Multirotor Trim using Loose Aerodynamic Coupling

Austin D. Thai
Graduate Student
Boston University
Boston, MA

Beatrice Roget
Senior Scientist
Science and Technology Corporation
Moffett Field, CA

Jay Sitaraman
Senior Scientist
Parallel Geometric Algorithms LLC
Sunnyvale, CA

Sheryl M. Grace
Associate Professor
Boston University
Boston, MA

ABSTRACT

A multirotor trim module is developed for the HPCMP CREATE^{TM}-AV Helios rotorcraft simulation code. Trimmed free-flight simulation results are presented for two multirotor configurations, using rotor frequencies and aircraft attitudes as the control variables. The loose-coupling procedure is used to achieve trim, where aerodynamic loading on the rotor blades and fuselage are computed using a simplified aerodynamic model, and modified at each coupling iteration using the airloads computed by the higher fidelity CFD based aerodynamics. Two different optimization methods are tested: a least-square regression algorithm, with the norm of the loads at the center of gravity as the objective function, and a nonlinear constrained optimization code, with the total power as the objective function, and with constraints applied to satisfy trim. First, a commercial small-scale UAV is simulated in forward flight. A reference model for mid-scale UAM applications is then trimmed in hover to demonstrate the module's ability to model and trim a complex configuration.

NOTATION

c	chord length, m
C_d	drag coefficient
C_l	lift coefficient
dD	drag on blade element, N
dF_y	y-force on blade element, N
dF_z	z-force on blade element, N
dL	lift on blade element, N
ds	blade element length, m
F^{CFD}	airloads from CFD, $N, N\text{-}m$
F^{LL}	airloads from simplified model, $N, N\text{-}m$
J	control Jacobian
k	coupling iteration
q	control variables vector, $rad/s, rad$
R	rotor radius, m
R_1	inertial to aircraft frame rotation matrix
R_2	aircraft to hub frame rotation matrix
R_3	hub to rotor frame rotation matrix
R_4	rotor to airfoil frame rotation matrix
s	spanwise coordinate, m
v_1	aircraft velocity vector, m/s
v_2	rotation velocity vector, m/s
v_3	inflow velocity vector, m/s
V_{air}	wind velocity vector, m/s
$V_{induced}$	induced velocity, m/s
V_{tot}	total velocity vector, m/s
ΔF	delta airloads, $N, N\text{-}m$
Γ	rotor speed vector, rad/s
ΔF	delta airloads, $N, N\text{-}m$
Λ	loads objective function
ρ	air density, kg/m^3
Ψ_{CFD}	loads on aircraft CG, $N, N\text{-}m$
Ψ_{target}	desired loads on aircraft CG, $N, N\text{-}m$
Ω	rotor speed, rad/s
Ω_{min}	smallest rotor speed, rad/s

Presented at the VFS Aeromechanics for Advanced Vertical Flight Technical Meeting, San Jose, CA, January 21—23, 2020.

INTRODUCTION

The burgeoning eVTOL (electric vertical take-off and landing) field, pushed by increased interest in unmanned aerial vehicles (UAV) and urban air mobility (UAM), creates a need for improved modeling capabilities. eVTOL aircraft, which are typically multirotor/propeller designs, are favored by the urban flight industry due to their unique advantage in decreasing the environmental impact because of the low-carbon emission motor-driven propulsion. A range of low-to-high-fidelity modeling tools are required to perform design optimization and mitigate risk before prototyping and flight testing of such aircraft.

It has been shown in traditional rotorcraft research that accurate airloads prediction requires fully-coupled trim calculations performed with high-fidelity computational fluid dynamics (CFD) (Ref. 1). Furthermore, for UAV and UAM ve-

hicles, the increased number of rotors and intricate fuselage designs give rise to complex aerodynamic interactions that can only be accurately captured using fully-coupled CFD calculations under trimmed conditions. However, multirotor eVTOL aircrafts, unlike a traditional rotorcraft, use individual rotor frequencies instead of rotor pitch to achieve trim. Thus, well-established comprehensive rotorcraft analysis methods cannot readily be used to find a trim solution, and a new, more flexible trim module is required. Additionally, because of the increased aerodynamic interactions between rotor wakes and fuselage components, it is important to fully account for the airloads on all components when computing the trim solution. Finally, for more complex configurations involving a large number of rotors, the number of control variables may exceed the six target loads at the center of gravity, and a robust optimization method is required to identify the optimal solution to this under-determined problem.

The HPCMP CREATE^{TM}-AV Helios is a software framework for rotorcraft aeromechanics simulations developed by the Army Aviation Development Directorate. It can provide high-fidelity and mid-fidelilty CFD solutions and has been used for trim calculations of traditional rotorcraft. In this paper, a new module is developed for Helios that is capable of computing rigid-body multirotor, frequency-based trim for coupled CFD calculations. The module uses a loose-coupling approach in which the rotor airloads are communicated from the CFD solution periodically at coupling iterations. The newly developed capability is demonstrated by performing full free-flight trim simulation of the DJI Phantom 3 quadcopter drone at multiple flight speeds. Following that, fully trimmed hover simulation of the more complex Uber eCRM-001 aircraft is performed.

BACKGROUND

There have been multiple computational studies of the aerodynamic properties of small scale quadrotors in hover (Refs. 2–5). The flow interactions in hover are very different from forward flight, due to the flow asymmetry on the advancing and retreating rotor sides as well as rotor-rotor wake interactions.

In the past, researchers have used lower-fidelity calculation methods to analyze multirotor aircraft in forward flight (Refs. 3, 6). These studies used a comprehensive analysis tool based on blade element theory (BET) to determine the trimmed control variables. BET models are not able to capture complex flow physics such as flow separation and interactional effects such as blade-vortex interactions. In order to obtain truly accurate trim states, a CFD analysis must be used that models both rotors and fuselage and the trim algorithm must also account for these geometrical components.

The initial effort to utilize CFD in a quadrotor trim calculation was demonstrated by Roget et al., who simulated the DJI Phantom 3 in trimmed forward flight by coupling Helios with the Rotorcraft Comprehensive Analysis System (RCAS) (Ref. 7). Although the force convergence was reasonable, the trim variables did not converge to steady values. This was attributed to the unsteady, non-periodic flow field that occurs because each rotor is operating at a different frequency. It was observed that averaging the loads over each revolution resulted in improved convergence of the control variables. Even though the authors were able to use a modified version of the RCAS code to model rotor speed control, extending the capability to more complex multirotor configurations was not trivial, and it was deemed preferable to develop a dedicated trim module that would be designed to accommodate arbitrarily complex configurations.

Other related efforts include that of Diaz et al. who performed simulations of NASA's Side-by-Side eVTOL concept aircraft (Ref. 8). They used a loose coupling approach with CAMRAD II for the CSD (computational structural dynamics) solution and Overflow for the CFD solution, trimming only for rotor pitch angles. Similarly, Jia et al. (Ref. 9) used Helios to simulate NASA's quadrotor eVTOL concept, using RCAS to obtain the CSD solution, and a combination of Overflow for the near-body solver and SAMCART for the off-body CFD solvers. They also trimmed for collective and cyclic pitch only, while keeping rotor speeds constant and equal for all four rotors.

Computations related to the Uber vehicle were previously presented by Casalino et al. They showed a flow visualization of the vehicle in flight for demonstration purposes but did not include discussion of their simulation method, mesh information, flight parameters, aerodynamic properties or flow solution (Ref. 10). The focus of their paper was instead on the Dassault Systems 3DS eVTOL demonstrator for UAM development. The vehicle was not trimmed and constant speeds were used for all rotors.

The approach that is used in the present research builds on these past studies. A trim optimization code has been developed, using the least-square regression algorithm to adjust control variables and achieve a trimmed flight condition. When used alone, the airloads are computed by a simple blade element theory (BET) model for the rotor blades, and constant aerodynamic coefficients are assumed for non-blade bodies. The accuracy of the trim calculation is then improved through loose-coupling with a CFD solution. Helios is used in this work to obtain a high or mid fidelity CFD solution. The loose-coupling method modifies the airloads computed by the simplified BET model via the CFD predicted airloads at specified coupling iterations. This method is first applied to the DJI Phantom 3 quadrotor and then further demonstrated on a more complex multirotor vehicle, the Uber eCRM-001 vehicle. In contrast to the work by Casalino et al, computed results are presented for the trimmed hover state for the Uber vehicle in this paper.

METHODOLOGY

The loose-coupling procedure utilized in Helios involves iterative exchange of loads and displacement between a CFD solver and a simplified aerodynamics based trim analysis at prescribed intervals (Ref. 11).

Convergence is achieved when the difference between CFD-computed airloads and airloads computed by the simplified

model in the trim module remains constant at the each iteration. In this paper, two methods for computing higher fidelity unsteady aerodynamic loads are used. First, full CFD performed using mStrand as the near-body solver is applied to the DJI Phantom simulation. Second, a combined lifting line near-body and CFD off-body approach (ROAM) is applied to the Uber eCRM-001 simulation. For both examples, the CFD based results are coupled to the newly developed trim module. The trim module is based on blade element theory and incorporates an optimization method for power minimization. All facets of this methodology are described briefly in this section.

CREATE^{TM}-AV Helios

The HPCMP CREATE^{TM}-AV Helios is a rotorcraft simulation code developed by the U.S. Army's Aviation Development Directorate (Refs. 12, 13). Helios enables overset high-fidelity simulations that can use several different near-body high-fidelity Navier Stokes solvers, including NASA's FUN3D and OVERFLOW, KCFD, or the in-house developed mStrand solver (Refs. 14, 15). In the present study, the mStrand solver was chosen for modeling the DJI Phantom 3 due to its ability to use strand-based volume grids, which can be automatically generated from a surface geometry description at run-time by Helios (Ref. 16). The mStrand solver uses a compressible finite volume scheme with 2nd order discretization in both space and time. Helios also enables coupling of the CFD solution with several comprehensive analysis codes, including RCAS and CAMRAD II. The trim module developed in this study provides an alternate solution to obtain rigid-body trimmed CFD simulations.

For the off-body flow computations, Helios uses SAMCART, a Cartesian mesh solver capable of adaptive mesh refinement (AMR). SAMCART implements a 5th order finite difference scheme for its inviscid terms and a 4th order discretization for its viscous terms. The limit used for identification during the AMR process is the q-criterion scaled by the square of the shear strain rate. Both SAMCART and mStrand solve the unsteady Reynolds-averaged Navier-Stokes (URANS) equations and the Spalart-Allmaras (SA) turbulence model was used for turbulence closure. The interpolation between solvers is handled by the PUNDIT module, whereas the Melodi module applies motions and deformations to the appropriate mesh, and reduces the airloads from the CFD solvers onto the control points used for output and for coupling with the CSD modules. The new trim module communicates directly with Melodi to prescribe new controls and obtain the reduced CFD airloads. Helios utilizes a python based software integration framework to orchestrate the execution sequence and data exchange between various solver modules.

Reduced-order Aerodynamic Model

The reduced order aerodynamic model (ROAM) in Helios is a mid-fidelity CFD approach that couples an actuator line model (Ref. 17) for the rotor blades and an immersed boundary representation for bluff bodies such as fuselages and nacelles. A mesh is created for use in SAMCART that encompasses the rotors, rotor-wakes, fuselage, and nacelle regions. The Navier Stoke equations are modified to include source terms derived from the actuator line model that account for the effect of the rotors. In the actuator line model, lifting lines are discretized into a set of control points and the flow induced velocities at these control points are interpolated from the SAMCART mesh system used to model the flow field. To exclude the effects of bound circulation, an array of sensors is created located on a circle around each control point and the velocity at the control point is found as an average of all of the sensors. Once the velocities at control points are obtained, linear aerodynamic theory augmented with airfoil tables is used to determine the force coefficients and compute the force distribution. The force distribution is then embedded into the flow field equations through the SAMCART source terms. The transfer of velocities using interpolation and resultant forces using source terms is the basis of the coupling structure between the actuator line model and the Navier-Stokes equations.

The immersed boundary method used to model components such as the fuselage is a popular geometry representation technique first developed by Peskin (Ref. 18), where only a Cartesian grid system is used to model the entire flow field and the body of interest is represented as masked areas in the Cartesian system. In general, only the discrete representation of the body surface is required for the computation to proceed. In the context of Helios, the ability of SAMCART to provide adaptively refined Cartesian meshes is utilized to provide a well resolved mesh system near the component's boundary. A fast ray tracing algorithm is utilized to cut away and mask the areas of the Cartesian grid that are within the solid walls. The first few layers of nodes within the solid wall are used as interpolation points to enforce boundary conditions. The minimum distance point to the solid wall is estimated and the direction of minimum distance is used to reflect the interpolation points on to the flow field and interpolate their solution values.

Details of the entire ROAM implementation can be found in (Ref. 19). Note that the ROAM approach is able to capture the rotor-rotor interactions and rotor-fuselage interactions that are critical for aerodynamics of multirotor vehicles. The blade aerodynamic loading itself is however less accurate than discrete blades simulations. The computations themselves are an order of magnitude faster and hence ROAM serves as a mid-fidelity design tool. The results presented for the Uber eCRM-001 use this approach within Helios for the CFD computations.

Generalized trim analysis for rotorcraft

A new trim module is under development in Helios to generalize rotorcraft trim analysis to include individual rotor speeds as well as blade root angles and aircraft attitudes as control variables. This trim module uses a simplified model based on

blade element theory (BET) and lifting line aerodynamics to compute total airloads for an aircraft with rigid components.

In the blade element theory, each blade is divided into several spanwise elements and the rotor forces and moments are computed by integrating elemental forces over the entire blade and over one rotor revolution. The relative velocity at each blade airfoil element is the sum of the influences from rotor rotation speed, incoming wind, and the induced inflow through the rotor disk. Specifically, the new trim module uses a hierarchical frame system in which the relative air velocity components are computed with respect to the inertial frame. The hierarchy has four levels, each related to a corresponding rotation matrix: R_1 describes the rotation from the inertial frame to the aircraft-attached flight dynamic frame, based on the current aircraft pitch, roll and yaw attitude angles. R_2 describes the rotation from the aircraft flight dynamics frame to each rotor hub frame, based on angles provided as a user input. The z-axis is the axis of rotation, pointing in the direction of positive lift and the x-axis defines the location of azimuth 0. R_3 describes the rotation from the hub frame to the rotating blade root frame, using the rotation angle and blade number. Finally, R_4 describes the rotation from the rotating blade root frame to the local airfoil frame, for each control point on the blade, based on the user-provided blade geometry.

The velocity due to aircraft speed, v_1, blade rotation, v_2, and induced inflow, v_3, can then be expressed in the airfoil frame as:

$$v_1 = (R_1 R_2 R_3 R_4)^T V_{air} \tag{1}$$

$$v_2 = -(R_4)^T \Gamma s \tag{2}$$

$$v_3 = (R_4)^T V_{induced} \tag{3}$$

where Γ is the rotor rotation vector: $\Gamma = (0\ 0\ \Omega)^T$,
V_{air} is the wind velocity, s is the spanwise distance of the blade element, and $V_{induced}$ is the induced velocity. The total velocity, V_{tot} is the sum of these velocity contributions. The angle of attack, α, is determined using the y and z-components of the total velocity. The induced inflow is assumed to be uniform over the rotor disk and is computed iteratively based on simple momentum theory. The module can accept airfoil tables for aerodynamic coefficient look-up, however, in this study, the drag coefficient is assumed constant for the rotor blades and the lift coefficient is assumed to be a linear function of the angle of attack (a drag coefficient of 0.015 and a lift curve slope of 5.8 were used). Therefore, the lift, dL and drag, dD on each blade element are:

$$dL = \frac{1}{2}\rho V_{tot}^2 C_l(\alpha) c\, ds \tag{4}$$

$$dD = \frac{1}{2}\rho V_{tot}^2 C_d c\, ds \tag{5}$$

Finally, the forces in the airfoil frame can be calculated:

$$dF_y = dL\sin(\alpha) - dD\cos(\alpha) \tag{6}$$

$$dF_z = dL\cos(\alpha) + dD\sin(\alpha) \tag{7}$$

For non-blade bodies such as fuselage, nacelles, and rotor hubs, the aerodynamic forces and moments are assumed to be constant, and provided as a user input.

The control variables are the rotor speeds and the vehicle attitudes. These RPM-based controls are typical of multirotor aircraft which do not have collective and cyclic pitch inputs. The trim problem involves achieving zero total forces and moments at the vehicle center of gravity, when considering both the aerodynamic loading as well as the weight of each component.

The typical trim state is described by six equations, one for each of the linear and angular accelerations. Because of the fact that there are three degrees of freedom related to the flight attitudes, vehicles with more than three rotors have more than six control variables, and thus represent under-determined nonlinear systems. A Newton type iterative scheme is utilized to solve the nonlinear problem. Linearization around the current state leads to an under-determined linear system that can be solved using a least-square regularization approach. The residual vector $F(q)$ at each Newton iteration is a six component vector with each component representing the error in satisfying each of the nonlinear trim constraint equations respectively. The least-square solution seeks to find the state q that minimizes the norm of the residual vector $F(q)$. The Jacobian matrix J is necessary for finding the change in state at each Newton iterate. Given

$$J = \frac{\partial F}{\partial q} \tag{8}$$

The linearization around current state q gives

$$F(q+\delta q) \approx F(q) + J\delta q = 0 \tag{9}$$

The change in control variables (or direction of descent) to minimize the force residual can then be expressed as:

$$\delta q = -(J^T J)^{-1} J^T F(q) \tag{10}$$

and the states are modifed at each Newton iteration k as

$$q_{k+1} = q_k - ({J_k}^T J_k)^{-1} {J_k}^T F(q_k) \tag{11}$$

To improve convergence, the Jacobian matrix J_k is recomputed at every Newton iteration.

One can either use automatic differentiation or a finite difference approximation to obtain the Jacobian matrix. Automatic differentiation is the process of computing derivatives using the chain rule simultaneously with standard operations. The dual number automatic differentiation tool was tested. It employs Fortran's derived data type to store both the functional value of a variable and its derivative with respect to the controls. The method was much slower than the finite difference approach due to the large amount of required computations to obtain the derivative. As such, the finite difference approach has been used for this study. Furthermore limits are applied to the variation of each state variable to mitigate instabilities.

This least-squares method can successfully identify a trim solution for most flight conditions. However, if multiple trim

solutions exist for an under-determined system, the resulting solution is not necessarily optimal. The proposed alternative is to explore power minimization under the constraint to maintain trimmed flight conditions.

For this purpose, the SLSQP code from J. Williams (Ref. 20), based on the original sequential quadratic programming algorithm by Kraft (Ref. 21), was incorporated in the trim module. The SLSQP (sequential least squares quadratic programming) nonlinear constrained optimization code can be used to solve nonlinear programming problems that seek to minimize a scalar performance index subject to nonlinear equality and inequality constraints as well as bounds on the variables. In the present study, the SLSQP code is used to attempt total power minimization with zero vehicle accelerations in each direction as the nonlinear optimization constraints. Bounds on the control variables are also used based on user inputs in the trim input file. However, while testing this method as part of the stand-alone trim module, it was observed that convergence of the control variables was not always achieved. Additionally, the identified trim solution was found to depend on the initial conditions used and does not always correspond to the global power minimum, as will be shown later. To alleviate these issues, both the least-squares (LS) and SLSQP methods are attempted at each coupling iteration, and the trim solution corresponding to the least total power is selected. In case both algorithms fail to achieve trim, the previous trim solution is maintained and trim is attempted again at the next iteration of the loose-coupling approach.

Trim module input file

The input deck derives its file format from Fortran 90's namelist function. It currently consists of five sections, outlining the aircraft properties, the control parameters, the body definitions, the rotor definitions, and the rotor properties.

Aircraft properties section: Here, the user may specify the number of aerodynamic bodies, such as fuselage components, wings, and rotors. Here, the number of rotor types is also specified. Each rotor type will have one rotor rotation direction and associated chord and twist files. Finally, the number of control variables and control targets are listed.

Control parameters section: Here, the controls and any control constraints are specified. For example, co-rotating rotors are constrained to operate at the same rotor frequency. The perturbation values used in the finite-difference approximation of the optimization gradients are specified here.

Body definitions section: Here, the user specifies the mass, location, orientation, and body area of each aerodynamic body. The mass is used in the trim calculation and contributes to the aircraft weight. The body area is used to approximate the drag around the vehicle, which is currently assumed to have a constant drag coefficient. The aerodynamic loads on the bodies are coupled to the CFD calculations, so this drag estimate is only used as a way to initialize the simulation.

Rotor definitions section: Here, the user inputs the rotor locations and orientations of each rotor. Care must be taken to ensure that rotor frames defined in the trim module match the location of the blade reference frames in Helios.

Rotor properties section: Here, the user outlines the rotor speeds, direction of rotation, rotor radius, and number of blades of individual rotors. To simplify property assignment, rotors with identical aerodynamic properties may be grouped.

Coupling the trim module with CFD

Loose aerodynamic coupling is the periodic exchange of aerodynamic forces and moments between the CFD solver and the trim module. In the simplified aerodynamic model used by the trim module, the forces and moments are computed using blade element theory modified by a correction term:

$$F_k = F_k^{LL} + \Delta F_{k-1}. \tag{12}$$

The delta airloads term, ΔF_k, is equal to the difference between the current CFD solution and the blade element theory solution:

$$\Delta F_k = F_k^{CFD} - F_k^{LL} \tag{13}$$

where F_k^{CFD} is the periodic airloads vector, obtained by interpolating forces calculated by the CFD near-body solver, onto control points locations (corresponding to quarter chord points for a blade, or a single reference point for other bodies). This operation is performed by the Melodi module in Helios and is described in detail in (Ref. 7). This correction term can also be represented by the sum of the previous correction and the change in the CFD and BET predictions between the last two iterations:

$$\Delta F_k = \Delta F_{k-1} + \Delta F_k^{CFD} - \Delta F_k^{LL}. \tag{14}$$

Therefore, convergence is established when the delta airloads term remains constant over multiple coupling iterations, i.e. the aerodynamic loading computed by both CFD and simplified aerodynamics become invariant. The trim module, written in Fortran 90 is coupled using the Python-based software integration framework that is implemented in Helios.

After computing the first set of control variables to initialize the simulation (based only on BET), the first coupled iteration typically occurs after one full rotor revolution of the main rotor (defined as the slowest rotor in the model), in order to allow initial transient in the flow solution to die down. The time elapsed between subsequent loose-coupling iterations can be smaller, but should be large enough that the periodic airloads due to the new set of control variables can be reconstituted from the airloads time history of each blade. Assuming blades are identical and neglecting the transient due to the sudden change in controls, this time can in theory be as short as the longest rotor period divided by the number of rotor blades. However, for multirotor vehicle trim this longest rotor period may vary significantly and it is difficult to predict the required loose coupling sequence beforehand. To avoid processing an incomplete airloads time history, in the present study, loose coupling iterations are conservatively set to occur every 360 deg based on the slowest rotor at the start of the simulation.

In the future, it would be more efficient to automatically compute the loose-coupling sequence at run-time based on current rotor frequencies so that the total simulation time can be minimized. The loose coupling algorithm currently implemented is summarized in Algorithm 1.

Algorithm 1: Loose Coupling Method

k=0
t_0=0
t=0
Compute initial trim solution
$\rightarrow$ controls : q_k
$\rightarrow$ periodic airloads from simplified model : F_k^{LL}

while $k <$ *total number of coupling iterations* **do**

  while $(t - t_0) <$ *coupling duration* **do**
    Compute CFD solution at time t using controls q_k
    t $\leftarrow t + \Delta t$
  end

  Compute periodic airloads, using load stitching and time history corresponding to:

  - last $2\pi/\Omega$ for rotor blades
  - last $2\pi/\Omega_{min}$ for other bodies

  $\rightarrow$ periodic airloads from CFD : F_k^{CFD}

  Compute trim solution using BET, with airloads augmented by: $\Delta F_k = F_k^{CFD} - F_k^{LL}$
  $\rightarrow$ controls : q_{k+1}
  $\rightarrow$ periodic airloads from simplified model : F_{k+1}^{LL}

  k $\leftarrow k+1$
  $t_0 \leftarrow t$
end

DJI PHANTOM 3 RESULTS

The DJI Phantom 3 UAV was chosen for this paper because of its commercial success and large experimental database (Refs. 22, 23). Carl Russell of NASA Ames provided the authors of this study with the laser scanned geometry of both the rotor and the fuselage for the DJI Phantom 3 configuration. The vehicle, with a maximum RPM of 7500 and a 24 cm rotor diameter, has a tip chord length of approximately 1.2 cm and a corresponding tip Reynolds number of 120,000. The gross weight was set to 1 kg, which is typical of the configuration without the camera and gimbal. This UAV is flown in a square formation with two front rotors and two rear rotors so that the camera heading matches the flight direction, as shown in Fig. 1. Small-scale quadrotors that rely on RPM-based control typically operate two diagonal clockwise rotors and two counter-clockwise rotors, as seen in Fig. 1. The wind direction labeling and rotor numbering from this figure will be referenced throughout the paper and are indicative of the orientation used for the simulations.

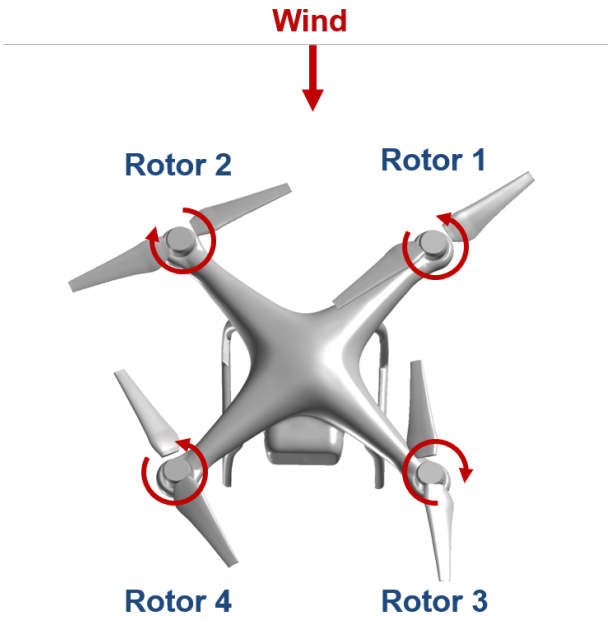


Figure 1: Top view of the DJI Phantom 3 in forward flight. The direction of the incoming wind, the rotor numbering scheme used, and the rotor direction of rotation are labeled.

Trim module results

The trim module was first tested stand-alone for the DJI Phantom 3 for aircraft velocities ranging from 0 m/s to 20 m/s. The forces and moments converged across the entire spectrum of flight speeds, and the resulting controls are plotted in Fig. 2. At higher flight speeds, the vehicle starts to pitch down, and thus the front rotors (Rotors 1 and 2) converge to a lower rotor speed than the rear rotors (Rotors 3 and 4). This can be explained by the need to counteract the positive pitching moment caused by the rotor aerodynamic drag forces applied at each hub center and located above the aircraft center of gravity. This phenomenon becomes less evident at the lower flight speeds as the drag force reduces. In the DJI geometry, the motors are oriented slightly away from the fuselage center presumably to minimize rotor-rotor aerodynamic interactions. This small inclination of the rotor normal vectors is responsible for the rotor speeds converging to four different values to satisfy trim.

CFD coupled results

Fully-coupled simulations of the DJI Phantom 3 were performed at both 10 m/s and 15 m/s forward flight as well as in hover. The surface mesh was generated using a parameterized blade representation (bladegen) supported by Helios with 65 points around each blade airfoil section. A leading edge spacing of 0.0008 chord and a trailing edge spacing of 0.0019 chord were used. These characteristics were chosen to match that of Roget et al. (Ref. 7). The mesh system can be found in Fig. 7a, where the mStrand volume mesh, in blue, can be seen overlaid on the SAMCART off-body mesh, in black. The off-body mesh is visibly coarse in front of the aircraft and

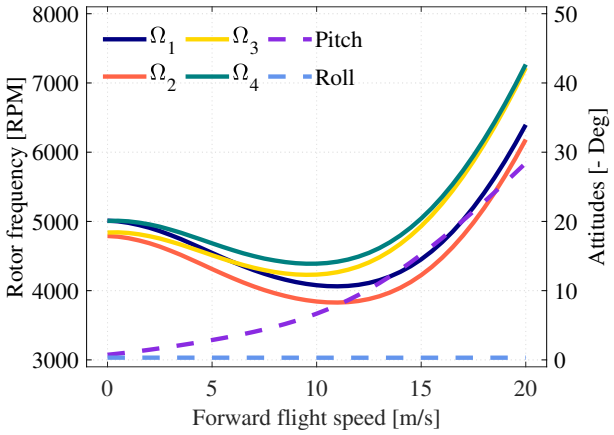


Figure 2: Resulting control variables for the DJI Phantom 3 at various flight speeds without CFD coupling.

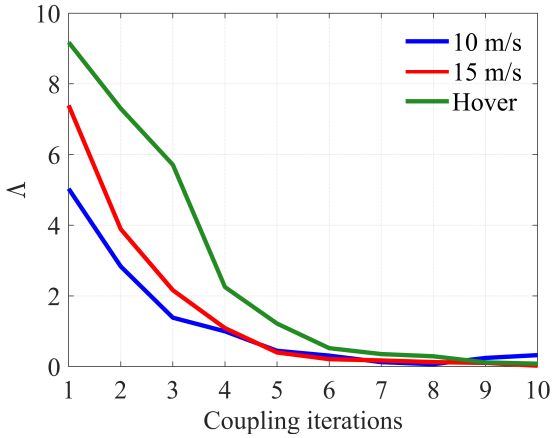


Figure 3: Convergence of the loads objective function for the DJI Phantom 3.

fine in the wake, as the AMR algorithm automatically adjusts the off-body mesh refinement based on vorticity levels. The image was extracted at the 8th loose coupling iteration step, and the flow visualization images for the corresponding time step are shown in Figs. 7b-7d, where different views of forward flight are rendered using the iso-surfaces of q-criterion colored by velocity magnitude. The vortices generated by the rotor blade are well-resolved, and the rotor wake is satisfactorily captured by SAMCART. The flowfield is shown to be highly interactive, and the fuselage is subject to large amounts of wake impingement.

In order to demonstrate the convergence of the forces, an objective function, Λ, is defined that represents the norm of the difference between the computed CFD airloads at the center of gravity, Ψ_{CFD}, and the desired airloads, Ψ_{target}, required to achieve trim:

$$\Lambda = \|\Psi_{CFD} - \Psi_{target}\|. \tag{15}$$

In trimmed flight, this objective function should be minimized and therefore is a good indicator of convergence. The evolution of Λ for the DJI cases is shown in Fig. 3. For all three cases, the objective function is shown to approach a minimum after around 5-6 coupling iterations. However, further coupling iterations show that convergence then stalls and the objective function is not reduced further.

To investigate the resultant trim states, the convergence of the trim control variables is plotted in Figs. 9-11. The aircraft attitudes seem to converge to a steady value around 5-6 coupling iterations, as expected from the analysis of the loads objective function. As shown from the stand-alone trim analysis, the vehicle progressively requires a larger negative pitch as the vehicle speed increases. However, although the rotor frequencies seem to be approaching an equilibrium state, the values are still fairly oscillatory, especially at the lower flight speeds. One main reason for this behavior is that since each rotor operates at a different frequency, due to varying rotor-rotor interactions, the rotor airloads are not exactly periodic and vary slightly at each loose coupling iteration, even when the vehicle is in trim. It is expected that using phase averaging, in which the airloads used for loose coupling are the average over multiple rotor revolutions, will improve the convergence of the trim variables. Phase averaging mitigates the impact of transience in RPM changes by reducing abrupt differences in the delta airloads values, enabling the controller to approach the solution more steadily.

Contour plots for the rotor airloads are shown over the last coupling iteration for each of the fully-coupled cases in Figs. 13-14. The results in forward flight exhibit good qualitative behavior, with expected trends of higher sectional normal force on the advancing sides of the blades, which vary from rotor to rotor depending on the rotor rotation direction. The airloads in hover are more uniform due to the fact that there is no incoming wind. More importantly, the rotor frequencies are not equal, which will have implications on noise, and suggests that a trim controller is necessary even in hover. The airloads for all cases show an increase in sectional normal force on the rotors corresponding to the azimuthal range that passes over the fuselage, around 315° for the front rotors and 135° for the rear rotors. Therefore, it is evident that the CFD solver is able to capture rotor-fuselage interactions that are important for the overall vehicle aerodynamics.

UBER ECRM-001 RESULTS

Uber Elevate has made available three common reference models for eVTOL applications based on the flight requirements of a 60 mile sizing range with 4 passengers and 1 pilot (Ref. 24). The eCRM-001 was chosen due to both its compact volume as well as the lift and cruise capabilities of the outer push propellers. The vehicle geometry properties and CAD geometry are available on the Uber Air website in OpenVSP format. Although the weight of the vehicle was not specifically reported by Uber, an operating gross weight of 2000 kg was assumed based on their reported estimate of a typical eVTOL aircraft weight (Ref. 25). This estimate agrees with the current literature on vehicle weights in the lift and cruise category of eVTOL vehicles with similar passenger capacity (Refs. 26, 27).

The eCRM-001 aircraft has four pairs of co-rotating, in-phase, lift rotors and two tiltrotor push propellers for a total of ten

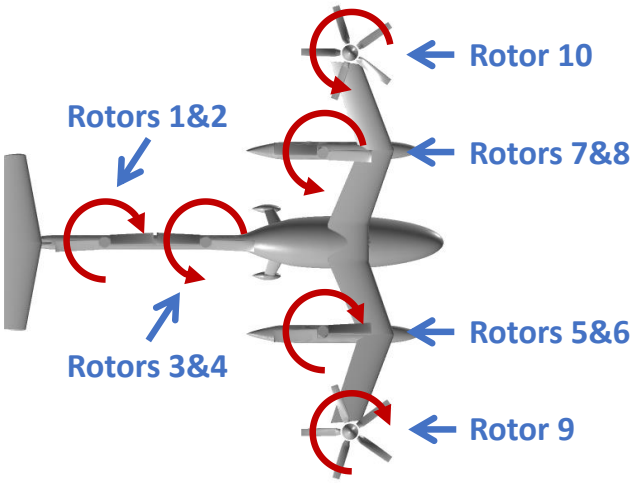


Figure 4: Top view of the Uber eCRM-001 in hover. The rotor numbers and rotor direction of rotation labeled are used in this study.

rotors. Each pair of co-rotating lift rotors was constrained to have the same rotor speed in the trim controller to match the design. A top view of the eCRM-001 with numbered rotors as well as rotation directions can be found in Fig. 4. The lift rotors, rotors 1-8, have two blades each while the push propellers, rotors 9-10, each have five blades. The rotor directions all match that of the OpenVSP geometry, except for rotors 9-10. These changes are not expected to have a significant impact on the trim state in hover. The rotor diameter for all rotors is 108 inches.

The lift rotors in this study were modeled as extrusions of a NACA 0012 airfoil. The chord is kept at a constant 0.2427R, but the twist was interpolated with a spline around discrete control points. Each lift rotor has its own hub, so there are a total of 8 hubs that rotate in sync with each rotor pair. The push propellers have a circular cross section at the root, which starts at 0.1R. The propeller blade surface meshes were generated at run time using bladegen and match the exact geometry based on the OpenVSP file.

The OpenVSP geometry file provided by Uber kept the eCRM-001 in the cruise configuration, and therefore multiple adjustments were needed to simulate hovering flight. The eCRM-001's lift rotors are meant to retract in cruise, and the original model had them stored inside the fuselage. Therefore, for this study the rotors and their corresponding hubs were raised. A gap was added between the hubs of each co-rotating pair to avoid issues with volume mesh generation. The wing tip geometry was not perfectly compatible with the outer nacelle due to surface intersections. Therefore, the wing tips was adjusted in order to allow space for the outer tilt-rotor nacelle. All other bodies on the fuselage were joined before the mesh generation process. No other simplifications were made, and the rest of the fuselage geometry was modeled exactly from Uber's provided VSP file. The differences between the model provided by Uber and that used in the simulation can be seen in Figs. 15a-15b. The new Helios Graphical User Interface

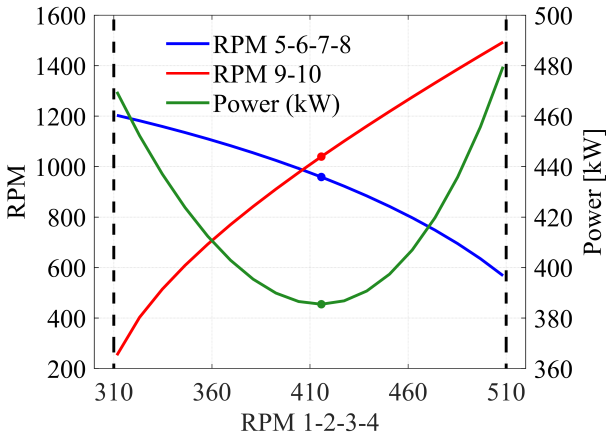


Figure 5: Various trim solutions for the Uber eCRM-001 in hover. The dots mark the minimum power solution and the dashed lines mark the limit of possible trim solutions.

HIGEN was used to create the Helios input file corresponding to the eCRM-001 geometry in hover. Fig. 15c shows a screenshot of the model in HIGEN. The GUI greatly facilitates assembly for such a complex problem, with a total of 37 separate bodies, by providing direct visual feedback during model set-up and allowing bodies to be positioned with respect to any existing frame in the hierarchy. In addition, HIGEN includes a utility to extract blade properties such as chord, twist and quarter-chord directly from the surface mesh.

Several control constraints were applied to simplify the model for trim. The pitch, roll, and yaw were all constrained to zero, which should be true for a hover condition. In addition, several rotor groups were set with equal rotor frequencies. Specifically, due to the symmetry of the vehicle, rotors 1 through 4 (located on the tail) made up the first group, rotors 5 through 8 (located on the wings) the second group and finally the tilt propellers 9 and 10 were also grouped together with equal rotor speed magnitude. The controls were thus limited to variations in the rotor speeds of rotor 1, 5, and 9.

Trim stand-alone results

The trim module was run stand-alone to test the simplified model of the Uber eCRM-001 in hover. To explore the possible trim solutions of the vehicle, the rotor frequencies of the first rotor group, corresponding to rotors 1 through 4, were fixed to constant values. Since the aircraft is operating in hover there is no drag force and no trim constraint equation is necessary in the X-direction. Furthermore, only two trim constraints need to be explicitly satisfied – the vertical force F_z and the aircraft pitching moment M_y at the aircraft CG – as the other trim constraints will be automatically satisfied by virtue of symmetry about the X-Z plane. Then, due to the fact that there are only two remaining controls, corresponding to rotor speeds of the other rotor groups, the trim system has an equal number of equations and unknowns and is fully determined. The trim module was used to calculate the remaining controls and the results are shown in Fig. 5. The blue lines cor-

respond to rotors 5 through 8, the red lines correspond to the propellers 9 and 10, and the green line represents the power of each trim state. No trim solution exists when the rotor speed for the first group is below 310 RPM or above 510 RPM, and the dashed black lines outline these limits. The global optimum solution, where power is minimized to 385 kW occurs with the first rotor group at 416 RPM, the second one at 959 RPM, and the third one at 1040 RPM.

CFD results

The full aircraft model for Uber eCRM-001 with body conforming grids for all rotors, hubs, nacelle and fuselage is complete. However, because of the large size of the model and extensive computational resource requirements, fully coupled computations could not be completed within the time constraints of this work. These computations are ongoing at this time. Therefore, results obtained using the mid-fidelity aerodynamic methodology (ROAM) coupled to the trim module is presented here. The iso-surfaces of vorticity magnitude colored by pressure coefficient are shown in Fig. 16. The actuator line method of ROAM is able to produce complex wake structures from the rotors that are resolved well by SAMCART. While the blade aerodynamic loads are not as accurate as full body conforming simulations, the ROAM approach is still able to resolve rotor-rotor and rotor-fuselage interactions. Owing to the placement of rotors, a significant download of approximately 3600 N (18% of gross weight) is calculated by ROAM's implementation of the immersed boundary method. These interactional effects directly impact the trim state. Therefore, the download used by the preliminary ROAM calculation was used to initialize the fully-coupled simulation.

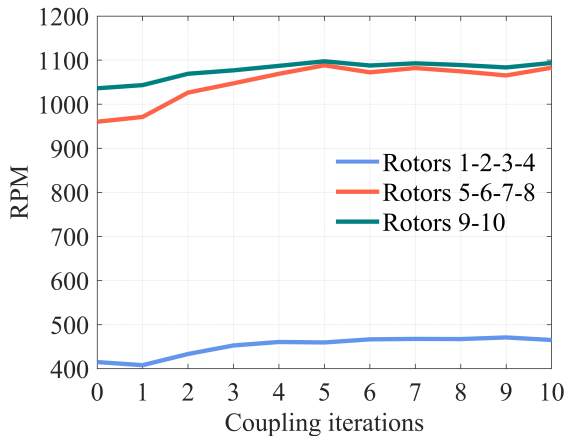


Figure 6: Convergence of the control variables for the Uber eCRM-001 during a fully coupled simulation with ROAM.

The progression of the control variables for the Uber eCRM-001 during the fully-coupled simulation is shown in Fig. 6. The trim controller demonstrates good convergence, with the rotor speeds all approaching steady values. Note that the initial conditions for the fully coupled trim are chosen as the ones that give the minimum power solution in trim stand-alone computations. Therefore, the fully coupled solution tends to converge close to the trim stand-alone solution corresponding to the minimum power results. The resultant power at the final coupling iteration was 429 kW, with the first rotor group at 465 RPM, the second group at 1083 RPM, and the third group at 1094 RPM.

CONCLUSIONS

A new generalized trim analysis module is under development for the rotorcraft simulation code CREATE-AV^{TM} Helios. This module uses blade element theory for stand-alone analysis and is capable of loose aerodynamic coupling methodology in which the airloads difference with the CFD solution is exchanged periodically. The control variables used for trim can include individual rotor frequencies as well as the traditional blade root angles and vehicle attitudes. The module was tested for two different configurations that represent the range from small-scale UAV to mid-scale UAM.

The convergence of the loads objective function for the DJI Phantom 3 demonstrated an approach to equilibrium, but significant oscillations in the control variables remained present towards the end of the simulation. These oscillations are postulated to be caused by the inherent aperiodicity in the flow field because of variable rotor frequencies. It is possible that phase averaging of the loads will help to smooth the oscillations and therefore improve convergence, and this will be explored in the future. However, the flow solution and airloads appeared well-resolved overall, and important flow interactions were well captured by the CFD solution approach.

The Uber eCRM-001 vehicle was simulated in hover, and this more complex configuration revealed several important details about the trim algorithm. The optimization method and initial conditions chosen have significant impact on the stand-alone trim solution. Various trim states were discovered by fixing the RPM of one group of rotors, and the systematic search of the design space yielded a global optimum solution with minimum power. The trim module can converge to a different equilibrium solutions depending on the chosen initial conditions, and a more robust optimization strategy is required to identify the global optimum.

The Uber vehicle was modeled using the trim module in a fully-coupled simulation with ROAM, a reduced-order aerodynamic model in Helios that implements the immersed boundary method for the fuselage, nacelles and hubs, and actuator lines for the rotor blades. The ROAM solver is able to capture rotor-fuselage interactions, and the download played a significant role in the trim state. However, due to the complexity of the configuration, many constraints are needed to obtain the solution, which potentially over-simplify the trim problem. The constraints, consisting of rotor RPM groupings and fixed zero aircraft attitudes, are good approximations for the hover flight condition but may not be suitable for low speed flight or even hover with gusts and cross-winds. Complex flow interactions could result in non-symmetric rotor speeds as seen in the DJI Phantom simulations. Therefore, the next

step is to simulate the Uber vehicle with fully-coupled CFD and to integrate a more robust procedure for optimizing trim for complex configurations with multiple rotors.

Author contact: Austin Thai adthai@bu.edu.

ACKNOWLEDGMENTS

The authors would like to thank Carl Russell for providing the DJI Phantom 3 geometry. The first author would like to thank Roger Strawn for the opportunity to intern with the Aviation Development Directorate. The first author would also like to acknowledge both NSF Award # 1728277 and the Boston University Mechanical Engineering department for funding graduate studies. The second and third authors are grateful for support from the HPCMP CREATE-AV program for continued development of Helios. Material presented in this paper is a product of the CREATE-AV Element of the Computational Research and Engineering for Acquisition Tools and Environments (CREATE) Program sponsored by the U.S. Department of Defense HPC Modernization Program Office.

REFERENCES

1. Datta, A., and Johnson, W., "An Assessment of the State-of-the-art in Multidisciplinary Aeromechanical Analyses," AHS Specialist's Conference on Aeromechanics, San Francisco, CA, January 2008.

2. Yoon, S., Diaz, P. V., Jr., D. D. B., Chan, W. M., and Theodore, C. R., "Computational Aerodynamic Modeling of Small Quadcopter Vehicles," AHS 73rd Annual Forum, Fort Worth, TX, May 2017.

3. Diaz, P. V., and Yoon, S., "High-Fidelity Computational Aerodynamics of Multi-Rotor Unmanned Aerial Vehicles," 2018 AIAA Aerospace Sciences Meeting, Kissimmee, FL, January 8–12, 2018.

4. Thai, A. D., Jain, R., and Grace, S. M., "CFD Validation of Small Quadrotor Performance using CREATE-AV Helios," VFS 75th Annual Forum, Philadelphia, PA, May 2019.

5. Thai, A. D., and Grace, S. M., "Prediction of Small Quadrotor Blade Induced Noise," 25th AIAA/CEAS Aeroacoustics Conference, Delft, The Netherlands, May 2019.

6. Misiorowski, M., Gandhi, F., and Oberai, A. A., "A Computational Study on Rotor Interactional Effects for a Quadcopter in Edgewise Flight," AHS 74th Annual Forum, Phoenix, AZ, May 2018.

7. Roget, B., Sitaraman, J., Blumenstein, R., Taheri, M., and Saberi, H., "Advanced Rotorcraft Aeromechanics Simulations using HPCMP CREATE-AV Helios," VFS 75th Annual Forum, Philadelphia, PA, May 2019.

8. Diaz, P. V., Johnson, W., Ahmad, J., and Yoon, S., "Computational Study of the Side-by-Side Urban Air Taxi Concept," VFS 75th Annual Forum, Philadelphia, PA, May 2019.

9. Jia, Z., and Lee, S., "Acoustic Analysis of a Quadrotor eVTOL Design via High-Fidelity Simulations," 25th AIAA/CEAS Aeroacoustics Conference, Delft, The Netherlands, May 2019.

10. Casalino, D., van der Velden, W. C., and Romani, G., "Community Noise of Urban Air Transportation Vehicles," AIAA Scitech 2019 Forum, San Diego, California, January 2019.

11. Potsdam, M., Yeo, H., and Johnson, W., "Rotor Airloads Prediction using Loose Aerodynamic/Structural Coupling," AHS 60th Annual Forum, Baltimore, MD, June 2004.

12. Sankaran, V., Wissink, A., Datta, A., Sitaraman, J., Jayaraman, B., Potsdam, M., Katz, A., Kamkar, S., Roget, B., Mavriplis, D., Saberi, H., Chen, W.-B., Johnson, W., and Strawn, R., "Overview of the Helios Version 2.0 Computational Platform for Rotorcraft Simulations," 49th AIAA Aerospace Sciences Meeting, Orlando, FL, 2011.

13. Wissink, A., Jayaraman, B., Datta, A., Sitaraman, J., Potsdam, M., Kamkar, S., Mavriplis, D., Yang, Z., Jain, R., Lim, J., and Strawn, R., "Capability Enhancements in Version 3 of the Helios High-Fidelity Rotorcraft Simulation Code," 50th AIAA Aerospace Sciences Meeting, Nashville, TN, 2012.

14. Lakshminarayan, V. K., Sitaraman, J., and Wissink, A. M., "Application of Strand Grid Framework to Complex Rotorcraft Simulations," *Journal of the American Helicopter Society*, Vol. 62, 2017, pp. 1–16.

15. Lakshminarayan, V. K., Sitaraman, J., Roget, B., and Wissink, A. M., "Simulation of Complex Geometries Using Automatically Generated Strand Meshes," 2018 AIAA Aerospace Sciences Meeting, Kissimmee, FL, January 8–12, 2018.

16. Roget, B., Sitaraman, J., Lakshminarayan, V., and Wissink, A., "Prismatic Mesh Generation Using Minimum Distance Fields," Tenth International Conference on Computational Fluid Dynamics (ICCFD10-219), Barcelona, Spain, 2018.

17. Rai, R. K., Gopalan, H., Sitaraman, J., Mirocha, J. D., and Miller, W. O., "A code-independent generalized actuator line model for wind farm aerod ynamics over simple and complex terrain," *Environmental modelling & software*, Vol. 94, 2017, pp. 172–185.

18. Peskin, C. S., "The immersed boundary method," *Acta numerica*, Vol. 11, 2002, pp. 479–517.

19. Jude, D. P., Sitaraman, J., and Brazell, M., "Enhancements to Overset Methods for achieving higher accuracy and solution convergence," Scitech 2020, Orlando, Florida, January 6–11, 2020.

20. Williams, J., "Modern Fortran Edition of SLSQP Optimizer," Online; accessed 2019.

21. Kraft, D., "A Software Package for Sequential Quadratic Programming," Technical Report DFVLR-FB 88-28, Institut für Dynamik der Flugsysteme, 1988.

22. Russell, C., and Sekula, M., "Comprehensive Analysis Modeling of Small-Scale UAS Rotors," AHS 73rd Annual Forum, Fort Worth, TX, May 2017.

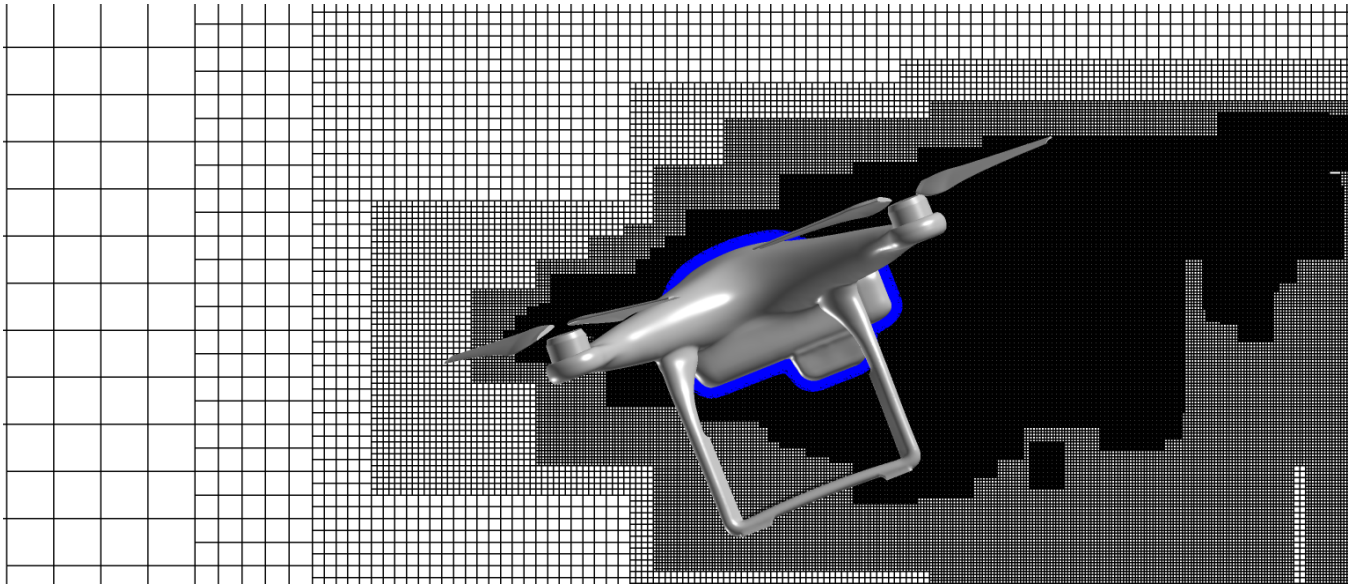
23. Russell, C., Willink, G., Theodore, C., Jung, J., and Glasner, B., "Wind Tunnel and Hover Performance Test Results for Multicopter UAS Vehicles," Technical Report TM—2018–219758, NASA, 2018.

24. UberElevate, "eVTOL Common Reference Models," Online; accessed 2019.

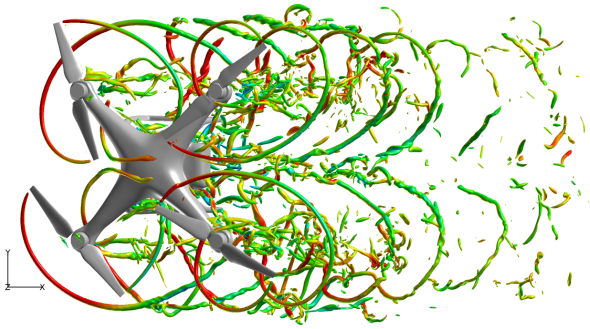
25. UberElevate, "Fast-Forwarding to a Future of On-Demand Urban Air Transportation," Online; accessed 2019.

26. Duffy, M. J., Wakayama, S., Hupp, R., Lacy, R., and Stauffer, M., "A Study in Reducing the Cost of Vertical Flight with Electric Propulsion," AHS 73rd Annual Forum, Fort Worth, TX, May 2017.

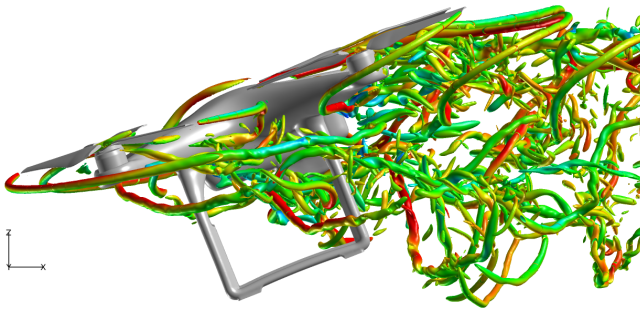
27. Polaczyk, N., Trombino, E., Wei, D. P., and Mitici, M., "A Review of Current Technology and Research in Urban On-Demand Air Mobility Applications," VFS Autonomous VTOL Technical meeting and Electric VTOL Symposium, Mesa, Arizona, January 2019.



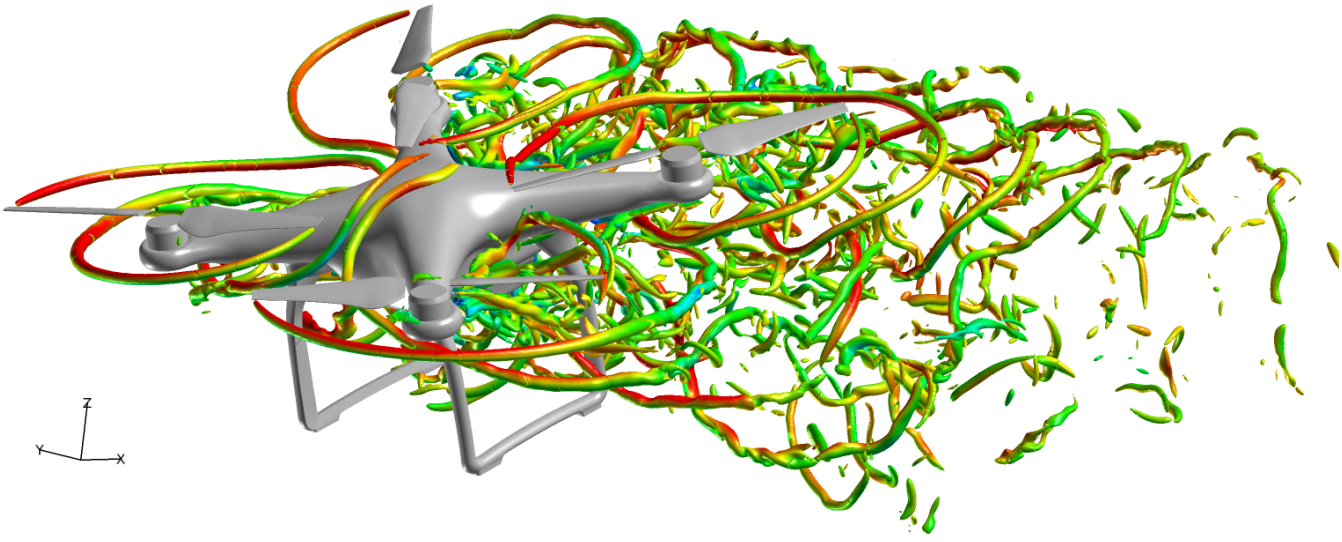
(a) Mesh system



(b) Top view

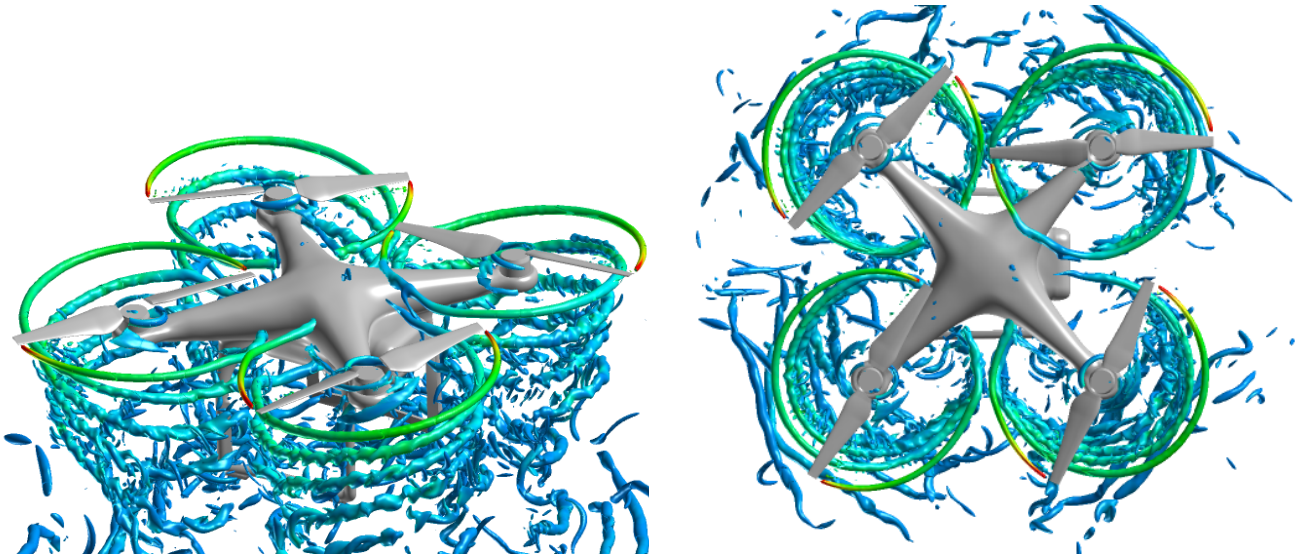


(c) Side view



(d) Oblique view

Figure 7: Flow visualization of the DJI Phantom 3 in forward flight at 10 m/s, with isosurfaces of q-criterion colored by velocity magnitude (red indicates higher magnitude).



(a) Oblique view (b) Top view

Figure 8: Flow visualization of the DJI Phantom 3 hover, with isosurfaces of q-criterion colored by vorticity magnitude (red indicates higher magnitude).

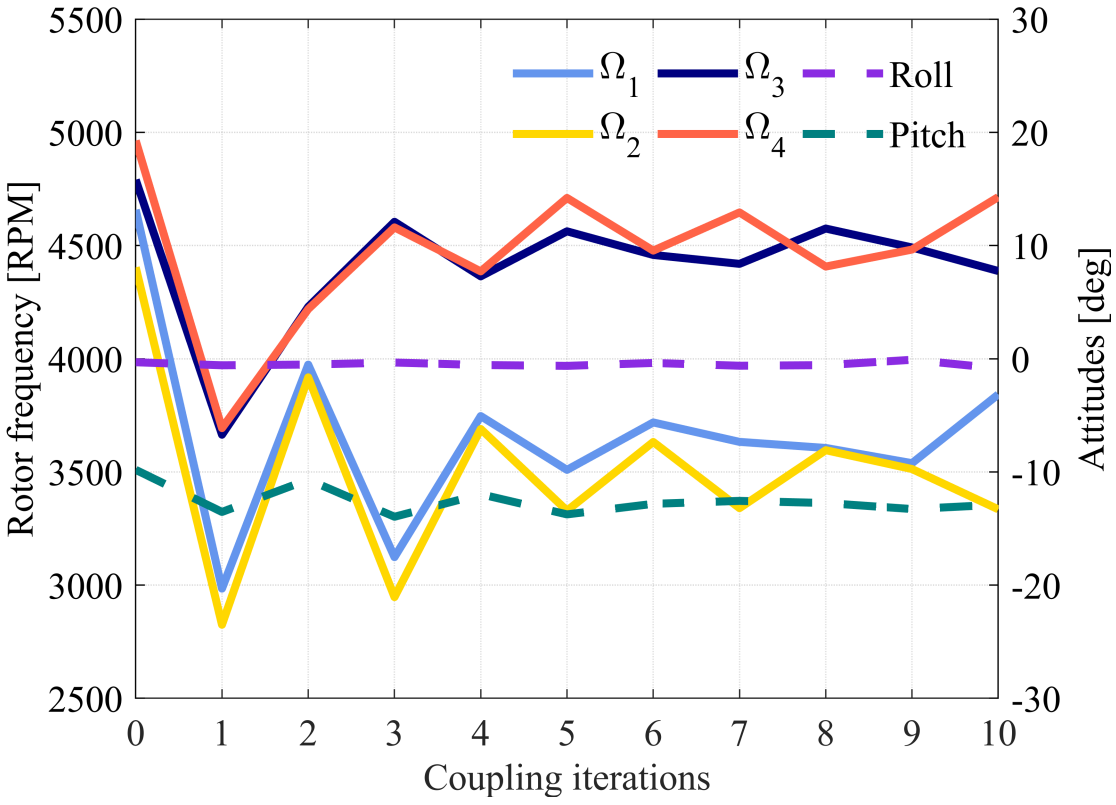


Figure 9: Convergence of trim variables for the DJI Phantom 3 at 10 m/s.

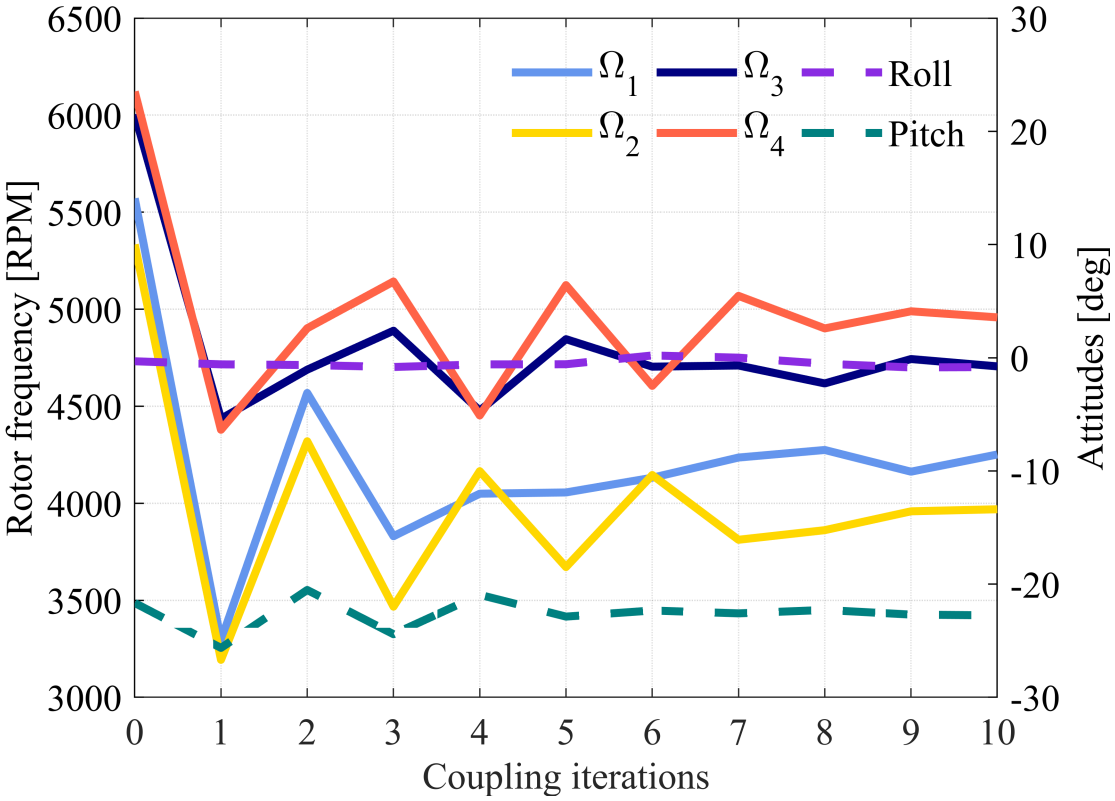


Figure 10: Convergence of trim variables for the DJI Phantom 3 at 15 m/s.

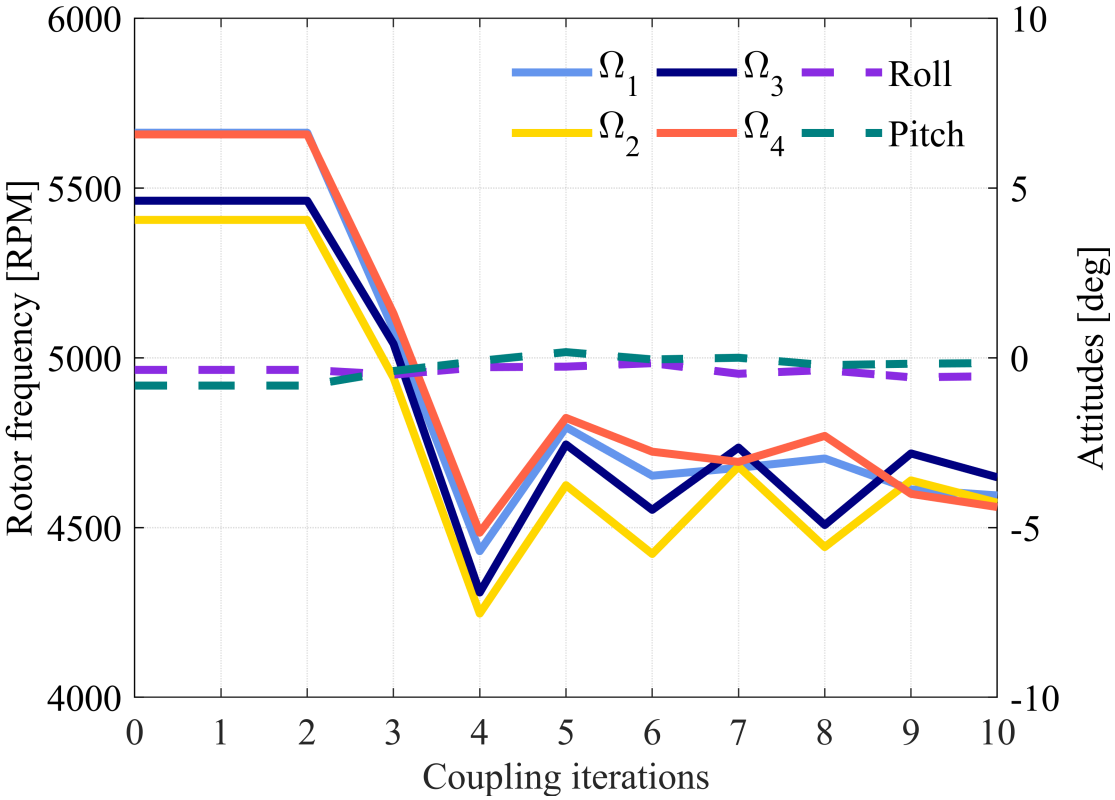


Figure 11: Convergence of trim variables for the DJI Phantom 3 in hover.

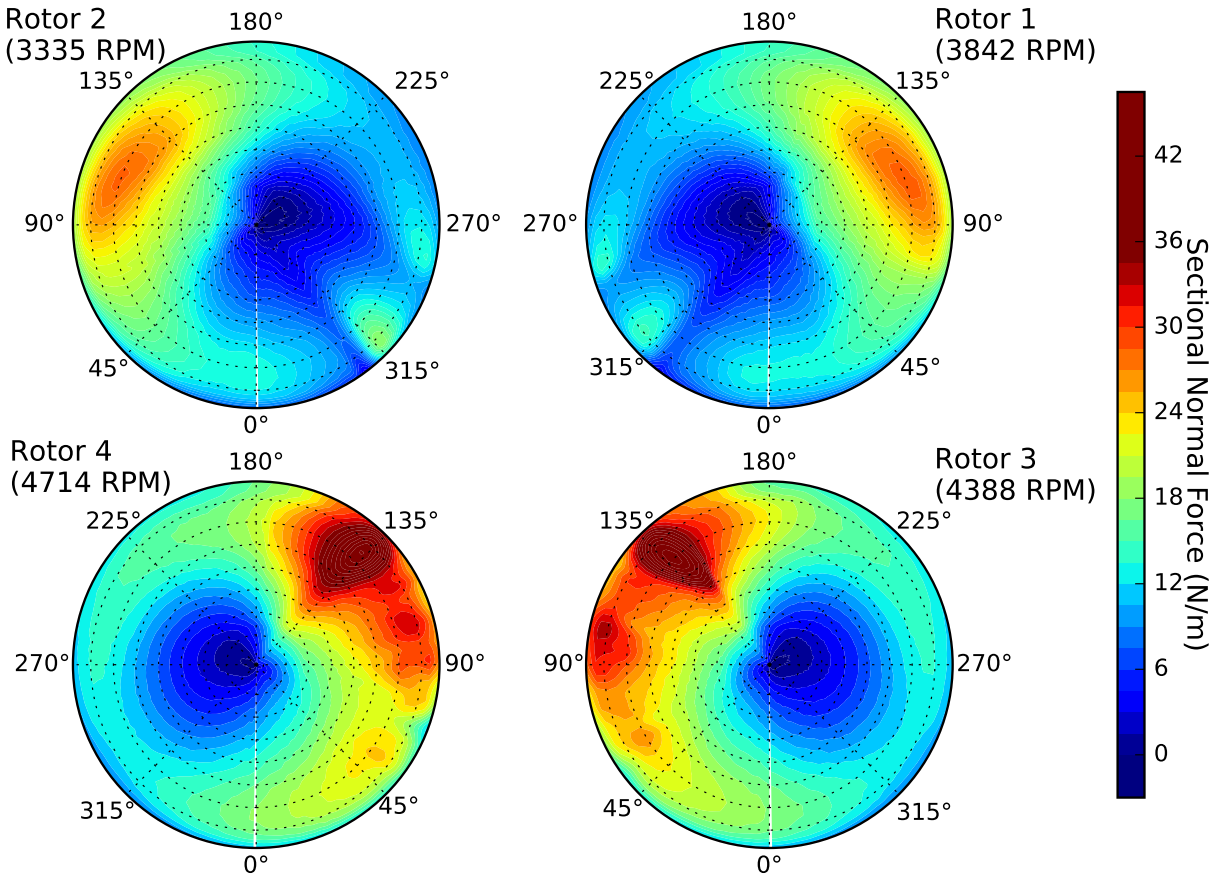


Figure 12: Airloads over the last coupling iteration for the DJI Phantom in forward flight at 10 m/s.

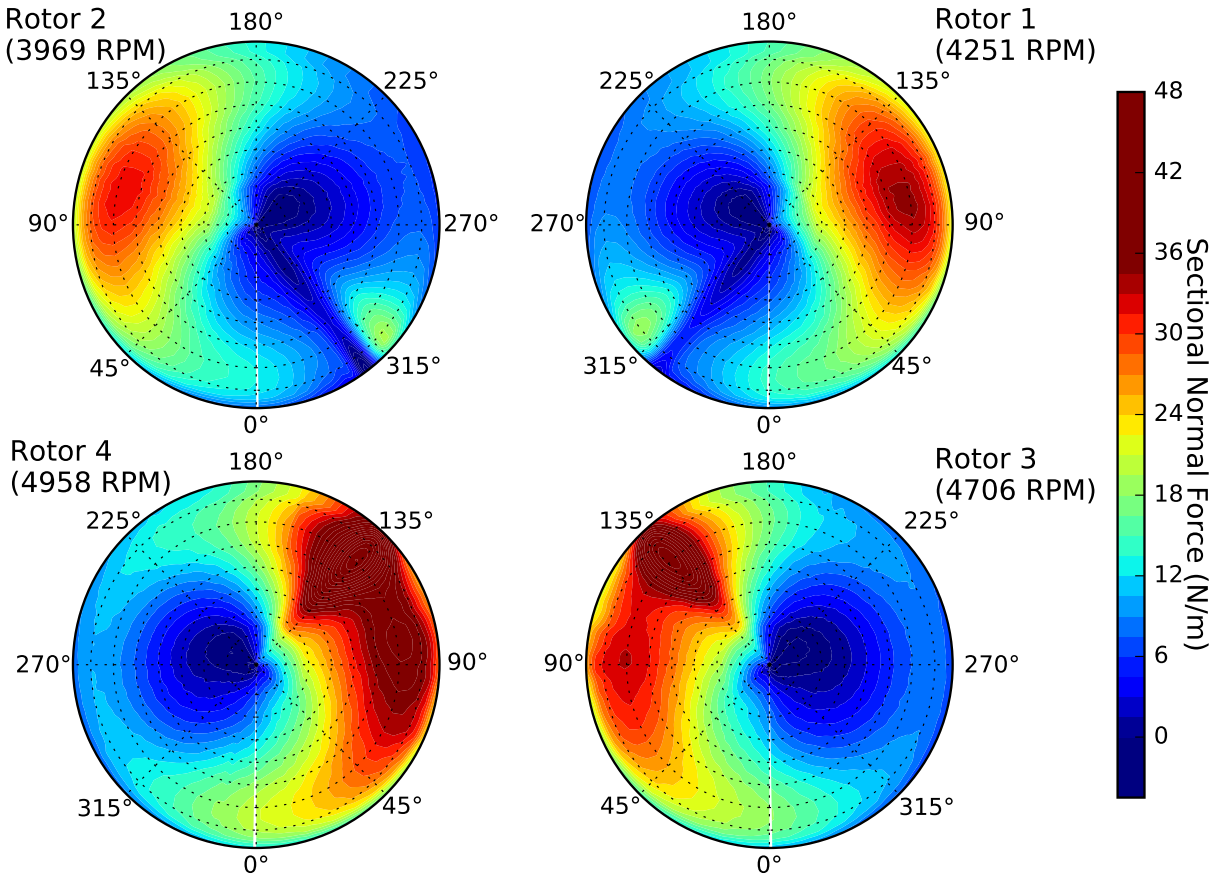


Figure 13: Airloads over the last coupling iteration for the DJI Phantom in forward flight at 15 m/s.

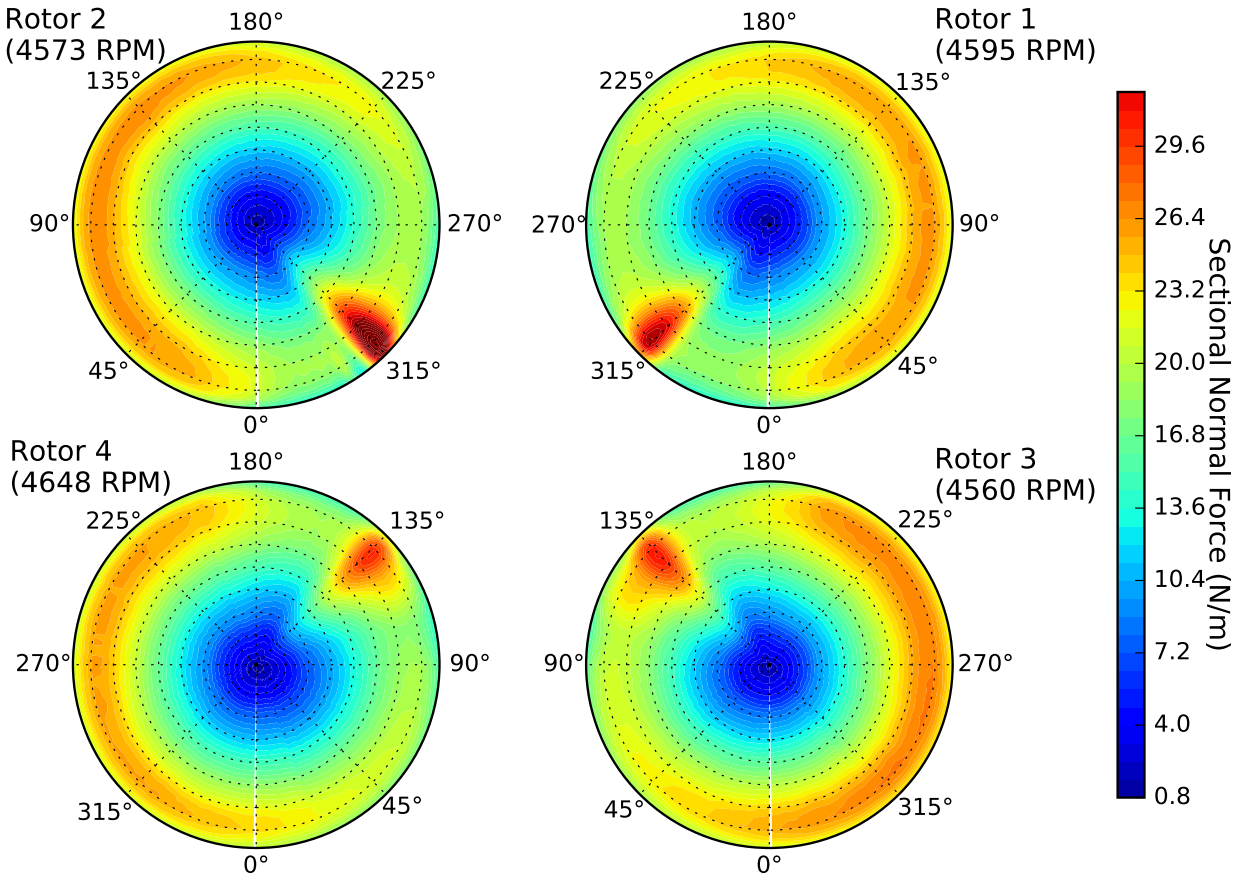
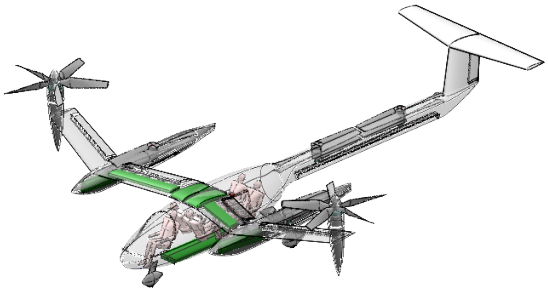
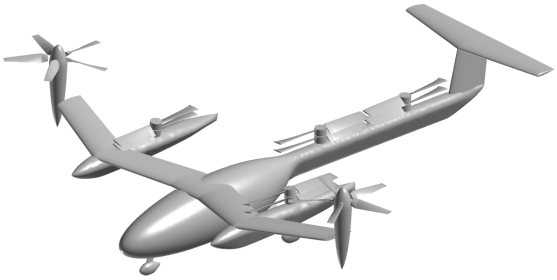


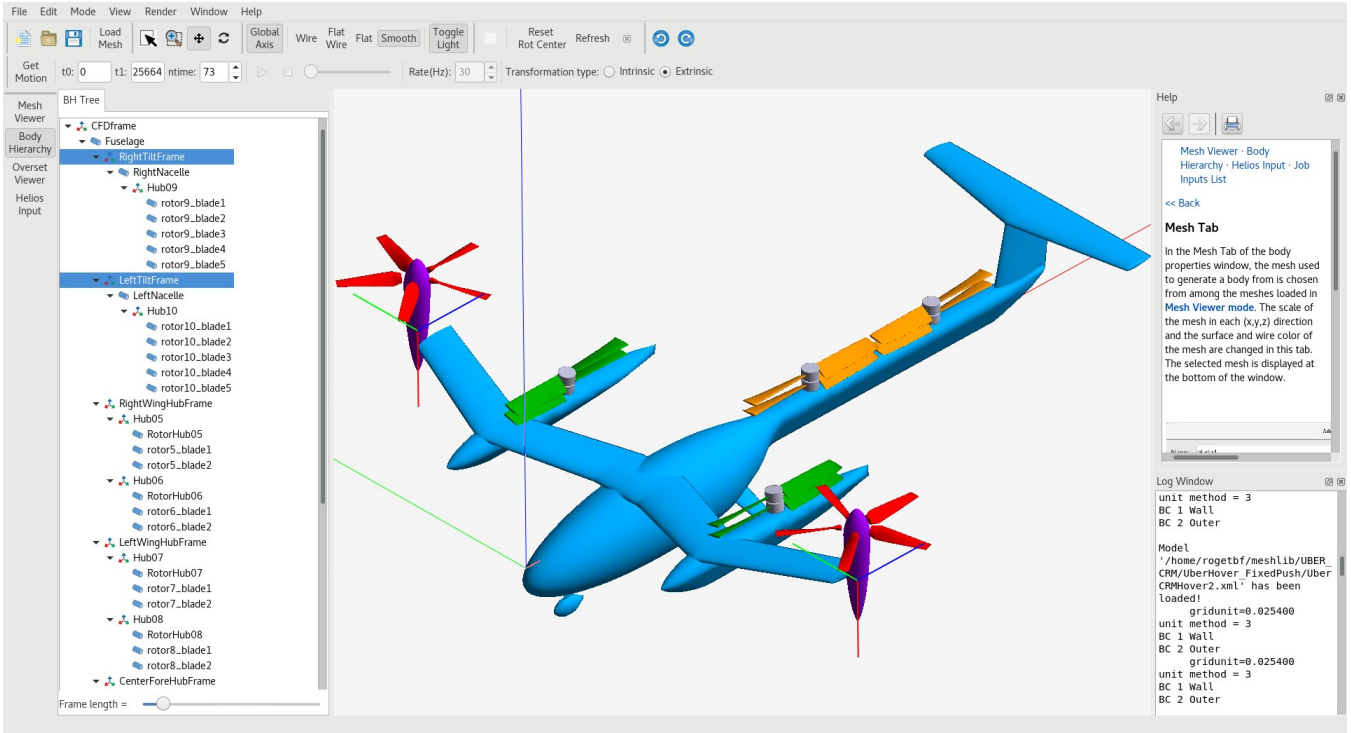
Figure 14: Airloads over the last coupling iteration for the DJI Phantom in hover.



(a) OpenVSP rendering.



(b) Simplified geometry.



(c) Uber eCRM-001 model in HIGEN.

Figure 15: Various views of the Uber eCRM-001 vehicle geometry.

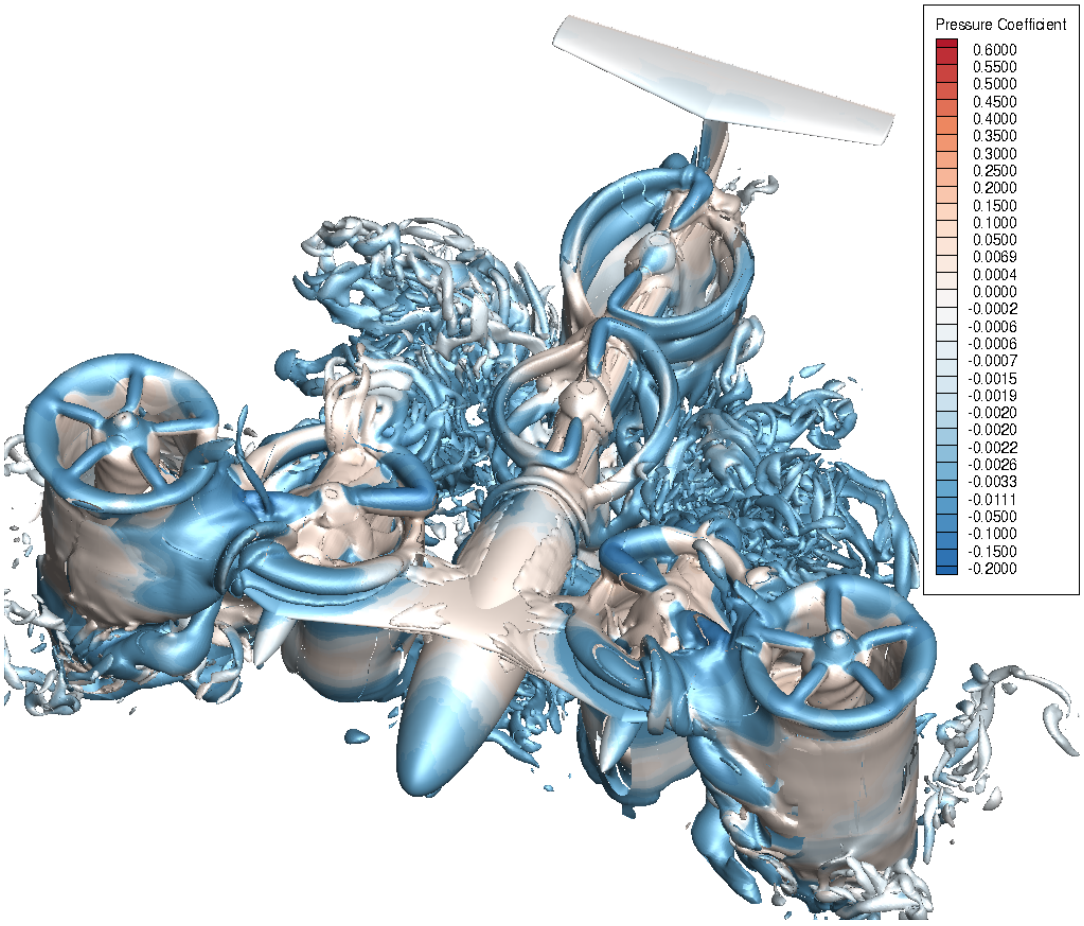


Figure 16: Simulation of Uber eCRM-001 in hover using ROAM. Iso-surfaces of vorticity magnitude are colored by pressure coefficient.