

Best Practices for Predicting UAM Noise with CAMRAD II and CHARM for AARON

*Lauren Weist
Natasha Schatzman
Nathan Burnside
Dorsa Shirazi*

*Ames Research Center
Moffett Field, California*

NASA STI Program . . . in Profile

Since its founding, NASA has been dedicated to the advancement of aeronautics and space science. The NASA scientific and technical information (STI) program plays a key part in helping NASA maintain this important role.

The NASA STI Program operates under the auspices of the Agency Chief Information Officer. It collects, organizes, provides for archiving, and disseminates NASA's STI. The NASA STI Program provides access to the NASA Aeronautics and Space Database and its public interface, the NASA Technical Report Server, thus providing one of the largest collections of aeronautical and space science STI in the world. Results are published in both non-NASA channels and by NASA in the NASA STI Report Series, which includes the following report types:

- **TECHNICAL PUBLICATION.** Reports of completed research or a major significant phase of research that present the results of NASA programs and include extensive data or theoretical analysis. Includes compilations of significant scientific and technical data and information deemed to be of continuing reference value. NASA counterpart of peer-reviewed formal professional papers, but having less stringent limitations on manuscript length and extent of graphic presentations.
- **TECHNICAL MEMORANDUM.** Scientific and technical findings that are preliminary or of specialized interest, e.g., quick release reports, working papers, and bibliographies that contain minimal annotation. Does not contain extensive analysis.
- **CONTRACTOR REPORT.** Scientific and technical findings by NASA-sponsored contractors and grantees.

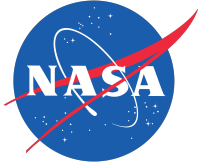
- **CONFERENCE PUBLICATION.** Collected papers from scientific and technical conferences, symposia, seminars, or other meetings sponsored or co-sponsored by NASA.
- **SPECIAL PUBLICATION.** Scientific, technical, or historical information from NASA programs, projects, and missions, often concerned with subjects having substantial public interest.
- **TECHNICAL TRANSLATION.** English-language translations of foreign scientific and technical material pertinent to NASA's mission.

Specialized services also include creating custom thesauri, building customized databases, and organizing and publishing research results.

For more information about the NASA STI Program, see the following:

- Access the NASA STI program home page at <http://www.sti.nasa.gov>
- E-mail your question to help@sti.nasa.gov
- Fax your question to the NASA STI Information Desk at 443-757-5803
- Phone the NASA STI Information Desk at 443-757-5802
- Write to:
STI Information Desk
NASA Center for AeroSpace Information
7115 Standard Drive
Hanover, MD 21076-1320

NASA/TM-20250009389



Best Practices for Predicting UAM Noise with CAMRAD II and CHARM for AARON

Lauren Weist

Natasha Schatzman

Nathan Burnside

Dorsa Shirazi

Ames Research Center

Moffett Field, California

National Aeronautics and
Space Administration

Ames Research Center
Moffett Field
California
94035

March 2026

Acknowledgments

The authors would like to thank the following people for their technical advice and assistance on developing these best practices: Dr. Doug Boyd, Dr. Wayne Johnson, and Dr. Leonard Lopes. We would also like to thank Dr. Bill Warmbrodt for his professional guidance over the course of this work. This work was funded under the NASA Revolutionary Vertical Lift Technology project.

The use of trademarks or names of manufacturers in this report is for accurate reporting and does not constitute an official endorsement, either expressed or implied, of such products or manufacturers by the National Aeronautics and Space Administration.

Available from:

NASA Center for AeroSpace Information
7115 Standard Drive
Hanover, MD 21076-1320

National Technical Information Service
5301 Shawnee Road
Alexandria, VA 22312

Available electronically at <http://www.sti.nasa.gov>

Best Practices for Predicting UAM Noise with CAMRAD II and CHARM for AARON

Lauren Weist

Natasha Schatzman

Nathan Burnside

Dorsa Shirazi

National Aeronautics and Space Administration
Ames Research Center
Moffett Field, CA 94035

Summary

As eVTOL vehicles become more prominent in the US market, the need for accurate rotor noise prediction increases. The NASA Revolutionary Vertical Lift Technology project aims to address this issue by developing the NASA RVLТ Conceptual Design Toolchain. As a part of that Toolchain, two mid-fidelity codes, CAMRAD II and CHARM, can be used with the rotor noise prediction tool AARON to complete these predictions. As these tools can often be complicated to use, these best practices were developed to ensure predictions are completed correctly for more accurate results. The best practices are described in this report and are then applied to three models: a simple rotor and two NASA RVLТ reference vehicles: the Quiet Single Main Rotor and the Side-by-Side. Additionally, an example of using experimental data to validate the codes is presented for a DJI Phantom 3 rotor.

Contents

1	Introduction	9
1.1	Toolchain	9
1.2	Reference vehicles	11
1.3	Goals for best practices	11
2	Methods	12
2.1	Tools	12
2.1.1	CAMRAD II	12
2.1.2	CHARM	13
2.1.3	AARON	13
2.1.4	Python libraries	14
2.2	Vehicle models	14
2.2.1	The simple rotor model	14
2.2.2	The QSMR model	15
2.2.3	The Side-by-Side model	17
3	Comprehensive code inputs	18
3.1	Methodology of increasing complexity	18
3.2	Airfoils	19
3.3	Blade models	19
3.4	Rotor wake	20
3.5	Trim settings	22
3.6	Increased fidelity for acoustics	23
3.6.1	Free wake	23
3.6.2	Azimuthal resolution	23
3.6.3	Trim convergence and revolutions	25
3.7	Multirotor considerations	26
4	Acoustic code inputs	26
4.1	Thickness and loading noise	26
4.2	Broadband self-noise	27
5	Outputs and visuals	31
5.1	Commonly used output variables	31
5.2	Metadata	31
5.2.1	Iso-surfaces	31
5.2.2	Disk plots	32
5.3	Acoustic plot types	32
6	Application of best practices	34
6.1	Microphone placement	34
6.2	Noise results for each vehicle	35
6.2.1	Simple rotor	36

6.2.2	QSMR	42
6.2.3	Side-by-Side	47
7	Acoustic prediction for experimental comparison	52
7.1	Experimental isolated rotor in hover	52
7.1.1	Testing facility	52
7.1.2	Rotor geometry	52
7.1.3	Rotor test set-up and operation	52
7.1.4	Rotor control	53
7.1.5	Acoustic setup and acquisition	55
7.1.6	Acoustic data processing	58
7.2	Simulation setup	59
7.2.1	Modeling assumptions	59
7.2.2	Comprehensive analysis settings	59
7.2.3	Performance validation	60
7.3	Experimental and computational acoustic results and validation	63
8	Concluding remarks	65
A	Input files for each model	68
A.1	Simple rotor	68
A.2	QSMR	82
A.3	Side-by-Side	103
A.4	DJI Phantom 3	128
B	Python scripts	152
C	MATLAB scripts	164

List of Figures

1	NASA RVL T Conceptual Design Toolchain workflow from Ref. 1.	10
2	Simple rotor geometric mock-up in OpenVSP for visualization purposes.	15
3	QSMR mock-up in OpenVSP.	16
4	Side-by-Side visualization in OpenVSP, only showing the parts of the vehicle modeled.	17
5	Blade panel study and graphical representation for the QSMR in CAMRAD II ($M_{tip} = 0.63$, $\frac{C_T}{\sigma} = 0.07$, $\mu = 0.25$).	21
6	Blade panel study and graphical representation for the QSMR in CHARM ($M_{tip} = 0.63$, $\frac{C_T}{\sigma} = 0.07$, $\mu = 0.25$).	22
7	The progression of wake modeling for CAMRAD II and CHARM cases.	23
8	Graphical representation of the spanwise and azimuthal spacing for aerodynamic inflow calculations in CAMRAD II. Black represents the final computation done by post-trim, while red represents the initial azimuthal calculation.	24
9	Airfoil geometry inputs for BPM model shown on an example airfoil section.	28
10	BPM parameter study for QSMR in forward flight free wake for cases 1 through 7 for total broadband self-noise ($V_{inf} = 53.9$ m/s, $\frac{C_T}{\sigma} = 0.07$, $\mu = 0.25$, $\alpha_s = -3^\circ$, $\theta_0 = 4^\circ$).	30
11	QSMR forward flight free wake, single blade z loading metadata for CAMRAD II and CHARM ($V_{inf} = 53.9$ m/s, $\frac{C_T}{\sigma} = 0.07$, $\mu = 0.25$, $\alpha_s = -3^\circ$, $\theta_0 = 4^\circ$).	33
12	Microphone locations for the Side-by-Side graphically represented. The wind is moving in the negative y direction as shown.	35
13	Thickness noise acoustic pressure time history with CAMRAD II and CHARM using AARON for the simple rotor in forward flight with a free wake model ($V_{inf} = 33.5$ m/s, $\frac{C_T}{\sigma} = 0.001$, $\mu = 0.14$, $\alpha = 5^\circ$, $\theta_0 = -1^\circ$).	38
14	Loading noise acoustic pressure time history with CAMRAD II and CHARM using AARON for the simple rotor in forward flight with a free wake model ($V_{inf} = 33.5$ m/s, $\frac{C_T}{\sigma} = 0.001$, $\mu = 0.14$, $\alpha = 5^\circ$, $\theta_0 = -1^\circ$).	39
15	Broadband self-noise spectra with CAMRAD II and CHARM using AARON for the simple rotor in forward flight with a free wake model ($V_{inf} = 33.5$ m/s, $\frac{C_T}{\sigma} = 0.001$, $\mu = 0.14$, $\alpha = 5^\circ$, $\theta_0 = -1^\circ$).	40
16	Total noise spectra with CAMRAD II and CHARM using AARON for the simple rotor in forward flight with a free wake model ($V_{inf} = 33.5$ m/s, $\frac{C_T}{\sigma} = 0.001$, $\mu = 0.14$, $\alpha = 5^\circ$, $\theta_0 = -1^\circ$).	41
17	Thickness noise acoustic pressure time history with CAMRAD II and CHARM using AARON for the QSMR in forward flight with a free wake model ($V_{inf} = 54$ m/s, $\frac{C_T}{\sigma} = 0.08$, $\mu = 0.25$, $\alpha = 1.4^\circ$, $\theta_0 = 4.7^\circ$).	43
18	Loading noise acoustic pressure time history with CAMRAD II and CHARM using AARON for the QSMR in forward flight with a free wake model ($V_{inf} = 54$ m/s, $\frac{C_T}{\sigma} = 0.08$, $\mu = 0.25$, $\alpha = 1.4^\circ$, $\theta_0 = 4.7^\circ$).	44
19	Broadband self-noise spectra with CAMRAD II and CHARM using AARON for the QSMR in forward flight with a free wake model ($V_{inf} = 54$ m/s, $\frac{C_T}{\sigma} = 0.08$, $\mu = 0.25$, $\alpha = 1.4^\circ$, $\theta_0 = 4.7^\circ$).	45

20	Total noise spectra with CAMRAD II and CHARM using AARON for the QSMR in forward flight with a free wake model ($V_{inf} = 54$ m/s, $\frac{C_T}{\sigma} = 0.08$, $\mu = 0.25$, $\alpha = 1.4^\circ$, $\theta_0 = 4.7^\circ$).	46
21	Thickness noise acoustic pressure time history with CAMRAD II and CHARM using AARON for the Side-by-Side in forward flight with a free wake model ($V_{inf} = 25.72$ m/s, $\frac{C_T}{\sigma} = 0.093$, $\mu = 0.15$, $\alpha = 0.6^\circ$, $\theta_0 = 6^\circ$).	48
22	Loading noise acoustic pressure time history with CAMRAD II and CHARM using AARON for the Side-by-Side in forward flight with a free wake model ($V_{inf} = 25.72$ m/s, $\frac{C_T}{\sigma} = 0.093$, $\mu = 0.15$, $\alpha = 0.6^\circ$, $\theta_0 = 6^\circ$).	49
23	Broadband self-noise spectra with CAMRAD II and CHARM using AARON for the Side-by-Side in forward flight with a free wake model ($V_{inf} = 25.72$ m/s, $\frac{C_T}{\sigma} = 0.093$, $\mu = 0.15$, $\alpha = 0.6^\circ$, $\theta_0 = 6^\circ$).	50
24	Total noise spectra with CAMRAD II and CHARM using AARON for the Side-by-Side in forward flight with a free wake model ($V_{inf} = 25.72$ m/s, $\frac{C_T}{\sigma} = 0.093$, $\mu = 0.15$, $\alpha = 0.6^\circ$, $\theta_0 = 6^\circ$).	51
25	Photograph of the two-bladed DJI Phantom 3 rotor.	53
26	DJI Phantom 3 airfoil distribution from an in-house 3D scan.	53
27	DJI Phantom 3 blade chord and twist distributions from an in-house 3D scan.	54
28	Left: DJI Phantom 3, Right: DJI Phantom 3 isolated rotor mounted in the anechoic chamber.	54
29	Pictorial top view of microphone and rotor locations in the anechoic chamber (not drawn to scale).	55
30	Visual of DJI Phantom 3 isolated rotor in anechoic chamber with installation of microphone.	55
31	Front view of 8-inch array with microphones flush mounted in a non-conductive plate.	56
32	RPM for each 32-second data point including the following phases: a) 4 seconds of rotor settle time, b) 2 seconds of RPM ramp and stabilization, c) 24 seconds of analysis, and d) 2 seconds of ramp down.	57
33	Circulation checkout test for DJI Phantom 3 isolated rotor at 8,000 RPM in anechoic chamber with exhaust nozzle blocked.	57
34	In-plane near- and far-field experimental results for three rotor RPMs.	58
35	Electrical power versus RPM for isolated DJI Phantom 3 in hover for experiment and simulation.	60
36	Thrust versus RPM for isolated DJI Phantom 3 in hover for experiment and simulation.	61
37	DJI Phantom 3 isolated rotor free wake, single blade z loading metadata for CAMRAD II and CHARM for a.) 4,000 RPM, b.) 6,000 RPM, and c.) 8,000 RPM.	62
38	Unfiltered and filtered experimental, CAMRAD II, and CHARM acoustic time histories for the DJI Phantom 3 isolated rotor for a) 4,000 RPM, b) 6,000 RPM, and c) 8,000 RPM.	64

List of Tables

1	Simple rotor major model parameters.	15
2	QSMR rotor major model parameters.	16
3	Side-by-Side major model parameters. Airfoil and blade chord length variables vary over the length of the blade. For specific cord length and airfoil distribution locations, see Appendix A.3.	18
4	Broadband self-noise parameter namelist variables. ASNIFM requires inputs for all rotors, thus the N notation in the pyaaron variable. The N should be replaced with the rotor number that is being considered. *Input is for thickness calculations, but located in the ASNIFM inputs section, and is therefore included for completeness.	27
5	QSMR BPM inputs for ASNIFM.	29
6	QSMR ASNIFM input parameter study cases.	29
7	Microphone locations for the simple rotor, QSMR, and Side-by-Side in terms of both angle and Cartesian coordinates with the origin being the center of gravity of the vehicle.	36

Nomenclature

Acronyms

AAM	Advanced Air Mobility
AARON	ANOPP2 Aeroacoustic Rotor Noise tool
AF1AIFM	ANOPP2 Farassat Formulation 1A Internal Functional Module
ANOPP2	Aircraft NOise Prediction Program - Second Generation
ASNIFM	ANOPP2 Self Noise Internal Function Module
BPM	Brooks-Pope-Marcolini semi-empirical model
BVI	Blade-Vortex Interaction
BWI	Blade-Wake Interaction
CAMRAD II	Comprehensive Analytical Model of Rotorcraft Aerodynamics and Dynamics II
CFD	Computational Fluid Dynamics
CG	Center of Gravity
CHARM	Comprehensive Hierarchical Aeromechanics Rotorcraft Model
ESC	Electronic Speed Controller
eVTOL	electric Vertical Take-Off and Landing
F1A	Farassat's Formulation 1A solution to the Ffowcs Williams-Hawkings equation
FAA	Federal Aviation Administration
NACA	National Advisory Committee for Aeronautics
NDARC	The NASA Design and Analysis of RotorCraft
NOTAR	'No Tail Rotor' anti-torque system
OASPL	OverAll Sound Pressure Level
OpenVSP	Open Vehicle Sketch Pad parametric geometry design tool
QSMR	Quiet Single Main Rotor
RCOTools	RotorCraft Optimization Tools
RPM	Revolutions Per Minute
RVLT	Revolutionary Vertical Lift Technology
SPL	Sound Pressure Level
UAM	Urban Air Mobility

Variables

BL_T	Boundary layer trip setting
c	Airfoil chord length [ft]
C_L	Coefficient of lift
$\frac{C_T}{\sigma}$	Coefficient of thrust normalized by rotor solidity
F_Z	Rotor force in the z direction [lbs]
h	Airfoil maximum height [ft]
M_{tip}	Blade tip Mach number
R	Rotor radius [ft]
r/R	Blade radial station
t/c	Thickness-to-chord ratio
V_{ind}	Rotor induced velocity [ft/s]
V_{inf}	Free stream velocity [ft/s]

α_0	Zero lift angle of attack [deg]
α_{eff}	Effective angle of attack [deg]
β_{1C}	Longitudinal tip path plane flapping angle [deg]
β_{1S}	Lateral tip path plane flapping angle [deg]
θ_0	Rotor collective angle [deg]
θ_{TE}	Airfoil trailing edge wedge angle [deg]
μ	Advance ratio
σ	Geometric solidity
ϕ	Microphone elevation angle [deg]
ψ	Microphone azimuth angle [deg]
Ω	Rotor rotation speed [rad/s]

1 Introduction

The NASA Revolutionary Vertical Lift Technology (RVLT) project aims to set the standard of rotorcraft design for the U.S. Government, industry, and academia for both traditional helicopters and new UAM/AAM configurations. In order to do this, RVLT created the Conceptual Design Toolchain, a suite of codes aiming to guide a designer through all stages of the conceptual design process. This toolchain includes the ability to conduct acoustic predictions, which is of particular interest to the Urban Air Mobility (UAM) field, as many UAM vehicles will be operated in public areas where noise complaints could result if acoustics is not considered. In response to this interest, this report aims to create a set of best practices for using the Toolchain for acoustic prediction with comprehensive analysis. This stage in the Toolchain was selected for the development of best practices because it is the earliest stage in the Toolchain where one can obtain data of a high enough fidelity to use for acoustic prediction, and because it is the best stage currently for iterating on an acoustic design. Currently, the Toolchain contains a choice of two industry codes for comprehensive analysis, while acoustic prediction is completed using U.S. Government developed software.

The current Toolchain is shown in Figure 1, and these best practices were developed for steps 2 and 3 of the process. Step 2 of the Toolchain is the comprehensive analysis step and can currently be performed using either the Comprehensive Analytical Model of Rotorcraft Aerodynamics and Dynamics II (CAMRAD II) (Ref. 2) or the Comprehensive Hierarchical Aeromechanics Rotorcraft Model (CHARM) (Ref. 3). Step 3 of the Toolchain is the acoustic prediction step and can be conducted with the Aircraft NOise Prediction Program - Second Generation (ANOPP2) tool, Aeroacoustic Rotor Noise (AARON) tool, and the Python wrapper, pyaaron (Ref. 4).

In order to make these codes more accessible to a wider audience, a published document of best practices for acoustic prediction with comprehensive analysis was desired. While each code has a user’s manual, and there are several papers using this portion of the Toolchain (some notable publications are Silva (Ref. 1), Kottapalli (Ref. 5, 6), and Weist (Ref. 7)), this document is the most comprehensive “best practices” paper to date. It is important to note that while these best practices focus on two comprehensive codes, the goal of this publication is not to compare the accuracy of CAMRAD II and CHARM, but instead to develop the ideal running parameters for either code with acoustic prediction. While the two codes will be shown against each other, the differences in results will not be discussed except for when it is due to specific settings that are necessary for running the code. As these best practices were developed specifically for use with the RVLT Toolchain, many of the examples presented are vehicles developed as NASA RVLT reference vehicles, with exception of the initial simplified rotor and the DJI Phantom 3, which were selected for other reasons. More information on the reference vehicles and those used in this work are presented in section 1.2.

1.1 Toolchain

The NASA RVLT Conceptual Design Toolchain, shown in Figure 1 focuses on the series of steps a designer must perform to complete the conceptual design of a rotorcraft. The complete Toolchain contains multiple codes covering different steps of the conceptual design space. The work presented in this paper focuses on two steps of the Toolchain: comprehensive analysis and acoustic prediction, shown in boxes 2 and 3 in Figure 1, but all 3 steps will be described in this section for completeness.

The Toolchain has three major steps: the vehicle design step, the comprehensive analysis step,

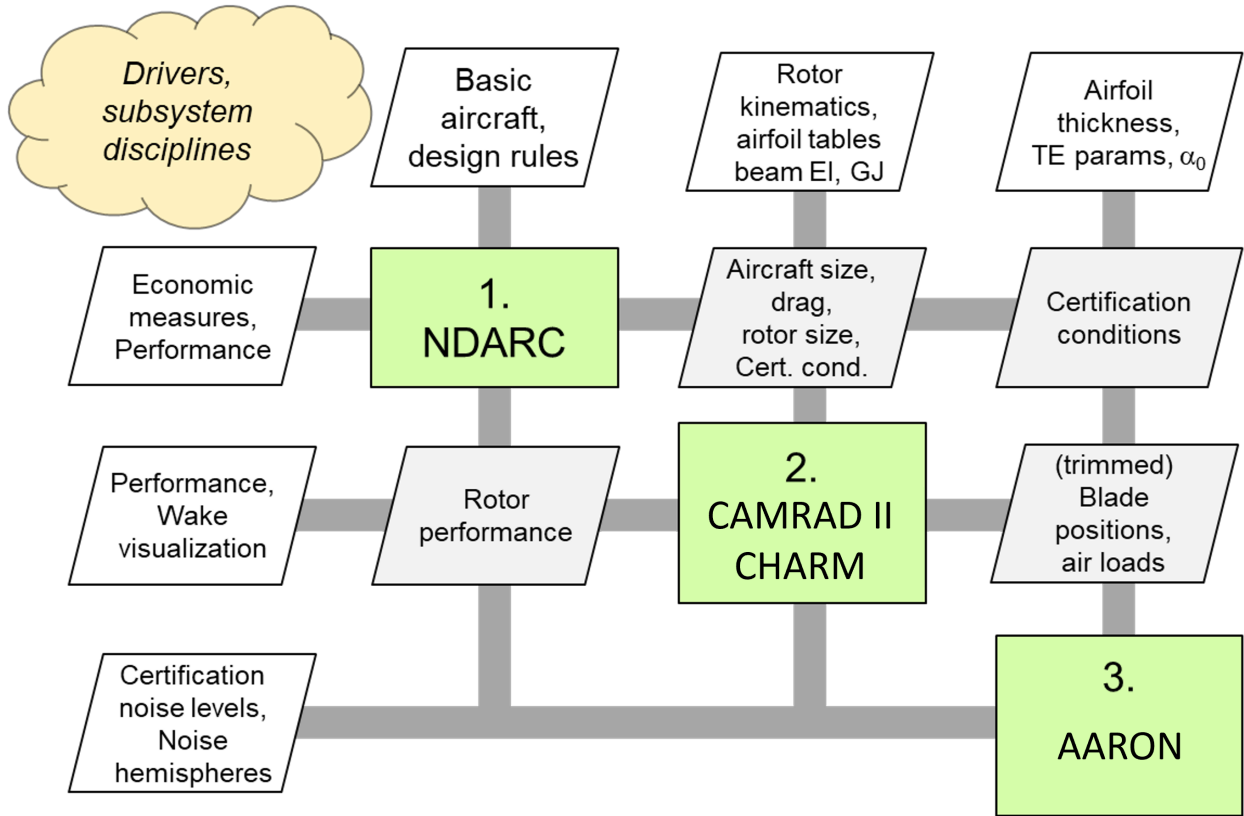


Figure 1.—NASA RVL Conceptual Design Toolchain workflow from Ref. 1.

and the acoustics step. The vehicle design step, step 1, is comprised of the NASA-developed code NDARC (“The NASA Design and Analysis of RotorCraft”). NDARC is a low fidelity sizing tool designed to help an engineer test multiple variables to create a rotorcraft capable of conducting a mission, or set of specific missions. NDARC, while an important step of the Toolchain, is not capable of producing data that can be used for acoustic prediction, due to its low fidelity nature. For acoustic prediction purposes, it is instead recommended to size a vehicle with NDARC, then move the vehicle to step 2, where these best practices will aid in turning the model into one that can be used for acoustic prediction (Ref. 8).

The second step of the Toolchain, comprehensive analysis, contains multidisciplinary computational tools that combine dynamics, structures, and aerodynamics to predict various characteristics of a rotorcraft, such as performance, dynamics, and handling qualities. Comprehensive analysis can have varying levels of fidelity, and can predict a large amount of data that the skilled engineer can use (Ref. 9). In the Toolchain, the comprehensive analysis step builds upon the sized low-fidelity aircraft produced by NDARC and adds the details necessary to produce predictions of rotorcraft performance and behavior. As there is no current NASA-developed comprehensive analysis tool, the Toolchain instead includes two modern, industry-developed codes to act as the example, those being CAMRAD II and CHARM. This step is necessary to conduct acoustic prediction as acoustic prediction tools require high-resolution blade loading, geometry, and location data in order to conduct their computations. Only the comprehensive analysis results with relevance to the acous-

tics analysis are explored in these best practices, although both comprehensive analysis codes can perform many other types of analyses.

The final step in the Toolchain is acoustic prediction, which is completed by another NASA-developed code, AARON. Rotorcraft acoustic prediction codes utilize different formulations (or solutions) of the Ffowcs Williams-Hawkings equation, known as Farassat’s Formulations (Ref. 10). The most prevalent formulation used today is Farassat’s Formulation 1A. The compact versions of this Formulation are used in the current work presented here. This Formulation relates the blade motion and blade loading to acoustic pressure time histories at observer (microphone) locations. The Brooks-Pope-Marcolini model will also be used to compute broadband self-noise. These models will be described more in depth later in the paper. This step is the final stage in the conceptual design process, as noise is a large hurdle for UAM companies with regards to acceptance and FAA certification (Ref. 11). Using this code in conjunction with the comprehensive analysis code allows a designer to gain a full understanding as to how a specific rotorcraft design will perform, sound, and handle.

1.2 Reference vehicles

In addition to the Toolchain, the NASA RVLTL project has developed a series of UAM reference vehicles. The goal of these vehicles is to provide the UAM industry with representative vehicles for use with design and analysis tools. These vehicles are developed using the tools in the Toolchain, and therefore provide NASA-developed vehicles that can be easily published into data sets and tutorials. This data can then be used by the wider UAM community and by other government organizations to aid in UAM vehicle and UAM tool development.

These vehicles have multiple variations of each configuration. These variations include number of occupants, control schemes, and powerplants to explore the design space further (Ref. 12). For example, the vehicles have turboshaft, turboelectric, and battery versions to explore different powerplant options. The vehicle configurations include: the Quadrotor, the Side-by-Side, the Tiltwing, the Lift Plus Cruise, the Quiet Single Main Rotor (QSMR), the Tiltduct, and the Tiltrotor. These vehicles are periodically updated to align with industry standards, and new vehicles are also being developed (Ref. 13).

For this work, the best practices were focused on rotorcraft configurations with increasing modeling complexity in order to aid a new user in building understanding. Two of the four models selected were RVLTL reference vehicles: the Quiet Single Main Rotor and the Side-by-Side. Two additional models were also considered: a simplified rotor for initial calculations, and a DJI Phantom 3 rotor for experimental validation.

1.3 Goals for best practices

This report aims to develop best practices for the use of AARON with CAMRAD II or CHARM, as highlighted in the Toolchain. These codes are complicated, and they require a specific practice in order for them to interact correctly. Specific settings need to be used in both CAMRAD II and CHARM to produce data that leads to accurate acoustic predictions. These best practices are then compared with experimental results to ensure their accuracy. Additionally, this work aims to make this prediction process easier for new or novice users of this workflow and increase the standardization of practices. This information is intended to be available to a wide audience within academia, industry, and government.

It is important to note that these practices are not designed to compare the results of CAMRAD

II and CHARM. While both codes will be used to produce similar results, and those results will be shown together, comments related to determining the most accurate comprehensive analysis code are not made. Instead, the goal is to identify how to produce results using either code, with the decision of which code to use being up to the end user. Differences in models or capabilities will be noted when important, but this is purely informative and does not express any opinion on which code is “better.”

2 Methods

To approach the development of best practices, a methodology of increasing complexity was selected. This applied to both the vehicles selected and the models used within the codes. For both CAMRAD II and CHARM, the fidelity of the codes can be incrementally changed, so that the lower fidelity models need less complex user inputs, while the higher fidelity models provide more detailed results. These best practices walk a user through this methodology.

The vehicle models were also selected for increasing difficulty, beginning with a simple main rotor, then a NOTAR vehicle, then a multirotor configuration, before ending with an experimental case (which is entirely contained in its own section for clarity). The vehicles selected and the rationale for their selection are detailed later in this section.

2.1 Tools

This section details the important elements of each code as pertaining to the specific best practices. This section provides an introduction to each code, but for an high-detail description, it is recommended that a user reference the corresponding citations and users manual for each tool.

2.1.1 CAMRAD II

CAMRAD II, one of the two comprehensive analysis codes within the Toolchain, is a mid-fidelity tool developed by Johnson Aeronautics. This code can be used to compute rotor and rotorcraft performance, loads, vibration, response, and stability. For a full description of the capabilities of the tool, see Ref. 2. While this code has many use cases, in this effort, the trim solution is used to find the steady state operating point for the vehicle in a specified flight condition. This solution is then analyzed by the code to produce an aeromechanical analysis of the vehicle and its performance. A lifting line mode, steady two-dimensional airfoil tables, and a vortex wake is used for the majority of aerodynamic calculations. Corrections are also made for unsteady and yawed flow. For acoustics, the required outputs are blade loading and blade position information. These data are located in the output file for use by subsequent codes in the workflow. The blade loading lifting line model accounts for all inflow, rotor interactions, and wake interactions encountered by the rotor.

CAMRAD II uses FORTRAN namelist files for user input variables. These namelists are separated into .run and .list files. While these files function in the same way, they should be used intentionally to separate variables that change and those that are constant. Using the .list file to define the vehicle geometry, desired output structure, and atmospheric conditions, for example, is a way to ensure that variables that should not change can be kept separate from those that are changing. The .run file, on the other hand, can be used for variables that will change, such as trim conditions. The other inputs required are airfoil tables, which the CAMRAD II input program can

convert from C81 tables to .tab format. For examples of how CAMRAD II input files are organized, see Appendix A.

The CAMRAD II input files are sorted into blocks by class, type, and name. CAMRAD II has a set of standard inputs, which are referred to as scenarios and represented by the variable OPSCEN, which sets the variables for various standard rotor configurations. Then, the user can use the list and run files to alter the vehicle to match their specifications. If the desired configuration is not included in a scenario, then the user can use the core input methodologies to edit the ‘core’ rotor model. All work here is completed via changes to the outer inputs, the shell.

2.1.2 CHARM

The other comprehensive analysis option present in the RVL Toolchain is CHARM, developed by Continuum Dynamics, Inc. In this work, CHARM version 7.3 is used. CHARM is a mid-fidelity code that models rotorcraft and fixed wing dynamics and aerodynamics. Like CAMRAD II, CHARM uses lifting line models for the aerodynamics, although it can also use the vortex lattice lifting surface method. Additionally, CHARM uses the unique Constant Vorticity Contour method for wake modeling. These two methods combined create a free-vortex wake model capable of capturing 3D and interactional aerodynamic effects. CHARM is similar to CAMRAD II in terms of capabilities and uses but does differ some in architecture, such as with the user inputs. For more information on CHARM, see Ref. 3.

CHARM user inputs are structured using a series of .inp files, each with a specific layout and input structure, which is provided in the user’s manual and must be followed exactly. For the work conducted in these best practices, the following input files were used, where ‘[Name]’ is consistent among all files. It is recommended, but not required, to have the ‘[Name]’ be the name of the vehicle.

- Run Characteristics Input File ([Name].inp)
- Rotor Wake Input File ([Name]rw.inp)
- Blade Geometry Input File ([Name]bg.inp)
- Blade Dynamics Input File ([Name]bd.inp)
- 2-D Airfoil Section Data Input File ([Name]af.inp)
- Trim Input File ([Name]trim.inp)

This list does not include all of the possible input files that can be used in CHARM; instead, it lists the files used to develop these best practices, and is a good starting point for a new user. The information in these files must follow a prescribed format in order for the code to run. Because of this, it is recommended to start with the example files for a given configuration, then editing the file to match the desired vehicle.

2.1.3 AARON

Within the RVL Toolchain, acoustic prediction is completed by AARON. AARON is a FORTRAN user tool developed to interface with ANOPP2. ANOPP2 is a suite of tools for acoustic

prediction, and contains a framework, tools, and functional models used to compute noise. AARON reduces the workload of ANOPP2 for a rotorcraft specific use case, simplifying the ANOPP2 process to a single input file formatted in namelist format. AARON is able to account for multi-rotor configurations, and includes options for filtering and microphone distribution types such as single mic, line array, or hemisphere.

Specifically, AARON accesses the ANOPP2 Farassat Formulation 1A Internal Functional Module (AF1AIFM) to predict tonal noise and the ANOPP2 Self Noise Internal Function Module (ASNIFM) to predict broadband noise. AF1AIFM is an integral formulation of the Ffowcs Williams-Hawkings equations, known as Farassat’s Formulation 1A (F1A) (Ref. 10). This formulation is used to predict thickness and loading noise for rotating bodies, such as a rotor blades. Specifically, this work utilizes the compact thickness and compact loading models of F1A. ASNIFM is used to access a modified Brooks-Pope-Marcolini (BPM) semi-empirical model, as published in NASA RP 1218 (Ref. 14). As the original BPM model was designed for symmetric airfoils, the implementation used must correct for camber, sectional twist, and induced angle to generate an equivalent angle of attack at which the NACA 0012 would be. This allows for the low-fidelity prediction of broadband self-noise from the rotor (Ref. 4).

2.1.4 Python libraries

These codes were accessed using two specific Python libraries, RCOTools and pyaaron. RCOTools is a suite of Python utilities and wrappers designed to facilitate data transfer within the RVLt Conceptual Design Toolchain. RCOTools is capable of reading, modifying, and writing files as well as executing codes and transferring data between them (Ref. 15). RCOTools reduces the need for a user to manually run each code and port data, instead creating a unified run methodology that helps automate the process.

The second Python library, pyaaron, leverages RCOTools to further automate the running of AARON with comprehensive codes. pyaaron, in addition to automating the process, adds several capabilities to AARON such as pre-defined FAA certification conditions. pyaaron specifically focuses on connecting CAMRAD II and CHARM with AARON, and is able to simplify the user effort required significantly.

2.2 Vehicle models

In order to develop best practices for a wide range of possible rotorcraft models, a methodology of increasing complexity was adopted. The first configuration explored was a simple four-bladed main rotor with no twist or taper and idealized airfoil tables. Then, in order to explore more complex geometries that are publicly available, the RVLt reference vehicles were selected. These vehicles model possible UAM configurations, and are an example of what industry might create. For this effort, two reference vehicles were selected, the QSMR and the Side-by-Side. The models were explored in order of increasing model complexity, starting with the simple rotor, then the QSMR, and finally the Side-by-Side. Details of the experimental rotor, the DJI Phantom 3, will be discussed in section 7.1.

2.2.1 The simple rotor model

The first rotor analyzed to develop best practices was a simple, isolated rotor with four blades, each with a constant chord, uniform airfoil distribution, and no twist. To further simplify the

model, the blades were modeled as hingeless and rigid, and used an idealized airfoil. This airfoil is a modified NACA 0012 to have reduced angle of attack stations, no stall, and no moments. This case, called the ‘simple rotor’ in this report, was developed by Dr. Leonard Lopes at NASA Langley, who published simulation results for this rotor (Ref. 16). The acoustics models for the simple rotor contain no tail rotor or fuselage.

Figure 2 shows a mock-up of the rotor in the Open Vehicle Sketch Pad (OpenVSP), and Table 1 lists the major parameters of the rotor used in predictions. The actual models used in CAMRAD II and CHARM are lifting line models and do not have the full blade surface as shown in the figure, but the models do have the aerodynamic properties of the rotor shown in Figure 2.

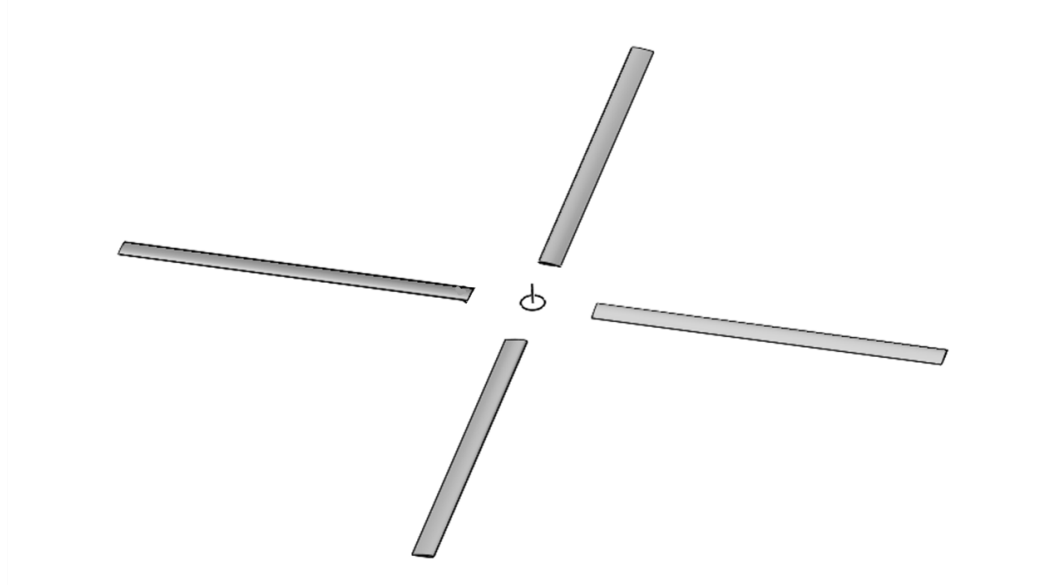


Figure 2.—Simple rotor geometric mock-up in OpenVSP for visualization purposes.

Number of blades	4
Rotor radius (R)	10 m
Chord length	0.5 m
Geometric solidity (σ)	0.06366
Angular velocity (Ω)	23.69 rad/s
Linear twist rate	0 deg/span
Airfoil sections	Idealized

Table 1.—Simple rotor major model parameters.

2.2.2 The QSMR model

Once the simple rotor was complete, a vehicle with more complicated rotor geometry was explored to introduce more realistic conditions while still exploring single rotors. This vehicle is

the Quiet Single Main Rotor, ‘QSMR’, an RVL T reference vehicle (Ref. 17). This vehicle has a single rotor and a no-tail-rotor (NOTAR) configuration, eliminating tail rotor noise.

While still being a single rotor helicopter, the QSMR blades have twist, taper, and droop, as well as varying airfoil sections. The airfoils of the QSMR have an inboard VR-12 airfoil section, while the tip (after 0.94R) has an SSC-A09 airfoil section. The hub of this model is also no longer rigid, allowing for flap, lead-lag, and feathering. Figure 3 shows a graphical representation of the rotor being modeled, and Table 2 lists the major parameters of the QSMR model used in this effort.

For both the CAMRAD II and CHARM models, a tail rotor was included to provide the anti-torque that would be provided by the NOTAR, allowing the vehicle to trim properly. For CAMRAD II, a fuselage was also modeled to account for the vehicle weight. The tail rotor and fuselage, however, were not accounted for in the acoustic analysis, so they are ignored completely by the acoustic analysis. This was done to focus on the rotor specific noise, and to be consistent between CAMRAD II and CHARM.

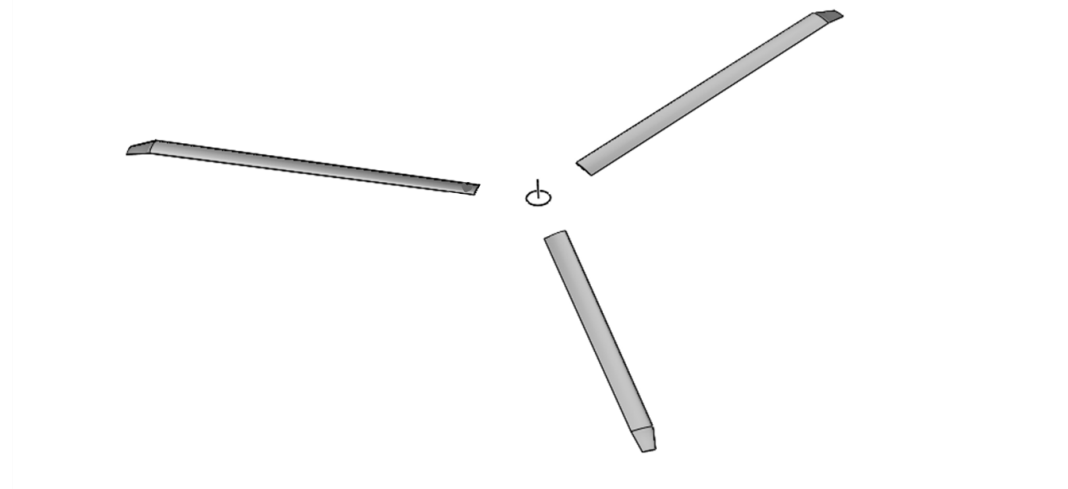


Figure 3.—QSMR mock-up in OpenVSP.

Number of blades	3
Rotor radius (R)	5.04 m
Chord length	0.756 m
Geometric solidity (σ)	0.044
Angular velocity (Ω)	42.37 rad/s
Linear twist rate	-12 deg/span
Airfoil sections	VR-12 & SSC-A09

Table 2.—QSMR rotor major model parameters.

2.2.3 The Side-by-Side model

The third vehicle examined in this study was the Side-by-Side. This vehicle has two side-by-side main rotors that counter-rotate to provide anti-torque. This vehicle was selected to allow the development of best practices for multirotor applications. The Side-by-Side was selected over other multirotor configurations in order to reduce the run time of the comprehensive codes, as more rotors tend to increase computation time. Therefore, the multirotor configuration with the minimum number of rotors that still included rotor interactions was selected.

The Side-by-Side is in many ways similar to the QSMR. Both vehicles are designed to fly the same mission, support the same number of passengers, and have the same airfoils, but the geometry and control schemes differ. The Side-by-Side is controlled via two counter-rotating rotors and a horizontal tail. The rotors are inter-meshed, and therefore do experience wake interaction. Each of the Side-by-Side rotors has a radius of 3.2 m (10.51 ft), and four blades, each with a -16 deg/span linear twist distribution. In order to simplify the model and focus on rotor interactions, no fuselage or wings were modeled, although they do exist in the reference vehicle. From here, the methodology of modeling rotor-rotor interactions can be applied to increasing numbers of rotors, making this vehicle the logical stopping point for the development of best practices. Figure 4 shows a OpenVSP visualization of the rotors/tail modeled in both CAMRAD II and CHARM, and Table 3 lists the major rotor parameters.

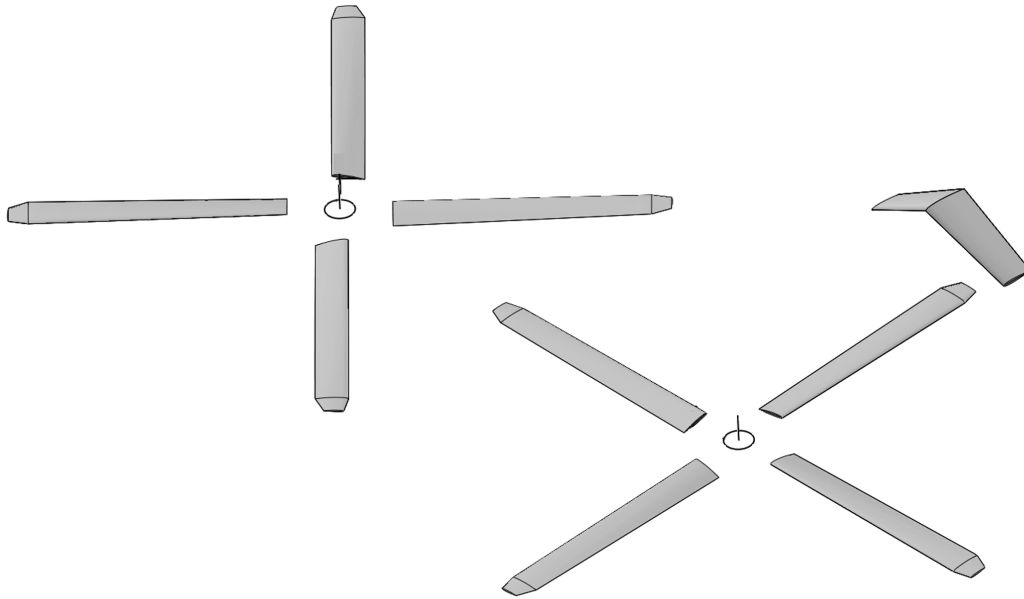


Figure 4.—Side-by-Side visualization in OpenVSP, only showing the parts of the vehicle modeled.

Number of blades	4
Rotor radius (R)	3.203 m
Blade chord length	0.217 m & 0.130 m
Geometric solidity (σ)	0.083
Angular velocity (Ω)	52.36 rad/s
Linear twist rate	-16 deg/span
Airfoil sections	VR-12 & SSC-A09

Table 3.—Side-by-Side major model parameters. Airfoil and blade chord length variables vary over the length of the blade. For specific cord length and airfoil distribution locations, see Appendix A.3.

3 Comprehensive code inputs

For all rotors used in this work, the initial models were completed by modifying a sample case that is similar in configuration to the final desired model. Both comprehensive codes have a series of sample inputs for different configurations. Starting from a sample case reduces the amount of knowledge a user needs in formatting and base model building, which decreases the time it takes to get a model running. This method also ensures that all parameters required for trim are accounted for, even if they will not be used in the acoustic analysis. Making the vehicle model as complete as possible will lead to more accurate acoustic results, even if components such as the fuselage or tail rotor are not currently being considered directly in the acoustic analysis, as it will improve the accuracy of the trim and wake.

3.1 Methodology of increasing complexity

The first best practice that should be applied when using CAMRAD II or CHARM in general, which also applies to acoustic prediction, is to work with increasing steps of model complexity. By minimizing the difficulty of the initial model, mistakes become more easily identifiable. Minimizing complexity also helps with learning the architecture of either code, ensuring the basics are well understood before adding more complex elements. For example, the study conducted for this work began with a simple rotor with rigid, hingeless blades, an idealized airfoil, and a simple wake model. This allows a user to get comfortable with the comprehensive code layout and syntax with a minimum number of variables and an easy-to-verify case. This is where a user may become comfortable with rotation directions, spatial orientation, hub model, and trim settings. Upcoming sections will delve into the specific input parameters of each code, but the ideology of starting simple is an important first step.

Once a user is comfortable with the simple rotor case, then increasing complexity is recommended. A user should always start with a pre-existing case, usually an example case that is provided with the CAMRAD II or CHARM software distributions. These cases are already organized in a logical way with correct syntax. A user can modify the case to match their desired configuration instead of starting from a blank text file. For example, the QSMR could be modeled by modifying the traditional main rotor-tail rotor configuration.

In a similar vein, it is good practice to start with an steady, level flight condition. Often, it is beneficial to start in hover or simple forward flight. Additionally, using a no-trim hover condition to verify model placement, rotation direction, and rotor parameters such as solidity, twist, and rotor radius is beneficial as it reduces computation time. Then, moving on to a trimmed hover

or forward flight condition can identify steady trim parameters. Trim settings will be discussed in detail later in this document.

3.2 Airfoils

Another area where working with increasing complexity is beneficial is in the airfoil tables. Both CAMRAD II and CHARM generate aerodynamic predictions using airfoil tables, which provide sectional lift, drag, and moment coefficients for a number of angle of attack and Mach number pairs. These aerodynamic values are determined either experimentally or computationally via a flow solver such as OVERFLOW (Ref. 18). By using airfoil tables, the comprehensive codes are able to conduct aerodynamic predictions that are both quick and accurate.

Airfoil tables are stored in ‘C81’ format, which can be difficult to read for a new user, and therefore hard to verify. While most users will only ever use pre-made airfoil tables, such as the NACA series of tables, it is still helpful to start with a simplified table when beginning a new model. For this work, that simplified table is a modified NACA 0012 airfoil that has limited Mach number-angle of attack pairs, no stall, and no moments. A simplified airfoil table like this helps eliminate the possibility of errors coming from the airfoils.

Once the case with the idealized airfoil is complete, then the user can move on to a more complicated airfoil table. For this effort, the airfoils used were a blending of two C81 tables, the SSC-A09 and the VR-12. These airfoils were generated by Boeing and Sikorsky during the development and testing of the RAH-66 Comanche helicopter, and were selected as they are well understood and representative of current technologies (Ref. 19). The airfoil tables for the DJI Phantom 3 rotor were created using airfoil geometry obtained from a 3D scan of the blades. Many small UAM vehicles have blade geometries that vary frequently over the length of the blade, which complicate the modeling of them. The methodology used for the DJI Phantom 3 rotor model is further discussed in Section 7.1.

Using accurate airfoil tables will have an impact on the performance of the vehicle, so it is important to ensure that correct airfoil tables are being used with either comprehensive analysis code. Generating new airfoil tables is a study in and of itself, so it will not be covered in this work. Instead, it is simply recommended to make sure the airfoils are coming from a good source and do not have any glaring omissions.

3.3 Blade models

Another important parameter when using comprehensive analysis for acoustic prediction is the blade model geometry. CAMRAD II and CHARM have different methodologies for describing blade geometry, and this can have an effect on both thickness and loading noise if done incorrectly. The first blade parameter to note is the blade twist. CAMRAD II and CHARM have different methodologies for defining blade twist. For CAMRAD II, the blade twist can either be linear or nonlinear. This work modeled linear blades only, and so the blade twist was specified by one parameter, TWISTL, which was applied linearly over the blade with 0 deg twist at the three-quarter chord location. For nonlinear blades, TWISTA can be used to specify a nonlinear twist as a function of radius. CHARM specifies twist using TWRD (blade root twist at zero collective in degrees) and TWSTGD (twist per segment) in the `_bg.inp` file. Therefore, the user can either use the CHARM blade generation ‘makebg’ supplemental code, or write their own code. The user must define the root twist and then the twist per segment in such a way that 0 deg twist is at the three-quarter chord location to match with the convention for linearly-twisted blades. For the

best practices effort, a MATLAB script was written to compute the per-segment values, as it is a geometric math problem. This script is included in Appendix C. If thickness noise is not aligned in the results between CAMRAD II and CHARM, twist is one of the parameters that is likely to be causing issues with the rotor geometry.

The next blade specific characteristic that needs to be carefully selected is the aerodynamic sectioning of the blade. Both CAMRAD II and CHARM use ‘aerodynamic panels’ to separate the blade into spanwise sections. These sections help the codes segment the aerodynamic and near-wake calculations into a discrete number of blade loading calculations, and in the case of CAMRAD II, near-wake trailers. CAMRAD II then uses the near-wake trailers to define the rolled-up wake vortex, which is then trailed from the tip of the blade. Because CAMRAD II defines its aerodynamic and near-wake sections using one variable (NPANEL), adding too many aerodynamic panels can have an adverse and nonphysical effect on loading noise.

CAMRAD II defines its number of panels using the variables NPANEL and REDGE. As the near-wake sections are defined by the number of panels, too many panels can lead to too many near-wake sections in areas of complex flow, which can cause the code to not converge. But, too few panels does not sufficiently define the near-wake and can lead to inaccuracies. The CAMRAD II manual recommends 15 to 25 panels for a blade. To see the effect of these panels on loading noise, a study of 10, 20, 30, and 50 panels were conducted for the QSMR, shown in Figure 5. The 30 and 50 panel case, having too many panels, did not converge and is therefore not included in the figure. The 10 panel case shows a significant increase in loading noise as compared to the 20 panel case, showcasing the inaccuracies in blade loading with too few panels. Figure 5 also provides a graphical representation of the selected spacing of the aerodynamic panels, which were placed with more panels near the blade tip to account for more complex wake interactions.

In CHARM, the requirement for number of spanwise segments is different than CAMRAD II. CHARM defines its wake trailers and aerodynamic panels separately, and so in order to get accurate wake results, only aerodynamic panels need to be considered. To be compatible with pyaaron, CHARM uses the constant vorticity contour wake method and the lifting-line method for aerodynamics. CHARM states that generally 50 aerodynamic panels are considered sufficient, and the code has a programmatic maximum of 100 panels. Additionally, CHARM spaces its aerodynamic segments equally by default (this can be changed to cosine spacing with ICOS), while CAMRAD II defaults the segments to uneven spacing.

To verify the panel recommendation, just like with CAMRAD II, the same CHARM case was run for 25, 50, and 100 panels, shown in Figure 6. Unlike with CAMRAD II, the number of panels had very little effect on noise, only causing some extraneous noise in the 25 panel case. These noise spikes can be attributed to there being too few aerodynamic panels resolving the blade loading. Therefore, when modeling rotorcraft in CHARM, setting the number of aerodynamic panels to be somewhere around 50 with the variable NSPAN is considered best practice.

3.4 Rotor wake

One of the more computationally expensive steps in this part of the Toolchain is modeling the rotor wake. Because of this, both codes have multiple tiers of fidelity settings, which will be explained in this section. Each wake model has increasing complexity and increasing computation time, but produces more accurate results. Both tools have three main categories of wake models: simple wake, rigid wake, and free wake.

It is recommended to start with a ‘simple wake’ model for the rotor wake. A simple wake is the

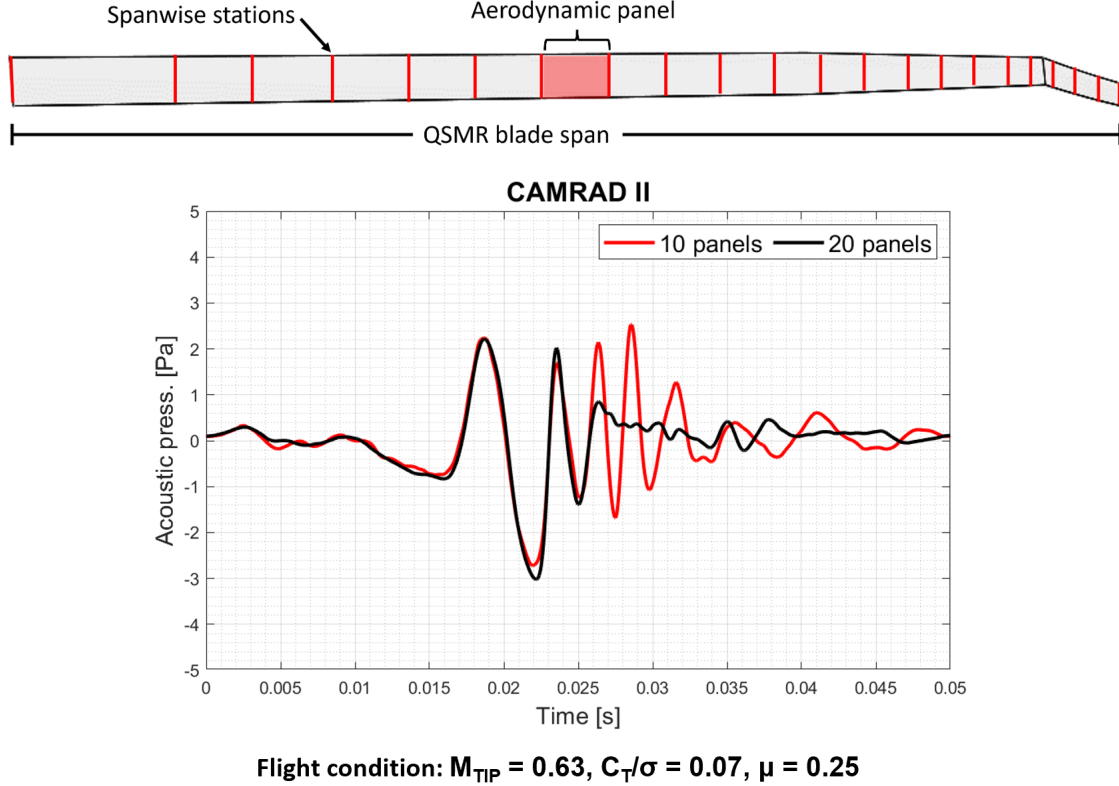


Figure 5.—Blade panel study and graphical representation for the QSMR in CAMRAD II ($M_{tip} = 0.63$, $\frac{C_T}{\sigma} = 0.07$, $\mu = 0.25$).

least computationally expensive rotor wake, but has the least accurate results. The simple wake uses uniform inflow to calculate wake-induced velocities over the rotor disk. This inflow is a quasi-static representation of the wake from momentum theory, and only accounts for linear variations over the rotor disk.

The next step in both computational time and result accuracy is the rigid wake or prescribed wake. Either can be used in this stage, as both model the wake as purely a skewed helix, with the prescribed wake models having empirical values that can be used to tweak the accuracy of results, as these models were initially designed for hovering rotors. The rigid wake is more complex than the simple wake and does account for some wake interactions but does not account for wake contractions or vortex distortions, which are both common phenomena in rotor wakes.

The final wake fidelity before CFD is required is the free wake. The free wake elements experience the induced velocities of all other elements, and therefore experience wake interactions, wake contractions, and vortex distortions. While more computationally expensive, this model provides the best results that comprehensive analysis can produce (Ref. 20). CAMRAD II's free wake uses the 'General Method' which uses wake elements to model a vortex line that has the equivalent velocity of a vortex sheet. This vortex line is then combined with the lifting line method to update

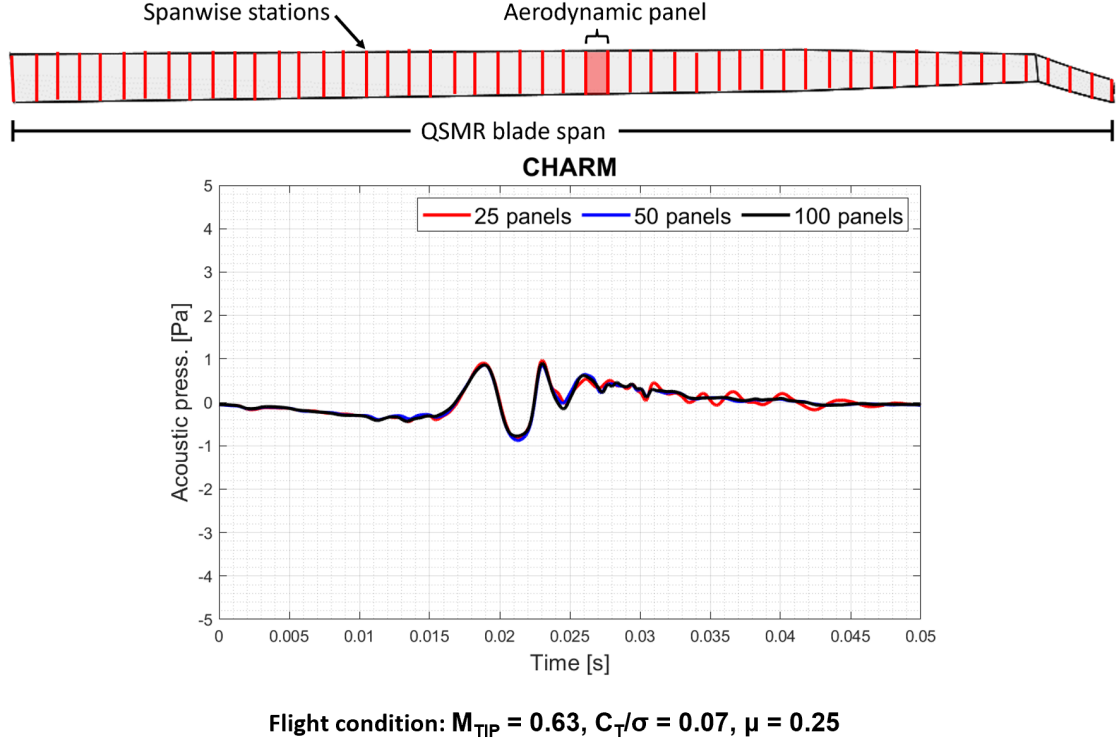


Figure 6.—Blade panel study and graphical representation for the QSMR in CHARM ($M_{tip} = 0.63$, $\frac{C_T}{\sigma} = 0.07$, $\mu = 0.25$).

the forces and moments on the blade to include wake influences (Ref. 2). CHARM’s wake model uses a Constant Vorticity Contour method, which uses a vortex lattice grid to determine the wake influences for blade forces and moments (Ref. 3).

Figure 7 shows the ideal progression from simple wake to free wake. This methodology of increasing complexity and accuracy saves computational time and allows a user to become comfortable with the wake model before introducing more complicated ones. The rigid wake is most frequently used while setting up for a free wake, as it is a good model to use to decrease computational cost while debugging a model. For initial model setup, it is recommended to use a simple wake, but for final results, the free wake is desirable.

3.5 Trim settings

Some of the most frequent differences between CAMRAD II and CHARM are often small variable differences. Some variables have different definitions between the two codes, such as the aerodynamic panels being differently defined. Any such differences encountered during this work are presented here.

The first variable representation difference between CAMRAD II and CHARM is what they use to trim coefficient of thrust. In CAMRAD II, the variable ‘CTTRIM’ trims the variable $\frac{C_T}{\sigma}$, while CHARM trims direct C_T with the variable ‘CT.’ As thrust coefficient is one of the major trim targets used in comprehensive analysis, it is important to be specific when defining the value.

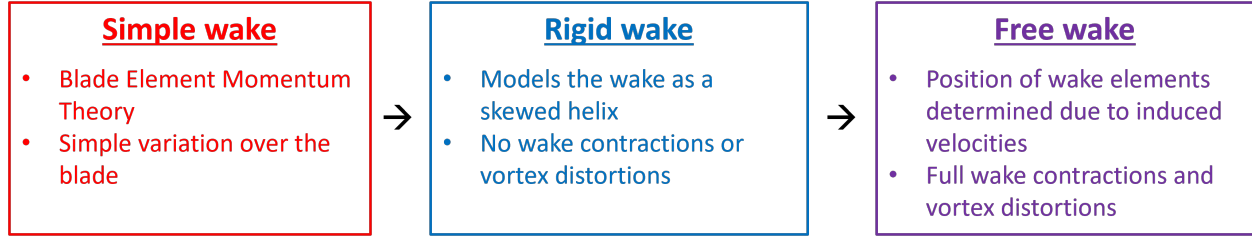


Figure 7.—The progression of wake modeling for CAMRAD II and CHARM cases.

Thrust also has a major impact on loading noise, meaning that trimming to different values can lead to large differences in acoustic results.

Another variable difference between CAMRAD II and CHARM is the definition of flapping angles. The underlying equation for flapping models can either be positive or negative, depending on convention. CHARM uses the legacy ‘NACA convention,’ which calls for a negative Fourier series for flapping angles, whereas CAMRAD II uses the more modern positive Fourier series. This means that when a user is trying to define the flapping angles, β_{1c} and β_{1s} , CAMRAD II and CHARM will have the same magnitude but opposite signs.

3.6 Increased fidelity for acoustics

Once both the CAMRAD II and CHARM models are converging to acceptable levels, it is then time to prepare the models to produce accurate acoustic results. Both codes require specific inputs to produce acoustic outputs. These parameters, however, increase the computation time as they require higher fidelity computations. Therefore, it is considered prudent to first get either code converging before applying the acoustic requirements.

3.6.1 Free wake

The first requirement of a comprehensive code when conducting acoustic predictions is to use a free wake. As acoustic results are influenced heavily by interactional aerodynamics such as blade-vortex interactions (BVI), it is important that the wake contains all interactions. Often, the wake interacting with a blade can produce highly tonal acoustic impulses, which can cause increased noise. Therefore, setting the wake to a free wake is imperative to getting accurate acoustic results.

In CAMRAD II, the free wake is obtained by setting LEVEL to 3 in the trim class (LEVEL=1 for simple wake and LEVEL=2 for rigid wake). For CHARM, the process is similar, and includes setting NOWAKE to 0 (for simple wake, set NOWAKE to 1 and for rigid wake, leave NOWAKE=0 and add QUICKLOOK=1 to the beginning of the run characteristics input file). The only downside to using a free wake is increased computation time and higher chance for user error. That is why it is necessary to begin models using a simple wake, but to then move to a free wake when producing acoustic results. All acoustic results from here onward are conducted using the free wake.

3.6.2 Azimuthal resolution

One of the most important steps in preparing CAMRAD II and CHARM to conduct acoustic predictions is to increase the azimuthal resolution. Azimuthal resolution is the number of discrete locations around the rotor disk for which the comprehensive code calculates or updates the aero-

dynamic calculation. Comprehensive codes requires only a fairly coarse azimuthal grid resolution to trim a rotor or vehicle because trim involves average forces and moments. Using an azimuth resolution that is too high during trim can cause coupled airloads and structures to either take considerable time to calculate or fail entirely, depending on the algorithm. But, acoustic predictions require high resolution airloads in order to capture high frequency noise content, which is of particular relevance to UAM applications. CAMRAD II and Charm implemented methods of calculation in which the initial calculation is performed with low azimuthal resolution, and a subsequent, single, high resolution step follows.

For CAMRAD II, the high resolution step is calculated by something called ‘post-trim.’ Post-trim takes the completed aerodynamic calculation and the low azimuthal resolution and calculates one additional revolution. This additional revolution is calculated with the higher azimuthal resolution, which is then provided to the acoustic prediction code. Often, the difference between these two resolutions is a factor of ten. Taking the QSMR as an example, the initial resolution was 36 azimuthal steps, while the post-trim resolution was 360 azimuthal steps. Figure 8 shows a graphical representation of the difference between these two calculations for azimuthal spacing. This increases the azimuthal resolution from 10° to 1° , which is considered more applicable to acoustic methodologies.

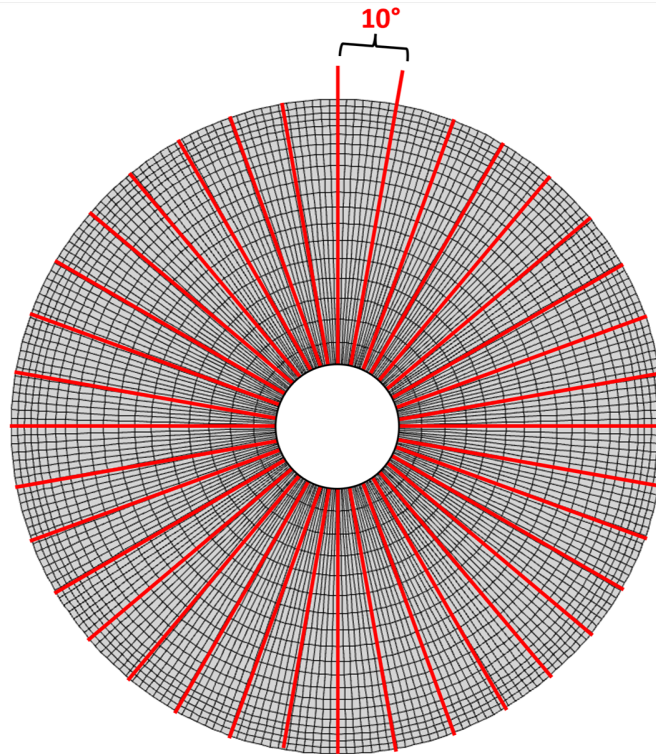


Figure 8.—Graphical representation of the spanwise and azimuthal spacing for aerodynamic inflow calculations in CAMRAD II. Black represents the final computation done by post-trim, while red represents the initial azimuthal calculation.

For CHARM, the high resolution step is conducted by a method called ‘reconstruction.’ Unlike CAMRAD II, which does an additional revolution, CHARM takes the last revolution of the low resolution airloads and back-calculates the revolution to have higher azimuthal resolution. Just like with CAMRAD II, the high-resolution calculation in CHARM should have a resolution 10 times greater than that of the low-resolution revolutions. Typical resolution values are 24 and 240 steps, for the low- and high-resolution revolutions, respectively. CHARM is a little more sensitive to the high-resolution airloads than CAMRAD II, so it is recommended to try a few resolutions, as sometimes CHARM cannot calculate as high as 240 steps, and a lower number is required. It is recommended to do a resolution study in CHARM by testing several resolutions and looking for anomalies. Two example anomalies are: more peaks and valleys than expected (it should be on the order of the number of blades for major peaks), or high loading sections that do not correspond to a frequent wake interaction area such as the advancing side of the rotor. Engineering judgement and comparison with similar cases may be necessary to identify the correct resolution.

3.6.3 Trim convergence and revolutions

Both CAMRAD II and CHARM carry out their calculations using loops. That is, they do calculations in terms of revolutions, with each revolution getting closer to the desired trim configuration. There are two ways in which a user is able to specify how many revolutions the code will calculate: setting a convergence goal or specifying the number of revolutions. The first option, setting a convergence goal, is the primary method that is used to specify how many revolutions a calculation will do. In either CAMRAD II or CHARM, this is completed by setting the trim goal using MTRIM, MNAME, and VNAME or ICOLL and ITRIM, respectively.

For example, a common trim goal is rotor coefficient of thrust and a common set variable is blade collective. In order to find the desired rotor coefficient of thrust, the comprehensive code changes blade collective until the resulting rotor thrust is close enough to the desired rotor thrust and does not vary more than the specified error. For CHARM, the convergence criteria is specified by CONVG1, CONVG2, and CONVG3, representing degrees of pitch, percent of thrust, and percent of torque, respectively. For CAMRAD II, the trim convergence tolerances can be relaxed with either the RELAX_ (relaxation factor, reduces the difference between iterations) or TOLER_ (tolerance below which the iteration is considered converged variables. The underscores (_) for TOLER and RELAX variables can be various letters, depending on the loop being considered. For example, TOLERT and RELAXT parameters control the Trim loop; TOLERC and RELAXC control the circulation loop; etc.

Although recommended for most cases, there are some cases where trimming the rotor is not ideal. For example, if modeling a rotor in a wind tunnel, that rotor may not necessarily have the ability to trim. Instead, the parameters such as blade collective, RPM, and windspeed are set, and the resulting forces and moments are recorded. The code still needs to produce a complete wake and a steady-state solution. Therefore, a user would use the number of revolutions to specify how many calculations the comprehensive code will compute. For CHARM, NREV is the variable that sets the number of revolutions. A sufficiently high number of iterations are required to allow for convergence. For this work, an NREV of 50 was used and then increased if convergence was not yet achieved. For CAMRAD II, this variable is set using the ITER_ command, where the underscore represents which parameter category is being iterated, for example, ITERC covers the number of iterations for the circulation calculation. As a default, when no trim is specified, CAMRAD II

conducts one near wake and one far wake iteration, and 40 motion iterations, to name a few. Often, the default is acceptable, although sometimes more iterations are needed.

3.7 Multirotor considerations

When going from a single rotor case to a multirotor case, several key considerations must be made in order to correctly model these configurations. The first of these considerations is rotor interactions. Rotor interaction is a major source of noise that is increasing in importance as multirotor configurations become more popular. As rotors become closer together in space, their wakes begin to interact. For comprehensive models, this means increased calculations as wakes interact with each other. For CHARM, the intricacies of the wake interactions are applied to multirotor configurations automatically. For CAMRAD II, the wake interactions need to be defined within the namelist using multiple variables.

Within the WAKE type of the ROTOR class, the forward flight wake elementary scenario (OPSCEN =1) is a good starting point for the wake model. Turning off the elementary scenario subsequently (OPSCEN =0) then allows changes to be made to fine-tune the wake model. Increasing the tip vortex core radius (COREWG) and the vortex core (CORE), tend to help with wake convergence. The vortex core model, free wake model, and wake extents can also be tuned to aid in convergence and accuracy. The Side-by-Side model used in this effort shows an example of what settings can be used for rotor-rotor interactions and accurate wake convergence (see Appendix A.3 for the Side-by-Side wake model).

4 Acoustic code inputs

Within the RVL Toolchain, acoustic prediction is conducted using AARON, a rotor noise tool that simplifies inputs for ANOPP2 to streamline rotor source noise predictions. Additionally, AARON is accessed using pyaaron, a set of Python libraries to automate this process. Most of the user effort for performing noise prediction, especially for tonal noise, is done in the comprehensive analysis step. The only user inputs required at the acoustic prediction phase are items such as paths to file locations, option selection for tonal noise, and broadband self-noise input parameters. This section covers the necessary inputs for AARON as well as how to identify the inputs for the BPM model used in broadband self-noise prediction.

4.1 Thickness and loading noise

As mentioned previously, the number of inputs required to generate thickness and loading noise (tonal noise) predictions are fairly small. Blade geometry, position, and loading data are all provided by the comprehensive code. Therefore, the only user supplied inputs are filtering options, atmospheric properties, windowing options, and run methodologies. Tonal noise is governed by the ANOPP2 Farrassat Formulations Internal Functional Module (AFFIFM), namely the ANOPP2 Formulation 1A Internal Functional Module (AF1AIFM). This model, as the name implies, uses Farassat's Formulation 1A of the Ffowcs Williams-Hawkings equation. The model can also use the compact form of Formulation 1A, where observer time derivatives are turned into source time derivatives, and then a compact assumption is applied to the equation (Ref. 4).

4.2 Broadband self-noise

Broadband self-noise prediction within AARON is implemented inside the ANOPP2 Self Noise Internal Functional Module (ASNIFM) and uses a modified Brooks-Pope-Marcolini model when using comprehensive analysis (Ref. 14). This model is a low fidelity semi-empirical model that uses a combination of user-supplied inputs and comprehensive code outputs to produce an estimation of broadband self-noise. ASNIFM uses the effective angle of attack of the blade and the induced velocity of the rotor disk from the comprehensive code, then modifies it using the parameters supplied by the user. This estimation is a good first prediction of what broadband self-noise may be generated by the rotor, although for accurate results, higher fidelity prediction methods may be recommended.

Unlike the tonal noise predictions, the ASNIFM model requires more user inputs. These parameters are directly supplied by the user in the `pyaaron` namelist file as a function of non-dimensional rotor radius. It is recommended to separate the blade into sections based on where there are changes in the geometry. For example, the QSMR was partitioned into 4 sections to account for where the airfoil geometry changed from the SSC-A09 airfoil to the VR-12 airfoil. Then, a user can define the variables for each section shown in Table 4. These airfoil station sections and their corresponding values are read into ASNIFM and then interpolated to the comprehensive code aerodynamic panel center locations. Figure 9 shows graphically where each of the ASNIFM input variables can be found on the airfoil.

Name	Symbol	pyaaron variable	Definition
Broadband radial stations	-	<i>rbb.rN</i>	Input radial stations for broadband calculations.
Thickness-to-chord radial stations	-	<i>rtoc.rN</i>	Input radial stations for thickness over chord calculations.*
Trailing edge wedge angle	θ_{TE}	<i>teang.rN</i>	Angle between the upper and lower airfoil surfaces as if trailing edge contains no blunting.
Trailing edge wedge thickness	h	<i>teth.rN</i>	Height at the end of the trailing edge wedge, where the blunting is applied.
Maximum thickness-to-chord ratio	t/c	<i>toc.rN</i>	Maximum thickness of the airfoil divided by the chord length.*
Boundary layer trip setting	BL_T	<i>trip.rN</i>	Switch to consider a turbulent trailing edge and use a tripped boundary layer equation.
Zero-lift angle of attack	α_0	<i>ang0.rN</i>	Angle of attack value for which the airfoil is generating 0 lift at the local Mach number.

Table 4.—Broadband self-noise parameter namelist variables. ASNIFM requires inputs for all rotors, thus the N notation in the `pyaaron` variable. The N should be replaced with the rotor number that is being considered. *Input is for thickness calculations, but located in the ASNIFM inputs section, and is therefore included for completeness.

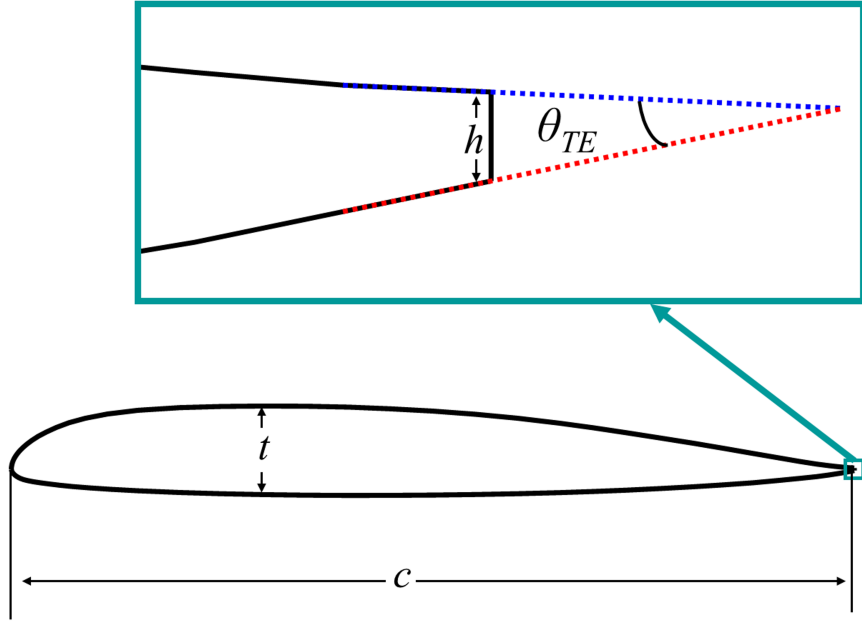


Figure 9.—Airfoil geometry inputs for BPM model shown on an example airfoil section.

Two of the parameters described in Table 4 are difficult to ascertain without additional codes: trailing edge wedge angle and zero-lift angle of attack. Trailing edge wedge angle requires a line to be drawn from the upper and lower surfaces of the airfoil. This can be done by fitting a line to the curve of the upper and lower surfaces, and then extending this line until the top and bottom surfaces meet. It is recommended to either export the airfoil geometry for manual plotting, or to write a code that computes it.

Additionally, the zero-lift angle of attack requires some external coding to identify. The recommended process is to find the Mach number at each broadband radial station, then go to that Mach number in the airfoil table associated with that section. Then, the code should find the location where $C_L = 0$ and interpolate the associated angle of attack. This value is the zero-lift angle of attack, which can then be input into pyaaron. It is important to note that because the Mach number will vary at each radial station along the blade span, the zero-lift angle of attack will be different at each radial station. Because of this, it is important to have an adequate distribution of radial station (r/R) values, even if many of the other broadband variables will only change a few times, because the zero-lift angle of attack will change relatively frequently. This work used r/R steps of 0.05, though a finer resolution could be used for even better approximations, although not all may be used by the code when interpolating to the comprehensive code aerodynamic panel centers. Example MATLAB scripts for identifying all pyaaron broadband self-noise inputs for the QSMR are included in Appendix C.

To better understand the sensitivity of each of the user inputs on the broadband noise, a parameter variation study was conducted. The vehicle selected for this study was the QSMR in forward flight with the free wake model. First, a ‘baseline’ case was established, which contained

the correct inputs for the ASNIFM model of the QSMR. Table 5 shows the correct inputs for the radial stations selected for the QSMR, based on the flight condition and blade geometry. This became the baseline case for the parameter variation study.

r/R	Airfoil	t/c	α_0	h	θ_{TE}
0.00	VR-12	0.106	-1.470	0.0021	14.31
0.05	VR-12	0.106	-1.470	0.0021	14.31
0.10	VR-12	0.106	-1.470	0.0021	14.31
0.15	VR-12	0.106	-1.470	0.0021	14.31
0.20	VR-12	0.106	-1.470	0.0021	14.31
0.25	VR-12	0.106	-1.470	0.0021	14.31
0.30	VR-12	0.106	-1.470	0.0021	14.31
0.35	VR-12	0.106	-1.470	0.0021	14.31
0.40	VR-12	0.106	-1.470	0.0021	14.31
0.45	VR-12	0.106	-1.470	0.0021	14.31
0.50	VR-12	0.106	-1.470	0.0021	14.31
0.55	VR-12	0.106	-1.477	0.0021	14.31
0.60	VR-12	0.106	-1.482	0.0021	14.31
0.65	VR-12	0.106	-1.488	0.0021	14.31
0.70	VR-12	0.106	-1.492	0.0021	14.31
0.75	VR-12	0.106	-1.492	0.0021	14.31
0.80	VR-12	0.106	-1.487	0.0021	14.31
0.85	VR-12	0.106	-1.505	0.0021	14.31
0.94	SSC-A09	0.09	-1.048	0.0023	8.16
1.00	SSC-A09	0.09	-1.013	0.0023	8.16

Table 5.—QSMR BPM inputs for ASNIFM.

The method of this study was to isolate each ASNIFM input parameter to see its effect on total broadband noise. This method identified how sensitive the BPM model is to each user input, to better understand to what extent an error or estimation in these inputs may change the results. Six of these isolation studies were conducted, shown in Table 6.

Case	Change from baseline
1	Baseline
2	$t/c = 0$
3	$\alpha_0 = 0$
4	$h = 0$
5	$\theta_{TE} = 0$
6	$BL_T = 1$
7	$t/c, \alpha_0, h, \theta_{TE} = 0; BL_T = 1$

Table 6.—QSMR ASNIFM input parameter study cases.

In order to see the effects of changes to the ASNIFM inputs, total broadband noise results were plotted in an acoustic spectra plot from 0 to 20,000 Hz. Figure 10 shows results for both CAMRAD

II and CHARM via outlined and solid shapes, respectively. The baseline case is plotted as a dashed and solid line to show how each parameter variation differs from the baseline. Generally, the results show that changes in inputs do lead to minor changes in broadband noise. The differences between CAMRAD II and CHARM for most cases was also minimal, only around 1 to 2 dB at the most. For most frequencies and most cases, CAMRAD II and CHARM were nearly identical in their results.

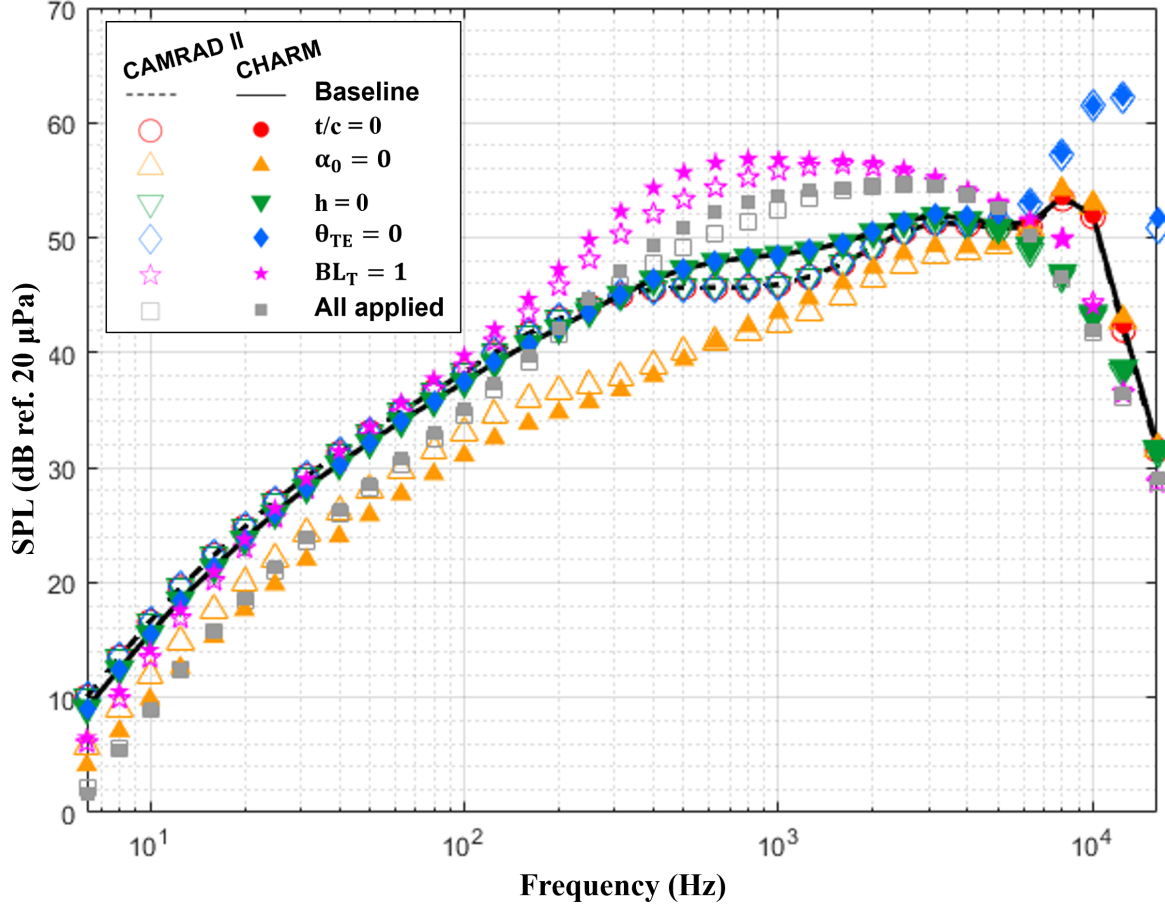


Figure 10.—BPM parameter study for QSMR in forward flight free wake for cases 1 through 7 for total broadband self-noise ($V_{inf} = 53.9$ m/s, $\frac{C_T}{\sigma} = 0.07$, $\mu = 0.25$, $\alpha_s = -3^\circ$, $\theta_0 = 4^\circ$).

As for the variation study, most parameters caused at least a minor change in sound pressure level. Only one case, the maximum thickness over chord case (case 2), did not change the results at all. Case 3, the zero-lift angle of attack study, caused the most low-frequency change in noise, reducing the SPL by roughly 6 dB from 0 to 6,000 Hz. Case 4, the trailing edge height case, followed the baseline for most of the frequencies, only reducing noise above 6,000 Hz. Case 5, the trailing edge angle case, caused a sharp, roughly 10 dB increase in noise over 6,000 Hz, but otherwise did not change the baseline. Case 6, the boundary layer tripped case, caused an increase in noise in the 100 Hz to 6,000 Hz case, and then a reduction in SPL over 6,000 Hz. Finally, case 7 showed

all the previous settings applied at once, and did show a constant difference from the baseline case, leading to a general reduction in low and high frequency noise, with an increase in mid-frequency noise.

This study shows that major changes to the ASNIFM model will have a reflection in the broadband self-noise results, but that the semi-empirical model is relatively resistant to small errors. The majority of the noise prediction leverages the output of the comprehensive code, and user inputs only slightly adjust the values. This means that focusing on a correct comprehensive run is the best way to get an accurate self-noise prediction, while slight errors in user inputs to AARON have less of an impact, but cannot be ignored completely, as they do contribute to the self-noise profile of the rotor.

5 Outputs and visuals

A major facet of running the RVLTC Conceptual Design Toolchain is interpreting outputs and plots. Both the comprehensive codes and AARON output a wide variety of information for a user to peruse. While all of the information is useful, some plots and outputs are important to check for their influence on acoustics and to ensure correct acoustic results. This includes both the output variables from the comprehensive codes and the plots produced by AARON.

5.1 Commonly used output variables

Comprehensive code outputs are usually hundreds to thousands of lines long, and it is difficult to glean important information from them quickly. Therefore, it is highly recommended to make use of parsing scripts to extract key data. For this report, Python with the RCOTools library was used to extract data from the output files, put them into a spreadsheet, and then look for differences between the CAMRAD II and CHARM codes. These scripts are included in Appendix B. The parameters that were consistently extracted for this work are as follows: advance ratio, atmospheric density, β_{1C} , β_{1S} , blade collective, coefficient of thrust, coefficient of thrust divided by geometric solidity, flight speed, geometric solidity, lateral cyclic, linear twist, longitudinal cyclic, radius, rotor pitch angle, RPM, speed of sound, tip Mach number, and tip speed.

5.2 Metadata

Within AARON, the AF1AIFM ANOPP2 module has the option to produce ‘metadata.’ This metadata is a series of Tecplot .plt files that contain plots of the internal parameters from the ANOPP2 run. There is one file per blade, and the files are in the format of Af1aifm_Metadata_#.plt, where # is the blade number. These files can then be opened in Tecplot together. This allows a user to look at the core variables of an ANOPP2 run and understand their influence on outputs. There are two ways to look at metadata: as a time-based iso-surface or as a 2D disk plot.

5.2.1 Iso-surfaces

One of the visualization methods in the ANOPP2 metadata is the ‘iso-surface.’ Iso-surfaces allow for the animation of the metadata in time. The user can also apply vectors to the iso-surfaces to observe force and other similar magnitudes in the time domain. By using the animation feature, a user can see rotation direction, blade placement, and blade loading, among other variables. The iso-surface was used extensively when developing these models for acoustic prediction, and is an easy way to check for major errors in the model.

To read in metadata, first open Tecplot and load in all the metadata files found in the metadata folder that is generated when ‘form_metadata’ is set to ‘all’ in the pyaaron namelist. Once imported, the only other initialization step is to make sure to go to ‘plot’ then ‘axis’ and ensure the dependency is set to ‘xyz dependent’ and the ‘X to Z ratio’ is 1 in order to ensure proper scaling. Then, simply check the ‘iso-surfaces’ box, show the mesh and vectors, and animate the results. Iso-surfaces and disk plots are the best ways to visualize the comprehensive results and ensure the rotors are operating as expected.

5.2.2 Disk plots

Metadata disk plots allow a user to visualize parameters for one full revolution of the blade. These are generated from the same files as the iso-surfaces, except that the ‘contour’ box in the ‘zone layers’ category is used instead of the ‘iso-surface’ category in ‘derived objects.’ In total, AARON produces 48 variables in the metadata that can be output, plotted, or examined to aid in understanding the noise prediction results. While this does take an understanding of rotorcraft or noise parameters, looking at forces, velocities, induced velocities, time derivatives, and displacements can help identify issues in the model.

As the calculations in this work are all steady and periodic, a single revolution is representative of all revolutions. For this effort, the z loading and z induced velocities were the major parameters that were analyzed for debugging. The z loading has a direct influence on noise, as the blade loading is the primary driver of loading noise. Figure 11 shows the z loading disk plots for both CAMRAD II and CHARM for the QSMR in forward flight. By comparing these two plots, differences in the rotor inflow models can be inferred.

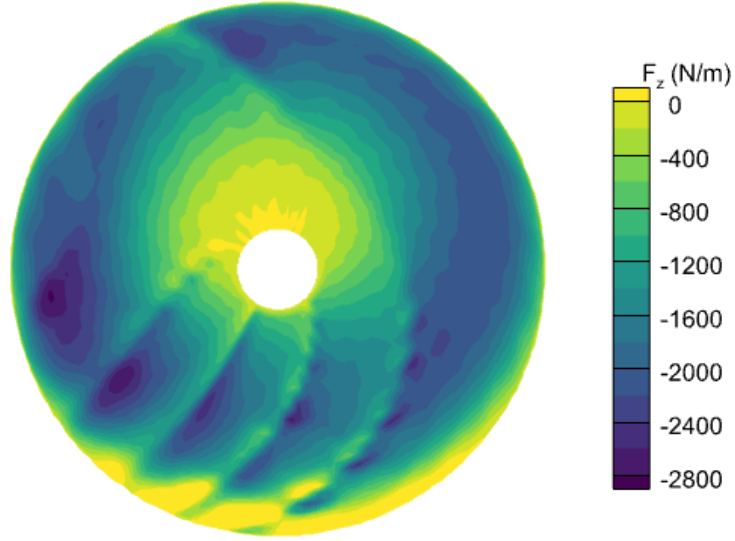
By studying Figure 11, the locations of differences in the blade loading can be identified. Firstly, the blade tip loading is different between CAMRAD II and CHARM, where CAMRAD II has low to 0 blade loading along the outer 1/4 of the blade, and CHARM has rather high loading around the outer 1/4 of the blade. This implies differences in either blade tip modeling, tip loss factor, or tip vortex modeling. Additionally, it can be identified that CAMRAD II has much stronger blade interactions, shown by the sweeping vertical lines along the rotor disk. Both models captured well these interactions and their locations but prescribe different strengths to them, as the CAMRAD II results display much darker lines than those of CHARM. These lines are likely caused by blade-vortex interactions, for which CAMRAD II is predicting a stronger effect on rotor inflow. The differences can be attributed to differences in model implementation, since both CAMRAD II and CHARM are using a lifting line model. Differences in model implementation tends to lead to results that are similar but with some variability. By using metadata in the manner described and combining it with a knowledge of rotorcraft blade loading, more understanding about the acoustic results can be obtained.

Additionally, as stated previously, the ASNIFM module uses the effective angle of attack (α_{eff}) and induced velocity (V_{ind}) for broadband self-noise predictions. These values are accessible in the metadata and can be used to understand self-noise sources from the comprehensive analysis.

5.3 Acoustic plot types

There are many ways to interpret and display acoustic prediction results, so only a sample of options will be presented here. The three types of plots most often seen in the rotorcraft acoustics field are plots of acoustic pressure time history, frequency spectra, and integrated sound metrics (such as OASPL).

CAMRAD II



CHARM

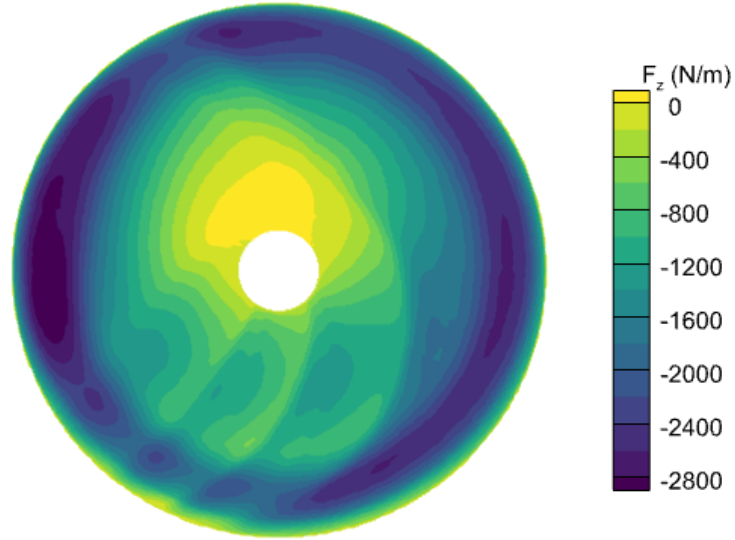


Figure 11.—QSMR forward flight free wake, single blade z loading metadata for CAMRAD II and CHARM ($V_{inf} = 53.9$ m/s, $\frac{C_T}{\sigma} = 0.07$, $\mu = 0.25$, $\alpha_s = -3^\circ$, $\theta_0 = 4^\circ$).

Acoustic pressure time history plots are a method of visualizing acoustic results in the time domain. For rotorcraft applications examined here, the results are periodic and a function of blade

passage. All individual thickness and loading plots in the results section of this paper are examples of acoustic pressure time history plots. To get a better idea of relative magnitude, the acoustic time history can be integrated into a metric like OASPL. This provides one value per observer. While OASPL provides a single, integrated value for each observer, care must be taken when using integrated values. This is because different acoustic pressure time histories can generate the same integrated value.

The other plotting method used often is a frequency spectrum, which takes the acoustic pressure time history and converts it into the frequency domain using a Fourier series transformation. These results allow the acoustics to be broken down by frequency, which can aid in noise source identification. For example, low frequency noise is commonly attributed to tonal noise sources, while high frequency noise is usually due to broadband self-noise sources. The broadband self-noise and total noise plots from the results section are examples of frequency spectra plots.

6 Application of best practices

To provide practical example cases of the best practices, the acoustics of three vehicles were predicted using the practices developed in this work: the simple rotor, the QSMR, and the Side-by-Side. The thickness and loading acoustic pressure time history plots as well as the broadband self-noise and total noise spectra are presented. These results are for the free wake and forward flight condition at nine different microphones. Intermediary wake as well as additional flight condition results are omitted for brevity.

6.1 Microphone placement

For each vehicle, tonal noise results were produced for nine observer locations. These locations are represented graphically with respects to Cartesian (x , y , and z) and Spherical (azimuth, elevation, and radius) coordinates for the Side-by-Side in Figure 12. Additionally, in-plane microphones are shown in red while out-of-plane microphones are shown in blue. In the figures following, microphones will be identified with ψ denoting azimuth angle and ϕ denoting elevation angle. These microphones remain a fixed distance from the noise source, and move with the vehicle. Microphone locations were selected to capture a sufficient range of thickness noise (in-plane) and loading noise (out-of-plane and in front of the rotor). As this is a forward flight condition, the focus is on the front and underside of the vehicle. Of course, more microphone locations provide more information about the rotor, but it is prudent to downselect to fewer microphones in key locations in order to reduce run time and improve result understanding and clarity, as too many microphones can sometimes obfuscate key results.

For this paper, the microphones are at the same angles for each vehicle but are scaled to be ten rotor radii away from the CG of each vehicle. This was done to help scale the results by the size of the vehicle modeled to give some commonality between results, although comparison between the vehicles was not the goal of this work. There are five in-plane mics that span a semicircle in front of the vehicle and four out of plane mics that make a quarter circle below and in front of the rotors. Table 7 provides the exact locations of the mics for each of the three models in X , Y , and Z coordinates as measured from the CG of each model as well as the corresponding azimuth and elevation angles.

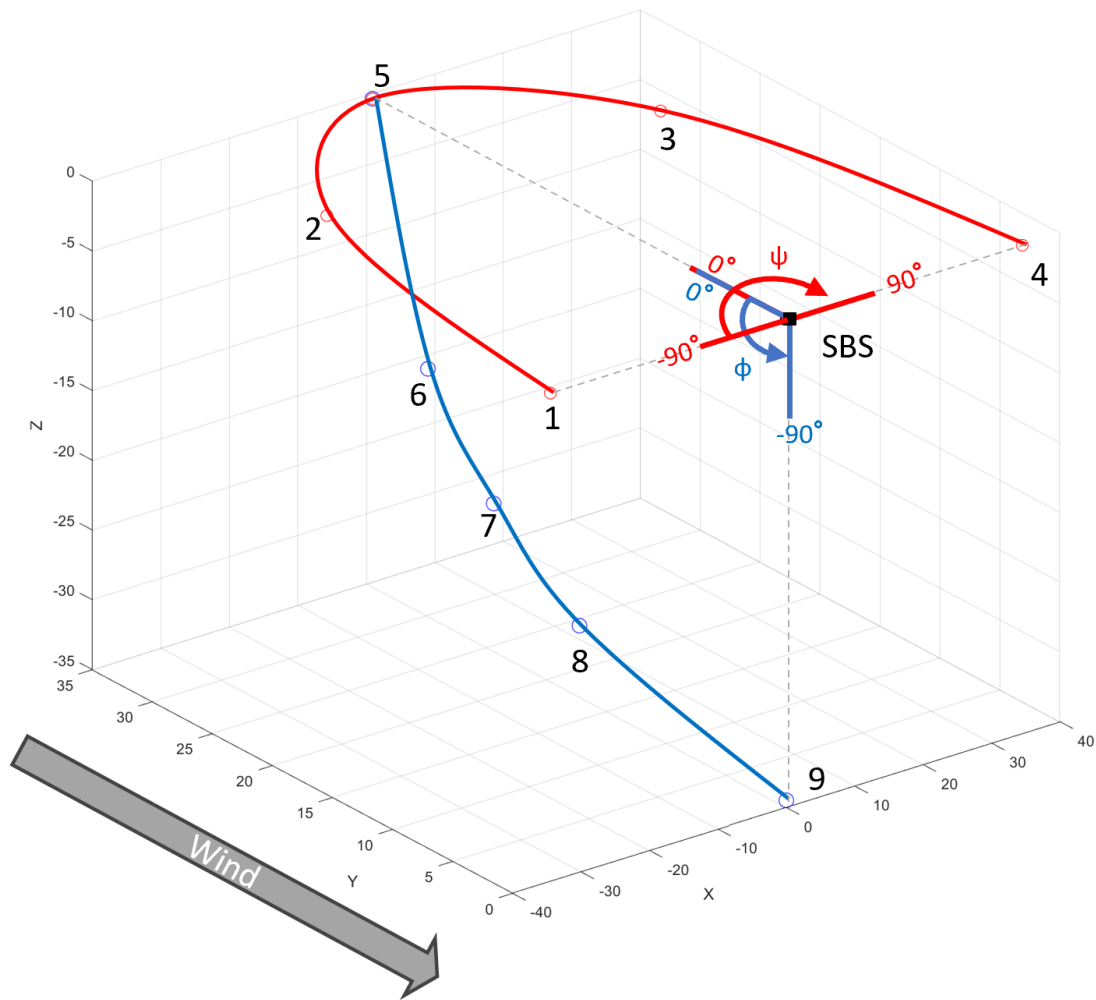


Figure 12.—Microphone locations for the Side-by-Side graphically represented. The wind is moving in the negative y direction as shown.

6.2 Noise results for each vehicle

To support the release of the best practices, three vehicle models were explored, and the final results are presented here. The vehicles were selected with increasing complexity, starting with a simple rotor, then the QSMR, then the Side-by-Side. This ensures the capture of both traditional main rotor-tail rotor noise as well as noise from a more novel ‘eVTOL-type’ rotor. Final results were considered achieved when the rotor had a fully trimmed free wake. The acoustic results of CAMRAD II and CHARM will not align exactly for loading noise, as the models used for both rotor wake and blade loading differ between the two codes. Thickness noise is expected to align well, as that is supplied by rotor geometry, which should not change between the two codes.

This section shows the free wake, forward flight results for each rotor to both demonstrate the best practices as well as provide results to users for both examples and verification. The thickness

Mic	Simple rotor					QSMR			Side-by-Side		
	$\psi(^{\circ})$	$\phi(^{\circ})$	X(m)	Y(m)	Z(m)	X(m)	Y(m)	Z(m)	X(m)	Y(m)	Z(m)
1	-90	0	-32.8	0	0	-54.2	0	0	-34.5	0	0
2	-45	0	-23.2	23.2	0	-38.3	38.3	0	-24.4	24.4	0
3	45	0	23.2	23.2	0	38.3	38.3	0	24.4	24.4	0
4	90	0	32.8	0	0	54.2	0	0	34.5	0	0
5	0	0	0	32.8	0	0	54.2	0	0	34.5	0
6	0	-30	0	28.4	-16.4	0	46.9	-27.1	0	29.9	-17.2
7	0	-45	0	23.2	-23.2	0	38.3	-38.3	0	24.4	-24.4
8	0	-60	0	16.4	-28.4	0	27.1	-46.9	0	17.2	-29.9
9	0	-90	0	0	-32.8	0	0	-54.2	0	0	-34.5

Table 7.—Microphone locations for the simple rotor, QSMR, and Side-by-Side in terms of both angle and Cartesian coordinates with the origin being the center of gravity of the vehicle.

and loading noise will be presented in terms of acoustic pressure time history. The broadband self-noise and total noise will be presented in terms of acoustic spectra, as AARON does not generate broadband self-noise acoustic pressure time history results (because the self-noise model is a frequency based model, not a time based model). Broadband self-noise results are shown in terms of noise source components for both CAMRAD II and CHARM. In ASNIFM, these sources are called: bluntness, pressure side, suction side, separation, laminar, tip, and total. Bluntness refers to trailing edge bluntness noise caused by the vortex shedding from a blunt trailing edge. Pressure side and suction side refer to the noise produced by the turbulent boundary layer on the pressure side (bottom) and suction side (top) of the airfoil. Separation refers to the noise produced by flow separating from the airfoil and producing turbulent eddies. Laminar flow was not encountered in any of the results of this work, but would be produced when there is laminar flow over the majority of the airfoil that results in vortex shedding at the trailing edge. This condition is not always encountered. Tip refers to the tip-vortex formation noise, which occurs when vortices are formed at the blade tip (Ref. 14). Finally, total refers to the sum of the results from all of these sources. Total provides a comprehensive image of the broadband noise predicted in ASNIFM.

All results are unfiltered and unedited from the AARON output, with the exception of the total noise spectra plots. Total noise spectra plots were produced by logarithmically adding the tonal spectra (combination of thickness and loading noise) with the broadband-self spectra (combination of broadband self-noise results).

6.2.1 Simple rotor

The first rotor considered, the simple rotor, has the least complex acoustic results due to the simplified nature of both the rotor geometry and the airfoil tables. Figure 13 shows the acoustic pressure time history for thickness noise and Figure 14 shows the acoustic pressure time history for loading noise. Results are shown for the first full revolution of the rotor for the nine microphone locations depicted in Figure 12, where ψ denotes azimuth angle and ϕ denotes elevation angle. Acoustic results produced using CHARM are shown as a blue solid line, and acoustic results by CAMRAD II are shown as a black dashed line.

The interesting nature of the acoustic results for this rotor is due to the intentions behind the rotor design. This rotor was initially developed to study thickness noise, and because of that, it

has a thick, symmetric airfoil that produces high magnitude thickness noise. The blade loading, on the other hand, is very low, due to the reduced thrust and lower flight speed of this model. This leads to the loading noise of the simple rotor being quite low. As expected, the thickness noise for the simple rotor between CAMRAD II and CHARM has minimal variation, while the loading noise is the same between the two codes for the mean results, but contains more variations.

The broadband self-noise results of the simple rotor are shown in Figure 15. They are presented as a frequency spectra between 0 and 16,000 Hz for all nine mic locations. AARON separates the broadband noise into five separate noise sources and one summary noise source (total noise). For each noise source, the CAMRAD II results are dotted lines, labeled with ‘CII’ in the legend, and the CHARM results are solid, labeled with ‘CH’ in the legend. For the simple rotor, the dominant self-noise sources are the pressure side (orange) and suction side (blue) of the blade. The pressure side is the turbulent boundary layer on the bottom of the rotor, while the suction side is the turbulent boundary layer on the top of the blade. These results imply that the simple rotor has a turbulent boundary layer around the whole blade, causing a large amount of broadband self-noise. It is interesting to note that CAMRAD II predicted a small, but nonzero, tip noise source, whereas CHARM did not, shown by the pink lines on the plot. There is no solid line, as CHARM output a negative SPL number, which is well below the threshold of human hearing ($20 \mu\text{Pa}$).

The total noise (combination of thickness, loading, and broadband self) is presented in Figure 16 for all nine microphone locations. The frequency spans from 0 to 16,000 Hz for both CAMRAD II and CHARM. The total noise results were obtained by logarithmically adding the tonal noise spectra and the broadband self-noise spectra. CAMRAD II generated results are shown as a black dashed line, and CHARM results are shown as a red solid line. Differences seen between the two codes for both the tonal and self noise propagate to the total noise, showing some differences in results between CAMRAD II and CHARM.

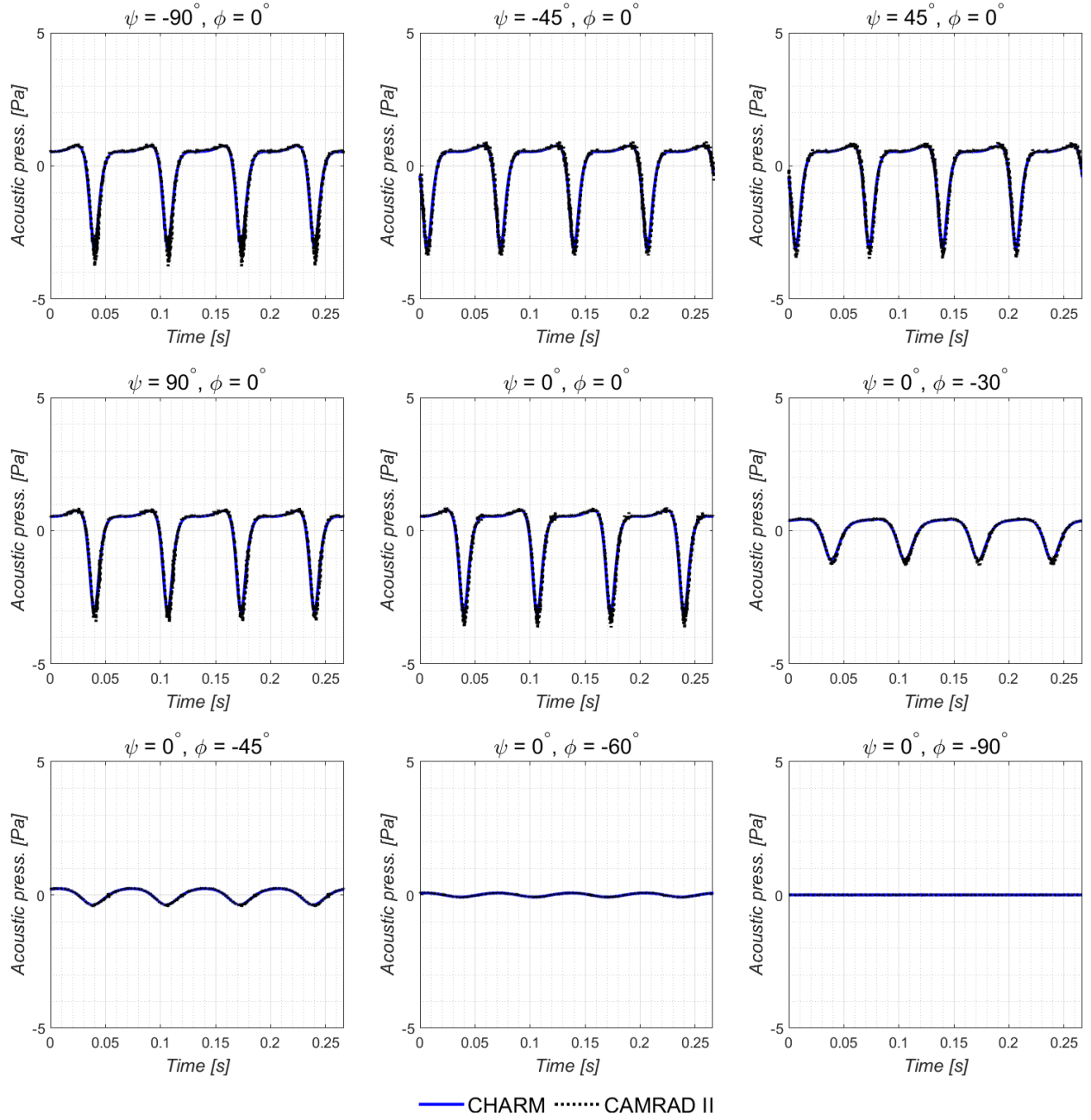


Figure 13.—Thickness noise acoustic pressure time history with CAMRAD II and CHARM using AARON for the simple rotor in forward flight with a free wake model ($V_{inf} = 33.5$ m/s, $\frac{C_T}{\sigma} = 0.001$, $\mu = 0.14$, $\alpha = 5^\circ$, $\theta_0 = -1^\circ$).

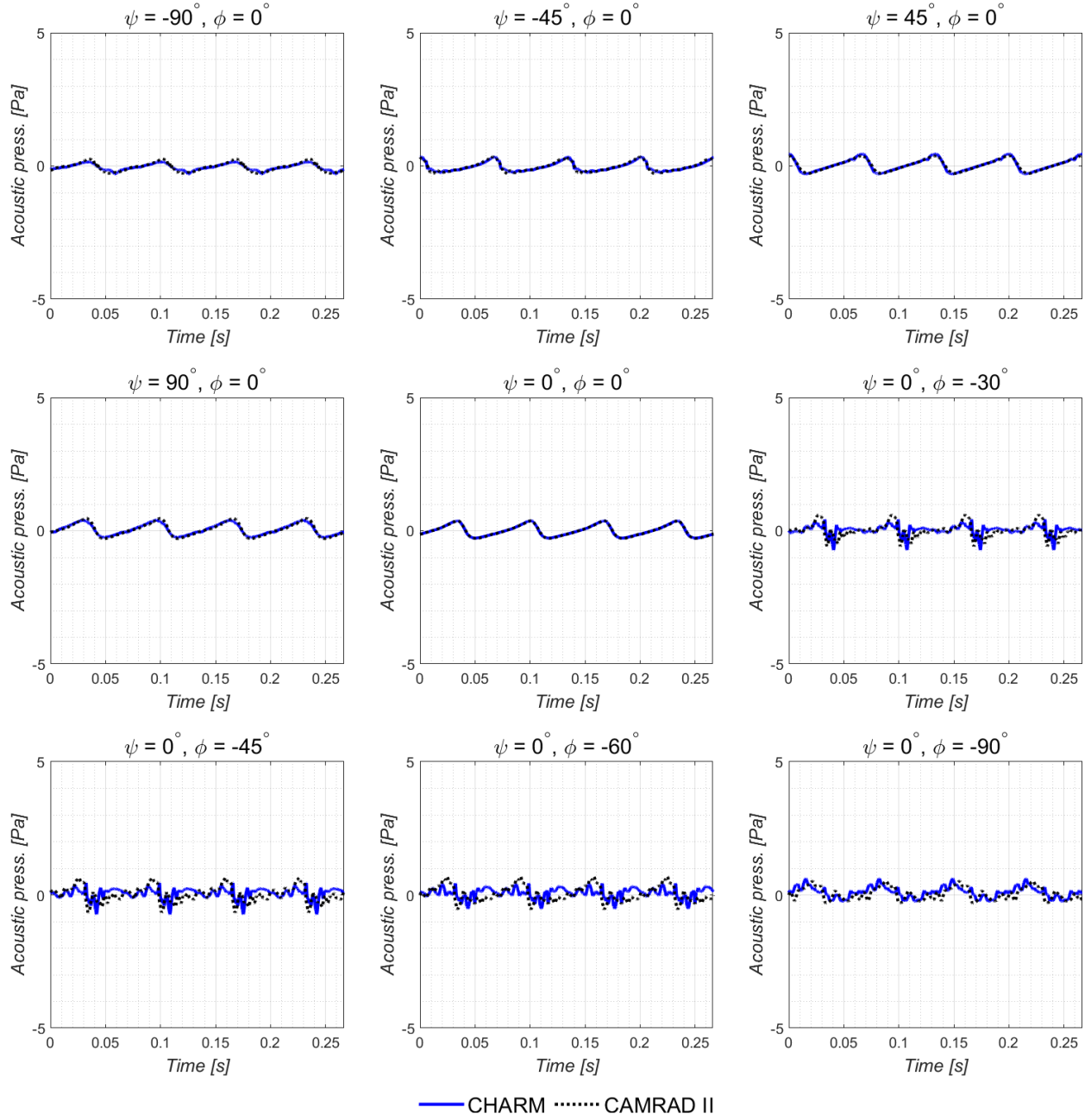


Figure 14.—Loading noise acoustic pressure time history with CAMRAD II and CHARM using AARON for the simple rotor in forward flight with a free wake model ($V_{inf} = 33.5$ m/s, $\frac{C_T}{\sigma} = 0.001$, $\mu = 0.14$, $\alpha = 5^\circ$, $\theta_0 = -1^\circ$).

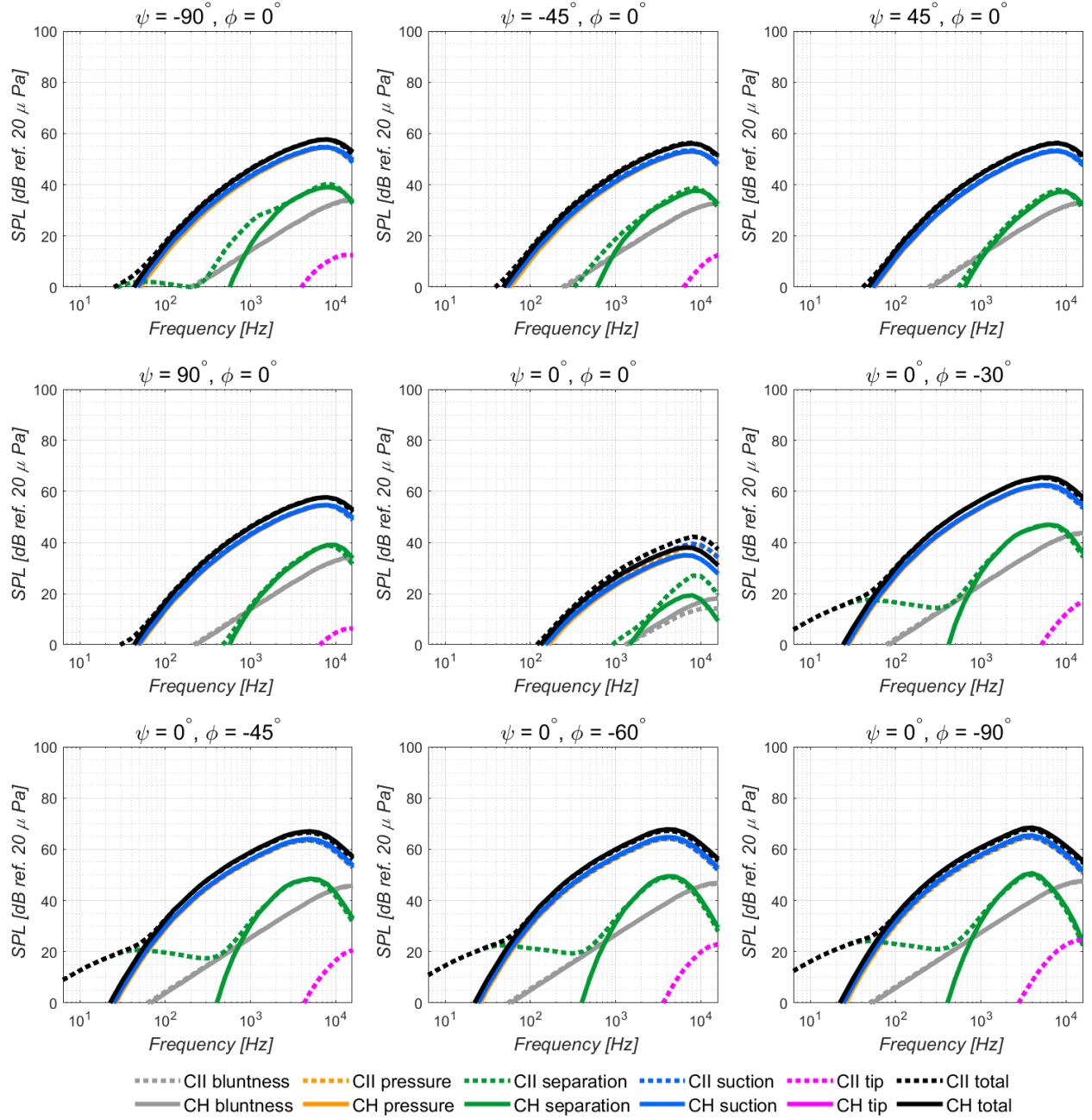


Figure 15.—Broadband self-noise spectra with CAMRAD II and CHARM using AARON for the simple rotor in forward flight with a free wake model ($V_{inf} = 33.5$ m/s, $\frac{C_T}{\sigma} = 0.001$, $\mu = 0.14$, $\alpha = 5^\circ$, $\theta_0 = -1^\circ$).

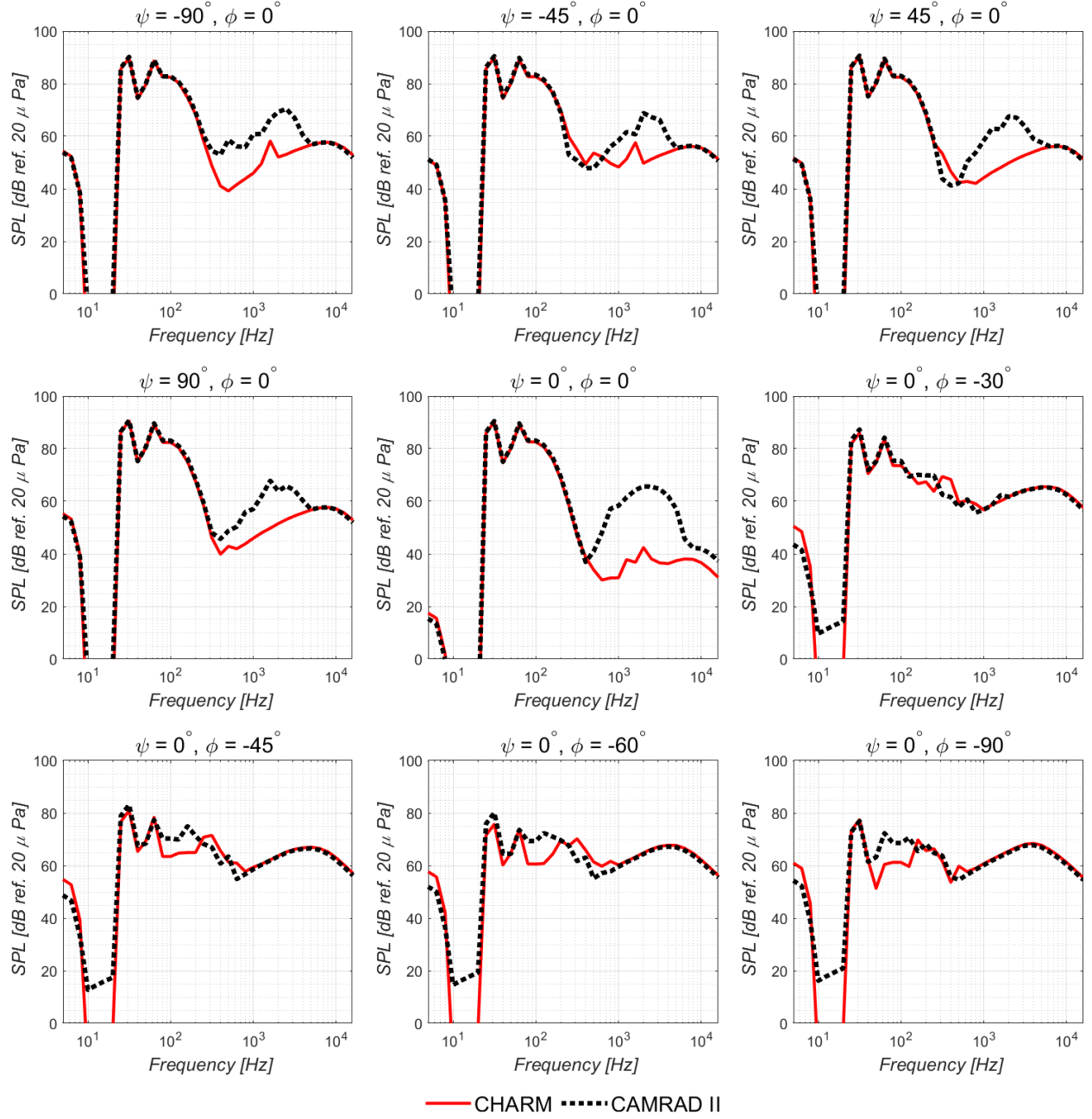


Figure 16.—Total noise spectra with CAMRAD II and CHARM using AARON for the simple rotor in forward flight with a free wake model ($V_{inf} = 33.5$ m/s, $\frac{C_T}{\sigma} = 0.001$, $\mu = 0.14$, $\alpha = 5^\circ$, $\theta_0 = -1^\circ$).

6.2.2 QSMR

The second rotor considered, the main rotor of the QSMR, is a rotor designed for quiet flight. The rotor geometry was created with acoustics as a major design parameter, while still maintaining adequate performance. Figure 17 and Figure 18 depict the acoustic pressure time history results for the QSMR in forward flight. Results are shown for a full rotor rotation for the nine microphone locations depicted in Figure 12, where ψ denotes azimuth angle and ϕ denotes elevation angle. Acoustic results produced using CHARM are shown as a blue solid line and acoustic results by CAMRAD II are shown as a black dashed line. Because the main rotor of the QSMR had strategic design decisions (see Ref. 17) made to reduce noise, it has a much lower thickness noise level when compared to the simple rotor. Any mics where the elevation is zero degrees has less than 1 Pascal of thickness noise, with the thickness noise becoming practically zero as the observer moves further out of plane of the rotor.

The loading noise for the QSMR has a higher acoustic pressure than that of the simple rotor, which makes sense when comparing the blade loading and flight speed between the two rotors. The QSMR is moving at 53.9 m/s compared to the simple rotor at 33.5 m/s. Additionally, the QSMR has a C_T/σ of 0.08, while the simple rotor has a C_T/σ of 0.001. The higher blade loading generated by the QSMR leads to an increase in noise. Additionally, by comparing the acoustic results in Figure 18 with the loading disk metadata from Figure 11, it is clear that there are some sort of wake or vortex interactions across the loading disk, causing the lines across the disk as seen in Figure 11. This presents itself in the loading noise, characterized by a sharp peak and dip in the noise results. Figure 11 also explains the differences between the acoustic results, as CAMRAD II is predicting stronger interactions compared to CHARM, which translates to higher loading noise peaks. Other than those peaks, there is generally good agreement between CAMRAD II and CHARM for the loading noise. Getting an exact match between the two codes would be difficult without experimental data to be able to fine-tune the model, as it is difficult to tell which code is more ‘correct’ or if the results are somewhere in the middle.

The broadband self-noise results for the QSMR are shown in Figure 19 as acoustic spectra. Like with the simple rotor, results are shown for total self-noise as well as for the various self-noise components for both CAMRAD II and CHARM. The total self-noise of the QSMR varies between CAMRAD II and CHARM, depending on how much turbulent boundary layer separation self-noise is being predicted. CHARM predicts higher noise for separation, which increases the total noise results. Generally, the QSMR produced less self-noise than the self-noise produced by the simple rotor.

The total noise results are shown in Figure 20 for all nine microphone locations. Results are shown from 0 to 16,000 Hz for both CAMRAD II and CHARM, with CAMRAD II generated results shown as a black dashed line, and CHARM results shown as a red solid line. As with the simple rotor, there are some noticeable differences between the CAMRAD II and CHARM results, due to differences in both the tonal and broadband self-noise results.

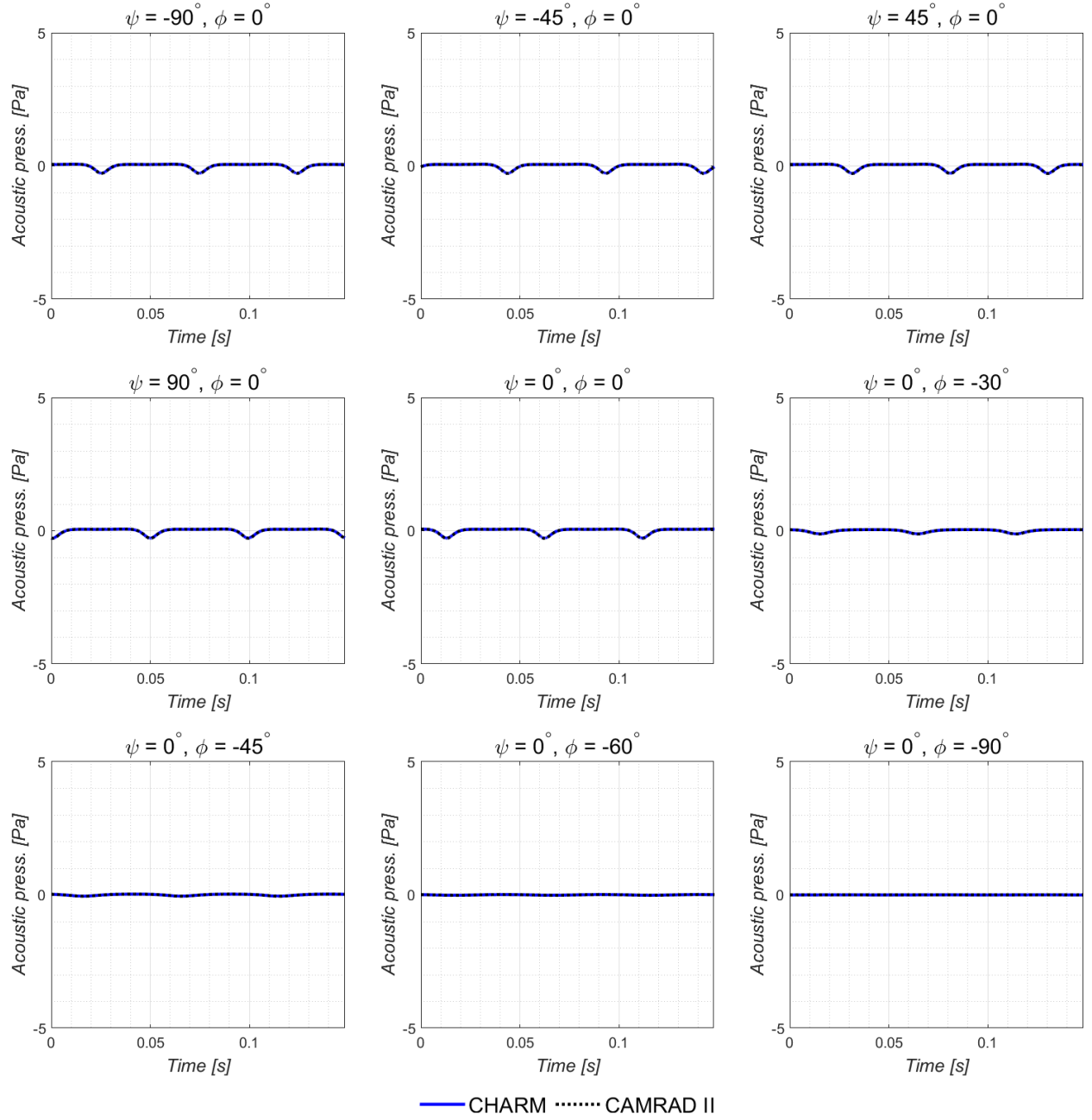


Figure 17.—Thickness noise acoustic pressure time history with CAMRAD II and CHARM using AARON for the QSMR in forward flight with a free wake model ($V_{inf} = 54$ m/s, $\frac{C_T}{\sigma} = 0.08$, $\mu = 0.25$, $\alpha = 1.4^\circ$, $\theta_0 = 4.7^\circ$).

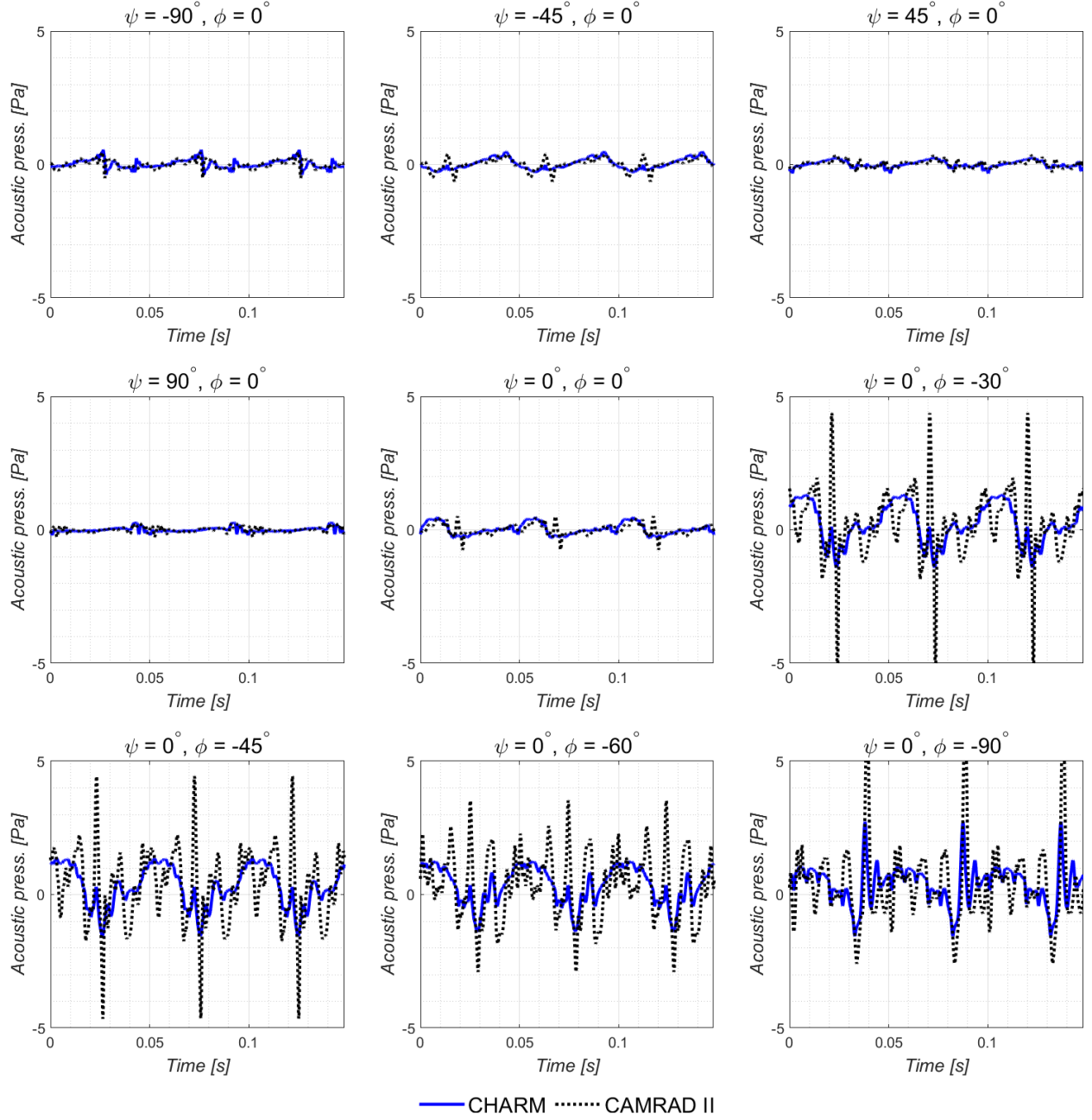


Figure 18.—Loading noise acoustic pressure time history with CAMRAD II and CHARM using AARON for the QSMR in forward flight with a free wake model ($V_{inf} = 54$ m/s, $\frac{C_T}{\sigma} = 0.08$, $\mu = 0.25$, $\alpha = 1.4^\circ$, $\theta_0 = 4.7^\circ$).

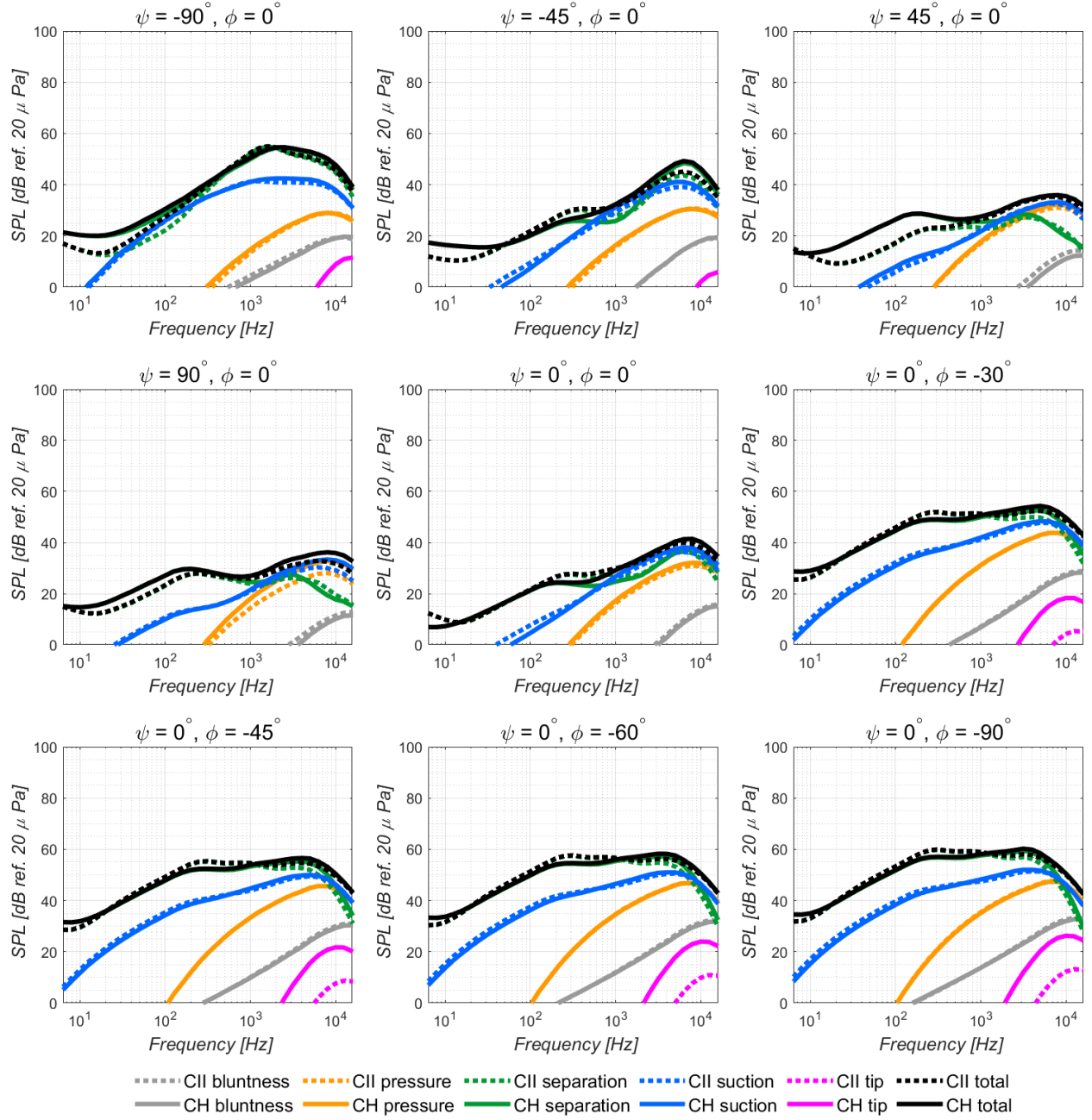


Figure 19.—Broadband self-noise spectra with CAMRAD II and CHARM using AARON for the QSMR in forward flight with a free wake model ($V_{inf} = 54$ m/s, $\frac{C_T}{\sigma} = 0.08$, $\mu = 0.25$, $\alpha = 1.4^\circ$, $\theta_0 = 4.7^\circ$).

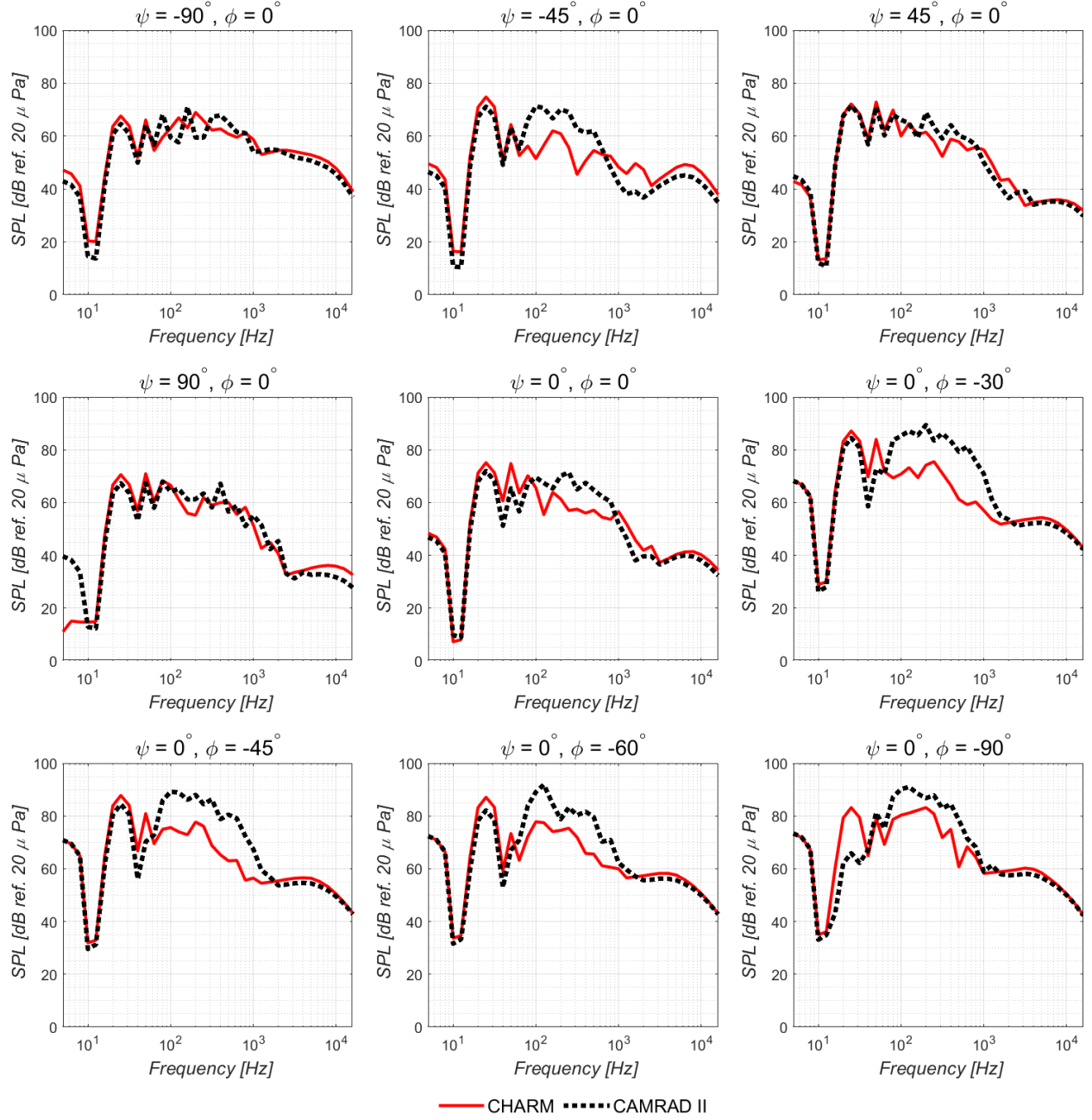


Figure 20.—Total noise spectra with CAMRAD II and CHARM using AARON for the QSMR in forward flight with a free wake model ($V_{inf} = 54$ m/s, $\frac{C_T}{\sigma} = 0.08$, $\mu = 0.25$, $\alpha = 1.4^\circ$, $\theta_0 = 4.7^\circ$).

6.2.3 Side-by-Side

The final rotorcraft considered was the Side-by-Side. The rotors from the Side-by-Side were selected in order to highlight rotor-rotor interactions and their influence on noise. Figure 21 and Figure 22 depict the acoustic pressure time history thickness and loading results for the Side-by-Side. Results are shown for the first full rotor rotation for the nine microphone locations depicted in Figure 12, where ψ denotes azimuth angle and ϕ denotes elevation angle. Acoustic results produced using CHARM are shown as a blue solid line and acoustic results by CAMRAD II are shown as a black dashed line. The thickness noise results are similar in magnitude to those of the QSMR, but as there are more blades, there are more blade passages in the time history. Generally, for each rotation, the number of thickness noise spikes corresponds to the number of blades. The microphones between ψ of -45° and 45° have fewer passages than microphones between ψ of -90° and 90° due to the radial locations of the microphones.

Like the thickness noise results, the loading noise results have more blade passage peaks due to the increased number of blades. Also, due to the interactions between the two rotors, the results are less smooth than those of the simple rotor or the QSMR. CAMRAD II, in particular, predicts high loading noise peaks for the out-of-plane microphones, caused by some sort of rotor-wake interaction between the two rotors. The loading noise is relatively low in-plane, except for the $\psi = -90^\circ$ case, for which CHARM is predicting higher loading noise. Again, this could be due to the rotor interactions.

Figure 23 shows the broadband self-noise results for the Side-by-Side rotor. These results fall somewhere between the simple rotor and the QSMR in terms of magnitude. Like the simple rotor and QSMR, the dominant noise source is boundary layer separation noise. The other noise sources are much lower in magnitude and do not contribute as much to the total broadband self-noise. The maximum self-noise cases are found in the out-of-plane mics, with mics six through nine having similar magnitude broadband results.

Total noise spectra results are shown in Figure 24. Results are plotted from 0 to 16,000 Hz for all nine microphones for both CAMRAD II and CHARM. CAMRAD II results are represented by a black dashed line, and CHARM results are represented by a red solid line. Like with the last two cases, the results between CAMRAD II and CHARM are quite similar, with some variation between the two due to differences in the models.

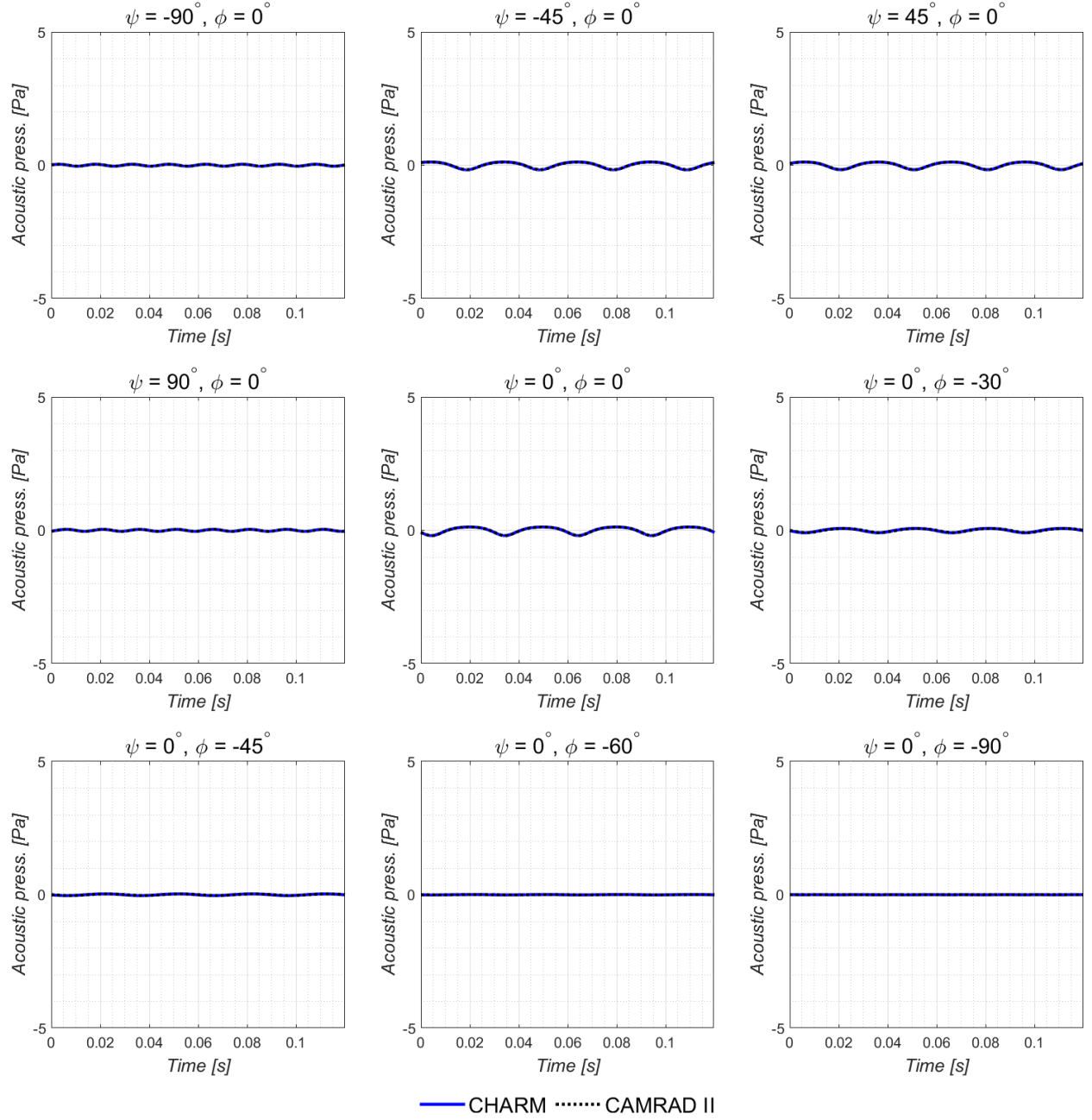


Figure 21.—Thickness noise acoustic pressure time history with CAMRAD II and CHARM using AARON for the Side-by-Side in forward flight with a free wake model ($V_{inf} = 25.72$ m/s, $\frac{C_T}{\sigma} = 0.093$, $\mu = 0.15$, $\alpha = 0.6^\circ$, $\theta_0 = 6^\circ$).

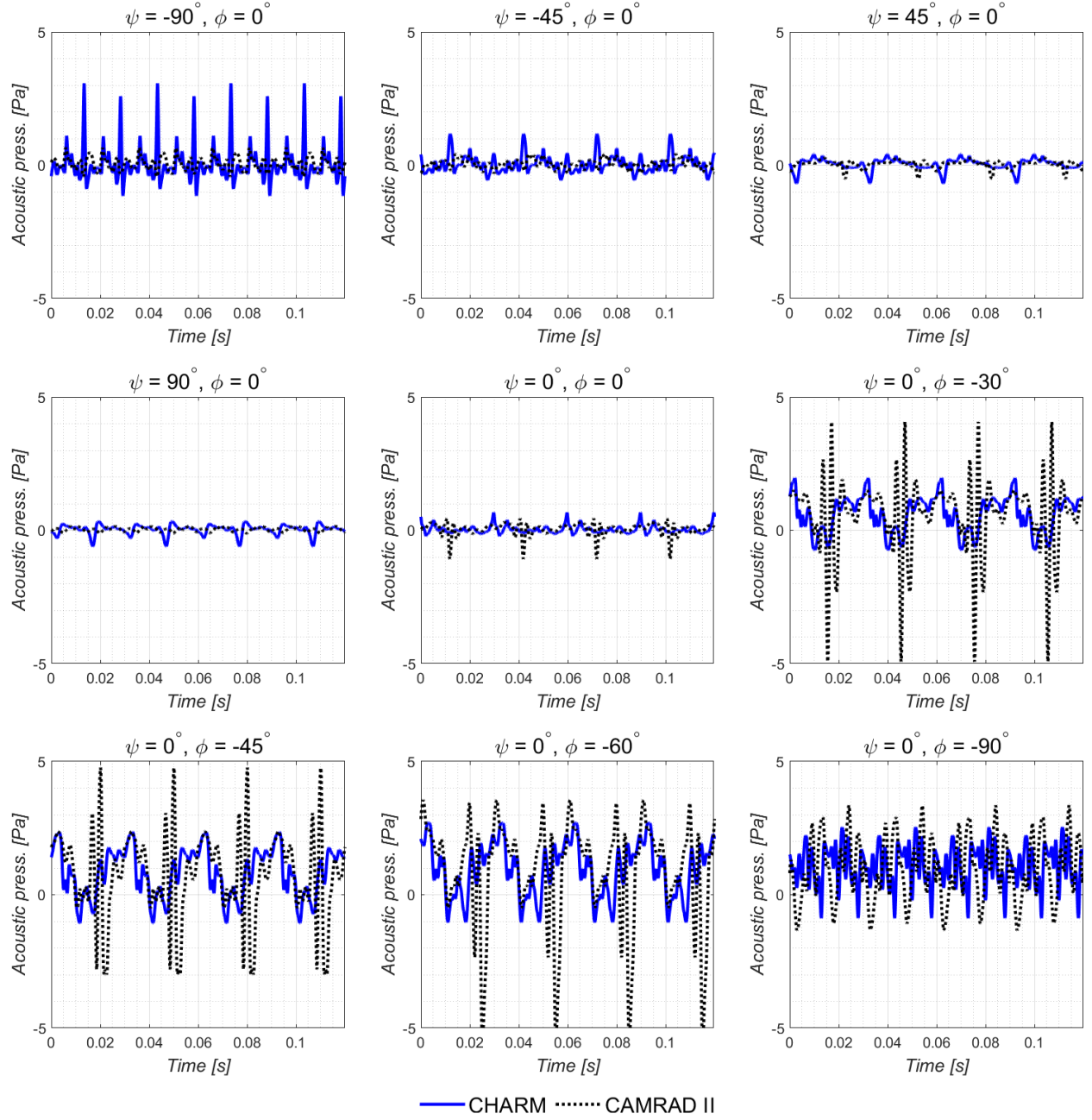


Figure 22.—Loading noise acoustic pressure time history with CAMRAD II and CHARM using AARON for the Side-by-Side in forward flight with a free wake model ($V_{inf} = 25.72$ m/s, $\frac{C_T}{\sigma} = 0.093$, $\mu = 0.15$, $\alpha = 0.6^\circ$, $\theta_0 = 6^\circ$).

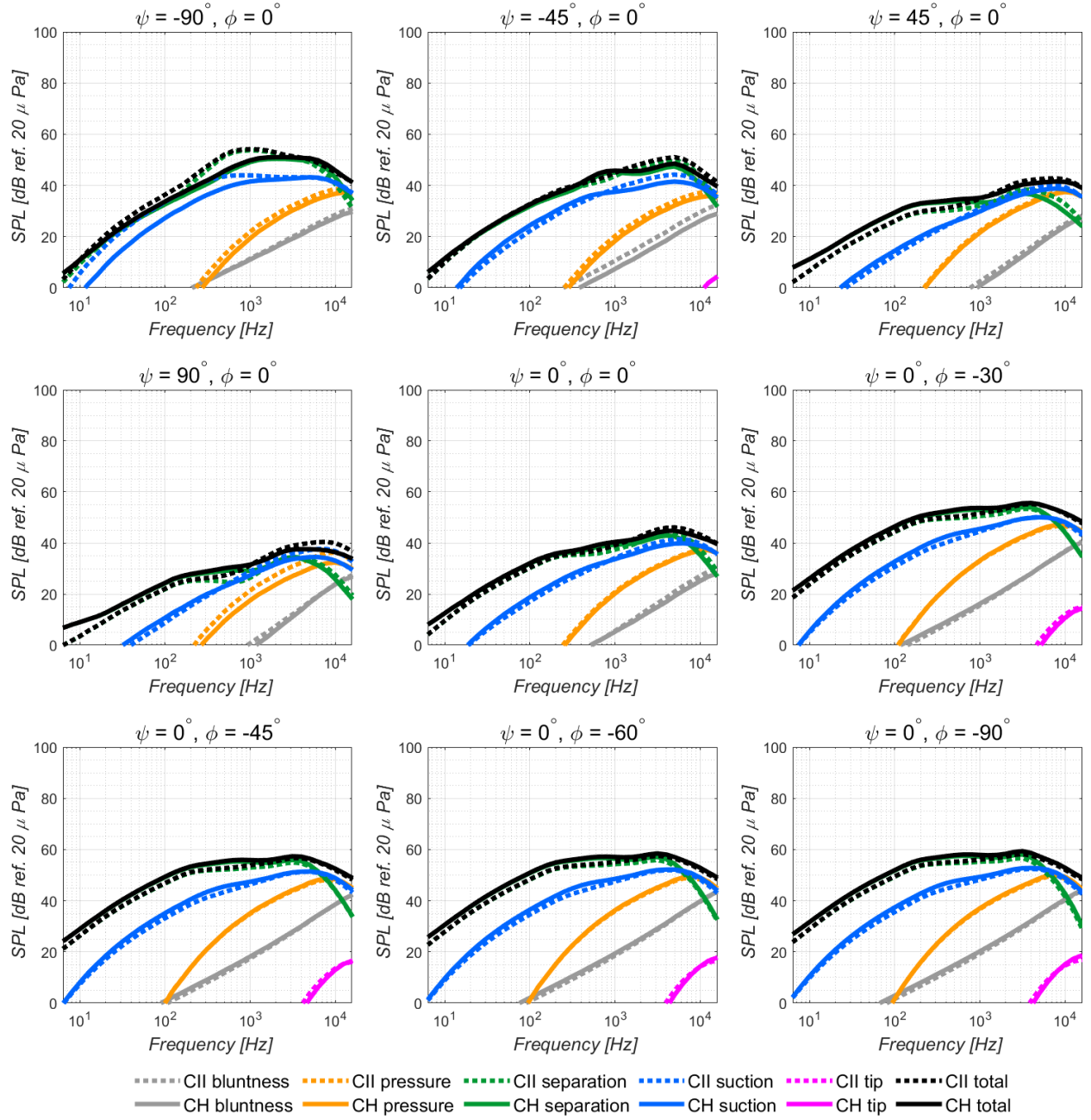


Figure 23.—Broadband self-noise spectra with CAMRAD II and CHARM using AARON for the Side-by-Side in forward flight with a free wake model ($V_{inf} = 25.72$ m/s, $\frac{C_T}{\sigma} = 0.093$, $\mu = 0.15$, $\alpha = 0.6^\circ$, $\theta_0 = 6^\circ$).

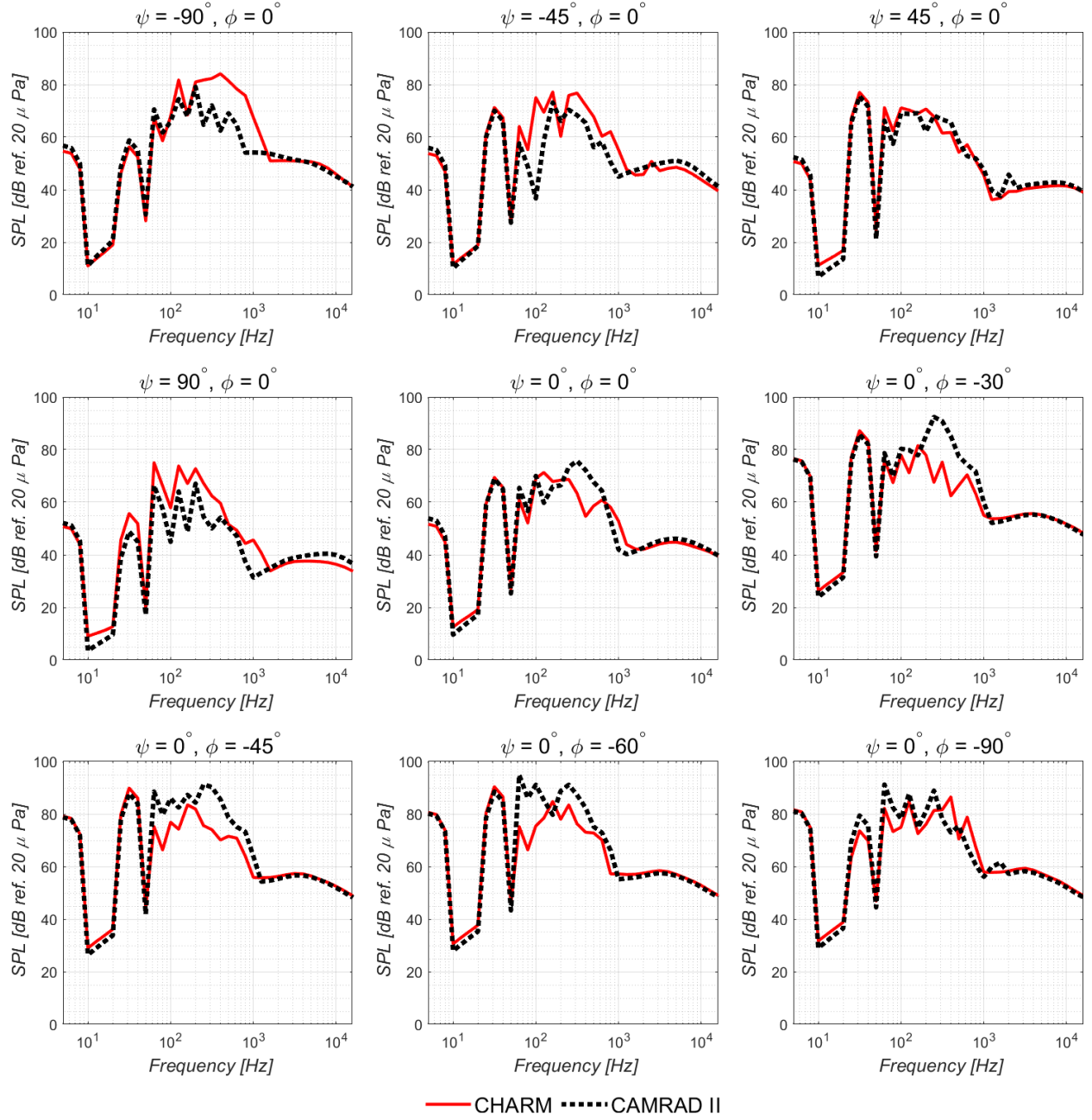


Figure 24.—Total noise spectra with CAMRAD II and CHARM using AARON for the Side-by-Side in forward flight with a free wake model ($V_{inf} = 25.72$ m/s, $\frac{C_T}{\sigma} = 0.093$, $\mu = 0.15$, $\alpha = 0.6^\circ$, $\theta_0 = 6^\circ$).

7 Acoustic prediction for experimental comparison

To develop best practices for most common use cases of AARON, post-test predictions for an experimental effort were conducted. This work was coupled with a study into the accuracy of phased arrays with an isolated DJI Phantom 3 rotor, and so results are shown for CAMRAD II, CHARM, a single mic, and a phased array. This section includes a full test description as well as post-prediction efforts and best practices for both CAMRAD II and CHARM. Often, when comparing with experimental data, comprehensive analysis results require tuning, as experimental conditions are never as ideal or exact as predictions are. To facilitate further studies with this data and to make the work repeatable, test data can be made available.

7.1 Experimental isolated rotor in hover

In Spring 2025, an isolated DJI Phantom 3 rotor was tested in the NASA Ames Research Center Anechoic Chamber to assess the use of microphone phased arrays for measuring loading and thickness noise of a hovering rotor. This study was selected to be included in the best practices as it was a recent and fundamental test conducted in an anechoic environment, which made it ideal for comparison with comprehensive analysis predictions. Details of the test’s facility, rotor, and microphones are provided below.

7.1.1 Testing facility

The NASA Ames Research Center Anechoic Chamber is located under the National Full-Scale Aerodynamics Complex 40- by 80-Foot Wind Tunnel. The chamber is 25 feet by 18 feet measured from the tips of the foam wedges that serve to make the chamber anechoic. The facility is 11 ft tall, also measured from the tips of the wedges. The wedges are 18 inches deep, resulting in a chamber that is anechoic down to 250 Hz. For more information, see Ref. 21. The chamber has no free-stream velocity, making it a hover only chamber, but it does include an exhaust duct, which is used to reduce circulation from the rotor by venting at a low speed.

7.1.2 Rotor geometry

The DJI Phantom 3 isolated rotor test used a commercial-off-the-shelf, two-bladed, 23.88 cm (0.87343 inches) diameter rotor, as shown in Figure 25. The blades are plastic and made via injection molding. Prior to testing, the blade was 3D scanned to identify airfoil properties for use within simulation efforts. As the airfoil sections are not publicly released, this was the best way to obtain geometry for use with prediction. As expected, results revealed spanwise variation of airfoil geometry, twist, and chord. Figure 26 shows a graphical representation of the airfoil distribution. Figure 27 shows the chord length and twist distributions for the rotor as a function of fractional radius, all obtained from the 3D scan of the blade. Additional testing was also conducted to determine material properties and deflections, which all fed into the comprehensive analysis models of the rotor.

7.1.3 Rotor test set-up and operation

A single DJI Phantom 3 rotor was mounted on a sting with a constant shaft angle of 0 degrees in the anechoic chamber, shown in Figure 28. RPM was the only control method, with a maximum



Figure 25.—Photograph of the two-bladed DJI Phantom 3 rotor.

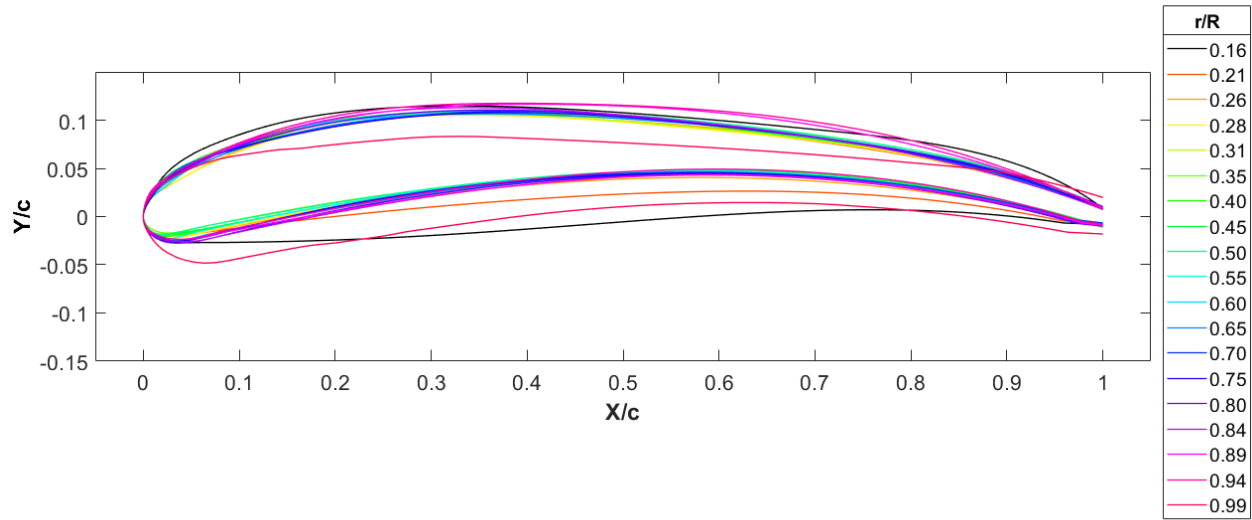


Figure 26.—DJI Phantom 3 airfoil distribution from an in-house 3D scan.

of 8,000 due to a motor current limit. Motor voltage and current were averaged for each data point, and RPM was measured at the acoustic sampling rate of 102.4 kHz. Forces and moments were assumed to be similar to those determined by Russell et al. (Ref. 22). These forces and moments were determined from the testing of an isolated DJI Phantom 3 rotor in the U.S. Army 7- by 10-ft Wind Tunnel at NASA Ames Research Center. As with the current test, RPM was the only control variable used in the wind tunnel test. In order to use the results from Ref. 22 as a baseline, a similar RPM sweep to that conducted in the wind tunnel was used, specifically 2,500 to 8,000 RPM.

7.1.4 Rotor control

The rotor was mounted directly to a DC motor that was controlled via a Turnigy Plush 30A Electronic Speed Control (ESC) mounted on the test stand. The speed control signal for the ESC was generated by a Pololu Mini Maestro 12-channel servo controller connected to the USB port of the data acquisition computer. Motor power was supplied by a DCS 40-25 power supply capable of

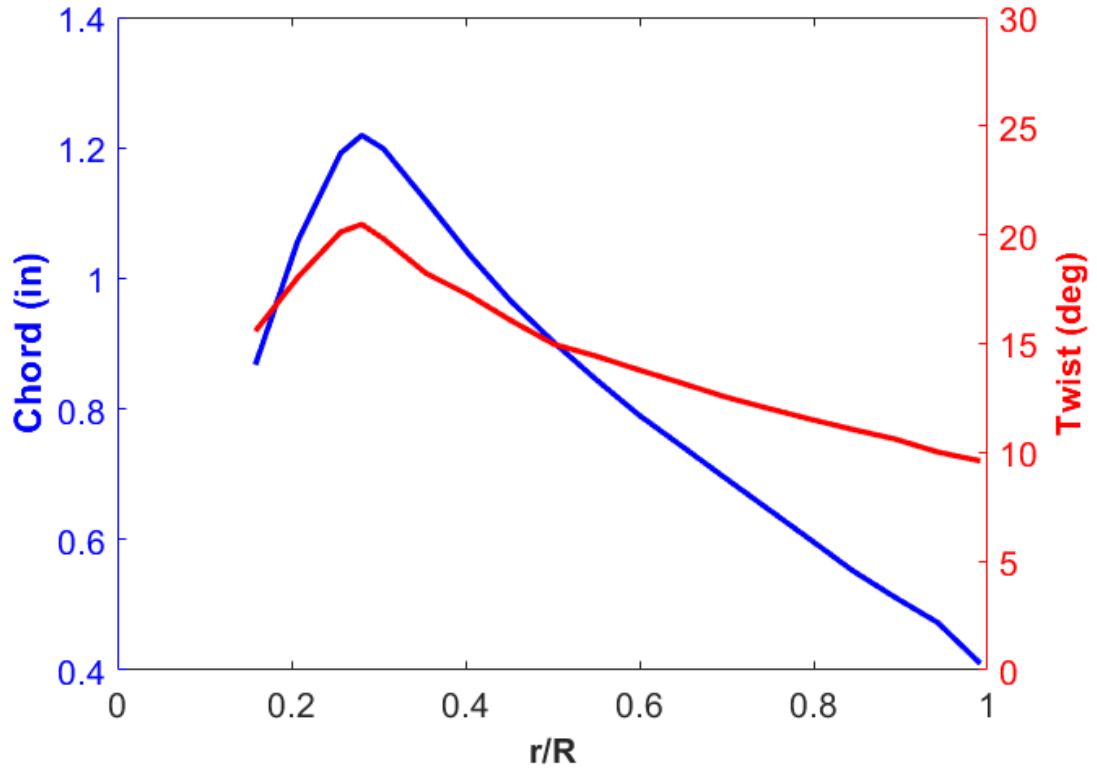


Figure 27.—DJI Phantom 3 blade chord and twist distributions from an in-house 3D scan.

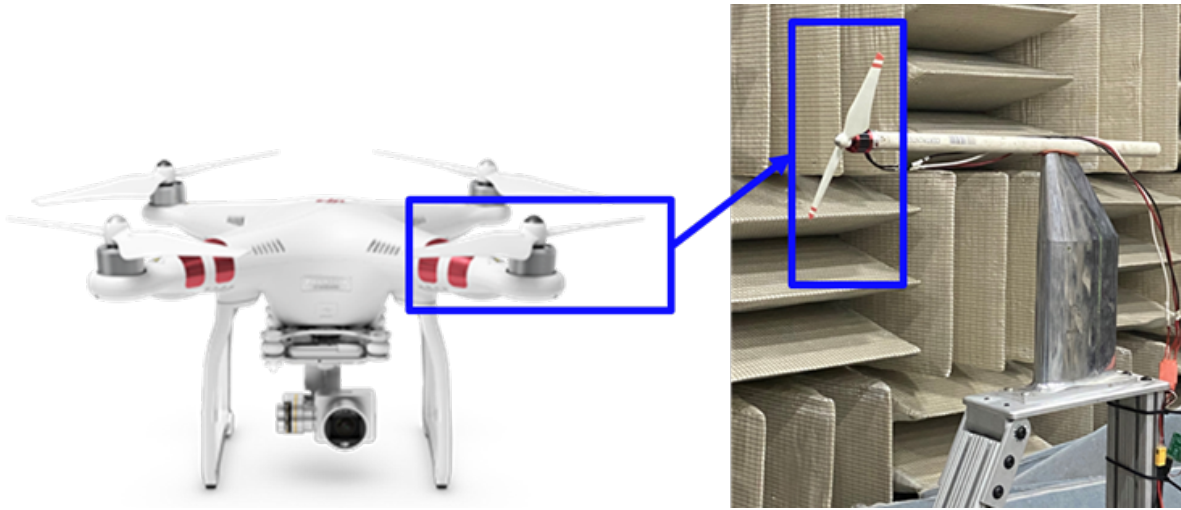


Figure 28.—Left: DJI Phantom 3, Right: DJI Phantom 3 isolated rotor mounted in the anechoic chamber.

supplying up to 1 kW of power (40 V at 25 A). The scripting feature of the Pololu servo controller was used to run an automated speed sweep with settle and ramp times. The servo controller also generated a trigger that started data acquisition to ensure steady, repeatable conditions.

7.1.5 Acoustic setup and acquisition

Acoustic data were acquired in the rotor plane at distances varying from 48 inches to 144 inches as well as at a single out-of-plane mic positioned below the rotor. For both observer angles, a single microphone and microphone phased array were installed in the chamber at the locations shown in Figure 29. The out-of-plane observer location was 188.5 inches (40.1 rotor radii) at 29.2° below the rotor. See Figure 30 for the installation of the rotor and microphone in the anechoic chamber.

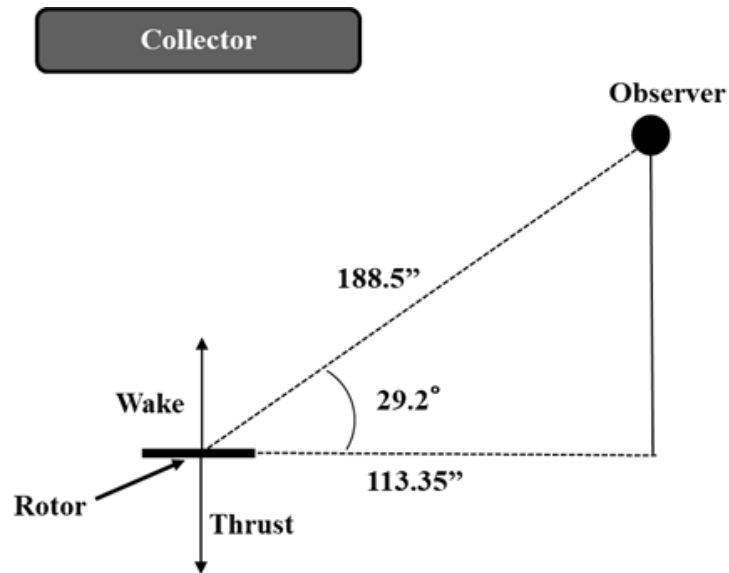


Figure 29.—Pictorial top view of microphone and rotor locations in the anechoic chamber (not drawn to scale).

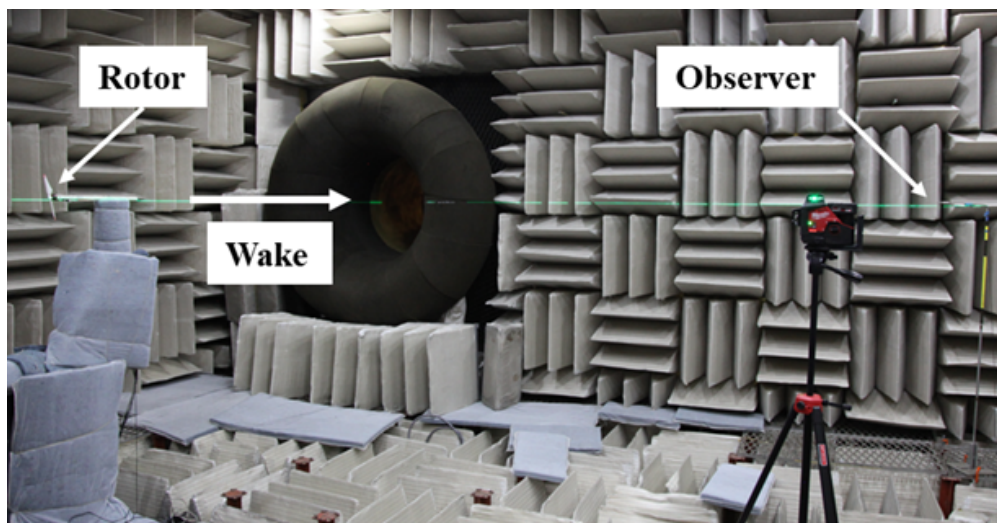


Figure 30.—Visual of DJI Phantom 3 isolated rotor in anechoic chamber with installation of microphone.

The single microphone was a GRAS $\frac{1}{4}$ inch free-field microphone model 40BF-S with the grid removed to reduce incidence angle effects. This microphone was mounted on an aluminum pole. The microphone array consisted of 24 GRAS $\frac{1}{4}$ 40BF-S microphones arranged in a 3-arm pattern with an 8 inch diameter as shown in Figure 31. The array fairing was designed as an in-flow array with a stainless-steel array windscreen; however, the windscreen was not needed for this study and was removed.



Figure 31.—Front view of 8-inch array with microphones flush mounted in a non-conductive plate.

Acoustic measurements for the single microphone and microphone phased array were acquired at a 102.4 kHz sampling rate. A laser tachometer was used to identify the 1/rev frequency of the rotor, and acoustic data was acquired for 32 seconds including settling time and RPM ramp time. Only 24 seconds of data was analyzed, ensuring that at least 1,000 revolutions were recorded for all RPMs (as shown in Figure 32). The blue line in the plot depicts the RPM of the rotor across the entire data point, and the vertical black line separates the data point into the corresponding phases.

A circulation checkout test was performed to ensure acoustic measurements contained no sign of recirculation effects for all RPMs (Ref. 23). Figure 33 shows a spectrogram of SPL over the entire duration of an 8,000 RPM test point to ensure recirculation was not encountered. The top of the plot shows the spectrogram itself as a function of frequency, time, and SPL. The top of the plot shows the rotor RPM for the duration of the data point, to show that the rotor RPM was spun up and then held steady. A lack of recirculation is indicated by the spectrogram containing straight lines for the entire duration of the rotor rotation. As recirculation essentially presents itself as a wake interaction, it would impact the SPL generated by the rotor, causing changes in the spectrogram.

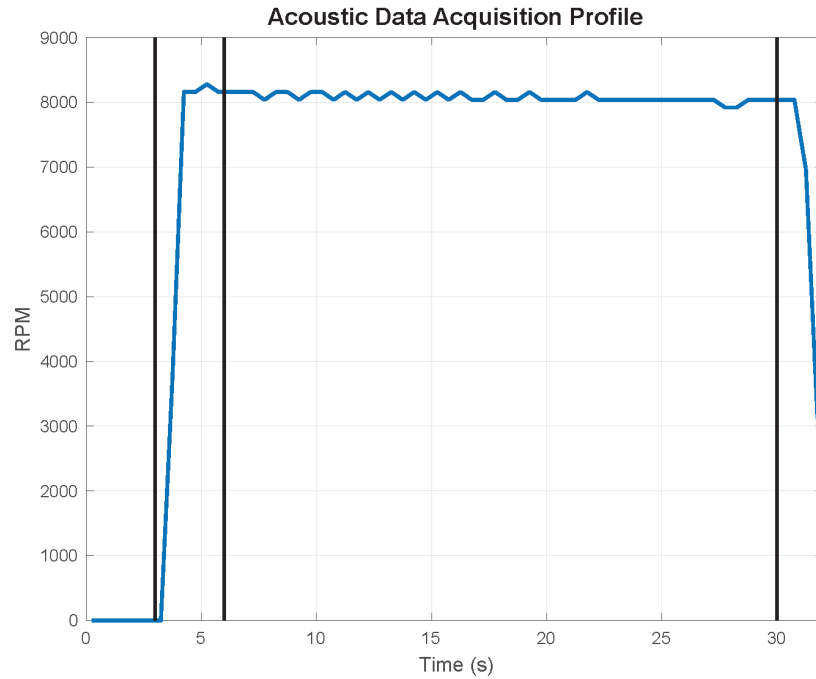


Figure 32.—RPM for each 32-second data point including the following phases: a) 4 seconds of rotor settle time, b) 2 seconds of RPM ramp and stabilization, c) 24 seconds of analysis, and d) 2 seconds of ramp down.

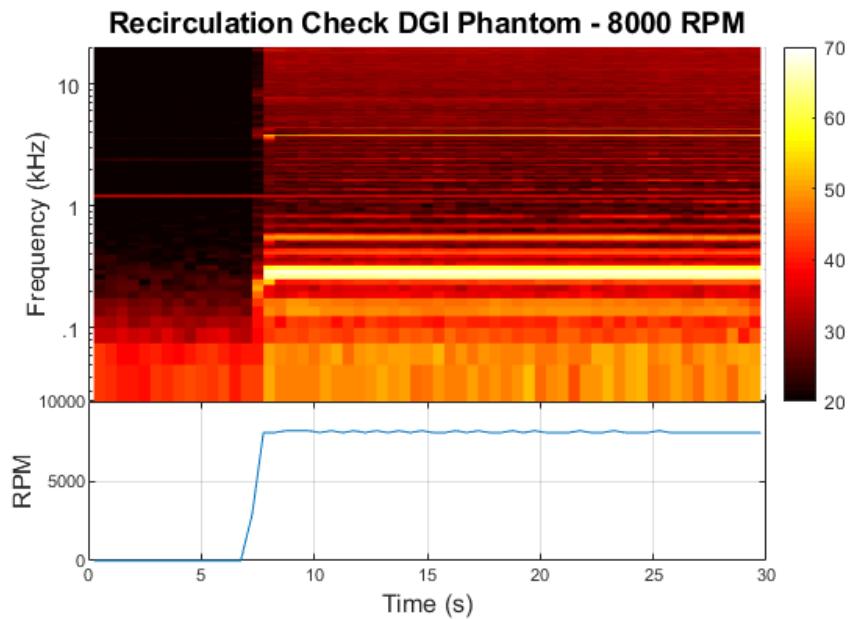


Figure 33.—Circulation checkout test for DJI Phantom 3 isolated rotor at 8,000 RPM in anechoic chamber with exhaust nozzle blocked.

An experimental near- and far-field study was conducted for the in-plane rotor data as shown in Figure 34. Experimental results are shown as solid lines in black, blue, and red for 2500 RPM, 5000 RPM, and 8000 RPM, respectively. The acoustic far-field is considered achieved when the OASPL results trend with $1/r$, where r is the distance from the rotor in terms of rotor radii. Therefore, the $1/r$ curves are shown as dashed lines for 2500, 5000, and 8000 RPM (once again in black, blue, and red, respectively). At the maximum RPM of 8,000 results show that as few as 10 rotor radii are needed to achieve acoustic far-field properties. However, below 8,000 RPM the acoustic far-field was beyond the farthest measurements.

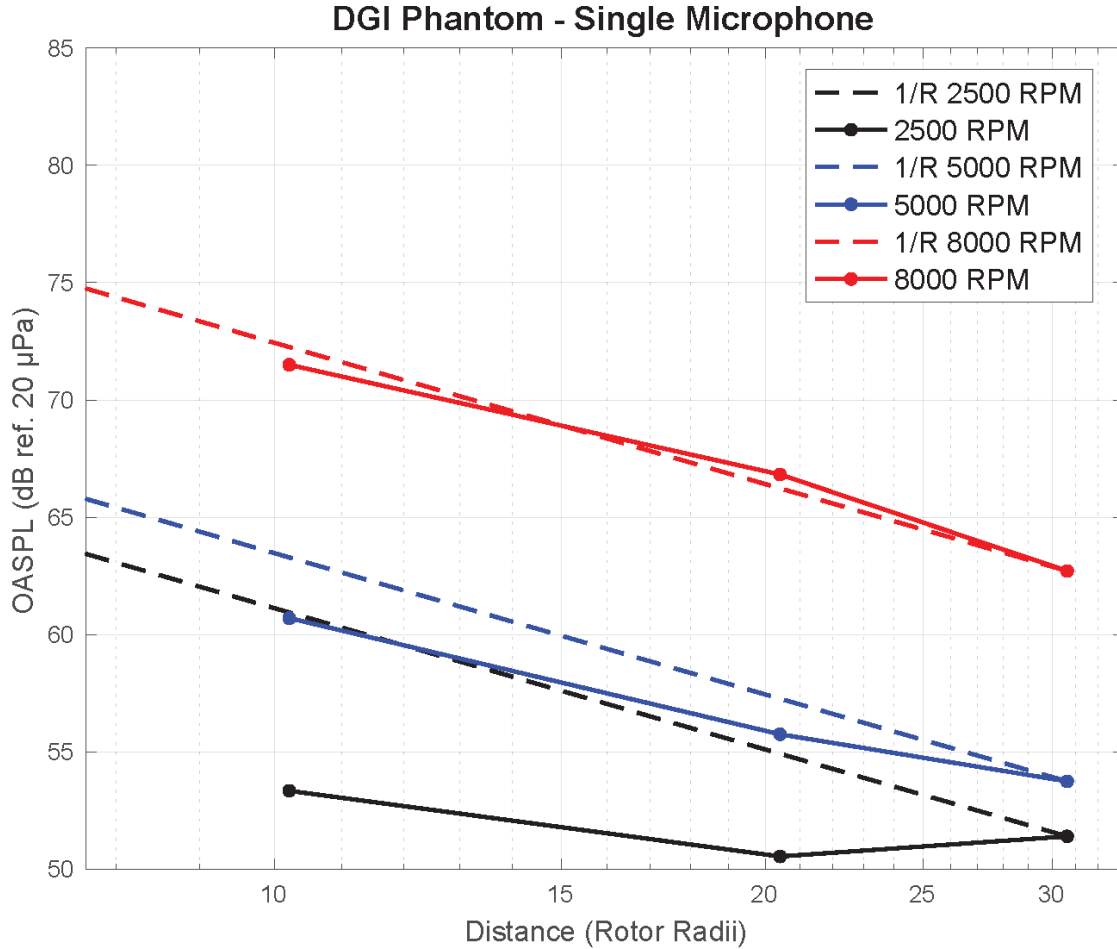


Figure 34.—In-plane near- and far-field experimental results for three rotor RPMs.

7.1.6 Acoustic data processing

To identify the thickness and loading noise the data were harmonically averaged using the pulse train from the 1/rev tachometer for each data point. Efforts were made to identify the maximum

positive pressure in the signal and associate the peak with 0 degrees azimuthal position making comparisons between the array, free-field microphone, and computational data easier. Array data were processed using the OptiNav Beamforming Interactive plugin for ImageJ, which produced a beamformed time signal that could then be harmonically averaged to match with the single mic results. Although not discussed here for brevity, 1/12-octave spectra were also produced using conventional beamforming and compared with free-field results. The 1/12-octave spectra results can be made available upon request.

7.2 Simulation setup

An isolated two-bladed DJI Phantom 3 rotor was simulated in CAMRAD II and CHARM using a free wake in hover. Outputs from CAMRAD II and CHARM were then passed to AARON in order to compute thickness and loading noise. Broadband noise was not considered for this effort as comparisons were made in the time domain, and BPM does not produce time domain results for broadband self-noise. Best practices from earlier in this paper were applied to this effort. Computational modeling assumptions, performance validation, and acoustics results are discussed.

7.2.1 Modeling assumptions

Due to the complex airfoil distribution of the rotor, only two airfoil sections were modeled using two in-house generated airfoil tables. The first airfoil extends from r/R of 0 to r/R of 0.85. The second airfoil extends from r/R of 0.85 to the tip. The same airfoil tables were used in both CAMRAD II and CHARM. Only the rotor blade geometry was modeled. The motor, test stand, and microphone strut noise are not accounted for in the simulation results. Additionally, facility limitations, such as any imperfections or reflections from the anechoic chamber, are not taken into account. Therefore, some differences between the experimental and prediction acoustic results are expected. Additionally, as previously mentioned, only thickness and loading noise were analyzed for this validation effort. Therefore, experimental results were averaged to remove broadband self-noise, and the BPM self-noise model was not used.

7.2.2 Comprehensive analysis settings

One major best practice takeaway from this study is that when comparing thrust between prediction and experiment, the the relationship between thrust and collective is rarely predicted exactly. It is a best practice to use collective to help match predicted thrust to the experimental measurements. One of the major reasons for the collective-thrust relationship to differ between prediction and experiment is the differences in airfoil coordinates or airfoil zero-lift angle of attack. For this effort, a collective of 7 degrees was needed in the model to account for these differences. This 7 degree collective value shifts the zero-lift angle of attack to match the experimental value, while leaving lift curve slope, inflow, and all other parameters unchanged.

Secondly, as mentioned earlier in the paper, OPSCEN in CAMRAD II can be used to set a standard scenario for predictions. For the forward flight cases discussed, the forward flight scenario, $OPSCEN = 1$ (for class=ROTOR, type=WAKE), was used. Since this prediction is for hover, the hovering scenario, $OPSCEN = 2$, should be used, to avoid the implementation of improper settings and the potential for convergence issues. In addition to these two best practices, all the best practices described above applied to this effort.

7.2.3 Performance validation

Prior to computing thickness and loading noise from the comprehensive model, the performance of the isolated DJI Phantom 3 rotor model in hover was validated to ensure the model was accurate. The study was done by comparing the thrust and power results from the comprehensive models with the thrust and power recorded experimentally from Reference 22 for a sweep from 2,500 RPM to 8,000 RPM. Figure 35 compares the electrical power and Figure 36 compares thrust, both comparing the comprehensive model with the results from experiment (including uncertainty). This was done as the current anechoic experiment did not use a load cell and therefore does not generate thrust data, although it did generate electrical power data, which is shown.

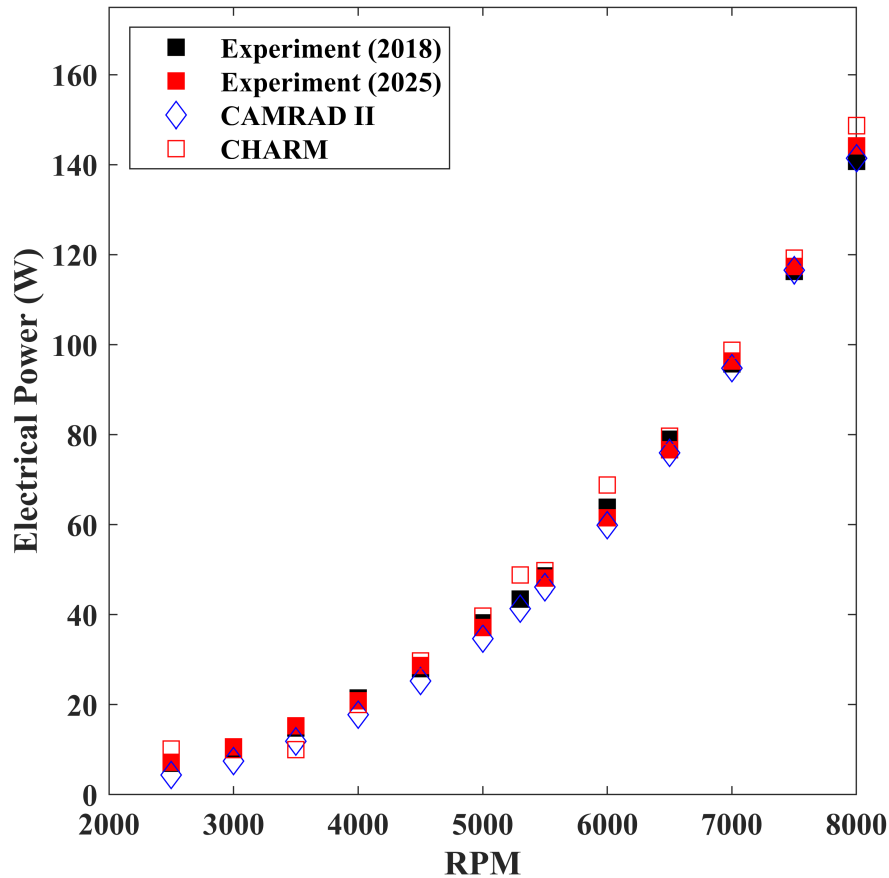


Figure 35.—Electrical power versus RPM for isolated DJI Phantom 3 in hover for experiment and simulation.

Figure 36 compares the thrust of the rotor in pounds versus RPM for the 2018 experiment, CAMRAD II, and CHARM. The experimental data is shown as black solid squares, CAMRAD II is hollow blue diamonds, and CHARM is red hollow squares. Error bars are provided to show the acceptable variation in the data. These results show that the CAMRAD II and CHARM predictions are matching the experiment within the expected limits. This validates the models and their yielded acoustic results.

Additionally, electrical power from both the 2018 and 2025 experiments were compared with

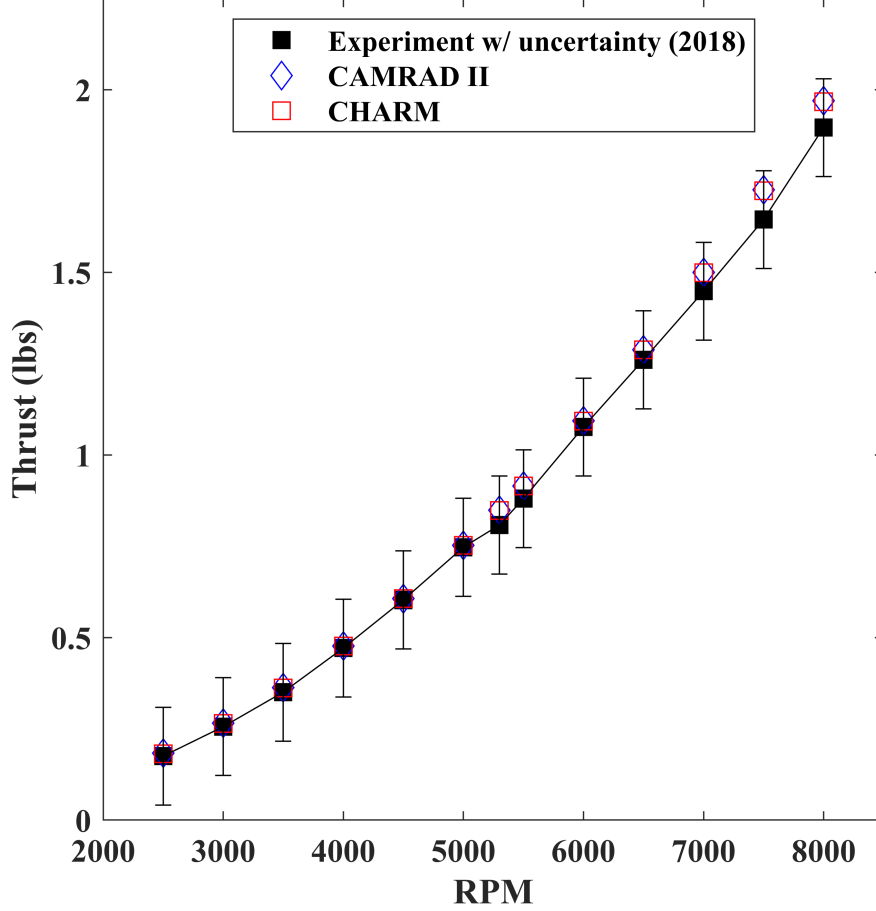


Figure 36.—Thrust versus RPM for isolated DJI Phantom 3 in hover for experiment and simulation.

CAMRAD II and CHARM predictions in Figure 35. The experimental data is once again shown with solid squares (black for 2018 and red for 2025), CAMRAD II is shown with hollow blue diamonds, and CHARM is shown with hollow red squares. The power results between the 2018 and 2025 tests match well, showing that the similar setups were able to produce comparable results. The CAMRAD II and CHARM results also align well with experiment, following the curve as expected.

To further compare the results between CAMRAD II and CHARM, z loading metadata disk plots were generated for both CAMRAD II and CHARM for 4,000 RPM, 6,000 RPM, and 8,000 RPM (see Figure 37). These plots show one revolution of data for one blade. As expected, there is a difference in force between each RPM, with a higher RPM corresponding to a higher z loading. Also as expected, there are minimal differences between CAMRAD II and CHARM, as z-loading and thrust are similar parameters, and thrust was shown to match well between the two codes (Figure 36). There are slight differences between the CAMRAD II and CHARM models, leading to some minor differences in the radial distributions of z loading. But, generally, results between CAMRAD II and CHARM match, meaning there should be agreement in loading noise between the two models.

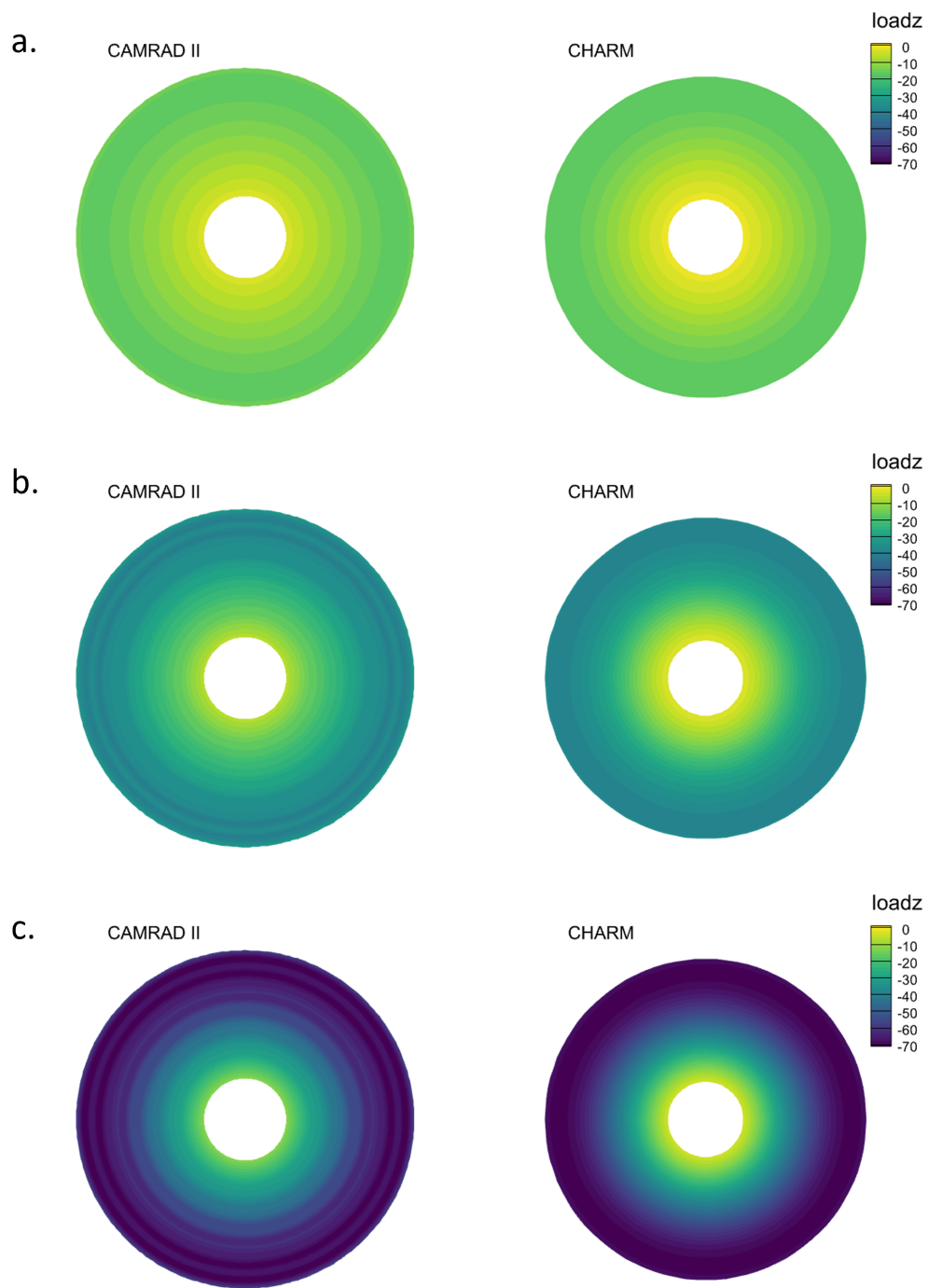


Figure 37.—DJI Phantom 3 isolated rotor free wake, single blade z loading metadata for CAMRAD II and CHARM for a.) 4,000 RPM, b.) 6,000 RPM, and c.) 8,000 RPM.

7.3 Experimental and computational acoustic results and validation

The experimental and computational total noise acoustic time histories for 4,000, 6,000, and 8,000 RPM are presented in Figure 38. Please note that subplots a, b, and c all have different vertical axis ranges. Experimental results provided include array microphone 1 (a single mic within the array), time domain beamforming (the total array result), and measurements from the singular free-field microphone. Computational results provide total tonal noise from CAMRAD II and CHARM for a single microphone placed in the same location of the array and single mic. Results provided include unfiltered and filtered time histories.

For each RPM condition presented, a Butterworth band-pass filter was applied to both the experimental and computational acoustic time histories to reduce the noise in the data and suppress broadband self-noise and extraneous noise sources such as stand vibrations. For each RPM condition, the Butterworth band-pass filter order and limits were chosen based on how well the computational tonal noise predictions were preserved before and after filtering. The Butterworth order and limits for each prediction condition were then applied to the corresponding experimental condition to remove non-tonal noise sources. As shown in Figure 38, there are minimal differences between unfiltered and filter acoustic prediction results for CAMRAD II and CHARM. The mean acoustic pressure is captured for both the filtered and unfiltered data, with the filters smoothing the results of the experiment. Experimental acoustic results compare closer to predictions when the Butterworth band-pass filter is applied for all conditions, as it removes noise sources not modeled in the predictions.

Unfiltered experimental results for 4,000 RPM reveal significant high frequency content, though the blade pulse locations (phase) compare well against CAMRAD II and CHARM prediction locations, see Figure 38a. A 3rd order Butterworth band-pass filter of 7 Hz and 533 Hz was selected to ensure computational prediction tonal noise was preserved. Applying the filter reduced high frequency content in the experimental results and peak-to-peak differences were reduced.

As RPM is increased to 6,000 RPM (Figure 38b), high frequency content is present in the unfiltered experimental results, though the total noise compares well in term of magnitude and phase to predictions from CAMRAD II and CHARM. A 3rd order Butterworth band-pass filter of 10 Hz and 1,000 Hz was selected to ensure computational prediction tonal noise was preserved. Similar to the 4,000 RPM case, applying the filter reduced high frequency content in the experimental results and peak-to-peak discrepancies with the predictions were reduced.

An RPM of 8,000 resulted in larger differences in peak-to-peak noise between experiment and predictions. As RPM is increased to 8,000 RPM (Figure 38c), some high frequency content is present in the unfiltered experimental results, though the total noise compares well in terms of magnitude and phase to predictions from CAMRAD II and CHARM. A 1st order Butterworth band-pass filter of 13 Hz and 1,333 Hz was selected to ensure tonal noise was preserved. Applying the filter reduced the high frequency content in the experimental results, though only a slight improvement in the experiment/prediction correlation of the peak-to-peak noise was observed.

A difference in magnitude between the experimental and predicted results is observed, though phase is predicted well for all conditions presented. These differences may be due to the reduced airfoil stations modeled in the predictions, which could cause differences in loading noise. It is apparent from the experimental results that higher frequency broadband noise is dominant at lower RPMs and periodic thickness and loading noise becomes dominant as rotor RPM increases. Minimal differences in tonal noise between CAMRAD II and CHARM are seen. Furthermore, for all conditions, using best practices resulted in minimal differences in thickness noise between

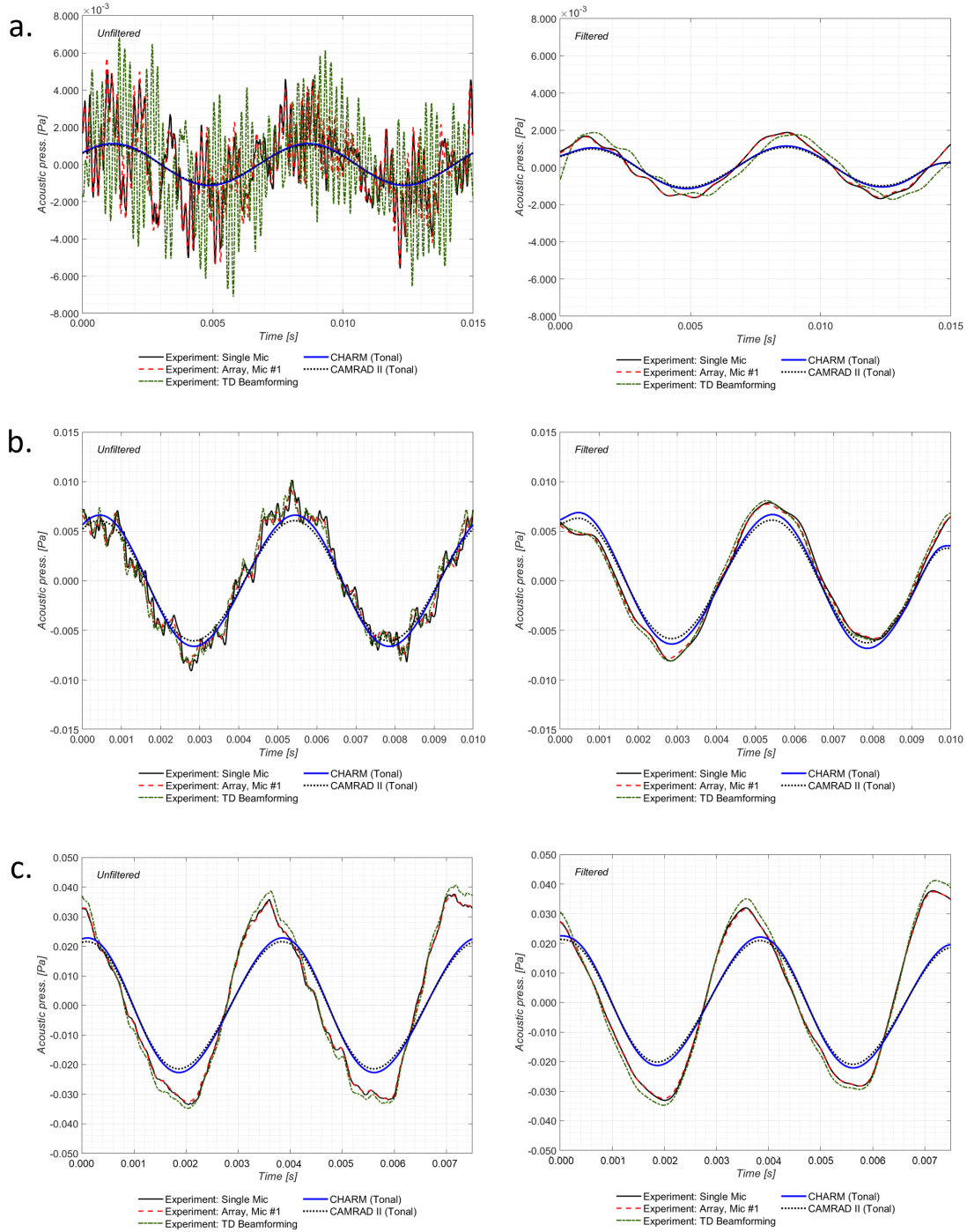


Figure 38.—Unfiltered and filtered experimental, CAMRAD II, and CHARM acoustic time histories for the DJI Phantom 3 isolated rotor for a) 4,000 RPM, b) 6,000 RPM, and c) 8,000 RPM.

CAMRAD II and CHARM, which highlights the ability to similarly model the blade geometry in each code. For all RPMs, CHARM reveals a slightly higher loading term, which could be due to the model dissimilarities seen in the z loading (Figure 37).

The final takeaway from the experimental comparison effort is that experimental results can often highlight missing information in the prediction models that should be included to improve accuracy. For this effort specifically, improving the model of the airfoils and including high frequency broadband noise would improve the results. But, as long as the modeled geometry sufficiently represents the physical geometry, comprehensive analysis with AARON can do an adequate job at predicting rotor discrete frequency noise.

8 Concluding remarks

In conclusion, best practices were developed for the use of CAMRAD II or CHARM with AARON in order to support the RVLT Conceptual Design Toolchain. Comprehensive analysis code inputs, acoustic code inputs, best practices, and debugging and visualization methodologies were all discussed. Additionally, acoustic results were provided for three vehicles: a simple rotor, the Quiet Single Main Rotor, and the Side-by-Side. Thickness, loading, broadband self-noise, and total noise results were presented for each vehicle. Additionally, an example of prediction/experiment correlation was shown for a test of a DJI Phantom 3 rotor in the ARC Anechoic Chamber. This document delivers best practices for a new user of the NASA RVLT Conceptual Design Toolchain for acoustics for multiple use cases, from a simple rotor to an experimental test.

References

1. Silva, C.; and Johnson, W.: Practical Conceptual Design of Quieter Urban VTOL Aircraft. *The Vertical Flight Society's 77th Annual Forum & Technology Display*, May 2021. Virtual.
2. Johnson, W.: CAMRAD II Comprehensive Analytical Model of Rotorcraft Aerodynamics and Dynamics. *Johnson Aeronautics*, 2005.
3. Wachspress, D.; Quackenbush, T.; and Boschitsch, A.: Rotorcraft Interactional Aerodynamics with Fast Vortex/Fast Panel Methods. *Journal of the American Helicopter Society*, vol. 48, no. 4, October 2003, pp. 223–235.
4. Lopes, L. V.; and Burley, C. L.: ANOPP2 User's Manual: Version 1.2. NASA TM 2016-219342, October 2016.
5. Kottapalli, S.; Silva, C.; and Boyd, D.: Effect of Rotor Blade Elasticity on UAM Quadrotor Acoustics. *Vertical Flight Society 79th Annual Forum & Technology Display*, May 2023. West Palm Beach, FL.
6. Kottapalli, S.; and Silva, C.: Prediction of Quadrotor Acoustics Using RVLTL Toolchain. *Vertical Flight Society Aeromechanics for Advanced Vertical Flight Technical Meeting*, Jan 2022. San Jose, CA.
7. Weist, L.; and Schatzman, N.: RABBIT: A Rapid Low Fidelity BVI Prediction Tool - Comparison and Validation using the NASA RVLTL Toolchain. *Vertical Flight Society Aeromechanics for Advanced Vertical Flight Technical Meeting*, Jan 2022. San Jose, CA.
8. Johnson, W.: NDARC - NASA Design and Analysis of Rotorcraft: Theory - Vol 1. NASA TP 20220000355, January 2022.
9. Johnson, W.: A History of Rotorcraft Comprehensive Analysis. NASA TP 216012, NASA, April 2012.
10. Farassat, F.: Derivation of Formulations 1 and 1A of Farassat. NASA TM 2007-214853, March 2007.
11. Rizzi, S. A.; Huff, D. L.; Boyd, D. D.; Bend, P.; Henderson, B. S.; Pascioni, K. A.; Sargent, D. C.; Josephson, D. L.; Marsan, M.; He, H.; and Snider, R.: Urban Air Mobility Noise: Current Practice, Gaps, and Recommendations. NASA TP 5007433, October 2020.
12. Silva, C.; Johnson, W. R.; Patterson, M. D.; and Antcliff, K. R.: VTOL Urban Air Mobility Concept Vehicles for Technology Development. *AIAA Aviation Technology, Integration, and Operations Conference*, June 2018. Atlanta, GA.
13. Silva, C.: Updates to Nasa Urban Air Mobility Reference Vehicles. *The Vertical Flight Society Electric VTOL Symposium*, February 2024. Santa Clara, CA.
14. Brooks, T. F.; Pope, D. S.; and Marcolini, M. A.: Airfoil Self-Noise and Prediction. NASA RP 1218, NASA, July 1989.

15. Meyn, L.: Rotorcraft Optimization Tools: Incorporating Rotorcraft Design Codes into Multi-Disciplinary Design, Analysis, and Optimization. *AHS Technical Meeting on Aeromechanics Design for Vertical Lift*, January 2018. San Francisco, CA.
16. Lopes, L. V.: Compact Assumption Applied to Monopole Term of Farassat's Formulations. *Journal of Aircraft*, vol. 54, no. 5, September-October 2017, p. 1649–1663.
17. Johnson, W.: A Quiet Helicopter for Air Taxi Operations. *Vertical Flight Society Aeromechanics for Advanced Vertical Flight Technical Meeting*, January 2020. San Jose, CA.
18. Kallstrom, K.: Exploring Airfoil Table Generation using XFOIL and OVERFLOW. *VFS Aeromechanics for Advanced Vertical Flight Technical Meeting*, January 2022. San Jose, CA.
19. Kallstrom, K.; and Shirazi, D.: Airfoil Table Generation and Validation for the VR-12 and SSC-A09 Airfoils and Quadrotor Performance Prediction. *The Vertical Flight Society's 6th Decennial Aeromechanics Specialists' Conference*, February 2024. Santa Clara, CA.
20. Johnson, W.: A General Free Wake Geometry Calculation for Wings and Rotors. *American Helicopter Society 51st Annual Forum*, May 1995. Fort Worth, TX.
21. Burnside, N. J.; and Horne, W. C.: Development of In-Flow Phased Microphone Array Windscreen Corrections for Small and Large Arrays. *AIAA Aviation and Aeronautics Forum and Exposition*, June 2023. San Diego, CA.
22. Russell, C.; Willink, G.; Theodore, C.; Jung, J.; and Glasner, B.: Wind Tunnel and Hover Performance Test Results for Multicopter UAS Vehicles. NASA TM 2018–219758, February 2018.
23. Weitsman, D.; Stephenson, J. H.; ; and Zawodny, N. S.: Effects of flow recirculation on acoustic and dynamic measurements of rotary-wing systems operating in closed anechoic chambers. *Journal of the Acoustic Society of America*, vol. 148, no. 3, 2020, p. 1325–1336.

Appendix A

Input files for each model

A.1 Simple rotor

The following files are the CAMRAD II and CHARM input files for the simple rotor. File names are relatively arbitrary as long as the changes are reflected in the run logs. Folders are in standard font and files are in bold. All files are included in this appendix as they were used.

For CAMRAD II, the input files are as follows and are placed in the run directory as organized here. One file to note, '**other-settings-for-noise.list**', contains the inputs specifically required to output data for AARON. This does not have to be its own file, and is later included in the .run or .list files. It is separated here to highlight it.

- CAMRAD_simple
 - camradii
 - * **rotorall.run**
 - camradii-inputs
 - * airfoil
 - **idealairfoil.run**
 - **ideal.txt**
 - * inputs
 - **rotor_input.run**
 - **rotor.list**
 - **other-settings-for-noise.list**

For CHARM, the input files required are as listed below. All files go in the same directory, so there is no file structure presented. The noise specific settings are included in **ideal.inp** from lines 42 to 49.

- **ideal.inp**
- **idealbg.inp**
- **idealbd.inp**
- **idealrw.inp**
- **0012.inp**

CAMRAD II run file for the simple rotor. Filename is 'rotorall.run.'

```

1  #!/bin/csh
2
3  set camrad='C:\Users\lweist\Documents\2024\CAMRAD II\camradii_v5.0_win-x64_dist\bin'
4
5  # Simple main rotor
6  setenv BLADEAIRFOIL1 ../camradii-inputs/airfoil/ideal.tab
7  setenv SHELLINPUT     ../camradii-inputs/inputs/rotor.bin
8  setenv PLOTFILE       ../camradii/lopesrotor.plot
9
10 $camrad > rotorall.out << 'endofinput'
11 &NLJOB PLFILE=1,&END
12 !=====
13 &NLDEF class='CASE',&END
14 &NLVAL CODE='LOW SPEED WAKE',&END ! description
15 &NLDEF class='TRIM',&END
16 &NLVAL
17     OPTRIM = 1,
18     WINDIN=0,WIND=110.1, ! operating condition
19     VTIPIN=1,VTIP=772.93,
20     LEVEL=3,NWPRNT=1,
21     MTRIM=1, ! wake loop
22     CTRIM=0.0009375, ! wind tunnel trim
23     MNAME='CT/S',
24     VNAME='COLL',
25     COLL=-1,
26     RELAXT=.25, ! trim loop
27     MHARMR=2*1,MHARMA=2*1, ! harmonics
28     RELAXC(2)=2*.05, ! convergence
29     pitch = 5.,
30 &END
31 &NLDEF class='TRIM ROTOR',name='ROTOR 1',&END
32 &NLVAL
33     OPMODE=0,DOFB=1,0,1,9*0, ! degrees of freedom
34     MASEN=1,MWSEN=1, ! output
35 &END
36 !=====
37 &NLDEF class='ROTOR',type='WAKE',name='ROTOR 1',&END
38 &NLVAL RFW=3.,MFWG=3,NPOFF=0,&END ! wake extent
39 !=====
40 &NLDEF class='AIRFRAME',type='STRUCTURE',&END
41 &NLVAL
42     TITLE='WIND TUNNEL SUPPORT', ! description
43     CONFIG=0,OPFREE=0,OPAERO=0, ! wind tunnel
44     ASHAFT=0.,ACANT=0., ! geometry
45 &END
46 !=====
47 &NLDEF class='ROTOR',type='STRUCTURE',name='ROTOR 1',&END
48 &NLVAL OPBEAM=0,KNODE=0,LOCKPL=0, ! rigid blade
49     TWISTL=0.,RADIUS=32.808.,
50     SIGMA=.064,
51 &END
52 !=====
53 &NLDEF class='ROTOR',type='AERODYNAMICS',name='ROTOR 1',&END
54 &NLVAL
55     ASWEEP=4*0.,CHORD=4*1.64042,! ! 0% taper
56 &END
57 !=====
58 !%%%%%%%%%%
59 ! reduced output
60 &NLDEF class='TRIM',&END
61 &NLVAL NWPRNT=0,&END ! wake loop
62 !%%%%%%%%%%
63 !=====
64 &NLDEF action='end of shell',&END
65 &NLDEF action='end of core',&END
66 'endofinput'

```

CAMRAD II input run file for the simple rotor. Filename is 'rotor_input.run.'

```
1  #!/bin/csh
2
3  set camrad='C:\Users\lweist\Documents\2024\CAMRAD II\camradii_v5.0_win-x64_dist\bin'
4
5  rm -f rotor.bin
6  rm -f rotor_input.out
7
8  #
9  setenv INPUTLIST1 rotor.list
10 setenv INPUTLIST2 other-settings-for-noise.list
11 setenv OUTPUTFILE rotor.bin
12
13 $camrad > rotor_input.out << 'endofinput'
14 BATCH
15 &NLJOB NLISTI=2,NLISTC=0,&END
16 'endofinput'
17
18 exit
19
```

CAMRAD II input list file for the simple rotor. Filename is 'rotor.list.'

```

1  ! Rotor in Wind Tunnel
2  !=====
3  &NLDEF class='CASE',&END
4  &NLVAL
5      TITLE='SAMPLE JOBS:  ROTOR IN WIND TUNNEL',           ! description
6      OPUNIT=1,OPDENS=3,DENSE=.00237689,TEMP=59.,          ! environment
7  &END
8  !=====
9  &NLDEF class='TRIM',&END
10 &NLVAL
11     VELIN=2,WINDIN=2,VTIPIN=1,VTIP=650.,                 ! operating condition
12     LEVEL=1,                                              ! wake loop
13     COLL=10.,CTTRIM=.08,MTRIM=3,                          ! wind tunnel trim
14     MNAME='CT/S      ','BETAS      ','BETAC      ',
15     VNAME='COLL      ','LATCYC      ','LNGCYC      ',
16     MHARMR=10,MHARMA=10,                                  ! harmonics
17     DOFA=6*0,DOFM=5*0,                                    ! degrees of freedom
18 &END
19 &NLDEF class='TRIM ROTOR',name='ROTOR 1',&END
20 &NLVAL OPMODE=0,DOFB=12*1,&END                             ! degrees of freedom
21 !=====
22 &NLDEF class='FLUTTER',&END
23 &NLVAL
24     DOFA=6*0,DOFM=5*1,DOFD=8*0,
25 &END                                     ! degrees of freedom
26 &NLDEF class='FLUTTER ROTOR',name='ROTOR 1',&END
27 &NLVAL
28     OPWAKE=4,OPVATR=2,OPVRTA=2,
29     OPMODE=1,DOFM=2*1,38*0,DOFL=3*1,NMPRNT=1,
30 &END
31 !=====
32 &NLDEF class='AIRFRAME',type='STRUCTURE',&END
33 &NLVAL
34     TITLE='SAMPLE JOBS:  WIND TUNNEL SUPPORT'             ! description
35     CONFIG=0,OPFREE=0,OPAERO=0,                            ! wind tunnel
36     MASSR=9.6,                                              ! inertia
37     HSP=2.,OPSPM=0,                                         ! control system
38     NLOC=6,                                                  ! locations
39     FSLOC(5)=5.,-5.,WLLOC(5)=-7.,-7.,                      ! 5=center,6=nose
40     NMODE=5,                                                 ! test module
41     QMASS=5*100.,QFREQ=2.1,2.6,20.,22.,23.,QDAMP=2*.02,3*.2,
42     LSHAPE(1,1,1) = 1., 0., 0.,
43     LSHAPE(1,1,2) = 0., 1., 0.,
44     LSHAPE(1,1,3) = 1., 0., 0.,
45     LSHAPE(1,1,4) = 0., 1., 0.,
46     LSHAPE(1,1,5) = 0., 0., 1.,
47     ASHAPE(1,1,1) = 0., -1.5, 0.,
48     ASHAPE(1,1,2) = 1.5, 0., 0.,
49     ASHAPE(1,1,3) = 0., 0., 0.,
50     ASHAPE(1,1,4) = 0., 0., 0.,
51     ASHAPE(1,1,5) = 0., 0., 0.,
52 &END
53 &NLDEF class='AIRFRAME',type='AERODYNAMICS',&END
54 &NLVAL
55     VISRC=0,                                                ! test module
56     NBODY=1,LOCBC=5,LOCBN=6,
57     LENGTH=20.,THICK=.5,SHAPE=3,
58 &END
59 &NLDEF class='AIRFRAME',type='CONTROL',&END
60 &NLVAL &END
61 &NLDEF class='AIRFRAME',type='DRIVE TRAIN',&END
62 &NLVAL
63     CONFIG=0,OPGOV=0,
64     GEAR=7.,KRS=10000.,IRS=.1,
65     KES=10000.,IENG=50.,DENG=1.,
66 &END

```

CAMRAD II input list file for the simple rotor. Filename is 'rotor.list.'

```

67  !=====
68  &NLDEF class='ROTOR',type='STRUCTURE',name='ROTOR 1',&END
69  &NLVAL                                     ! description
70  TITLE='SAMPLE JOBS: HELICOPTER MAIN ROTOR, SWEPT/TAPERED TIP',
71  RADIUS=32.808,NBLADE=4,ROTATE=1,SIGMA=.075,
72  VTIPN=650.,
73  HINGE=6,EFLAP=.04,EPITCH=.065,           ! blade root
74  KFLAP=1000.,KLAG=1000.,DLAG=2000.,
75  CONTRL=2,PITCH=1,KPITCH=0.,LOCKP=1,      ! control system
76  XSP=-.023,YSP=.045,ZSP=-.1,
77  XPH=-.023,YPH=.045,ZPH=0.,
78  EPH=.07,KPL=80000.,LOCKPL=1,LOCKSP=0,
79  OPBEAM=2,DRELST=.05,KNODE=3,RNODE=.94,.35,.64, ! elastic blade
80  NSEN=3,QUANT=3*4,RLOAD=.4,.6,.8,         ! loads
81  TWIN=1,TWISTL=0.,                        ! linear twist
82  ETPP=.94,                                ! tpp sensor
83  NPROP=3,RPROP=0.,.94,1.,                 ! section properties
84  ZEA=3*0.,ZQC=3*0.,ZC=3*0.,ZI=3*0.,
85  XEA=3*0.,XQC=3*0.,XC=3*0.,XI=3*0.,
86  XEA=2*0.,.03464,                          ! 30 deg sweep
87  XQC=2*0.,.03464,
88  KP=3*.011,KT=3*.011,EA=3*20000000.,
89  EIFLAP=3*68000.,EILAG=3*800000.,GJ=3*90000.,
90  MASS=3*.12,ITHETA=3*.0056,IPOLAR=3*.0056,
91  &END
92  !=====
93  &NLDEF class='ROTOR',type='AERODYNAMICS',name='ROTOR 1',&END
94  &NLVAL
95  NPANEL=20,                                ! aero panels
96  REDGE=.12,.20,.28,.35,.42,.48,.54,.59,.64,.69,.73,
97  .77,.81,.84,.87,.90,.92,.94,.96,.98,1.0,
98  NPROP=4,RPROP=0.,.94,.9401,1.,           ! aero properties
99  CHORD=4*1.64042,ASWEEP=4*0.,             ! 0% taper
100 NSEN=6,OPREF=6*4,                        ! aerodynamic sensors
101 QUANT= 5,6,25,35,82,82,                  ! lambda,lambda-int
102 IDENT= 1,1, 0, 0, 0, 0,                 ! alpha,theta
103 AXIS= 3,3, 0, 0, 1, 3,                   ! Fx,Fz
104 OPSCL= 2,2, 1, 1, 2, 2,                  !
105 NAPLOT=1,1, 4, 1, 0, 0,                  !
106 &END
107 !=====
108 &NLDEF class='ROTOR',type='INFLOW',name='ROTOR 1',&END
109 &NLVAL &END
110 &NLDEF class='ROTOR',type='WAKE',name='ROTOR 1',&END
111 &NLVAL
112 OPSCEN=1,                                ! forward flight wake
113 TWIST=0.,RICWG=.15,
114 &END
115 &NLDEF class='ROTOR',type='WAKE',name='ROTOR 1',&END
116 &NLVAL OPSCEN=0,&END
117 !=====
118 &NLDEF class='TABLES',&END
119 &NLVAL &END
120 !=====
121 &NLDEF action='end of shell',&END
122 !=====
123 !=====
124 &NLDEF action='end of core',&END
125

```

CAMRAD II noise settings for the simple rotor. Filename is 'other-settings-for-noise.list.'

```

1  !=====
2  ! other settings for noise
3  !=====
4  &NLDEF class='TRIM', &END
5  &NLVAL
6      MPSIH = 360,
7      OPPOST = 1,
8      ITERF = 3,
9  &END
10 !=====
11 ! ROTOR 1
12 !=====
13 &NLDEF class='TRIM ROTOR', name='ROTOR 1', &END
14 &NLVAL
15     MHSEN = 1, NHFILE = 1, MHTIME = 360,
16     MCSSEN = 1, NCFILE = 1, MCTIME = 360,
17     MBSSEN = 1, NBFILE = 1, MBTIME = 360,
18     MPSEN = 1, NPFILE = 1, MPTIME = 360,
19     MASEN = 1, NAFILE = 1, MATIME = 360,
20 &END
21 !-----
22 &NLDEF class='ROTOR', type='STRUCTURE', name='ROTOR 1', &END
23 &NLVAL
24     OPSND = 1,
25 &END
26 !-----
27 &NLDEF class='ROTOR', type='AERODYNAMICS', name='ROTOR 1', &END
28 &NLVAL
29     NSEN = 7,
30     QUANT = 5, 5, 5, 53, 81, 81, 64,
31     IDENT = 0, 0, 0, 1, 1, 1, 1,
32     AXIS = 1, 2, 3, 0, 1, 3, 0,
33     OPSCL = 0, 0, 0, 1, 2, 2, 2,
34     OPREF = 1, 1, 1, 1, 2, 2, 2,
35 &END
36 !-----
37 &NLDEF class='ROTOR', type='WAKE', name='ROTOR 1', &END
38 &NLVAL
39     OPFWG = 3,
40     CORE(1) = 0.8,
41     OPTVIC = 0,
42     OPRUDP = 0,
43 &END
44 !=====
45 &NLDEF action='end of shell', &END
46 &NLDEF action='end of core', &END
47
48
49
50
51
52
53
54
55
56
57

```

CAMRAD II airfoil run file for the simple rotor. Filename is 'idealairfoil.run.'

```
1  #!/bin/csh
2
3  set camrad='C:\Users\lweist\Documents\2024\CAMRAD II\camradii_v5.0_win-x64_dist\bin'
4  rm -f ideal.out
5  rm -f ideal.tab
6  setenv INPUTDECK1 ideal.txt
7  setenv OUTPUTTABLE ideal.tab
8  $camrad > ideal.out << 'endofinput'
9  BATCH
10 &NLJOB
11     OPFILE = 7,
12     OPSRC  = 1,
13 &END
14 &NLTABL
15     OPFORM = 1,
16     RNTRP  = 0,
17     TITLE  = 'ideal airfoil',
18 &END
19 'endofinput'
20
```

CAMRAD II ideal airfoil deck for the simple rotor. Filename is 'ideal.txt.'

```

1  IDEAL AIRFOIL (NO STALL, CLA=.1/DEG)
2  IDEAL AIRFOIL (NO STALL)
3  MACH      ALPHA
4  2          7      CL
5          -180.    -90.001    -90.      0.    90.      90.001    180.
6  0.0          0.      9.00000    -9.00000    0.      9.00000    -9.00000    0.
7  1.0          0.      9.00000    -9.00000    0.      9.00000    -9.00000    0.
8  2          2      CM
9          -180.      180.
10 0.0          0.      0.
11 1.0          0.      0.
12 2          2      CD
13          -180.      180.
14 0.0          .0080    .0080
15 1.0          .0080    .0080
16

```

CHARM input file for the simple rotor. Filename is 'ideal.inp.'

```

1  ideal.inp: run characteristics for Lopes rotor
2  NROTOR
3  1
4  PATHNAME
5  ../
6  INPUT FILENAMES
7  idealrw.inp
8  idealbg.inp
9  idealbd.inp
10 0012.inp
11 none
12 SSPD      RHO
13 1116.43701 0.0023768924
14 SFRAME
15 1
16 UCG      VCG      WCG      PCG      QCG      RCG
17 110.0    0.0      0.0      0.0      0.0      0.0
18 NPSI     NREV     CONVG+  CONVG2  CONVG3  MREV
19 36       200      -1.0    -1.0    -1.0    0
20 IRST     IFREE    IGPR
21 0         0        0
22 IVOUT     NRS      ROUT
23 4         20      0.1      0.2      0.25     0.3      0.35     0.4      0.45
24 0.5
25 0.55     0.6      0.65     0.7      0.75     0.8      0.85     0.9      0.92
26 0.94
27 0.96     0.98
28 NPRINT   IBLPLT   IFILPLT
29 0         0        3         3         3         3
30 KOPT
31 0
32 ISCAN
33 1
34 ISTRSS
35 0
36 IFV      IQUIK1
37 2        1
38 IFVFW    IFVBL    IFVSU    IFVSC    IFVLS
39 1         1        1         1         1
40 ISURF
41 0
42 ISHIP
43 0
44 IRECON    NOISE    KBROADBAND KSEL
45 1         -4       1         1
46 ACOUSTICS_CODE
47 2
48 NCHDRC    NPSIRC    IVROLL    CDRMIN    IRECONTYPE
49 1         360      1         0.02     1
50 NLS
51 0

```

CHARM blade geometry file for the simple rotor. Filename is 'idealbg.inp.'

```
1  idealbg.inp: blade geometry input file for lopes rotor
2  NSEG
3    20
4  CUTOUT
5    10.8268
6  SL(ISEG)
7    20*1.0991
8  CHORD(ISEG)
9    21*1.640
10 ELOFSG(ISEG) - (elastic axis offset)
11    21*0.000
12 SWEEP(ISEG)
13    20*0.000
14 TWRD (Blade root twist at zero collective in degrees)
15    0.000
16 TWSTGD(ISEG)
17    20*0.000
18 ANHD(ISEG)
19    20*0.000
20 THIKND(ISEG)
21    20*0.1200
22 KFLAP(ISEG)
23    20*0
24 FLAPND(ISEG)
25    20*0.0000
26 FLHNGE(ISEG)
27    20*0.0000
28 FLDEFL(ISEG)
29    20*0.000
30 NCAM
31    0
32 NCHORD  NSPAN  ICOS
33    1      100   -1
34
```

CHARM blade dynamics file for the simple rotor. Filename is 'idealbd.inp'

```

1  idealbd.inp: blade dynamics input file for lopes rotor, hingeless
2  ISTRM
3  0
4  ISTRIP  IFPC  IAERO
5  -1      0      3
6  ICOMP   IRVFLO  ISTFLO
7  1       0       1
8  IART    HINGE   PRECONE
9  0       0.000   0.00
10 NMODE
11 1
12 NMDFLP  NMDTOR  NMDLAG  NMDELG
13 1       0       0       0
14 IFXMDE  IRRIGID
15 0       1
16 FREQMD  GMASS      for Mode 1
17 1.039   0.2308
18 NRBARS
19 50
20 RBARS(IR), IR=1,NRBARS
21 0.000 0.020 0.041 0.061 0.082 0.102 0.122 0.143 0.163 0.184
22 0.204 0.224 0.245 0.265 0.286 0.306 0.327 0.347 0.367 0.388
23 0.408 0.429 0.449 0.469 0.490 0.510 0.531 0.551 0.571 0.592
24 0.612 0.633 0.653 0.673 0.694 0.714 0.735 0.755 0.776 0.796
25 0.816 0.837 0.857 0.878 0.898 0.918 0.939 0.959 0.980 1.000
26 Rigid flap mode
27 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
28 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
29 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
30 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
31 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
32 0.608 0.689 0.769 0.850 0.930 1.011 1.092 1.172 1.253 1.333
33 1.414 1.494 1.575 1.655 1.736 1.817 1.897 1.978 2.058 2.139
34 2.219 2.300 2.381 2.461 2.542 2.622 2.703 2.783 2.864 2.944
35 3.025 3.106 3.186 3.267 3.347 3.428 3.508 3.589 3.669 3.750
36 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
37 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
38 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
39 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
40 1.053 1.053 1.053 1.053 1.053 1.053 1.053 1.053 1.053 1.053
41 1.053 1.053 1.053 1.053 1.053 1.053 1.053 1.053 1.053 1.053
42 1.053 1.053 1.053 1.053 1.053 1.053 1.053 1.053 1.053 1.053
43 1.053 1.053 1.053 1.053 1.053 1.053 1.053 1.053 1.053 1.053
44 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
45 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
46 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
47 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
48 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
49 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
50 First elastic bending mode
51 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
52 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
53 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
54 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
55 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
56 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
57 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
58 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
59 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
60 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000

```

CHARM blade dynamics file for the simple rotor. Filename is 'idealbd.inp.'

49	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
50	0.000	0.000	0.000	-0.131	-0.358	-0.571	-0.771	-0.956	-1.128	-1.286
	-1.430	-1.560	-1.677	-1.779	-1.868	-1.943	-2.004	-2.051	-2.084	-2.104
51	-2.109	-2.101	-2.079	-2.043	-1.993	-1.930	-1.852	-1.761	-1.656	-1.537
	-1.404	-1.257	-1.097	-0.923	-0.734	-0.532	-0.316	-0.087	0.157	0.414
52	0.686	0.971	1.270	1.582	1.909	2.250	2.604	2.972	3.354	3.750
53	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
54	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
55	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
56	-3.158	-3.158	-3.158	-3.058	-2.877	-2.697	-2.516	-2.335	-2.154	-1.973
	-1.792	-1.611	-1.430	-1.249	-1.068	-0.888	-0.707	-0.526	-0.345	-0.164
57	0.017	0.198	0.379	0.560	0.741	0.921	1.102	1.283	1.464	1.645
	1.826	2.007	2.188	2.369	2.550	2.731	2.911	3.092	3.273	3.454
58	3.635	3.816	3.997	4.178	4.359	4.540	4.720	4.901	5.082	5.263
59	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
60	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
61	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
62										

CHARM rotor wake file for the simple rotor. Filename is 'idealrw.inp.'

```

1  idealrw.inp: rotor wake input file  for lopes rotor
2  NBLADE  OMEGA
3    4      23.5585385
4  IROTAT      XROTOR      X,Y,Z tilt      ITILT
5    1      0.0  0.0  0.0      0.0  5.0  0.0      0
6  ICOLL      COLLD      CT
7    1      0.000      0.00006345
8  ITRIM      A1W      B1W      A1S      B1S
9    0      0      0.0      -.14      0.09
10 NOWAKE      ICNVCT      NWAKES      NPWAKE      IFAR      MBCVE
11    0      0      1      0      0      0
12 KSCHEME      KPC
13    0      0
14 NCUT      AOV LAP      ISKEW      IUNS
15    1      -1.000      1      1
16 NZONE      (NVORT(I), I=1,NZONE)
17    3      14      6      2
18 (NPFTW(I), I=1,NZONE)
19 120 120 120
20 CORLIM Zone 1 Min core radii
21 0.01
22 CORLIM Zone 1 Max core radii
23 1.0
24 CORLIM Zone 2 Min core radii
25 0.01
26 CORLIM Zone 2 Max core radii
27 1.0
28 CORLIM Zone 3 Min core radii
29 0.1
30 CORLIM Zone 3 Max core radii
31 0.1
32 CUTLIM Zone 1 Min cutoff distance
33 0.01
34 CUTLIM Zone 1 Max cutoff distance
35 1.0
36 CUTLIM Zone 2 Min cutoff distance
37 0.01
38 CUTLIM Zone 2 Max cutoff distance
39 1.0
40 CUTLIM Zone 3 Min cutoff distance
41 0.1
42 CUTLIM Zone 3 Max cutoff distance
43 0.1
44 IDYNM
45 1
46 SRAD      SHGHT
47 0.0      0.0
48 NHHI (Higher harmonic cyclic pitch input flag)
49 0
50

```

CHARM airfoil file for the simple rotor. Filename is '0012.inp.'

```

1  ideal.inp: run characteristics for Lopes rotor
2  NROTOR
3  1
4  PATHNAME
5  ../
6  INPUT FILENAMES
7  idealrw.inp
8  idealbg.inp
9  idealbd.inp
10 0012.inp
11 none
12 SSPD      RHO
13 1116.43701 0.0023768924
14 SFRAME
15 1
16 UCG      VCG      WCG      PCG      QCG      RCG
17 110.0    0.0      0.0      0.0      0.0      0.0
18 NPSI     NREV     CONVG+   CONVG2   CONVG3   MREV
19 36       200      -1.0    -1.0    -1.0     0
20 IRST     IFREE    IGPR
21 0        0        0
22 IVOUT    NRS      ROUT
23 4        20      0.1      0.2      0.25     0.3      0.35     0.4      0.45
24 0.5      0.6      0.65     0.7      0.75     0.8      0.85     0.9      0.92
25 0.94     0.96     0.98
26 NPRINT   IBLPLT   IFILPLT
27 0        0        3        3        3        3
28 KOPT
29 0
30 ISCAN
31 1
32 ISTRSS
33 0
34 IFV      IQUIK1
35 2        1
36 IFVFW    IFVBL    IFVSU    IFVSC    IFVLS
37 1        1        1        1        1
38 ISURF
39 0
40 ISHIP
41 0
42 IRECON    NOISE    KBROADBAND KSEL
43 1         -4       1         1
44 ACOUSTICS_CODE
45 2
46 NCHDRC    NPSIRC    IVROLL   CDRMIN   IRECONTYPE
47 1         360      1        0.02     1
48 NLS
49 0

```

A.2 QSMR

The following files are the CAMRAD II and CHARM input files for the QSMR. As the QSMR has more complicated inputs than the previous vehicle model, these inputs provide an example of how extensive commenting can help with understanding the input files. All variable definitions are from the CAMRAD II user's manual, and are simply included to help the user. Additionally, airfoil tables for the SSC-A09 and VR-12 airfoils are omitted as they are proprietary and have yet to be published.

- CAMRAD_QSMR
 - camradii
 - * **qsmr700.run**
 - camradii-inputs
 - * airfoil
 - **helicopter_airfoil.run**
 - Airfoil tables omitted, but go here
 - * inputs
 - **qsmr_input.run**
 - **qsmr.list**

For CHARM, the input files required are as listed below. Unlike the simple rotor case, the QSMR in CHARM is modeled as a main rotor- tail rotor configuration, for control purposes, and also has a fuselage. This leads to both 'mr' for main rotor inputs and 'tr' for tail rotor inputs. The trim file, '**standardtrim.inp**,' is the trim control matrix for a main rotor- tail rotor configuration, and '**fusedata_table.inp**' is the CHARM example fuselage file. Airfoil tables are once again omitted as they have yet to be officially published.

- **qsmr.inp**
- **qsmrmr_bg.inp**
- **qsmrmr_bd.inp**
- **qsmrmr_rw.inp**
- **qsmrtr_bg.inp**
- **qsmrtr_bd.inp**
- **qsmrtr_rw.inp**
- **standardtrim.inp**
- **fusedata_table.inp**

CAMRAD II run file for the QSMR. Filename is 'qsmr700.run.'

```
1  #!/bin/csh
2
3  set camrad='C:\Users\lweist\Documents\2024\CAMRAD II\camradii_v5.0_win-x64_dist\bin'
4  # Quiet Single Main Rotor Helicopter
5
6  setenv BLADEAIRFOIL1 ../camradii-inputs/airfoil/helicopter.tab
7  setenv BLADEAIRFOIL2 ../camradii-inputs/airfoil/ideal.astd.txt.tab
8  setenv SHELLINPUT    ../camradii-inputs/inputs/qsmr.bin
9  setenv PLOTFILE      ../camradii/qsmr.plot
10
11 $camrad > qsmr700.out << 'endofinput'
12 &NLJOB
13     NCASES = 1,
14     OUTRES = 1,
15     PLFILE = 0,
16 &END
17 !=====
18 &NLDEF class='CASE',&END
19 &NLVAL
20     TMTASK = 1, !trim task
21     FLTASK = 1, !flutter task
22     OPDENS = 1,
23     ALTMSL = 0.0, !altitude above sea level
24 &END
25 !=====
26 &NLDEF class='TRIM',&END
27 &NLVAL
28     VELIN = 0,
29     SPEED = 177.22,
30     CLIMB = 0.0,
31
32     LEVEL = 3, 1, !rotor wake analysis level
33     ITERF = 3, ! number of iterations, nonuniform inflow with free wake geometry
34
35     OPTRIM = 1, ! Conduct trim
36     OPPOST = 1, ! Conduct post trim
37     MPSI = 24,
38     MPSIH = 240,
39
40     COLL = 4.93045, !Exact trim values
41     LATCYC = -1.56454,
42     LNGCYC = 3.46010,
43     PEDAL = -26.9856,
44     PITCH = 1.45399,
45     ROLL = -0.659481,
46
47     TOLERC = 2.0, !Circulation loop tolerance
48 &END
49 !=====
50 &NLDEF class='AIRFRAME',type='STRUCTURE',&END
51 &NLVAL
52     OPAERO = 0, !Turn off airframe aerodynamics
53 &END
54 !=====
55 &NLDEF class='TRIM ROTOR',name='ROTOR 1',&END
56 &NLVAL
57     MPSEN = 1, MPTIME = 240,
58     MASEN = 1, MATIME = 240,
59 &END
60 !=====
61 &NLDEF class='TRIM ROTOR',name='ROTOR 2',&END
62 &NLVAL
63     MPSEN = 1, MPTIME = 240, !Output blade position sensors for TR
64     MASEN = 1, MATIME = 240, !Output blade aerodynamic sensors for TR
65 &END
66 !=====
```

CAMRAD II run file for the QSMR. Filename is 'qsmr700.run.'

```

67  &NLDEF class='ROTOR',type='AERODYNAMICS',name='ROTOR 1',&END !All these are to output
    acoustics
68  &NLVAL
69      NSEN      = 7,
70      QUANT     = 5, 5, 5, 53, 81, 81, 64,
71      IDENT     = 0, 0, 0, 1, 1, 1, 1,
72      AXIS      = 1, 2, 3, 0, 1, 3, 0,
73      OPSCL     = 0, 0, 0, 1, 2, 2, 2,
74      OPREF     = 1, 1, 1, 1, 2, 2, 2,
75  &END
76  &NLDEF class='ROTOR',type='AERODYNAMICS',name='ROTOR 2',&END
77  &NLVAL
78      NSEN      = 7,
79      QUANT     = 5, 5, 5, 53, 81, 81, 64,
80      IDENT     = 0, 0, 0, 1, 1, 1, 1,
81      AXIS      = 1, 2, 3, 0, 1, 3, 0,
82      OPSCL     = 0, 0, 0, 1, 2, 2, 2,
83      OPREF     = 1, 1, 1, 1, 2, 2, 2,
84  &END
85  !=====
86  &NLDEF class='ROTOR',type='INFLOW',name='ROTOR 1',&END
87  &NLVAL
88  &END
89  !=====
90  &NLDEF class='ROTOR',type='STRUCTURE',name='ROTOR 1',&END
91  &NLVAL
92      OPSND     = 1,
93  &END
94  &NLDEF class='ROTOR',type='STRUCTURE',name='ROTOR 2',&END
95  &NLVAL
96      OPSND     = 1,
97      NBLADE    = 12,
98      SIGMA     = 0.50303,
99  &END
100 !=====
101 &NLDEF class='ROTOR',type='AERODYNAMICS',name='ROTOR 2',&END
102 &NLVAL
103     CHORD      = 2*0.084283,
104 &END
105 !=====
106 &NLDEF class='ROTOR',type='WAKE',name='ROTOR 1',&END
107 &NLVAL
108     OPSCEN = 1, !Turn on elementary scenario forward flight - gives a good starting
        position
109 &END
110 &NLDEF class='ROTOR',type='WAKE',name='ROTOR 1',&END
111 &NLVAL
112     OPSCEN = 0, !turn scenario back off (this allows you to change things after the
        initialization previously)
113     CORE(1) = 0.8, !vortex core, tip vortices (collocation points on wing set)
114     COREWG = 0.8, !tip vortex core radius (fraction mean chord)
115     OPTVIC = 0, !do not suppress wing-vortex interaction by using larger core radius
116     OPFWG = 3, !free geometry model, general method (separate rotors)
117 &END
118 !=====
119 &NLDEF action='end of shell',&END
120 &NLDEF action='end of core',&END
121 'endofinput'
122 exit
123

```

CAMRAD II input run file for the QSMR. Filename is 'qsmr_input.run.'

```
1  #!/bin/csh
2  set camrad='C:\Users\lweist\Documents\2024\CAMRAD II\camradii_v5.0_win-x64_dist\bin'
3  #
4  setenv INPUTLIST1 qsmr.list
5  setenv OUTPUTFILE qsmr.bin
6
7  ${camrad} > qsmr_input.out << 'endofinput'
8  BATCH
9  &NLJOB
10         NLISTI = 1,
11         NLISTC = 0,
12  &END
13  'endofinput'
14  exit
15
```

CAMRAD II input list file for the QSMR. Filename is 'qsmr.list.'

```

1  !=====
2  !   Quite Single Main Rotor Helicopter
3  !=====
4  &NLDEF class='CASE',&END
5  &NLVAL
6      TITLE      = 'QUIET SINGLE MAIN ROTOR HELICOPTER',
7      OPUNIT     = 1,
8      DENSE      = 0.002297,
9      TEMP       = 59.0,
10     CSOUND     = 1136.,
11 &END
12 !=====
13 &NLDEF class='TRIM',&END
14 &NLVAL
15     VELIN      = 1,
16     WINDIN     = 1,
17     VTIPIN    = 1,
18     VTIP       = 700.,
19     LEVEL      = 2*1,
20     COLL       = 10.,
21     MTRIM      = 6,
22     MNAME      = 'FORCE X ','FORCE Y ','FORCE Z ','MOMENT X ','MOMENT Y ','MOMENT Z ',
23     VNAME      = 'COLL ','LATCYC ','LNGCYC ','PEDAL ','PITCH ','ROLL ',
24
25     MHARMR     = 2*1, !rotor part solution, number of harmonics
26     MHARMA     = 2*1, !airframe part solution, number of harmonics
27     DOFA       = 6*0, !vectors defining degrees of freedom in trim solution, zero
28
29     OPPID      = 2, !perturbation identification of derivative matrix, second order
        difference
30     TOLERT     = 0.1, !convergence control, tolerance
31     TOLERC     = 0.1, RELAXC = 0.1, ITERC = 200, !circulation loop (successive
        substitution) convergence control
32     TOLERR     = 2*0.2, RELAXR = 2*0.5, MREVR = 2*200, !rotor part solution, convergence
        control
33 &END
34 !=====
35 &NLDEF class='TRIM ROTOR',name='ROTOR 1',&END
36 &NLVAL
37     OPMODE     = 0,
38     DOFB       = 1,0,1,9*0, !Blade degrees of freedom - need to investigate further
39 &END
40 &NLDEF class='TRIM ROTOR',name='ROTOR 2',&END
41 &NLVAL
42     OPMODE     = 0,
43     DOFB       = 0,0,1,9*0,
44 &END
45 !=====
46 &NLDEF class='FLUTTER',&END
47 &NLVAL
48     OPBLD      = 1, !task definition, independent blade analysis
49 &END
50 &NLDEF class='FLUTTER ROTOR', name='ROTOR 1',&END
51 &NLVAL
52     OPMODE     = 1,
53     DOFB       = 1,0,1,9*0,
54 &END
55 &NLDEF class='FLUTTER ROTOR', name='ROTOR 2',&END
56 &NLVAL
57     OPMODE     = 1,
58     DOFB       = 1,0,1,9*0,
59 &END
60 !=====
61 &NLDEF class='AIRFRAME',type='STRUCTURE',&END
62 &NLVAL
63     TITLE      = 'HELICOPTER AIRFRAME',

```

CAMRAD II input list file for the QSMR. Filename is 'qsmr.list.'

```

64
65     CONFIG      = 1,
66     RGEAR(2)    = 18.4088, !transmission gear ratio (ratio rotor#n rotational speed to
                          parent rotational speed)
67
68     WEIGHT      = 3428., !gross weight (lb or kg)
69     IXX          = 300., !moment of inertia Ixx
70     IYY          = 2600., !moment of inertia Iyy
71     IZZ          = 2600., !moment of inertia Izz
72
73     FSCG        = 0.00, !fuselage station center of gravity
74     WLCG        = -3.30, !buttline center of gravity
75     BLCG        = 0.00, !waterline center of gravity
76
77     FSRTR       = 0.00, 19.82, !fuselage station rotor hub
78     BLRTR       = 0.00, 0.00, !buttline rotor hub
79     WLRTR       = 0.00, 0.66, !waterline rotor hub
80
81     ASHAFT      = -3.00, 0.00, !shaft angle of attack  $\theta_R$  (deg)
82
83     FSWB        = 0.00, !fuselage station wing-body center of action
84     BLWB        = 0.00, !buttline wing-body center of action
85     WLWB        = -4.13, !waterline wing-body center of action
86
87     FSHT        = 19.82, !fuselage station horizontal tail center of action
88     BLHT        = 0.00, !buttline horizontal tail center of action
89     WLHT        = 0.66, !waterline horizontal tail center of action
90
91     FSVT        = 18.17, !fuselage station vertical tail center of action
92     BLVT        = 0.00, !buttline vertical tail center of action
93     WLVT        = -4.13, !waterline vertical tail center of action
94     &END
95     !=====
96     &NLDEF class='AIRFRAME', type='AERODYNAMICS', &END
97     &NLVAL
98         DRGOW    = 5.15, !minimum drag d/q
99         DRGVW    = 5.15, !minimum drag d/q
100        DRGIW    = 0.00, !minimum drag d/q
101
102        LFTAH    = 44.79, !Horizontal tail lift derivative with angle of attack  $l\alpha/q$ 
103        LFTAV    = 29.21, !Vertical tail lift derivative with angle of attack  $l\alpha/q$ 
104        IHTL     = 0.00, !Horizontal tail incidence angle for lift
105        IVTL     = -9.00, !Vertical tail incidence angle for lift
106
107        AMAXW    = 20.00,
108        AMAXH    = 30.00, ! Horizontal tail stall angle of attack
109        AMAXV    = 30.00, ! Vertical tail stall angle of attack
110    &END
111    !=====
112    &NLDEF class='AIRFRAME', type='CONTROL', &END
113    &NLVAL &END
114    !=====
115    ! main rotor
116    !=====
117    &NLDEF class='AIRFRAME', type='STRUCTURE', &END
118    &NLVAL
119        MASSR(1) = 4.0946, !equivalent hub inertia mass
120        HSP(1)   = 1.652, !distance swashplate below hub
121    &END
122    !=====
123    &NLDEF class='ROTOR', type='STRUCTURE', name='ROTOR 1', &END
124    &NLVAL
125        TITLE    = 'HELICOPTER MAIN ROTOR',
126        RADIUS    = 16.52,
127        NBLADE    = 3,
128        ROTATE    = 1,

```

CAMRAD II input list file for the QSMR. Filename is 'qsmr.list.'

```

129      SIGMA      = 0.044,
130      VTIPN      = 700.,
131
132      HINGE       = 3, !hinge configuration: 0 for no hinges, 1 for flap hinge, 2 for lag
133                hinge, 3 for both hinges
134      EFLAP      = 0.045, !flap hinge radial station
135      ELAG       = 0.05, !lag hinge radial station
136      EPITCH     = 0.06, !pitch bearing radial station
137      DLAG       = 2000., !lag hinge damping
138
139      CONTRL     = 2, !control system configuration, 2 for swashplate mechanism with control
140      PITCH      = 1, !Pitch bearing setting, 1 for pitch bearing
141      KPITCH     = 0., !Pitch bearing spring
142      LOCKP      = 1, !Pitch bearing joint motion (locked if 0)
143      LOCKPL     = 0, !Pitch link joint motion (locked if 0)
144      LOCKSP     = 0, !swashplate or actuator joint motion (locked if 0)
145
146      XSP        = 0.03, !pitch link attachment to swashplate chordwise position
147      YSP        = 0.045, !pitch link attachment to swashplate spanwise position
148      ZSP        = -0.1, !pitch link attachment to swashplate normal position
149
150      XPH        = 0.03, !pitch link attachment to pitch horn chordwise position
151      YPH        = 0.045, !pitch link attachment to pitch horn spanwise position
152      ZPH        = 0.0, !pitch link attachment to pitch horn normal position
153      EPH        = 0.061, !pitch horn attachment to blade radial station
154
155      OPBEAM     = 1, !blade analysis, 1 rigid beam,
156      DRELST     = 0.04, !rigid elements, minimum length for elastic beam (fraction radius)
157      KNODE      = 3, !number of additional nodes
158      RNODE      = 0.35, 0.64, 0.94, !location of additional nodes
159
160      TWIN       = 1, !blade twist input format
161      TWISTL     = -12., !linear twist rate (deg)
162
163      NPROP      = 4, !blade section properties, number of radial stations
164      RPROP      = 0., 0.5, 0.94, 1.0, !Blade section prperties locations
165
166      ZEA        = 8*0., !beam axis normal locus
167      ZQC        = 8*0., !quarter chord normal locus
168      ZC         = 8*0., !tension center normal offset
169      ZI         = 8*0., !inertial properties (mass weighted) normal offset
170      XEA        = 8*0., !beam axis chordwise locus
171      XQC        = 8*0., !quarter chord chordwise locus
172      XC         = 8*0., !tension center, chordwise offset
173      XI         = 8*0., !inertial properties (mass weighted) chordwise offset
174
175      XEA(4)     = 0.0, ! 0.0 deg sweep at r=0.94 !beam axis chordwise locus
176      XQC(4)     = 0.0, ! 0.0 deg sweep at r=0.94 !quarter chord chordwise locus
177
178      MASS       = 8*0.0826, ! Lock number = 5.33 inertia section mass
179
180      KP         = 8*0.012, !structural properties polar radius of gyration
181      KT         = 8*0.012, !extension/torsion coupling stiffness
182      EA         = 8*250000000., !axial stiffness
183      EIFLAP     = 8*1500000., !flap (out-of-plane) bending stiffness
184      EILAG      = 8*20000000., !lag (inplane) bending stiffness
185      GJ         = 8*200000., !torsion stiffness
186      ITHETA     = 8*0.0001, !section moment of inertia
187      IPOLAR     = 8*0.0001, !section polar inertia
188
189      &END
190      !=====
191      &NLDEF class='ROTOR',type='AERODYNAMICS',name='ROTOR 1',&END
192      &NLVAL
193      NPANEL     = 20,
194      REDGE      = 0.15, 0.22, 0.29, 0.36, 0.42, 0.48, 0.54, 0.59, 0.64, 0.69, 0.73,
195                .77, 0.81, 0.84, 0.87, 0.90, 0.92, 0.94, 0.96, 0.98, 1.00, !span

```

CAMRAD II input list file for the QSMR. Filename is 'qsmr.list.'

```

194      station at edges of panels (NPANEL+1 values)
194      NPROP      = 4, !number of span stations
195      RPROP      = 0., 0.94, 0.9401, 1.0, !location of span stations
196      CHORD      = 3*0.7363, 0.4418, ! 0.6 taper at r=0.94
197      ASWEEP     = 2*0., 2*0., ! 0.0 deg sweep at r=0.94
      sweep angle of quarter-chord relative to wing reference line (+ aft from inboard to
      outboard; deg)
198      NSEN       =10, OPREF = 10*4, ! aerodynamic sensors
199      QUANT      = 5, 6, 25, 31, 32, 33, 35, 71, 82, 82, ! lambda, lambdaint, alpha
200      IDENT      = 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ! uT, uR, uP, theta
201      AXIS       = 3, 3, 0, 0, 0, 0, 0, 0, 1, 3, ! L, Fx, Fz
202      OPSCL      = 2, 2, 1, 2, 2, 2, 1, 2, 2, 2, !
203      NAPLOT     = 1, 1, 4, 1, 1, 1, 1, 1, 1, 1, ! printer-plot selection 1 for
      time history
204  &END
205  !=====
206  &NLDEF class='ROTOR', type='INFLOW', name='ROTOR 1', &END
207  &NLVAL
208      KHLMDA     = 1.2, !empirical correction factor for momentum theory for inflow, hover
209      KFLMDA     = 2.0, !empirical correction factor for momentum theory for inflow, forward
      flight
210      FMLMDA     = 2*0., !linear inflow variation produced by hub moments and angular
      velocity, hub moment gradient factor
211  &END
212  !=====
213  &NLDEF class='ROTOR', type='WAKE', name='ROTOR 1', &END
214  &NLVAL
215      OPSCEN     = 1,
216  &END
217  !=====
218  &NLDEF class='ROTOR', type='WAKE', name='ROTOR 1', &END
219  &NLVAL
220      OPSCEN     = 0,
221      TWIST      = -12.,
222      RICWG      = 0.15,
223      OPFWG      = 0, !free geometry model -- not sure what 0 does
224      OPINTW     = 0, !rotor wake, nonuniform inflow interference velocity, wing-body - 0 to
      suppress
225      OPINTH     = 0, !rotor wake, nonuniform inflow interference velocity, horizontal tail
      - 0 to suppress
226      OPINTV     = 0, !rotor wake, nonuniform inflow interference velocity, vertical tail -
      0 to suppress
227  &END
228  !=====
229  ! tail rotor
230  !=====
231  &NLDEF class='AIRFRAME', type='STRUCTURE', &END
232  &NLVAL
233      MASSR(2)   = 0.1263, !equivalent hub inertia, mass
234      HSP(2)     = 0.064, !position of swashplate, distance swashplate below hub
235  &END
236  !=====
237  &NLDEF class='ROTOR', type='STRUCTURE', name='ROTOR 2', &END
238  &NLVAL
239      TITLE      = 'HELICOPTER TAIL ROTOR',
240      RADIUS     = 0.64,
241      NBLADE     = 13,
242      ROTATE     = 1,
243      SIGMA      = 0.5033,
244      VTIPN     = 500.,
245
246      HINGE      = 3,
247      EFLAP     = 0.23,
248      ELAG       = 0.231,
249      EPITCH     = 0.24,
250

```

CAMRAD II input list file for the QSMR. Filename is 'qsmr.list.'

```

251     AFLAP      = 30., ! control system (30 deg delta3), nonlinear aerodynamic model, wing
        flap angle
252     DLAG       = 10., !lag hinge, damping
253
254     CONTRL     = 2,
255     PITCH      = 1, !Pitch bearing setting, 1 for pitch bearing
256     KPITCH     = 0., !Pitch bearing spring
257     LOCKP      = 1, !Pitch bearing joint motion (locked if 0)
258     LOCKPL     = 0, !Pitch link joint motion (locked if 0)
259     LOCKSP     = 0, !swashplate or actuator joint motion (locked if 0)
260
261     XSP        = 0.03, !pitch link attachment to swashplate, chordwise position
262     YSP        = 0.2127, !pitch link attachment to swashplate, spanwise position
263     ZSP        = -0.1, !pitch link attachment to swashplate, normal position
264
265     XPH        = 0.03, !pitch link attachment to pitch horn (relative hub origin),
        chordwise position
266     YPH        = 0.2127, !pitch link attachment to pitch horn (relative hub origin),
        spanwise position
267     ZPH        = 0.0, !pitch link attachment to pitch horn (relative hub origin), normal
        position
268     EPH        = 0.241, !pitch horn attachment to blade, radial station
269
270     OPBEAM     = 1, !blade analysis element kind (1=rigid beam)
271     KNODE      = 0, !number of additional nodes
272
273     TWIN       = 1,
274     TWISTL     = 0.,
275
276     NPROP      = 2, !flexbeam section properties, number of radial stations
277     RPROP      = 0.,1., !flexbeam section properties, radial stations
278
279     ZEA        = 2*0., !beam axis normal locus
280     ZQC        = 2*0., !quarter chord normal locus
281     ZC         = 2*0., !tension center normal offset
282     ZI         = 2*0., !inertial properties (mass weighted) normal offset
283     XEA        = 2*0., !beam axis chordwise locus
284     XQC        = 2*0., !quarter chord chordwise locus
285     XC         = 2*0., !tension center, chordwise offset
286     XI         = 2*0., !inertial properties (mass weighted) chordwise offset
287
288     KP         = 2*0.022, !structural properties polar radius of gyration
289     KT         = 2*0.022, !extension/torsion coupling stiffness
290     EA         = 2*2000000., !axial stiffness
291     EIFLAP     = 2*850., !flap (out-of-plane) bending stiffness
292     EILAG      = 2*25000., !lag (inplane) bending stiffness
293     GJ         = 2*1500., !torsion stiffness
294     MASS       = 2*0.0152,
295     ITHETA     = 2*0.000001, !section moment of inertia
296     IPOLAR     = 2*0.000001, !section polar inertia
297     &END
298     !=====
299     &NLDEF class='ROTOR',type='AERODYNAMICS',name='ROTOR 2',&END
300     &NLVAL
301     NPANEL     = 8,
302     REDGE      = 0.25, 0.35, 0.45, 0.55, 0.65, 0.75, 0.85, 0.95, 1.00,
303     NPROP      = 2,
304     RPROP      = 0., 1.,
305     CHORD      = 2*.0778,
306     BTIP       = 0.95, !tip loss parameter, tip flow corrections
307     OPSTLL     = 0, ! no stall (NOTAR)
308     &END
309     !=====
310     &NLDEF class='ROTOR',type='INFLOW',name='ROTOR 2',&END
311     &NLVAL &END
312     !=====

```

CAMRAD II input list file for the QSMR. Filename is 'qsmr.list.'

```
313  &NLDEF class='ROTOR',type='WAKE',name='ROTOR 2',&END
314  &NLVAL &END
315  !=====
316  &NLDEF class='TABLES',&END
317  &NLVAL
318      AFTABL    = 1, !table kind (0 same table for all rotors, 1 separate tables)
319  &END
320  !=====
321  &NLDEF action='end of shell',&END
322  &NLDEF action='end of core',&END
323
```

CAMRAD II airfoil run file for the QSMR. Filename is 'helicopter_airfoil.run.'

```
1  #!/bin/csh
2  set camrad='C:\Users\lweist\Documents\2024\CAMRAD II\camradii_v5.0_win-x64_dist\bin'
3  #
4  setenv INPUTDECK1  vr12_rcc.c81
5  setenv INPUTDECK2  ssca09.c81
6  setenv OUTPUTTABLE helicopter.tab
7
8  ${camrad} > helicopter_airfoil.out << 'endofinput'
9  BATCH
10 &NLJOB
11     OPFILE = 7,
12     OPSRC  = 1,
13 &END
14 &NLTABL
15     OPFORM = 2,
16     RNTRP  = 0,
17     TITLE  = 'Single Main Rotor Helicopter Airfoils (VR12/SSCA09)'
18     NRB    = 2,
19     R      = 0.85, 0.95,
20 &END
21 'endofinput'
22
23 exit
24
```

CHARM run input file for the QSMR. Filename is 'qsmr.inp.'

```

1  QSMR vehicle input file
2  NROTOR    IMKS
3  2         0
4  PATHNAME
5  ../
6  INPUT FILES FOR ROTOR 1
7  qsmrmr_rw.inp
8  qsmrmr_bg.inp
9  qsmrmr_bd.inp
10 qsmrmr_af.inp
11 none
12 INPUT FILES FOR ROTOR 2
13 qsmrtr_rw.inp
14 qsmrtr_bg.inp
15 qsmrtr_bd.inp
16 qsmrtr_af.inp
17 none
18 SSPD      RHO
19 1116.45   0.002377
20 SFRAME    SMODE
21 2         1
22 WINDAC
23 -177.24   0.0      0.0
24 NPSI      NREV     CONVG+   CONVG2   CONVG3   MREV
25 24        200      -1.0     -1.0     -1.0     0
26 IRST      IFREE    IGPR
27 0         0        0
28 IVOUT      NRS      ROUT
29 4         8        0.5      0.6      0.7      0.8      0.85      0.9      0.95
30 0.99
31 NPRINT    IBLPLT   IFILPLT
32 0         -1       3        3        3        3
33 KOPT
34 0
35 ISCAN
36 0
37 ISTRSS
38 0
39 IFV        IQUIK1
40 2         1
41 IFVFW      IFVBL    IFVSU    IFVSC    IFVLS
42 1         1        1        1        1
43 ISURF
44 0
45 ISHIP
46 0
47 IRECON     NOISE    KBROADBAND KSEL
48 1         -4        1        1
49 ACOUSTICS_CODE
50 2
51 NCHDRC     NPSIRC   IVROLL   CDRMIN   IRECONTYPE
52 1         360      1        0.02    1
53 NLS
54 0
55 NAC
56 1
57 XAC
58 0.0        0.0      0.0
59 YAW        PITCH    ROLL
60 0.0        0.0      0.0
61 VAC
62 0.0        0.0      0.0
63 OMCg
64 0.0        0.0      0.0
65 TRIMFILENAME
standardtrim.inp

```

CHARM run input file for the QSMR. Filename is 'qsmr.inp.'

```
66  WEIGHT    FUSPLANA FUSSIDEA FUSREFL
67    3428.0    0.0        0.0        0.0
68  DFRACFUSE
69    0.0        0.0
70  CGAC
71    0.0        0.0        3.3
72  FUSFILENAME
73  fusedata_table.inp
74  KACR
75    1
76  NTRAJ
77    0
```

CHARM main rotor blade geometry input file for the QSMR. Filename is 'qsmrmr_bg.inp.'

```

1  QSMR main rotor blad geometry input
2  NSEG
3    20
4  CUTOUT
5    2.478
6  SL(1:NSEG)
7    3*1.15640, 3*0.99120, 3*0.82600, 3*0.66080, 3*0.49560, 5*0.33040
8  CHORD(1:NSEG+1)
9    18*0.73630, 0.63846, 0.54013, 0.44180,
10 ELOFSG(1:NSEG+1)
11  21*0.00000
12 SWEEP(1:NSEG)
13  20*0.00000
14 TWRD (Twist at CUTOUT)
15  7.20000
16 TWSTGD(1:NSEG)
17  3*-0.84000, 3*-0.72000, 3*-0.60000, 3*0.48000, 3*-0.36000, 5*-0.24000
18 ANHD(1:NSEG)
19  20*0.00000
20 THIKND(1:NSEG+1)
21  18*0.12000, 3*0.09000
22 KFLAP(1:NSEG)
23  20*0
24 FLAPND(1:NSEG)
25  20*0.00000
26 FLHNGE(1:NSEG)
27  20*0.00000
28 FLDEFL(1:NSEG)
29  20*0.00000
30 NCAM
31  0
32 NCHORD  NSPAN  ICOS
33  1      50    0
34

```

CHARM main rotor blade dynamics input file for the QSMR. Filename is 'qsmrmr_bd.inp'

```

1  QSMR main rotor blade dynamics input
2  ISTRM
3  0
4  ISTRIP  IFPC  IAERO
5  -1      0      0
6  ICOMP  IRVFLO  ISTFLO
7  1      0      1
8  IART  HINGE  PRECONE
9  1      0.0450  0.000
10 NMODE
11 1
12 NMDFLP  NMDTOR  NMDLAG  NMDELG
13 1      0      0      0
14 IFXMDE  IRIGID
15 0      0
16 FREQMD  GMASS      for Mode 1
17 1.035  136.8
18 NRBARS
19 51
20 RBARS(IR), IR=1,NRBARS
21 0.000  0.020  0.040  0.060  0.080  0.100  0.120  0.140  0.160  0.180
22 0.200  0.220  0.240  0.260  0.280  0.300  0.320  0.340  0.360  0.380
23 0.400  0.420  0.440  0.460  0.480  0.500  0.520  0.540  0.560  0.580
24 0.600  0.620  0.640  0.660  0.680  0.700  0.720  0.740  0.760  0.780
25 0.800  0.820  0.840  0.860  0.880  0.900  0.920  0.940  0.960  0.980
26 1.000
27 Rigid flap mode
28 0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000
29 0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000
30 0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000
31 0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000
32 0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000
33 0.000
34 0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000
35 0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000
36 0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000
37 0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000
38 0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000
39 0.000
40 0.000  0.000  0.000  0.259  0.605  0.951  1.297  1.643  1.989  2.335
41 2.681  3.027  3.373  3.719  4.065  4.411  4.757  5.103  5.449  5.795
42 6.141  6.487  6.833  7.179  7.525  7.871  8.217  8.563  8.909  9.255
43 9.601  9.947  10.293  10.639  10.985  11.330  11.676  12.022  12.368  12.714
44 13.060 13.406 13.752 14.098 14.444 14.790 15.136 15.482 15.828 16.174
45 16.520
46 0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000
47 0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000
48 0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000
49 0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000
50 0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000
51 0.000
52 0.000  0.000  0.000  1.047  1.047  1.047  1.047  1.047  1.047  1.047
53 1.047  1.047  1.047  1.047  1.047  1.047  1.047  1.047  1.047  1.047
54 1.047  1.047  1.047  1.047  1.047  1.047  1.047  1.047  1.047  1.047
55 1.047  1.047  1.047  1.047  1.047  1.047  1.047  1.047  1.047  1.047
56 1.047  1.047  1.047  1.047  1.047  1.047  1.047  1.047  1.047  1.047
57 1.047
58 0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000
59 0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000
60 0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000
61 0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000
62 0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000
63 0.000
64

```

CHARM main rotor rotor wake input file for the QSMR. Filename is 'qsmrmr_rw.inp.'

```

1  QSMR main rotor rotor wake input
2  NBLADE      OMEGA      SOLIDITY
3    3         42.373      0.044
4  IROTAT      XROTOR
5    1         0.0        0.0        0.0        X,Y,Z tilt      ITILT
6    1         0.0        0.0        0.0        -3.0        0.0        0
7  ICOLL      COLLD      CT
8    0         0.00      0.2175928
9  ITRIM      A1W      B1W      A1S      B1S
10   0         0.0        0.0        0.0        0.0
11 NOWAKE      ICNVCT      NWAKES      NPWAKE      IFAR      MBCVE
12   0         0         1         0         0         0
13 KSCHEME      KPC
14   0         0
15 NCUT      AOV LAP      ISKEW      IUNS
16   1         -1         1         1
17 NZONE      NVORT      30         2      BUFFER      KGMCHG      KPERIODIC
18   3         30         30         2.0        0         1
19 ICORE      AKINEM      A1      PCOREM      CRMON
20   1         0.0        0.001      0.0        0.0
21 NPTFW
22   48         48         72
23 (CORLIM(NV,IZONE,1), NV=1,NVORT(IZONE) IZONE=1 (Min core radii)
24 15*0.5 15*0.01
25 (CORLIM(NV,IZONE,2), NV=1,NVORT(IZONE) IZONE=1 (Max core radii)
26 1.0
27 (CORLIM(NV,IZONE,1), NV=1,NVORT(IZONE) IZONE=2 (Min core radii)
28 0.01
29 (CORLIM(NV,IZONE,2), NV=1,NVORT(IZONE) IZONE=2 (Max core radii)
30 1.0
31 (CORLIM(NV,IZONE,1), NV=1,NVORT(IZONE) IZONE=3 (Min core radii)
32 0.5 0.1
33 (CORLIM(NV,IZONE,2), NV=1,NVORT(IZONE) IZONE=3 (Max core radii)
34 0.5 0.1
35 (CUTLIM(NV,IZONE,1), NV=1,NVORT(IZONE) IZONE=1 (Min cutoff distances)
36 15*0.5 15*0.
37 (CUTLIM(NV,IZONE,2), NV=1,NVORT(IZONE) IZONE=1 (Max cutoff distances)
38 1.0
39 (CUTLIM(NV,IZONE,1), NV=1,NVORT(IZONE) IZONE=2 (Min cutoff distances)
40 0.01
41 (CUTLIM(NV,IZONE,2), NV=1,NVORT(IZONE) IZONE=2 (Max cutoff distances)
42 1.0
43 (CUTLIM(NV,IZONE,1), NV=1,NVORT(IZONE) IZONE=2 (Min cutoff distances)
44 0.5 0.1
45 (CUTLIM(NV,IZONE,2), NV=1,NVORT(IZONE) IZONE=2 (Max cutoff distances)
46 0.5 0.1
47 IDYNM
48   1
49 SRAD      SHGHT
50   0.0      0.0
51 NHHI
52   0

```

CHARM tail rotor blade geometry input file for the QSMR. Filename is 'qsmrtr_bg.inp.'

```
1  QSMR tail rotor blade geometry input
2  NSEG
3    8
4  CUTOUT
5    0.16000
6  SL(1:NSEG)
7    7*0.06400,    0.03200
8  CHORD(1:NSEG+1)
9    9*0.084283
10 ELOFSG(1:NSEG+1)
11    9*0.00000
12 SWEEP(1:NSEG)
13    8*0.00000
14 TWRD (Twist at CUTOUT)
15    0.00000
16 TWSTGD(1:NSEG)
17    8*0.00000
18 ANHD(1:NSEG)
19    8*0.00000
20 THIKND(1:NSEG+1)
21    7*0.12000,    2*0.09000
22 KFLAP(1:NSEG)
23    8*0
24 FLAPND(1:NSEG)
25    8*0.00000
26 FLHNGE(1:NSEG)
27    8*0.00000
28 FLDEFL(1:NSEG)
29    8*0.00000
30 NCAM
31    0
32 NCHORD  NSPAN  ICOS
33    1      -50    0
34
```

CHARM tail rotor blade dynamics input file for the QSMR. Filename is 'qsmtr_bd.inp'

```

1  QSMR tail rotor blade dynamics input
2  ISTRM
3  0
4  ISTRIP  IFPC  IAERO
5  -1      0      0
6  ICOMP   IRVFLO  ISTFLO
7  1       0       0
8  IART    HINGE   PRECONE
9  1      0.2300   0.000
10 NMODE
11 1
12 NMDFLP  NMDTOR  NMDLAG  NMDELG
13 1       0       0       0
14 IFXMDE   IRIGID
15 0        0
16 FREQMD   GMASS      for Mode 1
17 1.203    150.
18 NRBARS
19 51
20 RBARS(IR), IR=1,NRBARS
21 0.000 0.020 0.040 0.060 0.080 0.100 0.120 0.140 0.160 0.180
22 0.200 0.220 0.240 0.260 0.280 0.300 0.320 0.340 0.360 0.380
23 0.400 0.420 0.440 0.460 0.480 0.500 0.520 0.540 0.560 0.580
24 0.600 0.620 0.640 0.660 0.680 0.700 0.720 0.740 0.760 0.780
25 0.800 0.820 0.840 0.860 0.880 0.900 0.920 0.940 0.960 0.980
26 1.000
27 Rigid flap mode
28 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
29 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
30 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
31 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
32 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
33 0.000
34 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
35 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
36 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
37 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
38 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
39 0.000
40 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
41 0.000 0.000 0.008 0.025 0.042 0.058 0.075 0.091 0.108 0.125
42 0.141 0.158 0.175 0.191 0.208 0.224 0.241 0.258 0.274 0.291
43 0.308 0.324 0.341 0.357 0.374 0.391 0.407 0.424 0.441 0.457
44 0.474 0.490 0.507 0.524 0.540 0.557 0.574 0.590 0.607 0.623
45 0.640
46 -0.577 -0.577 -0.577 -0.577 -0.577 -0.577 -0.577 -0.577 -0.577 -0.577
47 -0.577 -0.577 -0.577 -0.577 -0.577 -0.577 -0.577 -0.577 -0.577 -0.577
48 -0.577 -0.577 -0.577 -0.577 -0.577 -0.577 -0.577 -0.577 -0.577 -0.577
49 -0.577 -0.577 -0.577 -0.577 -0.577 -0.577 -0.577 -0.577 -0.577 -0.577
50 -0.577 -0.577 -0.577 -0.577 -0.577 -0.577 -0.577 -0.577 -0.577 -0.577
51 -0.577
52 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
53 0.000 0.000 1.299 1.299 1.299 1.299 1.299 1.299 1.299 1.299
54 1.299 1.299 1.299 1.299 1.299 1.299 1.299 1.299 1.299 1.299
55 1.299 1.299 1.299 1.299 1.299 1.299 1.299 1.299 1.299 1.299
56 1.299 1.299 1.299 1.299 1.299 1.299 1.299 1.299 1.299 1.299
57 1.299
58 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
59 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
60 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
61 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
62 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
63 0.000
64

```

CHARM tail rotor rotor wake input file for the QSMR. Filename is 'qsmrtr_rw.inp.'

```

1  QSMR tail rotor rotor wake input
2  NBLADE      OMEGA      SOLIDITY
3    12      780.034      0.050303
4  IROTAT      XROTOR      X,Y,Z tilt      ITILT
5    1      -19.82      0.0      -0.66      90.0      0.0      0.0      0
6  ICOLL      COLLD      CT
7    0      0.00      0.003522917
8  ITRIM      A1W      B1W      A1S      B1S
9    0      0.0      0.0      0.0      0.0
10 NOWAKE      ICNVCT      NWAKES      NPWAKE      IFAR      MBCVE
11    0      2      1      0      0      0
12 KSCHEME      KPC
13    0      0
14 NCUT      AOV LAP      ISKEW      IUNS
15    1      -1      1      1
16 NZONE      NVORT      BUFFER      KGMCHG      KPERIODIC
17    3      30      30      2      2.0      0      1
18 ICORE      AKINEM      A1      PCOREM      CRMON
19    1      0.0      0.001      0.0      0.0
20 NPTEFW
21    48      48      72
22 (CORLIM(NV,IZONE,1), NV=1,NVORT(IZONE) IZONE=1 (Min core radii)
23 15*0.5 15*0.01
24 (CORLIM(NV,IZONE,2), NV=1,NVORT(IZONE) IZONE=1 (Max core radii)
25 1.0
26 (CORLIM(NV,IZONE,1), NV=1,NVORT(IZONE) IZONE=2 (Min core radii)
27 0.01
28 (CORLIM(NV,IZONE,2), NV=1,NVORT(IZONE) IZONE=2 (Max core radii)
29 1.0
30 (CORLIM(NV,IZONE,1), NV=1,NVORT(IZONE) IZONE=3 (Min core radii)
31 0.5 0.1
32 (CORLIM(NV,IZONE,2), NV=1,NVORT(IZONE) IZONE=3 (Max core radii)
33 0.5 0.1
34 (CUTLIM(NV,IZONE,1), NV=1,NVORT(IZONE) IZONE=1 (Min cutoff distances)
35 15*0.5 15*0.
36 (CUTLIM(NV,IZONE,2), NV=1,NVORT(IZONE) IZONE=1 (Max cutoff distances)
37 1.0
38 (CUTLIM(NV,IZONE,1), NV=1,NVORT(IZONE) IZONE=2 (Min cutoff distances)
39 0.01
40 (CUTLIM(NV,IZONE,2), NV=1,NVORT(IZONE) IZONE=2 (Max cutoff distances)
41 1.0
42 (CUTLIM(NV,IZONE,1), NV=1,NVORT(IZONE) IZONE=2 (Min cutoff distances)
43 0.5 0.1
44 (CUTLIM(NV,IZONE,2), NV=1,NVORT(IZONE) IZONE=2 (Max cutoff distances)
45 0.5 0.1
46 IDYNM
47    1
48 SRAD      SHGHT
49    0.0      0.0
50 NHHI
51    0
52

```

CHARM vehicle trim input file for the QSMR. Filename is 'standardtrim.inp.'

```

1  standardtrim.inp: Standard inputs for single main rotor / tail rotor helicopter trim
2  Fx   Fy   Fz   Mx   My   Mz                               Aircraft forces and moments to
   trim
3  0.5  0.5  0.5  0.5  0.5  0.5
4  Fx   Fy   Fz   Mx   My   Mz                               Optional trim values (default
   is 0)
5  0.0  0.0  0.0  0.0  0.0  0.0
6  Fx   Fy   Fz   Mx   My   Mz                               Trim tolerance (fraction of
   weight)
7  0.2  0.2  0.2  0.2  0.2  0.2
8  Coll Lat Long Pedal Throttle Tilt Flap1 Flap2 Elev Rudder Addl      Pilot
   controls
9  1     1     1     1     0     0     0     0     0     0     0
10 Roll Pitch Sideslip
11 1     1     0
12 Number of aerodynamic components (rotors and wings) used to trim
13 2
14 Comp RPM Coll Lat Long Tilt Flaps(4)
   Controls used by each aero component
15 1     0     1     2     3     0     0 0 0
   0                               Aero Component 1: Main rotor
16 2     0     4     0     0     0     0 0 0
   0                               Aero Component 2: Tail rotor
17 Ctrl Zero Coll Lat Long Pedal Throttle Tilt Flap1 Flap2 Elev Rudder
   Addl Mixer: Component/Pilot control gains
18 1 0.0 1.0 0.0 0.0 0.0 1.0 0.0 0.0 0.0 0.0 0.0
   0.0 Rotor 1 Col 1:1 with Pilot Col Stick
19 2 0.0 0.0 1.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
   0.0 Rotor 1 Lat 1:1 with Pilot Lat Stick
20 3 0.0 0.0 0.0 1.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
   0.0 Rotor 1 Lon 1:1 with Pilot Lon Stick
21 4 0.0 0.0 0.0 0.0 1.0 0.0 0.0 0.0 0.0 0.0 0.0
   0.0 Rotor 2 Col 1:1 with Pilot Pedal
22 Initial Pilot Control Settings Based on the Gains Above
23 4.31 -2.07 4.21 23.46 0.0 0.0 0.0 0.0 0.0 0.0 0.0
24

```

CHARM fuselage input file for the QSMR. Filename is 'fusedata.table.inp.'

```

1  ZERO OUT fuselage forces to match with CAMRAD II OPAERO = 0 for fuselage.
2  9  3      ! X-force, body frame
3  2  1      ! Y-force, body frame
4  11 1      ! Z-force, body frame
5  2  1      ! X (roll) moment, body frame
6  10 1      ! Y (pitch) moment, body frame
7  1  7      ! Z (yaw) moment, body frame
8      -90.   0.   90.
9  -180.   0.   0.   0.
10 -160.   0.   0.   0.
11 -90.    0.   0.   0.
12 -30.    0.   0.   0.
13  0.     0.   0.   0.
14  20.    0.   0.   0.
15  90.    0.   0.   0.
16  160.   0.   0.   0.
17  180.   0.   0.   0.
18 -180.  180.
19  0.00  0.00
20 -180. -160. -120. -60. -20.  0.   20.   60.  120.  160.  180.
21  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00
22 -180.  180.
23  0.00  0.00
24 -205. -160. -130. -60. -25.  25.   60.  130.  155.  200.
25  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00
26 -90. -70. -25.  0.0  25.   70.   90.
27  0.00  0.00  0.00  0.00  0.00  0.00  0.00
28

```

A.3 Side-by-Side

The following files are the CAMRAD II and CHARM input files for the Side-by-Side. As the SBS has two main rotors, there are both rotor list files and an aircraft list file to combine them. Special attention should be paid to the wake model to ensure proper rotor-rotor interactions are captured. As with the QSMR, airfoil tables for the SSC-A09 and VR-12 airfoils are omitted as they are proprietary and have yet to be published.

- CAMRADII_SBS
 - camradii
 - * **side-vtip-fly.run**
 - camradii-inputs
 - * airfoil
 - **sidebyside_airfoil.run**
 - Airfoil tables omitted, but go here
 - * inputs
 - **sidebyside_input.run**
 - **sidebysidea.list**
 - **sidebyside1.list**
 - **sidebyside2.list**

For CHARM, the input files required are as listed below. The SBS contains two main rotors and a horizontal tail broken into left and right components. Both rotors and horizontal tails are able to share blade geometry, blade dynamics, and airfoil input files, but each requires its own wake input file. The trim file is also different from the QSMR and was written for the control scheme of the SBS specifically. Airfoil tables for the main rotors are once again omitted as they have yet to be officially published.

- **sbscruise.inp**
- **sbsbg.inp**
- **sbsbd.inp**
- **sbsrRrw.inp**
- **sbsLrw.inp**
- **htailbg.inp**
- **htailbd.inp**
- **htailaf.inp**
- **lefthtailrw.inp**
- **righthtailrw.inp**
- **sbstrim.inp**
- **fusedata_table.inp**

CAMRAD II run file for the Side-by-Side. Filename is 'side-vtip-fly.run.'

```

1  #!/bin/csh
2
3  set camrad='C:\Users\lweist\Documents\2024\CAMRAD II\camradii_v5.0_win-x64_dist\bin'
4
5  # Side by Side Helicopter
6  # forward flight
7
8  setenv BLADEAIRFOIL1 ../camradii-inputs/airfoil/sidebyside.tab
9  setenv SHELLINPUT ../camradii-inputs/inputs/sidebyside.bin
10 $camrad/camradii > > side550fly.out << 'endofinput'
11 &NLJOB NCASES=1,OPINIT=0,OUTRES=1, PLFILE=0,&END
12 !=====
13 &NLDEF class='CASE',&END
14 &NLVAL
15     OPDENS=2,ALTMSL=0.,TEMP=77., ! operating condition
16 &END
17 &NLDEF class='TRIM',&END
18 &NLVAL
19     VKTS=50.,VTIP=552.46, ! operating condition
20     LEVEL=2*3,NWPRNT=0,ITERF=3, ! free wake
21     MPSI = 24,
22     OPTRIM=1, ! trim
23     COLL=9.,LATCYC=0.,LNGCYC=-0.35,PEDAL=0.,PITCH=0.,ROLL=0.,
24     RELAXT=.1,OPPID=2,ITERT=80,TOLERT=.2, ! convergence
25     RELAXC(2)=2*.05,ITERC=100,TOLERC=.2,
26     TOLERR=2*.5,
27     OPPOST=1,MPSIH=240, ! post trim
28 &END
29 &NLDEF class='TRIM ROTOR',name='ROTOR 1',&END
30 &NLVAL DOFB(2)=0,&END ! no lag
31 &NLDEF class='TRIM ROTOR',name='ROTOR 2',&END
32 &NLVAL DOFB(2)=0,&END ! no lag
33 !=====
34 ! sound sensors
35 &NLDEF class='TRIM ROTOR',name='ROTOR 1',&END
36 &NLVAL
37     MPSEN=1,
38     MPTIME=240,
39     MWSEN = 1,
40     NWPRNT = 1,
41     NWFILE = 1,
42 &END
43 &NLDEF class='TRIM ROTOR',name='ROTOR 2',&END
44 &NLVAL
45     MPSEN=1,
46     MPTIME=240,
47     MWSEN = 1,
48     NWPRNT = 1,
49     NWFILE = 1,
50 &END
51 &NLDEF class='ROTOR',type='STRUCTURE',name='ROTOR 1',&END
52 &NLVAL OPSND=1,&END
53 &NLDEF class='ROTOR',type='STRUCTURE',name='ROTOR 2',&END
54 &NLVAL OPSND=1,&END
55 ! aero sensors
56 &NLDEF class='TRIM ROTOR',name='ROTOR 1',&END
57 &NLVAL MASEN=1,MATIME=240,&END
58 &NLDEF class='TRIM ROTOR',name='ROTOR 2',&END
59 &NLVAL MASEN=1,MATIME=240,&END
60 &NLDEF class='ROTOR',type='AERODYNAMICS',name='ROTOR 1',&END
61 &NLVAL
62     NSEN=7,
63     QUANT=5,5,5,53,81,81,64,
64     IDENT=0,0,0,1,1,1,1,
65     AXIS=1,2,3,0,1,3,0,
66     OPSCL=0,0,0,1,2,2,2,

```

CAMRAD II run file for the Side-by-Side. Filename is 'side-vtip-fly.run.'

```

67      OPREF=1,1,1,1,2,2,2,
68 &END
69 &NLDEF class='ROTOR',type='AERODYNAMICS',name='ROTOR 2',&END
70 &NLVAL
71     NSEN=7,
72     QUANT=5,5,5,53,81,81,64,
73     IDENT=0,0,0,1,1,1,1,
74     AXIS=1,2,3,0,1,3,0,
75     OPSCL=0,0,0,1,2,2,2,
76     OPREF=1,1,1,1,2,2,2,
77 &END
78 !=====
79 &NLDEF class='ROTOR',type='WAKE',name='ROTOR 1',action='init',&END
80 &NLVAL OPSCEN=1,&END ! forward flight wake
81 &NLDEF class='ROTOR',type='WAKE',name='ROTOR 1',&END
82 &NLVAL OPSCEN=0,
83     CORE=0.4,CORE(3)=1.,COREWG=0.4, ! vortex core
84     OPFWG=4,OPDISP=2*0, ! wake model
85     RNW=0.25,RFW=3.,RDW=3.,MFWG=3, ! wake extent
86     OPFW=2, ! dual peak
87     OPINTW=0,OPINTH=0,OPINTV=0, ! no interference
88     CORE=0.8,COREWG=0.8,OPTVIC=0, ! core
89 &END
90 !!-----
91 &NLDEF class='ROTOR',type='WAKE',name='ROTOR 2',action='init',&END
92 &NLVAL OPSCEN=1,&END ! forward flight wake
93 &NLDEF class='ROTOR',type='WAKE',name='ROTOR 2',&END
94 &NLVAL OPSCEN=0,
95     CORE=0.4,CORE(3)=1.,COREWG=0.4, ! vortex core
96     OPFWG=4,OPDISP=2*0, ! wake model
97     RNW=0.25,RFW=3.,RDW=3.,MFWG=3, ! wake extent
98     OPFW=2, ! dual peak
99     OPINTW=0,OPINTH=0,OPINTV=0, ! no interference
100    CORE=0.8,COREWG=0.8,OPTVIC=0, ! core
101 &END
102 !=====
103 &NLDEF class='AIRFRAME',type='STRUCTURE',&END
104 &NLVAL
105     WEIGHT=3787.,
106     FSRTTR= 0.000, 0.000,
107     BLRTR= 9.18, -9.18,
108     WLRTR= 3.152, 3.152,
109     WLCG= -1.051,WLWB= -1.051,FSHT= 12.608,WLHT= 0.525,FSVT= 12.608,WLVT= 0.525,
110     MASSR=2*2.4951,HSP=2*1.0507,
111 &END
112 &NLDEF class='AIRFRAME',type='AERODYNAMICS',&END
113 &NLVAL DRGOW=5.30,DRGVW=5.30,LFTAH=16.62,LFTAV= 8.67,&END
114 !-----
115 &NLDEF class='ROTOR',type='STRUCTURE',name='ROTOR 1',&END
116 &NLVAL RADIUS=10.55,NBLADE=4,SIGMA=0.08319,VTIPN=552.3,MASS=4*0.05937,&END
117 &NLDEF class='ROTOR',type='AERODYNAMICS',name='ROTOR 1',&END
118 &NLVAL CHORD=3*0.71194226,.4265,&END
119 !-----
120 &NLDEF class='ROTOR',type='STRUCTURE',name='ROTOR 2',&END
121 &NLVAL RADIUS=10.55,NBLADE=4,SIGMA=0.08319,VTIPN=552.3,MASS=4*0.05937,&END
122 &NLDEF class='ROTOR',type='AERODYNAMICS',name='ROTOR 2',&END
123 &NLVAL CHORD=3*0.71194226,.4265,&END
124 !=====
125 &NLDEF action='end of shell',&END
126 &NLDEF action='end of core',&END
127 !#####
128 'endofinput'

```

CAMRAD II input run file for the Side-by-Side. Filename is 'sidebyside_input.run.'

```
1  #!/bin/csh
2  set camrad='C:\Users\lweist\Documents\2024\CAMRAD II\camradii_v5.0_win-x64_dist\bin'
3  #
4  setenv INPUTLIST1 sidebysidea.list
5  setenv INPUTLIST2 sidebyside1.list
6  setenv INPUTLIST3 sidebyside2.list
7  setenv OUTPUTFILE sidebyside.bin
8  $camrad/input_camrad_lm > sidebyside_input.out << 'endofinput'
9  BATCH
10  &NLJOB NLISTI=3,NLISTC=0,&END
11  'endofinput'
12  #
13
```

CAMRAD II input aircraft list file for the Side-by-Side. Filename is 'sidebysidea.list.'

```

1  ! Side by Side Helicopter
2  !=====
3  &NLDEF class='CASE',&END
4  &NLVAL
5      TITLE='SIDE BY SIDE HELICOPTER',          ! description
6      OPUNIT=1,DENSE=0.00237689,TEMP=59.,        ! environment
7  &END
8  !=====
9  &NLDEF class='TRIM',&END
10 &NLVAL
11     VELIN=1,WINDIN=1,VTIPIN=1,VTIP=550.,        ! operating condition
12     LEVEL=2*1,                                  ! wake loop
13     COLL=6.,MTRIM=3,                             ! free flight trim
14     MNAME='FORCE X ','FORCE Z ','MOMENT Y ',
15     VNAME='COLL ','LNGCYC ','PITCH ',
16     MHARMR=2*1,MHARMA=2*1,MHARMD=2*1,          ! harmonics
17     DOFA=6*0,                                    ! degrees of freedom
18 &END
19 &NLDEF class='TRIM ROTOR',name='ROTOR 1',&END
20 &NLVAL OPMODE=0,DOFB=12*1,&END                  ! degrees of freedom
21 &NLDEF class='TRIM ROTOR',name='ROTOR 2',&END
22 &NLVAL OPMODE=0,DOFB=12*1,&END                  ! degrees of freedom
23 !=====
24 &NLDEF class='AIRFRAME',type='STRUCTURE',&END
25 &NLVAL
26     TITLE='SIDE BY SIDE HELICOPTER AIRFRAME',    ! description
27     CONFIG=5,NROTOR=2,                          ! multirotor
28     WEIGHT=3468.,                                ! inertia
29     IXX=500.,IYY=150.,IZZ=600,
30     FSCG=0.,WLCG=0.,                             ! geometry
31     FSRTR=0.,0.,BLRTR=9.18,-9.18,WLRTR=3.15,3.15,
32     FSWB=0.,WLWB=0.,FSHT=12.61,WLHT=0.,FSVT=12.61,WLVT=0.,
33     ASHAFT=-3,-3,
34 &END
35 &NLDEF class='AIRFRAME',type='AERODYNAMICS',&END
36 &NLVAL
37     DRGOW=5.29,DRGVW=5.29,DRGIW=0.,
38 &END
39 &NLDEF class='AIRFRAME',type='CONTROL',&END
40 &NLVAL
41     TCRIN=2*1,
42     TCR(1,1,1)=1.,TCR(1,2,1)=-1.,TCR(3,3,1)=-1.,TCR(3,4,1)= 1.,
43     TCR(1,1,2)=1.,TCR(1,2,2)= 1.,TCR(3,3,2)=-1.,TCR(3,4,2)=-1.,
44 &END
45 !=====
46 &NLDEF class='TABLES',&END
47 &NLVAL &END
48 !=====
49 &NLDEF action='end of shell',&END
50 &NLDEF action='end of core',&END

```

CAMRAD II input rotor 1 list file for the Side-by-Side. Filename is 'sidebyside1.list.'

```

1  ! Side by Side Helicopter Right Rotor
2  !=====
3  &NLDEF class='AIRFRAME',type='STRUCTURE',&END
4  &NLVAL
5      MASSR(1)=1.9422,           ! inertia
6      HSP(1)=1.051,             ! control system
7      AZREF(1)=0.,              ! rotor phasing
8  &END
9  !=====
10 &NLDEF class='ROTOR',type='STRUCTURE',name='ROTOR 1',&END
11 &NLVAL
12     TITLE='SIDE BY SIDE HELICOPTER RIGHT ROTOR',           ! description
13     RADIUS=10.51,NBLADE=4,ROTATE=1,SIGMA=.0832,
14     VTIPN=550.,
15     HINGE=3,EFLAP=.045,ELAG=.05,EPITCH=.06,              ! blade root
16     DLAG=1000.,
17     CONTRL=2,PITCH=1,KPITCH=0.,LOCKP=1,                   ! control system
18     XSP=.03,YSP=.045,ZSP=-.1,
19     XPH=.03,YPH=.045,ZPH=.0,
20     EPH=.061,LOCKPL=0,LOCKSP=0,
21     OPBEAM=1,DRELST=.04,KNODE=3,RNODE=.35,.64,.94,        ! rigid blade
22     TWIN=1,TWISTL=-16.07,
23     NPROP=4,RPROP=0.,.5,.94,1.,                           ! section properties
24     ZEA=4*0.,ZQC=4*0.,ZC=4*0.,ZI=4*0.,
25     XEA=4*0.,XQC=4*0.,XC=4*0.,XI=4*0.,
26     XEA(4)=0.01608,XQC(4)=0.01608,
27     EDS = 0.94, SWEEP = 15., ! 15 deg sweep at r=.94
28     KP=4*.012,KT=4*.012,EA=4*250000000.,
29     EIFLAP=4*1500000.,EILAG=4*20000000.,GJ=4*200000.,
30     MASS=4*.0462,ITHETA=4*.002,IPOLAR=4*.002,
31 &END
32 !=====
33 &NLDEF class='ROTOR',type='AERODYNAMICS',name='ROTOR 1',&END
34 &NLVAL
35     NPANEL=19,           ! CSMOD: was 20 aero panels
36     REDGE=.16,.26,.36,.42,.48,.54,.59,.64,.69,.73,.77,.81,.84,.87,.90,.92,.94,.96,.98,
37     1.0,
38     NPROP=4,RPROP=0.,.94,.9401,1.,           ! aero properties
39     CHORD=3*0.71194226,.4265,                ! .6 taper at r=.94
40     ASWEEP=2*0.,2*15.,                       ! 15 deg sweep at r=.94
41     NSEN=10,OPREF=10*4,                      ! aerodynamic sensors
42     QUANT= 5,6,25,31,32,33,35,71,82,82,      ! lambda,lambdaint,alpha
43     IDENT= 0,0, 0, 0, 0, 0, 0, 0, 0, 0,      ! uT,uR,uP,theta
44     AXIS=  3,3, 0, 0, 0, 0, 0, 0, 1, 3,      ! L,Fx,Fz
45     OPSCL= 2,2, 1, 2, 2, 2, 1, 2, 2, 2,      !
46     NAPLOT=1,1, 4, 1, 1, 1, 1, 1, 1, 1,      !
47 &END
48 !=====
49 &NLDEF class='ROTOR',type='INFLOW',name='ROTOR 1',&END
50 &NLVAL
51     KHLMDA=1.2,KFLMDA=2.,FMLMDA=2*0.,
52 &END
53 &NLDEF class='ROTOR',type='WAKE',name='ROTOR 1',&END
54 &NLVAL TWIST=-16.,RICWG=.15,OPFWG=4,OPDISP=2*0,OPINTW=0,OPINTH=0,OPINTV=0,&END
55 !=====
56 &NLDEF action='end of shell',&END
57 &NLDEF action='end of core',&END

```

CAMRAD II input rotor 2 list file for the Side-by-Side. Filename is 'sidebyside2.list.'

```

1  ! Side by Side Helicopter Left Rotor
2  !=====
3  &NLDEF class='AIRFRAME',type='STRUCTURE',&END
4  &NLVAL
5      MASSR(2)=1.9422,                ! inertia
6      HSP(1)=1.051,                  ! control system
7      AZREF(2)=45.,                  ! rotor phasing
8  &END
9  !=====
10 &NLDEF class='ROTOR',type='STRUCTURE',name='ROTOR 2',&END
11 &NLVAL
12     TITLE='SIDE BY SIDE HELICOPTER LEFT ROTOR',          ! description
13     RADIUS=10.51,NBLADE=4,ROTATE=-1,SIGMA=.0832,
14     VTIPN=550.,
15     HINGE=3,EFLAP=.045,ELAG=.05,EPITCH=.06,              ! blade root
16     DLAG=1000.,
17     CONTRL=2,PITCH=1,KPITCH=0.,LOCKP=1,                  ! control system
18     XSP=.03,YSP=.045,ZSP=-.1,
19     XPH=.03,YPH=.045,ZPH=.0,
20     EPH=.061,LOCKPL=0,LOCKSP=0,
21     OPBEAM=1,DRELST=.04,KNODE=3,RNODE=.35,.64,.94,        ! rigid blade
22     TWIN=1,TWISTL=-16.07,
23     NPROP=4,RPROP=0.,.5,.94,1.,                          ! section properties
24     ZEA=4*0.,ZQC=4*0.,ZC=4*0.,ZI=4*0.,
25     XEA=4*0.,XQC=4*0.,XC=4*0.,XI=4*0.,
26     XEA(4)=0.01608,XQC(4)=0.01608,
27     EDS = 0.94, SWEEP = 15., ! 15 deg sweep at r=.94
28     KP=4*.012,KT=4*.012,EA=4*250000000.,
29     EIFLAP=4*1500000.,EILAG=4*20000000.,GJ=4*200000.,
30     MASS=4*.0462,ITHETA=4*.002,IPOLAR=4*.002,
31 &END
32 !=====
33 &NLDEF class='ROTOR',type='AERODYNAMICS',name='ROTOR 2',&END
34 &NLVAL
35     NPANEL=19,                ! CSMOD: was 20 aero panels
36     REDGE=.16,.26,.36,.42,.48,.54,.59,.64,.69,.73,.77,.81,.84,.87,.90,.92,.94,.96,.98,1.0
37     /
38     NPROP=4,RPROP=0.,.94,.9401,1.,          ! aero properties
39     CHORD=3*0.71194226,.4265,                ! .6 taper at r=.94
40     ASWEEP=2*0.,2*15.,                      ! 15 deg sweep at r=.94
41     NSEN=10,OPREF=10*4,                    ! aerodynamic sensors
42     QUANT= 5,6,25,31,32,33,35,71,82,82,      ! lambda,lambdaint,alpha
43     IDENT= 0,0, 0, 0, 0, 0, 0, 0, 0, 0,      ! uT,uR,uP,theta
44     AXIS=  3,3, 0, 0, 0, 0, 0, 0, 1, 3,      ! L,Fx,Fz
45     OPSCL= 2,2, 1, 2, 2, 2, 1, 2, 2, 2,      !
46     NAPLOT=1,1, 4, 1, 1, 1, 1, 1, 1, 1,      !
47 &END
48 !=====
49 &NLDEF class='ROTOR',type='INFLOW',name='ROTOR 2',&END
50 &NLVAL
51     KHLMDA=1.2,KFLMDA=2.,FMLMDA=2*0.,
52 &END
53 &NLDEF class='ROTOR',type='WAKE',name='ROTOR 1',&END
54 &NLVAL TWIST=-16.,RICWG=.15,OPFWG=4,OPDISP=2*0,OPINTW=0,OPINTH=0,OPINTV=0,&END
55 !=====
56 &NLDEF action='end of shell',&END
57 &NLDEF action='end of core',&END

```

CAMRAD II airfoil run file for the Side-by-Side. Filename is 'sidebyside_airfoil.run.'

```
1  #!/bin/csh
2  set camrad='C:\Users\lweist\Documents\2024\CAMRAD II\camradii_v5.0_win-x64_dist\bin'
3  #
4  setenv INPUTDECK1 vr12_rcc.c81
5  setenv INPUTDECK2 ssca09.c81
6  setenv OUTPUTTABLE sidebyside.tab
7  $camrad/input_camrad_lm > sidebyside_airfoil.out << 'endofinput'
8  BATCH
9  &NLJOB OPFILE=7,OPSRC=1,&END
10 &NLTABL OPFORM=2,RNTRP=0,
11     TITLE='Side-by-Side Main Rotor Airfoils (VR12/SSCA09)'
12     NRB=2,R=.85,.95,
13 &END
14 'endofinput'
15
```

CHARM run input file for the Side-by-Side. Filename is 'sbscruise.inp.'

```

1  sbscruise.inp: SBS aircraft in cruise
2  NROTOR    IMKS
3    4        0
4  PATHNAME
5  ../
6  INPUT FILENAMES for right rotor
7  rotorRrw.inp
8  rotorbg.inp
9  rotorbd.inp
10 rotorl_af.inp
11 none
12 INPUT FILENAMES for left rotor
13 rotorLrw.inp
14 rotorbg.inp
15 rotorbd.inp
16 rotorl_af.inp
17 none
18 INPUT FILENAMES for right horizontal tail
19 righthtailrw.inp
20 htailbg.inp
21 htailbd.inp
22 htailaf.inp
23 none
24 INPUT FILENAMES for left horizontal tail
25 lefthtailrw.inp
26 htailbg.inp
27 htailbd.inp
28 htailaf.inp
29 none
30 SSPD      RHO
31  1135.66  0.002377
32 SFRAME    SMODE
33    2        1
34 WINDAC
35    0.0      0.0      0.0
36 NPSI      NREV      CONVG+  CONVG2  CONVG3  MREV
37    24      30      0.0001  0.1      0.001  0
38 IRST      IFREE      IGPR
39    0        0
40 IVOUT      NRS      ROUT
41    4        20      0.15      0.2      0.25      0.3      0.35      0.4      0.45
42    0.5
43    0.55      0.6      0.65      0.7      0.75      0.8      0.85      0.9      0.925
44    0.95
45    0.975      0.999
46 NPRINT    IBLPLT    IFILPLT
47    0        -1      3      3      3      3
48 KOPT
49    0
50 ISCAN
51    0
52 ISTRSS
53    0
54 IFV      IQUIK1
55    2        1
56 IFVFW      IFVBL      IFVSU      IFVSC      IFVLS
57    1        1      1      1      1
58 ISURF
59    0
60 ISHIP
61    0
62 IRECON      NOISE      KBROADBAND KSEL
63    1        -4      0      0
64 ACOUSTICS_CODE
65    2
66 NCHDRC      NPSIRC      IVROLL      CDRMIN      IRECONTYPE

```

CHARM run input file for the Side-by-Side. Filename is 'sbscruise.inp.'

```
65      1          240          1          0.02      1
66      NLS
67      0
68      NAC
69      1
70      XAC
71      0.0          0.0          0.0
72      YAW          PITCH          ROLL
73      0.0          2.0          0.0
74      VAC
75      84.39          0.0          0.0
76      OMCG
77      0.0          0.0          0.0
78      TRIMFILENAME
79      sbstrim.inp
80      WEIGHT      FUSPLANA  FUSSIDEA  FUSREFL
81      3787          56.4          56.4          17.5
82      DFRACFUSE
83      0.5          0.5          0.0          0.0
84      CGAC
85      0.0          0.0          3.15
86      FUSFILENAME
87      fusedata_table.inp
88      KACR
89      1
90      NTRAJ
91      0
```

CHARM main rotor blade geometry input file for the Side-by-Side. Filename is 'sbsbg.inp'

```

1  bladebg.inp: CHARM Blade Geometry Input File made from blade.inp
2  KBGEOM
3    0
4  NSEG
5    29
6  CUTOUT
7    1.728160      -0.000000      0.000000
8  SL(ISEG)
9    25*0.30008 0.2647764 0.2647764 0.2647764 0.5295528
10 CHORD(ISEG)
11    29*0.71194226      0.426509
12 ELOFSG(ISEG) - (elastic axis offset)
13    30*0.000000
14 SWEEP(ISEG)
15    28*0.00 15
16 TWRD (Blade root twist at zero collective in degrees)
17    12
18 TWSTGD(ISEG)
19    25*-0.45696      3*-0.4032000000000002      -0.8064000000000002
20 ANHD(ISEG)
21    29*0.00
22 THIKND(ISEG)
23    0.106265      0.106265      0.106265      0.106265      0.106265
24    0.106265      0.106265      0.106265      0.106265      0.106265
25    0.106265      0.106265      0.106265      0.106265      0.106265
26    0.106265      0.106265      0.106265      0.106265      0.106265
27    0.106265      0.106265      0.106265      0.106265      0.105447
28    0.100836      0.096177      0.091469      0.090172      0.090172
29 KFLAP(ISEG)
30    29*0
31 FLAPND(ISEG)
32    29*0.0
33 FLHNGE(ISEG)
34    29*0.0
35 FLDEFL(ISEG)
36    29*0.0
37 NCAM
38    0
39 NCHORD NSPAN ICOS
40    1      50      -1
41
42

```

CHARM main rotor blade dynamics input file for the Side-by-Side. Filename is 'sbsbd.inp.'

```

1  rotorbd.inp: Blade Dynamics Input File for Side-by-Side main rotor
2  ISTRM
3  0
4  ISTRIP  IFPC  IAERO
5  -1      0      1
6  ICOMP  IRVFLO  ISTFLO
7  1      1      2
8  IART  HINGE  PRECONE
9  1      0.045  0.000
10 NMODE
11 1
12 NMDFLP  NMDTOR  NMDLAG  NMDELG
13 1      0      0      0
14 IFXMDE  IRRIGID
15 0      0
16 FREQMD  GMASS      for Mode 1
17 1.035  27.75
18 NRBARS
19 51
20 RBARS(IR), IR=1,NRBARS
21 0.000 0.020 0.040 0.060 0.080 0.100 0.120 0.140 0.160 0.180
22 0.200 0.220 0.240 0.260 0.280 0.300 0.320 0.340 0.360 0.380
23 0.400 0.420 0.440 0.460 0.480 0.500 0.520 0.540 0.560 0.580
24 0.600 0.620 0.640 0.660 0.680 0.700 0.720 0.740 0.760 0.780
25 0.800 0.820 0.840 0.860 0.880 0.900 0.920 0.940 0.960 0.980
26 1.000
27 Rigid flap mode
28 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
29 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
30 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
31 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
32 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
33 0.000
34 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
35 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
36 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
37 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
38 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
39 0.000
40 0.000 0.000 0.000 0.170 0.396 0.622 0.848 1.074 1.301 1.527
41 1.753 1.979 2.205 2.431 2.658 2.884 3.110 3.336 3.562 3.788
42 4.015 4.241 4.467 4.693 4.919 5.146 5.372 5.598 5.824 6.050
43 6.276 6.503 6.729 6.955 7.181 7.407 7.634 7.860 8.086 8.312
44 8.538 8.764 8.991 9.217 9.443 9.669 9.895 10.121 10.348 10.574
45 10.800
46 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
47 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
48 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
49 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
50 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
51 0.000
52 0.000 0.000 0.000 1.047 1.047 1.047 1.047 1.047 1.047 1.047
53 1.047 1.047 1.047 1.047 1.047 1.047 1.047 1.047 1.047 1.047
54 1.047 1.047 1.047 1.047 1.047 1.047 1.047 1.047 1.047 1.047
55 1.047 1.047 1.047 1.047 1.047 1.047 1.047 1.047 1.047 1.047
56 1.047 1.047 1.047 1.047 1.047 1.047 1.047 1.047 1.047 1.047
57 1.047
58 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
59 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
60 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
61 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
62 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
63 0.000

```

CHARM right main rotor rotor wake input file for the Side-by-Side. Filename is 'sbsrRrw.inp.'

```

1  rotorRrw.inp Rotor Wake Input File for NASA Side-by-side concept
2  NBLADE  OMEGA
3    4      52.364
4  IROTAT      XROTOR      X,Y,Z tilt  ITILT
5    -1      0.0 -9.18 0.0      0.0 0.0 45.0      1
6  ICOLL  COLL      CT
7    1      5.0      0.0093
8  ITRIM      A1W      B1W      A1S      B1S
9    0      0.0      0.0      0.0      0.0
10 NOWAKE  ICNVCT  NWAKES  NPWAKE  IFAR  MBCVE
11    0      0      1      0      0      0
12 KSCHEME  KPC
13    0      0
14 NCUT  AOV LAP  ISKEW  IUNS
15    1      -1      1      1
16 NZONE  (NVORT(I), I=1,NZONE)  BUFFER  KGMCHG  KPERIODIC
17    3      24 24 2 2.0      2      1
18 ICORE  AKINEM  A1  PCOREM  CRMON
19    1      0.0 0.001 0.0      0.0
20 (NPTFW(I), I=1,NZONE)
21 96 96 96
22 (CORLIM(NV,IZONE,1), NV=1,NVORT(IZONE) IZONE=1 (Min core radii)
23 0.01
24 (CORLIM(NV,IZONE,2), NV=1,NVORT(IZONE) IZONE=1 (Max core radii)
25 1.0
26 (CORLIM(NV,IZONE,1), NV=1,NVORT(IZONE) IZONE=2 (Min core radii)
27 0.01
28 (CORLIM(NV,IZONE,2), NV=1,NVORT(IZONE) IZONE=2 (Max core radii)
29 1.0
30 (CORLIM(NV,IZONE,1), NV=1,NVORT(IZONE) IZONE=3 (Min core radii)
31 0.01
32 (CORLIM(NV,IZONE,2), NV=1,NVORT(IZONE) IZONE=3 (Max core radii)
33 1.0
34 (CUTLIM(NV,IZONE,1), NV=1,NVORT(IZONE) IZONE=1 (Min cutoff distances)
35 0.01
36 (CUTLIM(NV,IZONE,2), NV=1,NVORT(IZONE) IZONE=1 (Max cutoff distances)
37 1.0
38 (CUTLIM(NV,IZONE,1), NV=1,NVORT(IZONE) IZONE=2 (Min cutoff distances)
39 0.01
40 (CUTLIM(NV,IZONE,2), NV=1,NVORT(IZONE) IZONE=2 (Max cutoff distances)
41 1.0
42 (CUTLIM(NV,IZONE,1), NV=1,NVORT(IZONE) IZONE=2 (Min cutoff distances)
43 0.01
44 (CUTLIM(NV,IZONE,2), NV=1,NVORT(IZONE) IZONE=2 (Max cutoff distances)
45 1.0
46 IDYNM
47    1
48 SRAD  SHGHT
49    0.0 0.0
50 NHHI (Higher harmonic cyclic pitch input flag)
51    0
52

```

CHARM left main rotor rotor wake input file for the Side-by-Side. Filename is 'sbsLrw.inp.'

```

1  rotorLrw.inp Rotor Wake Input File for NASA Side-by-side concept
2  NBLADE  OMEGA
3    4      52.364
4  IROTAT      XROTOR      X,Y,Z tilt  ITILT
5    1      0.0 9.18 0.0      0.0 0.0 0.0      1
6  ICOLL  COLL      CT
7    1      5.0      0.0093
8  ITRIM      A1W      B1W      A1S      B1S
9    0      0.0      0.0      0.0      0.0
10 NOWAKE  ICNVCT  NWAKES  NPWAKE  IFAR  MBCVE
11    0      0      1      0      0      0
12 KSCHEME  KPC
13    0      0
14 NCUT  AOV LAP  ISKEW  IUNS
15    1      -1      1      1
16 NZONE  (NVORT(I), I=1,NZONE)  BUFFER  KGMCHG  KPERIODIC
17    3      24 24 2 2.0      2      1
18 ICORE  AKINEM  A1  PCOREM  CRMON
19    1      0.0 0.001 0.0      0.0
20 (NPTFW(I), I=1,NZONE)
21 96 96 96
22 (CORLIM(NV,IZONE,1), NV=1,NVORT(IZONE) IZONE=1 (Min core radii)
23 0.01
24 (CORLIM(NV,IZONE,2), NV=1,NVORT(IZONE) IZONE=1 (Max core radii)
25 1.0
26 (CORLIM(NV,IZONE,1), NV=1,NVORT(IZONE) IZONE=2 (Min core radii)
27 0.01
28 (CORLIM(NV,IZONE,2), NV=1,NVORT(IZONE) IZONE=2 (Max core radii)
29 1.0
30 (CORLIM(NV,IZONE,1), NV=1,NVORT(IZONE) IZONE=3 (Min core radii)
31 0.01
32 (CORLIM(NV,IZONE,2), NV=1,NVORT(IZONE) IZONE=3 (Max core radii)
33 1.0
34 (CUTLIM(NV,IZONE,1), NV=1,NVORT(IZONE) IZONE=1 (Min cutoff distances)
35 0.01
36 (CUTLIM(NV,IZONE,2), NV=1,NVORT(IZONE) IZONE=1 (Max cutoff distances)
37 1.0
38 (CUTLIM(NV,IZONE,1), NV=1,NVORT(IZONE) IZONE=2 (Min cutoff distances)
39 0.01
40 (CUTLIM(NV,IZONE,2), NV=1,NVORT(IZONE) IZONE=2 (Max cutoff distances)
41 1.0
42 (CUTLIM(NV,IZONE,1), NV=1,NVORT(IZONE) IZONE=2 (Min cutoff distances)
43 0.01
44 (CUTLIM(NV,IZONE,2), NV=1,NVORT(IZONE) IZONE=2 (Max cutoff distances)
45 1.0
46 IDYNM
47    1
48 SRAD  SHGHT
49    0.0 0.0
50 NHHI (Higher harmonic cyclic pitch input flag)
51    0
52

```

CHARM horizontal tail blade geometry input file for the Side-by-Side. Filename is 'htailbg.inp.'

```
1  htailbg.inp: CHARM Blade Geometry Input File for half horiz tail
2  NSEG
3    1
4  CUTOUT
5    0.0
6  SL(ISEG)
7    3.5
8  CHORD(ISEG)
9    1.76738  1.237166
10 ELOFSG(ISEG) - (elastic axis offset)
11    2*0.0
12 SWEEP(ISEG)
13    6.8717
14 TWRD (Blade root twist at zero collective in degrees)
15    0.0
16 TWSTGD(ISEG)
17    0.0
18 ANHD(ISEG)
19    0.0
20 THIKND(ISEG)
21    2*.12
22 KFLAP(ISEG) - no information yet
23    1
24 FLAPND(ISEG)
25    0.25
26 FLHNGE(ISEG)
27    0.0
28 FLDEFL(ISEG)
29    0.0
30 NCAM
31    0
32 NCHORD  NSPAN  ICOS
33    4      10    0
34
```

CHARM horizontal tail blade dynamics input file for the Side-by-Side. Filename is 'htailbd.inp'

```

1  htailbd.inp: CHARM Blade Dynamics Input File for half horizontal tail
2  ISTRM
3  0
4  ISTRIP  IFPC  IAERO
5  -1      0      1
6  ICOMP   IRVFLO  ISTFLO
7  1       0       3
8  IART    HINGE   PRECONE
9  0       0.000   0.000
10 NMODE
11 1
12 NMDFLP  NMDTOR  NMDLAG  NMDELG
13 1       0       0       0
14 IFXMDE   IRIGID
15 0        1
16 FREQMD   GMASS      for Mode 1
17 1.000    2.229
18 NRBARS
19 51
20 RBARS(IR), IR=1,NRBARS
21 0.000 0.020 0.040 0.060 0.080 0.100 0.120 0.140 0.160 0.180
22 0.200 0.220 0.240 0.260 0.280 0.300 0.320 0.340 0.360 0.380
23 0.400 0.420 0.440 0.460 0.480 0.500 0.520 0.540 0.560 0.580
24 0.600 0.620 0.640 0.660 0.680 0.700 0.720 0.740 0.760 0.780
25 0.800 0.820 0.840 0.860 0.880 0.900 0.920 0.940 0.960 0.980
26 1.000
27 Rigid flap mode
28 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
29 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
30 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
31 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
32 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
33 0.000
34 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
35 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
36 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
37 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
38 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
39 0.000
40 0.000 0.061 0.121 0.182 0.242 0.303 0.364 0.424 0.485 0.546
41 0.606 0.667 0.727 0.788 0.849 0.909 0.970 1.031 1.091 1.152
42 1.212 1.273 1.334 1.394 1.455 1.516 1.576 1.637 1.697 1.758
43 1.819 1.879 1.940 2.001 2.061 2.122 2.182 2.243 2.304 2.364
44 2.425 2.486 2.546 2.607 2.667 2.728 2.789 2.849 2.910 2.970
45 3.031
46 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
47 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
48 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
49 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
50 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
51 0.000
52 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000
53 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000
54 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000
55 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000
56 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000
57 1.000
58 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
59 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
60 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
61 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
62 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
63 0.000
64

```

CHARM horizontal tail airfoil input file for the Side-by-Side. Filename is 'htailaf.inp.'

```

1      2  0
2      0.0  1.0
3      0.12  0.12
4      COMMENT#1
5      COMMENT#2
6      0012                113911651047
7      0.      .20      .30      .40      .50      .60      .7      .75      .8      .9      1.
8      -180.  0.      0.      0.      0.      0.      0.      0.      0.      0.      0.
9      -172.5 .78      .78      .78      .78      .78      .78      .78      .78      .78      .78
10     -161.  .62      .62      .62      .62      .62      .62      .62      .62      .62      .62
11     -147.  1.      1.      1.      1.      1.      1.      1.      1.      1.      1.
12     -129.  1.      1.      1.      1.      1.      1.      1.      1.      1.      1.
13     -49.   -1.18 -1.18 -1.18 -1.18 -1.18 -1.18 -1.18 -1.18 -1.18 -1.18
14     -39.   -1.18 -1.18 -1.18 -1.18 -1.18 -1.18 -1.18 -1.18 -1.18 -1.18
15     -21.   -.8     -.8     -.81    -.83    -.85    -.85    -.85    -.71    -.68    -.64
16     -16.5  -1.007 -1.007 -.944    -.96    -.965   -.965   -.965   -.795   -.76    -.7
17     -15.   -1.19 -1.19 -1.09 -1.055 -.99    -.98    -.98    -.83    -.79    -.72
18     -14.   -1.333 -1.333 -1.22 -1.096 -1.    -.97    -.97    -.84    -.805   -.73    -.73
19     -13.   -1.334 -1.334 -1.28 -1.12    -1.    -.96    -.96    -.85    -.815   -.735   -.735
20     -12.   -1.255 -1.255 -1.26 -1.13    -1.    -.947   -.94    -.85    -.82    -.74    -.74
21     -11.   -1.161 -1.161 -1.19 -1.12    -.994   -.93    -.923   -.85    -.81    -.74
22     -10.   -1.055 -1.055 -1.01 -1.082 -.985   -.91    -.90    -.845   -.805   -.73    -.73
23     -8.     -.844   -.844   -.88    -.907   -.922   -.87    -.84    -.82    -.77    -.695   -.695
24     -6.     -.633   -.633   -.66    -.684   -.741   -.77    -.75    -.77    -.72    -.593   -.593
25     -4.     -.422   -.422   -.440   -.456   -.494   -.544   -.578   -.627   -.603   -.396   -.396
26     -2.     -.211   -.211   -.22    -.228   -.247   -.272   -.313   -.350   -.395   -.2     -.2
27     0.      0.      0.      0.      0.      0.      0.      0.      0.      0.      0.
28     2.      .211   .211   .22    .228   .247   .272   .313   .350   .395   .2     .2
29     4.      .422   .422   .44    .456   .494   .544   .578   .627   .603   .396   .396
30     6.      .633   .633   .66    .684   .741   .77    .75    .77    .72    .593   .593
31     8.      .844   .844   .88    .907   .922   .87    .84    .82    .77    .695   .695
32     10.     1.055  1.055  1.1    1.082 .985    .91    .90    .845   .805   .73    .73
33     11.     1.161  1.161  1.19  1.12    .994    .93    .923   .850   .810   .74    .74
34     12.     1.255  1.255  1.26  1.13    1.    .947    .94    .85    .82    .74    .74
35     13.     1.334  1.334  1.28  1.12    1.    .96    .96    .85    .815   .735   .735
36     14.     1.333  1.333  1.22  1.096  1.    .97    .97    .84    .805   .73    .73
37     15.     1.19   1.19   1.09  1.055 .99     .98    .98    .83    .79    .73    .73
38     16.5    1.007  1.007 .944    .96    .965   .965   .965   .795   .76    .7     .7
39     21.     .8     .8     .81    .83    .85    .85    .85    .71    .68    .64    .64
40     39.     1.18   1.18   1.18  1.18   1.18  1.18  1.18  1.18  1.18  1.18  1.18
41     49.     1.18   1.18   1.18  1.18   1.18  1.18  1.18  1.18  1.18  1.18  1.18
42     129.    -1.     -1.     -1.     -1.     -1.     -1.     -1.     -1.     -1.     -1.     -1.
43     147.    -1.     -1.     -1.     -1.     -1.     -1.     -1.     -1.     -1.     -1.     -1.
44     161.    -.62    -.62    -.62    -.62    -.62    -.62    -.62    -.62    -.62    -.62    -.62
45     172.5   -.78    -.78    -.78    -.78    -.78    -.78    -.78    -.78    -.78    -.78    -.78
46     180.    0.      0.      0.      0.      0.      0.      0.      0.      0.      0.      0.
47     0.      0.      .18    .28    .38    .48    .62    .72    .77    .82    .92    1.0
48     -180.   .022    .022    .022    .022    .022    .022    .022    .022    .022    .022    .022
49     -175.   .062    .062    .062    .062    .062    .062    .062    .062    .062    .062    .062
50     -170.   .132    .132    .132    .132    .132    .132    .132    .132    .132    .132    .132
51     -165.   .242    .242    .242    .242    .242    .242    .242    .242    .242    .242    .242
52     -160.   .302    .302    .302    .302    .302    .302    .302    .302    .302    .302    .302
53     -140.   1.042    1.042    1.042    1.042    1.042    1.042    1.042    1.042    1.042    1.042    1.042
54     -120.   1.652    1.652    1.652    1.652    1.652    1.652    1.652    1.652    1.652    1.652    1.652
55     -110.   1.852    1.852    1.852    1.852    1.852    1.852    1.852    1.852    1.852    1.852    1.852
56     -100.   2.022    2.022    2.022    2.022    2.022    2.022    2.022    2.022    2.022    2.022    2.022
57     -90.    2.022    2.022    2.022    2.022    2.022    2.022    2.022    2.022    2.022    2.022    2.022
58     -80.    1.962    1.962    1.962    1.962    1.962    1.962    1.962    1.962    1.962    1.962    1.962
59     -70.    1.842    1.842    1.842    1.842    1.842    1.842    1.842    1.842    1.842    1.842    1.842
60     -60.    1.662    1.662    1.662    1.662    1.662    1.662    1.662    1.662    1.662    1.662    1.662
61     -50.    1.392    1.392    1.392    1.392    1.392    1.399    1.392    1.392    1.392    1.392    1.392
62     -30.    .562    .562    .562    .562    .562    .562    .562    .562    .562    .562    .562
63     -21.    .332    .332    .332    .332    .332    .332    .332    .332    .332    .342    .342
64     -16.    .155    .155    .181    .207    .235    .257    .274    .292    .305    .342    .342
65     -15.    .102    .102    .148    .181    .209    .233    .252    .271    .282    .298    .298
66     -14.    .038    .038    .099    .146    .180    .212    .233    .249    .260    .293    .293

```

CHARM horizontal tail airfoil input file for the Side-by-Side. Filename is 'htailaf.inp.'

67	-13.	.0264	.0264	.0455	.094	.148	.191	.216	.231	.239	.272	.292
68	-12.	.022	.022	.030	.06	.111	.164	.198	.211	.220	.252	.291
69	-11.	.0196	.0196	.0232	.038	.078	.135	.17	.192	.202	.232	.275
70	-10.	.0174	.0174	.0189	.0259	.053	.105	.145	.176	.186	.213	.254
71	-9.	.0154	.0154	.0159	.0187	.0351	.077	.122	.159	.172	.199	.232
72	-8.	.0138	.0138	.0138	.0147	.0220	.053	.101	.140	.155	.183	.214
73	-7.	.0122	.0122	.0122	.0123	.0141	.035	.082	.111	.139	.169	.192
74	-6.	.011	.011	.011	.011	.011	.0212	.0615	.082	.12	.14	.17
75	-5.	.01	.01	.01	.01	.01	.0132	.038	.054	.088	.111	.14
76	-4.	.0093	.0093	.0093	.0093	.0093	.01	.0167	.03	.0575	.095	.112
77	-3.	.0088	.0088	.0088	.0088	.0088	.009	.0102	.0175	.0355	.086	.102
78	-2.	.0085	.0085	.0085	.0085	.0085	.0085	.0086	.0117	.0240	.081	.098
79	-1.	.0083	.0083	.0083	.0083	.0083	.0083	.0083	.0091	.0175	.078	.096
80	0.	.008	.008	.008	.008	.008	.008	.008	.008	.0137	.078	.095
81	1.	.0083	.0083	.0083	.0083	.0083	.0083	.0083	.0091	.0175	.078	.096
82	2.	.0085	.0085	.0085	.0085	.0085	.0085	.0086	.0117	.024	.081	.098
83	3.	.0088	.0088	.0088	.0088	.0088	.0090	.0102	.0175	.0355	.086	.102
84	4.	.0093	.0093	.0093	.0093	.0093	.01	.0167	.03	.0575	.095	.112
85	5.	.01	.01	.01	.01	.01	.0132	.038	.054	.088	.111	.14
86	6.	.011	.011	.011	.011	.011	.0212	.0615	.082	.12	.14	.17
87	7.	.0122	.0122	.0122	.0123	.0141	.035	.082	.111	.139	.169	.192
88	8.	.0138	.0138	.0138	.0147	.022	.053	.101	.14	.155	.183	.214
89	9.	.0154	.0154	.0159	.0187	.0351	.077	.122	.159	.172	.199	.232
90	10.	.0174	.0174	.0189	.0259	.053	.105	.145	.176	.186	.213	.254
91	11.	.0196	.0196	.0232	.038	.078	.135	.17	.192	.202	.232	.275
92	12.	.022	.022	.03	.06	.111	.164	.198	.211	.22	.252	.291
93	13.	.0264	.0264	.0455	.094	.148	.191	.216	.231	.239	.272	.292
94	14.	.038	.038	.099	.146	.18	.212	.233	.249	.26	.293	.293
95	15.	.102	.102	.148	.181	.209	.233	.252	.271	.282	.298	.298
96	16.	.155	.155	.181	.207	.235	.257	.274	.292	.305	.342	.342
97	21.	.332	.332	.332	.332	.332	.332	.332	.332	.332	.342	.342
98	30.	.562	.562	.562	.562	.562	.562	.562	.562	.562	.562	.562
99	50.	1.392	1.392	1.392	1.392	1.392	1.392	1.392	1.392	1.392	1.392	1.392
100	60.	1.662	1.662	1.662	1.662	1.662	1.662	1.662	1.662	1.662	1.662	1.662
101	70.	1.842	1.842	1.842	1.842	1.842	1.842	1.842	1.842	1.842	1.842	1.842
102	80.	1.962	1.962	1.962	1.962	1.962	1.962	1.962	1.962	1.962	1.962	1.962
103	90.	2.022	2.022	2.022	2.022	2.022	2.022	2.022	2.022	2.022	2.022	2.022
104	100.	2.022	2.022	2.022	2.022	2.022	2.022	2.022	2.022	2.022	2.022	2.022
105	110.	1.852	1.852	1.852	1.852	1.852	1.852	1.852	1.852	1.852	1.852	1.852
106	120.	1.652	1.652	1.652	1.652	1.652	1.652	1.652	1.652	1.652	1.652	1.652
107	140.	1.042	1.042	1.042	1.042	1.042	1.042	1.042	1.042	1.042	1.042	1.042
108	160.	.302	.302	.302	.302	.302	.302	.302	.302	.302	.302	.302
109	165.	.242	.242	.242	.242	.242	.242	.242	.242	.242	.242	.242
110	170.	.132	.132	.132	.132	.132	.132	.132	.132	.132	.132	.132
111	175.	.062	.062	.062	.062	.062	.062	.062	.062	.062	.062	.062
112	180.	.022	.022	.022	.022	.022	.022	.022	.022	.022	.022	.022
113		.00	.30	.40	.50	.6	.7	.75	.8	.9	1.	
114	-180.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	
115	-170.	.4	.4	.4	.4	.4	.4	.4	.4	.4	.4	
116	-165.	.3	.3	.3	.3	.3	.3	.3	.3	.3	.3	
117	-160.	.3	.3	.3	.3	.3	.3	.3	.3	.3	.3	
118	-135.	.5	.5	.5	.5	.5	.5	.5	.5	.5	.5	
119	-90.	.5	.5	.5	.5	.5	.5	.5	.5	.5	.5	
120	-30.	.174	.184	.196	.214	.235	.25	.264	.277	.298	.319	
121	-23.	.112	.118	.128	.144	.157	.171	.183	.206	.232	.258	
122	-16.	.073	.078	.086	.097	.108	.117	.137	.176	.200	.224	
123	-15.	.054	.065	.073	.084	.097	.111	.133	.173	.195	.217	
124	-14.	0.	.027	.054	.068	.086	.103	.127	.167	.189	.211	
125	-13.	0.	.0015	.025	.05	.074	.093	.122	.163	.184	.205	
126	-12.	0.	0.	.002	.03	.06	.083	.116	.157	.176	.195	
127	-11.	0.	0.	-.003	.014	.046	.074	.108	.149	.17	.191	
128	-10.	0.	0.	-.0015	.002	.032	.065	.10	.142	.163	.184	
129	-9.	0.	0.	0.	-.003	.016	.054	.089	.132	.154	.176	
130	-8.	0.	0.	0.	-.004	.005	.041	.082	.123	.145	.167	
131	-7.	0.	0.	0.	0.	-.004	.0275	.072	.1125	.136	.160	
132	-6.	0.	0.	0.	0.	-.003	.016	.0625	.10	.125	.150	

CHARM horizontal tail airfoil input file for the Side-by-Side. Filename is 'htailaf.inp.'

```

133 -4. 0. 0. 0. 0. 0. .005 .04 .076 .102 .128
134 -3. 0. 0. 0. 0. 0. -.0025 .026 .0665 .087 .108
135 -2. 0. 0. 0. 0. 0. 0. .013 .053 .07 .087
136 -1. 0. 0. 0. 0. 0. 0. .0035 .033 .045 .057
137 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.
138 1. 0. 0. 0. 0. 0. 0. -.0035 -.033 -.045 -.057
139 2. 0. 0. 0. 0. 0. 0. -.013 -.053 -.07 -.087
140 3. 0. 0. 0. 0. 0. .0025 -.026 -.0665 -.087 -.108
141 4. 0. 0. 0. 0. 0. -.005 -.04 -.076 -.102 -.128
142 6. 0. 0. 0. 0. 0. .003 -.016 -.0625 -.1 -.125 -.150
143 7. 0. 0. 0. 0. .004 -.0275 -.072 -.1125 -.136 -.160
144 8. 0. 0. 0. .004 -.005 -.041 -.082 -.123 -.145 -.167
145 9. 0. 0. 0. .003 -.016 -.054 -.089 -.132 -.154 -.176
146 10. 0. 0. .0015 -.002 -.032 -.065 -.1 -.142 -.163 -.184
147 11. 0. 0. .003 -.014 -.046 -.074 -.108 -.149 -.17 -.191
148 12. 0. 0. -.002 -.03 -.06 -.083 -.116 -.157 -.176 -.195
149 13. 0. -.0015 -.025 -.05 -.074 -.093 -.122 -.163 -.184 -.205
150 14. 0. -.027 -.054 -.068 -.086 -.103 -.127 -.167 -.189 -.211
151 15. -.054 -.065 -.073 -.084 -.097 -.111 -.133 -.173 -.195 -.217
152 16. -.073 -.078 -.086 -.097 -.108 -.117 -.137 -.176 -.20 -.224
153 23. -.112 -.118 -.128 -.144 -.157 -.171 -.183 -.206 -.232 -.258
154 30. -.174 -.184 -.196 -.214 -.235 -.250 -.264 -.277 -.298 -.319
155 90. -.5 -.5 -.5 -.5 -.5 -.5 -.5 -.5 -.5 -.5
156 135. -.5 -.5 -.5 -.5 -.5 -.5 -.5 -.5 -.5 -.5
157 160. -.3 -.3 -.3 -.3 -.3 -.3 -.3 -.3 -.3 -.3
158 165. -.3 -.3 -.3 -.3 -.3 -.3 -.3 -.3 -.3 -.3
159 170. -.4 -.4 -.4 -.4 -.4 -.4 -.4 -.4 -.4 -.4
160 180. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.
161 COMMENT#1
162 COMMENT#2
163 0012 113911651047
164 0. .20 .30 .40 .50 .60 .7 .75 .8 .9 1.
165 -180. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.
166 -172.5 .78 .78 .78 .78 .78 .78 .78 .78 .78 .78
167 -161. .62 .62 .62 .62 .62 .62 .62 .62 .62 .62
168 -147. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1.
169 -129. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1.
170 -49. -1.18 -1.18 -1.18 -1.18 -1.18 -1.18 -1.18 -1.18 -1.18 -1.18
171 -39. -1.18 -1.18 -1.18 -1.18 -1.18 -1.18 -1.18 -1.18 -1.18 -1.18
172 -21. -.8 -.8 -.81 -.83 -.85 -.85 -.85 -.71 -.68 -.64
173 -16.5 -1.007 -1.007 -.944 -.96 -.965 -.965 -.965 -.795 -.76 -.7 -.7
174 -15. -1.19 -1.19 -1.09 -1.055 -.99 -.98 -.98 -.83 -.79 -.72 -.72
175 -14. -1.333 -1.333 -1.22 -1.096 -1. -.97 -.97 -.84 -.805 -.73 -.73
176 -13. -1.334 -1.334 -1.28 -1.12 -1. -.96 -.96 -.85 -.815 -.735 -.735
177 -12. -1.255 -1.255 -1.26 -1.13 -1. -.947 -.94 -.85 -.82 -.74 -.74
178 -11. -1.161 -1.161 -1.19 -1.12 -.994 -.93 -.923 -.85 -.81 -.74 -.74
179 -10. -1.055 -1.055 -1.01 -1.082 -.985 -.91 -.90 -.845 -.805 -.73 -.73
180 -8. -.844 -.844 -.88 -.907 -.922 -.87 -.84 -.82 -.77 -.695 -.695
181 -6. -.633 -.633 -.66 -.684 -.741 -.77 -.75 -.77 -.72 -.593 -.593
182 -4. -.422 -.422 -.440 -.456 -.494 -.544 -.578 -.627 -.603 -.396 -.396
183 -2. -.211 -.211 -.22 -.228 -.247 -.272 -.313 -.350 -.395 -.2 -.2
184 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.
185 2. .211 .211 .22 .228 .247 .272 .313 .350 .395 .2 .2
186 4. .422 .422 .44 .456 .494 .544 .578 .627 .603 .396 .396
187 6. .633 .633 .66 .684 .741 .77 .75 .77 .72 .593 .593
188 8. .844 .844 .88 .907 .922 .87 .84 .82 .77 .695 .695
189 10. 1.055 1.055 1.1 1.082 .985 .91 .90 .845 .805 .73 .73
190 11. 1.161 1.161 1.19 1.12 .994 .93 .923 .850 .810 .74 .74
191 12. 1.255 1.255 1.26 1.13 1. .947 .94 .85 .82 .74 .74
192 13. 1.334 1.334 1.28 1.12 1. .96 .96 .85 .815 .735 .735
193 14. 1.333 1.333 1.22 1.096 1. .97 .97 .84 .805 .73 .73
194 15. 1.19 1.19 1.09 1.055 .99 .98 .98 .83 .79 .73 .73
195 16.5 1.007 1.007 .944 .96 .965 .965 .965 .795 .76 .7 .7
196 21. .8 .8 .81 .83 .85 .85 .85 .71 .68 .64 .64
197 39. 1.18 1.18 1.18 1.18 1.18 1.18 1.18 1.18 1.18 1.18 1.18
198 49. 1.18 1.18 1.18 1.18 1.18 1.18 1.18 1.18 1.18 1.18 1.18

```

CHARM horizontal tail airfoil input file for the Side-by-Side. Filename is 'htailaf.inp.'

```

199 129. -1. -1. -1. -1. -1. -1. -1. -1. -1. -1. -1.
200 147. -1. -1. -1. -1. -1. -1. -1. -1. -1. -1. -1.
201 161. -.62 -.62 -.62 -.62 -.62 -.62 -.62 -.62 -.62 -.62 -.62
202 172.5 -.78 -.78 -.78 -.78 -.78 -.78 -.78 -.78 -.78 -.78 -.78
203 180. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.
204 0. .18 .28 .38 .48 .62 .72 .77 .82 .92 1.0
205 -180. .022 .022 .022 .022 .022 .022 .022 .022 .022 .022 .022
206 -175. .062 .062 .062 .062 .062 .062 .062 .062 .062 .062 .062
207 -170. .132 .132 .132 .132 .132 .132 .132 .132 .132 .132 .132
208 -165. .242 .242 .242 .242 .242 .242 .242 .242 .242 .242 .242
209 -160. .302 .302 .302 .302 .302 .302 .302 .302 .302 .302 .302
210 -140. 1.042 1.042 1.042 1.042 1.042 1.042 1.042 1.042 1.042 1.042 1.042
211 -120. 1.652 1.652 1.652 1.652 1.652 1.652 1.652 1.652 1.652 1.652 1.652
212 -110. 1.852 1.852 1.852 1.852 1.852 1.852 1.852 1.852 1.852 1.852 1.852
213 -100. 2.022 2.022 2.022 2.022 2.022 2.022 2.022 2.022 2.022 2.022 2.022
214 -90. 2.022 2.022 2.022 2.022 2.022 2.022 2.022 2.022 2.022 2.022 2.022
215 -80. 1.962 1.962 1.962 1.962 1.962 1.962 1.962 1.962 1.962 1.962 1.962
216 -70. 1.842 1.842 1.842 1.842 1.842 1.842 1.842 1.842 1.842 1.842 1.842
217 -60. 1.662 1.662 1.662 1.662 1.662 1.662 1.662 1.662 1.662 1.662 1.662
218 -50. 1.392 1.392 1.392 1.392 1.392 1.399 1.392 1.392 1.392 1.392 1.392
219 -30. .562 .562 .562 .562 .562 .562 .562 .562 .562 .562 .562
220 -21. .332 .332 .332 .332 .332 .332 .332 .332 .332 .342 .342
221 -16. .155 .155 .181 .207 .235 .257 .274 .292 .305 .342 .342
222 -15. .102 .102 .148 .181 .209 .233 .252 .271 .282 .298 .298
223 -14. .038 .038 .099 .146 .180 .212 .233 .249 .260 .293 .293
224 -13. .0264 .0264 .0455 .094 .148 .191 .216 .231 .239 .272 .292
225 -12. .022 .022 .030 .06 .111 .164 .198 .211 .220 .252 .291
226 -11. .0196 .0196 .0232 .038 .078 .135 .17 .192 .202 .232 .275
227 -10. .0174 .0174 .0189 .0259 .053 .105 .145 .176 .186 .213 .254
228 -9. .0154 .0154 .0159 .0187 .0351 .077 .122 .159 .172 .199 .232
229 -8. .0138 .0138 .0138 .0147 .0220 .053 .101 .140 .155 .183 .214
230 -7. .0122 .0122 .0122 .0123 .0141 .035 .082 .111 .139 .169 .192
231 -6. .011 .011 .011 .011 .011 .0212 .0615 .082 .12 .14 .17
232 -5. .01 .01 .01 .01 .01 .0132 .038 .054 .088 .111 .14
233 -4. .0093 .0093 .0093 .0093 .0093 .01 .0167 .03 .0575 .095 .112
234 -3. .0088 .0088 .0088 .0088 .0088 .009 .0102 .0175 .0355 .086 .102
235 -2. .0085 .0085 .0085 .0085 .0085 .0085 .0086 .0117 .0240 .081 .098
236 -1. .0083 .0083 .0083 .0083 .0083 .0083 .0083 .0091 .0175 .078 .096
237 0. .008 .008 .008 .008 .008 .008 .008 .008 .0137 .078 .095
238 1. .0083 .0083 .0083 .0083 .0083 .0083 .0083 .0091 .0175 .078 .096
239 2. .0085 .0085 .0085 .0085 .0085 .0085 .0086 .0117 .024 .081 .098
240 3. .0088 .0088 .0088 .0088 .0088 .0090 .0102 .0175 .0355 .086 .102
241 4. .0093 .0093 .0093 .0093 .0093 .01 .0167 .03 .0575 .095 .112
242 5. .01 .01 .01 .01 .01 .0132 .038 .054 .088 .111 .14
243 6. .011 .011 .011 .011 .011 .0212 .0615 .082 .12 .14 .17
244 7. .0122 .0122 .0122 .0123 .0141 .035 .082 .111 .139 .169 .192
245 8. .0138 .0138 .0138 .0147 .022 .053 .101 .14 .155 .183 .214
246 9. .0154 .0154 .0159 .0187 .0351 .077 .122 .159 .172 .199 .232
247 10. .0174 .0174 .0189 .0259 .053 .105 .145 .176 .186 .213 .254
248 11. .0196 .0196 .0232 .038 .078 .135 .17 .192 .202 .232 .275
249 12. .022 .022 .03 .06 .111 .164 .198 .211 .22 .252 .291
250 13. .0264 .0264 .0455 .094 .148 .191 .216 .231 .239 .272 .292
251 14. .038 .038 .099 .146 .18 .212 .233 .249 .26 .293 .293
252 15. .102 .102 .148 .181 .209 .233 .252 .271 .282 .298 .298
253 16. .155 .155 .181 .207 .235 .257 .274 .292 .305 .342 .342
254 21. .332 .332 .332 .332 .332 .332 .332 .332 .332 .342 .342
255 30. .562 .562 .562 .562 .562 .562 .562 .562 .562 .562 .562
256 50. 1.392 1.392 1.392 1.392 1.392 1.392 1.392 1.392 1.392 1.392 1.392
257 60. 1.662 1.662 1.662 1.662 1.662 1.662 1.662 1.662 1.662 1.662 1.662
258 70. 1.842 1.842 1.842 1.842 1.842 1.842 1.842 1.842 1.842 1.842 1.842
259 80. 1.962 1.962 1.962 1.962 1.962 1.962 1.962 1.962 1.962 1.962 1.962
260 90. 2.022 2.022 2.022 2.022 2.022 2.022 2.022 2.022 2.022 2.022 2.022
261 100. 2.022 2.022 2.022 2.022 2.022 2.022 2.022 2.022 2.022 2.022 2.022
262 110. 1.852 1.852 1.852 1.852 1.852 1.852 1.852 1.852 1.852 1.852 1.852
263 120. 1.652 1.652 1.652 1.652 1.652 1.652 1.652 1.652 1.652 1.652 1.652
264 140. 1.042 1.042 1.042 1.042 1.042 1.042 1.042 1.042 1.042 1.042 1.042

```

CHARM horizontal tail airfoil input file for the Side-by-Side. Filename is 'htailaf.inp.'

```

265 160. .302 .302 .302 .302 .302 .302 .302 .302 .302 .302
266 165. .242 .242 .242 .242 .242 .242 .242 .242 .242 .242
267 170. .132 .132 .132 .132 .132 .132 .132 .132 .132 .132
268 175. .062 .062 .062 .062 .062 .062 .062 .062 .062 .062
269 180. .022 .022 .022 .022 .022 .022 .022 .022 .022 .022
270 .00 .30 .40 .50 .6 .7 .75 .8 .9 1.
271 -180. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.
272 -170. .4 .4 .4 .4 .4 .4 .4 .4 .4 .4
273 -165. .3 .3 .3 .3 .3 .3 .3 .3 .3 .3
274 -160. .3 .3 .3 .3 .3 .3 .3 .3 .3 .3
275 -135. .5 .5 .5 .5 .5 .5 .5 .5 .5 .5
276 -90. .5 .5 .5 .5 .5 .5 .5 .5 .5 .5
277 -30. .174 .184 .196 .214 .235 .25 .264 .277 .298 .319
278 -23. .112 .118 .128 .144 .157 .171 .183 .206 .232 .258
279 -16. .073 .078 .086 .097 .108 .117 .137 .176 .200 .224
280 -15. .054 .065 .073 .084 .097 .111 .133 .173 .195 .217
281 -14. 0. .027 .054 .068 .086 .103 .127 .167 .189 .211
282 -13. 0. .0015 .025 .05 .074 .093 .122 .163 .184 .205
283 -12. 0. 0. .002 .03 .06 .083 .116 .157 .176 .195
284 -11. 0. 0. -.003 .014 .046 .074 .108 .149 .17 .191
285 -10. 0. 0. -.0015 .002 .032 .065 .10 .142 .163 .184
286 -9. 0. 0. 0. -.003 .016 .054 .089 .132 .154 .176
287 -8. 0. 0. 0. -.004 .005 .041 .082 .123 .145 .167
288 -7. 0. 0. 0. 0. -.004 .0275 .072 .1125 .136 .160
289 -6. 0. 0. 0. 0. -.003 .016 .0625 .10 .125 .150
290 -4. 0. 0. 0. 0. 0. .005 .04 .076 .102 .128
291 -3. 0. 0. 0. 0. 0. -.0025 .026 .0665 .087 .108
292 -2. 0. 0. 0. 0. 0. 0. .013 .053 .07 .087
293 -1. 0. 0. 0. 0. 0. 0. .0035 .033 .045 .057
294 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.
295 1. 0. 0. 0. 0. 0. 0. -.0035 -.033 -.045 -.057
296 2. 0. 0. 0. 0. 0. 0. -.013 -.053 -.07 -.087
297 3. 0. 0. 0. 0. 0. .0025 -.026 -.0665 -.087 -.108
298 4. 0. 0. 0. 0. 0. -.005 -.04 -.076 -.102 -.128
299 6. 0. 0. 0. 0. 0. .003 -.016 -.0625 -.1 -.125 -.150
300 7. 0. 0. 0. 0. .004 -.0275 -.072 -.1125 -.136 -.160
301 8. 0. 0. 0. .004 -.005 -.041 -.082 -.123 -.145 -.167
302 9. 0. 0. 0. .003 -.016 -.054 -.089 -.132 -.154 -.176
303 10. 0. 0. .0015 -.002 -.032 -.065 -.1 -.142 -.163 -.184
304 11. 0. 0. .003 -.014 -.046 -.074 -.108 -.149 -.17 -.191
305 12. 0. 0. -.002 -.03 -.06 -.083 -.116 -.157 -.176 -.195
306 13. 0. -.0015 -.025 -.05 -.074 -.093 -.122 -.163 -.184 -.205
307 14. 0. -.027 -.054 -.068 -.086 -.103 -.127 -.167 -.189 -.211
308 15. -.054 -.065 -.073 -.084 -.097 -.111 -.133 -.173 -.195 -.217
309 16. -.073 -.078 -.086 -.097 -.108 -.117 -.137 -.176 -.20 -.224
310 23. -.112 -.118 -.128 -.144 -.157 -.171 -.183 -.206 -.232 -.258
311 30. -.174 -.184 -.196 -.214 -.235 -.250 -.264 -.277 -.298 -.319
312 90. -.5 -.5 -.5 -.5 -.5 -.5 -.5 -.5 -.5 -.5
313 135. -.5 -.5 -.5 -.5 -.5 -.5 -.5 -.5 -.5 -.5
314 160. -.3 -.3 -.3 -.3 -.3 -.3 -.3 -.3 -.3 -.3
315 165. -.3 -.3 -.3 -.3 -.3 -.3 -.3 -.3 -.3 -.3
316 170. -.4 -.4 -.4 -.4 -.4 -.4 -.4 -.4 -.4 -.4
317 180. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.
318

```

CHARM left horizontal tail rotor wake input file for the Side-by-Side. Filename is 'lefthtailrw.inp.'

```

1  lefthtailrw.inp: Rotor Wake Input File for left horizontal tail
2  NBLADE  OMEGA
3    1      0.0000
4  IROTAT      XROTOR      X,Y,Z tilt      ITILT
5    -1    -18.222    0.0    1.128    -30.0    0.0    90.0      1
6  ICOLL    COLL      CT      for rotor 1
7    0      0.0      .002
8  ITRIM      A1W      B1W      A1S      B1S
9    0      0.0      0.0      0.0      0.0
10 NOWAKE    ICNVCT    NWAKES    NPWAKE    IFAR      MBCVE
11    0      2      1      0      0      0
12 KSCHEME    KPC
13    0      0
14 NCUT      AOV LAP      ISKEW      IUNS      RMVWAKE
15    1      -1      1      0      0.05
16 NZONE      (NVORT(I), I=1,NZONE)  BUFFER    KGMCHG    KPERIODIC
17    -4      20  20  20  20      2.0      0      1
18 NZFSW
19    4
20 (NPTFW(I), I=1,NZONE)  DTWAKE_GAIN
21    24  24  24  24      10.0
22 (CORLIM(NV,IZONE,1), NV=1,NVORT(IZONE)  IZONE=1 (Min core radii)
23    0.01
24 (CORLIM(NV,IZONE,2), NV=1,NVORT(IZONE)  IZONE=1 (Max core radii)
25    1.0
26 (CORLIM(NV,IZONE,1), NV=1,NVORT(IZONE)  IZONE=2 (Min core radii)
27    0.01
28 (CORLIM(NV,IZONE,2), NV=1,NVORT(IZONE)  IZONE=2 (Max core radii)
29    1.0
30 (CORLIM(NV,IZONE,1), NV=1,NVORT(IZONE)  IZONE=3 (Min core radii)
31    0.01
32 (CORLIM(NV,IZONE,2), NV=1,NVORT(IZONE)  IZONE=3 (Max core radii)
33    1.0
34 (CORLIM(NV,IZONE,1), NV=1,NVORT(IZONE)  IZONE=4 (Min core radii)
35    0.01
36 (CORLIM(NV,IZONE,2), NV=1,NVORT(IZONE)  IZONE=4 (Max core radii)
37    1.0
38 (CUTLIM(NV,IZONE,1), NV=1,NVORT(IZONE)  IZONE=1 (Min cutoff distances)
39    0.01
40 (CUTLIM(NV,IZONE,2), NV=1,NVORT(IZONE)  IZONE=1 (Max cutoff distances)
41    1.0
42 (CUTLIM(NV,IZONE,1), NV=1,NVORT(IZONE)  IZONE=2 (Min cutoff distances)
43    0.01
44 (CUTLIM(NV,IZONE,2), NV=1,NVORT(IZONE)  IZONE=2 (Max cutoff distances)
45    1.0
46 (CUTLIM(NV,IZONE,1), NV=1,NVORT(IZONE)  IZONE=3 (Min cutoff distances)
47    0.01
48 (CUTLIM(NV,IZONE,2), NV=1,NVORT(IZONE)  IZONE=3 (Max cutoff distances)
49    1.0
50 (CUTLIM(NV,IZONE,1), NV=1,NVORT(IZONE)  IZONE=4 (Min cutoff distances)
51    0.01
52 (CUTLIM(NV,IZONE,2), NV=1,NVORT(IZONE)  IZONE=4 (Max cutoff distances)
53    1.0
54 IDYNM
55    1
56 SRAD      SHGHT
57    0.0      0.0
58 NHHI (Higher harmonic cyclic pitch input flag)
59    0
60

```

CHARM right horizontal tail rotor wake input file for the Side-by-Side. Filename is 'righttailrw.inp.'

```

1  righttailrw.inp: Rotor Wake Input File for right horizontal tail
2  NBLADE  OMEGA
3      1      0.0000
4  IROTAT      XROTOR      X,Y,Z tilt      ITILT
5      1      -18.222  0.0  1.128  30.0  0.0  90.0      1
6  ICOLL  COLL  CT      for rotor 1
7      0      0.0  .002
8  ITRIM      A1W  B1W  A1S  B1S
9      0      0.0  0.0  0.0  0.0
10 NOWAKE  ICNVCT  NWAKES  NPWAKE  IFAR  MBCVE
11      0      2      1      0      0      0
12 KSCHEME  KPC
13      0      0
14 NCUT  AOV LAP  ISKEW  IUNS  RMVWAKE
15      1      -1      1      0      0.05
16 NZONE  (NVORT(I), I=1,NZONE)  BUFFER  KGMCHG  KPERIODIC
17      -4      20  20  20  20      2.0      0      1
18 NZFSW
19      4
20 (NPTFW(I), I=1,NZONE)  DTWAKE_GAIN
21      24  24  24  24      10.0
22 (CORLIM(NV,IZONE,1), NV=1,NVORT(IZONE)  IZONE=1 (Min core radii)
23      0.01
24 (CORLIM(NV,IZONE,2), NV=1,NVORT(IZONE)  IZONE=1 (Max core radii)
25      1.0
26 (CORLIM(NV,IZONE,1), NV=1,NVORT(IZONE)  IZONE=2 (Min core radii)
27      0.01
28 (CORLIM(NV,IZONE,2), NV=1,NVORT(IZONE)  IZONE=2 (Max core radii)
29      1.0
30 (CORLIM(NV,IZONE,1), NV=1,NVORT(IZONE)  IZONE=3 (Min core radii)
31      0.01
32 (CORLIM(NV,IZONE,2), NV=1,NVORT(IZONE)  IZONE=3 (Max core radii)
33      1.0
34 (CORLIM(NV,IZONE,1), NV=1,NVORT(IZONE)  IZONE=4 (Min core radii)
35      0.01
36 (CORLIM(NV,IZONE,2), NV=1,NVORT(IZONE)  IZONE=4 (Max core radii)
37      1.0
38 (CUTLIM(NV,IZONE,1), NV=1,NVORT(IZONE)  IZONE=1 (Min cutoff distances)
39      0.01
40 (CUTLIM(NV,IZONE,2), NV=1,NVORT(IZONE)  IZONE=1 (Max cutoff distances)
41      1.0
42 (CUTLIM(NV,IZONE,1), NV=1,NVORT(IZONE)  IZONE=2 (Min cutoff distances)
43      0.01
44 (CUTLIM(NV,IZONE,2), NV=1,NVORT(IZONE)  IZONE=2 (Max cutoff distances)
45      1.0
46 (CUTLIM(NV,IZONE,1), NV=1,NVORT(IZONE)  IZONE=3 (Min cutoff distances)
47      0.01
48 (CUTLIM(NV,IZONE,2), NV=1,NVORT(IZONE)  IZONE=3 (Max cutoff distances)
49      1.0
50 (CUTLIM(NV,IZONE,1), NV=1,NVORT(IZONE)  IZONE=4 (Min cutoff distances)
51      0.01
52 (CUTLIM(NV,IZONE,2), NV=1,NVORT(IZONE)  IZONE=4 (Max cutoff distances)
53      1.0
54 IDYNM
55      1
56 SRAD  SHGHT
57      0.0  0.0
58 NHHI (Higher harmonic cyclic pitch input flag)
59      0
60

```

CHARM vehicle trim input file for the Side-by-Side. Filename is 'sbstrim.inp.'

```

1  sbcruisetrim.inp: Control mixer for SBS aircraft
2  Fx    Fy    Fz    Mx    My    Mz                Aircraft forces and moments to trim
3  1      0      1      0      1      0
4  Fx    Fy    Fz    Mx    My    Mz                Value to trim to
5  0.0    0.0    0.0    0.0    0.0    0.0
6  Fx    Fy    Fz    Mx    My    Mz                Trim tolerance (fraction of weight)
7  .0002 .0002 .0002 .0002 .0002 .0002
8  Coll  Lat   Long  Pedal  Throttle  Tilt  Flap1  Flap2  Elev  Rudder  Addl      Pilot
   controls
9  1      0      0      0      1      0      0      0      0      0      0
10 Roll   Pitch  Sideslip                Attitude controls used to trim
11 0      1      0
12 Number of aerodynamic components (rotors and wings) used to trim
13 4
14 Comp  RPM   Coll   Lat   Long   Tilt   Flaps(4)                Controls used by each
   aero component
15 1      0      1      0      2      0      0  0  0  0      Aero Component 1:
   Right rotor
16 2      0      3      0      4      0      0  0  0  0      Aero Component 2:
   Left rotor
17 3      0      0      0      0      0      5  0  0  0      Aero Component 5:
   Right horizontal tail
18 4      0      0      0      0      0      6  0  0  0      Aero Component 6:
   Left horizontal tail
19 Ctrl Zero  Coll   Lat   Long   Pedal  Throttle  Tilt  Flap1  Flap2  Elev  Rudder  Addl
   Mixer: Component/Pilot control gains
20 1      0.0    1.0   -1.0    0      0      0      0      0      0      0      0
   Rotor 1 Coll for Fz; Diff Coll for Mx
21 2      0.0    0      0      0      0      1.0    0      0      0      0      0
   Rotor 1 Long for Fx; Diff Long for Mz in hover
22 3      0.0    1.0    1.0    0      0      0      0      0      0      0      0
   Rotor 2 Coll for Fz; Diff Coll for Mx
23 4      0.0    0      0      0      0      1.0    0      0      0      0      0
   Rotor 2 Long for Fx; Diff Long for Mz in hover
24 5      0.0    0      0   -1.0   -1.0    0      0      0      0      0      0
   Right horizontal tail: My ; Diff for Mz in cruise
25 6      0.0    0      0   -1.0    1.0    0      0      0      0      0
   0 Left horizontal tail My : Diff for Mz in cruise
26 Initial Pilot Control Settings Based on the Gains Above
27                0.0    0.0    0.0    0.0    0.0    0.0    0.0    0.0    0.0    0.0

```

CHARM fuselage input file for the Side-by-Side. Filename is 'fusedata_table.inp.'

```

1  Default fuselage aerodynamic coefficients - Table 4B.4 from Padfield
2  9   3      ! X-force, body frame
3  2   1      ! Y-force, body frame
4 11   1      ! Z-force, body frame
5  2   1      ! X (roll) moment, body frame
6 10   1      ! Y (pitch) moment, body frame
7  1   7      ! Z (yaw) moment, body frame
8
9      -90.   0.   90.
10 -180.   .1   .1   .1
11 -160.   .08  .08  .08
12 -90.    0.   0.   0.
13 -30.   -.07 -.07 -.07
14  0.    -.08 -.08 -.08
15 20.    -.07 -.07 -.07
16 90.    0.   0.   0.
17 160.   .08  .08  .08
18 180.   .10  .10  .10
19 -180.   180.
20 0.00   0.00
21 -180. -160. -120. -60. -20.  0.   20.   60.  120.  160.  180.
22 0.00   0.15  1.30  1.30  0.15  0.00 -0.15 -1.30 -1.30 -0.15  0.0
23 -180.   180.
24 0.00   0.00
25 -205. -160. -130. -60. -25.  25.   60.  130.  155.  200.
26 0.02 -0.03  0.10  0.10 -0.04  0.02 -0.10 -0.10  0.02 -0.03
27 -90.  -70.  -25.   0.0  25.   70.   90.
28 -10. -0.10  0.005  0.0 -0.005  0.10  0.10

```

A.4 DJI Phantom 3

The following files are the CAMRAD II and CHARM input files for the DJI Phantom 3 rotor. These models are fairly straightforward but do show how to integrate more complicated airfoil tables into a CAMRAD II or CHARM run. The filenames and structure are as follows.

- CAMRADII_DJI
 - camradii
 - * **dji.run**
 - camradii-inputs
 - * airfoil
 - **helicopter_airfoil.run**
 - **Ph3_AF12.c81**
 - **Ph3_AF12.run**
 - * inputs
 - **rotor_input.run**
 - **rotor.list**
 - **other-settings-for-noise.list**

For CHARM, the input files required are as listed below. Once again, this is a fairly straightforward model that shows how to integrate more complicated airfoil geometry.

- **dji.inp**
- **dji_bg.inp**
- **dji_bd.inp**
- **dji_rw.inp**
- **dji_af.inp**

CAMRAD II run file for the DJI Phantom 3. Filename is 'dji.run.'

```

1  #!/bin/csh
2
3  set camradii=${HOME}/bin/camradii-v5.0-dynamic
4
5  rm -f dji.plot
6  rm -f dji.out
7
8  setenv CAMRAD_SIZER 2000000000
9  setenv CAMRAD_SIZEI 1000000000
10 setenv CAMRAD_SIZEEC 500000000
11 setenv CORE_SIZER 2000000
12 setenv CORE_SIZEI 5000000
13 setenv CORE_SIZEEC 50000000
14 setenv TABLE_SIZER 1000000
15 setenv TABLE_SIZEI 100000
16 setenv TABLE_SIZEEC 100000
17 setenv SHELL_SIZER 400000
18 setenv SHELL_SIZEI 100000
19 setenv SHELL_SIZEEC 100000
20
21 setenv BLADEAIRFOIL1 ../camradii-inputs/airfoil/helicopter.tab
22 setenv SHELLINPUT ../camradii-inputs/inputs/rotor.bin
23 setenv PLOTFILE dji.plot
24
25 ${camradii} > dji.out << 'endofinput'
26 &NLJOB
27     NCASES = 1,
28     OPINIT = 0,
29     OUTRES = 1,
30     PLFILE = 0,
31 &END
32 !=====
33 &NLDEF class='CASE', &END
34 &NLVAL
35     TMTASK = 1,
36     FLTASK = 2,
37     OPDENS = 1,
38 &END
39 !=====
40 &NLDEF class='TRIM', &END
41 &NLVAL
42     NGFILE = 0,
43     WINDIN = 1,
44     WKTS = 0.0,
45     VTIPIN = 2,
46     RPM = 4000.,
47     OPTRIM = 0,
48     LEVEL = 3, ITERF = 5, RELAXF = 0.5,
49     MTRIM = 1, MNAME = 'THRUST', VNAME = 'COLL',
50     COLL = 7,
51     MHARMR = 1,
52     MPSI = 24,
53     OPOST = 1,
54     MPSIH = 240,
55     ITERC = 200,
56 &END
57 !=====
58 &NLDEF class='TRIM ROTOR', name='ROTOR 1', &END
59 &NLVAL
60     MPSEN = 1, MPTIME = 240,
61     MASEN = 1, MATIME = 240,
62     MWSEN = 1, NWPRNT = 0,
63     NWFILE = 1,

```

CAMRAD II run file for the DJI Phantom 3. Filename is 'dji.run.'

```

64      DOFB      = 2*0, 1, 9*0,
65      &END
66      !=====
67      &NLDEF class='FLUTTER', &END
68      &NLVAL &END
69      &NLDEF class='FLUTTER ROTOR', name='ROTOR 1', &END
70      &NLVAL &END
71      !=====
72      &NLDEF class='ROTOR', type='INFLOW', name='ROTOR 1', &END
73      &NLVAL
74      OPFFLI = 0,
75      &END
76      !=====
77      &NLDEF class='ROTOR', type='STRUCTURE', name='ROTOR 1', &END
78      &NLVAL
79      OPSND = 1,
80      &END
81      !=====
82      &NLDEF class='ROTOR', type='AERODYNAMICS', name='ROTOR 1', &END
83      &NLVAL
84      NSEN      = 7,
85      QUANT      = 5, 5, 5, 53, 81, 81, 64,
86      IDENT      = 0, 0, 0, 1, 1, 1, 1,
87      AXIS      = 1, 2, 3, 0, 1, 3, 0,
88      OPSC      = 0, 0, 0, 1, 2, 2, 2,
89      OPREF      = 1, 1, 1, 1, 2, 2, 2,
90      NPANEL     = 19,
91      REDGE      = 0.20, 0.28, 0.35, 0.42, 0.48, 0.54, 0.59, 0.64, 0.69, 0.73, 0.77, 0.81,
92                  0.84, 0.87, 0.90, 0.92, 0.94, 0.96, 0.98, 1.00,
93      &END
94      !=====
95      ! hover wake scenario
96      &NLDEF class='ROTOR', type='WAKE', name='ROTOR 1', &END
97      &NLVAL OPSCEN = 2, &END
98      &NLDEF class='ROTOR', type='WAKE', name='ROTOR 1', &END
99      &NLVAL
100      OPSCEN = 0,
101      OPFWG = 3,
102      OPTVIC = 0,
103      OPICWG = 1,
104      RICWG = 0.395325,
105      TWIST = -5.963,
106      OPRTV = 0,
107      CORE = 0.5,
108      COREWG = 0.5,
109      OPNW = 0,
110      OPDISP = 0,0,
111      &END
112      !=====
113      &NLDEF action='end of shell', &END
114      &NLDEF action='end of core', &END
115      'endofinput'
116      exit
117

```

CAMRAD II input run file for the DJI Phantom 3. Filename is 'rotor_input.run.'

```
1  #!/bin/csh
2  #set camrad='~/Macintosh/camrad50'
3  set camrad='/home/ddboyd/bin/inputii-v5.0-dynamic'
4
5  rm -f rotor.bin
6  rm -f rotor_input.out
7
8  #
9  setenv INPUTLIST1 rotor.list
10 setenv INPUTLIST2 other-settings-for-noise.list
11 setenv OUTPUTFILE rotor.bin
12
13 $camrad > rotor_input.out << 'endofinput'
14 BATCH
15 &NLJOB
16         NLISTI = 2,
17         NLISTC = 0,
18 &END
19 'endofinput'
20
21 exit
```

CAMRAD II input aircraft list file for the DJI Phantom 3. Filename is 'rotor.list.'

```

1  !=====
2  ! DJI Phantom single rotor
3  !=====
4  !
5  ! ddb: 07/03/2024:
6  !     - assume this is a hover case           (this is a hover case)
7  !     - removed all tabs from the file.        (readability)
8  !     - added "," to THETAC, THETAI, TWISTA entries (readability, transportability)
9  !     - aligned text for readability          (readability)
10 !     - MPSI = 24, MPSIWG = 24                 (SOP... could use 36)
11 !     - MPSIH = 240, MxTIME = 240              (in
other-settings-for-noise.list, 10x MPSI as SOP)
12 !     - ITERF = 5, RELAXF = 0.5                (hover free wake)
13 !     - SIGMA = 0.10067                        (based on CAMRAD II calculation
of geometry solidity)
14 !     - MASS = 9.5                             (to get LOCK number ~5.)
15 !     - MASSR = 0.17987                        (based on CAMRAD II calculation
rotor mass)
16 !     - RPM = 5100.                             (based on prior work)
17 !     - ITERC = 200                             (need more iterations than the
default)
18 !     - VTIPN = 63.76                          (based on RPM = 5100.)
19 !     - CONTRL = 0,                             (no swashplate)
20 !     - PITCH = 2, LOCKP = 1, KPITCH = 1.0e+10, (PB with control so that pitch
trim can be done if desired)
21 !     - COLL = 0.0,                             (coll = 0.0 because twist
includes the blade pitch at 0.75R)
22 !     - PITCH = 0.0,                             (shaft tilt = 0.0 for hover)
23 !     - KNODE = 3
24 !     - RNODE = 0.15860, .28080, 0.7           (corners in the twist and chord,
plus 0.7 to break up longer seg)
25 !
26 !=====
27 &NLDEF class='CASE',&END
28 &NLVAL
29     TITLE = 'DJI Phantom single rotor',
30     OPUNIT = 2,
31 &END
32 !=====
33 &NLDEF class='TRIM',&END
34 &NLVAL
35     WINDIN = 2, WVCL = 0.0,
36     VTIPIN = 2, RPM = 5100.,
37     LEVEL=2*1,                                     ! wake loop
38     COLL=0.,                                       ! wind tunnel trim
39     MTRIM=1,
40     MNAME='CT/S', VNAME='COLL',
41     ITERF = 5,
42     RELAXT=.5,                                     ! trim loop, Change if you
have convergence factors
43     OPPART=3,                                     ! teetering rotor
44     MHARMR=1, MHARMA=0, MPSIAV=1,                 ! harmonics
45     DOFA=6*0, DOFM=2*0,                           ! degrees of freedom
46     MPSI=36,
47 &END
48 !=====
49 &NLDEF class='TRIM ROTOR',name='ROTOR 1',&END
50 &NLVAL
51     NAPRNT=1, NAFILE=1, MATIME=72,                ! output
52     OPMODE=0, DOFG=1, DOFB=1, 0, 1, 9*0,          ! degrees of freedom
53 &END

```

CAMRAD II input aircraft list file for the DJI Phantom 3. Filename is 'rotor.list.'

```

54  !=====
55  &NLDEF class='FLUTTER',&END
56  &NLVAL
57      MPSIAV=1,                                ! no average
58      DOFA=6*0,DOFM=2*1,DOFD=8*0,              ! degrees of freedom
59  &END
60  !=====
61  &NLDEF class='FLUTTER ROTOR',name='ROTOR 1',&END
62  &NLVAL
63      OPWAKE=4,OPVATR=2,OPVRTA=2,                ! trim inflow
64      OPMODE=0,DOFG=1,DOFB=1,0,1,9*0,DOFL=3*0,    ! degrees of freedom
65  &END
66  !=====
67  &NLDEF class='AIRFRAME',type='STRUCTURE',&END
68  &NLVAL
69      TITLE  ='DJI Phantom single rotor',
70  CONFIG=0,OPFREE=0,OPAERO=0,                    ! wind tunnel
71      MASSR=8.4,                                ! inertia
72      NMODE=2,                                   ! test module
73      ASHAFT=0
74      QMASS=2*1000.,QFREQ=2.,4.,QDAMP=2*.1,
75      LSHAPE(1,1,1) = 1., 0., 0.,
76      LSHAPE(1,1,2) = 0., 1., 0.,
77      ASHAPE(1,1,1) = 0., -1.5, 0.,
78      ASHAPE(1,1,2) = 1.5, 0., 0.,
79  &END
80  !=====
81  &NLDEF class='AIRFRAME',type='AERODYNAMICS',&END
82  &NLVAL &END
83  !=====
84  &NLDEF class='AIRFRAME',type='CONTROL',&END
85  &NLVAL &END
86  !=====
87  &NLDEF class='ROTOR',type='STRUCTURE',name='ROTOR 1',&END
88  &NLVAL
89      TITLE  ='DJI Phantom rotor 1',
90      RADIUS = 0.11938,
91      NBLADE = 2,
92      ROTATE = 1,
93      SIGMA  = 0.10067,
94      VTIPN  = 63.76,
95      HINGE   = 0,
96      OPBEAM  = 1,
97
98      CONTRL  = 0,
99      PITCH   = 2,  LOCKP = 1,  KPITCH = 1.0e+10,
100
101      KNODE   = 3,
102      RNODE   = 0.15860, 0.28080, 0.70000,
103
104      TWIN     = 0,
105      NPROP    = 21,
106      RPROP    = 0.0000, 0.1586, 0.2076, 0.2566, 0.2808, 0.3056, 0.3546, 0.4036,
107                0.4526, 0.5016, 0.5506, 0.5996, 0.6486, 0.6976, 0.7466, 0.7956, 0.8446,
108                0.8936, 0.9426, 0.9916, 1.0000,
109      TWISTA   = 15.5764, 15.5764, 18.0597, 20.1300, 20.4846, 19.8223, 18.2339, 17.2376,
110                16.0526, 14.9715, 14.4197, 13.7872, 13.1839, 12.5631, 12.0373, 11.5345, 11.0596,
111                10.6164, 10.0165, 9.6125, 9.6125,
112      THETAC   = 15.5764, 15.5764, 18.0597, 20.1300, 20.4846, 19.8223, 18.2339, 17.2376,
113                16.0526, 14.9715, 14.4197, 13.7872, 13.1839, 12.5631, 12.0373, 11.5345, 11.0596,

```

CAMRAD II input aircraft list file for the DJI Phantom 3. Filename is 'rotor.list.'

```

109      10.6164, 10.0165, 9.6125, 9.6125,
      THETA1 = 15.5764, 15.5764, 18.0597, 20.1300, 20.4846, 19.8223, 18.2339, 17.2376,
      16.0526, 14.9715, 14.4197, 13.7872, 13.1839, 12.5631, 12.0373, 11.5345, 11.0596,
      10.6164, 10.0165, 9.6125, 9.6125,
110      MASS = 9.5,
111  &END
112  !=====
      =====
113  &NLDEF class='ROTOR',type='AERODYNAMICS',name='ROTOR 1',&END
114  &NLVAL
115      NPANEL = 63,
116      REDGE = 0.330625, 0.341250, 0.351875, 0.362500, 0.373125, 0.383750, 0.394375,
      0.405000, 0.415625, 0.426250, 0.436875, 0.447500, 0.458125, 0.468750, 0.479375,
      0.490000, 0.500625, 0.511250, 0.521875, 0.532500, 0.543125, 0.553750, 0.564375,
      0.575000, 0.585625, 0.596250, 0.606875, 0.617500, 0.628125, 0.638750, 0.649375,
      0.660000, 0.670625, 0.681250, 0.691875, 0.702500, 0.713125, 0.723750, 0.734375,
      0.745000, 0.755625, 0.766250, 0.776875, 0.787500, 0.798125, 0.808750, 0.819375,
      0.830000, 0.840625, 0.851250, 0.861875, 0.872500, 0.883125, 0.893750, 0.904375,
      0.915000, 0.925625, 0.936250, 0.946875, 0.957500, 0.968125, 0.978750, 0.989375, 1.0,
117      NPROP = 21,
118      RPROP = 0.000000, 0.158600, 0.207600, 0.256600, 0.280800, 0.305600, 0.354600,
      0.403600, 0.452600, 0.501600, 0.550600, 0.599600, 0.648600, 0.697600, 0.746600,
      0.795600, 0.844600, 0.893600, 0.942600, 0.991600, 1.000000,
119      CHORD = 0.02203704, 0.02203704, 0.02687574, 0.03027172, 0.03096514, 0.03044952,
      0.02842768, 0.02635504, 0.02449830, 0.02290572, 0.02144776, 0.02007362, 0.01886966,
      0.01765300, 0.01645158, 0.01525016, 0.01403350, 0.01298956, 0.01200658, 0.01041400,
      0.010414,
120  &END
121  !=====
      =====
122  &NLDEF class='ROTOR',type='INFLOW',name='ROTOR 1',&END
123  &NLVAL
124  &END
125  !=====
      =====
126  &NLDEF class='ROTOR', type='WAKE',name='ROTOR 1',&END
127  &NLVAL
128      OPSCEN = 2,
129  &END
130  !=====
      =====
131  &NLDEF class='ROTOR', type='WAKE',name='ROTOR 1',&END
132  &NLVAL
133      OPSCEN = 0,
134      OPFWG = 3,
135      CORE(1) = 0.8,
136      COREWG = 0.8,
137      OPDISP = 0,0,
138      TWIST = -10,
139      RICWG = 0.2,
140      MPSIWG = 24,
141  &END
142  !=====
      =====
143  &NLDEF class='TABLES',&END
144  &NLVAL AFTABL=1,&END
145  !=====
      =====
146  &NLDEF action='end of shell',&END
147  &NLDEF action='end of core',&END

```

CAMRAD II noise settings for the DJI Phantom 3. Filename is 'other-settings-for-noise.list.'

```

1  !=====
2  ==
3  ! other settings for noise
4  !=====
5  ==
6  &NLDEF class='TRIM', &END
7  &NLVAL
8      MPSIH = 240,
9      OPPOST = 1,
10     ITERF = 3,
11 &END
12 !=====
13 ==
14 ! ROTOR 1
15 !=====
16 ==
17 &NLDEF class='TRIM ROTOR', name='ROTOR 1', &END
18 &NLVAL
19     MHSEN = 1, NHFILE = 1, MHTIME = 240,
20     MCSEN = 1, NCFILE = 1, MCTIME = 240,
21     MBSEN = 1, NBFILE = 1, MBTIME = 240,
22     MPSEN = 1, NPFILE = 1, MPTIME = 240,
23     MASEN = 1, NAFILE = 1, MATIME = 240,
24 &END
25 !-----
26 &NLDEF class='ROTOR', type='STRUCTURE', name='ROTOR 1', &END
27 &NLVAL
28     OPSND = 1,
29 &END
30 !-----
31 &NLDEF class='ROTOR', type='AERODYNAMICS', name='ROTOR 1', &END
32 &NLVAL
33     NSEN = 7,
34     QUANT = 5, 5, 5, 53, 81, 81, 64,
35     IDENT = 0, 0, 0, 1, 1, 1, 1,
36     AXIS = 1, 2, 3, 0, 1, 3, 0,
37     OPSCL = 0, 0, 0, 1, 2, 2, 2,
38     OPREF = 1, 1, 1, 1, 2, 2, 2,
39 &END
40 !-----
41 &NLDEF class='ROTOR', type='WAKE', name='ROTOR 1', &END
42 &NLVAL
43     OPFWG = 3,
44     CORE(1) = 0.8,
45     COREWG = 0.8,
46     OPTVIC = 0,
47     OPRUDP = 0,
48 &END
49 !=====
50 ==
51 &NLDEF action='end of shell', &END
52 &NLDEF action='end of core', &END
53

```

CAMRAD II airfoil run file for the DJI Phantom 3. Filename is 'helicopter_airfoil.run.'

```
1  #!/bin/csh
2  set camrad='~/Macintosh/camrad50'
3  #
4  setenv INPUTDECK1 Ph3_AF12.c81
5  setenv INPUTDECK2 Ph3_AF19.c81
6  setenv OUTPUTTABLE helicopter.tab
7  $camrad/input_camrad_lm > helicopter_airfoil.out << 'endofinput'
8  BATCH
9  &NLJOB OPFILE=7,OPSRC=1,&END
10 &NLTABL OPFORM=2,RNTRP=0,
11     TITLE='DJI: Two Airfoils'
12     NRB=2,R=.85,1.0,
13 &END
14 'endofinput'
15
```

CAMRAD II airfoil C81 file for the DJI Phantom 3. Filename is 'Ph3_AF12.c81.'

```

1  PHANTOM 3 AF12                                637 653 637
2      0.0800 0.1200 0.1600 0.2000 0.2400 0.2800
3  -180.00 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
4  -172.50 0.7800 0.7800 0.7800 0.7800 0.7800 0.7800
5  -161.00 0.6200 0.6200 0.6200 0.6200 0.6200 0.6200
6  -147.00 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000
7  -129.00 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000
8  -49.00-1.1800-1.1800-1.1800-1.1800-1.1800-1.1800
9  -39.00-1.1800-1.1800-1.1800-1.1800-1.1800-1.1800
10 -21.00-0.8000-0.8000-0.8000-0.8000-0.8000-0.8000
11 -20.00-0.6914-0.4466-0.4452-0.6886-0.6917-0.6957
12 -18.00-0.6219-0.6175-0.4109-0.6261-0.6182-0.6210
13 -16.00-0.5557-0.5491-0.3735-0.5454-0.5504-0.5474
14 -14.00-0.4914-0.4868-0.3465-0.4802-0.4781-0.4829
15 -12.00-0.4442-0.4268-0.3204-0.4190-0.4182-0.4152
16 -10.00-0.3887-0.3808-0.2936-0.3689-0.3663-0.3638
17  -8.00-0.3661-0.3451-0.2689-0.3302-0.3331-0.3213
18  -6.00-0.3146-0.3000-0.2639-0.2947-0.2933-0.2942
19  -4.00-0.1831-0.1804-0.1881-0.1929-0.1462-0.1255
20  -2.00 0.0049 0.0376 0.1172 0.1493 0.1808 0.2072
21    0.00 0.2911 0.3949 0.4433 0.4716 0.4773 0.4817
22    2.00 0.6353 0.6825 0.6725 0.7069 0.7184 0.7318
23    4.00 0.8520 0.8719 0.8526 0.8990 0.9161 0.9326
24    6.00 1.0226 1.0557 1.0391 1.0988 1.1175 1.1363
25    8.00 1.1131 1.2200 1.1662 1.2218 1.2298 1.2423
26   10.00 1.0837 1.1677 1.1554 1.1544 1.1612 1.1578
27   12.00 1.0545 1.0831 1.0839 1.0921 1.0957 1.1105
28   14.00 1.0925 1.0980 1.1046 1.1135 1.1250 1.1204
29   16.00 1.0820 1.0792 1.0786 1.0826 1.0855 1.0832
30   18.00 1.1083 1.1020 1.1010 1.1051 1.1074 1.1131
31   20.00 1.1582 1.1509 1.1523 1.1550 1.1607 1.1686
32   21.00 0.8000 0.8000 0.8000 0.8000 0.8000 0.8000
33   39.00 1.1800 1.1800 1.1800 1.1800 1.1800 1.1800
34   49.00 1.1800 1.1800 1.1800 1.1800 1.1800 1.1800
35  129.00-1.0000-1.0000-1.0000-1.0000-1.0000-1.0000
36  147.00-1.0000-1.0000-1.0000-1.0000-1.0000-1.0000
37  161.00-0.6200-0.6200-0.6200-0.6200-0.6200-0.6200
38  172.50-0.7800-0.7800-0.7800-0.7800-0.7800-0.7800
39  180.00 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
40      0.0800 0.1200 0.1600 0.2000 0.2400 0.2800
41  -180.00 0.0220 0.0220 0.0220 0.0220 0.0220 0.0220
42  -175.00 0.0620 0.0620 0.0620 0.0620 0.0620 0.0620
43  -170.00 0.1320 0.1320 0.1320 0.1320 0.1320 0.1320
44  -165.00 0.2420 0.2420 0.2420 0.2420 0.2420 0.2420
45  -160.00 0.3020 0.3020 0.3020 0.3020 0.3020 0.3020
46  -140.00 1.0420 1.0420 1.0420 1.0420 1.0420 1.0420
47  -120.00 1.6520 1.6520 1.6520 1.6520 1.6520 1.6520
48  -110.00 1.8520 1.8520 1.8520 1.8520 1.8520 1.8520
49  -100.00 2.0220 2.0220 2.0220 2.0220 2.0220 2.0220
50   -90.00 2.0220 2.0220 2.0220 2.0220 2.0220 2.0220
51   -80.00 1.9620 1.9620 1.9620 1.9620 1.9620 1.9620
52   -70.00 1.8420 1.8420 1.8420 1.8420 1.8420 1.8420
53   -60.00 1.6620 1.6620 1.6620 1.6620 1.6620 1.6620
54   -50.00 1.3920 1.3920 1.3920 1.3920 1.3920 1.3920
55   -30.00 0.5620 0.5620 0.5620 0.5620 0.5620 0.5620
56   -21.00 0.3320 0.3320 0.3320 0.3320 0.3320 0.3320
57   -20.00 0.2582 0.2224 0.2206 0.2550 0.2560 0.2586
58   -18.00 0.2357 0.2334 0.2048 0.2455 0.2320 0.2329
59   -16.00 0.2139 0.2102 0.1862 0.2072 0.2142 0.2079
60   -14.00 0.1899 0.1876 0.1715 0.1843 0.1825 0.1883
61   -12.00 0.1699 0.1621 0.1528 0.1582 0.1585 0.1572
62   -10.00 0.1423 0.1387 0.1308 0.1338 0.1330 0.1334
63    -8.00 0.1217 0.1154 0.1071 0.1105 0.1112 0.1086
64    -6.00 0.0962 0.0937 0.0846 0.0891 0.0879 0.0869
65    -4.00 0.0682 0.0642 0.0568 0.0618 0.0572 0.0558
66    -2.00 0.0470 0.0408 0.0241 0.0296 0.0265 0.0235

```

CAMRAD II airfoil C81 file for the DJI Phantom 3. Filename is 'Ph3_AF12.c81.'

```

67      0.00 0.0371 0.0296 0.0135 0.0217 0.0201 0.0190
68      2.00 0.0346 0.0261 0.0125 0.0204 0.0191 0.0182
69      4.00 0.0374 0.0297 0.0169 0.0243 0.0230 0.0221
70      6.00 0.0470 0.0367 0.0236 0.0295 0.0279 0.0268
71      8.00 0.0672 0.0467 0.0319 0.0364 0.0373 0.0374
72     10.00 0.1019 0.0642 0.0651 0.0652 0.0649 0.0668
73     12.00 0.1194 0.1057 0.1047 0.1031 0.1025 0.1013
74     14.00 0.1387 0.1331 0.1310 0.1292 0.1279 0.1322
75     16.00 0.1876 0.1834 0.1825 0.1804 0.1819 0.1873
76     18.00 0.2307 0.2286 0.2287 0.2264 0.2309 0.2334
77     20.00 0.2646 0.2633 0.2635 0.2648 0.2667 0.2703
78     21.00 0.3320 0.3320 0.3320 0.3320 0.3320 0.3320
79     30.00 0.5620 0.5620 0.5620 0.5620 0.5620 0.5620
80     50.00 1.3920 1.3920 1.3920 1.3920 1.3920 1.3920
81     60.00 1.6620 1.6620 1.6620 1.6620 1.6620 1.6620
82     70.00 1.8420 1.8420 1.8420 1.8420 1.8420 1.8420
83     80.00 1.9620 1.9620 1.9620 1.9620 1.9620 1.9620
84     90.00 2.0220 2.0220 2.0220 2.0220 2.0220 2.0220
85    100.00 2.0220 2.0220 2.0220 2.0220 2.0220 2.0220
86    110.00 1.8520 1.8520 1.8520 1.8520 1.8520 1.8520
87    120.00 1.6520 1.6520 1.6520 1.6520 1.6520 1.6520
88    140.00 1.0420 1.0420 1.0420 1.0420 1.0420 1.0420
89    160.00 0.3020 0.3020 0.3020 0.3020 0.3020 0.3020
90    165.00 0.2420 0.2420 0.2420 0.2420 0.2420 0.2420
91    170.00 0.1320 0.1320 0.1320 0.1320 0.1320 0.1320
92    175.00 0.0620 0.0620 0.0620 0.0620 0.0620 0.0620
93    180.00 0.0220 0.0220 0.0220 0.0220 0.0220 0.0220
94      0.0800 0.1200 0.1600 0.2000 0.2400 0.2800
95   -180.00 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
96   -170.00 0.4000 0.4000 0.4000 0.4000 0.4000 0.4000
97   -165.00 0.3000 0.3000 0.3000 0.3000 0.3000 0.3000
98   -160.00 0.3000 0.3000 0.3000 0.3000 0.3000 0.3000
99   -135.00 0.5000 0.5000 0.5000 0.5000 0.5000 0.5000
100   -90.00 0.5000 0.5000 0.5000 0.5000 0.5000 0.5000
101   -30.00 0.1740 0.1740 0.1740 0.1740 0.1740 0.1740
102   -23.00 0.1120 0.1120 0.1120 0.1120 0.1120 0.1120
103   -20.00 0.0469 0.0132 0.0134 0.0481 0.0486 0.0486
104   -18.00 0.0324 0.0321 0.0043 0.0293 0.0331 0.0337
105   -16.00 0.0189 0.0187-0.0028 0.0194 0.0169 0.0193
106   -14.00 0.0076 0.0064-0.0091 0.0065 0.0071 0.0046
107   -12.00-0.0030-0.0030-0.0141-0.0030-0.0038-0.0032
108   -10.00-0.0094-0.0116-0.0189-0.0118-0.0123-0.0129
109    -8.00-0.0154-0.0173-0.0220-0.0181-0.0199-0.0188
110    -6.00-0.0206-0.0282-0.0222-0.0269-0.0268-0.0219
111    -4.00-0.0411-0.0412-0.0351-0.0374-0.0489-0.0530
112    -2.00-0.0537-0.0590-0.0709-0.0766-0.0816-0.0856
113      0.00-0.0720-0.0856-0.0875-0.0906-0.0896-0.0890
114      2.00-0.0925-0.0908-0.0848-0.0895-0.0904-0.0919
115      4.00-0.0844-0.0810-0.0758-0.0823-0.0842-0.0857
116      6.00-0.0774-0.0750-0.0704-0.0776-0.0787-0.0796
117      8.00-0.0720-0.0682-0.0566-0.0625-0.0620-0.0612
118     10.00-0.0716-0.0459-0.0463-0.0465-0.0463-0.0474
119     12.00-0.0633-0.0548-0.0550-0.0539-0.0530-0.0514
120     14.00-0.0584-0.0558-0.0546-0.0526-0.0512-0.0546
121     16.00-0.0821-0.0812-0.0815-0.0808-0.0823-0.0864
122     18.00-0.1059-0.1062-0.1070-0.1056-0.1093-0.1114
123     20.00-0.1254-0.1269-0.1268-0.1284-0.1296-0.1325
124     23.00-0.1120-0.1120-0.1120-0.1120-0.1120-0.1120
125     30.00-0.1740-0.1740-0.1740-0.1740-0.1740-0.1740
126     90.00-0.5000-0.5000-0.5000-0.5000-0.5000-0.5000
127    135.00-0.5000-0.5000-0.5000-0.5000-0.5000-0.5000
128    160.00-0.3000-0.3000-0.3000-0.3000-0.3000-0.3000
129    165.00-0.3000-0.3000-0.3000-0.3000-0.3000-0.3000
130    170.00-0.4000-0.4000-0.4000-0.4000-0.4000-0.4000
131    180.00 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
132      250000.0

```

CAMRAD II airfoil C81 file for the DJI Phantom 3. Filename is 'Ph3_AF19.c81.'

```

1  PHANTOM 3 AF19                      637 653 637
2      0.0800 0.1200 0.1600 0.2000 0.2400 0.2800
3  -180.00 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
4  -172.50 0.7800 0.7800 0.7800 0.7800 0.7800 0.7800
5  -161.00 0.6200 0.6200 0.6200 0.6200 0.6200 0.6200
6  -147.00 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000
7  -129.00 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000
8  -49.00-1.1800-1.1800-1.1800-1.1800-1.1800-1.1800
9  -39.00-1.1800-1.1800-1.1800-1.1800-1.1800-1.1800
10 -21.00-0.8000-0.8000-0.8000-0.8000-0.8000-0.8000
11 -20.00-0.4506-0.6644-0.6663-0.6652-0.6685-0.6733
12 -18.00-0.4298-0.6059-0.6050-0.6035-0.6059-0.6094
13 -16.00-0.5522-0.5480-0.5437-0.5441-0.5456-0.5477
14 -14.00-0.3803-0.4935-0.4955-0.4937-0.4907-0.4910
15 -12.00-0.3647-0.4496-0.4510-0.4447-0.4498-0.4426
16 -10.00-0.3610-0.4128-0.4106-0.4077-0.4048-0.4109
17  -8.00-0.3382-0.3921-0.3894-0.4006-0.3858-0.3818
18  -6.00-0.3345-0.3720-0.3745-0.3767-0.3782-0.3412
19  -4.00-0.2731-0.2761-0.2769-0.2815-0.2869-0.2192
20  -2.00-0.1062-0.0760-0.0353-0.0001 0.0158 0.0303
21   0.00 0.1575 0.2483 0.2946 0.3226 0.3512 0.3724
22   2.00 0.4507 0.5920 0.6563 0.6962 0.7224 0.7213
23   4.00 0.6521 0.8332 0.8887 0.9222 0.9480 0.9353
24   6.00 0.8211 1.0121 1.0007 1.0085 1.0270 0.9928
25   8.00 0.9120 0.9968 1.0342 1.0983 1.1359 1.1575
26  10.00 1.0499 1.0948 1.1305 1.1345 1.1265 1.1161
27  12.00 1.0099 1.0032 1.0120 1.0051 0.9959 0.9899
28  14.00 0.9550 0.9562 0.9514 0.9516 0.9530 0.9551
29  16.00 0.9761 0.9726 0.9757 0.9740 0.9776 0.9822
30  18.00 1.0163 1.0123 1.0198 1.0270 0.7538 1.0293
31  20.00 0.7608 1.0632 1.0651 1.0769 0.7654 1.0806
32  21.00 0.8000 0.8000 0.8000 0.8000 0.8000 0.8000
33  39.00 1.1800 1.1800 1.1800 1.1800 1.1800 1.1800
34  49.00 1.1800 1.1800 1.1800 1.1800 1.1800 1.1800
35 129.00-1.0000-1.0000-1.0000-1.0000-1.0000-1.0000
36 147.00-1.0000-1.0000-1.0000-1.0000-1.0000-1.0000
37 161.00-0.6200-0.6200-0.6200-0.6200-0.6200-0.6200
38 172.50-0.7800-0.7800-0.7800-0.7800-0.7800-0.7800
39 180.00 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
40   0.0800 0.1200 0.1600 0.2000 0.2400 0.2800
41 -180.00 0.0220 0.0220 0.0220 0.0220 0.0220 0.0220
42 -175.00 0.0620 0.0620 0.0620 0.0620 0.0620 0.0620
43 -170.00 0.1320 0.1320 0.1320 0.1320 0.1320 0.1320
44 -165.00 0.2420 0.2420 0.2420 0.2420 0.2420 0.2420
45 -160.00 0.3020 0.3020 0.3020 0.3020 0.3020 0.3020
46 -140.00 1.0420 1.0420 1.0420 1.0420 1.0420 1.0420
47 -120.00 1.6520 1.6520 1.6520 1.6520 1.6520 1.6520
48 -110.00 1.8520 1.8520 1.8520 1.8520 1.8520 1.8520
49 -100.00 2.0220 2.0220 2.0220 2.0220 2.0220 2.0220
50  -90.00 2.0220 2.0220 2.0220 2.0220 2.0220 2.0220
51  -80.00 1.9620 1.9620 1.9620 1.9620 1.9620 1.9620
52  -70.00 1.8420 1.8420 1.8420 1.8420 1.8420 1.8420
53  -60.00 1.6620 1.6620 1.6620 1.6620 1.6620 1.6620
54  -50.00 1.3920 1.3920 1.3920 1.3920 1.3920 1.3920
55  -30.00 0.5620 0.5620 0.5620 0.5620 0.5620 0.5620
56  -21.00 0.3320 0.3320 0.3320 0.3320 0.3320 0.3320
57  -20.00 0.2249 0.2508 0.2545 0.2508 0.2519 0.2535
58  -18.00 0.2132 0.2310 0.2320 0.2282 0.2287 0.2298
59  -16.00 0.2104 0.2080 0.2050 0.2049 0.2049 0.2055
60  -14.00 0.1787 0.1839 0.1842 0.1826 0.1807 0.1808
61  -12.00 0.1599 0.1605 0.1595 0.1570 0.1582 0.1558
62  -10.00 0.1402 0.1359 0.1336 0.1320 0.1310 0.1321
63   -8.00 0.1100 0.1095 0.1076 0.1095 0.1064 0.1059
64   -6.00 0.0838 0.0845 0.0824 0.0815 0.0810 0.0740
65   -4.00 0.0547 0.0557 0.0525 0.0511 0.0509 0.0359
66   -2.00 0.0318 0.0329 0.0279 0.0245 0.0222 0.0103

```

CAMRAD II airfoil C81 file for the DJI Phantom 3. Filename is 'Ph3_AF19.c81.'

```

67      0.00 0.0252 0.0295 0.0253 0.0224 0.0202 0.0088
68      2.00 0.0268 0.0282 0.0232 0.0204 0.0187 0.0069
69      4.00 0.0347 0.0317 0.0262 0.0232 0.0212 0.0090
70      6.00 0.0453 0.0332 0.0312 0.0320 0.0314 0.0204
71      8.00 0.0498 0.0519 0.0491 0.0476 0.0471 0.0475
72     10.00 0.0796 0.0759 0.0738 0.0729 0.0724 0.0728
73     12.00 0.1184 0.1152 0.1118 0.1128 0.1150 0.1176
74     14.00 0.1711 0.1651 0.1668 0.1678 0.1692 0.1717
75     16.00 0.2082 0.2059 0.2043 0.2061 0.2079 0.2102
76     18.00 0.2396 0.2389 0.2430 0.2441 0.2167 0.2447
77     20.00 0.2481 0.2700 0.2696 0.2744 0.2419 0.2741
78     21.00 0.3320 0.3320 0.3320 0.3320 0.3320 0.3320
79     30.00 0.5620 0.5620 0.5620 0.5620 0.5620 0.5620
80     50.00 1.3920 1.3920 1.3920 1.3920 1.3920 1.3920
81     60.00 1.6620 1.6620 1.6620 1.6620 1.6620 1.6620
82     70.00 1.8420 1.8420 1.8420 1.8420 1.8420 1.8420
83     80.00 1.9620 1.9620 1.9620 1.9620 1.9620 1.9620
84     90.00 2.0220 2.0220 2.0220 2.0220 2.0220 2.0220
85    100.00 2.0220 2.0220 2.0220 2.0220 2.0220 2.0220
86    110.00 1.8520 1.8520 1.8520 1.8520 1.8520 1.8520
87    120.00 1.6520 1.6520 1.6520 1.6520 1.6520 1.6520
88    140.00 1.0420 1.0420 1.0420 1.0420 1.0420 1.0420
89    160.00 0.3020 0.3020 0.3020 0.3020 0.3020 0.3020
90    165.00 0.2420 0.2420 0.2420 0.2420 0.2420 0.2420
91    170.00 0.1320 0.1320 0.1320 0.1320 0.1320 0.1320
92    175.00 0.0620 0.0620 0.0620 0.0620 0.0620 0.0620
93    180.00 0.0220 0.0220 0.0220 0.0220 0.0220 0.0220
94          0.0800 0.1200 0.1600 0.2000 0.2400 0.2800
95   -180.00 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
96   -170.00 0.4000 0.4000 0.4000 0.4000 0.4000 0.4000
97   -165.00 0.3000 0.3000 0.3000 0.3000 0.3000 0.3000
98   -160.00 0.3000 0.3000 0.3000 0.3000 0.3000 0.3000
99   -135.00 0.5000 0.5000 0.5000 0.5000 0.5000 0.5000
100   -90.00 0.5000 0.5000 0.5000 0.5000 0.5000 0.5000
101   -30.00 0.1740 0.1740 0.1740 0.1740 0.1740 0.1740
102   -23.00 0.1120 0.1120 0.1120 0.1120 0.1120 0.1120
103   -20.00 0.0008 0.0264 0.0248 0.0267 0.0273 0.0280
104   -18.00-0.0064 0.0134 0.0127 0.0143 0.0148 0.0154
105   -16.00 0.0042 0.0027 0.0034 0.0034 0.0038 0.0043
106   -14.00-0.0159-0.0059-0.0071-0.0070-0.0058-0.0053
107   -12.00-0.0190-0.0133-0.0144-0.0134-0.0150-0.0131
108   -10.00-0.0227-0.0181-0.0188-0.0188-0.0185-0.0207
109   -8.00-0.0270-0.0235-0.0232-0.0243-0.0233-0.0229
110   -6.00-0.0285-0.0276-0.0276-0.0265-0.0251-0.0239
111   -4.00-0.0331-0.0365-0.0366-0.0358-0.0347-0.0476
112   -2.00-0.0373-0.0417-0.0486-0.0548-0.0573-0.0582
113    0.00-0.0528-0.0655-0.0711-0.0740-0.0780-0.0802
114    2.00-0.0740-0.0918-0.0998-0.1052-0.1088-0.1068
115    4.00-0.0770-0.0963-0.1011-0.1043-0.1069-0.1025
116    6.00-0.0723-0.0780-0.0727-0.0744-0.0759-0.0688
117    8.00-0.0347-0.0434-0.0449-0.0504-0.0531-0.0536
118   10.00-0.0272-0.0279-0.0291-0.0281-0.0265-0.0253
119   12.00-0.0295-0.0303-0.0293-0.0315-0.0348-0.0378
120   14.00-0.0568-0.0536-0.0573-0.0592-0.0612-0.0638
121   16.00-0.0735-0.0732-0.0722-0.0754-0.0774-0.0799
122   18.00-0.0886-0.0891-0.0899-0.0888-0.0613-0.0946
123   20.00-0.0676-0.1049-0.1039-0.1010-0.0670-0.1078
124   23.00-0.1120-0.1120-0.1120-0.1120-0.1120-0.1120
125   30.00-0.1740-0.1740-0.1740-0.1740-0.1740-0.1740
126   90.00-0.5000-0.5000-0.5000-0.5000-0.5000-0.5000
127  135.00-0.5000-0.5000-0.5000-0.5000-0.5000-0.5000
128  160.00-0.3000-0.3000-0.3000-0.3000-0.3000-0.3000
129  165.00-0.3000-0.3000-0.3000-0.3000-0.3000-0.3000
130  170.00-0.4000-0.4000-0.4000-0.4000-0.4000-0.4000
131  180.00 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
132      250000.0

```

CHARM run input file for the DJI Phantom 3. Filename is 'dji.inp.'

```

1  DJI rotor in hover
2  NROTOR    IMKS
3  1         1
4  PATHNAME
5  ../
6  INPUT FILENAMES for rotor 1
7  dji_rw.inp
8  dji_bg.inp
9  dji_bd.inp
10 dji_af.inp
11 none
12 SSPD      RHO
13 340.29    1.225
14 SFRAME    SMODE
15 0          0
16 ADV       ALPHAS
17 0.0       0.0
18 NPSI      NREV      CONVG+   CONVG2   CONVG3   MREV
19 96        200       -1.0     -1.0    -1.0     0
20 IRST      IFREE     IGPR
21 0         0         1
22 IVOUT     NRS       ROUT
23 3         8         0.3      0.4      0.5      0.6      0.7      0.8      0.9
24 0.95
24 NPRINT    IBLPLT    IFILPLT
25 0         -1        3        3        3        3
26 KOPT
27 0
28 ISCAN
29 1
30 ISTRSS
31 0
32 IFV       IQUIK1
33 -1        1
34 MXHLEV    FMDG      NCPU
35 12        1.0      12
36 IROHFV    IROHFP    NUPDATE  IREALT
37 2         1        0         0
38 IFVFW     IFVBL     IFVSU    IFVSC    IFVLS
39 1         1        1         1        1
40 ISURF
41 0
42 ISHIP
43 0
44 IRECON     NOISE     KBROADBAND KSEL
45 0          4         0         0
46 ACOUSTICS_CODE APERIODIC
47 3          0
48 NLS
49 0

```

CHARM rotor blade geometry input file for the DJI Phantom 3. Filename is 'dji.bg.inp.'

```

1  rotor in hover
2  NSEG
3      20
4  CUTOUT
5      0.023876000
6  SL
7      0.004775200      0.004775200      0.004775200      0.004775200
8      0.004775200
9      0.004775200      0.004775200      0.004775200      0.004775200
10     0.004775200      0.004775200      0.004775200      0.004775200
11     0.004775200
12  CHORD
13     0.022037040      0.022037040      0.026875740      0.030271720
14     0.030965140
15     0.030449520      0.028427680      0.026355040      0.024498300
16     0.022905720
17     0.021447760      0.020073620      0.018869660      0.017653000
18     0.016451580
19     0.015250160      0.014033500      0.012989560      0.012006580
20     0.010414000
21     0.010414000
22  ELOFSG
23  SWEEP
24     21*0.0000000
25  TWRD
26     17.674535102
27  TWSTGD
28     1.754097959      1.044244625      -1.117372788      -1.231401224
29     -0.813306122
30     -0.953483265      -0.909247755      -0.683773061      -0.465930204
31     -0.516088163
32     -0.492489796      -0.503704082      -0.463346939      -0.422934694
33     -0.407943673
34     -0.387673469      -0.364771837      -0.446222041      -0.420149796
35     -0.260538776
36  ANHD
37     21*0.0000000
38  THCKND
39     21*0.1000000
40  KFLAP
41     20*0
42  FLAPND
43     20*0.0000000
44  FLHNGE
45     20*0.0000000
46  FLDEFL
47     20*0.0000000
48  NCAM
49     0
50  NCHORD  NSPAN  ICOS
51     1      20      0
52

```

CHARM rotor blade dynamics input file for the DJI Phantom 3. Filename is 'dji_bd.inp.'

```

1 tempbd.inp
2 ISTRM
3 0
4 ISTRIP IFPC IAERO
5 -1 0 1
6 ICOMP IRVFLO ISTFLO
7 1 0 1
8 IART HINGE PRECONE
9 0 0.0000 0.000
10 NMODE
11 1
12 NMDFLP NMDTOR NMDLAG NMDELG
13 1 0 0 0
14 IFXMDE IRIGID
15 0 1
16 FREQMD GMASS for Mode 1
17 1.000 0.5909E-05
18 NRBARS
19 51
20 RBARS(IR), IR=1,NRBARS
21 0.000 0.020 0.040 0.060 0.080 0.100 0.120 0.140 0.160 0.180
22 0.200 0.220 0.240 0.260 0.280 0.300 0.320 0.340 0.360 0.380
23 0.400 0.420 0.440 0.460 0.480 0.500 0.520 0.540 0.560 0.580
24 0.600 0.620 0.640 0.660 0.680 0.700 0.720 0.740 0.760 0.780
25 0.800 0.820 0.840 0.860 0.880 0.900 0.920 0.940 0.960 0.980
26 1.000
27 Rigid flap mode
28 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
29 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
30 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
31 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
32 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
33 0.000
34 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
35 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
36 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
37 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
38 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
39 0.000
40 0.000 0.002 0.005 0.007 0.010 0.012 0.014 0.017 0.019 0.021
41 0.024 0.026 0.029 0.031 0.033 0.036 0.038 0.041 0.043 0.045
42 0.048 0.050 0.053 0.055 0.057 0.060 0.062 0.064 0.067 0.069
43 0.072 0.074 0.076 0.079 0.081 0.084 0.086 0.088 0.091 0.093
44 0.096 0.098 0.100 0.103 0.105 0.107 0.110 0.112 0.115 0.117
45 0.119
46 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
47 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
48 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
49 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
50 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
51 0.000
52 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000
53 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000
54 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000
55 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000
56 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000
57 1.000
58 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
59 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
60 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
61 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
62 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
63 0.000
64

```

CHARM rotor rotor wake input file for the DJI Phantom 3. Filename is 'dji_rw.inp.'

```

1  rotor in hover
2  NBLADE      OMEGA
3  2           837.758041
4  IROTAT      XROTOR           XTILT           ITILT
5  1           0.0           0.0           0.0           0.0           0.0           1
6  ICOLL       COLLD      CT
7  1           7.0           0.01596573
8  ITRIM       A1s      B1s      a1s      b1s
9  0           0.0           0.0           0.0           0.0
10 NOWAKE      ICNVCT      NWAKES      NPWAKE      IFAR      MBCVE
11 0           0           1           1           1           0
12 KSCHEME     KPC
13 0           0
14 NCUT        AOV LAP      ISKEW      IUNS
15 1           -1.0         0           1
16 NZONE       NVORT           BUFFER      KGMCHG      KPERIODIC
17 3           30           30           2           1.2         0           0
18 ICORE       AKINEM      A1           PCOREM      CRMON
19 1           0.0           0.005      0.0           0.0
20 NPTFW
21 192         192         288
22 (CORLIM(NV, IZONE, 1), NV=1, NVORT(IZONE) IZONE=1 (Min core radii)
23 15*0.5 15*0.01
24 (CORLIM(NV, IZONE, 2), NV=1, NVORT(IZONE) IZONE=1 (Max core radii)
25 1.0
26 (CORLIM(NV, IZONE, 1), NV=1, NVORT(IZONE) IZONE=2 (Min core radii)
27 0.1
28 (CORLIM(NV, IZONE, 2), NV=1, NVORT(IZONE) IZONE=2 (Max core radii)
29 1.0
30 (CORLIM(NV, IZONE, 1), NV=1, NVORT(IZONE) IZONE=3 (Min core radii)
31 0.5 0.01
32 (CORLIM(NV, IZONE, 2), NV=1, NVORT(IZONE) IZONE=3 (Max core radii)
33 0.5 0.01
34 (CUTLIM(NV, IZONE, 1), NV=1, NVORT(IZONE) IZONE=1 (Min cutoff distances)
35 15*0.5 15*0.01
36 (CUTLIM(NV, IZONE, 2), NV=1, NVORT(IZONE) IZONE=1 (Max cutoff distances)
37 1.0
38 (CUTLIM(NV, IZONE, 1), NV=1, NVORT(IZONE) IZONE=2 (Min cutoff distances)
39 0.01
40 (CUTLIM(NV, IZONE, 2), NV=1, NVORT(IZONE) IZONE=2 (Max cutoff distances)
41 1.0
42 (CUTLIM(NV, IZONE, 1), NV=1, NVORT(IZONE) IZONE=2 (Min cutoff distances)
43 0.5 0.01
44 (CUTLIM(NV, IZONE, 2), NV=1, NVORT(IZONE) IZONE=2 (Max cutoff distances)
45 0.5 0.01
46 IDYNM
47 1
48 SRAD        SHGHT
49 0.0         0.0
50 NHHI
51 0
52

```

CHARM rotor airfoil input file for the DJI Phantom 3. Filename is 'dji_af.inp.'

```

1      3  0
2      0.00  0.85  1.00
3      0.10  0.10  0.10
4  COMMENT#1
5  COMMENT#2
6  PHANTOM 3 AF12      837 853 837
7      0.0000  0.0800  0.1200  0.1600  0.2000  0.2400  0.2800  0.3500
8  -180.00  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000
9  -172.50  0.7800  0.7800  0.7800  0.7800  0.7800  0.7800  0.7800
10 -161.00  0.6200  0.6200  0.6200  0.6200  0.6200  0.6200  0.6200
11 -147.00  1.0000  1.0000  1.0000  1.0000  1.0000  1.0000  1.0000
12 -129.00  1.0000  1.0000  1.0000  1.0000  1.0000  1.0000  1.0000
13  -49.00 -1.1800 -1.1800 -1.1800 -1.1800 -1.1800 -1.1800 -1.1800
14  -39.00 -1.1800 -1.1800 -1.1800 -1.1800 -1.1800 -1.1800 -1.1800
15  -21.00 -0.8000 -0.8000 -0.8000 -0.8000 -0.8000 -0.8000 -0.8000
16  -20.00 -0.6914 -0.6914 -0.4466 -0.4452 -0.6886 -0.6917 -0.6957
17  -18.00 -0.6219 -0.6219 -0.6175 -0.4109 -0.6261 -0.6182 -0.6210
18  -16.00 -0.5557 -0.5557 -0.5491 -0.3735 -0.5454 -0.5504 -0.5474
19  -14.00 -0.4914 -0.4914 -0.4868 -0.3465 -0.4802 -0.4781 -0.4829
20  -12.00 -0.4442 -0.4442 -0.4268 -0.3204 -0.4190 -0.4182 -0.4152
21  -10.00 -0.3887 -0.3887 -0.3808 -0.2936 -0.3689 -0.3663 -0.3638
22   -8.00 -0.3661 -0.3661 -0.3451 -0.2689 -0.3302 -0.3331 -0.3213
23   -6.00 -0.3146 -0.3146 -0.3000 -0.2639 -0.2947 -0.2933 -0.2942
24   -4.00 -0.1831 -0.1831 -0.1804 -0.1881 -0.1929 -0.1462 -0.1255
25   -2.00  0.0049  0.0049  0.0376  0.1172  0.1493  0.1808  0.2072
26    0.00  0.2911  0.2911  0.3949  0.4433  0.4716  0.4773  0.4817
27    2.00  0.6353  0.6353  0.6825  0.6725  0.7069  0.7184  0.7318
28    4.00  0.8520  0.8520  0.8719  0.8526  0.8990  0.9161  0.9326
29    6.00  1.0226  1.0226  1.0557  1.0391  1.0988  1.1175  1.1363
30    8.00  1.1131  1.1131  1.2200  1.1662  1.2218  1.2298  1.2423
31   10.00  1.0837  1.0837  1.1677  1.1554  1.1544  1.1612  1.1578
32   12.00  1.0545  1.0545  1.0831  1.0839  1.0921  1.0957  1.1105
33   14.00  1.0925  1.0925  1.0980  1.1046  1.1135  1.1250  1.1204
34   16.00  1.0820  1.0820  1.0792  1.0786  1.0826  1.0855  1.0832
35   18.00  1.1083  1.1083  1.1020  1.1010  1.1051  1.1074  1.1131
36   20.00  1.1582  1.1582  1.1509  1.1523  1.1550  1.1607  1.1686
37   21.00  0.8000  0.8000  0.8000  0.8000  0.8000  0.8000  0.8000
38   39.00  1.1800  1.1800  1.1800  1.1800  1.1800  1.1800  1.1800
39   49.00  1.1800  1.1800  1.1800  1.1800  1.1800  1.1800  1.1800
40  129.00 -1.0000 -1.0000 -1.0000 -1.0000 -1.0000 -1.0000 -1.0000
41  147.00 -1.0000 -1.0000 -1.0000 -1.0000 -1.0000 -1.0000 -1.0000
42  161.00 -0.6200 -0.6200 -0.6200 -0.6200 -0.6200 -0.6200 -0.6200
43  172.50 -0.7800 -0.7800 -0.7800 -0.7800 -0.7800 -0.7800 -0.7800
44  180.00  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000
45      0.0000  0.0800  0.1200  0.1600  0.2000  0.2400  0.2800  0.3500
46 -180.00  0.0220  0.0220  0.0220  0.0220  0.0220  0.0220  0.0220
47 -175.00  0.0620  0.0620  0.0620  0.0620  0.0620  0.0620  0.0620
48 -170.00  0.1320  0.1320  0.1320  0.1320  0.1320  0.1320  0.1320
49 -165.00  0.2420  0.2420  0.2420  0.2420  0.2420  0.2420  0.2420
50 -160.00  0.3020  0.3020  0.3020  0.3020  0.3020  0.3020  0.3020
51 -140.00  1.0420  1.0420  1.0420  1.0420  1.0420  1.0420  1.0420
52 -120.00  1.6520  1.6520  1.6520  1.6520  1.6520  1.6520  1.6520
53 -110.00  1.8520  1.8520  1.8520  1.8520  1.8520  1.8520  1.8520
54 -100.00  2.0220  2.0220  2.0220  2.0220  2.0220  2.0220  2.0220
55  -90.00  2.0220  2.0220  2.0220  2.0220  2.0220  2.0220  2.0220
56  -80.00  1.9620  1.9620  1.9620  1.9620  1.9620  1.9620  1.9620
57  -70.00  1.8420  1.8420  1.8420  1.8420  1.8420  1.8420  1.8420
58  -60.00  1.6620  1.6620  1.6620  1.6620  1.6620  1.6620  1.6620
59  -50.00  1.3920  1.3920  1.3920  1.3920  1.3920  1.3920  1.3920
60  -30.00  0.5620  0.5620  0.5620  0.5620  0.5620  0.5620  0.5620
61  -21.00  0.3320  0.3320  0.3320  0.3320  0.3320  0.3320  0.3320
62  -20.00  0.2582  0.2582  0.2224  0.2206  0.2550  0.2560  0.2586
63  -18.00  0.2357  0.2357  0.2334  0.2048  0.2455  0.2320  0.2329
64  -16.00  0.2139  0.2139  0.2102  0.1862  0.2072  0.2142  0.2079
65  -14.00  0.1899  0.1899  0.1876  0.1715  0.1843  0.1825  0.1883
66  -12.00  0.1699  0.1699  0.1621  0.1528  0.1582  0.1585  0.1572

```

CHARM rotor airfoil input file for the DJI Phantom 3. Filename is 'dji_af.inp.'

67	-10.00	0.1423	0.1423	0.1387	0.1308	0.1338	0.1330	0.1334	0.1334
68	-8.00	0.1217	0.1217	0.1154	0.1071	0.1105	0.1112	0.1086	0.1086
69	-6.00	0.0962	0.0962	0.0937	0.0846	0.0891	0.0879	0.0869	0.0869
70	-4.00	0.0682	0.0682	0.0642	0.0568	0.0618	0.0572	0.0558	0.0558
71	-2.00	0.0470	0.0470	0.0408	0.0241	0.0296	0.0265	0.0235	0.0235
72	0.00	0.0371	0.0371	0.0296	0.0135	0.0217	0.0201	0.0190	0.0190
73	2.00	0.0346	0.0346	0.0261	0.0125	0.0204	0.0191	0.0182	0.0182
74	4.00	0.0374	0.0374	0.0297	0.0169	0.0243	0.0230	0.0221	0.0221
75	6.00	0.0470	0.0470	0.0367	0.0236	0.0295	0.0279	0.0268	0.0268
76	8.00	0.0672	0.0672	0.0467	0.0319	0.0364	0.0373	0.0374	0.0374
77	10.00	0.1019	0.1019	0.0642	0.0651	0.0652	0.0649	0.0668	0.0668
78	12.00	0.1194	0.1194	0.1057	0.1047	0.1031	0.1025	0.1013	0.1013
79	14.00	0.1387	0.1387	0.1331	0.1310	0.1292	0.1279	0.1322	0.1322
80	16.00	0.1876	0.1876	0.1834	0.1825	0.1804	0.1819	0.1873	0.1873
81	18.00	0.2307	0.2307	0.2286	0.2287	0.2264	0.2309	0.2334	0.2334
82	20.00	0.2646	0.2646	0.2633	0.2635	0.2648	0.2667	0.2703	0.2703
83	21.00	0.3320	0.3320	0.3320	0.3320	0.3320	0.3320	0.3320	0.3320
84	30.00	0.5620	0.5620	0.5620	0.5620	0.5620	0.5620	0.5620	0.5620
85	50.00	1.3920	1.3920	1.3920	1.3920	1.3920	1.3920	1.3920	1.3920
86	60.00	1.6620	1.6620	1.6620	1.6620	1.6620	1.6620	1.6620	1.6620
87	70.00	1.8420	1.8420	1.8420	1.8420	1.8420	1.8420	1.8420	1.8420
88	80.00	1.9620	1.9620	1.9620	1.9620	1.9620	1.9620	1.9620	1.9620
89	90.00	2.0220	2.0220	2.0220	2.0220	2.0220	2.0220	2.0220	2.0220
90	100.00	2.0220	2.0220	2.0220	2.0220	2.0220	2.0220	2.0220	2.0220
91	110.00	1.8520	1.8520	1.8520	1.8520	1.8520	1.8520	1.8520	1.8520
92	120.00	1.6520	1.6520	1.6520	1.6520	1.6520	1.6520	1.6520	1.6520
93	140.00	1.0420	1.0420	1.0420	1.0420	1.0420	1.0420	1.0420	1.0420
94	160.00	0.3020	0.3020	0.3020	0.3020	0.3020	0.3020	0.3020	0.3020
95	165.00	0.2420	0.2420	0.2420	0.2420	0.2420	0.2420	0.2420	0.2420
96	170.00	0.1320	0.1320	0.1320	0.1320	0.1320	0.1320	0.1320	0.1320
97	175.00	0.0620	0.0620	0.0620	0.0620	0.0620	0.0620	0.0620	0.0620
98	180.00	0.0220	0.0220	0.0220	0.0220	0.0220	0.0220	0.0220	0.0220
99		0.0000	0.0800	0.1200	0.1600	0.2000	0.2400	0.2800	0.3500
100	-180.00	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
101	-170.00	0.4000	0.4000	0.4000	0.4000	0.4000	0.4000	0.4000	0.4000
102	-165.00	0.3000	0.3000	0.3000	0.3000	0.3000	0.3000	0.3000	0.3000
103	-160.00	0.3000	0.3000	0.3000	0.3000	0.3000	0.3000	0.3000	0.3000
104	-135.00	0.5000	0.5000	0.5000	0.5000	0.5000	0.5000	0.5000	0.5000
105	-90.00	0.5000	0.5000	0.5000	0.5000	0.5000	0.5000	0.5000	0.5000
106	-30.00	0.1740	0.1740	0.1740	0.1740	0.1740	0.1740	0.1740	0.1740
107	-23.00	0.1120	0.1120	0.1120	0.1120	0.1120	0.1120	0.1120	0.1120
108	-20.00	0.0469	0.0469	0.0132	0.0134	0.0481	0.0486	0.0486	0.0486
109	-18.00	0.0324	0.0324	0.0321	0.0043	0.0293	0.0331	0.0337	0.0337
110	-16.00	0.0189	0.0189	0.0187	-0.0028	0.0194	0.0169	0.0193	0.0193
111	-14.00	0.0076	0.0076	0.0064	-0.0091	0.0065	0.0071	0.0046	0.0046
112	-12.00	-0.0030	-0.0030	-0.0030	-0.0141	-0.0030	-0.0038	-0.0032	-0.0032
113	-10.00	-0.0094	-0.0094	-0.0116	-0.0189	-0.0118	-0.0123	-0.0129	-0.0129
114	-8.00	-0.0154	-0.0154	-0.0173	-0.0220	-0.0181	-0.0199	-0.0188	-0.0188
115	-6.00	-0.0206	-0.0206	-0.0282	-0.0222	-0.0269	-0.0268	-0.0219	-0.0219
116	-4.00	-0.0411	-0.0411	-0.0412	-0.0351	-0.0374	-0.0489	-0.0530	-0.0530
117	-2.00	-0.0537	-0.0537	-0.0590	-0.0709	-0.0766	-0.0816	-0.0856	-0.0856
118	0.00	-0.0720	-0.0720	-0.0856	-0.0875	-0.0906	-0.0896	-0.0890	-0.0890
119	2.00	-0.0925	-0.0925	-0.0908	-0.0848	-0.0895	-0.0904	-0.0919	-0.0919
120	4.00	-0.0844	-0.0844	-0.0810	-0.0758	-0.0823	-0.0842	-0.0857	-0.0857
121	6.00	-0.0774	-0.0774	-0.0750	-0.0704	-0.0776	-0.0787	-0.0796	-0.0796
122	8.00	-0.0720	-0.0720	-0.0682	-0.0566	-0.0625	-0.0620	-0.0612	-0.0612
123	10.00	-0.0716	-0.0716	-0.0459	-0.0463	-0.0465	-0.0463	-0.0474	-0.0474
124	12.00	-0.0633	-0.0633	-0.0548	-0.0550	-0.0539	-0.0530	-0.0514	-0.0514
125	14.00	-0.0584	-0.0584	-0.0558	-0.0546	-0.0526	-0.0512	-0.0546	-0.0546
126	16.00	-0.0821	-0.0821	-0.0812	-0.0815	-0.0808	-0.0823	-0.0864	-0.0864
127	18.00	-0.1059	-0.1059	-0.1062	-0.1070	-0.1056	-0.1093	-0.1114	-0.1114
128	20.00	-0.1254	-0.1254	-0.1269	-0.1268	-0.1284	-0.1296	-0.1325	-0.1325
129	23.00	-0.1120	-0.1120	-0.1120	-0.1120	-0.1120	-0.1120	-0.1120	-0.1120
130	30.00	-0.1740	-0.1740	-0.1740	-0.1740	-0.1740	-0.1740	-0.1740	-0.1740
131	90.00	-0.5000	-0.5000	-0.5000	-0.5000	-0.5000	-0.5000	-0.5000	-0.5000
132	135.00	-0.5000	-0.5000	-0.5000	-0.5000	-0.5000	-0.5000	-0.5000	-0.5000

CHARM rotor airfoil input file for the DJI Phantom 3. Filename is 'dji_af.inp.'

```

133 160.00 -0.3000 -0.3000 -0.3000 -0.3000 -0.3000 -0.3000 -0.3000 -0.3000
134 165.00 -0.3000 -0.3000 -0.3000 -0.3000 -0.3000 -0.3000 -0.3000 -0.3000
135 170.00 -0.4000 -0.4000 -0.4000 -0.4000 -0.4000 -0.4000 -0.4000 -0.4000
136 180.00 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
137 COMMENT#1
138 COMMENT#2
139 PHANTOM 3 AF12 837 853 837
140 0.0000 0.0800 0.1200 0.1600 0.2000 0.2400 0.2800 0.3500
141 -180.00 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
142 -172.50 0.7800 0.7800 0.7800 0.7800 0.7800 0.7800 0.7800
143 -161.00 0.6200 0.6200 0.6200 0.6200 0.6200 0.6200 0.6200
144 -147.00 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000
145 -129.00 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000
146 -49.00 -1.1800 -1.1800 -1.1800 -1.1800 -1.1800 -1.1800 -1.1800
147 -39.00 -1.1800 -1.1800 -1.1800 -1.1800 -1.1800 -1.1800 -1.1800
148 -21.00 -0.8000 -0.8000 -0.8000 -0.8000 -0.8000 -0.8000 -0.8000
149 -20.00 -0.6914 -0.6914 -0.4466 -0.4452 -0.6886 -0.6917 -0.6957
150 -18.00 -0.6219 -0.6219 -0.6175 -0.4109 -0.6261 -0.6182 -0.6210
151 -16.00 -0.5557 -0.5557 -0.5491 -0.3735 -0.5454 -0.5504 -0.5474
152 -14.00 -0.4914 -0.4914 -0.4868 -0.3465 -0.4802 -0.4781 -0.4829
153 -12.00 -0.4442 -0.4442 -0.4268 -0.3204 -0.4190 -0.4182 -0.4152
154 -10.00 -0.3887 -0.3887 -0.3808 -0.2936 -0.3689 -0.3663 -0.3638
155 -8.00 -0.3661 -0.3661 -0.3451 -0.2689 -0.3302 -0.3331 -0.3213
156 -6.00 -0.3146 -0.3146 -0.3000 -0.2639 -0.2947 -0.2933 -0.2942
157 -4.00 -0.1831 -0.1831 -0.1804 -0.1881 -0.1929 -0.1462 -0.1255
158 -2.00 0.0049 0.0049 0.0376 0.1172 0.1493 0.1808 0.2072
159 0.00 0.2911 0.2911 0.3949 0.4433 0.4716 0.4773 0.4817
160 2.00 0.6353 0.6353 0.6825 0.6725 0.7069 0.7184 0.7318
161 4.00 0.8520 0.8520 0.8719 0.8526 0.8990 0.9161 0.9326
162 6.00 1.0226 1.0226 1.0557 1.0391 1.0988 1.1175 1.1363
163 8.00 1.1131 1.1131 1.1200 1.1662 1.2218 1.2298 1.2423
164 10.00 1.0837 1.0837 1.1677 1.1554 1.1544 1.1612 1.1578
165 12.00 1.0545 1.0545 1.0831 1.0839 1.0921 1.0957 1.1105
166 14.00 1.0925 1.0925 1.0980 1.1046 1.1135 1.1250 1.1204
167 16.00 1.0820 1.0820 1.0792 1.0786 1.0826 1.0855 1.0832
168 18.00 1.1083 1.1083 1.1020 1.1010 1.1051 1.1074 1.1131
169 20.00 1.1582 1.1582 1.1509 1.1523 1.1550 1.1607 1.1686
170 21.00 0.8000 0.8000 0.8000 0.8000 0.8000 0.8000 0.8000
171 39.00 1.1800 1.1800 1.1800 1.1800 1.1800 1.1800 1.1800
172 49.00 1.1800 1.1800 1.1800 1.1800 1.1800 1.1800 1.1800
173 129.00 -1.0000 -1.0000 -1.0000 -1.0000 -1.0000 -1.0000 -1.0000
174 147.00 -1.0000 -1.0000 -1.0000 -1.0000 -1.0000 -1.0000 -1.0000
175 161.00 -0.6200 -0.6200 -0.6200 -0.6200 -0.6200 -0.6200 -0.6200
176 172.50 -0.7800 -0.7800 -0.7800 -0.7800 -0.7800 -0.7800 -0.7800
177 180.00 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
178 0.0000 0.0800 0.1200 0.1600 0.2000 0.2400 0.2800 0.3500
179 -180.00 0.0220 0.0220 0.0220 0.0220 0.0220 0.0220 0.0220
180 -175.00 0.0620 0.0620 0.0620 0.0620 0.0620 0.0620 0.0620
181 -170.00 0.1320 0.1320 0.1320 0.1320 0.1320 0.1320 0.1320
182 -165.00 0.2420 0.2420 0.2420 0.2420 0.2420 0.2420 0.2420
183 -160.00 0.3020 0.3020 0.3020 0.3020 0.3020 0.3020 0.3020
184 -140.00 1.0420 1.0420 1.0420 1.0420 1.0420 1.0420 1.0420
185 -120.00 1.6520 1.6520 1.6520 1.6520 1.6520 1.6520 1.6520
186 -110.00 1.8520 1.8520 1.8520 1.8520 1.8520 1.8520 1.8520
187 -100.00 2.0220 2.0220 2.0220 2.0220 2.0220 2.0220 2.0220
188 -90.00 2.0220 2.0220 2.0220 2.0220 2.0220 2.0220 2.0220
189 -80.00 1.9620 1.9620 1.9620 1.9620 1.9620 1.9620 1.9620
190 -70.00 1.8420 1.8420 1.8420 1.8420 1.8420 1.8420 1.8420
191 -60.00 1.6620 1.6620 1.6620 1.6620 1.6620 1.6620 1.6620
192 -50.00 1.3920 1.3920 1.3920 1.3920 1.3920 1.3920 1.3920
193 -30.00 0.5620 0.5620 0.5620 0.5620 0.5620 0.5620 0.5620
194 -21.00 0.3320 0.3320 0.3320 0.3320 0.3320 0.3320 0.3320
195 -20.00 0.2582 0.2582 0.2224 0.2206 0.2550 0.2560 0.2586
196 -18.00 0.2357 0.2357 0.2334 0.2048 0.2455 0.2320 0.2329
197 -16.00 0.2139 0.2139 0.2102 0.1862 0.2072 0.2142 0.2079
198 -14.00 0.1899 0.1899 0.1876 0.1715 0.1843 0.1825 0.1883

```

CHARM rotor airfoil input file for the DJI Phantom 3. Filename is 'dji_af.inp.'

199	-12.00	0.1699	0.1699	0.1621	0.1528	0.1582	0.1585	0.1572	0.1572
200	-10.00	0.1423	0.1423	0.1387	0.1308	0.1338	0.1330	0.1334	0.1334
201	-8.00	0.1217	0.1217	0.1154	0.1071	0.1105	0.1112	0.1086	0.1086
202	-6.00	0.0962	0.0962	0.0937	0.0846	0.0891	0.0879	0.0869	0.0869
203	-4.00	0.0682	0.0682	0.0642	0.0568	0.0618	0.0572	0.0558	0.0558
204	-2.00	0.0470	0.0470	0.0408	0.0241	0.0296	0.0265	0.0235	0.0235
205	0.00	0.0371	0.0371	0.0296	0.0135	0.0217	0.0201	0.0190	0.0190
206	2.00	0.0346	0.0346	0.0261	0.0125	0.0204	0.0191	0.0182	0.0182
207	4.00	0.0374	0.0374	0.0297	0.0169	0.0243	0.0230	0.0221	0.0221
208	6.00	0.0470	0.0470	0.0367	0.0236	0.0295	0.0279	0.0268	0.0268
209	8.00	0.0672	0.0672	0.0467	0.0319	0.0364	0.0373	0.0374	0.0374
210	10.00	0.1019	0.1019	0.0642	0.0651	0.0652	0.0649	0.0668	0.0668
211	12.00	0.1194	0.1194	0.1057	0.1047	0.1031	0.1025	0.1013	0.1013
212	14.00	0.1387	0.1387	0.1331	0.1310	0.1292	0.1279	0.1322	0.1322
213	16.00	0.1876	0.1876	0.1834	0.1825	0.1804	0.1819	0.1873	0.1873
214	18.00	0.2307	0.2307	0.2286	0.2287	0.2264	0.2309	0.2334	0.2334
215	20.00	0.2646	0.2646	0.2633	0.2635	0.2648	0.2667	0.2703	0.2703
216	21.00	0.3320	0.3320	0.3320	0.3320	0.3320	0.3320	0.3320	0.3320
217	30.00	0.5620	0.5620	0.5620	0.5620	0.5620	0.5620	0.5620	0.5620
218	50.00	1.3920	1.3920	1.3920	1.3920	1.3920	1.3920	1.3920	1.3920
219	60.00	1.6620	1.6620	1.6620	1.6620	1.6620	1.6620	1.6620	1.6620
220	70.00	1.8420	1.8420	1.8420	1.8420	1.8420	1.8420	1.8420	1.8420
221	80.00	1.9620	1.9620	1.9620	1.9620	1.9620	1.9620	1.9620	1.9620
222	90.00	2.0220	2.0220	2.0220	2.0220	2.0220	2.0220	2.0220	2.0220
223	100.00	2.0220	2.0220	2.0220	2.0220	2.0220	2.0220	2.0220	2.0220
224	110.00	1.8520	1.8520	1.8520	1.8520	1.8520	1.8520	1.8520	1.8520
225	120.00	1.6520	1.6520	1.6520	1.6520	1.6520	1.6520	1.6520	1.6520
226	140.00	1.0420	1.0420	1.0420	1.0420	1.0420	1.0420	1.0420	1.0420
227	160.00	0.3020	0.3020	0.3020	0.3020	0.3020	0.3020	0.3020	0.3020
228	165.00	0.2420	0.2420	0.2420	0.2420	0.2420	0.2420	0.2420	0.2420
229	170.00	0.1320	0.1320	0.1320	0.1320	0.1320	0.1320	0.1320	0.1320
230	175.00	0.0620	0.0620	0.0620	0.0620	0.0620	0.0620	0.0620	0.0620
231	180.00	0.0220	0.0220	0.0220	0.0220	0.0220	0.0220	0.0220	0.0220
232		0.0000	0.0800	0.1200	0.1600	0.2000	0.2400	0.2800	0.3500
233	-180.00	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
234	-170.00	0.4000	0.4000	0.4000	0.4000	0.4000	0.4000	0.4000	0.4000
235	-165.00	0.3000	0.3000	0.3000	0.3000	0.3000	0.3000	0.3000	0.3000
236	-160.00	0.3000	0.3000	0.3000	0.3000	0.3000	0.3000	0.3000	0.3000
237	-135.00	0.5000	0.5000	0.5000	0.5000	0.5000	0.5000	0.5000	0.5000
238	-90.00	0.5000	0.5000	0.5000	0.5000	0.5000	0.5000	0.5000	0.5000
239	-30.00	0.1740	0.1740	0.1740	0.1740	0.1740	0.1740	0.1740	0.1740
240	-23.00	0.1120	0.1120	0.1120	0.1120	0.1120	0.1120	0.1120	0.1120
241	-20.00	0.0469	0.0469	0.0132	0.0134	0.0481	0.0486	0.0486	0.0486
242	-18.00	0.0324	0.0324	0.0321	0.0043	0.0293	0.0331	0.0337	0.0337
243	-16.00	0.0189	0.0189	0.0187	-0.0028	0.0194	0.0169	0.0193	0.0193
244	-14.00	0.0076	0.0076	0.0064	-0.0091	0.0065	0.0071	0.0046	0.0046
245	-12.00	-0.0030	-0.0030	-0.0030	-0.0141	-0.0030	-0.0038	-0.0032	-0.0032
246	-10.00	-0.0094	-0.0094	-0.0116	-0.0189	-0.0118	-0.0123	-0.0129	-0.0129
247	-8.00	-0.0154	-0.0154	-0.0173	-0.0220	-0.0181	-0.0199	-0.0188	-0.0188
248	-6.00	-0.0206	-0.0206	-0.0282	-0.0222	-0.0269	-0.0268	-0.0219	-0.0219
249	-4.00	-0.0411	-0.0411	-0.0412	-0.0351	-0.0374	-0.0489	-0.0530	-0.0530
250	-2.00	-0.0537	-0.0537	-0.0590	-0.0709	-0.0766	-0.0816	-0.0856	-0.0856
251	0.00	-0.0720	-0.0720	-0.0856	-0.0875	-0.0906	-0.0896	-0.0890	-0.0890
252	2.00	-0.0925	-0.0925	-0.0908	-0.0848	-0.0895	-0.0904	-0.0919	-0.0919
253	4.00	-0.0844	-0.0844	-0.0810	-0.0758	-0.0823	-0.0842	-0.0857	-0.0857
254	6.00	-0.0774	-0.0774	-0.0750	-0.0704	-0.0776	-0.0787	-0.0796	-0.0796
255	8.00	-0.0720	-0.0720	-0.0682	-0.0566	-0.0625	-0.0620	-0.0612	-0.0612
256	10.00	-0.0716	-0.0716	-0.0459	-0.0463	-0.0465	-0.0463	-0.0474	-0.0474
257	12.00	-0.0633	-0.0633	-0.0548	-0.0550	-0.0539	-0.0530	-0.0514	-0.0514
258	14.00	-0.0584	-0.0584	-0.0558	-0.0546	-0.0526	-0.0512	-0.0546	-0.0546
259	16.00	-0.0821	-0.0821	-0.0812	-0.0815	-0.0808	-0.0823	-0.0864	-0.0864
260	18.00	-0.1059	-0.1059	-0.1062	-0.1070	-0.1056	-0.1093	-0.1114	-0.1114
261	20.00	-0.1254	-0.1254	-0.1269	-0.1268	-0.1284	-0.1296	-0.1325	-0.1325
262	23.00	-0.1120	-0.1120	-0.1120	-0.1120	-0.1120	-0.1120	-0.1120	-0.1120
263	30.00	-0.1740	-0.1740	-0.1740	-0.1740	-0.1740	-0.1740	-0.1740	-0.1740
264	90.00	-0.5000	-0.5000	-0.5000	-0.5000	-0.5000	-0.5000	-0.5000	-0.5000

CHARM rotor airfoil input file for the DJI Phantom 3. Filename is 'dji_af.inp.'

```

265 135.00 -0.5000 -0.5000 -0.5000 -0.5000 -0.5000 -0.5000 -0.5000 -0.5000
266 160.00 -0.3000 -0.3000 -0.3000 -0.3000 -0.3000 -0.3000 -0.3000 -0.3000
267 165.00 -0.3000 -0.3000 -0.3000 -0.3000 -0.3000 -0.3000 -0.3000 -0.3000
268 170.00 -0.4000 -0.4000 -0.4000 -0.4000 -0.4000 -0.4000 -0.4000 -0.4000
269 180.00 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
270 COMMENT#1
271 COMMENT#2
272 PHANTOM 3 AF19 837 853 837
273 0.0000 0.0800 0.1200 0.1600 0.2000 0.2400 0.2800 0.3500
274 -180.00 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
275 -172.50 0.7800 0.7800 0.7800 0.7800 0.7800 0.7800 0.7800
276 -161.00 0.6200 0.6200 0.6200 0.6200 0.6200 0.6200 0.6200
277 -147.00 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000
278 -129.00 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000
279 -49.00 -1.1800 -1.1800 -1.1800 -1.1800 -1.1800 -1.1800 -1.1800
280 -39.00 -1.1800 -1.1800 -1.1800 -1.1800 -1.1800 -1.1800 -1.1800
281 -21.00 -0.8000 -0.8000 -0.8000 -0.8000 -0.8000 -0.8000 -0.8000
282 -20.00 -0.4506 -0.4506 -0.6644 -0.6663 -0.6652 -0.6685 -0.6733
283 -18.00 -0.4298 -0.4298 -0.6059 -0.6050 -0.6035 -0.6059 -0.6094
284 -16.00 -0.5522 -0.5522 -0.5480 -0.5437 -0.5441 -0.5456 -0.5477
285 -14.00 -0.3803 -0.3803 -0.4935 -0.4955 -0.4937 -0.4907 -0.4910
286 -12.00 -0.3647 -0.3647 -0.4496 -0.4510 -0.4447 -0.4498 -0.4426
287 -10.00 -0.3610 -0.3610 -0.4128 -0.4106 -0.4077 -0.4048 -0.4109
288 -8.00 -0.3382 -0.3382 -0.3921 -0.3894 -0.4006 -0.3858 -0.3818
289 -6.00 -0.3345 -0.3345 -0.3720 -0.3745 -0.3767 -0.3782 -0.3412
290 -4.00 -0.2731 -0.2731 -0.2761 -0.2769 -0.2815 -0.2869 -0.2192
291 -2.00 -0.1062 -0.1062 -0.0760 -0.0353 -0.0001 0.0158 0.0303
292 0.00 0.1575 0.1575 0.2483 0.2946 0.3226 0.3512 0.3724
293 2.00 0.4507 0.4507 0.5920 0.6563 0.6962 0.7224 0.7213
294 4.00 0.6521 0.6521 0.8332 0.8887 0.9222 0.9480 0.9353
295 6.00 0.8211 0.8211 1.0121 1.0007 1.0085 1.0270 0.9928
296 8.00 0.9120 0.9120 0.9968 1.0342 1.0983 1.1359 1.1575
297 10.00 1.0499 1.0499 1.0948 1.1305 1.1345 1.1265 1.1161
298 12.00 1.0099 1.0099 1.0032 1.0120 1.0051 0.9959 0.9899
299 14.00 0.9550 0.9550 0.9562 0.9514 0.9516 0.9530 0.9551
300 16.00 0.9761 0.9761 0.9726 0.9757 0.9740 0.9776 0.9822
301 18.00 1.0163 1.0163 1.0123 1.0198 1.0270 0.7538 1.0293
302 20.00 0.7608 0.7608 1.0632 1.0651 1.0769 0.7654 1.0806
303 21.00 0.8000 0.8000 0.8000 0.8000 0.8000 0.8000 0.8000
304 39.00 1.1800 1.1800 1.1800 1.1800 1.1800 1.1800 1.1800
305 49.00 1.1800 1.1800 1.1800 1.1800 1.1800 1.1800 1.1800
306 129.00 -1.0000 -1.0000 -1.0000 -1.0000 -1.0000 -1.0000 -1.0000
307 147.00 -1.0000 -1.0000 -1.0000 -1.0000 -1.0000 -1.0000 -1.0000
308 161.00 -0.6200 -0.6200 -0.6200 -0.6200 -0.6200 -0.6200 -0.6200
309 172.50 -0.7800 -0.7800 -0.7800 -0.7800 -0.7800 -0.7800 -0.7800
310 180.00 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
311 0.0000 0.0800 0.1200 0.1600 0.2000 0.2400 0.2800 0.3500
312 -180.00 0.0220 0.0220 0.0220 0.0220 0.0220 0.0220 0.0220
313 -175.00 0.0620 0.0620 0.0620 0.0620 0.0620 0.0620 0.0620
314 -170.00 0.1320 0.1320 0.1320 0.1320 0.1320 0.1320 0.1320
315 -165.00 0.2420 0.2420 0.2420 0.2420 0.2420 0.2420 0.2420
316 -160.00 0.3020 0.3020 0.3020 0.3020 0.3020 0.3020 0.3020
317 -140.00 1.0420 1.0420 1.0420 1.0420 1.0420 1.0420 1.0420
318 -120.00 1.6520 1.6520 1.6520 1.6520 1.6520 1.6520 1.6520
319 -110.00 1.8520 1.8520 1.8520 1.8520 1.8520 1.8520 1.8520
320 -100.00 2.0220 2.0220 2.0220 2.0220 2.0220 2.0220 2.0220
321 -90.00 2.0220 2.0220 2.0220 2.0220 2.0220 2.0220 2.0220
322 -80.00 1.9620 1.9620 1.9620 1.9620 1.9620 1.9620 1.9620
323 -70.00 1.8420 1.8420 1.8420 1.8420 1.8420 1.8420 1.8420
324 -60.00 1.6620 1.6620 1.6620 1.6620 1.6620 1.6620 1.6620
325 -50.00 1.3920 1.3920 1.3920 1.3920 1.3920 1.3920 1.3920
326 -30.00 0.5620 0.5620 0.5620 0.5620 0.5620 0.5620 0.5620
327 -21.00 0.3320 0.3320 0.3320 0.3320 0.3320 0.3320 0.3320
328 -20.00 0.2249 0.2249 0.2508 0.2545 0.2508 0.2519 0.2535
329 -18.00 0.2132 0.2132 0.2310 0.2320 0.2282 0.2287 0.2298
330 -16.00 0.2104 0.2104 0.2080 0.2050 0.2049 0.2049 0.2055

```

CHARM rotor airfoil input file for the DJI Phantom 3. Filename is 'dji_af.inp.'

331	-14.00	0.1787	0.1787	0.1839	0.1842	0.1826	0.1807	0.1808	0.1808
332	-12.00	0.1599	0.1599	0.1605	0.1595	0.1570	0.1582	0.1558	0.1558
333	-10.00	0.1402	0.1402	0.1359	0.1336	0.1320	0.1310	0.1321	0.1321
334	-8.00	0.1100	0.1100	0.1095	0.1076	0.1095	0.1064	0.1059	0.1059
335	-6.00	0.0838	0.0838	0.0845	0.0824	0.0815	0.0810	0.0740	0.0740
336	-4.00	0.0547	0.0547	0.0557	0.0525	0.0511	0.0509	0.0359	0.0359
337	-2.00	0.0318	0.0318	0.0329	0.0279	0.0245	0.0222	0.0103	0.0103
338	0.00	0.0252	0.0252	0.0295	0.0253	0.0224	0.0202	0.0088	0.0088
339	2.00	0.0268	0.0268	0.0282	0.0232	0.0204	0.0187	0.0069	0.0069
340	4.00	0.0347	0.0347	0.0317	0.0262	0.0232	0.0212	0.0090	0.0090
341	6.00	0.0453	0.0453	0.0332	0.0312	0.0320	0.0314	0.0204	0.0204
342	8.00	0.0498	0.0498	0.0519	0.0491	0.0476	0.0471	0.0475	0.0475
343	10.00	0.0796	0.0796	0.0759	0.0738	0.0729	0.0724	0.0728	0.0728
344	12.00	0.1184	0.1184	0.1152	0.1118	0.1128	0.1150	0.1176	0.1176
345	14.00	0.1711	0.1711	0.1651	0.1668	0.1678	0.1692	0.1717	0.1717
346	16.00	0.2082	0.2082	0.2059	0.2043	0.2061	0.2079	0.2102	0.2102
347	18.00	0.2396	0.2396	0.2389	0.2430	0.2441	0.2167	0.2447	0.2447
348	20.00	0.2481	0.2481	0.2700	0.2696	0.2744	0.2419	0.2741	0.2741
349	21.00	0.3320	0.3320	0.3320	0.3320	0.3320	0.3320	0.3320	0.3320
350	30.00	0.5620	0.5620	0.5620	0.5620	0.5620	0.5620	0.5620	0.5620
351	50.00	1.3920	1.3920	1.3920	1.3920	1.3920	1.3920	1.3920	1.3920
352	60.00	1.6620	1.6620	1.6620	1.6620	1.6620	1.6620	1.6620	1.6620
353	70.00	1.8420	1.8420	1.8420	1.8420	1.8420	1.8420	1.8420	1.8420
354	80.00	1.9620	1.9620	1.9620	1.9620	1.9620	1.9620	1.9620	1.9620
355	90.00	2.0220	2.0220	2.0220	2.0220	2.0220	2.0220	2.0220	2.0220
356	100.00	2.0220	2.0220	2.0220	2.0220	2.0220	2.0220	2.0220	2.0220
357	110.00	1.8520	1.8520	1.8520	1.8520	1.8520	1.8520	1.8520	1.8520
358	120.00	1.6520	1.6520	1.6520	1.6520	1.6520	1.6520	1.6520	1.6520
359	140.00	1.0420	1.0420	1.0420	1.0420	1.0420	1.0420	1.0420	1.0420
360	160.00	0.3020	0.3020	0.3020	0.3020	0.3020	0.3020	0.3020	0.3020
361	165.00	0.2420	0.2420	0.2420	0.2420	0.2420	0.2420	0.2420	0.2420
362	170.00	0.1320	0.1320	0.1320	0.1320	0.1320	0.1320	0.1320	0.1320
363	175.00	0.0620	0.0620	0.0620	0.0620	0.0620	0.0620	0.0620	0.0620
364	180.00	0.0220	0.0220	0.0220	0.0220	0.0220	0.0220	0.0220	0.0220
365		0.0000	0.0800	0.1200	0.1600	0.2000	0.2400	0.2800	0.3500
366	-180.00	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
367	-170.00	0.4000	0.4000	0.4000	0.4000	0.4000	0.4000	0.4000	0.4000
368	-165.00	0.3000	0.3000	0.3000	0.3000	0.3000	0.3000	0.3000	0.3000
369	-160.00	0.3000	0.3000	0.3000	0.3000	0.3000	0.3000	0.3000	0.3000
370	-135.00	0.5000	0.5000	0.5000	0.5000	0.5000	0.5000	0.5000	0.5000
371	-90.00	0.5000	0.5000	0.5000	0.5000	0.5000	0.5000	0.5000	0.5000
372	-30.00	0.1740	0.1740	0.1740	0.1740	0.1740	0.1740	0.1740	0.1740
373	-23.00	0.1120	0.1120	0.1120	0.1120	0.1120	0.1120	0.1120	0.1120
374	-20.00	0.0008	0.0008	0.0264	0.0248	0.0267	0.0273	0.0280	0.0280
375	-18.00	-0.0064	-0.0064	0.0134	0.0127	0.0143	0.0148	0.0154	0.0154
376	-16.00	0.0042	0.0042	0.0027	0.0034	0.0034	0.0038	0.0043	0.0043
377	-14.00	-0.0159	-0.0159	-0.0059	-0.0071	-0.0070	-0.0058	-0.0053	-0.0053
378	-12.00	-0.0190	-0.0190	-0.0133	-0.0144	-0.0134	-0.0150	-0.0131	-0.0131
379	-10.00	-0.0227	-0.0227	-0.0181	-0.0188	-0.0188	-0.0185	-0.0207	-0.0207
380	-8.00	-0.0270	-0.0270	-0.0235	-0.0232	-0.0243	-0.0233	-0.0229	-0.0229
381	-6.00	-0.0285	-0.0285	-0.0276	-0.0276	-0.0265	-0.0251	-0.0239	-0.0239
382	-4.00	-0.0331	-0.0331	-0.0365	-0.0366	-0.0358	-0.0347	-0.0476	-0.0476
383	-2.00	-0.0373	-0.0373	-0.0417	-0.0486	-0.0548	-0.0573	-0.0582	-0.0582
384	0.00	-0.0528	-0.0528	-0.0655	-0.0711	-0.0740	-0.0780	-0.0802	-0.0802
385	2.00	-0.0740	-0.0740	-0.0918	-0.0998	-0.1052	-0.1088	-0.1068	-0.1068
386	4.00	-0.0770	-0.0770	-0.0963	-0.1011	-0.1043	-0.1069	-0.1025	-0.1025
387	6.00	-0.0723	-0.0723	-0.0780	-0.0727	-0.0744	-0.0759	-0.0688	-0.0688
388	8.00	-0.0347	-0.0347	-0.0434	-0.0449	-0.0504	-0.0531	-0.0536	-0.0536
389	10.00	-0.0272	-0.0272	-0.0279	-0.0291	-0.0281	-0.0265	-0.0253	-0.0253
390	12.00	-0.0295	-0.0295	-0.0303	-0.0293	-0.0315	-0.0348	-0.0378	-0.0378
391	14.00	-0.0568	-0.0568	-0.0536	-0.0573	-0.0592	-0.0612	-0.0638	-0.0638
392	16.00	-0.0735	-0.0735	-0.0732	-0.0722	-0.0754	-0.0774	-0.0799	-0.0799
393	18.00	-0.0886	-0.0886	-0.0891	-0.0899	-0.0888	-0.0613	-0.0946	-0.0946
394	20.00	-0.0676	-0.0676	-0.1049	-0.1039	-0.1010	-0.0670	-0.1078	-0.1078
395	23.00	-0.1120	-0.1120	-0.1120	-0.1120	-0.1120	-0.1120	-0.1120	-0.1120
396	30.00	-0.1740	-0.1740	-0.1740	-0.1740	-0.1740	-0.1740	-0.1740	-0.1740

CHARM rotor airfoil input file for the DJI Phantom 3. Filename is 'dji_af.inp.'

```
397      90.00 -0.5000 -0.5000 -0.5000 -0.5000 -0.5000 -0.5000 -0.5000 -0.5000
398     135.00 -0.5000 -0.5000 -0.5000 -0.5000 -0.5000 -0.5000 -0.5000 -0.5000
399     160.00 -0.3000 -0.3000 -0.3000 -0.3000 -0.3000 -0.3000 -0.3000 -0.3000
400     165.00 -0.3000 -0.3000 -0.3000 -0.3000 -0.3000 -0.3000 -0.3000 -0.3000
401     170.00 -0.4000 -0.4000 -0.4000 -0.4000 -0.4000 -0.4000 -0.4000 -0.4000
402     180.00  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000
403
```

Appendix B

Python scripts

This sections includes the Python scripts used for this effort. The scripts were run via Anaconda and made heavy use of both RCOTools and pyAARON. These were primarily used to automate both the running process as well as the parsing of the output files. The scripts are as follows:

- Run script for CHARM with AARON
- Run script for CAMRAD II with AARON
- A parsing script which extracts key parameters from CAMRAD II and CHARM output files and stores them in a spreadsheet.

Python script for running CHARM with AARON.

```

1  nmltmpl = """
2  &global_inputs
3      show_options      = 'True',
4      data_source       = 'charm',
5      charm_charfile    = 'sbscrui.inp',
6      charm_rundir      = 'CHARM_SBS',
7      charm_config      = 'CHARM_SSH',
8      aaronrun_dir      = 'CHARM_aaron_mic{micnum}',
9      aaron_mpi_or_serial = 'serial',
10     AMAT_mpi_or_serial = 'serial',
11     camradiiwake_filter = 0,
12 /
13 &aaron_inputs
14     analysis_type      = 1,
15     nreception_times   = 2049,
16     AMAT_modeX         = 'False',
17     AMAT_modeX_npoints = 2,
18     wt_ifilter         = 0,
19     compute_flg        = 1,
20     compute_self       = 1,
21     form_obs_type      = 1,
22     form_obs_point     = {formobs},
23     form_metadata      = 'All',
24     use_rotor          = 1, 1, 0, 0,
25     form_verbosity     = 'Verbose',
26 /
27 ! -----
28 ! radials for input of t/c and t/c for each rotor
29 ! -----
30
31 rtoc_r1      = 0.00, 0.85, 0.95, 1.0,
32 rtoc_r2      = 0.00, 0.85, 0.95, 1.0,
33
34 toc_r1       = 0.12, 0.12, 0.09, 0.09,
35 toc_r2       = 0.12, 0.12, 0.09, 0.09,
36
37 ! -----
38 ! everything below is for self noise
39 ! -----
40
41 ! -----
42 ! radials for input of TE self noise information
43 ! -----
44
45 rbb_r1       = 0.000, 0.050, 0.100, 0.150, 0.200, 0.250, 0.300,
46               0.350, 0.400, 0.450, 0.500, 0.550, 0.600, 0.650, 0.700, 0.750, 0.800,
47               0.850, 0.950, 1.000,
48 rbb_r2       = 0.000, 0.050, 0.100, 0.150, 0.200, 0.250, 0.300,
49               0.350, 0.400, 0.450, 0.500, 0.550, 0.600, 0.650, 0.700, 0.750, 0.800,
50               0.850, 0.950, 1.000,
51
52 ! -----
53 ! zero lift angles (extracted from C81 tables at 1041. RPM)
54 ! -----
55 ang0_r1      = 18*-1.2963, -1.4702, -1.4702,
56 ang0_r2      = 18*-1.2963, -1.4702, -1.4702,
57
58 ! -----
59 ! TE thickness (scaled from HART II)
60 ! -----
61 teth_r1      = 20*0.0023,
62 teth_r2      = 20*0.0001,
63
64 ! -----
65 ! TE wedge angle (assumed same as HART II)

```

Python script for running CHARM with AARON.

```

63      ! -----
64
65      teang_r1          = 20*18.,
66      teang_r2          = 20*18.,
67
68      /
69      &amat_inputs
70      /
71      """
72
73      import subprocess
74
75      setuplist = [
76      (1,      '-34.4816,  0,          0          '),
77      (2,      '-24.3822, 24.3822,  0          '),
78      (3,      '24.3822,  24.3822,  0          '),
79      (4,      '34.4816,  0,          0          '),
80      (5,      '0,          34.4816,  0          '),
81      (6,      '0,          29.8620, -17.2408 '),
82      (7,      '0,          24.3822, -24.3822 '),
83      (8,      '0,          17.2408, -29.8620 '),
84      (9,      '0,          0,          -34.4816 ')]
85
86      firstrun = True
87
88      for micnum, formobs in setuplist:
89          runcharm = 1
90          text = nmltmpl.format(micnum=micnum, formobs=formobs)
91          micnml = f'run-ssh{micnum}.nml'
92
93          if runcharm == 1 and micnum == 1:
94              with open(micnml, 'w') as fobj:
95                  fobj.write(text)
96                  cmds = ['run_pyaaron', '--nml', micnml]
97                  print("See " + str(cmds))
98                  subprocess.run(cmds)
99          else:
100              with open(micnml, 'w') as fobj:
101                  fobj.write(text)
102                  cmds = ['run_pyaaron', '--nml', micnml, '--skip', 'comprehensive']
103                  print("See " + str(cmds))
104                  subprocess.run(cmds)

```

Python script for running CAMRAD II with AARON.

```

1  nmltmpl = """
2  &global_inputs
3      show_options          = 'True',
4      data_source           = 'camradii',
5      camradrun_dir         = 'CAMRADII_SBS/camradii',
6      camrad_prefix         = 'side-vtip-fly',
7      camrad_config         = 'CAMRAD_SSH',
8      camrad_input_af1      = 'CAMRADII_SBS/camradii-inputs/airfoil/sidebyside_airfoil.run',
9      camrad_input_bin      = 'CAMRADII_SBS/camradii-inputs/inputs/sidebyside_input.run',
10     !
11     aaronrun_dir          = 'CAMRAD_aaron_mic{micnum}',
12     aaron_mpi_or_serial   = 'serial',
13     AMAT_mpi_or_serial    = 'serial',
14     aaron_mpi_nproc       = 36,
15     camradiiwake_filter = 0,
16 /
17 &aaron_inputs
18     analysis_type         = 1,
19     nreception_times       = 2049,
20     AMAT_modeX            = 'False',
21     AMAT_modeX_npoints    = 2,
22     wt_ifilter            = 0,
23     compute_flg           = 1,
24     compute_self          = 1,
25     form_obs_type         = 1,
26     form_obs_point        = {formobs},
27     form_metadata         = 'All',
28     use_rotor             = 1, 1,
29     form_verbosity        = 'Verbose'
30 /
31 ! -----
32 ! radials for input of t/c and t/c for each rotor
33 ! -----
34 rtoc_r1                  = 0.00, 0.85, 0.95, 1.0,
35 rtoc_r2                  = 0.00, 0.85, 0.95, 1.0,
36
37 toc_r1                   = 0.12, 0.12, 0.09, 0.09,
38 toc_r2                   = 0.12, 0.12, 0.09, 0.09,
39
40 ! -----
41 ! everything below is for self noise
42 ! -----
43
44 ! -----
45 ! radials for input of TE self noise information
46 ! -----
47
48 rbb_r1                   = 0.000, 0.050, 0.100, 0.150, 0.200, 0.250, 0.300,
49 0.350, 0.400, 0.450, 0.500, 0.550, 0.600, 0.650, 0.700, 0.750, 0.800,
50 0.850, 0.950, 1.000,
51 rbb_r2                   = 0.000, 0.050, 0.100, 0.150, 0.200, 0.250, 0.300,
52 0.350, 0.400, 0.450, 0.500, 0.550, 0.600, 0.650, 0.700, 0.750, 0.800,
53 0.850, 0.950, 1.000,
54
55 ! -----
56 ! zero lift angles (extracted from C81 tables at 1041. RPM)
57 ! -----
58 ang0_r1                  = 18*-1.2963, -1.4702,-1.4702,
59 ang0_r2                  = 18*-1.2963, -1.4702,-1.4702,
60
61 ! -----
62 ! TE thickness (scaled from HART II)
63 ! -----
64
65 teth_r1                  = 20*0.0023,

```

Python script for running CAMRAD II with AARON.

```

63     teth_r2                = 20*0.0001,
64
65     ! -----
66     ! TE wedge angle (assumed same as HART II)
67     ! -----
68
69     teang_r1                = 20*18.,
70     teang_r2                = 20*18.,
71 /
72 &amat_inputs
73 /
74 """
75
76 setuplist = [
77 (1,      '-34.4816,  0,          0          '),
78 (2,      '-24.3822, 24.3822,  0          '),
79 (3,      '24.3822,  24.3822,  0          '),
80 (4,      '34.4816,  0,          0          '),
81 (5,      '0,         34.4816,  0          '),
82 (6,      '0,         29.8620, -17.2408 '),
83 (7,      '0,         24.3822, -24.3822 '),
84 (8,      '0,         17.2408, -29.8620 '),
85 (9,      '0,         0,         -34.4816 ')]
86
87 import shutil
88 import os
89 import subprocess
90 import rcotools.camrad.camrad_core as cc
91 from rcotools.utils import set_active_dir, ppNestedDict
92
93 # Set path to example directory
94 CURRENT_DIR = os.path.dirname(os.path.abspath(__file__))
95
96 def main(verbose=True):
97     run_cii = 1
98     run_aaron = 1
99
100     # Specify location of original CAMRAD job file directories
101     origdir = os.path.join(CURRENT_DIR)
102     jobsdir = os.path.join(origdir, 'CAMRADII_SBS/camradii')
103     inputdir = os.path.join(origdir, 'CAMRADII_SBS/camradii-inputs/inputs')
104     airfoildir = os.path.join(origdir, 'CAMRADII_SBS/camradii-inputs/airfoil')
105
106     # Define and create example directory
107     newdir = os.path.join(CURRENT_DIR, 'results')
108     # Make directory absolute to avoid path issues
109     newdir = os.path.abspath(newdir)
110     # Set active/output directory for example case, delete any existing files
111     prevcwd = set_active_dir(newdir, deleteifexists=False)
112     if run_cii == 1:
113         # Set active/output directory for example case, delete any existing files
114         prevcwd = set_active_dir(newdir, deleteifexists=True)
115         # Generate input from elementary.list using elementary.csh job input
116         jobfile = 'sidebyside_input.run'
117         basename = os.path.splitext(jobfile)[0]
118         stdout = basename + '.out'
119         infpath = os.path.join(inputdir, jobfile)
120         if verbose:
121             print("Parsing %s..." % infpath)
122         job = cc.parseCamradInput(infpath)
123
124         # Set output file paths to new directory
125         job.set_env_outputdir(newdir)
126
127         # Execute the job
128         exitonerror = job.execute(stdout=stdout, execdir=newdir, jobtype='input_camrad')

```

Python script for running CAMRAD II with AARON.

```

129         if verbose:
130             if exitonerror is False:
131                 print("Successful INPUT run.")
132             else:
133                 print("Run for INPUT failed.")
134
135         # Generate input from naca0012.c81 using naca0012.csh job input
136         jobfile = 'sidebyside_airfoil.run'
137         basename = os.path.splitext(jobfile)[0]
138         stdout = basename + '.out'
139         infpath = os.path.join(airfoildir, jobfile)
140         if verbose:
141             print("Parsing %s..." % infpath)
142         job = cc.parseCamradInput(infpath)
143
144         # Set output file paths to new directory
145         job.set_env_outputdir(newdir)
146
147         # Execute the job
148         exitonerror = job.execute(stdout=stdout, execdir=newdir, jobtype='input_camrad')
149         if verbose:
150             if exitonerror is False:
151                 print("Successful airfoil run.")
152             else:
153                 print("Run for airfoil failed.")
154
155         jobfile = 'side-vtip-fly.run'
156         basename = os.path.splitext(jobfile)[0]
157         stdout = basename + '.out'
158         infpath = os.path.join(jobsdir, jobfile)
159         if verbose:
160             print("Parsing %s..." % infpath)
161         job = cc.parseCamradInput(infpath, nowarnings=True) # Parse job, suppress
162         warnings
163
164         # Set output file paths to new directory
165         job.set_env_outputdir(newdir)
166
167         job.update_env({'BLADEAIRFOIL1': 'sidebyside.tab', 'SHELLINPUT': 'sidebyside.bin'
168         ,
169                     'CAMRAD_SIZER': '3200000000',
170                     'CAMRAD_SIZEI': '100000000',
171                     'CAMRAD_SIZEZ': '20000000',
172                     'CORE_SIZER': '2000000',
173                     'CORE_SIZEI': '2000000',
174                     'CORE_SIZEZ': '4000000',
175                     'TABLE_SIZER': '1000000',
176                     'TABLE_SIZEI': '100000',
177                     'TABLE_SIZEZ': '100000',
178                     'SHELL_SIZER': '400000',
179                     'SHELL_SIZEI': '100000',
180                     'SHELL_SIZEZ': '100000'})
181
182         # Execute the job
183         if verbose:
184             print("Running CamradJob.execute()...")
185         exitonerror = job.execute(stdout=stdout, execdir=newdir)
186         if verbose:
187             if exitonerror is False:
188                 print("Successful run.")
189             else:
190                 print("Run failed.")
191
192         # Parse the solution
193         soln = cc.CamradSolution(stdout)

```

Python script for running CAMRAD II with AARON.

```
193     # Show that the radius is 22 feet
194     radius = soln.getByAcceStext('CASE 1%SUMMARY%CONFIGURATION%ROTOR 1%RADIUS (FT)')
195     print('\nThe radius of Rotor 1 in the output summary is: %f (FT)' % radius)
196
197     # Print out the Operating Conditions from the CASE 1 SUMMARY
198     print('\nThe summary operating consitions are:\n')
199     ppNestedDict(soln['CASE 1']['SUMMARY']['OPERATING CONDITION'])
200
201     # Restore cwd
202     os.chdir(prevcwd)
203
204     ##### Now pyAARON
205     if run_aaron == 1:
206         shutil.copyfile(os.path.join(newdir, 'side-vtip-fly.out'), os.path.join(jobdir,
207             'side-vtip-fly.out'))
208
209         for micnum, formobs in setuplist:
210
211             text = nmltmpl.format(micnum=micnum, formobs=formobs)
212             micnml = f'run-ssh_cii_python{micnum}.nml'
213
214             with open(micnml, 'w') as fobj:
215                 fobj.write(text)
216             cmds = ['run_pyaaron', '--nml', micnml, '--skip', 'comp']
217             print("See " + str(cmds))
218             subprocess.run(cmds)
219
220 if __name__ == "__main__":
221     main(verbose=True)
```

Python script for parsing CAMRAD II and CHARM output files.

```

1  # A Parsing script that pulls results from CAMRAD II and CHARM as well as the percent
    difference between them, then populates a spreadsheet for easy comparison
2  # Developed by Lauren Weist (lauren.p.weist@nasa.gov)
3  """
4  The original camrad_parse_output.py is in rcotools/camrad.
5  The original charm_parse_output.py is in rcotools/charm.
6  The original parser_support.py is in rcotools/Utils.
7  """
8
9  # Import parsing modules from the local directory instead of from rcotools - MUST have
    camrad_parse_output.py, charm_parse_output.py, and parser_support.py with the code for
    it to run (they are included with RCOTools)
10 import os
11 import pandas as pd
12 import statistics
13 import camrad_parse_output as cpo
14 import charm_parse_output as chpo
15 from parser_support import getByAccesstext
16 from parser_support import AccessDict
17
18 # Change the rootrundir to match your run directory - this is the base folder where the
    spreadsheet will be generated
19 rootrundir =
    "C:\\Users\\lweist\\Documents\\2025\\Acs_Eval\\Side-by-Side\\Side_by_side_freewake_ff\\"
20 sheetname = 'cii_ch_comp.xlsx'
21 SheetOut = os.path.join(rootrundir, sheetname)
22
23 # These are the variable names of what we will pull from CII and CHARM. These populate
    to the spreadsheet to help define variables
24 Title = ['Code used']
25 twist = ['linear twist (deg)']
26 rho = ['density']
27 c = ['speed of sound']
28 fltspd = ['flight speed (ft/sec)']
29 solid = ['Solidity']
30 Mtip = ['tip Mach number']
31 tip = ['rotor tip speed']
32 omega = ['Omega (rad/s)']
33 pitch = ['pitch angle (deg)']
34 coll = ['collective']
35 lat = ['lateral cyclic']
36 lng = ['longitudinal cyclic']
37 rad = ['radius']
38 CTs = ['total CT/s']
39 CT = ['total CT']
40 mu = ['Advance ratio (mu)']
41 coning = ['coning (deg)']
42 betac = ['betac (deg)']
43 betas = ['betas (deg)']
44
45 # Only change k if you are running more than 1 case in CAMRAD II
46 k = 1
47
48 # This combines the root directory and the location of your CAMRAD II .out file
49 fpath = rootrundir + "CAMRADII_SBS\\camradii\\side-vtip-fly.out"
50
51 # These lines pull the variable values from CAMRAD II's solution - the variable names
    were found using the camradview RCOTools utility
52 soln = cpo.parseAnalysis(fpath)
53 Title += ["CAMRAD II"]
54 twist += [soln.getByAccesstext("CASE "+str(k)+"%SHELL INPUT DATA%ROTOR:STRUCTURE:ROTOR
    1%BLADE SECTION PROPERTIES%TWISTL")]
55 rho += [soln.getByAccesstext("CASE "+str(k)+"%SUMMARY%OPERATING
    CONDITION%ENVIRONMENT%DENSITY (SLUG/FT**3)")]
56 c += [soln.getByAccesstext("CASE "+str(k)+"%SUMMARY%OPERATING
    CONDITION%ENVIRONMENT%SOUND SPEED (FT/SEC)")]

```

Python script for parsing CAMRAD II and CHARM output files.

```

57 fltspd += [soln.getByAccesstext("CASE "+str(k)+"%TRIM SOLUTION:ROTORCRAFT
PERFORMANCE%OPERATING CONDITION%FLIGHT SPEED%VELOCITY (FT/SEC)")]
58 solid += [soln.getByAccesstext("CASE "+str(k)+"%SUMMARY%CONFIGURATION%ROTOR 1%SOLIDITY
")]
59 Mtip += [soln.getByAccesstext("CASE "+str(k)+"%TRIM SOLUTION:ROTORCRAFT
PERFORMANCE%OPERATING CONDITION%ROTOR SPEED%TIP MACH NUMBER")]
60 tip += [soln.getByAccesstext("CASE "+str(k)+"%TRIM SOLUTION:ROTORCRAFT
PERFORMANCE%OPERATING CONDITION%ROTOR SPEED%TIP SPEED (FT/SEC)")]
61 omega += [soln.getByAccesstext("CASE "+str(k)+"%TRIM SOLUTION:ROTORCRAFT
PERFORMANCE%OPERATING CONDITION%ROTOR SPEED%OMEGA (RAD/SEC)")]
62 pitch += [soln.getByAccesstext("CASE "+str(k)+"%TRIM SOLUTION:ROTORCRAFT
PERFORMANCE%OPERATING CONDITION%AIRFRAME ORIENTATION%PITCH ANGLE (DEG)")]
63 coll += [soln.getByAccesstext("CASE "+str(k)+"%TRIM SOLUTION:ROTORCRAFT
PERFORMANCE%CONTROL SETTINGS%ROTORCRAFT%PILOT CONTROL%COLLECTIVE")]
64 lat += [soln.getByAccesstext("CASE "+str(k)+"%TRIM SOLUTION:ROTORCRAFT
PERFORMANCE%CONTROL SETTINGS%ROTORCRAFT%PILOT CONTROL%LATERAL CYCLIC")]
65 lng += [soln.getByAccesstext("CASE "+str(k)+"%TRIM SOLUTION:ROTORCRAFT
PERFORMANCE%CONTROL SETTINGS%ROTORCRAFT%PILOT CONTROL%LONGITUDINAL CYCLIC")]
66 rad += [soln.getByAccesstext("CASE "+str(k)+"%SUMMARY%CONFIGURATION%ROTOR 1%RADIUS
(FT)")]
67 CTs += [statistics.mean([soln.getByAccesstext("CASE "+str(k)+"%TRIM SOLUTION:ROTOR 1
PERFORMANCE%ROTOR FORCES AND MOMENTS%SHAFT AXES%THRUST%CT/S"))]]
68 CT += [CTs[1]*solid[1]]
69 mu += [soln.getByAccesstext("CASE "+str(k)+"%TRIM SOLUTION:ROTOR 1
PERFORMANCE%OPERATING CONDITION%ADVANCE RATIO")]
70 coning += [soln.getByAccesstext("CASE "+str(k)+"%TRIM SOLUTION:ROTOR 1
PERFORMANCE%OPERATING CONDITION%CONING ANGLE")]
71 betac += [soln.getByAccesstext("CASE "+str(k)+"%TRIM SOLUTION:ROTOR 1
PERFORMANCE%OPERATING CONDITION%HUB PLANE%BC")]
72 betas += [soln.getByAccesstext("CASE "+str(k)+"%TRIM SOLUTION:ROTOR 1
PERFORMANCE%OPERATING CONDITION%HUB PLANE%BS")]
73
74
75 k = 2
76 # This combines the root directory and the location of your CHARM .dat files
77 fpath = rootrmdir + "CHARM_SBS\\"
78
79 # These lines pull the variable values from CHARMS's solution (the various .dat files) -
the variable names were found using the charmview RCOTools utility
80 soln = chpo.parseDatFiles(fpath, "sbscruise", nrotor=None, nowarnings=False, dicttype=
AccessDict)
81 Title += ["CHARM"]
82 twist += [soln.getByAccesstext("prop%TWIST")]
83 rho += [soln.getByAccesstext("perf%RHO")]
84 c += [soln.getByAccesstext("perf%SSPD")]
85 fltspd += [soln.getByAccesstext("prop%VELOCITY")]
86 solid += [soln.getByAccesstext("prop1%SOLIDITY")]
87 Mtip += [soln.getByAccesstext("prop%MTIP")]
88 tip += [soln.getByAccesstext("prop%TIPSPEED")]
89 omega += [soln.getByAccesstext("perf%ROTOR 1%OMEGA")]
90 pitch += [soln.getByAccesstext("perf%ROTOR 1%SHAFTANG")]
91 coll += [soln.getByAccesstext("sweep%Rotor 1%COLL")]
92 lat += [soln.getByAccesstext("sweep%Rotor 1%A1S")]
93 lng += [soln.getByAccesstext("sweep%Rotor 1%B1S")]
94 rad += [soln.getByAccesstext("prop%RADIUS")]
95 CT += [soln.getByAccesstext("sweep%Rotor 1%CT")]
96 CTs += [CT[2]/solid[2]]
97 mu += [soln.getByAccesstext("perf%ROTOR 1%ADVRAT")]
98 coning += [soln.getByAccesstext("sweep%Rotor 1%CONING")]
99 betac += [-1*soln.getByAccesstext("sweep%Rotor 1%als")] #multily by negative 1 to
account for the negative fourier transform in CHARM vs positive in CAMRAD II
100 betas += [-1*soln.getByAccesstext("sweep%Rotor 1%bls")]
101
102
103 #Get percent differences for comparison. If a parameter is zero (such as flight speed in
hover) the spreadhseet will show a 'divide by zero' so that the code can continue

```

Python script for parsing CAMRAD II and CHARM output files.

```

running without throwing an error
104 Title += ["Percent difference"]
105 if ((twist[1] == 0) and (twist[2] == 0)):
106     twist += [0]
107 else:
108     twist += [round(abs(((twist[1] -twist[2])/((abs(twist[1])+abs(twist[2]))/2))*100),2
        )]
109
110 if ((rho[1] == 0) and (rho[2] == 0)):
111     rho += [0]
112 else:
113     rho += [round(abs(((rho[1] -rho[2])/((rho[1]+rho[2])/2))*100),2)]
114
115 if ((c[1] == 0) and (c[2] == 0)):
116     c += [0]
117 else:
118     c += [round(abs(((c[1] -c[2])/((c[1]+c[2])/2))*100),2)]
119
120 if ((fltspd[1] == 0) and (fltspd[2] == 0)):
121     fltspd += [0]
122 else:
123     fltspd += [round(abs(((fltspd[1] -fltspd[2])/((fltspd[1]+fltspd[2])/2))*100),2)]
124
125 if ((solid[1] == 0) and (solid[2] == 0)):
126     solid += [0]
127 else:
128     solid += [round(abs(((solid[1] -solid[2])/((solid[1]+solid[2])/2))*100),2)]
129
130 if ((Mtip[1] == 0) and (Mtip[2] == 0)):
131     Mtip += [0]
132 else:
133     Mtip += [round(abs(((Mtip[1] -Mtip[2])/((Mtip[1]+Mtip[2])/2))*100),2)]
134
135 if ((tip[1] == 0) and (tip[2] == 0)):
136     tip += [0]
137 else:
138     tip += [round(abs(((tip[1] -tip[2])/((tip[1]+tip[2])/2))*100),2)]
139
140 if ((omega[1] == 0) and (omega[2] == 0)):
141     omega += [0]
142 else:
143     omega += [round(abs(((omega[1] -omega[2])/((omega[1]+omega[2])/2))*100),2)]
144
145 if ((pitch[1] == 0) and (pitch[2] == 0)):
146     pitch += [0]
147 else:
148     pitch += [round(abs(((pitch[1] -pitch[2])/((pitch[1]+pitch[2])/2))*100),2)]
149
150 if ((coll[1] == 0) and (coll[2] == 0)):
151     coll += [0]
152 else:
153     coll += [round(abs(((coll[1] -coll[2])/((coll[1]+coll[2])/2))*100),2)]
154
155
156 if ((lat[1] == 0) and (lat[2] == 0)):
157     lat += [0]
158 else:
159     lat += [round(abs(((lat[1] -lat[2])/((lat[1]+lat[2])/2))*100),2)]
160
161 if ((lng[1] == 0) and (lng[2] == 0)):
162     lng += [0]
163 else:
164     lng += [round(abs(((lng[1] -lng[2])/((lng[1]+lng[2])/2))*100),2)]
165
166 if ((rad[1] == 0) and (rad[2] == 0)):
167     rad += [0]

```

Python script for parsing CAMRAD II and CHARM output files.

```

168     else:
169         rad      += [round(abs(((rad[1] -rad[2])/((rad[1]+rad[2])/2))*100),2)]
170
171     if ((CT[1] == 0) and (CT[2] == 0)):
172         CT      += [0]
173     else:
174         CT      += [round(abs(((CT[1] -CT[2])/((CT[1]+CT[2])/2))*100),2)]
175
176
177     if ((CTs[1] == 0) and (CTs[2] == 0)):
178         CTs     += [0]
179     else:
180         CTs     += [round(abs(((CTs[1] -CTs[2])/((CTs[1]+CTs[2])/2))*100),2)]
181
182     if ((mu[1] == 0) and (mu[2] == 0)):
183         mu      += [0]
184     else:
185         mu      += [round(abs(((mu[1] -mu[2])/((mu[1]+mu[2])/2))*100),2)]
186
187     if ((coning[1] == 0) and (coning[2] == 0)):
188         coning   += [0]
189     else:
190         coning   += [round(abs(((coning[1] -coning[2])/((coning[1]+coning[2])/2))*100),2)]
191
192     if ((betac[1] == 0) and (betac[2] == 0)):
193         betac    += [0]
194     else:
195         betac    += [round(abs(((betac[1] -betac[2])/((abs(betac[1])+abs(betac[2]))/2))*100),2
196         )]
197
198     if ((betas[1] == 0) and (betas[2] == 0)):
199         betas    += [0]
200     else:
201         betas    += [round(abs(((betas[1] -betas[2])/((abs(betas[1])+abs(betas[2]))/2))*100),2
202         )]
203
204     # Put all the variables into one data structure for writing to a spreadsheet
205     new_list = [Title, twist, rho, c, fltspd, solid, Mtip, tip, omega, pitch, coll, lat, lng,
206     rad, CT, CTs, mu, coning, betac, betas]
207
208     # Print the table of the parameters to the command line
209     dash = '-' * 73
210
211     for i in range(len(new_list)):
212         if i == 0:
213             print(dash)
214             print('{:<26s}{:<15s}{:<13s}{:<12s}'.format(new_list[i][0],new_list[i][1],new_list[
215             i][2],new_list[i][3]))
216             print(dash)
217         else:
218             print('{:<26s}{:<15.3f}{:<13.3f}{:<12.1f}'.format(new_list[i][0],new_list[i][1],
219             new_list[i][2],new_list[i][3]))
220
221     # These lines write the parameters to a spreadsheet
222     df = pd.DataFrame(new_list)
223     writer = pd.ExcelWriter(SheetOut, engine='xlsxwriter')
224
225     df.to_excel(writer, sheet_name='Comparison', index = False)
226
227     workbook = writer.book
228     worksheet = writer.sheets['Comparison']
229     worksheet.set_column('A:A', 22)
230     worksheet.set_column('B:DP', 15)
231
232     # Conditional formatting for the spreadsheet

```

Python script for parsing CAMRAD II and CHARM output files.

```
229 # Light red fill with dark red text.
230 format1 = workbook.add_format({'bg_color': '#FFC7CE',
231                               'font_color': '#9C0006'})
232
233 # Light yellow fill with dark yellow text.
234 format2 = workbook.add_format({'bg_color': '#FFEB9C',
235                               'font_color': '#9C6500'})
236
237 # Green fill with dark green text.
238 format3 = workbook.add_format({'bg_color': '#C6EFCE',
239                               'font_color': '#006100'})
240
241 worksheet.conditional_format(
242     'D3:D24', {'type': 'cell',
243               'criteria': '<=',
244               'value': 25,
245               'format': format3})
246
247 worksheet.conditional_format(
248     'D3:D24', {'type': 'cell',
249               'criteria': 'between',
250               'minimum': 25,
251               'maximum': 50,
252               'format': format2})
253
254 worksheet.conditional_format(
255     'D3:D24', {'type': 'cell',
256               'criteria': '>=',
257               'value': 50,
258               'format': format1})
259
260 # Saves the spreadsheet
261 writer.save()
262
263 # Prints to the command line the location and name of the generated spreadsheet
264 print("See " + rootrundir + str(sheetname) + " for a spreadsheet of these parameters")
```

Appendix C

MATLAB scripts

Some of the geometric calculations and most of the plotting for this effort were conducted in MATLAB. To provide examples on how to calculate things like microphone distance and blade geometry, as well as how to process and plot AARON results, these scripts were included. The MATLAB scripts included below are as follows:

- Script determining the distance in Cartesian space for the microphones for each vehicle
- Custom CHARM blade geometry generation code. A capability similar to this is also provided by CDI, but this script was used instead to help the user understand what is happening inside the code.
- Plotting script for tonal noise acoustic pressure time history
- Plotting script for broadband self-noise spectra results
- Script for calculating total rotor noise spectra and plotting the results for all mics
- Script for calculating the pyAARON input parameters for broadband self-noise
- Function used by the previous script to find the zero-lift angle of attack for broadband self-noise predictions

MATLAB script for calculating the nine AARON microphone locations.

```
clear all; close all;
rotorradiusft = 10.51; %feet
%lopes: rotorradiusft = 10;
%qsmr: rotorradiusft = 16.52;
%sbs: rotorradiusft = 10.51;

rotorradiusm = rotorradiusft*3.28084; %if you need to convert to meters

%nine mics, 0 deg elevation is in-plane with rotor and 0 deg azimuth is
%over the tail (180 deg azimuth is in front of rotor)
mic_azimuth = [-90 -45 0 45 90];
mic_elev = [0 -30 -45 -60 -90];

for j = 1:length(mic_azimuth)
    %in plane mics: have x and y component, no z
    xlocazi(j,1) = rotorradiusm*sind(mic_azimuth(j));
    ylocazi(j,1) = rotorradiusm*cosd(mic_azimuth(j));
end
zlocazi = zeros(5,1);

for j = 1:length(mic_elev)
    %out of plane mics: have y and z component but no x
    ylocelev(j,1) = rotorradiusm*cosd(mic_elev(j));
    zlocelev(j,1) = rotorradiusm*sind(mic_elev(j));
end
xlocelev = zeros(5,1);

micsazi = [xlocazi ylocazi zlocazi];
micselev = [xlocelev ylocelev zlocelev];

mics = [micsazi;micselev];

fig1=figure('units','normalized','outerposition',[0 0 0.6 1]);
plot3(micsazi(:,1),micsazi(:,2),micsazi(:,3),'or','MarkerSize', 8); hold on;
plot3(micselev(:,1),micselev(:,2),micselev(:,3),'ob','MarkerSize', 10);
    hold on;
plot3(0,0,0,'sk','MarkerSize', 10,'MarkerFaceColor','k')
grid on;
xlabel("X"); ylabel("Y"); zlabel("Z");
saveas(gcf,'sbsmicloc.png');

figure;
plot(micsazi(:,2),micsazi(:,3),'or','MarkerSize', 8); hold on;
plot(micselev(:,2),micselev(:,3),'ob','MarkerSize', 10);
plot(0,0,'sk','MarkerSize', 10,'MarkerFaceColor','k')
grid on;
xlabel("Y"); ylabel("Z");

figure;
plot(micsazi(:,1),micsazi(:,2),'or','MarkerSize', 8); hold on;
```

MATLAB script for calculating the nine AARON microphone locations.

```
plot(micselev(:,1),micselev(:,2), 'ob', 'MarkerSize', 10); hold on;  
plot(0,0, 'sk', 'MarkerSize', 10, 'MarkerFaceColor', 'k')  
grid on;  
xlabel("x"); ylabel("y");
```

Published with MATLAB® R2022b

MATLAB script for generating CHARM blade geometry.

```
clear all;
% These are the parameters a user can input
radius = 10.507; %ft
linear_twist = -16; %deg total
taper = 0.6; %0.6R taper
taper_loc = 0.94; %r/R
sweep = 15; %deg
sweep_loc = 0.94; %r/R
airfoil_change = 0.85; %r/R
root_cutout = 1.728160; %feet
nsecs = 20;
root_twist = 12;
chord = 0.71194226;

% Calculations
% Get applicable radius and segments, check that they add up
radminuscutout = radius - root_cutout;

% Root segment
section_before_airf_change = floor(airfoil_change*nsecs);
%17
rad_before_airf = radminuscutout*airfoil_change;
%7.4561
segment_root = rad_before_airf/section_before_airf_change;
%0.4386

% Tip segment
tiprad = 1-taper_loc;
%0.06
tiprad_ft = tiprad*radminuscutout;
%0.5263
section_tip = floor(nsecs*tiprad);
%1
segment_tip = tiprad_ft/section_tip;

% Middle section
rad_left = radminuscutout - rad_before_airf-tiprad_ft;
%0.74561
frac = sweep_loc-airfoil_change;
%0.090
section_before_tip_change = floor((sweep_loc*nsecs)-
section_before_airf_change); %1
segment_midsec = rad_left/section_before_tip_change;
%0.7895

%make sure the nsecs is correctly done due to rounding
totsecs = section_before_airf_change + section_before_tip_change +
section_tip;
while totsecs < nsecs
    section_before_airf_change = floor(airfoil_change*nsecs)+1;
    segment_root = rad_before_airf/section_before_airf_change;
```

MATLAB script for generating CHARM blade geometry.

```
    totsecs = section_before_airf_change + section_before_tip_change +
    section_tip;
end

% Checks
segment_check = segment_root*section_before_airf_change + segment_tip*...
    section_tip + segment_midsec*section_before_tip_change + root_cutout;
segs = zeros(1,20);

for ii=1:section_before_airf_change
    segs(1,ii) = segment_root;
end
for ii=section_before_airf_change+1:section_before_tip_change+...
    section_before_airf_change
    segs(1,ii) = segment_midsec;
end
for ii=section_before_tip_change+section_before_airf_change
+1:section_before_tip_change+...
    section_before_airf_change+section_tip
    segs(1,ii) = segment_tip;
end

% now calculate the twist per segment
segs_ft = zeros(nsegs+1,1);
segs_ft(1,1) = root_cutout;
for ii = 2:nsegs+1
    segs_ft(ii) = segs_ft(ii-1)+segs(ii-1);
end
segs_r = (segs_ft-root_cutout)./(radius);
twist_perseg = linear_twist.*segs_r + root_twist;
twist_delta = zeros(nsegs,1);
for ii = 1:nsegs
    twist_delta(ii) = twist_perseg(ii+1)-twist_perseg(ii);
end

% Account for taper
% taper = 0.6; %0.6R taper
% taper_loc = 0.94; %r/R

chordlength_at_taper = taper*chord;
```

Published with MATLAB® R2022b

MATLAB script for plotting tonal noise acoustic pressure time history results.

```
clear all; close all;

fig1=figure('units','normalized','outerposition',[0 0 0.5 1]);
root = pwd;
t = tiledlayout(3,3);
t.Padding = 'none';
j=1;
azi = [-90 -45 45 90 0 0 0 0 0];
elev = [0 0 0 0 0 -30 -45 -60 -90];
revtime = 1/(52.366*(180/pi))*360;

% Thickness noise
for i = 1:9
    [timesingle(:,i), Thicknesssingle(:,i)] = importtecfile(strcat(root,...
        "\CHARM_aaron_mic", num2str(i), "\af1aifm_thickness_apth.plt"));
    [timecii(:,i), Thicknesscii(:,i)] = importtecfile(strcat(root,...
        "\CAMRAD_aaron_mic", num2str(i), "\af1aifm_thickness_apth.plt"));
    nexttile

    plot(timesingle(:,i), Thicknesssingle(:,i), 'b', 'linewidth', 2); hold on;
    plot(timesingle(:,i), Thicknesscii(:,i), ':k', 'linewidth', 2);
    ylabel('Acoustic press. [Pa]', 'fontsize', 12, 'fontangle', 'italic')
    xlabel('Time [s]', 'fontsize', 12, 'fontangle', 'italic')
    title(strcat("\psi = ", num2str(azi(i)), "\^{circ}, \phi = ",
        num2str(elev(i)), "\^{circ}"), 'fontsize', 14, 'fontweight', 'normal');
    ylim([-5 5])
    xlim([0 revtime])
    grid on; grid minor;
end

lgd = legend('CHARM', 'CAMRAD II', 'fontsize',
    14, 'Orientation', 'horizontal', 'Location', 'South');
lgd.Layout.Tile = 'south';
legend('boxoff')
saveas(gcf, 'sbsfw_thickness.png');

% Loading noise
fig2=figure('units','normalized','outerposition',[0 0 0.5 1]);
t = tiledlayout(3,3);
t.Padding = 'none';

for i = 1:9
    [timesingle(:,i), loadingsingle(:,i)] = importtecfile(strcat(root,...
        "\CHARM_aaron_mic", num2str(i), "\af1aifm_loading_apth.plt"));
    [timecii(:,i), loadingcii(:,i)] = importtecfile(strcat(root,...
        "\CAMRAD_aaron_mic", num2str(i), "\af1aifm_loading_apth.plt"));

    nexttile

    plot(timesingle(:,i), loadingsingle(:,i), 'b', 'linewidth', 2); hold on;
    plot(timesingle(:,i), loadingcii(:,i), ':k', 'linewidth', 2);
    ylabel('Acoustic press. [Pa]', 'fontsize', 12, 'fontangle', 'italic')
```

MATLAB script for plotting tonal noise acoustic pressure time history results.

```
        xlabel('Time [s]', 'fontsize', 12, 'fontangle', 'italic')
        title(strcat("\psi = ", num2str(azi(i)), "\^{\circ}, \phi = ", num2str(elev(i)), "\^{\circ}"), 'fontsize', 14, 'fontweight', 'normal');
        ylim([-5 5])
        xlim([0 revtime])
        grid on; grid minor;
end

lgd = legend('CHARM', 'CAMRAD II', 'fontsize', 14, 'Orientation', 'horizontal');
lgd.Layout.Tile = 'south';
legend('boxoff')
saveas(gcf, 'sbsfw_loading.png');
```

Published with MATLAB® R2022b

MATLAB script for plotting broadband self-noise spectra results.

```
clear all
close all

fig1=figure('units','normalized','outerposition',[0 0 0.5 1]);
PATH_HEAD = pwd;
t = tiledlayout(3,3);
t.Padding = 'none';
azi = [-90 -45 45 90 0 0 0 0];
elev = [0 0 0 0 -30 -45 -60 -90];
Pref=(2*10^-5)^2;

for j = 1:9

    PATH_HEAD_CAMRAD_mic1=fullfile(PATH_HEAD, strcat("CAMRAD_aaron_mic", num2str(j), "\"));
    PATH_HEAD_CHARM_mic1=fullfile(PATH_HEAD,
    strcat("CHARM_aaron_mic", num2str(j), "\"));

    %%%%%%%%%% Bluntness %%%%%%%%%%
    PATH_mic1=fullfile(PATH_HEAD_CAMRAD_mic1, "asnifm_Bluntness_pbs.plt");
    startRow = 5;
    formatSpec = '%23f%23f%[\n\r]';
    fileID = fopen(PATH_mic1, 'r');
    dataArray = textscan(fileID,
    formatSpec, 'Delimiter', '', 'WhiteSpace', '', 'TextType', 'string', 'HeaderLines'
    , startRow-1, 'ReturnOnError', false, 'EndOfLine', '\r\n');
    fclose(fileID);
    Frequency_Bluntness(:,j) = dataArray(:, 1); CAMRAD_Bluntness_mic(:,j) =
    (dataArray(:, 2));
    clearvars filename startRow formatSpec fileID dataArray ans;

    %%%%%%%%%% Laminar %%%%%%%%%%
    PATH_mic1=fullfile(PATH_HEAD_CAMRAD_mic1, "asnifm_Laminar_vs_pbs.plt");
    startRow = 5;
    formatSpec = '%23f%23f%[\n\r]';
    fileID = fopen(PATH_mic1, 'r');
    dataArray = textscan(fileID,
    formatSpec, 'Delimiter', '', 'WhiteSpace', '', 'TextType', 'string', 'HeaderLines'
    , startRow-1, 'ReturnOnError', false, 'EndOfLine', '\r\n');
    fclose(fileID);
    Frequency_Laminar(:,j) = dataArray(:, 1); CAMRAD_Laminar_mic(:,j) =
    (dataArray(:, 2));
    clearvars filename startRow formatSpec fileID dataArray ans;

    %%%%%%%%%% PressureSide %%%%%%%%%%
    PATH_mic1=fullfile(PATH_HEAD_CAMRAD_mic1, "asnifm_Pressure_Side_pbs.plt");
    startRow = 5;
    formatSpec = '%23f%23f%[\n\r]';
    fileID = fopen(PATH_mic1, 'r');
    dataArray = textscan(fileID,
    formatSpec, 'Delimiter', '', 'WhiteSpace', '', 'TextType', 'string', 'HeaderLines'
    , startRow-1, 'ReturnOnError', false, 'EndOfLine', '\r\n');
    fclose(fileID);
```

MATLAB script for plotting broadband self-noise spectra results.

```
Frequency_PressureSide(:,j) = dataArray(:, 1);
CAMRAD_PressureSide_mic(:,j) = (dataArray(:, 2));
clearvars filename startRow formatSpec fileID dataArray ans;

%%%%%%%%%%%%% Separation %%%%%%%%%%%%%%
PATH_mic1=fullfile(PATH_HEAD_CAMRAD_mic1, "asnifm_Separation_pbs.plt");
startRow = 5;
formatSpec = '%23f%23f%[\n\r]';
fileID = fopen(PATH_mic1,'r');
dataArray = textscan(fileID,
formatSpec, 'Delimiter', '', 'WhiteSpace', '', 'TextType', 'string', 'HeaderLines'
,startRow-1, 'ReturnOnError', false, 'EndOfLine', '\r\n');
fclose(fileID);
Frequency_Separation(:,j) = dataArray(:, 1); CAMRAD_Separation_mic(:,j) =
(dataArray(:, 2));
clearvars filename startRow formatSpec fileID dataArray ans;

%%%%%%%%%%%%% SuctionSide %%%%%%%%%%%%%%
PATH_mic1=fullfile(PATH_HEAD_CAMRAD_mic1, "asnifm_Suction_Side_pbs.plt");
startRow = 5;
formatSpec = '%23f%23f%[\n\r]';
fileID = fopen(PATH_mic1,'r');
dataArray = textscan(fileID,
formatSpec, 'Delimiter', '', 'WhiteSpace', '', 'TextType', 'string', 'HeaderLines'
,startRow-1, 'ReturnOnError', false, 'EndOfLine', '\r\n');
fclose(fileID);
Frequency_SuctionSide(:,j) = dataArray(:, 1); CAMRAD_SuctionSide_mic(:,j)
= (dataArray(:, 2));
clearvars filename startRow formatSpec fileID dataArray ans;

%%%%%%%%%%%%% Tip %%%%%%%%%%%%%%
PATH_mic1=fullfile(PATH_HEAD_CAMRAD_mic1, "asnifm_Tip_pbs.plt");
startRow = 5;
formatSpec = '%23f%23f%[\n\r]';
fileID = fopen(PATH_mic1,'r');
dataArray = textscan(fileID,
formatSpec, 'Delimiter', '', 'WhiteSpace', '', 'TextType', 'string', 'HeaderLines'
,startRow-1, 'ReturnOnError', false, 'EndOfLine', '\r\n');
fclose(fileID);
Frequency_Tip(:,j) = dataArray(:, 1); CAMRAD_Tip_mic(:,j) = (dataArray(:,
2));
clearvars filename startRow formatSpec fileID dataArray ans;

%%%%%%%%%%%%% Total %%%%%%%%%%%%%%
PATH_mic1=fullfile(PATH_HEAD_CAMRAD_mic1, "asnifm_Total_noise_pbs.out.dat");
startRow = 5;
formatSpec = '%23f%23f%[\n\r]';
fileID = fopen(PATH_mic1,'r');
dataArray = textscan(fileID,
formatSpec, 'Delimiter', '', 'WhiteSpace', '', 'TextType', 'string', 'HeaderLines'
,startRow-1, 'ReturnOnError', false, 'EndOfLine', '\r\n');
fclose(fileID);
```

MATLAB script for plotting broadband self-noise spectra results.

```
Frequency_Total(:,j) = dataArray(:, 1); CAMRAD_Total_mic(:,j) =
(dataArray(:, 2));
clearvars filename startRow formatSpec fileID dataArray ans;

% CHARM
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
PATH_mic1=fullfile(PATH_HEAD_CHARM_mic1, "asnifm_Bluntness_pbs.plt");
startRow = 5;
formatSpec = '%23f%23f%[\n\r]';
fileID = fopen(PATH_mic1,'r');
dataArray = textscan(fileID,
formatSpec, 'Delimiter', '', 'WhiteSpace', '', 'TextType', 'string', 'HeaderLines'
,startRow-1, 'ReturnOnError', false, 'EndOfLine', '\r\n');
fclose(fileID);
Frequency_Bluntness(:,j) = dataArray(:, 1); CHARM_Bluntness_mic(:,j) =
(dataArray(:, 2));
clearvars filename startRow formatSpec fileID dataArray ans;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
PATH_mic1=fullfile(PATH_HEAD_CHARM_mic1, "asnifm_Laminar_vs_pbs.plt");
startRow = 5;
formatSpec = '%23f%23f%[\n\r]';
fileID = fopen(PATH_mic1,'r');
dataArray = textscan(fileID,
formatSpec, 'Delimiter', '', 'WhiteSpace', '', 'TextType', 'string', 'HeaderLines'
,startRow-1, 'ReturnOnError', false, 'EndOfLine', '\r\n');
fclose(fileID);
Frequency_Laminar(:,j) = dataArray(:, 1); CHARM_Laminar_mic(:,j) =
(dataArray(:, 2));
clearvars filename startRow formatSpec fileID dataArray ans;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
PATH_mic1=fullfile(PATH_HEAD_CHARM_mic1, "asnifm_Pressure_Side_pbs.plt");
startRow = 5;
formatSpec = '%23f%23f%[\n\r]';
fileID = fopen(PATH_mic1,'r');
dataArray = textscan(fileID,
formatSpec, 'Delimiter', '', 'WhiteSpace', '', 'TextType', 'string', 'HeaderLines'
,startRow-1, 'ReturnOnError', false, 'EndOfLine', '\r\n');
fclose(fileID);
Frequency_PressureSide(:,j) = dataArray(:, 1); CHARM_PressureSide_mic(:,j)
= (dataArray(:, 2));
clearvars filename startRow formatSpec fileID dataArray ans;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
PATH_mic1=fullfile(PATH_HEAD_CHARM_mic1, "asnifm_Separation_pbs.plt");
startRow = 5;
formatSpec = '%23f%23f%[\n\r]';
fileID = fopen(PATH_mic1,'r');
dataArray = textscan(fileID,
formatSpec, 'Delimiter', '', 'WhiteSpace', '', 'TextType', 'string', 'HeaderLines'
,startRow-1, 'ReturnOnError', false, 'EndOfLine', '\r\n');
fclose(fileID);
```

MATLAB script for plotting broadband self-noise spectra results.

```
Frequency_Separation(:,j) = dataArray(:, 1); CHARM_Separation_mic(:,j) =
(dataArray(:, 2));
clearvars filename startRow formatSpec fileID dataArray ans;

%%%%%%%%% SuctionSide %%%%%%%%%%
PATH_mic1=fullfile(PATH_HEAD_CHARM_mic1, "asnifm_Suction_Side_pbs.plt");
startRow = 5;
formatSpec = '%23f%23f%[\n\r]';
fileID = fopen(PATH_mic1,'r');
dataArray = textscan(fileID,
formatSpec, 'Delimiter', '', 'WhiteSpace', '', 'TextType', 'string', 'HeaderLines'
,startRow-1, 'ReturnOnError', false, 'EndOfLine', '\r\n');
fclose(fileID);
Frequency_SuctionSide(:,j) = dataArray(:, 1); CHARM_SuctionSide_mic(:,j) =
(dataArray(:, 2));
clearvars filename startRow formatSpec fileID dataArray ans;

%%%%%%%%% Tip %%%%%%%%%%
PATH_mic1=fullfile(PATH_HEAD_CHARM_mic1, "asnifm_Tip_pbs.plt");
startRow = 5;
formatSpec = '%23f%23f%[\n\r]';
fileID = fopen(PATH_mic1,'r');
dataArray = textscan(fileID,
formatSpec, 'Delimiter', '', 'WhiteSpace', '', 'TextType', 'string', 'HeaderLines'
,startRow-1, 'ReturnOnError', false, 'EndOfLine', '\r\n');
fclose(fileID);
Frequency_Tip(:,j) = dataArray(:, 1); CHARM_Tip_mic(:,j) = (dataArray(:,
2));
clearvars filename startRow formatSpec fileID dataArray ans;

%%%%%%%%% Total %%%%%%%%%%
PATH_mic1=fullfile(PATH_HEAD_CHARM_mic1, "asnifm_Total_noise_pbs.out.dat");
startRow = 5;
formatSpec = '%23f%23f%[\n\r]';
fileID = fopen(PATH_mic1,'r');
dataArray = textscan(fileID,
formatSpec, 'Delimiter', '', 'WhiteSpace', '', 'TextType', 'string', 'HeaderLines'
,startRow-1, 'ReturnOnError', false, 'EndOfLine', '\r\n');
fclose(fileID);
Frequency_Total(:,j) = dataArray(:, 1); CHARM_Total_mic(:,j) =
(dataArray(:, 2));
clearvars filename startRow formatSpec fileID dataArray ans;

%%%%%%%%%add total tonal

PATH_mic1=fullfile(PATH_HEAD_CHARM_mic1, "af1aifm_total_noise_pbs.out.dat");
startRow = 5;
formatSpec = '%23f%23f%[\n\r]';
fileID = fopen(PATH_mic1,'r');
dataArray = textscan(fileID,
formatSpec, 'Delimiter', '', 'WhiteSpace', '', 'TextType', 'string', 'HeaderLines'
,startRow-1, 'ReturnOnError', false, 'EndOfLine', '\r\n');
```

MATLAB script for plotting broadband self-noise spectra results.

```
fclose(fileID);
Frequency_Totaltonal(:,j) = dataArray(:, 1); CHARM_Total_mictonal(:,j) =
(dataArray(:, 2));
clearvars filename startRow formatSpec fileID dataArray ans;

%%%%%%%%% Total %%%%%%%%%%

PATH_mic1=fullfile(PATH_HEAD_CAMRAD_mic1, "af1aifm_total_noise_pbs.out.dat");
startRow = 5;
formatSpec = '%23f%23f%[\n\r]';
fileID = fopen(PATH_mic1,'r');
dataArray = textscan(fileID,
formatSpec, 'Delimiter', ',', 'WhiteSpace', '', 'TextType', 'string', 'HeaderLines'
,startRow-1, 'ReturnOnError', false, 'EndOfLine', '\r\n');
fclose(fileID);
Frequency_Totaltonal(:,j) = dataArray(:, 1); CAMRAD_Total_mictonal(:,j) =
(dataArray(:, 2));
clearvars filename startRow formatSpec fileID dataArray ans;

nexttile

semilogx(Frequency_Bluntness(:,j), CAMRAD_Bluntness_mic(:,j), ':',
'Color', [.6 .6 .6], 'linewidth', 3, 'DisplayName', 'CII bluntness'); hold on
semilogx(Frequency_Bluntness(:,j), CHARM_Bluntness_mic(:,j), 'Color',
[.6 .6 .6], 'linewidth', 3, 'DisplayName', 'CH bluntness');

semilogx(Frequency_PressureSide(:,j), CAMRAD_PressureSide_mic(:,j), ':',
'Color', [1 .6 0], 'linewidth', 3, 'DisplayName', 'CII pressure');
semilogx(Frequency_PressureSide(:,j),
CHARM_PressureSide_mic(:,j), 'Color', [1 .6
0], 'linewidth', 3, 'DisplayName', 'CH pressure');

semilogx(Frequency_Separation(:,j),
CAMRAD_Separation_mic(:,j), ':', 'Color',
[0 .6 .2], 'linewidth', 3, 'DisplayName', 'CII separation');
semilogx(Frequency_Separation(:,j), CHARM_Separation_mic(:,j), 'Color',
[0 .6 .2], 'linewidth', 3, 'DisplayName', 'CH separation');

semilogx(Frequency_SuctionSide(:,j),
CAMRAD_SuctionSide_mic(:,j), ':', 'Color', [0 .4
1], 'linewidth', 3, 'DisplayName', 'CII suction');
semilogx(Frequency_SuctionSide(:,j), CHARM_SuctionSide_mic(:,j), 'Color',
[0 .4 1], 'linewidth', 3, 'DisplayName', 'CH suction');

semilogx(Frequency_Tip(:,j),
CAMRAD_Tip_mic(:,j), ':', 'Color', 'm', 'linewidth', 3, 'DisplayName', 'CII
tip');
semilogx(Frequency_Tip(:,j),
CHARM_Tip_mic(:,j), 'Color', 'm', 'linewidth', 3, 'DisplayName', 'CH tip');

semilogx(Frequency_Total(:,j),
CAMRAD_Total_mic(:,j), ':', 'Color', 'k', 'linewidth', 3, 'DisplayName', 'CII
total');
```

MATLAB script for plotting broadband self-noise spectra results.

```
    semilogx(Frequency_Total(:,j), CHARM_Total_mic(:,j), 'Color','k'
, 'linewidth',3, 'DisplayName', 'CH total');

    xlabel('\it Frequency [Hz]', 'FontSize', 12)
    ylabel({'\it SPL [dB ref. 20 \mu Pa]'}, 'FontSize', 12)
    xlim([0, max(Frequency_Total(:,j))]); ylim([0, 100]); xticks ([0 10 100
1000 10000 200000]);
    title(strcat("\psi = ",num2str(azi(j)), "\^{\circ}, \phi =
",num2str(elev(j)), "\^{\circ}"), 'fontsize', 14, 'fontweight','normal');
    grid minor; grid on
end

lgd = legend('fontsize', 12, 'Orientation','vertical','NumColumns',6);
legend('boxoff')
lgd.Layout.Tile = 'south';
saveas(gcf, 'sbself_9.png');
```

Published with MATLAB® R2022b

MATLAB script for plotting total noise spectra results.

```
clear all
close all

fig1=figure('units','normalized','outerposition',[0 0 0.5 1]);
PATH_HEAD = pwd;
t = tiledlayout(3,3);
t.Padding = 'none';
azi = [-90 -45 45 90 0 0 0 0 0];
elev = [0 0 0 0 0 -30 -45 -60 -90];
Pref=(2*10^-5)^2;

for j = 1:9

    PATH_HEAD_CAMRAD_mic1=fullfile(PATH_HEAD, strcat("CAMRAD_aaron_mic", num2str(j), "\"));
    PATH_HEAD_CHARM_mic1=fullfile(PATH_HEAD,
    strcat("CHARM_aaron_mic", num2str(j), "\"));

    %%%%%%%%% Total %%%%%%%%%

    PATH_mic1=fullfile(PATH_HEAD_CAMRAD_mic1, "asnifm_Total_noise_pbs.out.dat");
    startRow = 5;
    formatSpec = '%23f%23f%[\n\r]';
    fileID = fopen(PATH_mic1,'r');
    dataArray = textscan(fileID,
    formatSpec, 'Delimiter', '', 'WhiteSpace', '', 'TextType', 'string', 'HeaderLines'
    ,startRow-1, 'ReturnOnError', false, 'EndOfLine', '\r\n');
    fclose(fileID);
    Frequencybb_Total(:,j) = dataArray{: , 1}; CAMRAD_Totalbb_mic(:,j) =
    (dataArray{: , 2});
    clearvars filename startRow formatSpec fileID dataArray ans;

    %%%%%%%%% Total %%%%%%%%%

    PATH_mic1=fullfile(PATH_HEAD_CHARM_mic1, "asnifm_Total_noise_pbs.out.dat");
    startRow = 5;
    formatSpec = '%23f%23f%[\n\r]';
    fileID = fopen(PATH_mic1,'r');
    dataArray = textscan(fileID,
    formatSpec, 'Delimiter', '', 'WhiteSpace', '', 'TextType', 'string', 'HeaderLines'
    ,startRow-1, 'ReturnOnError', false, 'EndOfLine', '\r\n');
    fclose(fileID);
    Frequencybb_Total(:,j) = dataArray{: , 1}; CHARM_Totalbb_mic(:,j) =
    (dataArray{: , 2});
    clearvars filename startRow formatSpec fileID dataArray ans;

    %%%%add total tonal

    PATH_mic1=fullfile(PATH_HEAD_CHARM_mic1, "af1aifm_total_noise_pbs.out.dat");
    startRow = 5;
    formatSpec = '%23f%23f%[\n\r]';
    fileID = fopen(PATH_mic1,'r');
```

MATLAB script for plotting total noise spectra results.

```

    dataArray = textscan(fileID,
formatSpec, 'Delimiter', ',', 'WhiteSpace', ' ', 'TextType', 'string', 'HeaderLines'
, startRow-1, 'ReturnOnError', false, 'EndOfLine', '\r\n');
    fclose(fileID);
    Frequency_Totaltonal(:,j) = dataArray(:, 1); CHARM_Total_mictonal(:,j) =
(dataArray(:, 2));
    clearvars filename startRow formatSpec fileID dataArray ans;

    %%%%%%%%%%%%% Total %%%%%%%%%%%%%

PATH_mic1=fullfile(PATH_HEAD_CAMRAD_mic1, "afiaifm_total_noise_pbs.out.dat");
    startRow = 5;
    formatSpec = '%23f%23f%[\n\r]';
    fileID = fopen(PATH_mic1, 'r');
    dataArray = textscan(fileID,
formatSpec, 'Delimiter', ',', 'WhiteSpace', ' ', 'TextType', 'string', 'HeaderLines'
, startRow-1, 'ReturnOnError', false, 'EndOfLine', '\r\n');
    fclose(fileID);
    Frequency_Totaltonal(:,j) = dataArray(:, 1); CAMRAD_Total_mictonal(:,j) =
(dataArray(:, 2));
    clearvars filename startRow formatSpec fileID dataArray ans;

    % add them
    freq_fix(:,j) = [5; Frequencybb_Total(:,j)];
    CAMRAD_Totalbb_mic2(:,j) = [0; CAMRAD_Totalbb_mic(:,j)];
    CAMRAD_Total_mictonal2(:,j) = [CAMRAD_Total_mictonal(:,j); 0; 0; 0; 0];

    CHARM_Totalbb_mic2(:,j) = [0; CHARM_Totalbb_mic(:,j)];
    CHARM_Total_mictonal2(:,j) = [CHARM_Total_mictonal(:,j); 0; 0; 0; 0];

    CHARM_tot(:,j) = log10(10.^CHARM_Total_mictonal2(:,j) +
10.^CHARM_Totalbb_mic2(:,j));
    camrad_tot(:,j) = log10(10.^CAMRAD_Total_mictonal2(:,j) +
10.^CAMRAD_Totalbb_mic2(:,j));

    nexttile
    semilogx(freq_fix(:,j),
CHARM_tot(:,j), 'Color', 'r', 'linewidth', 2); hold on;
    semilogx(freq_fix(:,j), camrad_tot(:,j), ':', 'Color', 'k', 'linewidth', 3);

    xlabel('\it Frequency [Hz]', 'FontSize', 12)
    ylabel({'\it SPL [dB ref. 20 \mu Pa]'}, 'FontSize', 12)
    xlim([0, max(freq_fix(:,j))]); ylim([0, 100]); xticks ([0 10 100 1000
10000 200000]);
    title(strcat("\psi = ", num2str(azi(j)), "\^{\circ}, \phi =
", num2str(elev(j)), "\^{\circ}"), 'fontsize', 14, 'fontweight', 'normal');
    grid minor; grid on
end

lgd = legend('CHARM', 'CAMRAD II', 'fontsize', 14, 'Orientation', 'horizontal');
legend('boxoff')
lgd.Layout.Tile = 'south';
saveas(gcf, 'sbspectra.png');

```

MATLAB script for generating pyAARON broadband self-noise inputs.

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%      Natasha Schatzman      %%%
%%%      NASA Ames Research Center %%%
%%%      June 12, 2023      %%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clear all
close all

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%      USER INPUT      %%%
%%%      %%%      %%%
%%%      %%%      %%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
Airfoil='E387';
chord= 0.03912; %(meters)
r_R=.1; %(meters)%radial location of interest
Radius=0.311023*r_R; %(meters)%radial location of interest
c=343;% (m/s) % speed of sound
RPM=5200;

omega=(RPM.*(2*pi))/60;
Vtip=omega.*Radius;
Mach_number=Vtip./c;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%      BEGIN SOLVER      %%%
%%%      %%%      %%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% If NACA airfoil entered
if Airfoil(1:4)== 'NACA'
    % Symmetrical 4-digit NACA airfoil
    if Airfoil(5:6)== '00'
        X_Upper=0:.001:1;
        X_Lower=0:.001:1;
        thickness=(str2num(Airfoil(end-1:end)))/100;

        Y_Upper=5*thickness*(0.2969*sqrt(X_Upper)-0.1260*X_Upper-0.3516*X_Upper.^2+0.2843...
            *X_Upper.^3-0.1015*X_Upper.^4);

        Y_Lower=-5*thickness*(0.2969*sqrt(X_Lower)-0.1260*X_Lower-0.3516*X_Lower.^2+0.2843...
            *X_Lower.^3-0.1015*X_Upper.^4);
    end
    % Cambered 4-digit NACA airfoil
    if Airfoil(5:6)~= '00'
    end
    Data_to_Write=["NACA00", str2num(Airfoil(end-1:end))];
    Data_to_Write(2:length(X_Upper)+length(X_Lower)+1,1)=[X_Upper,X_Lower]';
    Data_to_Write(2:length(Y_Upper)+length(Y_Lower)+1,2)=[Y_Upper,Y_Lower]';
    File_Name=fullfile(['Airfoil_Geometry\ ',Airfoil, '.txt']);
    writematrix(Data_to_Write,File_Name);
end
```

MATLAB script for generating pyAARON broadband self-noise inputs.

```
File_Name=fullfile(['Airfoil_Geometry\'',Airfoil, '.txt']);
File = readtable(File_Name);
X=File{1:end,1};
Y=File{1:end,2};
clear opts
Loc_upper=1;
Loc_lower=2;
[M,I]=find(X==1);
Upper_X=X(M(Loc_upper));
Upper_Y=Y(M(Loc_upper));
Lower_X=X(M(Loc_lower));
Lower_Y=Y(M(Loc_lower));
Trailing_Edge_Thickness=(Upper_Y-Lower_Y)*chord;
X_thickness=X(1:length(Y)/2)-X(length(Y)/2+1:end);
Y_thickness=Y(1:length(Y)/2)-Y(length(Y)/2+1:end);
Airfoil_Thickenss_max=max(Y_thickness);
slope_check_upper=Y(M(Loc_upper))-Y(M(Loc_upper)-1);
slope_check_lower=Y(M(Loc_lower))-Y(M(Loc_lower)-1);
check=1;
if slope_check_upper>=0 | slope_check_lower<=0
    M(1)=M(1)-check;
    M(2)=M(2)-check;
    check = check+1;
    slope_check_upper=Y(M(Loc_upper))-Y(M(Loc_upper)-1);
    slope_check_lower=Y(M(Loc_lower))-Y(M(Loc_lower)-1);
    if slope_check_upper>=0 | slope_check_lower<=0
        M(1)=M(1)-check;
        M(2)=M(2)-check;
        check = check+1;
        slope_check_upper=Y(M(Loc_upper))-Y(M(Loc_upper)-1);
        slope_check_lower=Y(M(Loc_lower))-Y(M(Loc_lower)-1);
    end
    if slope_check_upper>=0 | slope_check_lower<=0
        M(1)=M(1)-check;
        M(2)=M(2)-check;
        check = check+1;
        slope_check_upper=Y(M(Loc_upper))-Y(M(Loc_upper)-1);
        slope_check_lower=Y(M(Loc_lower))-Y(M(Loc_lower)-1);
    end
end
Lower_Line_X=[X(M(Loc_lower)-1), X(M(Loc_lower))];
Lower_Line_Y=[Y(M(Loc_lower)-1), Y(M(Loc_lower))];
Upper_Slope=(Y(M(Loc_upper)-1)- Y(M(Loc_upper)))/(X(M(Loc_upper)-1)-
    X(M(Loc_upper)));
Upper_B=-Upper_Slope*(X(M(Loc_upper)))+Y(M(Loc_upper));
Lower_Slope=(Y(M(Loc_lower)-1)- Y(M(Loc_lower)))/(X(M(Loc_lower)-1)-
    X(M(Loc_lower)));
Lower_B=-Lower_Slope*(X(M(Loc_lower)))+Y(M(Loc_lower));
Upper_X_new=1.03;
Upper_Y_new=Upper_Slope*Upper_X_new + Upper_B;
Upper_Line_X=[X(M(Loc_upper)-1), X(M(Loc_upper)), Upper_X_new];
Upper_Line_Y=[Y(M(Loc_upper)-1), Y(M(Loc_upper)), Upper_Y_new];
Lower_X_new=1.03;
Lower_Y_new=Lower_Slope*Lower_X_new + Lower_B;
```

MATLAB script for generating pyAARON broadband self-noise inputs.

```
Lower_Line_X=[X(M(Loc_lower)-1), X(M(Loc_lower)), Lower_X_new];
Lower_Line_Y=[Y(M(Loc_lower)-1), Y(M(Loc_lower)), Lower_Y_new];
v_1 = [X(M(Loc_upper)),Y(M(Loc_upper)),0] -
[X(M(Loc_upper)-1),Y(M(Loc_upper)-1),0];
v_2 = [X(M(Loc_lower)),Y(M(Loc_lower)),0] -
[X(M(Loc_lower)-1),Y(M(Loc_lower)-1),0];
Theta = atan2(norm(cross(v_1, v_2)), dot(v_1, v_2))*180/pi;
Trailing_Edge_Wedge_Angle=Theta;

% Solve for zero lift
C81_file=fullfile(['Airfoil_C81\','Airfoil, '.c81']);
% file = ls(C81_file);
file = C81_file %file(1,:);

% Format specifications
DataStartLine = 2;
VariableNames = {'Col1', 'Col2', 'Col3', 'Col4', 'Col5', 'Col6', 'Col7', ...
'Col8', 'Col9', 'Col10'};
VariableWidths = [7,7,7,7,7,7,7,7,7,7] ;
opts = fixedWidthImportOptions('VariableNames',VariableNames,...
'DataLines',DataStartLine,'VariableWidths',VariableWidths);
opts.ExtraColumnsRule = 'addvars';

% Read File
T = table2array(readtable(file,opts));
T = str2double(T(1:end-1,:));

% Parse
for data = 1:3 %1 for Cl, 2 for Cd, 3 for Cm
    if isempty(T)
        return
    end
    if isnan(T(2,1)) %data spans two lines
        lines_Cx = 2;
        [x,y]=find(isnan(T(3:2:end,1)));
        end_of_Cx = 2*x-1;
        Cx = [T(3:2:end_of_Cx,2:10),T(4:2:end_of_Cx+1,2:10)];
        if data == 1 %reads Cl data
            Mach_Cl = [T(1,:),T(2,:)];
            Mach_Cl = rmmissing(Mach_Cl);
            angle_Cl = T(3:2:end_of_Cx,1);
            Cl = rmmissing(Cx,2);
        elseif data == 2 %reads Cd data
            Mach_Cd = [T(1,:),T(2,:)];
            Mach_Cd = rmmissing(Mach_Cd);
            angle_Cd= T(3:2:end_of_Cx,1);
            Cd = rmmissing(Cx,2);
        elseif data == 3 %reads Cm data
            Mach_Cm = [T(1,:),T(2,:)];
            Mach_Cm = rmmissing(Mach_Cm);
            angle_Cm= T(3:2:end_of_Cx,1);
            Cm = rmmissing(Cx,2);
        end
        T(1:end_of_Cx+1,:) = [];
```

MATLAB script for generating pyAARON broadband self-noise inputs.

```
else %data spans only one line
    lines_Cx = 1;
    [x,y]=find(isnan(T(3:2:end,1)));
    if ~isempty(x)
        end_of_Cx = x-1;
    else
        end_of_Cx = length(T)-1;
    end
    Cx = T(2:end_of_Cx,2:10);
    if data == 1 %reads Cl data
        Mach_Cl = T(1,:);
        Mach_Cl = rmmissing(Mach_Cl);
        angle_Cl = T(2:end_of_Cx,1);
        Cl = rmmissing(Cx,2);
    elseif data == 2 %reads Cd data
        Mach_Cd = T(1,:);
        Mach_Cd = rmmissing(Mach_Cd);
        angle_Cd= T(2:end_of_Cx,1);
        Cd = rmmissing(Cx,2);
    elseif data == 3 %reads Cm data
        Mach_Cm = T(1,:);
        Mach_Cm = rmmissing(Mach_Cm);
        angle_Cm= T(2:end_of_Cx,1);
        Cm = rmmissing(Cx,2);
    end
    T(1:end_of_Cx+1,:) = []; %delete parsed data for recursive analysis
end

end

[MACH_Cl,ANGLE_Cl] = meshgrid(Mach_Cl,angle_Cl);

% [~,Top2MinLoc] = mink(abs(Mach(1,:) - SelectMach),2);
% if Mach(1, min(Top2MinLoc)) - Mach(1, max(Top2MinLoc)) >0; Top2MinLoc =
%   Top2MinLoc -1; end
% OurMachCols = [min(Mach(1,Top2MinLoc)),max(Mach(1,Top2MinLoc))];
% %disp(OurMachCols)
% OurCoLCols = CoL(:,min(Top2MinLoc):max(Top2MinLoc));
% NewCoL_Col = interp1(OurMachCols,OurCoLCols',SelectMach);
% OurAlphaCols = Alpha(:,min(Top2MinLoc):max(Top2MinLoc));
% NewAlpha_Col = interp1(OurMachCols,OurAlphaCols',SelectMach);
% RelevantCoLData = NewCoL_Col(NewAlpha_Col < MaxAlphaAllow & NewAlpha_Col >
%   MinAlphaAllow);
% RelevantAlphaData = NewAlpha_Col(NewAlpha_Col < MaxAlphaAllow & NewAlpha_Col
%   > MinAlphaAllow);
%
% CrossZero = find((RelevantCoLData(1:end-1) -
%   SelectCoL).*(RelevantCoLData(2:end) - SelectCoL) <= 0);
% New_X = [CrossZero;CrossZero+1];
% New_Y = RelevantCoLData(New_X);
% if numel(New_Y) == 2; New_Y = New_Y'; end
% slope = New_Y(2,:) - New_Y(1,:);
% int = New_Y(1,:) - slope.*New_X(1,:);
% NewAlphaRows = (SelectCoL - int)./slope;
```

MATLAB script for generating pyAARON broadband self-noise inputs.

```
%Zero_lift_angle=
interp1(1:length(RelevantAlphaData),RelevantAlphaData,NewAlphaRows);
MachMatrix = MACH_Cl;
CoLMatrix = Cl;
AlphaMatrix = ANGLE_Cl;
SelectMach =Mach_number;
SelectCoL = 0;

Zero_lift_angle=CalcAlpha(MachMatrix, CoLMatrix, AlphaMatrix, SelectMach,
    SelectCoL, 'MinAlphaAllow', -30, 'MaxAlphaAllow', 30);

hFig=figure(1);
subplot(2, 2, 1:2)
set(hFig, 'Position', [50 0 800 500])
plot(X,Y, '.k'); hold on
xlim([0, 1]);
axis equal
grid on
grid minor
xlabel('X', 'FontSize', 14, 'FontWeight', 'bold', 'fontname', 'times')
ylabel('Y', 'FontSize', 14, 'FontWeight', 'bold', 'fontname', 'times')
title(Airfoil, 'FontSize', 14, 'FontWeight', 'bold', 'fontname', 'times')
set(hFig, 'color', 'white');
subplot(2, 2, 3)
plot(X,Y, '.k');hold on
plot(Upper_Line_X(1:2),Upper_Line_Y(1:2), 'ob', 'MarkerFaceColor', 'b');
    hold on
plot(Lower_Line_X(1:2),Lower_Line_Y(1:2), 'or', 'MarkerFaceColor', 'r');
    hold on
plot(Upper_Line_X,Upper_Line_Y, ':', 'color', 'b', 'linewidth', 2); hold on
plot(Lower_Line_X,Lower_Line_Y, ':', 'color', 'r', 'linewidth', 2); hold on
xlim([.99, 1.03]);
axis equal
grid on
grid minor
xlabel('X', 'FontSize', 14, 'FontWeight', 'bold', 'fontname', 'times')
ylabel('Y', 'FontSize', 14, 'FontWeight', 'bold', 'fontname', 'times')

subplot(2, 2, 4)
XB=[-1 1];
YB=[-1 1];

a=plot(XB, YB);
set(a, 'Visible', 'off');
set(a, 'DefaultFigureVisible', 'on');
a.Visible = 'off';
Text_box=['Trailing edge wedge angle, teang (deg) = ',
    num2str(round(Theta,2))];
text(0,.9, Text_box, 'FontSize', 12, 'FontWeight', 'bold', 'fontname', 'times');
axis off
Text_box=['Trailing edge thickness, teth (m) = ',
    num2str(round(Trailing_Edge_Thickness,5))];
text(0,.7, Text_box, 'FontSize', 12, 'FontWeight', 'bold', 'fontname', 'times');
```

MATLAB script for generating pyAARON broadband self-noise inputs.

```
Text_box=['Zero lift angle, ang0 (deg) = ',  
    num2str(round(Zero_lift_angle,5))];  
text(0,.5, Text_box, 'FontSize', 12, 'FontWeight', 'bold', 'fontname', 'times');  
  
Text_box=['Thickness to chord, toc (ND) = ',  
    num2str(round(Airfoil_Thickenss_max,3))];  
text(0,.3, Text_box, 'FontSize', 12, 'FontWeight', 'bold', 'fontname', 'times');
```

Published with MATLAB® R2022b

MATLAB function to find the zero lift angle of attack in an airfoil table.

```
function MyAlpha = CalcAlpha(MachMatrix, CoLMatrix, AlphaMatrix, SelectMach,
    SelectCoL, varargin)
% Calculate MyAlpha based on inputs
% Inputs:
%   - MachMatrix: Matrix of Mach values
%   - CoLMatrix: Matrix of CoL values
%   - AlphaMatrix: Matrix of Alpha values
%   - SelectMach: Selected Mach value
%   - SelectCoL: Selected CoL value
%   - MinAlphaAllow (optional): Minimum allowed Alpha value (default: -30)
%   - MaxAlphaAllow (optional): Maximum allowed Alpha value (default: 30)
% Output:
%   - MyAlpha: Calculated Alpha value

% Create an input parser
parser = inputParser();
parser.KeepUnmatched = true;

% Define default values
defaultMinAlphaAllow = -30;
defaultMaxAlphaAllow = 30;

% Add optional parameter: MinAlphaAllow
addOptional(parser, 'MinAlphaAllow', defaultMinAlphaAllow, @isnumeric);

% Add optional parameter: MaxAlphaAllow
addOptional(parser, 'MaxAlphaAllow', defaultMaxAlphaAllow, @isnumeric);

% Parse the inputs
parse(parser, varargin{:});
MinAlphaAllow = parser.Results.MinAlphaAllow;
MaxAlphaAllow = parser.Results.MaxAlphaAllow;

% Find the closest Mach values
Top2MinLoc = [find(MachMatrix(1,:) <=
    SelectMach,1,'last'),find(MachMatrix(1,:) > SelectMach,1,'first')];
if numel(Top2MinLoc) == 1
    Top2MinLoc = [Top2MinLoc-1,Top2MinLoc];
end
OurMachCols = [min(MachMatrix(1, Top2MinLoc)), max(MachMatrix(1,
    Top2MinLoc))];

% Interpolate CoL and Alpha values
OurCoLCols = CoLMatrix(:, min(Top2MinLoc):max(Top2MinLoc));
NewCoL_Col = interp1(OurMachCols, OurCoLCols', SelectMach);
OurAlphaCols = AlphaMatrix(:, min(Top2MinLoc):max(Top2MinLoc));
NewAlpha_Col = interp1(OurMachCols, OurAlphaCols', SelectMach);

% Filter relevant CoL and Alpha data
RelevantCoLData = NewCoL_Col(NewAlpha_Col < MaxAlphaAllow & NewAlpha_Col >
    MinAlphaAllow);
```

MATLAB function to find the zero lift angle of attack in an airfoil table.

```
RelevantAlphaData = NewAlpha_Col(NewAlpha_Col < MaxAlphaAllow & NewAlpha_Col >
    MinAlphaAllow);

% Find crossing points
CrossZero = find((RelevantCoLData(1:end-1) - SelectCoL) .*
    (RelevantCoLData(2:end) - SelectCoL) <= 0);
CrossZeroXindx = zeros(2, numel(CrossZero));
CrossZeroCoLdata = zeros(2, numel(CrossZero));
CrossZeroCoLdata(1, :) = RelevantCoLData(CrossZero);
CrossZeroCoLdata(2, :) = RelevantCoLData(CrossZero + 1);
CrossZeroXindx(1, :) = CrossZero;
CrossZeroXindx(2, :) = CrossZero + 1;

% Calculate MyAlpha
slope = CrossZeroCoLdata(2, :) - CrossZeroCoLdata(1, :);
int = CrossZeroCoLdata(1, :) - slope .* CrossZeroXindx(1, :);
NewAlphaRows = (SelectCoL - int) ./ slope;

MyAlpha = interp1(1:length(RelevantAlphaData), RelevantAlphaData,
    NewAlphaRows);

end
```

Published with MATLAB® R2022b
