{l_\alpha}(\lambda - 2\theta_{tip}) \right] \quad (6)$$

$$C_{M_z} = \frac{\sigma\mu c_{tip}}{2\delta R} \left[C_{m_\alpha}(\delta - 1)(\lambda - 2\theta_{tip}) - 2C_{m_0}\delta \log(\delta) \right] \quad (7)$$

The inputs for such equations, the x array as named by the authors, are:

$$x = (C_{l_0}, C_{l_\alpha}, C_{d_0}, C_{d_\alpha}, C_{m_0}, C_{m_\alpha}, \delta, \theta_{tip}, c_{tip}) \quad (8)$$

Despite being possible to obtain the majority of these values from airfoil (if known) data or simulation, the authors chose to use a fully experimental approach. They conducted wind tunnel tests for different propellers obtained x using supervised and unsupervised learning approaches. Here, we will focus only on the supervised strategy. They also proposed a yet reduced order model, based on a second-order Taylor series expansion:

$$C_T = C_{T_{static}} + k_1\lambda + k_2\mu^2 + k_3\lambda^2 \quad (9)$$

$$C_y = k_4\mu + k_5\lambda\mu \quad (10)$$

$$C_{M_x} = C_{M_{x_{static}}} + k_6\lambda + k_7\mu^2 + k_8\lambda^2 \quad (11)$$

$$C_{M_y} = k_9\mu + k_{10}\lambda\mu \quad (12)$$

$$C_{M_z} = k_{11}\mu + k_{12}\lambda\mu \quad (13)$$

where the coefficients $C_{T_{static}}$, $C_{M_{x_{static}}}$, and k_1 to k_{12} can also be obtained using wind tunnel data and supervised or unsupervised learning. In this methodology, the inflow velocity is assumed to be function of the radius only, and not of the azimuth angles, which holds only for small angles. As pointed by Davoudi and Duraisamy [9], the impact of such simplification is hard to evaluate.

B. Davoudi and Duraisamy HBEM method

The authors [9] presented a methodology called Hybrid blade element momentum (*HBEM*). They show that thrust has different trends in blade element (BE) and momentum theories. According to BE, a larger inflow ratio leads to a lower thrust, whereas a larger thrust coefficient results in a higher inflow ratio in momentum theory. BEM based methods, including theirs, consists in finding thrust values that satisfy both momentum and blade element theories. Their algorithm can be summarized as:

- 1) An initial guess for thrust T is made
- 2) Calculate initial thrust coefficient using momentum theory C_T

- 3) Calculate the exact advance and climb ratio (λ_c and μ_c) given flight condition
- 4) Find uniform inflow ratio λ_0 from momentum theory
- 5) Use λ_0 to compute thrust coefficient from blade element theory C_{TBET}
- 6) Define $|C_{TMOMENTUM} - C_{TBET}|$ as an objective function to minimize
- 7) Iterate from step 2 until convergence

At the convergence point, the rolling and pitching moments can then be obtained as:

$$C_{M_x} = \frac{N_b}{4R^2\pi^2} \int_{r_{min}}^1 \int_0^{2\pi} -y C_l u^2 c(r) \cos(\Phi) d\psi dr \quad (14)$$

$$C_{M_y} = \frac{N_b}{4R^2\pi^2} \int_{r_{min}}^1 \int_0^{2\pi} x C_l u^2 c(r) \cos(\Phi) d\psi dr \quad (15)$$

where $c(r)$ is the chord distribution, Φ is the inflow angle, and u the effective net incident velocity on the blade. x and y are functions of the azimuth ψ angle and span location r , as defined below.

$$x(r, \Phi) = -rR \cos(\psi) \quad (16)$$

$$y(r, \Phi) = -rR \sin(\psi) \quad (17)$$

The authors published the original code ¹.

C. Leng et al. method

Also based on the same blade element and momentum theories, the methodology presented in [10] uses as inputs the chord and pitch angle distributions, and the propeller performance in axial condition for different advance ratios. The thrust and power coefficients are approximated as:

$$C_T(\mu, \lambda) = C_T(0, \lambda) \eta_T \quad (18)$$

$$C_P(\mu, \lambda) = C_P(0, \lambda) \eta_P \quad (19)$$

where $C_T(0, \lambda)$ and $C_P(0, \lambda)$ are the values for axial performance and η_T and η_P are thrust and power ratio, respectively, calculated according to:

$$\eta_T = 1 + \frac{(\mu \bar{r}')^2}{2(1 - \lambda_c / \lambda_{\infty 0T})} \quad (20)$$

$$\eta_P = 1 + \frac{(\mu \bar{r}')^2}{2(1 - \lambda_c / \lambda_{\infty 0P})} \quad (21)$$

where \bar{r}' is the position of the representative section in percentage radius (generally 75%, according to the authors), and $\lambda_{\infty 0T}$ and $\lambda_{\infty 0P}$ are models built using interpolation from axial performance data of zero thrust and power tip-speed ratio. Following the procedure, the normal force and in-plane moment coefficients are:

$$C_y = \frac{2\lambda_{\infty 0P} - \lambda_{\infty} \cos(\alpha_p)}{2\lambda_{\infty 0P} - \lambda_{\infty}} \sin(\alpha_p) \left. \frac{\partial C_N}{\partial \alpha_p} \right|_{\alpha_p=0, \lambda_{\infty}} \quad (22)$$

$$C_{M_x} = \frac{2\lambda_{\infty 0T} - \lambda_{\infty} \cos(\alpha_p)}{2\lambda_{\infty 0T} - \lambda_{\infty}} \sin(\alpha_p) \left. \frac{\partial C_n}{\partial \alpha_p} \right|_{\alpha_p=0, \lambda_{\infty}} \quad (23)$$

where the normal force and in-plane moment gradient around zero incidence can be calculated as functions of geometry and flight conditions.

D. Rubin and Zhao

In [11] the authors started from the classical momentum theory and Glauert's hypothesis to obtain a thrust model decomposed in two parts, the axial and the so called T_{wing} , sensitive to the parallel wind component to the rotor plane:

$$T = T_{axial} \left\{ 1 + \frac{(\sin \beta)^2}{\left[\cos \beta + \left(\frac{v_i}{V} \right) \right] \left[\sqrt{1 + 2 \left(\frac{v_i}{V} \right) \cos \beta + \left(\frac{v_i}{V} \right)^2} + \cos \beta + \left(\frac{v_i}{V} \right) \right]} \right\} \quad (24)$$

where v_i is the propeller average induced velocity at rotor disk.

¹https://github.com/behdad2018/HBEM_aero_flightsim_2019

Table 1 Summary of methodologies

Method	Inputs	Outputs
Gill and D’Andrea [8]	Wind tunnel testing data to estimate ($C_{l_0}, C_{l_\alpha}, C_{d_0}, C_{d_\alpha}, C_{m_0}, C_{m_\alpha}, \delta, \theta_{tip}, c_{tip}$)	$C_T, C_y, C_{M_x}, C_{M_y}, C_{M_z}$
Leng et al [10]	Radius, twist, and chord distribution and propeller axial performance	C_T, C_y, C_{M_x}
Davoudi and Duraisamy [9]	Radius, twist, and chord distribution	$C_T, C_{M_x}, C_{M_y}, C_{M_z}$
Rubin and Zhao [11]	propeller axial performance	C_T
CCBlade by Ning [15]	Radius, twist, and chord distribution, airfoil polars	C_T, C_{M_x}

E. CCBlade by Ning

Also based in blade element momentum theory, CCBlade [15] differs from the other similar methods as it is specifically thought to be used as a propeller design and optimization tool, having robust convergence. Its formulation and way of solving the residuals (similar to Davoudi and Duraisamy algorithm - Step 6) also enables derivatives calculation and the usage of gradient based optimizers. The current CCBlade methodology and implementation [15] present improvements to the original one presented by Ning [16]. It is an open source tool implemented in Julia with available documentation and tutorials².

F. Summary of methodologies

Table 1 shows a summary of inputs and outputs for each one of the evaluated methods. It can be noticed that they are different from each other in terms of inputs, which might make each one more attractive to a different design stage.

III. Propeller definition and test procedure

While there are endless options of motors and propellers for small UAVs in the market, in order to perform a fair benchmark between all the propellers we need to define one that geometry and airfoil are fully known, so all the methods can be applied without adding extra uncertainty sources. As this kind of precise information usually can not be found for commercial propellers, we chose to define one. In order to mimic an existing and already validated geometry, for chord and twist distribution we adopted values from APC6x4 according to Qblade [20], as shown in Table 9. The airfoil NACA4412 was also chosen for simplicity throughout all the spanwise sections with the exception the junction between blade and hub, where NACA4435 was used to ensure enough thickness and increase rigidity. The propeller was manufactured using the 3D printer Form 3+, from FORMLABS³. Figures 1a and 1b show the propeller right after the printing process, and the propeller after sanding and static balancing.

A. Test bench

The experiments were performed in the subsonic wind-tunnel located at ENAC’s (French Civil Aviation University) flight arena in Toulouse, France. The wind tunnel, designed by WindShape⁴, has a 1.5 m by 0.75 m open test section and is used both for research and educational purposes. In order to handle previous problems that were identified in other experiments conducted by Condomines et al. [21], the structure is designed to be as rigid and simple as possible, minimizing the distance between sensor and motor and undesirable aerodynamic effects on the measurements. The wind-tunnel speed is measured by a differential pressure sensor from SensorTechnics (LBAS 500U) which is calibrated previously by using a hot-wire anemometer. Force and moments are measured with a six-axis ATI Nano-17-E sensor⁵ calibrated with SI-25-0.25, with maximum force range of 25N for F_{xy} , 35N for F_z and 1/160 N resolution. The moment range is 250Nmm for M_{xyz} with a 1/32 Nmm resolution. As shown in Fig. 2a, motor and propeller are mounted to a

²<https://github.com/byuflowlab/CCBlade.jl> accessed in 05/06/2022

³<https://formlabs.com/eu/3d-printers/form-3/> accessed in 03/17/2022

⁴<https://windshape.com/technology/>

⁵https://www.ati-ia.com/products/ft/ft_models.aspx?id=Nano17 accessed in 05/06/2022



(a) After printing process



(b) Ready for test

Fig. 1 Propeller geometry



(a) Complete setup



(b) Rotation arm

Fig. 2 Test bench

small transition block, which is directly mounted on the Nano-17 sensor inside a "pylone-like" rotation arm, employed to house all the cables and the speed controller, reducing the drag of the whole structure. The rotating arm is manufactured by rapid prototyping of Onyx and continuous carbon fiber filaments via Markforged M2 printer⁶. This structure is connected to a robotic actuator by Hebi robotics⁷, which controls the rotation of the system and also outputs a feedback on the angular rate and position information. Finally, the actuator is fixed on a wooden plate placed in the wind tunnel test section. The speed controller used during the experiments was ESC32v3⁸. The main reason is that it can directly output the motor voltage, current, and RPM via a serial link while simultaneously controlling the RPM or throttle of the motor. For the data acquisition, the ATI Nano-17 sensor outputs are amplified with the original force transducer. A LABJACK T7⁹ board, connected to the operation desktop computer via USB, is used to convert the analog signals in differential mode with a 16-bit resolution by using 13-channels (6 sensor outputs, 6 individual reference grounds, and 1 common analog ground), data is recorded at 100Hz, and each signal channel settling time is set to 100 μ s.

This configuration for the test bench allowed us to minimize the number of sensors while ensuring reliability and full automation during test execution, as a single python script gathers speed controller telemetry (RPM, current, voltage) and Hebi actuator information. Table 2 summarizes the utilized instruments.

⁶<https://markforged.com/mark-two/> accessed in 05/06/2022

⁷hebirobotics.com accessed in 05/06/2022

⁸www.autoquad.org/esc32/ accessed in 05/06/2022

⁹<https://labjack.com/products/t7>

Table 2 Instruments used on the test bench.

Force/Moment Sensor	ATI Nano-17-E
Data acquisition Board	LabJack T7
ESC	Autoquad ESC32v3
Motor	T-Motor F-60
Actuator	HEBI X-5
Differential Pressure	SensorTechnics LBAS 500U

Table 3 No-load test procedure

Incidence angles [degrees]	Wind Tunnel throttles [%]
4 points in [-10,45] and 5 points in [45,90]	[25, 45, 65, 85, 90]

B. Test procedure

Even though it was designed to minimize aerodynamic effects, the rotation arm structure, shown in Fig. 2b, can influence the results as it adds undesirable forces and moments. In order to remove this effect, a no-load test was executed before every batch of propeller test. The no-load test consists in measuring forces and moments from the rotation arm without the insertion of the propellers for different incidence angles and wind speeds. The forces and moments obtained with the propeller could then be correctly obtained by subtracting the no-load result from the measurements. A total of seven repetitions were executed in different days and atmospheric conditions. For every test day, the no-load test case was executed immediately before the propeller experiment.

1. No-load test and post processing

Each execution was conducted for the set of incidence angles and wind tunnel throttle shown in Table 3, where 90% throttle represents about 9m/s wind speed, depending on the atmospheric conditions. The no-load test result post processing consisted in filtering the data using a low-pass filter, and then correcting it using the wind tunnel velocity model, as ensuring the same wind tunnel throttle does not imply in the same wind speed for different runs, as atmospheric conditions vary. In order to extend the no-load test result for different angles and velocities, six models were created, each of them for one force or moment, as a function of wind speed, incidence angle, and air density. The models were created using the polynomial feature from Sklearn [22], and were evaluated according to the mean squared error (MSE), that is zero for the perfect fit, and R^2 , that ideally is equal to one. Both measures are defined as:

$$R^2 = \frac{\sum_{i=1}^n (y_i - \bar{y})^2}{\sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad \text{and} \quad MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 \quad (25)$$

where n is the number of samples, y_i the observed value, \bar{y} the mean of the observations, and \hat{y}_i the estimated value for y_i . Table 4 shows the degree and size of training and test set employed. Figures 3a and 3b show the evaluation of thrust and torque models with test data (others forces and moments were omitted for the sake of clarity). They were trained with the data right after filtering, meaning that no unit transformation was applied initially.

Table 4 No-load polynomial model characteristics

Degree	3
Size of training samples	32
Size of test samples	8

2. Propeller test and post processing

The same strategy from the no-load test was adopted for the propeller in terms of filtering and output model training. Table 5 shows the different incidence angles, wind shape throttle, and motor throttle tested. Similarly to the wind shape

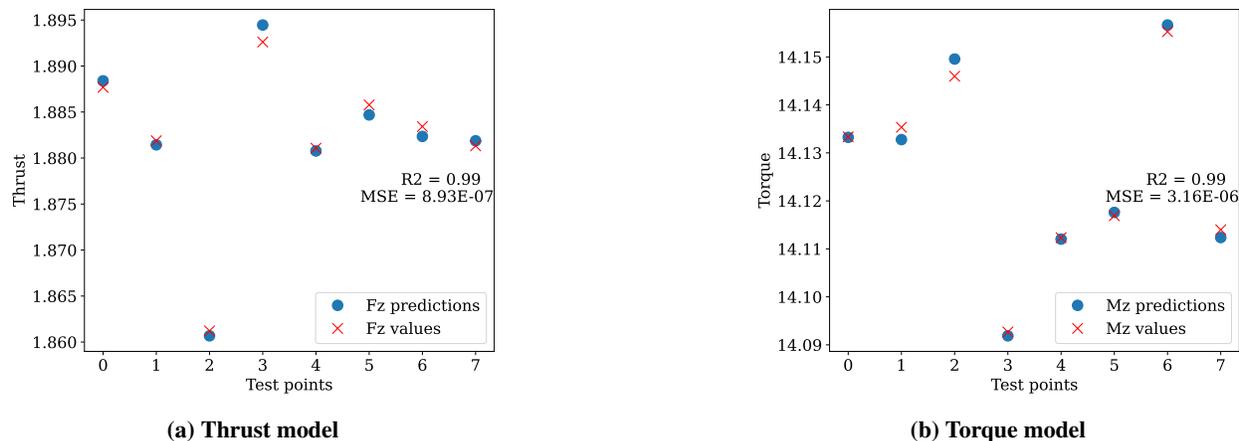


Fig. 3 Model assessment with test data

Table 5 Propeller test points

Incidence angles [degrees]	Wind Tunnel throttles [%]	Motor throttles [%]
6 points in [-10,45] and 7 points in [45,90]	[25, 45, 65, 85, 90]	[20, 30, 40, 50, 60]

throttle, the same motor throttle is not capable of ensuring the same propeller rpm and thus thrust, as it depends on several factors, being the battery charge the most important one. So, in order to compare results without such influence, we created polynomial models where the output thrust coefficient is a function of incidence angle, angular velocity, and wind speed. We opted to create one model per run, with the features described in Table 6. Similarly to the no-load case,

Table 6 Thrust polynomial model characteristics

Degree	3
Size of training sample	208
Size of test sample	52

we checked the fitting quality indicators of each model regarding each run in terms of MSE and R^2 . Table 7 shows the obtained metrics, which were considered to be acceptable.

Figures 4a and 4b show the obtained models, one for each run, for different values of wind speed and angular velocity. From these figures it is possible to observe that the quality of the models are affected by the inputs, showing more or less variability depending on the case. Such variations could be caused by several sources of uncertainty during the test and post processing, specially the velocity measurement, no-load test and modeling, and thrust measurement and modeling. From now on, only the average of the seven models is used, minimizing the effect of such variations.

IV. Implementation aspects

Having the models created using the experimental results, we were able to compare the different methodologies regarding their similarity with such baseline. We begin by first explaining how each methodology was implemented and used, to then compare the results for thrust prediction.

A. Overall strategy

In order to ensure a fair comparison between all the methodologies, we chose to use the original codes published by the authors, when published. This choice minimizes the possibility of inserting errors in the calculation, ensuring that only minor changes will be made in the available codes in order to make it compatible with our test environment. Table 8 shows the implementation aspects for each proposal. As we had to implement the method proposed by Gill and

Table 7 Quality of the fitting for thrust coefficient

Run	name	R^2	MSE
01	day 1 run 1	0.91	2.92e-06
02	day 1 run 1	0.84	4.16e-06
03	day 1 run 3	0.95	2.22e-06
04	day 2 run 1	0.86	8.45e-06
05	day 2 run 2	0.98	2.17e-06
06	day 3 run 1	0.99	3.04e-06
07	day 3 run 2	0.97	4.06e-06

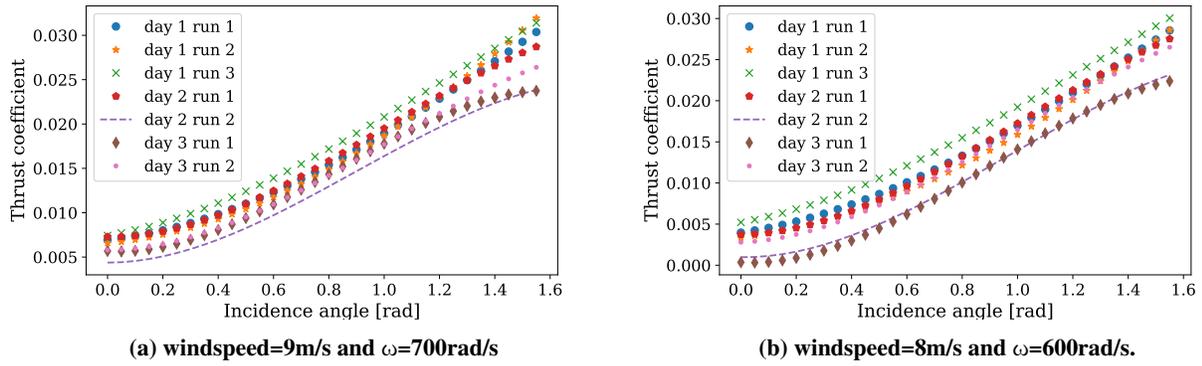


Fig. 4 Thrust coefficient experimental based models

D’Andrea, we used their open source data¹⁰ to validate our implementation.

Table 8 Implementation aspects

Method	Implementation	Programming language
Gill and D’Andrea [8]	Code not available, implemented for this paper	Python
Leng et al [10]	Not open source, but provided by the authors	Python
Davoudi and Duraisamy [9]	Open source	Matlab
CCBlade by Ning [15]	Open source	Julia

The main drawback of this choice is that it makes impossible to compare computational cost, as different implementation strategies programming languages were used. However none of the methods needs more than a few seconds to converge in a general use personal computer. So we choose not to analyze this now.

1. Gill and D’Andrea method 1

Initially, we used the same approach proposed by the authors to obtain the necessary inputs for his method. Such strategy consists in finding the set of parameters x from Eq. (8) using the experimental data. An optimization problem was defined as the minimization of the sum of the $RMSE$ for all forces and moments, defined as:

$$RMSE(x) = \sqrt{\frac{1}{N_m} \sum_{i=1}^{N_m} (C_y^x(\bar{\lambda}_{ci}, \bar{\mu}_i) - \bar{C}_{y,i})^2} \quad (26)$$

¹⁰<https://www.mdpi.com/2504-446X/3/4/77>

where N_m is the number of experimental points, $\bar{C}_{y,i}$, $\bar{\lambda}_{ci}$, and $\bar{\mu}_i$ are the measured force or moment coefficient, measured inflow ratio, and measure advance ratio respectively. C_y^x is the force or moment coefficient calculated for a given $\bar{\lambda}_{ci}$ and $\bar{\mu}_i$. An algorithm was used to interactively vary the parameters in x to minimize the overall fitting error, when comparing measured and calculated coefficients. To this optimization task, an evolutionary algorithm driver, similar to the one used by the authors, from OpenMDAO [23] was employed. All the equations needed for the application of this methodology were coded in Python. As this method calculates forces and moments using explicit equations, no convergence issues were found during the tests. From now on, this algorithm will be addressed as *Gill experimental*.

2. Gill and D'Andrea method 2

By studying the parameters in x presented in Eq. (8), one can observe that six of them depend on the airfoil, two derive from propeller geometry (chord and twist angle at the tip), and only one needs to be obtained via tests, which is the parameter δ that indicates the percentage of the span in which thrust starts to be generated. As eight of the nine values can be obtained analytically or depend on the geometry, we decided to use this method with the "theoretical" inputs, and not only the learning based obtained one. This was mainly done because the original proposed strategy is fully data based, which means that non-physical values of x could be obtained during the fitting process, giving the wrong idea that the set of equations are working. For this case, the span efficiency was considered to be 0.15, and the airfoil data were obtained using Xfoil [24] simulations. From now on, this algorithm will be addressed as *Gill analytical*.

3. Davoudi and Duraisamy HBEM method

In order to be compatible with our test environment, a few changes into the original code were necessary, but the Matlab implementation was kept to reduce intrusive changes into the original code. Throughout the execution of the tests, some convergence issues occurred, demanding minor changes into the original code, that were able to partially solve the problem. Such convergence problems were more common on the edges of the test envelope, specially high wind speeds and low propeller rpm.

4. Leng et al. method

The code provided by the authors of [10] was already implemented in Python, so very little effort was needed to integrate such methodology with our environment. Our experimental models were used to generate the axial performance inputs needed for this methodology. No convergence issues were observed.

5. CCBlade by Ning

A Julia wrapper was written to integrate this tool with our environment. The geometry file was generated directly with propeller information, the airfoil input file was generated with the help of the specific tutorial ¹¹. No convergence issues were observed.

V. Thrust results

Figures 5a to 7b show the obtained results for different values of wind speed and angular velocity.

From the results, it is possible to observe that all the methodologies are capable of calculating thrust with reasonable similarity to the experimental result. Even if an offset can be observed, the tendency with incidence angle variation is captured in a similar way; however, some specificities could be observed. Both implementations of Gill and D'Andrea's method *Gill analytical* (red line) and *Gill experimental* (brown line) have shown good agreement with the experimental results (blue stars). Even though the experimental approach seems to be the best fit, its use relies on the usage of experimental (or simulated) data for the training, making this option less attractive for initial design phases. On the other hand, the analytical approach also shows to reasonably agree with the experimental results, at a very lower experimental cost. The *HBEM* method (green line) also showed a good agreement with the experimental results. It usually has a small offset, but it is also capable of following the trend as the incidence angle increases. The convergence issues found throughout the execution might indicate that a more robust implementation might be needed before drawing conclusions about its precision. Leng's method (purple line) shows good agreement for small incidence angles, but it seems to overestimate the stall behaviour, showing very large drops in thrust. Such phenomenon was not found by any

¹¹<https://flow.byu.edu/CCBlade.jl/stable/howto/#Airfoil-Data> accessed in 05/06/2022

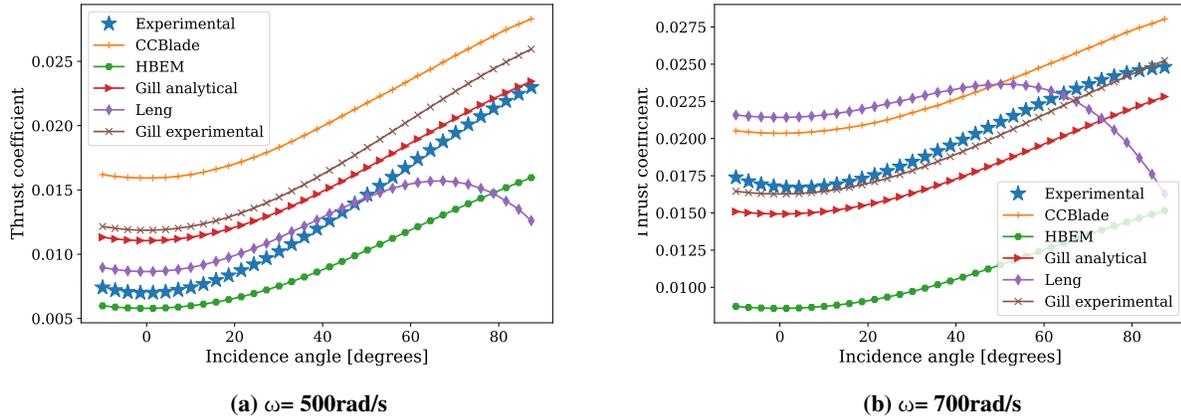


Fig. 5 Thrust coefficient for wind speed=5m/s

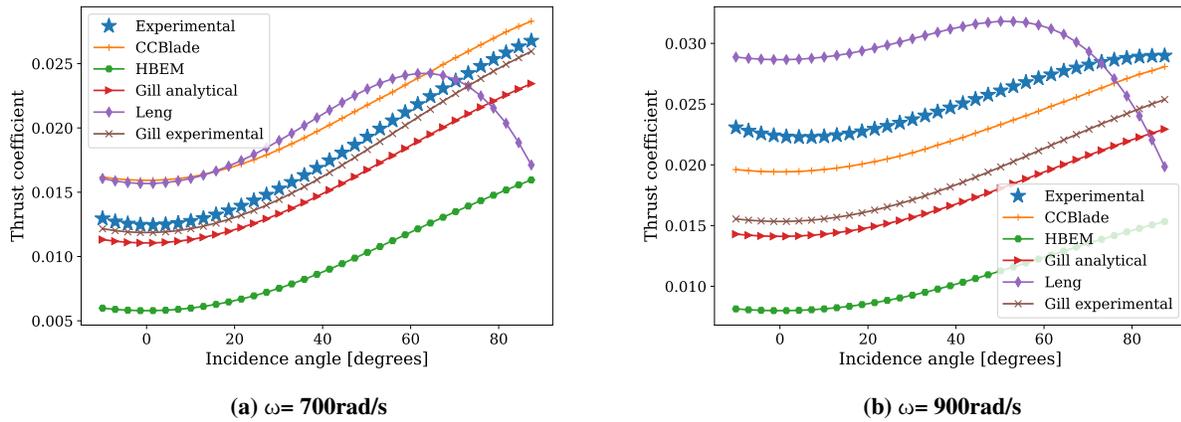


Fig. 6 Thrust coefficient for wind speed=7m/s

other method, and the experimental results only show very small drops. This behavior might come from the use of interpolations (and sometimes extrapolations), which might be adding numerical errors. CCBlade (orange line) has shown to be stable and precise, capable of following the incidence angle variation trends with precision. Even if an offset can be seen in the result, this behaviour might come from the fact that this tool uses airfoil polar analysis results. For this draft, we only used two airfoil polars (thus two Reynolds numbers). For the final paper, an automation between Reynolds number calculation for a given test velocity, followed by the corrected airfoil simulation, might probably lead to even better results.

For the final paper, we expect to draw more conclusions about each method pros and cons.

VI. Quadrotor vehicle simulation and flight experiments

Until now, we have been implying that the "best" method might be the one that is closer to the experimental result. But throughout this section, we intend to investigate if the numerical discrepancies we have found are actually physically meaningful.

In order to do that, we will use a quadrotor simulator, developed using PyBullet¹² physics engine. We also include INDI control laws, as presented by Smeur et al. [25] for both attitude and trajectory tracking. We intend to run simulations using the different methodologies for thrust calculation, and then compare the outcome with respect to final

¹²<https://pybullet.org/> accessed in 05/09/2022

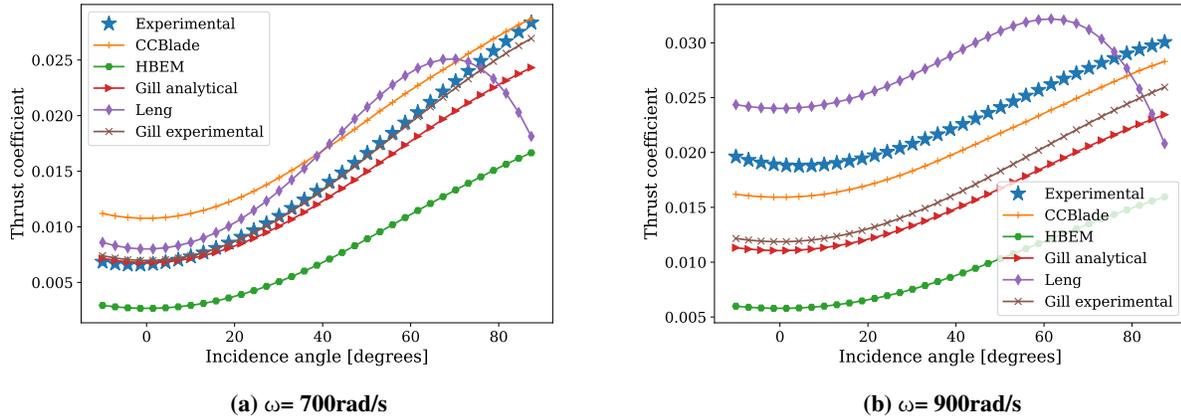


Fig. 7 Thrust coefficient for wind speed=9m/s

trajectory and control output.

The usage of a fully computationally environment also allows us to keep all the other simulation inputs constant, which in turn makes it possible to isolate the thrust calculation as the only factor that might generate different results throughout the test.

This section will be included in the final paper.

VII. Conclusions

We have presented, implemented, and compared four different methodologies for propeller thrust and moment calculation using a baseline open source propeller. We also wind tunnel tests conducted at ENAC's flight arena to support the comparison.

With respect to the application side, the low cost method developed by Gill and D'Andrea's with analytical inputs has shown to be very suitable for applications where very little information is known about the propeller and precision is not a problem (for control law testing, for instance). On the other hand, CCBlade has proven to be the best design and optimization tool, as it is capable of accurate calculating thrust and ensure not only stability but also small computational cost and gradient calculation.

For the final paper, we expect to include the overall method comparison, after also taking into account the simulation, which will allow us to conclude not only about numerical differences but also about their impact on more realistic scenarios.

Appendix

Table 9 shows the geometry of the propeller analyzed throughout this paper.

Acknowledgments

This work is part of the activities of ONERA - ISAE - ENAC joint research group.

Radius [m]	chord [m]	Twist angle [deg]
0.0045	0.013	20.0
0.0113	0.0135	35.0
0.0139	0.014	40.704
0.0165	0.0145	44.365
0.0191	0.015	40.704
0.0217	0.0155	36.939
0.0243	0.0152	34.154
0.0269	0.0150	31.404
0.0295	0.0148	28.881
0.0321	0.0146	26.719
0.0347	0.0143	24.867
0.0373	0.0140	23.270
0.0399	0.0136	21.886
0.0425	0.0132	20.676
0.0451	0.0127	19.600
0.0477	0.0122	18.626
0.0503	0.0116	17.731
0.0529	0.0111	16.892
0.0555	0.0106	16.104
0.0581	0.0101	15.371
0.0607	0.0095	14.696
0.0633	0.0089	14.078
0.0659	0.0083	13.508
0.0685	0.0076	12.978
0.0711	0.0070	12.481
0.0737	0.0061	11.968
0.075	0.00305	11.968

Table 9 Propeller geometry

References

- [1] Perroy, R. L., Meier, P., Collier, E., Hughes, M. A., Brill, E., Sullivan, T., Baur, T., Buchmann, N., and Keith, L. M., “Aerial Branch Sampling to Detect Forest Pathogens,” *Drones*, Vol. 6, No. 10, 2022. <https://doi.org/10.3390/drones6100275>, URL <https://www.mdpi.com/2504-446X/6/10/275>.
- [2] Saffre, F., Hildmann, H., Karvonen, H., and Lind, T., “Monitoring and Cordoning Wildfires with an Autonomous Swarm of Unmanned Aerial Vehicles,” *Drones*, Vol. 6, No. 10, 2022. <https://doi.org/10.3390/drones6100301>, URL <https://www.mdpi.com/2504-446X/6/10/301>.
- [3] Mahmudnia, D., Arashpour, M., Bai, Y., and Feng, H., “Drones and Blockchain Integration to Manage Forest Fires in Remote Regions,” *Drones*, Vol. 6, No. 11, 2022. <https://doi.org/10.3390/drones6110331>, URL <https://www.mdpi.com/2504-446X/6/11/331>.
- [4] Restás, A., “Drone Applications Fighting COVID-19 Pandemic;Towards Good Practices,” *Drones*, Vol. 6, No. 1, 2022. <https://doi.org/10.3390/drones6010015>, URL <https://www.mdpi.com/2504-446X/6/1/15>.
- [5] Dickson, L. C. D., Negus, S. R. B., Eizaguirre, C., Katselidis, K. A., and Schofield, G., “Aerial Drone Surveys Reveal the Efficacy of a Protected Area Network for Marine Megafauna and the Value of Sea Turtles as Umbrella Species,” *Drones*, Vol. 6, No. 10, 2022. <https://doi.org/10.3390/drones6100291>, URL <https://www.mdpi.com/2504-446X/6/10/291>.

- [6] Theys, B., Dimitriadis, G., Hendrick, P., and De Schutter, J., "Influence of propeller configuration on propulsion system efficiency of multi-rotor Unmanned Aerial Vehicles," *2016 International Conference on Unmanned Aircraft Systems (ICUAS)*, 2016, pp. 195–201. <https://doi.org/10.1109/ICUAS.2016.7502520>.
- [7] Serrano, D., Ren, M., Qureshi, A. J., and Ghaemi, S., "Effect of disk angle-of-attack on aerodynamic performance of small propellers," *Aerospace Science and Technology*, Vol. 92, 2019, pp. 901–914. <https://doi.org/https://doi.org/10.1016/j.ast.2019.07.022>, URL <https://www.sciencedirect.com/science/article/pii/S1270963818319473>.
- [8] Gill, R., and D'Andrea, R., "Computationally Efficient Force and Moment Models for Propellers in UAV Forward Flight Applications," *Drones*, Vol. 3, No. 4, 2019. <https://doi.org/10.3390/drones3040077>, URL <https://www.mdpi.com/2504-446X/3/4/77>.
- [9] Davoudi, B., and Duraisamy, K., "A Hybrid Blade Element Momentum Model for Flight Simulation of Rotary Wing Unmanned Aerial Vehicles," *AIAA Aviation 2019 Forum*, 2019. <https://doi.org/10.2514/6.2019-2823>, URL <https://arc.aiaa.org/doi/abs/10.2514/6.2019-2823>.
- [10] Leng, Y., Jardin, T., Moschetta, J.-M., and Bronz, M., "Analytic Model of Proprotor Forces and Moments at High Incidence," *Journal of the American Helicopter Society*, Vol. 66, No. 4, 2021, pp. 1–15. <https://doi.org/10.4050/JAHS.66.042002>, URL <https://oatao.univ-toulouse.fr/28398/>.
- [11] Rubin, R. L., and Zhao, D., "New Development of Classical Actuator Disk Model for Propellers at Incidence," *AIAA Journal*, Vol. 59, No. 3, 2021, pp. 1040–1054. <https://doi.org/10.2514/1.J059734>, URL <https://doi.org/10.2514/1.J059734>.
- [12] Bauersfeld, L., Kaufmann, E., Foehn, P., Sun, S., and Scaramuzza, D., "NeuroBEM: Hybrid Aerodynamic Quadrotor Model," *Robotics: Science and Systems XVII*, 2021. <https://doi.org/10.15607/rss.2021.xvii.042>, URL <http://dx.doi.org/10.15607/RSS.2021.XVII.042>.
- [13] Simmons, B. M., "System Identification for Propellers at High Incidence Angles," *Journal of Aircraft*, Vol. 58, No. 6, 2021, pp. 1336–1350. <https://doi.org/10.2514/1.C036329>, URL <https://doi.org/10.2514/1.C036329>.
- [14] Simmons, B. M., and Hatke, D. B., "Investigation of High Incidence Angle Propeller Aerodynamics for Subscale eVTOL Aircraft," Tech. Rep. NASA/TM-20210014010, NASA, May 2021. URL <https://ntrs.nasa.gov/citations/20210014010>, nTRS Author Affiliations: Langley Research Center NTRS Document ID: 20210014010 NTRS Research Center: Langley Research Center (LaRC).
- [15] Ning, A., "Using blade element momentum methods with gradient-based design optimization," *Structural and Multidisciplinary Optimization*, Vol. 64, No. 2, 2021, pp. 991–1014. <https://doi.org/10.1007/s00158-021-02883-6>, URL <https://doi.org/10.1007/s00158-021-02883-6>.
- [16] Ning, S. A., "A simple solution method for the blade element momentum equations with guaranteed convergence," *Wind Energy*, Vol. 17, No. 9, 2014, pp. 1327–1345. <https://doi.org/10.1002/we.1636>, URL <https://onlinelibrary.wiley.com/doi/abs/10.1002/we.1636>, [_eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/we.1636](https://onlinelibrary.wiley.com/doi/pdf/10.1002/we.1636).
- [17] Moore, K. R., and Ning, A., "Distributed Electric Propulsion Effects on Existing Aircraft Through Multidisciplinary Optimization," *2018 AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, American Institute of Aeronautics and Astronautics, 2018. <https://doi.org/10.2514/6.2018-1652>, URL <https://arc.aiaa.org/doi/abs/10.2514/6.2018-1652>, [_eprint: https://arc.aiaa.org/doi/pdf/10.2514/6.2018-1652](https://arc.aiaa.org/doi/pdf/10.2514/6.2018-1652).
- [18] Hendricks, E. S., Falck, R. D., Gray, J. S., Aretskin-Hariton, E., Ingraham, D., Chapman, J. W., Schnulo, S. L., Chin, J., Jasa, J. P., and Bergeson, J. D., "Multidisciplinary Optimization of a Turboelectric Tiltwing Urban Air Mobility Aircraft," *AIAA Aviation 2019 Forum*, American Institute of Aeronautics and Astronautics, 2019. <https://doi.org/10.2514/6.2019-3551>, URL <https://arc.aiaa.org/doi/10.2514/6.2019-3551>.
- [19] Ning, A., and Petch, D., "Integrated design of downwind land-based wind turbines using analytic gradients," *Wind Energy*, Vol. 19, No. 12, 2016, pp. 2137–2152. <https://doi.org/https://doi.org/10.1002/we.1972>, URL <https://onlinelibrary.wiley.com/doi/abs/10.1002/we.1972>.
- [20] Drela, M., "QPROP Formulation," , 2006. URL https://web.mit.edu/drela/Public/web/qprop/qprop_theory.pdf.
- [21] Condomines, J.-P., Bronz, M., Hattenberger, G., and Erdelyi, J.-F., "Experimental Wind Field Estimation and Aircraft Identification," *Proceedings of the International Micro Air Vehicles Conference and Flight Competition*, 2015.

- [22] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E., “Scikit-learn: Machine Learning in Python,” *Journal of Machine Learning Research*, Vol. 12, 2011, pp. 2825–2830.
- [23] 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>.
- [24] Drela, M., “XFOIL: An Analysis and Design System for Low Reynolds Number Airfoils,” *Low Reynolds Number Aerodynamics*, edited by T. J. Mueller, Springer Berlin Heidelberg, Berlin, Heidelberg, 1989, pp. 1–12.
- [25] Smeur, E., de Croon, G., and Chu, Q., “Cascaded incremental nonlinear dynamic inversion for MAV disturbance rejection,” *Control Engineering Practice*, Vol. 73, 2018, pp. 79–90. <https://doi.org/https://doi.org/10.1016/j.conengprac.2018.01.003>, URL <https://www.sciencedirect.com/science/article/pii/S0967066118300030>.