

7-2019

Prediction of Noise Associated with an Isolated UAV Propeller

Samuel O. Afari

Follow this and additional works at: <https://commons.erau.edu/edt>



Part of the [Aerospace Engineering Commons](#)

Scholarly Commons Citation

Afari, Samuel O., "Prediction of Noise Associated with an Isolated UAV Propeller" (2019). *Dissertations and Theses*. 463.

<https://commons.erau.edu/edt/463>

This Thesis - Open Access is brought to you for free and open access by Scholarly Commons. It has been accepted for inclusion in Dissertations and Theses by an authorized administrator of Scholarly Commons. For more information, please contact commons@erau.edu.

PREDICTION OF NOISE ASSOCIATED WITH AN ISOLATED UAV PROPELLER

A Thesis

Submitted to the Faculty

of

Embry-Riddle Aeronautical University

by

Samuel O. Afari

In Partial Fulfillment of the

Requirements for the Degree

of

Master of Science in Aerospace Engineering

July 2019

Embry-Riddle Aeronautical University

Daytona Beach, Florida

PREDICTION OF NOISE ASSOCIATED WITH AN ISOLATED UAV PROPELLER

by

Samuel O. Afari

A Thesis prepared under the direction of the candidate's committee chairman, Dr. Reda R. Mankbadi, Department of Aerospace Engineering, and has been approved by the members of the thesis committee. It was submitted to the School of Graduate Studies and Research and was accepted in partial fulfillment of the requirements for the degree of Master of Science in Aerospace Engineering.

THESIS COMMITTEE

Reda Mankbadi Digitally signed by Reda Mankbadi
Date: 2019.07.16 17:12:15 -04'00'

Chairman, Dr. Reda Mankbadi

Vladimir V. Golubev Digitally signed by Vladimir Golubev
DN: cn=Vladimir Golubev, o=EAU,
ou=Aerospace Engineering,
email=golubed19@grsu.edu, c=US
Date: 2019.07.16 12:08:58 +03'00'

Member, Dr. Vladimir V. Golubev



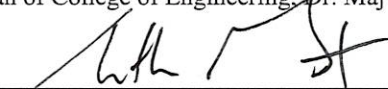
Member, Dr. John Ekaterinaris

M. Attia

Graduate Program Coordinator, Dr. Magdy Attia



Dean of College of Engineering, Dr. Maj Mirmirani



Senior Vice President for Academic Affairs and Provost, Dr. Lon Moeller

7 / 29 / 2019

Date

08/01/2019

Date

8/1/2019

Date

ACKNOWLEDGMENTS

I would like to express my deepest gratitude to my advisor Dr. Reda Mankbadi for his unrelenting support, patience and guidance throughout my study and research. I would also like to thank my committee members Dr. Vladimir Golubev, who took time to present my work at a conference, as well as gave me advise critical to my research, and Dr. John Ekaterinaris for his support and deep knowledge of turbulence flows. I would like to extend my gratitude for the support of the Florida Center for Advanced Aero-propulsion (FCAAP), as well as Embry-Riddle Aeronautical University for granting me access to the high-performance super cluster, VEGA without which this research would not have been possible.

Lastly, I would like to express my gratitude to my family and friends for enduring my arduous talks, scarce communication and long study nights.

TABLE OF CONTENTS

ACKNOWLEDGMENTS	iii
LIST OF TABLES	vi
LIST OF FIGURES	vii
SYMBOLS.....	x
ABBREVIATIONS	xii
ABSTRACT.....	xiii
1. Introduction.....	1
1.1. Motivation.....	1
1.2. Literature Review	1
1.2.1. Classifications.....	1
1.2.2. Review of Research Work	5
1.3. Propeller Noise	9
1.4. Problem Statement & Objectives	12
2. Methodology.....	14
2.1. Governing Equations.....	14
2.1.1. Navier-Stokes Equations.....	14
2.1.2. Reynolds Averaged Navier-Stokes Equations.....	15
2.1.3. Favre-Averaged Equations	16
2.2. Turbulence Modeling.....	17
2.2.1. URANS	17
2.2.2. Modification to Turbulence Model	18
2.2.3. The LES Switch.....	19
2.3. Computational Approach.....	22
2.3.1. Geometry and Grid Generation	22
2.3.2. Solver	27
2.3.3. Boundary Treatment.....	29
2.3.4. Flow Initialization.....	31
3. Far-Field Noise Prediction.....	32
3.1. Integral Methods.....	33
3.1.1. Lighthill's Analogy	33
3.1.2. Kirchhoff's Method.....	34
3.1.3. Surface Integral Formulations	35
3.1.4. Ffowcs Williams-Hawkings Method	36
4. Results and Discussion	41
4.1. Flow Field	41
4.2. Near Field Acoustics	45
4.3. Noise Sources	49
4.3.1. Blade Noise Sources.....	50

4.3.2.	Wake Noise Sources	51
4.4.	Far-Field Acoustics	58
4.4.1.	FWH Results on Coarser Mesh	59
4.4.2.	FWH Results on Fine Mesh.....	65
4.4.3.	Remarks	66
4.5.	Contributions to the Far-Field Noise.....	67
5.	Conclusions.....	77
5.1.	Limitations and Challenges	78
5.2.	Recommendations and Future Work	79
	REFERENCES	80
A.	Modification of the Spalart-Allmaras Turbulence Model	84

LIST OF TABLES

Table 2.1	31
Table 4.1	52

LIST OF FIGURES

<i>Figure 1.1</i> General Atomics MQ-1B predator (Griffs, Wilson, Schneider, & Pierpont, 2009)	2
<i>Figure 1.2</i> Northrup Grumman RQ-4A Global Hawk (Griffs, Wilson, Schneider, & Pierpont, 2009).....	2
<i>Figure 1.3</i> Zipline Medical Delivery Drone (Zipline, 2018).....	3
<i>Figure 1.4</i> DJI Phantom III (DJI, 2018).	4
<i>Figure 1.5</i> Volocopter 2X (Volocopter, 2019).	5
<i>Figure 1.6</i> Experimental Setup (Intaratep, Alexander, Devenport, and Grace, 2016).	6
<i>Figure 1.7</i> Experimental Setup (Zawodny, Boyd, and Burley, 2016).....	7
<i>Figure 1.8</i> Sound Pressure Level Spectra of experiment (Zawodny, Boyd, and Burley, 2016)	8
<i>Figure 1.9</i> Schematic of Rotor Noise (Hubbard, 1991).....	12
<i>Figure 2.1</i> Original 9450 propeller (a) (DJI, 2018) and CAD model (b)	22
<i>Figure 2.2</i> Surface grids (a) and cross-section of grid showing boundary layer resolution (b).....	23
<i>Figure 2.3</i> Location of the rotating zone and baffle	23
<i>Figure 2.4</i> Vertical cross-section of Coarse grid showing grid refinement Regions and extreme coarsening.	24
<i>Figure 2.5</i> Horizontal cross-section of Coarse grid at Y=0 showing grid refinement Regions and extreme coarsening.	25
<i>Figure 2.6</i> Vertical Cut through fine Grid structure	26
<i>Figure 2.7</i> Horizontal Cut through fine Grid at Y=0.....	26
<i>Figure 2.8</i> Control Volume showing neighboring cell centers and face area S_f	28
<i>Figure 2.9</i> Computational domain showing acoustic damping region.	30
<i>Figure 4.1</i> Rotor chord distribution comparison between CAD model and original used by Zawodny et al. (2016).....	42
<i>Figure 4.2</i> Propeller Thrust values compared with data from Zawodny et al. (2016).	42
<i>Figure 4.3</i> Results of velocity magnitude contours (a), compared with that of Yoon & Diaz (b).	43
<i>Figure 4.4</i> Results of vorticity magnitude contours (a), compared with that of Yoon & Diaz (b)	43
<i>Figure 4.5</i> Snapshots of Velocity Magnitude contours	44
<i>Figure 4.6</i> Snapshots of: (a) Vorticity Magnitude and (b) Q-criterion Iso-surfaces colored by vorticity.....	44
<i>Figure 4.7</i> Vertical slice showing snapshot of Vorticity Magnitude overlaid on Dilatation	

Field	45
<i>Figure 4.8</i> Vertical slice (left) and Horizontal slice (right) of the Dilatation Field.....	46
<i>Figure 4.9</i> Snapshot of Acoustic Pressure vertical slice (left), and horizontal cut through rotor plane (right).....	46
<i>Figure 4.10</i> Horizontal snapshot of Dilatation field at in the rotor plane $Y=0$, showing monopole distribution.....	47
<i>Figure 4.11</i> Snapshot of Dilatation fields sliced at $0.3R$	47
<i>Figure 4.12</i> Zoomed in Banded contours of Dilatation fields with focus on dipoles	48
<i>Figure 4.13</i> Linear plots of Dilatation and pressure fluctuation across the monopoles ...	48
<i>Figure 4.14</i> Dilatation Fields with Q-Criterion Iso-contours colored by velocity magnitude.....	49
<i>Figure 4.15</i> Streamlines colored by velocity magnitude	49
<i>Figure 4.16</i> Mean Pressure Distribution on blade surface	50
<i>Figure 4.17</i> Pressure fluctuations (p') distribution on blade surface.....	50
<i>Figure 4.18</i> Vertical and Horizontal Snapshots of u_r , u_θ , and u_z	54
<i>Figure 4.19</i> Snapshots of the normal components of $(\rho u_i u_j)'$	56
<i>Figure 4.20</i> Snapshots of the shear components of $(\rho u_i u_j)'$	57
<i>Figure 4.21</i> Experimental setup showing Far-field location (Intaratep et al. 2016).....	59
<i>Figure 4.22</i> FWH control surfaces considered	60
<i>Figure 4.23</i> Power Spectral Density of top-capped, and open-capped control surfaces ..	61
<i>Figure 4.24</i> Power Spectral Density of top-capped, and both-capped control surfaces...	61
<i>Figure 4.25</i> FWH surfaces at different Radii distances.....	62
<i>Figure 4.26</i> Power Spectral Density plots of the two FWH locations.....	63
<i>Figure 4.27</i> Power Spectral Density of HLU compared with Experimental Data	64
<i>Figure 4.28</i> Power Spectral Density of top-capped, and Blade surface control surfaces.	64
<i>Figure 4.29</i> Directivity plot at 1 st Blade passage frequency.....	65
<i>Figure 4.30</i> Fine Grid Power Spectral Density plot compared with experimental data. ..	66
<i>Figure 4.31</i> Regions of RANS (red) and LES (blue) in computational domain	67
<i>Figure 4.32</i> Time derivative of Mass flux distribution on the FWH surface: (a) Top cap (b) cylinder surface, (c) bottom cap	69
<i>Figure 4.33</i> The Time derivative of the Pressure term distribution on the FWH surface: (a) Top cap (b) cylinder surface, (c) bottom cap	69
<i>Figure 4.34</i> Location of linear slices around the cylinder	70
<i>Figure 4.35</i> Linear Plots comparison between mass flux and pressure derivatives	71

<i>Figure 4.36</i> Horizontal slice of Velocity magnitudes unscaled (left) and scaled (right)..	72
<i>Figure 4.37</i> Velocity Magnitude contours with polar coordinate vectors	72
<i>Figure 4.38</i> Velocity components in polar coordinate system.	73
<i>Figure 4.39</i> Time Derivative of Momentum Flux $\rho u_{r\theta}$ distribution on FWH surface.	74
<i>Figure 4.40</i> Time derivative of Momentum Flux $\rho u_{\theta u}$ distribution on FWH surface.	74
<i>Figure 4.41</i> Time derivative of Momentum Flux $\rho u_{z u}$ distribution on FWH surface.	74
<i>Figure 4.42</i> Linear plots of Time derivatives comparing Loading terms.....	75

SYMBOLS

a_∞, c_0	ambient speed of sound
C_T	coefficient of thrust
e_t	total energy
f_d	shielding function
H_m	Hankel function
J_m	Bessel function
L_i	loading components of vector defined in eq. (3.22)
L_M	$L_i M_i$
L_r	$L_i \hat{r}_i$
\dot{L}_r	time derivative of $L_r, \dot{L}_i \hat{r}_i$
M	local Mach number vector of source, with components M_i
M_r	Mach number of source in radiation direction, $M_i \hat{r}_i$
\hat{n}_i	component of unit outward normal vector to surface
p'	acoustic pressure, $p - p_{time\ average}$
P_{ij}	compressive stress tensor
Pr	Prandtl number
q_j	heat flux
R	rotor radius
r	distance between observer and source, $ x-y $
\hat{r}_i	component of unit vector in radiation direction
Re	Reynolds number
R_i	Richardson number
S_{ij}	strain rate tensor
T_{ij}	Lighthill stress tensor, $\rho' u_i u_j + (p' - c_0^2 \rho') \delta_{ij} - \sigma_{ij}$
T	ambient temperature
t	observer time
U_i	components of vector defined in eq. (3.22)
U_n	$U_i \hat{n}_i$
$U_{\hat{n}}$	$U_i \hat{n}_i$
\dot{U}_n	time derivative of $U_n, \dot{U}_i \hat{n}_i$
u_i	components of local fluid velocity
u_n	normal component of local fluid velocity
v_n	local normal velocity of source surface
w_p	wave speed
δ_{ij}	Kronecker delta, 1 for $i=j$, otherwise 0
κ	karman constant
μ	laminar viscosity
ν_t	turbulent eddy viscosity
$\tilde{\nu}$	modified eddy viscosity
ρ	density
σ_{ij}, τ_{ij}	viscous stress tensor
τ	source time

ϕ	mass flux
Ψ	low Reynolds number correction term
Ω	vorticity magnitude

ABBREVIATIONS

AMI	Arbitrary Mesh Interface
ANOPP	Aircraft Noise Prediction Program
APC	Advanced Precision Composites
BARC	Broadband Acoustic Rotor Codes
BPF	Blade Passage Frequency
BVI	Blade Vortex Interaction
CAA	Computational Aeroacoustics
CAD	Computer Aided Drawing
CFD	Computational Fluid Dynamics
DES	Detached Eddy Simulation
DDES	Delayed Detached Eddy Simulation
DNS	Direct Numerical Simulation
FAA	Federal Aviation Administration
FWH	Ffowcs-Williams-Hawkings
HLU	Hybrid LES/URANS
LEE	Linearized Euler Equations
LES	Large Eddy Simulation
MAV	Micro Air Vehicles
MSD	Modeled Stress Depletion
NACA	National Advisory Committee for Aeronautics
NASA	National Aeronautics and Space Agency
NSCBC	Navier-Stokes Characteristic Boundary Condition
OpenFOAM	Open Field Operations And Manipulations
OASPL	Overall Sound Pressure Level
PISO	Pressure-Implicit with Splitting of Operators
RANS	Reynolds Averaged Navier-Stokes Solver
RPM	Revolutions Per Minute
SALT	Structural Acoustic Loads & Transmission
SIMPLE	Semi-Implicit Method for Pressure-Linked Equations
SPL	Sound Pressure Level
TKE	Turbulent Kinetic Energy
UAM	Urban Air Mobility
UAV	Unmanned Aerial Vehicle
URANS	Unsteady RANS

ABSTRACT

Afari, Samuel Okyere MSAE, Embry-Riddle Aeronautical University, July 2019.

Prediction of Noise Associated with an Isolated UAV propeller.

The emergent field of interest in the Urban Air Mobility community is geared towards a world where aerial vehicles are commonplace. This poses the problem of the effects of the radiated noise. The present research presents an in-depth analysis of the noise generation mechanism of a propeller as a mode of propulsion of the said aerial vehicles. Numerical simulation utilizing a Hybrid Large-Eddy Simulation (LES) coupled with Unsteady Reynolds-Averaged Navier-Stokes (RANS) solver, is adopted on an isolated propeller modeled from the commercial DJI Phantom II 9450 propeller. The Spalart-Allmaras one equation turbulence model with rotation/curvature correction is used. The Farassat's 1A formulation of the Ffowcs-Williams-Hawkings equations are used with an off-body permeable porous stationary control surface for far-field noise predictions. The current results are found to be in good agreement with several observations including the thrust generated, the unsteady flow structure, and the radiated far-field sound spectra and directivity. A deeper study into the contributing sources of the noise generation both on the propeller surface, as well as in the swirling wake flow is performed.

1. Introduction

1.1. Motivation

Urban Air Mobility (UAM) vehicles are a broad spectrum of aerial vehicles that range from the largest commercial aircrafts, to the smallest Micro Air Vehicles (MAV). There has been a surge in the innovation and development of this scope. One of the main directions of this innovation is in the incorporation of these urban air mobility vehicles in the communities as a day-to-day vehicle as automotive vehicles are today. One of the major concerns in this line of innovation is the effects of the noise levels in the cities.

The need to address UAM noise is highlighted in several recent NASA workshops at Glenn Research Center and at Langley Research Center. Several industries are developing Unmanned Aerial Vehicles (UAV) for various new applications, such as aerial taxis (Textron, 2018). With the increase in the usage of UAV, the effect of its radiated noise on the community need to be minimized and regulated. Furthermore, in several military uses of UAV (e.g. reconnaissance) suppressing the noise is a must. The Federal Aviation Administration (FAA), as well as other international aviation regulators have imposed stringent regulations on aircraft noise. For these regulations to be met, UAM vehicles would have to be significantly quieter.

1.2. Literature Review

1.2.1. Classifications

Generally, Urban Air Mobility is a term coined for a highly automated passenger or cargo-carrying air transportation services. Under this umbrella, UAVs can be classified in terms of size, type and endurance. These are collectively classified based on their mode of propulsion.

Gas Turbines and Rockets

Pertaining mostly to the military are the large UAVs. These are usually fixed wing configuration and have a longer range and endurance. The modes of propulsions typically include rocket power as shown in *Figure 1.1* below of the General Atomics MQ-1B predator UAV. The most common mode of propulsion in this class of UAVs is the gas turbine engine, which typically has an endurance of about 32 hours (Griffs et al., 2009). An example of this class of UAV is the Northrup Grumman RQ-4A Global Hawk shown in *Figure 1.2*.



Figure 1.1 General Atomics MQ-1B predator (Griffs, Wilson, Schneider, & Pierpont, 2009)



Figure 1.2 Northrup Grumman RQ-4A Global Hawk (Griffs, Wilson, Schneider, & Pierpont, 2009)

Propellers

On the other spectrum of the classification are the small-scale UAVs. These class of vehicles are usually powered by propellers driven by electric motors and batteries, which currently have maximum endurance of up to 30 minutes of flight time. These are quite prominent commercially and are used in several industries. There are quite a few fixed-wing configurations under this classification on the market. Companies such as zipline, operate a fixed-wing UAV seen in *Figure 1.3* (Zipline, 2018), which delivered medical supplies to remote areas of regions in Rwanda and Ghana. The most common configuration is the rotary wing or multi-rotor UAVs such as the commercial DJI Phantom III seen in *Figure 1.4* (DJI, 2017). A great advantage of this configuration is its vertical takeoff and landing capabilities, which makes it ideal for flights in the cities. These types of drones are used in industries such as the movie industry for aerial shots, as well as construction for visual investigation of buildings. There has been some implementation of these configurations in firefighting by attaching thermal cameras to study the nature of the fires (Dslrpros, 2018). Companies like Amazon are also experimenting with package deliveries using these UAV types.



Figure 1.3 Zipline Medical Delivery Drone (Zipline, 2018).



Figure 1.4 DJI Phantom III (DJI, 2018).

As aforementioned, these multi-rotor configurations have the advantage of vertical takeoff, which become favorable in the cities and communities. With the successes of this configuration, a sizeable number of companies are developing scaled vehicles with VTOL capabilities, which would replace or augment modern transportation systems. These vehicles are essentially going to be scaled for human transportation. Companies such as Uber have already started an initiative called Uber Elevate (Uber, 2016) with which they have collaborated with several big named companies to design concepts and prototypes for this kind of UAM vehicles. Some have already started testing such as the Volocopter air taxi shown in *Figure 1.5*. One of the main targets set by this initiative is to address the issue of noise. According to Uber, VTOL vehicles operating from vertiports should approach noise levels of half that of a truck travelling on a residential road (75-80 dB(A) at 50 feet), which is approximately 62dB at 500ft altitude (Uber Elevate, 2016). An important point to note is that these vehicles would be operating in large number like cars driving in cities. This would significantly increase their overall noise levels and annoyance to the general populace. As such, it is imperative that the noise mechanism of multi-rotors is studied.



Figure 1.5 Volocopter 2X (Volocopter, 2019).

1.2.2. Review of Research Work

Several investigators have studied the noise associated with UAV. On the experimental side, Intaratep et al (2016) studied the quadcopter noise and aerodynamic performance. Static thrust and acoustic measurements were performed using the commercial DJI phantom II UAV at different rotor speeds as well different modes; that is 1, 2 and 4 rotors. Their results showed that operating at 4 rotor configurations, the spectral signature of the rotors was dominated by high and sustained tones at the blade passing frequency up to 6000Hz without deterioration. They also reported little to no difference in acoustic signatures for 1, 2 or 4 rotors with about 1-2dB change in OASPL at mid to high frequency range at same thrust setting. They noted the contribution of motor noise to be noticeable in the mid-frequency range. Their work shed more light in the acoustic signature of multi-rotor small scale UAVs.

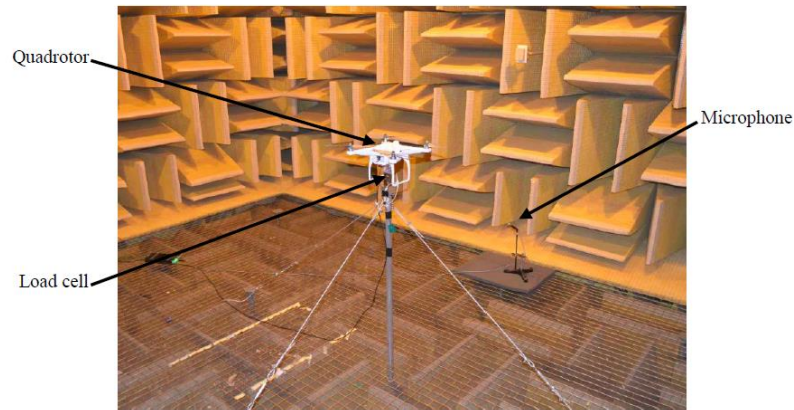


Figure 1.6 Experimental Setup (Intaratep, Alexander, Devenport, and Grace, 2016).

Lu et al (2016) and Feight et al (2016) also attempted to characterize the multi-rotor UAV noise and explored some noise-reduction technologies. Lu et al. researched on the use of ducts with acoustic absorption materials in an attempt to reduce the noise, but their results showed deleterious effects and gave some suggestions to improve the damping effects of using ducts. The aforementioned research works were performed on a full-body UAV, as such the effects of fluid-body interaction with the body of the vehicle was not discussed. To study the contribution of noise of the rotor only, we look at work done by Zawodny et al (2016). They performed an experimental study on an isolated rotor. Two rotor blades were tested, one of which was the original 9443 DJI Phantom III rotor blade, and the other was Advanced Precision Composites (APC) 11X4.7SF rotor. With their Structural Acoustic Loads and Transmission (SALT) chamber facility, they performed thrust and acoustic measurements using 5 microphones as shown in *Figure 1.7*. They classified the noise structure of the rotors into deterministic noise, which comprises tonal components such as loading and thickness noise, as well as Blade vortex interaction noise, and non-deterministic noise which is primarily broadband noise. Their results also confirmed the effects of motor noise to be in the mid-frequency range. They noted

significant differences in OASPL between the two rotors tested. They attributed the difference to the fact that the rotor tip speeds were different between the two blades. Acoustic directivity was also performed with their array of microphones and was seen to have the loudest noise occurring at a directivity of approximately 28 degrees below the rotor plane. Zawodny et al (2016) also performed numerical predictions using NASA's OVERFLOW code, with Unsteady RANS numerical simulation and overset gridding method. The one equation Spalart-Allmaras turbulence model was employed. Acoustic predictions were performed using ANOPP-PAS for the tonal or "deterministic", and a semi-empirical frequency domain broadband noise prediction based on airfoil self-noise were performed using BARC.

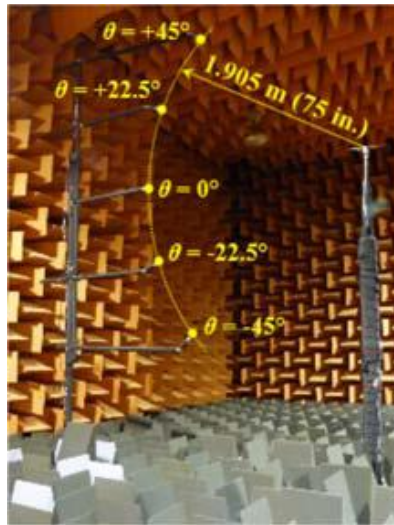


Figure 1.7 Experimental Setup (Zawodny, Boyd, and Burley, 2016).

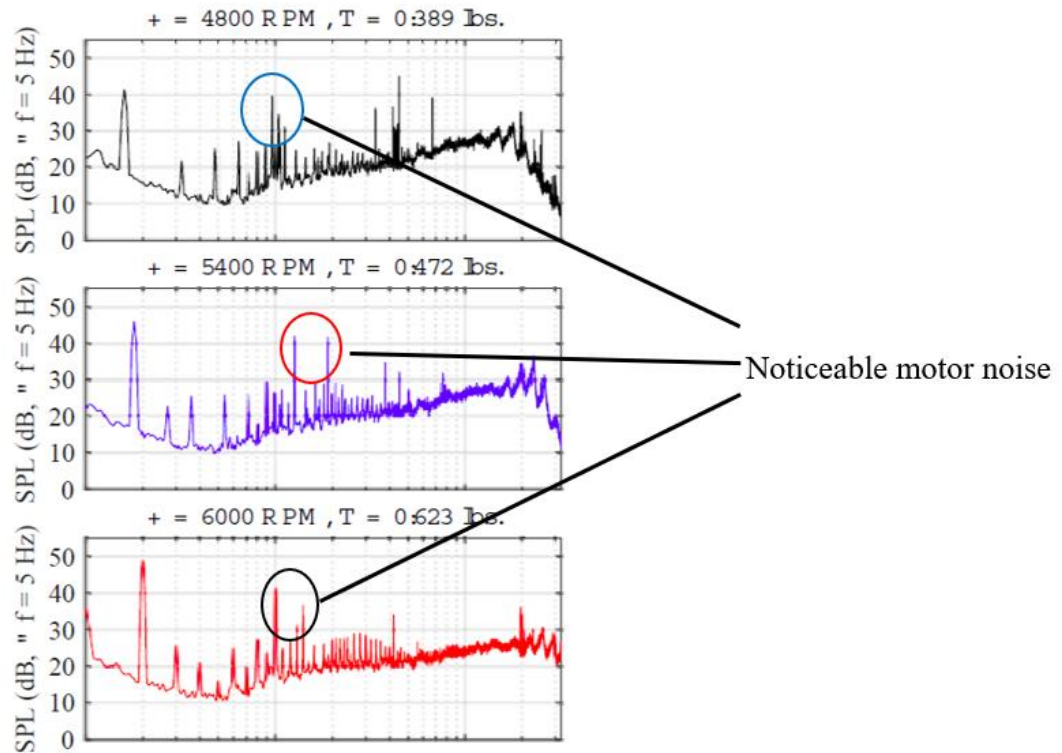


Figure 1.8 Sound Pressure Level Spectra of experiment (Zawodny, Boyd, and Burley, 2016)

Yoon & Diaz (2018) conducted high-fidelity simulations of the flow field associated with the commercial drone DJI III, as well as the SUI Endurance quadcopter, and considered a single rotor, as well as, multi-rotor interactions. The effects of weather were also discussed. They utilized a hybrid RANS/LES approach with the Spalart-Allmaras turbulence model. They showed that this model was enough in prediction the complex flow fields of the isolated rotor and was able to accurately predict the figure of merit. Perez & Lopez (2017) studied computationally the wake of a quadcopter propeller in hover. An isolated rotor was used in their study. A hybrid RANS/LES method and two turbulence models – the Spalart-Allmaras model and the $k-\omega$ models were tested. Their results indicated identical results between the two turbulence models with the exception in the turbulence viscosity field in the far wake.

1.3. Propeller Noise

UAV noise are produced by several sources including the propellers, the vehicle's flow acoustics, the motor noise, and the interaction among these sources. As a starting point, we focus here on the noise produced by an isolated UAV propeller in hover conditions. Historically, there has been quite an extensive work in the field of rotor and propeller noise. Some of the earliest theories in this field is probably Gutin's theory of the noise produced by the rotating pressure field of propellers developed in 1937 (Gutin, 1948). To discuss the nature of rotating propeller noise, we attempt to define Aerodynamic Noise and the sources as pertaining to propellers.

Aerodynamic noise is defined by Marte and Kurtz (1970) as sound generated due to the relative motion between a solid body or stream of fluid and the surrounding medium. Traditionally, propeller noise has been separated into rotational and vortex components (Marte & Kurtz, 1970). Rotational noise describes all sound which accounts for the deterministic components, with discrete frequencies occurring at harmonics of the blade passage frequencies. These are generally subdivided into thrust and torque noise (blade loading), and thickness noise. Theoretical works such as those by Gutin (1948) have been shown to estimate the sound pressure at the far-field using the equation:

$$p_m = \frac{169.3mBRM_t}{SA} \left[\frac{0.76P_h}{M_t^2} - T \cos\theta \right] J_{mB}(x) \quad (1.1)$$

where:

p = rms sound pressure level (SPL)

m = order of the harmonic

S = distance from propeller hub to observer

R = propeller radius

A = propeller disc area

P_h = absorbed power, horsepower

T = thrust

B = blade count

M_t = tip Mach number

J_{mB} = Bessel function of order mB

θ = angle between propeller axis and observer

The expression above gives good agreement with experimental results for the first few harmonics, and as such gives a good estimation for overall noise of propellers operating at moderate tip speeds. They do however tend to diverge at tip Mach numbers between 0.5 and 0.3 by over predicting the noise levels. Other methods such as unsteady RANS simulations have been shown to be able to predict these harmonics quite well (Zawodny et al, 2016). At these relatively low tip speeds, the vortex noise and turbulence induced noise, which are subsets of the broadband noise becomes quite relevant.

Vortex noise is generally defined as sound generated by the formation and shedding of vortices in the flow past a blade (Marte & Kurtz, 1970). These are a function of the span-wise velocity along the blade and generate the broadband of the shedding frequencies. This produces a dipole structure of acoustic radiation in which the strength of the source is proportional to sixth power of the section velocity (Hubbard & Regier, 1950). With this knowledge, the frequencies associated with the blade section at the tip tends to be highest in amplitude. Also, because the blade generates thrust, the dipole acoustic radiation combines with that from the trailing edge vortices, which make up the vortex noise. This is stipulated in literature (Hubbard & Regier, 1950) to be the most

significant source of broadband noise.

The other significant source of broadband noise of propellers is the turbulence-induced noise, which is noise generated by the motion of small-scale turbulence which is quadrupole in nature and can be quite significant for high speed flows. In low speed flows, there might be considerable amplification of these weak noise generation sources due to their interaction with the pressure field of the moving blade. This induces an acoustic radiation of a more efficient dipole type (Marte & Kurtz, 1970).

In general literature, the rotating blades are often divided into propellers, rotors, and fans. Here, we use “propellers” due to the unique regime of UAV blades. Rotors are generally described to pertain to helicopters, which may be different from that of a typical propeller due to the scale difference as well as the flow regimes in which they operate. The flow regimes are commonly defined using two non-dimensional parameters: the chord Reynolds number at 75% span, and the tip Mach number. These are defined as:

$$Re_c(0.75R) = \frac{\rho_\infty * \Omega * 0.75R * c}{\mu_\infty} \quad (1.2)$$

$$M_{tip} = \frac{\Omega * R}{a_\infty} \quad (1.3)$$

Where Ω the rotational speed in rad/s, R is the radius of the propeller, a_∞ is the speed of sound, ρ_∞ is the free-stream density, c is the blade chord, and μ_∞ is the free-stream air dynamic viscosity. Tip speeds of helicopter are usually in the range of $0.7 \leq M_{tip} \leq 0.8$ whereas that of small-scale UAVs tend to be in the range of $0.15 \leq M_{tip} \leq 0.3$. The chord Reynolds numbers of helicopters are of orders of 10^6 while those of small-scale UAV are about $10^4 - 10^5$. There is a lack of dynamic similarity between the two vehicles. Even in hover conditions, the rotors were seen to have significantly higher

noise levels compared with propellers in the higher frequencies (Goldstein, 1974). It was shown by Simons (1966) that significant load variations exist even in hover. A good overview of the legacy works in noise predictions of rotors have been detailed by Goldstein (1974). This further warrants an investigation into which regime the propellers used in the small-scale UAVs fall.

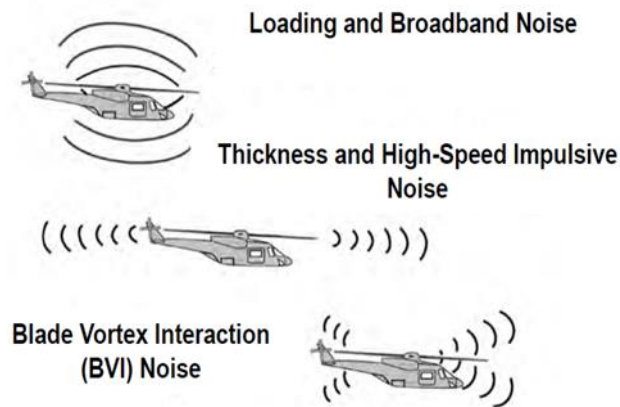


Figure 1.9 Schematic of Rotor Noise (Hubbard, 1991)

One of the goals of this research is to determine if there is significant difference in the acoustic signatures of the isolated UAV propellers, as well as the capabilities of using high-fidelity computational methods for noise predictions in this flow regime.

1.4. Problem Statement & Objectives

The emergent innovations of UAM technologies are gearing towards a world of aerial vehicles capable of vertical takeoff and landing. With these technologies lies the utilization of some form of rotor or propeller configuration. As previously discussed, one of the major plagues of this technology is the generation of noise from these propellers. These UAM vehicles would typically have more than one rotor or propeller, further increasing the acoustic signature of the vehicle. The goals of this thesis are:

- To first study numerically, the flow structures and to be able to accurately predict the flow field of an isolated propeller.
- Identify the noise generation sources of the propeller, both on the blade, and the wake.
- Accurately predict the noise generated by the isolated propeller using High-Fidelity CFD with acoustic extension techniques.

2. Methodology

To compute the flow-field of the isolated propeller, several computational approaches can be used, from the lower accuracy unsteady Reynolds-Averaged Navier-Stokes (URANS) to the highest accuracy, but computationally expensive Direct Numerical Simulations (DNS). For this research, we employ a Hybrid Large-Eddy Simulation/Unsteady Reynolds-Averaged Navier-Stokes (HLU) method, which intermediates between accuracy and computational cost (Mankbadi, Hixon, & Povinelli, 2000). In this chapter, the governing equations for this hybrid approach is discussed, the turbulence models and modifications are also provided. The computational approach, that is the geometry and grid generation process as well as the solver and boundary treatments used are detailed.

2.1. Governing Equations

In the computational domain, the unsteady flow field and the near-field acoustic field are directly resolved numerically. A Hybrid LES-URANS (HLU) approach is adopted here similar to the one used earlier by Mankbadi et al. (2000; 2016) in which URANS is used near the solid surfaces, where the LES resolution required to resolve the boundary layer is prohibitive. On the other hand, LES is used elsewhere.

2.1.1. Navier-Stokes Equations

The governing equations for the entire computational domain are the compressible, unsteady Navier-Stokes equations (Mankbadi, 1994):

$$\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x_j} (\rho u_j) = 0 \quad (2.1)$$

$$\frac{\partial}{\partial t}(\rho u_i) + \frac{\partial}{\partial x_j}[\rho u_i u_j + p \delta_{ij} - \tau_{ij}] = 0 \quad (2.2)$$

$$\frac{\partial}{\partial t}(\rho e_t) + \frac{\partial}{\partial x_j}[\rho u_j e_t + p u_j + q_j - u_j \tau_{ij}] = 0 \quad (2.3)$$

$$\tau_{ij} = 2\mu S_{ij} - \frac{1}{3} \frac{\partial u_k}{\partial x_k} \delta_{ij}; \quad e_t = e + \frac{u_k u_k}{2} \quad (2.4)$$

Where e_t is the total energy, q_j is the heat flux, and τ_{ij} is the viscous stress term, and S_{ij} is the strain rate tensor, with:

$$S_{ij} = \frac{1}{2} \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \quad (2.5)$$

$$q_j = -k \frac{\partial T}{\partial x_j} \equiv -C_p \frac{\mu}{Pr} \frac{\partial T}{\partial x_j} \quad (2.6)$$

Pr is the Prandtl number, and k is the thermal conductivity of the gas. μ is the laminar dynamic viscosity of air which is calculated from the Sutherland's law of viscosity:

$$\mu(T) = \mu_o \frac{T_o + C}{T + C} \left(\frac{T}{T_o} \right)^{3/2} \quad (2.7)$$

where μ_o , T_o , and C are the reference dynamic viscosity, temperature, and Sutherland's constant respectively for air, with values of $1.716E - 5 \frac{kg}{ms}$, $273.15K$, and $110.4K$ respectively.

2.1.2. Reynolds Averaged Navier-Stokes Equations

The Navier-Stokes equations (2.1-2.3) are most often not implemented in its entirety due to computational costs. Instead, the technique of averaging of the fluid variables is performed to divide the equations into steady and fluctuating or unsteady components.

The time average of a continuous signal of a field variable, ϕ , at any point, x is given by:

$$\overline{\phi(x)} = \frac{1}{t_2 - t_1} \int_{t_1}^{t_2} \phi(x, t) dt \quad (2.8)$$

The instantaneous or discrete signal of the fluid variable is then defined as,

$$\phi(x, t) = \overline{\phi(x)} + \phi'(x, t) \quad (2.9)$$

This decomposition is then applied to the fluid variables in the Navier-Stokes equations to constitute the incompressible Reynolds Averaged Navier-Stokes (RANS) equations:

$$\frac{\partial u_j}{\partial x_j} = 0 \quad (2.10)$$

$$\rho \frac{\partial u_j u_i}{\partial x_i} = -\frac{\partial p}{\partial x_i} + \rho \frac{\partial}{\partial x_j} (2\mu S_{ij} - \overline{u'_j u'_i}) \quad (2.11)$$

2.1.3. Favre-Averaged Equations

When dealing with compressibility effects such as that of the current research, it is often convenient to use a mass-weighted time average for the Navier-Stokes Equations. Since density is no longer constant, it is also included in the Reynolds decomposition process. The time averaging procedure described in 2.1.8 is then reformulated as:

$$\tilde{\phi}(x) = \frac{1}{\bar{\rho}(t_2 - t_1)} \int_{t_1}^{t_2} \rho(x, t) \phi(x, t) dt \quad (2.12)$$

The discrete decomposition in equation 2.1.9 is also reformulated as:

$$\phi_i = \tilde{\phi}_i + \phi_i'' \quad (2.13)$$

Applying this to the above Navier-Stokes Equations, the Favre-averaged Navier-Stokes Equations become (Mankbadi, 1994):

$$\frac{\partial \bar{\rho}}{\partial t} + \frac{\partial}{\partial x_i} [\bar{\rho} \tilde{u}_i] = 0 \quad (2.14)$$

$$\frac{\partial}{\partial t}(\bar{\rho}\tilde{u}_i) + \frac{\partial}{\partial x_j}[\bar{\rho}\tilde{u}_j\tilde{u}_i + \bar{p}\delta_{ij} - \tau_{ji}^{tot}] = 0 \quad (2.15)$$

$$\frac{\partial}{\partial t}(\bar{\rho}\tilde{e}_o) + \frac{\partial}{\partial x_j}[\bar{\rho}\tilde{u}_j\tilde{e}_o + \bar{p}\tilde{u}_j + q_j^{tot} - \tilde{u}_i\tau_{ji}^{tot}] = 0 \quad (2.16)$$

where

$$\tau_{ji}^{tot} = \tau_{ji}^{lam} + \tau_{ji}^{turb} \quad (2.17)$$

$$\tau_{ji}^{turb} \equiv -\overline{\rho u'_i u'_j} \approx 2\nu_t \left(S_{ij} - \frac{1}{3} \frac{\partial u_k}{\partial x_k} \delta_{ij} \right) - \frac{2}{3} \bar{\rho} k \delta_{ij} \quad (2.18)$$

The term ν_t is the turbulent eddy viscosity = $\rho\mu_t$, and k is the modeled turbulent kinetic energy (TKE) which is a measure of the energy contained in a turbulent flow, and is defined as:

$$k = \frac{1}{2} (\overline{u'_1 u'_1} + \overline{u'_2 u'_2} + \overline{u'_3 u'_3}) \quad (2.19)$$

2.2. Turbulence Modeling

With the HLU approach, URANS is computed in the near wall region, whereas an LES-like model is solved away from the wall. The goal of turbulence modeling is to get an approximate solution for the stress term τ_{ij} in the above governing equations.

2.2.1. URANS

The turbulent viscous stresses τ_{ij} are modeled using the one equation Spalart-Allmaras turbulence model (Spalart & Allmaras, 1992). The Spalart-Allmaras model has been shown to accurately predict the flow of small-scale rotors (Yoon & Diaz, 2018), and as such, is utilized in this study. The model uses the Boussinesq approximation, which relates the Reynolds stresses to the turbulent eddy viscosity. The eddy viscosity ν_t is obtained by first calculating a modified eddy viscosity ($\tilde{\nu}$) through the transport equation:

$$\frac{\partial \tilde{v}}{\partial t} + u_i \frac{\partial \tilde{v}}{\partial x_j} = \underbrace{c_{b1}(1 - f_{t2})\tilde{S}\tilde{v}}_{\text{Production}} - \underbrace{\left[c_{w1}f_w - \frac{c_{b1}}{\kappa^2}f_{t2} \right] \left(\frac{\tilde{v}}{d} \right)^2}_{\text{Destruction}} + \underbrace{\frac{1}{\sigma} \left[\frac{\partial}{\partial x_j} \left((\nu + \tilde{v}) \frac{\partial \tilde{v}}{\partial x_j} \right) + c_{b2} \frac{\partial \tilde{v}}{\partial x_i} \frac{\partial \tilde{v}}{\partial x_j} \right]}_{\text{Dissipation}} \quad (2.20)$$

Where ν is the kinematic viscosity, and the closure functions are defined as:

$$\tilde{S} = \Omega + \frac{\tilde{v}}{\kappa^2 d^2} f_{v2}; \quad (2.21)$$

\tilde{S} is the mean strain rate and Ω is the vorticity;

$$f_{v1} = \frac{\chi^3}{\chi^3 + c_{v1}^3}; \quad \chi = \frac{\tilde{v}}{\nu} \quad (2.22)$$

$$f_{v2} = \frac{\chi}{1 + \chi f_{v1}}; \quad f_{t2} = c_{t3} \exp(-c_{t4} \chi^2) \quad (2.23)$$

$$c_{w1} = \frac{c_{b1}}{\kappa^2} + \frac{1 + c_{b2}}{\sigma}; \quad f_w = g \left[\frac{1 + c_{w3}^6}{g^6 + c_{w3}^6} \right]^{1/6} \quad (2.24)$$

$$g = r + c_{w2}(r^6 - r); \quad r = \min \left[\frac{\tilde{v}}{\tilde{S} \kappa^2 d^2}, 10 \right] \quad (2.25)$$

Here κ is the Karman constant. The eddy viscosity is computed as $\nu_t = \tilde{v} f_{v1}$. The turbulence length scale d seen in equations 2.21 and 2.25 is defined as the distance to the nearest wall. The model coefficients are given as:

$$c_{b1} = 0.1355; \quad c_{b2} = 0.622; \quad \kappa = 0.41, \quad \sigma = 2/3; \quad c_{t3} = 1.2; \quad c_{t4} = 0.5; \quad c_{w2} = 0.3, \quad \text{and} \quad c_{w3} = 2.$$

2.2.2. Modification to Turbulence Model

To account for rotation and curvature of the wake physics, a rotation/curvature correction term f_{r1} is multiplied by the production term of the transport equation as suggested by Shur et. al (Shur, Strelets, Travin & Spalart, 2000):

$$f_{r1}(r^*, \hat{r}) = (1 + c_{r1}) \frac{2r^*}{1 + r^*} [1 - c_{r3} \tan^{-1}(c_{r2} \hat{r})] - c_{r1} \quad (2.26)$$

Where:

$$r^* = S/\omega; \quad S = \sqrt{2S_{ij}S_{ij}}; \quad \text{and} \quad \omega = \sqrt{2\omega_{ij}\omega_{ij}} \quad (2.27)$$

$$S_{ij} = 0.5 * \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right); \text{ and } \omega_{ij} = 0.5 * \left(\left(\frac{\partial u_i}{\partial x_j} - \frac{\partial u_j}{\partial x_i} \right) + 2\varepsilon_{mji}\Omega_m \right) \quad (2.28)$$

$$\hat{r} = \frac{2\omega_{ik}S_{jk}}{D^4} \left(\frac{DS_{ij}}{Dt} + (\varepsilon_{imn}S_{jn} + \varepsilon_{jmn}S_{in})\Omega_m \right); \text{ with } D^2 = 0.5(S^2 + \omega^2) \quad (2.29)$$

And DS_{ij}/Dt are the components of the Lagrangian derivative of the strain tensor, and ε is the Einstein summation convention. The constants used here are $c_{r1}=1.0$, $c_{r2}=12$, and $c_{r3}=1.0$.

Looking at the above modifications, we see the modified terms in equation 2.29 requires that a Lagrangian derivative be computed. This increases the complexity of the computation and thus takes more time to compute. Also, these modifications become quite cumbersome to program. As such, a simplification is suggested by Zhang and Yang (2013) to overcome this complication by employing the use of the Richardson number Ri defined by Hellsten (1998):

$$Ri = \frac{\omega}{S} \left(\frac{\omega}{S} - 1 \right) \quad (2.30)$$

The Richardson number is a parameter that represents a measure of the mean-flow deformation, thus can incorporate the effects of rotation and curvature. The rotation/curvature factor \hat{r} (equation 2.29) is recomputed as:

$$\hat{r} = \frac{\omega}{S} \left(\frac{\omega}{S} - 1 \right) \quad (2.31)$$

The above modifications were implemented in OpenFOAM CFD software.

2.2.3. The LES Switch

To switch from URANS to LES, we adopt Spalart et al (1997) approach in modifying the length scale d used in the equations 2.20, 2.21, and 2.25 to be replaced by the

switching function \tilde{d} , defined as:

$$\tilde{d} = \min(d, C_{DES}\Delta) \quad (2.32)$$

With d defined as the wall distance, and Δ defined as the local grid size = $\max(\Delta_x, \Delta_y, \Delta_z)$, and $C_{DES} = 0.65$. The switching mechanism works such that URANS is solved in the near wall region where $d < C_{DES}\Delta$, and solves LES in the outer regions where $d \geq C_{DES}\Delta$, that is:

- In the inner layer, ($d < C_{DES}\Delta$) $\rightarrow \tilde{d} = d$
- In the outer layer, ($d \geq C_{DES}\Delta$) $\rightarrow \tilde{d} = C_{DES}\Delta$

When the model is switched, the eddy viscosity becomes proportional to the local deformation rate and the turbulence length scale \tilde{d} , that is, $\nu_t \propto S\tilde{d}$. The modified transport equation 2.20 is then solved using the new length scale. It should be noted that the velocity term u_i changes to the filtered \bar{u}_i . This modification transforms the transport equation into a grid-dependent sub-grid scale-like model. The turbulence length scale \tilde{d} acts as an implicit filter, where the larger eddies are resolved, and the smaller eddies are modeled. Thus, the transport equation for the eddy viscosity, equation 2.20 behaves as a Smagorinsky-like sub-grid scale model in the outer layer and becomes grid dependent.

“Wall proximity” is calculated by the ratio of the eddy viscosity and the molecular viscosities. In the LES region, the sub-grid eddy viscosity decreases with grid refinement, and thus decreases the flow Reynolds number. To avoid this misinterpretation, a low Reynolds number correction term ψ is introduced into the switching function:

$$\tilde{d} = \min(d, \psi C_{DES}\Delta) \quad (2.33)$$

Where

$$\psi^2 = \min \left[10^2, \frac{1 - \frac{1 - C_{b1}}{C_{w1} \kappa^2 f_w^*} [f_{t2} + (1 - f_{t2}) f_{v2}]}{f_{v1} \max(10^{-10}, 1 - f_{t2})} \right] \quad (2.34)$$

In conditions where near wall grids are highly refined, that is $\delta > \Delta$, the switch might tend to prematurely switch to LES within the boundary layer. When this happens, there would be an imbalance of eddy viscosity in the boundary layer as the length scales of the turbulence would not be accurately resolved due to grid resolution deficiency. This causes a phenomenon call Modeled Stress Depletion (MSD), which would under-predict the turbulence generation from the boundary layer (Spalart et al, 2006). To avoid this, modifications to the switching function is suggested by Spalart et al (2006) is implemented called Delayed Detached Eddy Simulation (DDES).

$$\tilde{d} = d - f_d \max(0, d - \psi C_{DES} \Delta); \quad (2.35)$$

$$f_d = 1 - \tanh((8r_d)^3), \quad (2.36)$$

$$r_d = \frac{\nu + \nu_t}{\sqrt{U_{ij} U_{ij}} \kappa^2 d^2}; \kappa = 0.41 \quad (2.37)$$

By modifying the switching function, the boundary layer is “shielded” from the switching to LES using the shielding function f_d . The argument r_d is close to unity in the viscous sub-layer, and asymptotes to zero as the edge of the boundary layer is approached. As a result, f_d assumes a zero value within the boundary layer, and rapidly grows to 1 as r_d become far less than 1.

2.3. Computational Approach

2.3.1. Geometry and Grid Generation

The 9450 propellers of the commercial UAV DJI Phantom II (DJI, 2018) is used in this study. It has a diameter of $0.239m$, a C_{tip} of $0.01m$, and a pitch of approximately 12° . A CAD model was created based on the general dimensions and twist of the propeller. Catia V5 software is utilized in the CAD generation process. The blade geometry is made with a blended series of airfoil sections – NACA 6412 with pitch angle of $\sim 6^\circ$ for the root and tip, and NACA 8306 with $\sim 12^\circ$ pitch angle at the larger mid-section. A comparison between the Original blade and the modeled blade is shown below:



Figure 2.1 Original 9450 propeller (a) (DJI, 2018) and CAD model (b)

Two grids are generated for this study. A “coarse” unstructured grid, and a finer structured grid. The grids are generated using the commercial software Pointwise v18. The surface grid was generated using an unstructured mesh with advancing forth orthogonal grid method, which creates quadrilateral-dominant cells. A cylinder is created around the propeller, and an unstructured block is generated inside the cylinder with Pointwise’s “T-rex” grid grown off the surface of the propeller *Figure 2.2b*. This cylinder is implemented as a baffle zone for mesh rotation (see *Figure 2.3*). Both grids utilize the same surface grid refinement, and are wall-resolved, with a wall spacing of

$4.67 \times 10^{-5} m$, as well as the same rotating region refinement. For a rotational speed of 6000rpm, this corresponds to a non-dimensional y^+ of 10.

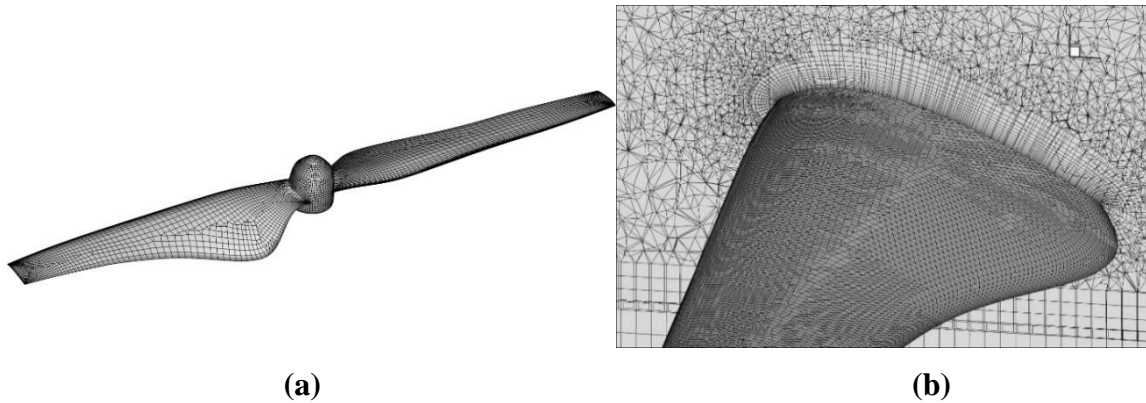


Figure 2.2 Surface grids (a) and cross-section of grid showing boundary layer resolution (b)

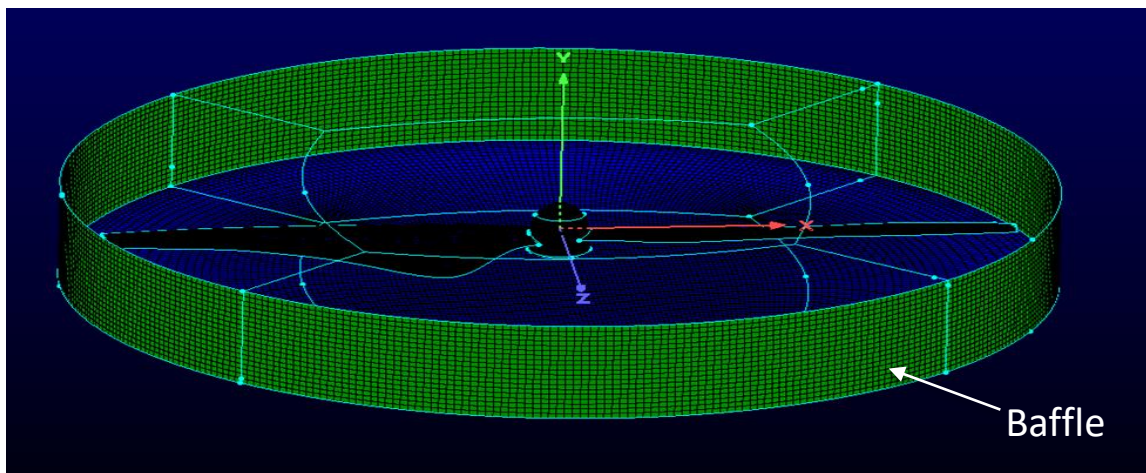


Figure 2.3 Location of the rotating zone and baffle

For the “coarse” unstructured grid, the near field is generated using tetrahedral cells. The cells are clustered and refined in the near field and around the blade and extends to approximately 2.5 radii away from the center of rotation. Cell sizes are resolved with ~ 30 points per wavelength which corresponds to a conservative acoustic wave cut-off frequency of ~ 1.5 kHz. The entire domain spans 30 radii from the center of rotation. An

advantage of utilizing an unstructured grid cell structure is the ability to easily employ aggressive grid stretching between refinement zones as seen in the figure below. This results in very good acoustic damping and avoids reflections at the boundaries. One of the goals of this thesis is to study the acoustics of the propeller. With this grid, larger data sampling and predictions are performed within a relatively shorter period, giving better results for acoustic spectral processing. A trade-off with this type of grid is post processing of flow fields are not ideal due to the quality of the grid resolution. The grid size is 7.9 million cells.

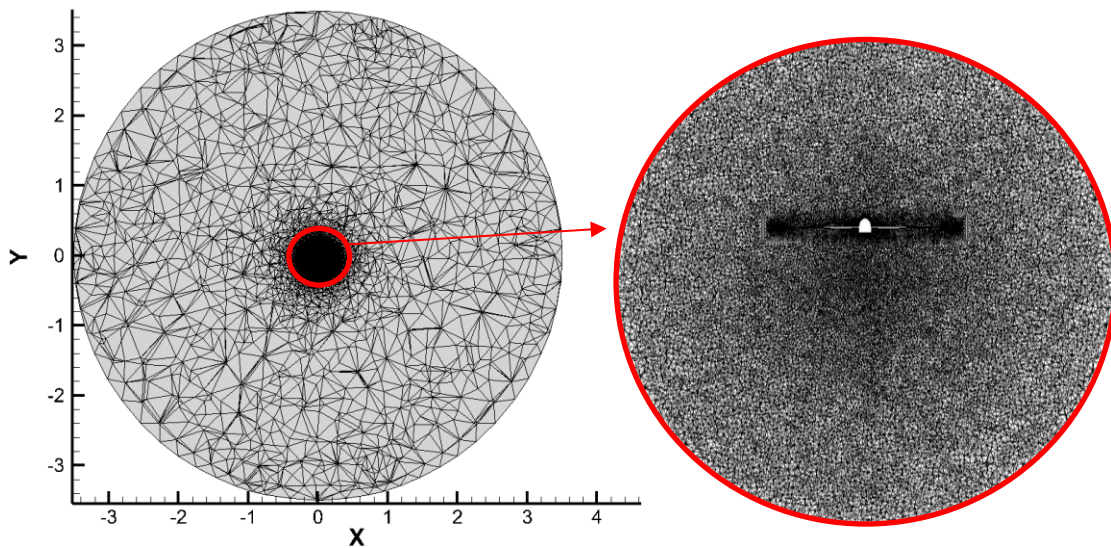


Figure 2.4 Vertical cross-section of Coarse grid showing grid refinement Regions and extreme coarsening.

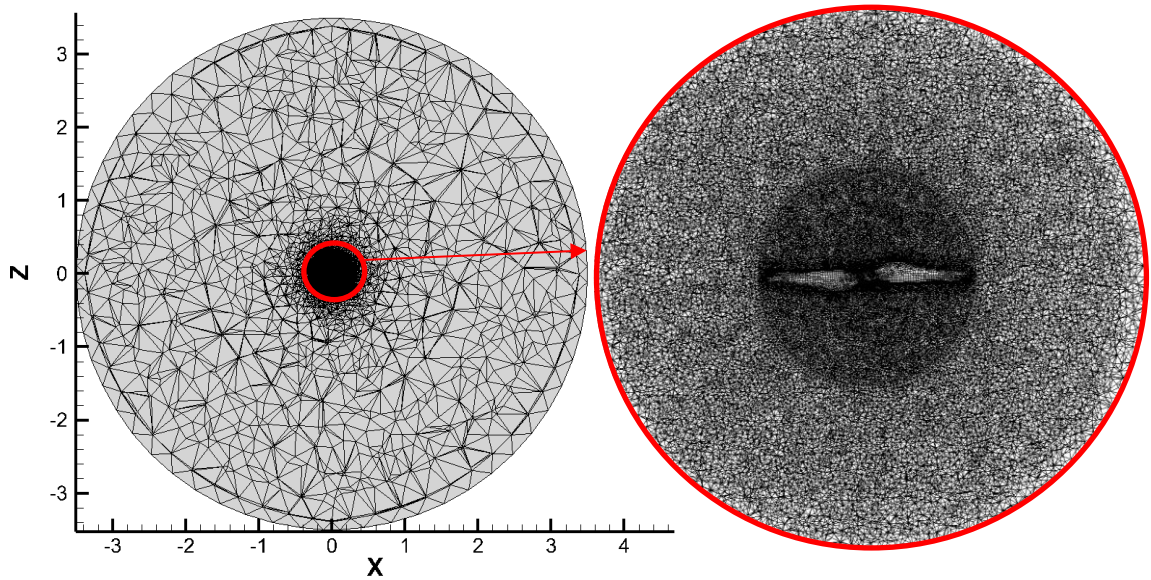


Figure 2.5 Horizontal cross-section of Coarse grid at $Y=0$ showing grid refinement Regions and extreme coarsening.

A finer grid is also generated with purely structured grid structure. In the near field, structured blocks with resolvable frequency of $\sim 2\text{kHz}$, corresponding to a uniform grid cell size of approximately 10% of the tip chord, which extends to 5.4 radii from the center of rotation. The grid is stretched to the outer walls of the domain. The entire domain spans 26 radii from the center. This is to ensure enough room for dissipation of disturbances. Again, these frequencies are computed with a minimum of 30 cells per wavelength to account for acoustic resolution. One challenge with grid generation for acoustics is reflection of waves. To avoid this, the domains are generated as spherical structures, which minimized the reflections from the computational domain boundaries. Good structured grid stretching is employed to better aid the dissipation of the wake roll-up. An approximate grid stretching ratio of 2:1 is utilized towards the outer boundaries. The total grid cell size is about 70 million cells.

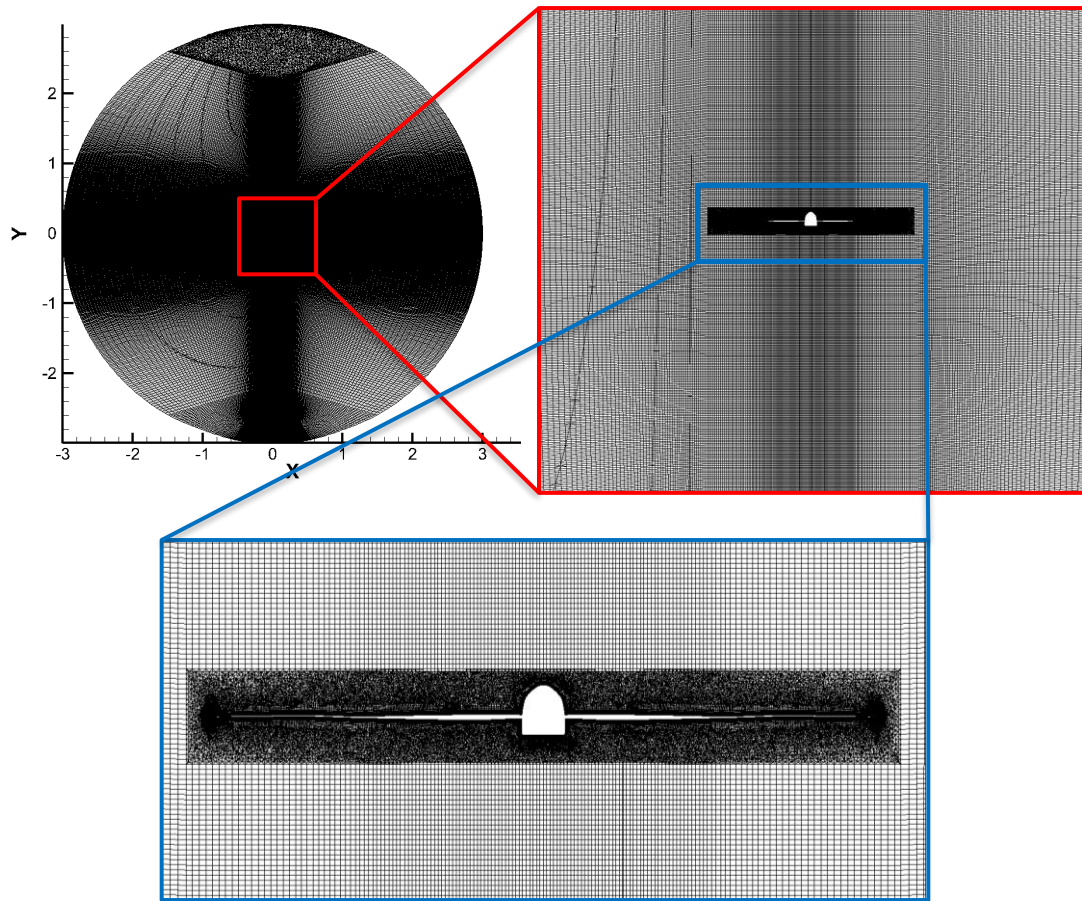


Figure 2.6 Vertical Cut through fine Grid structure

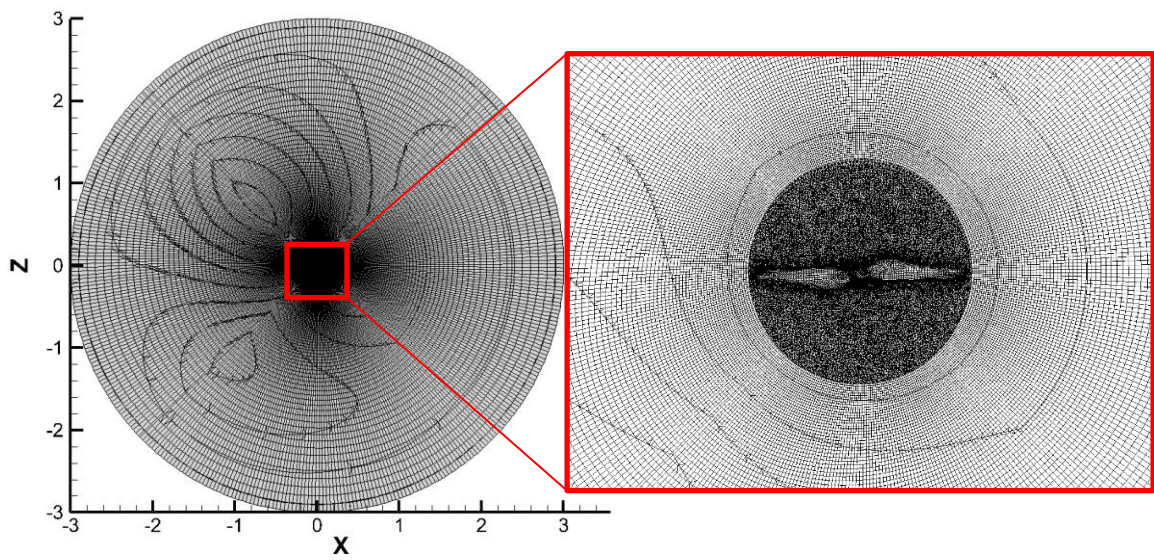


Figure 2.7 Horizontal Cut through fine Grid at $Y=0$

2.3.2. Solver

Open source finite volume code OpenFOAM v4.1 is employed in this thesis. Both unsteady RANS and Hybrid LES/URANS “DES” simulations are run. The pressure-based rhoPimpleFOAM, which is a transient compressible PIMPLE solver, is utilized to solve the Navier-Stokes and energy equations. This solver combines the PISO (Pressure Implicit with Splitting of Operator) and SIMPLE (Semi-Implicit Method for Pressure-linked Equations) algorithms. It uses the PISO-like time marching scheme with the pressure-velocity coupling performed in the SIMPLE algorithm. Utilizing this solver allows for flexibility in solution convergence by specifying several pressure correctors per loop. This allows for larger time stepping without divergence. The PIMPLE solver is utilized with 5 corrector loops per time step and 3 pressure corrections per loop, allowing for convergence of flow parameters per time step. An advantage of the pressure-based solver over the density-based solver is its utilization of inner-loop pressure correction steps, which increases solver stability.

The energy equation is computed using a calorically perfect, sensible enthalpy, and the ideal gas law is used for the equation of state. The laminar viscosity is calculated using the Sutherland’s law.

Mesh rotation is achieved using an arbitrary mesh interface (AMI). This is done by specifying a surface or interface between two regions. This is defined as a baffle. The region inside the baffle is set as a cell zone, which is passed to a dynamic library for mesh motion without deformation. The baffle or interface then acts as a medium for translation of the fluid variables from the rotating region to the stationary region.

OpenFOAM is a finite volume code, as such it integrates between the cell centers. It utilizes the Gauss theorem to convert the volume integrals into surface integrals.

Assuming ϕ represents the conservative form of all fluid flow equations, the general transport equation solved is:

$$\int_{V_p} \frac{\partial \rho \phi}{\partial t} dV + \int_{V_p} \nabla \cdot (\rho u \phi) dV - \int_{V_p} \nabla \cdot (\rho \Gamma_\phi \nabla \phi) dV = \int_{V_p} S_\phi(\phi) dV \quad (2.38)$$

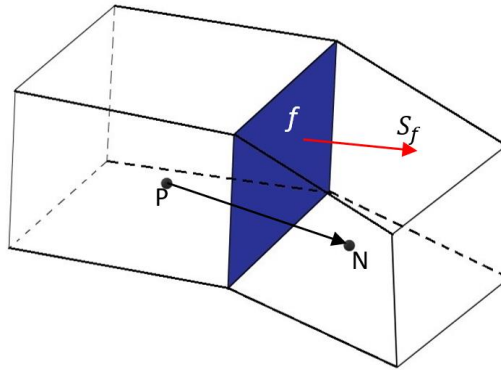


Figure 2.8 Control Volume showing neighboring cell centers and face area S_f

The first term in the equation above represents the temporal derivative, the second term represents the convective term, the third represents diffusion terms, and the last term on the right-hand side is the source terms.

Using the Gauss theorem, the volume integrals are converted to surface integrals, which can be represented as summations. The equations are then discretized spatially. OpenFOAM has an advantage of flexibility of the order of numerical discretization scheme to select. For this research, the second order accurate Gauss linear (central differencing) scheme is used for all the spatial derivatives.

To advance in time, the implicit, second-order accurate Crank-Nicolson method (Crank & Nicolson, 1947) is utilized. The Crank-Nicolson scheme includes an off-centering coefficient which can be modified to increase stability at the cost of the overall scheme order. For numerical stability, an off-centering coefficient of 0.4 is employed. A physical time step of 1.5×10^{-6} seconds, corresponding to about 0.05° rotor rotation at

6000 rpm is performed. This is used to initialize the URANS solution. After about 10 rotor rotations, the HLU turbulence model is switched on. The time step is reduced to 7.5×10^{-7} seconds. This is to ensure convergence of the LES section of the computation, as well as accurate resolution of the fluctuations in the flow and the acoustic field.

2.3.3. Boundary Treatment

Two major regions or surfaces are needed to be conditioned – the solid walls of the propeller blade, and the outer boundaries. The propeller surface is treated as a wall with zero gradient pressure and temperature boundary conditions. A No-Slip boundary condition with a moving wall velocity type is specified for the velocity.

The turbulent eddy viscosity is defined with a “nutUSpalding” wall function on the blade surface. This wall function is based on the special relation between y^+ and u^+ . The relation is given as follows:

$$y^+ = u^+ + \frac{1}{E} \left[e^{\kappa u^+} - 1 - \kappa u^+ - \frac{1}{2}(\kappa u^+)^2 - \frac{1}{6}(\kappa u^+)^3 \right] \quad (2.39)$$

Where κ is the von Karman constant, and E is 9.8 for smooth walls.

Since we need to directly capture the acoustics in the computational domain, attention is given to the selection of proper boundary treatment to avoid reflection from the computational boundaries or the introduction of spurious modes. The outer boundaries are modeled as outlets with the Navier-Stokes Characteristic Boundary Condition (NSCBC) implemented in OpenFOAM as a "Wave Transmissive" boundary condition (Poinsot & Lele, 1992). This is implemented by solving the mass flux (φ) equation at the boundaries:

$$\frac{D\varphi}{Dt} \approx \frac{\partial \varphi}{\partial t} + w_p \frac{\partial \varphi}{\partial n} = 0, \quad (2.40)$$

$$w_p = \frac{\varphi_p}{|S_f|} + \sqrt{\frac{\gamma}{\psi_p}} \quad (2.41)$$

where φ_p is the patch face flux, S_f is the patch face area vector, γ is the ratio of specific heats, and ψ_p is the compressibility which is the ratio of pressure and density. The wave speed w_p is computed as the sum of the velocity normal to the boundary and the speed of sound which is calculated from the second term on the right-hand side of equation 2.41. This is seen to work well with simple geometries, but some reflections are seen in when grid stretching is not properly done. Extra boundary conditioning is implemented using a source called “acousticDamping” (Wagner, Huttli, & Sagaut, 2007; Israeli & Orszag, 1981). This introduces an artificial damping source term in the governing equations, and it diminishes the strength of the waves within an absorbing region before they reach the boundaries to eliminate any residual reflections.

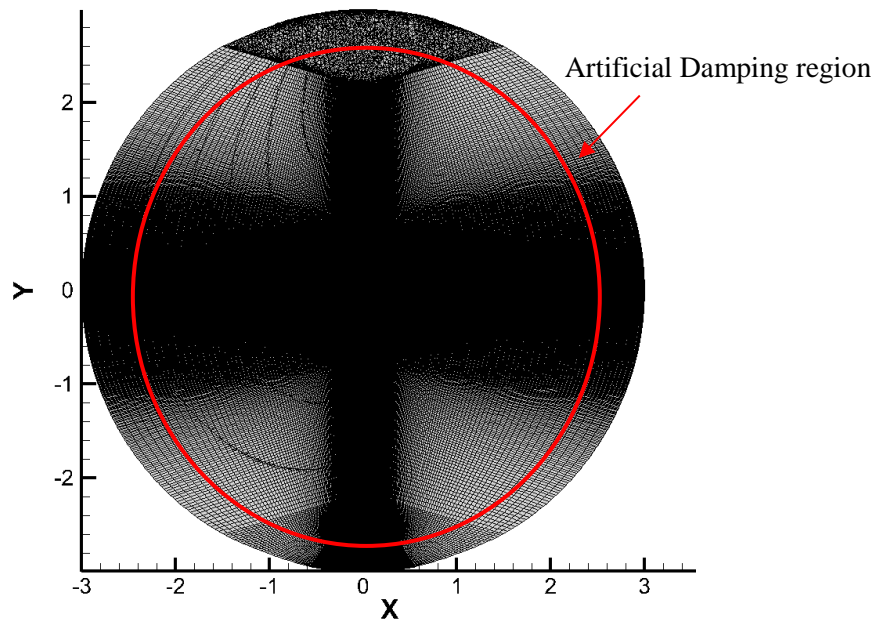


Figure 2.9 Computational domain showing acoustic damping region.

2.3.4. Flow Initialization

The flow is initialized with unsteady RANS with the computational domain initialized with quiescent free-stream conditions of pressure, velocity, and temperature of 101325 Pa, 0 m/s, and 298.15 K respectively. The turbulence parameters (ν_t & $\tilde{\nu}$) are initialized in the entire domain using the following equations:

$$\tilde{\nu} = 3\nu = 3 (1.562 \times 10^{-5}) = \frac{4.686 \times 10^{-5} m^2}{s} \quad (2.42)$$

$$\nu_t = \tilde{\nu} f_{\nu 1} = \tilde{\nu} \frac{\chi^3}{\chi^3 + c_{\nu 1}^3} = \frac{3.287 \times 10^{-6} m^2}{s} \quad (2.43)$$

Table 2.1

Boundary condition settings utilized in OpenFOAM

Boundary	nut (ν_t) $m^2 s^{-1}$	nuTilda ($\tilde{\nu}$) $m^2 s^{-1}$	P ($kg m^{-1} s^{-2}$)
Propeller	nutUSpaldingWallFunction 0	fixedValue 0	zeroGradient -
outlets	Calculated 3.287×10^{-6}	fixedValue 4.686×10^{-5}	waveTransmissive 101325
Internal Field	uniform 3.287×10^{-6}	uniform 4.686×10^{-5}	uniform 101325

Boundary	U (ms^{-1})	T (K)
Propeller	movingWallVelocity (0 0 0)	zeroGradient -
outlets	waveTransmissive (0 0 0)	waveTransmissive 298.5
Internal Field	uniform (0 0 0)	uniform 298.5

3. Far-Field Noise Prediction

In the current study, there is the need to investigate the noise signature of the propellers, and subsequently explore some noise reduction techniques. Due to the high costs involved with the experimental studies such as wind tunnel costs, as well as availability of anechoic chambers and acoustic equipment, more preference has been sought in implementation of numerical simulations as a cheaper alternative. The numerical simulations have become more favorable due to the evolution of computational power over the years. Accurate results in computational aero-acoustics (CAA) depend massively on the computation of the full, transient Navier-Stokes equations.

Aerodynamically generated sound is governed by nonlinear processes that are categorized into two: turbulence generated noise, and impulsive noise due to moving surfaces such as rotor noise (Lyrantzis, 2003). There are two major ways of predicting the noise. The most direct method is to extend the nonlinear computational domain to the far-field location. This requires that the flow field be accurately resolved up to the location of interest. Direct methods such as direct numerical simulations (DNS) or large-eddy simulations (LES) are often utilized. This results in a very large computational domain and requires an exorbitant amount of computational power. A simpler set of equations such as a Euler/Navier-Stokes model, or a full potential model can be used.

The Linearized Euler Equations (LEE) have been used to extend the CFD solutions to the far-field (Lim et al., 1993; Viswanathan & Sankar, 1995; Mankbadi et al, 1998). A hybrid zonal approach is utilized, where there is a near-field computation using accurate CFD code, and an extension of the solution to the mid-field using LEE. This method is more computationally effective since the LEE computations are less cumbersome. This

method is favorable for studying acoustic sources that are not extremely far away from the source.

3.1. Integral Methods

As with most realistic problems, the goal is to be able to predict the sound propagation in the far-field. Another approach to achieve this is the utilization of acoustic analogies. The Navier-stokes equations are reduced to a simplified wave equation with a source term. The far-field pressure is defined in terms of integrals over the volume of a surface encasing the sources.

3.1.1. Lighthill's Analogy

Lighthill (Lighthill, 1952) formulated an approach to describe the sound generated in an isolated turbulent region. By combining the mass and momentum conservation equations, the wave equation (3.1) is obtained:

$$\frac{\partial^2 \rho'}{\partial t^2} - c_0^2 \nabla^2 \rho' = \frac{\partial^2}{\partial x_i \partial x_j} T_{ij} \quad (3.1)$$

$$T_{ij} = \rho' u_i u_j + (p' - c_0^2 \rho') \delta_{ij} - \sigma_{ij} \quad (3.2)$$

T_{ij} is known as Lighthill's stress tensor. This is an exact, inhomogeneous wave equation where the source term T_{ij} is only important in the turbulent region. The fluid is at rest at infinity. Lighthill's analogy separated the analysis of aerodynamic noise into two parts – the sound generation due to the fluctuating sources T_{ij} , and the propagation of sound in the surrounding quiescent medium due to the fluctuating sources. The analogy specifically addresses the problem of sound generation by a region of turbulent flow into a quiescent medium. A solution to the Lighthill equation is given as:

$$p_s = \frac{1}{4\pi R_{ob} c_0^2} \iiint \left[\frac{\partial^2}{\partial x_i \partial x_j} \{ \rho u_i u_j \} + \frac{1}{c_0^2} \frac{\partial^2}{\partial t^2} \{ p - c_0^2 \rho \} \right] dV \quad (3.3)$$

The curly brackets $\{ \}$ indicate that the source term is calculated at the retarded time

$$t_r = t - \frac{|\mathbf{x} - \mathbf{y}|}{c_0} \quad (3.4)$$

Where p_s is the far-field pressure, R_{ob} is the distance between observer point and source, t is the current observation time measured at observation point. \mathbf{x} and \mathbf{y} are the locations of the observer and source respectively. The second source term is usually neglected.

Assuming that the sound source is compact, the equation can be approximated as:

$$p_s(\mathbf{x}, t) = \frac{1}{4\pi R_{ob} c_0^2} \iiint \left[\frac{\partial^2}{\partial t^2} \{ \rho u_i u_j \} \right] dV \quad (3.5)$$

3.1.2. Kirchhoff's Method

Another alternative method of prediction is the Kirchhoff method, which assumes the sound transmission is governed by the wave equation. It is based on the computation of the nonlinear near and mid-field numerically, and the evaluation of the linear far-field region using the linear Kirchhoff formulation. The control surface is assumed to enclose all the nonlinear sources.

$$\nabla^2 p - \frac{1}{c_0^2} \left(\frac{\partial}{\partial t} + U_\infty \frac{\partial}{\partial x} \right)^2 p = 0 \quad (3.6)$$

The sound pressure is then calculated using surface integrals with its normal and time derivatives.

$$p(x, y, z, t) = -\frac{1}{4\pi} \int_{S_o} \left[\frac{p}{r_o^2} \frac{\partial r_o}{\partial n_o} + \frac{1}{r_o} \frac{\partial p}{\partial n_o} + \frac{1}{c_\infty} \frac{\partial p}{\partial t} \frac{\partial r_o}{\partial n_o} \right]_\tau dS_o \quad (3.7)$$

Where $r_o = \sqrt{(x - x_0)^2 + (y - y_0)^2 + (z - z_0)^2}$ and $\tau = t - \frac{r_o}{c_\infty}$. The n denotes the

outward normal vector to the Kirchhoff surface. An alternative to equation 3.7 in

frequency domain was presented by Lyrintzis and Mankbadi (Lyrintzis and Mankbadi,

1996).

There have been many applications of the Kirchhoff method in various aero-acoustic studies. The classical Kirchhoff formulation is limited to a stationary surface. Farassat and Myers (1988) derived a Kirchhoff formulation for moving surfaces. Mankbadi and Lyrintzis (1996) performed far-field predictions of the jet noise using the Kirchhoff formulation.

3.1.3. Surface Integral Formulations

Mankbadi et al. (1998) developed a modified Green's function for a cylindrical control surface for far-field noise prediction using Surface Integral Formulation (SIF) which eliminates the need for normal derivatives. The Kirchhoff solution is given as:

$$p(\omega) = - \int \left[G \frac{\partial p}{\partial n} - p \frac{\partial G}{\partial n} \right] dS \quad (3.8)$$

Where G is the Green function. For an infinite cylinder of radius r_k , a Green function for an observer at r_o is given as:

$$G = \Xi \int_{-\infty}^{\infty} e^{ik_x(x_o-x)} \frac{H_m(qr_o)}{H_m(qr_k)} \cdot [J_m(qr)H_m(qr_k) - J_m(qr_k)H_m(qr)] dk_x \quad (3.9)$$

With

$$\Xi = \frac{1}{8\pi} \sum_{m=0}^{\infty} \varepsilon_m \cos[m(\phi - \phi_o)] \quad (3.10)$$

J_m is the m^{th} -order Bessel function and H_m is the m^{th} -order Hankel function of the first kind. k_x is the wavenumber of the acoustic disturbance on the surface in axial direction. The derivative of the Green's function is given by:

$$\frac{\partial G}{\partial r} = - \frac{1}{4\pi^2 r_k} \sum_{m=0}^{\infty} \varepsilon_m \cos[m(\phi - \phi_o)] \int_{-\infty}^{\infty} e^{ik_x(x_o-x)} \frac{H_m(qr_o)}{H_m(qr_k)} dk_x \quad (3.11)$$

The solution at the far field then becomes:

$$p(x_o, \phi_o, \omega) = \frac{1}{2\pi} \sum_{m=0}^{\infty} \cos m \phi_o \int p_m(x, \omega) \int_{\infty}^{\infty} e^{ik_x(x_o-x)} \frac{H_m(qr_o)}{H_m(qr_k)} dk_x \quad (3.12)$$

The above formulation describes the relation between the acoustic far field and the pressure distribution on a cylindrical surface surrounding the noise sources. The need for pressure normal derivatives is eliminated.

3.1.4. Ffowcs Williams-Hawkings Method

Another alternative method for acoustic predictions using the surface integral methods is the Ffowcs Williams-Hawkings (FWH) method. The original formulation assumed an integration over an impenetrable surface. A more general equation was derived for a porous surface as mentioned by Ffowcs Williams and Hawking (1969). Similarly, the quantities on the porous control surface is used to predict the far-field pressure. A complete comparison of the different formulations is discussed by Lyrintzis (2003).

In the FWH method, the time histories of all the flow variables are needed for the prediction. The far-field solution is computed using a surface and volume integral which stems from the quadrupole source terms. In many applications of this formulations, especially in the area of rotorcraft aeroacoustics (Strawn and Biswas, 1995; Di Francescantônio, 1997), the method has been implemented by taking the control surface to coincide with the surface of the blade or the solid body. However, the method is still valid when the control surface is off the body and is permeable (Pilon & Lyrintzis, 1997).

The FWH equations is derived by manipulating the governing equations to obtain the “generalized equations”. They are then combined to obtain a modified wave equation that leads to the integral expression of the porous FWH formulation.

Continuity Eq (Williams & Hawkings, 1969):

Set

$$\frac{\partial}{\partial t} \int_V \bar{\rho} dV = \frac{\partial}{\partial t} \int_{V(1)} \rho^{(1)} dV + \frac{\partial}{\partial t} \int_{V(2)} \rho^{(2)} dV \quad (3.13)$$

From the continuity Eq. for a moving surface:

$$\frac{\partial}{\partial t} \int_{V(1)} \rho^{(1)} dV = - \int_{\Sigma(1)} (\rho u_i)^{(1)} l_i d\Sigma - \int_S [\rho(u_i - v_i)]^{(1)} n_i dS \quad (3.14)$$

$$\frac{\partial}{\partial t} \int_{V(2)} \rho^{(2)} dV = - \int_{\Sigma(2)} (\rho u_i)^{(2)} l_i d\Sigma + \int_S [\rho(u_i - v_i)]^{(2)} n_i dS \quad (3.15)$$

Hence

$$\frac{\partial}{\partial t} \int_V \bar{\rho} dV = - \int_{\Sigma} (\overline{\rho u_i}) l_i d\Sigma + \int_S [\rho(u_i - v_i)]^{(2)}_{(1)} n_i dS \quad (3.16)$$

Using the divergence theorem,

$$\int_V \left[\frac{\partial \bar{\rho}}{\partial t} + \frac{\partial}{\partial x_i} (\overline{\rho u_i}) \right] dV = \int_S [\rho(u_i - v_i)]^{(2)}_{(1)} n_i dS \quad (3.17)$$

If an equation of the form $f=0$ defines the surface S such that $f<0$ in (1) and $f>0$ in (2), the integral over S can be rewritten as

$$\int_S [\rho(u_i - v_i)]^{(2)}_{(1)} n_i dS = \int_V [\rho(u_i - v_i)]^{(2)}_{(1)} n_i \delta(f) \left\{ \left(\frac{\partial f}{\partial x_i} \right)^2 \right\}^{0.5} dV \quad (3.18)$$

Hence, we obtain the generalized continuity eq.

$$\frac{\partial \bar{\rho}}{\partial t} + \frac{\partial}{\partial x_i} (\overline{\rho u_i}) = [\rho(u_i - v_i)]^{(2)}_{(1)} \delta(f) \frac{\partial f}{\partial x_i} \quad (3.19)$$

Similarly, the momentum equation gives the following generalized form

$$\frac{\partial}{\partial t} (\overline{\rho u_i}) + \frac{\partial}{\partial x_j} (\overline{\rho u_i u_j} + \bar{P}_{ij}) = [P_{ij} + \rho u_i (u_j - v_j)]^{(2)}_{(1)} \delta(f) \frac{\partial f}{\partial x_j} \quad (3.20)$$

Where

$$P_{ij} = (p - p_o) \delta_{ij} \quad (3.21)$$

Taking the temporal derivative of the generalized continuity equation, subtracting the divergence of the generalized momentum equation and using minor manipulations, it is possible to derive a generalized inhomogeneous wave equation (assuming that $|\nabla f|=1$ for

$f=0$):

$$\left(\frac{\partial^2}{\partial t^2} - c^2 \frac{\partial^2}{\partial x_j^2}\right) (\overline{\rho - \rho_0}) = \frac{\partial}{\partial t} [\rho_0 u_n \delta(f)] - \frac{\partial}{\partial x_i} [P_{ij} n_j \delta(f)] + \frac{\partial^2 T_{ij}}{\partial x_i \partial x_j} + \frac{\partial}{\partial t} [(\overline{\rho - \rho_0})(u_n - v_n) \delta(f)] - \frac{\partial}{\partial x_j} [\rho u_i (u_n - v_n) \delta(f)] \quad (3.22)$$

Setting some new variables as defined by Di Frances Antonio (1997):

$$U_n = (1 - \rho/\rho_0)v_n + (\rho/\rho_0) u_n \quad (3.23)$$

$$L_{ij} = P_{ij} + \rho u_i (u_n - v_n) \quad (3.24)$$

The equation can be rewritten as

$$\left(\frac{\partial^2}{\partial t^2} - c^2 \frac{\partial^2}{\partial x_j^2}\right) (\overline{\rho - \rho_0}) = \frac{\partial}{\partial t} [\rho_0 U_n \delta(f)] - \frac{\partial}{\partial x_i} [L_{ij} n_j \delta(f)] + \frac{\partial^2 T_{ij}}{\partial x_i \partial x_j} \quad (3.25)$$

The Green function \mathbf{G} of the unbounded three-dimensional space is defined as

$\mathbf{G} = \delta(\mathbf{g})/r$, where $\mathbf{r} = |\mathbf{x} - \mathbf{y}|$, $\mathbf{g} = t - \tau - r/c$, and where x and y are the observer and source position respectively, and t and τ are observer and source times. Executing a convolution of the equation with the Green function we obtain:

$$\begin{aligned} 4\pi c^2 (\rho - \rho_0) &= \frac{\partial}{\partial t} \int_S \left[\frac{\rho_0 u_n + (\rho - \rho_0)(u_n - v_n)}{r|1 - M_r|} \right]_{ret} dS + \\ &\frac{1}{c} \frac{\partial}{\partial t} \int_S \left[\frac{P'_{ij} + \rho u_r (u_n - v_n)}{r|1 - M_r|} \right]_{ret} dS + \int_S \left[\frac{P'_{ij} + \rho u_r (u_n - v_n)}{r^2|1 - M_r|} \right]_{ret} dS + \\ &\frac{1}{c^2} \frac{\partial^2}{\partial t^2} \int_V \left[\frac{T_{rr}}{r|1 - M_r|} \right]_{ret} dV + \frac{1}{c} \frac{\partial}{\partial t} \int_V \left[\frac{3T_{rr} - T_{ii}}{r^2|1 - M_r|} \right]_{ret} dV + \int_V \left[\frac{3T_{rr} - T_{ii}}{r^3|1 - M_r|} \right]_{ret} dV \end{aligned} \quad (3.26)$$

Where $\mathbf{M}_r = v_i r_i / c$ is the Mach number in the observer direction, $T_{rr} = T_{ij} r_i r_j$, and $T_{ii} = T_{11} + T_{22} + T_{33}$. Also, V is the volume external to the surface S ($f > 0$), and the subscript ref means the evaluation at the retarded time $\tau^* = t - r/c$. Note that the spatial derivative is being transformed in a time derivative using:

$$\frac{\partial}{\partial x_i} \left[\frac{\delta(g)}{r} \right]_{ret} = -\frac{1}{c} \frac{\partial}{\partial t} \left[\frac{r_i \delta(g)}{r^2} \right]_{ret} - \frac{r_i \delta(g)}{r^3} \quad (3.27)$$

In equation 3.26, each line of the right-hand term represents a different pressure term. The first line term is called thickness noise. The terms on the second line show the loading noise. The terms on the third line represent the quadrupole noise. In the porous formulation the terms lose their physical meaning, except the quadrupole noise. If the surface is rigid the thickness noise would be the sound generated by a distribution of monopoles, which is connected to a variation of mass. The loading noise would be the sound generated by a distribution of dipoles, which is connected to a variation of momentum.

The above formulation involves a volume integral, which is generally computationally expensive. As such Farassat's Formulation 1A developed by Brentner and Farassat (Brentner & Farassat, 1998) simplifies the equations by neglecting the quadrupole term.

Using $c^2(\rho - \rho_0) = p'$, the equation can be rewritten as

$$4\pi p'(x, t) = \frac{\partial}{\partial t} \int_S \left[\frac{\rho_0 u_n + (\rho - \rho_0)(u_n - v_n)}{r|1-M_r|} \right]_{ret} dS + \frac{1}{c} \frac{\partial}{\partial t} \int_S \left[\frac{P'_{ij} + \rho u_r (u_n - v_n)}{r|1-M_r|} \right]_{ret} dS + \int_S \left[\frac{P'_{ij} + \rho u_r (u_n - v_n)}{r^2|1-M_r|} \right]_{ret} dS \quad (3.28)$$

Moving the time derivatives inside the integrand, we obtain:

$$4\pi p'(x, t) = \int_{f=0} \left[\frac{\rho_0 (\dot{U}_n + U_{\dot{n}})}{r(1-M_r)^2} \right]_{ret} dS + \int_{f=0} \left[\frac{\rho_0 U_n ((r\dot{M}_r + c(M_r - M^2)))}{r^2(1-M_r)^3} \right]_{ret} dS + \frac{1}{c} \int_{f=0} \left[\frac{L_r}{r(1-M_r)^2} \right]_{ret} dS + \int_{f=0} \left[\frac{L_r - L_M}{r^2(1-M_r)^2} \right]_{ret} dS + \frac{1}{c} \int_{f=0} \left[\frac{L_r ((r\dot{M}_r + c(M_r - M^2)))}{r^2(1-M_r)^3} \right]_{ret} dS \quad (3.29)$$

Where U and M are the surface motion velocity and Mach number, r is the distance

between source and observer, L is given by equation 3.24. \dot{L}_r , \dot{U}_n , and \dot{M}_r represent the source time derivatives. The subscripts r or n denote a dot product of the vector with the unit vector in the radiation direction \hat{r} , or the unit vector in the surface normal direction \hat{n} respectively. The term $L_M = L_i M_i$. The term “*ret*” refers to retarded time. The formulation above assumes a control surface where $f=0$.

We note that the turbulence sources, as given by Lighthill’s stress tensor, do not appear explicitly in the solution. However, their influence the solution as they contribute to the pressure data on the FWH used in computing the far field. The above sets of equations are valid for moving control surfaces. For this study, the control surface utilized is stationary. As such the above solution reduces to:

$$4\pi p(x, t) = \int_{f=0} \left[\frac{\rho_0 \dot{U}_n}{r} \right]_{ret} dS + \frac{1}{c} \int_{f=0} \left[\frac{\dot{L}_r}{r} \right]_{ret} dS + \int_{f=0} \left[\frac{\dot{L}_r}{r^2} \right]_{ret} dS \quad (3.30)$$

with $\dot{U}_n = \dot{U} \hat{n}_i$, $\dot{L}_r = \dot{L} \hat{r}_i$, and $L_r = L_i \hat{r}_i$, and L_i defined in equation 3.24.

4. Results and Discussion

The simulations were performed on Embry-Riddle Aeronautical University's Vega cluster. The simulations were typically run on 360 CPU cores in parallel for the fine-grid case. A run time of approximately 650 CPU hours is typically needed. The results are sub-divided into four sections: the flow field, the near-field acoustics along with identification of the noise sources, the far-field noise predictions, and possible modifications for noise reduction.

4.1. Flow Field

We start first by comparing the thrust of our CAD-modeled propeller with corresponding data for the actual Type 9450 propeller of the commercial DJI Phantom II drone. The coefficient of thrust is calculated using:

$$C_T = \frac{T}{\rho_\infty \pi R^2 (\Omega R)^2} \quad (4.1)$$

Here, Ω is the rotor rotation rate, and R is the rotor tip radius. A range of rotor speeds were simulated, and the thrust coefficients were computed and compared in *Figure 4.2*. Close agreement can be seen between our results (HLU) with experimental and numerical results (OVERFLOW) performed by Zawodny et al (2016). Some slight deviations can be attributed to the fact that our geometry is slightly different than the tested one.

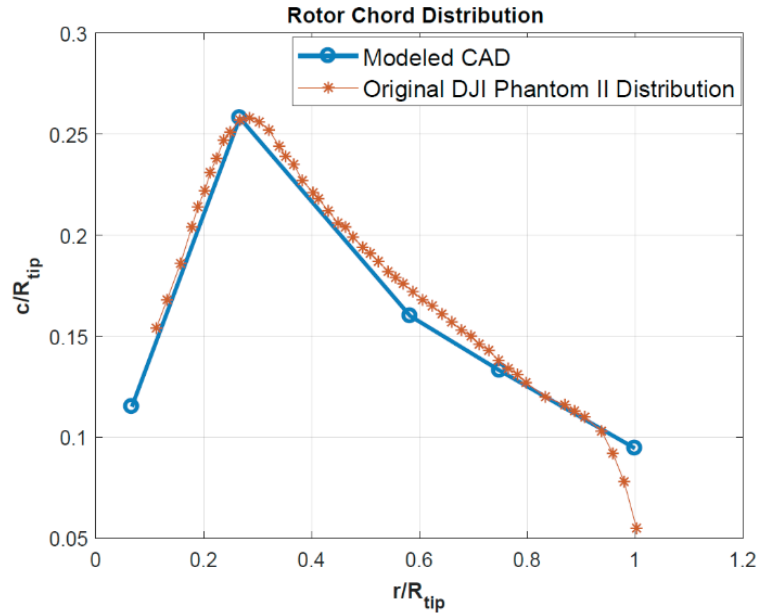


Figure 4.1 Rotor chord distribution comparison between CAD model and original used by Zawodny et al. (2016).

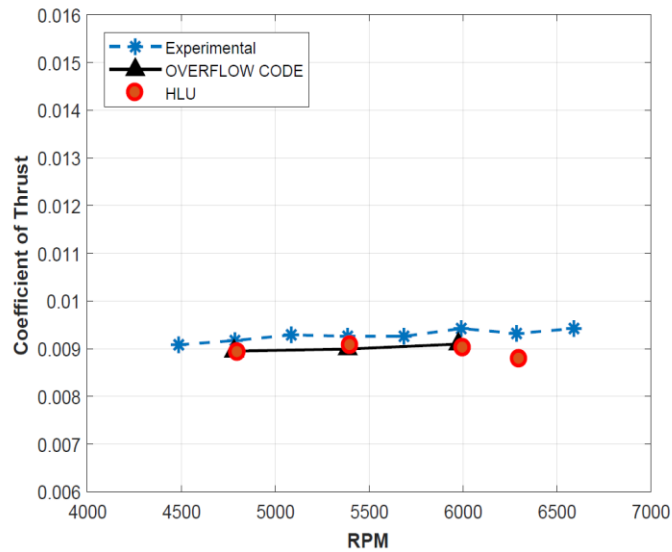


Figure 4.2 Propeller Thrust values compared with data from Zawodny et al. (2016).

For the subsequent results presented here, a rotor speed of 6000rpm is selected. A slice of the velocity and vorticity magnitude plots are shown in *Figure 4.3* and *Figure 4.4*. They are compared with numerical results performed by Yoon & Diaz (2018). We

can see general agreement in the flow structure. The numerical predictions of Yoon & Diaz show smaller eddies or roll-ups at the edges of the wake as compared to the current results. This is attributed to the difference in resolution. The results shown by Yoon & Diaz has a resolution of approximately 396 million grid points whereas the current result has 70 million grid points. However, as will be demonstrated later, the current resolution is adequate to resolve the acoustic field.

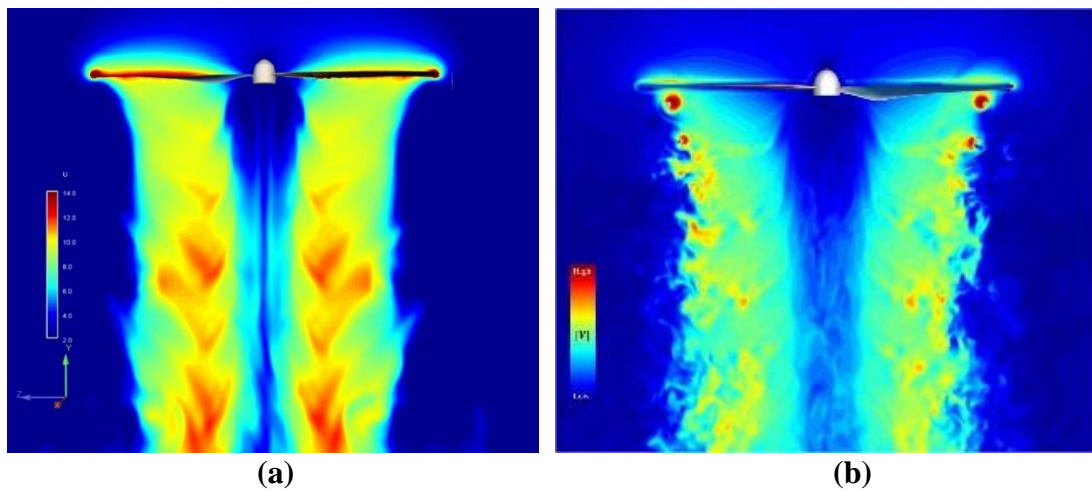


Figure 4.3 Results of velocity magnitude contours (a), compared with that of Yoon & Diaz (b).

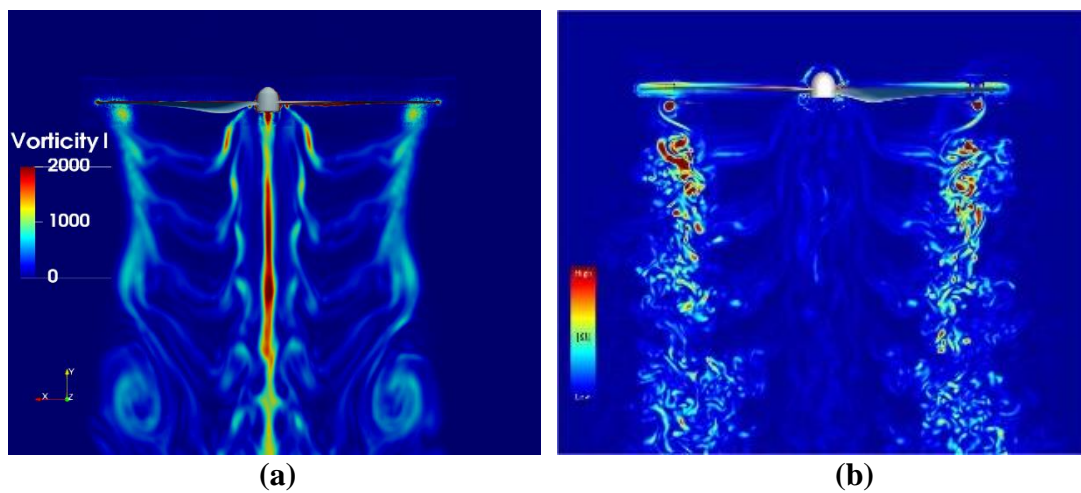


Figure 4.4 Results of vorticity magnitude contours (a), compared with that of Yoon & Diaz (b)

In *Figure 4.5* & *Figure 4.6*, we provide a full view of the flow field. *Figure 4.5* shows the magnitude of the velocity. The wake acts in some sense as a swirling jet that expands by entrainment, then ultimately decays downstream. *Figure 4.6* shows the vortical structure of the wake of the isolated propeller, along with Q-criterion iso-surfaces.

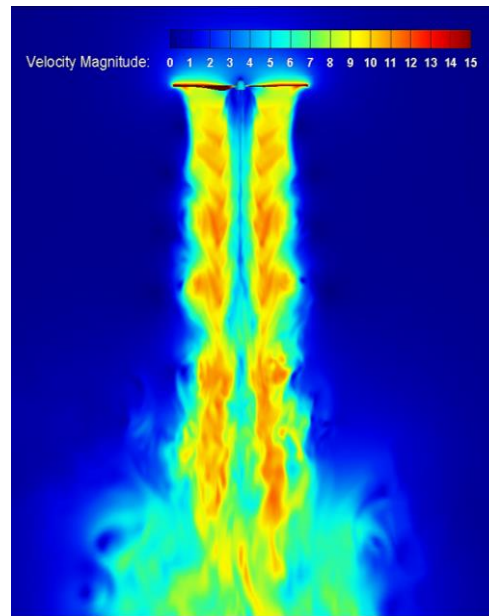


Figure 4.5 Snapshots of Velocity Magnitude contours

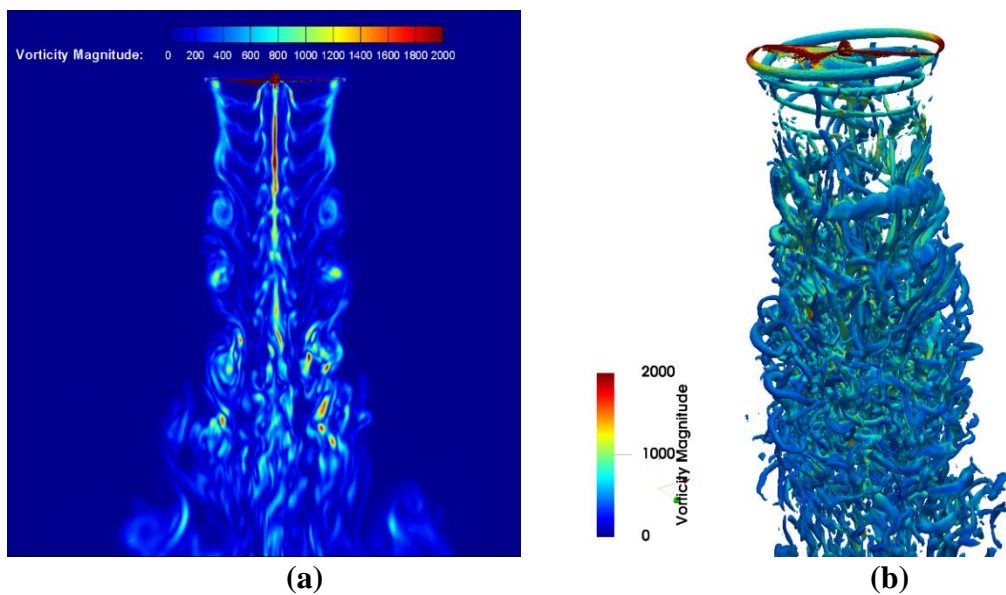


Figure 4.6 Snapshots of: (a) Vorticity Magnitude and (b) Q-criterion Iso-surfaces colored by vorticity.

4.2. Near Field Acoustics

Snapshots of the dilatation fields are shown in *Figure 4.7*. We note that by assuming 30 grid points are needed to capture the acoustic waves, our conservative estimates would be that we can accurately capture frequencies about to 1700 Hz. This covers the low-to-mid frequency range of the tonal and broadband noise content in the spectrum. Looking at the acoustic wave propagation, we can identify some characteristics of the generated noise. The strongest waves observed radiate from the propeller surface due to the loading noise component. The thickness noise (*Figure 4.10*) is radiated in the rotor plane but is overshadowed by the strength of the loading noise. Overlaid on the figure are the vorticity contours, which is dominant at wake region. The figure also shows that both the propeller and the wake flow are significant noise sources.

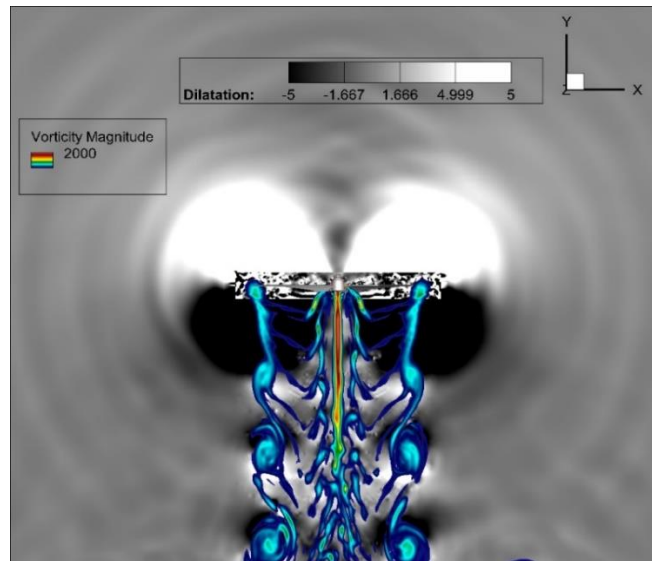


Figure 4.7 Vertical slice showing snapshot of Vorticity Magnitude overlaid on Dilatation Field

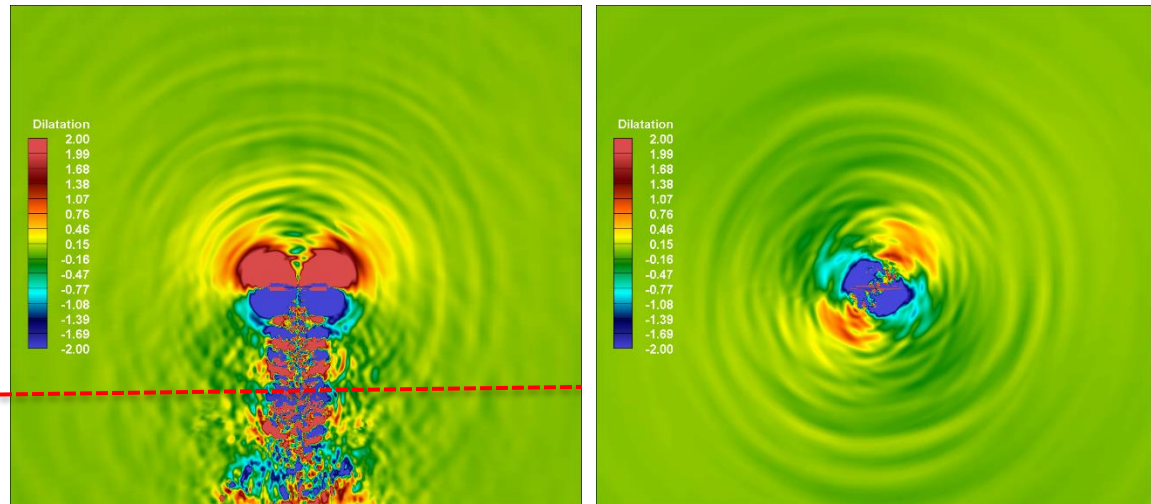


Figure 4.8 Vertical slice (left) and Horizontal slice (right) of the Dilatation Field

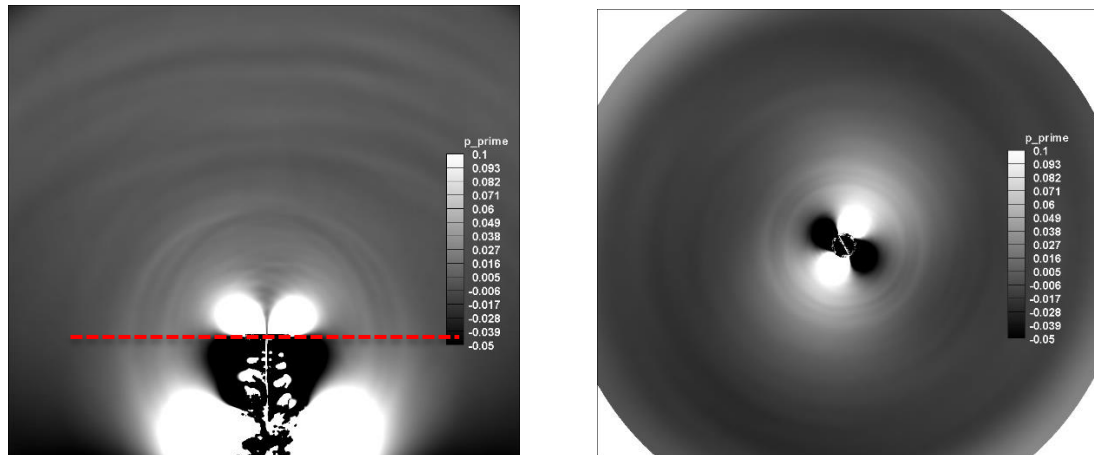


Figure 4.9 Snapshot of Acoustic Pressure vertical slice (left), and horizontal cut through rotor plane (right).

In the figure above, we attempt to plot out the contours of acoustic pressure. We zoom in to the region of the flow. On the vertical slice through the flow field (left image), we see that the magnitude of the acoustic waves is quite small. The bright spots seen in the images can be interpreted as the pressure fluctuations due to the convection of the flow field.

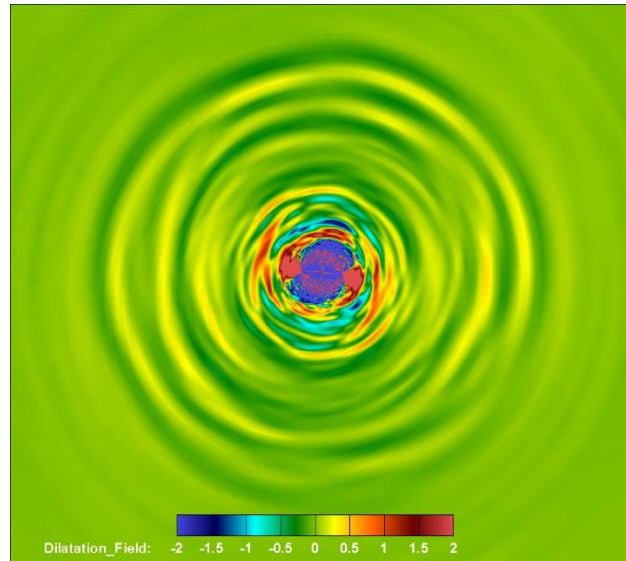


Figure 4.10 Horizontal snapshot of Dilatation field at in the rotor plane $Y=0$, showing monopole distribution.

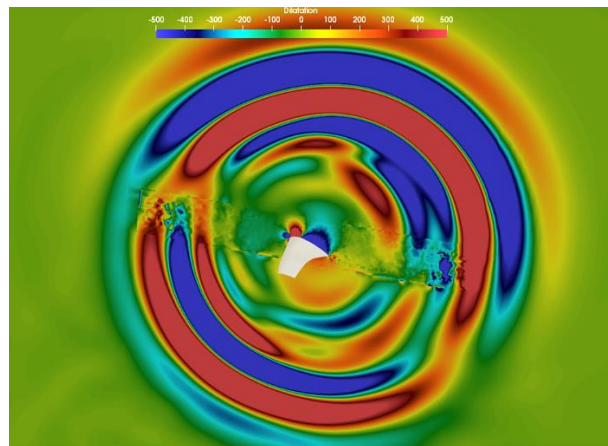


Figure 4.11 Snapshot of Dilatation fields sliced at $0.3R$

Further zooming in the vicinity of the blades and re-plotting the dilation with a different scale as shown in *Figure 4.11* and *Figure 4.12*, it shows that the blades acoustic field resembles that of a dipole radiating from the blade surface due to loading. This can clearly be seen in the *Figure 4.11* above. Banded contour plots are used to show the wavy nature of the monopoles above and below the rotor plane. This also explains the

black/white regions in *Figure 4.7*. Due to the rotating nature of the blades these pulsate.

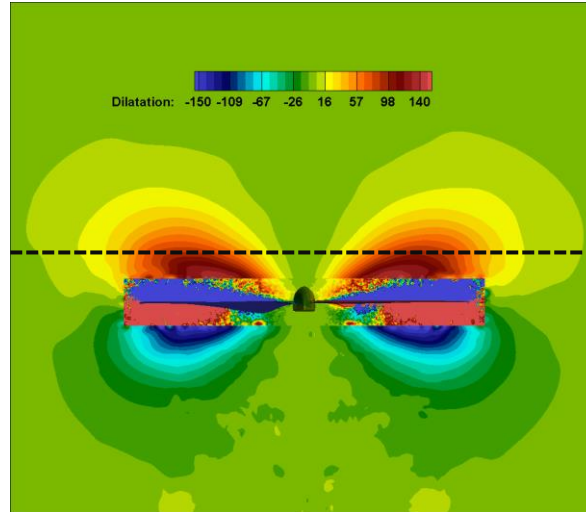


Figure 4.12 Zoomed in Banded contours of Dilatation fields with focus on dipoles

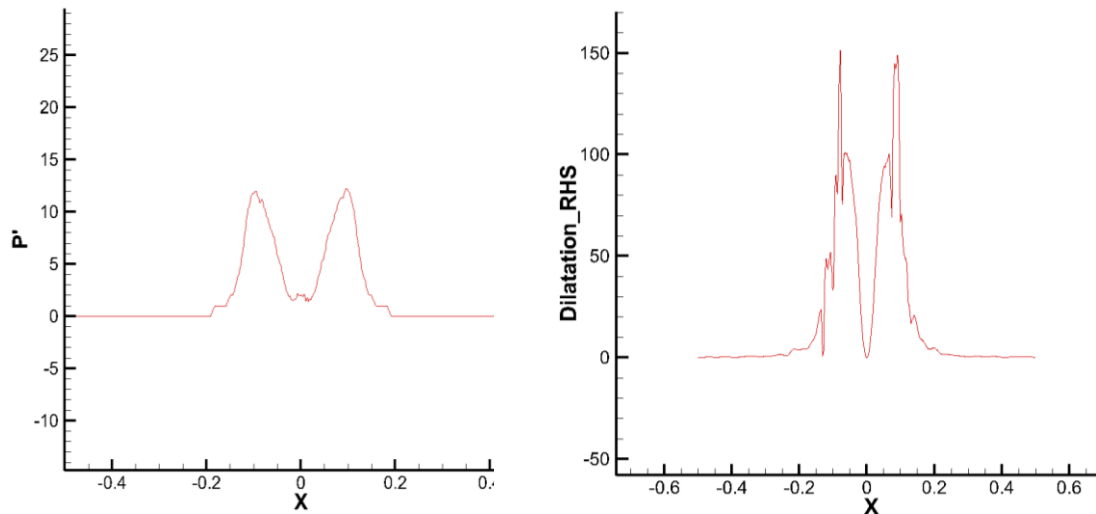


Figure 4.13 Linear plots of Dilatation and pressure fluctuation across the monopoles

The vortical structures are visualized using the Q-criterion Iso-contours. Their interaction with the acoustic fields is shown in *Figure 4.14*. A view of the swirl-like flow physics of the rotating propeller is also seen in the streamline plot shown in *Figure 4.15*.

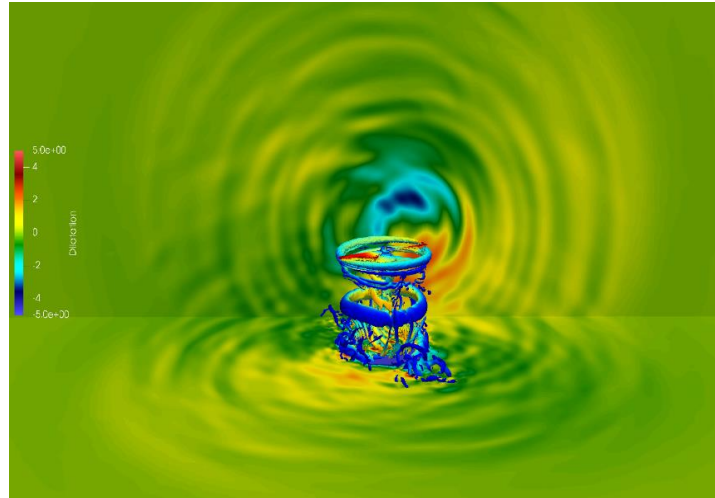


Figure 4.14 Dilatation Fields with Q-Criterion Iso-contours colored by velocity magnitude

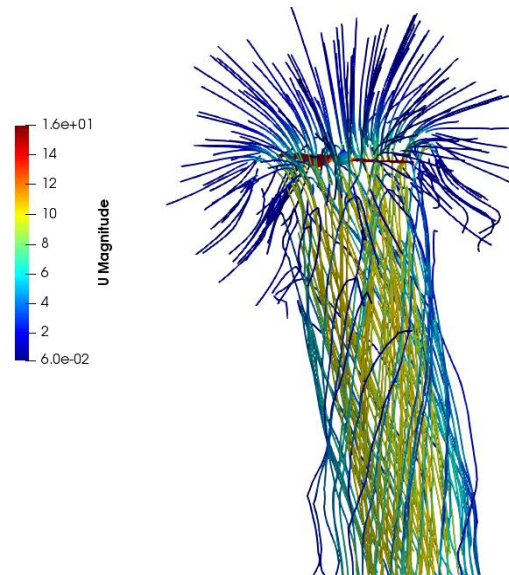


Figure 4.15 Streamlines colored by velocity magnitude

4.3. Noise Sources

In order to further characterize the noise mechanisms of isolated propellers, we attempt to identify the noise sources. The discussion is split into the blade noise sources and the wake noise sources.

4.3.1. Blade Noise Sources

The blade sources can be split into two parts. The steady components of the pressure, which produces the tones, and the fluctuating components, which may manifest as broadband in nature. Investigating the noise sources on the blade surface, we can see in *Figure 4.16* the mean pressure changes between the top and bottom surface, which produces the tonal noise. *Figure 4.17* shows that the most significant pressure fluctuations are observed on the bottom surface of the blade where the aerodynamic loading is the highest. Thus, the loading appears to be the dominant sound source. Some significant pressure fluctuations are seen near the leading edge, as well as at the blade tip. The difference in the static pressure is maximum near the tips.

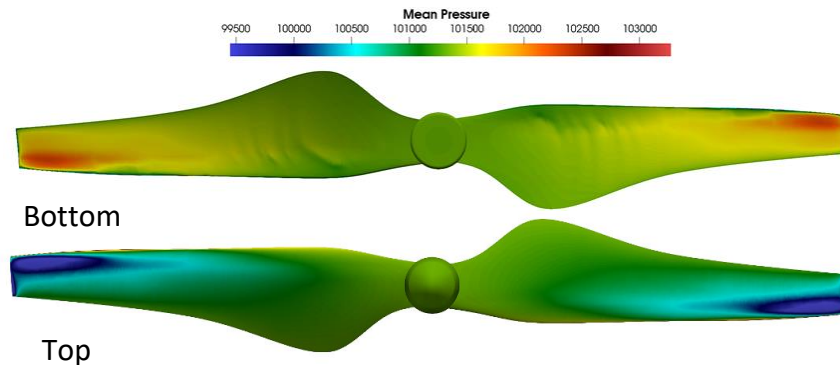


Figure 4.16 Mean Pressure Distribution on blade surface

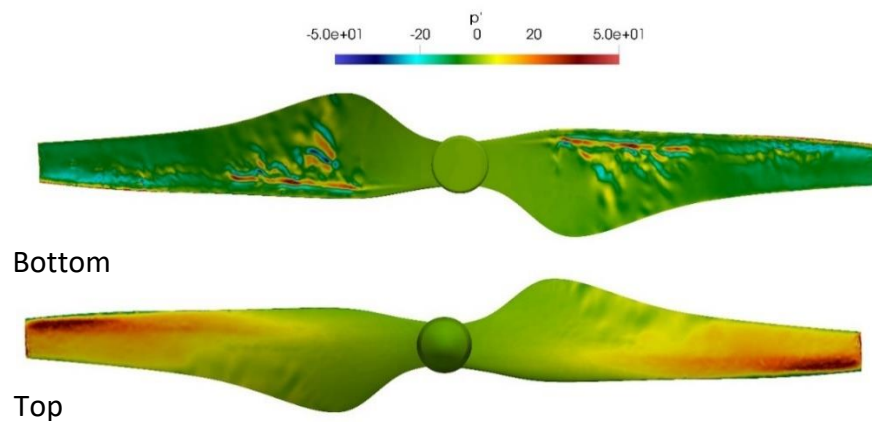


Figure 4.17 Pressure fluctuations (p') distribution on blade surface

4.3.2. Wake Noise Sources

Next, an attempt to characterize the sources of noise generated in the wake structures of the propeller. To investigate the role of the wake flow, as well as contributions from the blade surface on the radiated sound, we plot the Lighthill Stress tensor (T_{ij}) within the computational domain.

$$T_{ij} = \rho u_i u_j - \sigma_{ij} + (p' - c^2 \rho') \delta_{ij} \quad (4.2)$$

σ_{ij} is viscous stress tensor, δ_{ij} is the Kronecker delta, $p' = p - p_o$ and $\rho' = \rho - \rho_o$ and c is the speed of sound. The subscripts “o” denotes the ambient conditions. The entropy component $(p' - c^2 \rho') \delta_{ij}$ is usually neglected. Applying Reynolds decomposition to Lighthill Stress tensor, we can show that it has a steady part that produces no sound, and unsteady part. So, more emphasis is put on the unsteady part. The Lighthill tensor reduces then to:

$$T'_{ij} = (\rho u_i u_j)' - \sigma'_{ij} \quad (4.3)$$

For shear flows (wakes), the viscous component is usually neglected. The unsteady part of $T_{ij} = (\rho u_i u_j)'$, which contributes to quadrupole term is plotted.

Conversion to Cylindrical Coordinate System

Due to the physics of the wake structures, it would be better to evaluate the structures in a cylindrical coordinate system. For this, we transform our Cartesian coordinates to cylindrical using the following relations.

First, we transform the orientation of our axes so as to obtain a better representation in cylindrical coordinate system that is Z axis being in the wake direction:

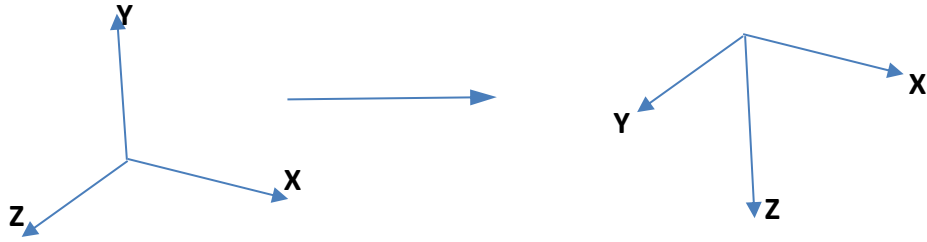


Table 4.1

Transformation of Cartesian coordinate system

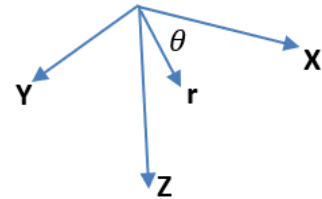
Current Cartesian	Transformed Cartesian
X	X
Y	Z
Z	Y
U	U
V	W
W	V

With:

$$\theta = \arctan(Z/X); r = \sqrt{x^2 + z^2}$$

$$u_r = u \cos \theta + w \sin \theta; u_\theta = w \cos \theta - u \sin \theta$$

$$u_z = v$$



The Lighthill tensors are extracted from the solver in runtime in Cartesian coordinate system in a form of symmetric tensors. From that, they are converted using the following

Mohr's tensor transformation:

$$\begin{bmatrix} S_{rr} & S_{r\theta} & S_{rz} \\ S_{\theta r} & S_{\theta\theta} & S_{\theta z} \\ S_{zr} & S_{z\theta} & S_{zz} \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} S_{xx} & S_{xy} & S_{xz} \\ S_{yx} & S_{yy} & S_{yz} \\ S_{zx} & S_{zy} & S_{zz} \end{bmatrix} \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (4.4)$$

Taking only the symmetric components of the tensor, it reduces to:

$$S_{rr} = (S_{xx}\cos\theta + S_{xy}\sin\theta)\cos\theta + (S_{xy}\cos\theta + S_{yy}\sin\theta)\sin\theta \quad (4.5)$$

$$S_{r\theta} = -(S_{xx}\cos\theta + S_{xy}\sin\theta)\sin\theta + (S_{xy}\cos\theta + S_{yy}\sin\theta)\cos\theta \quad (4.6)$$

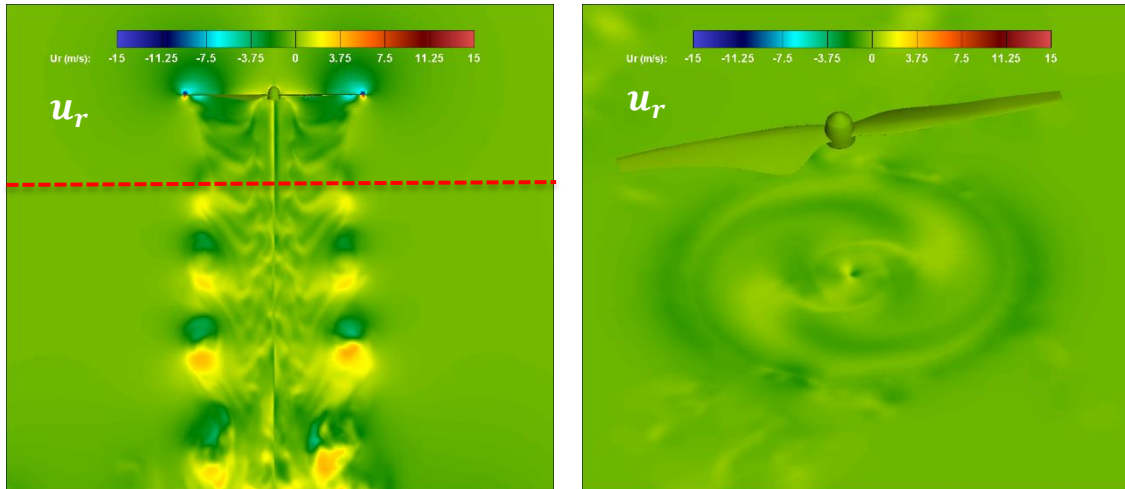
$$S_{\theta\theta} = (S_{xx}\sin\theta - S_{xy}\cos\theta)\sin\theta + (-S_{xy}\sin\theta + S_{yy}\cos\theta)\cos\theta \quad (4.7)$$

$$S_{rz} = S_{xz}\cos\theta + S_{yz}\sin\theta \quad (4.8)$$

$$S_{\theta z} = -S_{xz}\sin\theta + S_{yz}\cos\theta \quad (4.9)$$

$$S_{zz} = S_{zz} \quad (4.10)$$

Where S represents a tensor, which in our case would represent the total fluctuation tensor $(\rho u_i u_j)'$. To better understand which primitive component contributes most to the total fluctuation tensor, we plot out the components of the velocity fields in the cylindrical form.



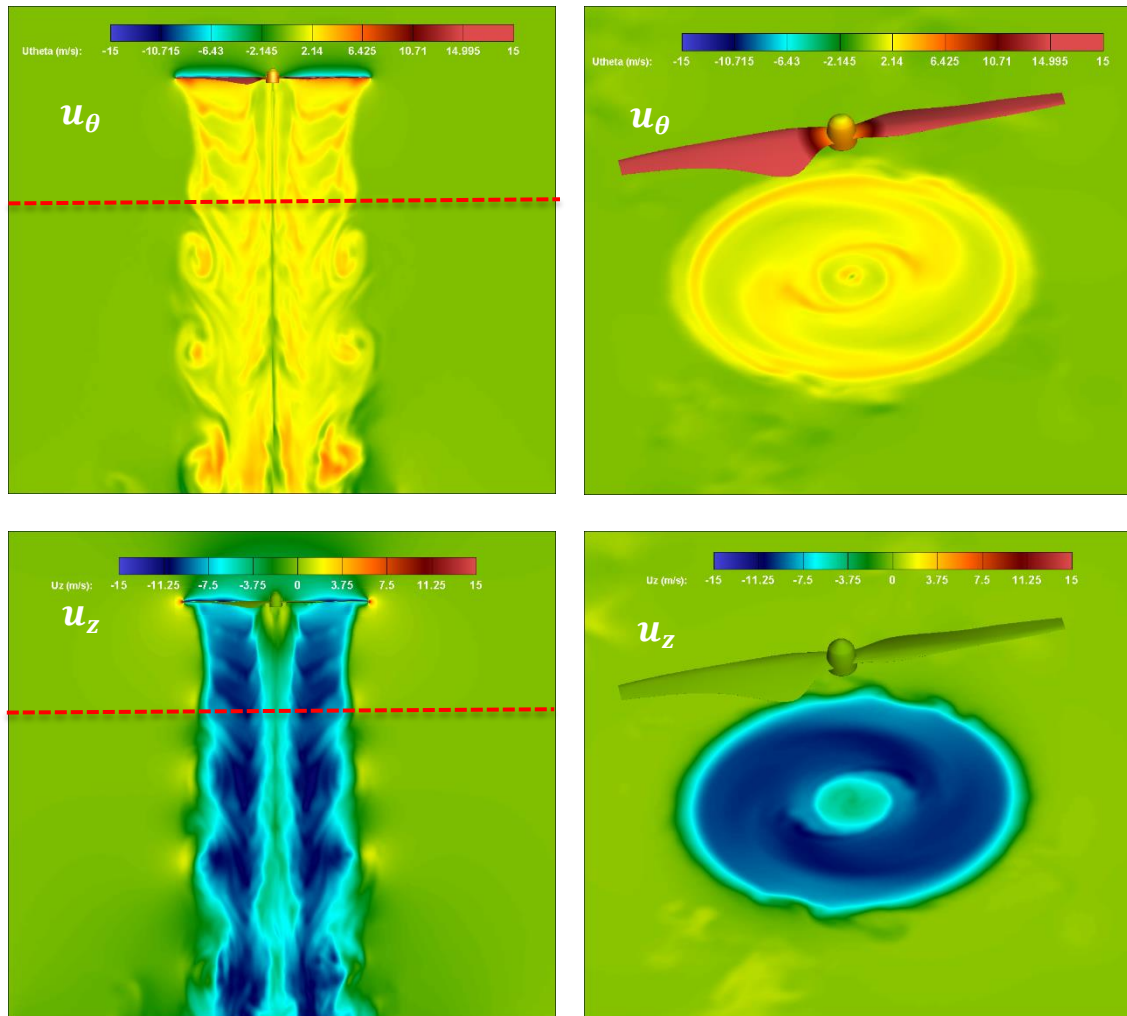


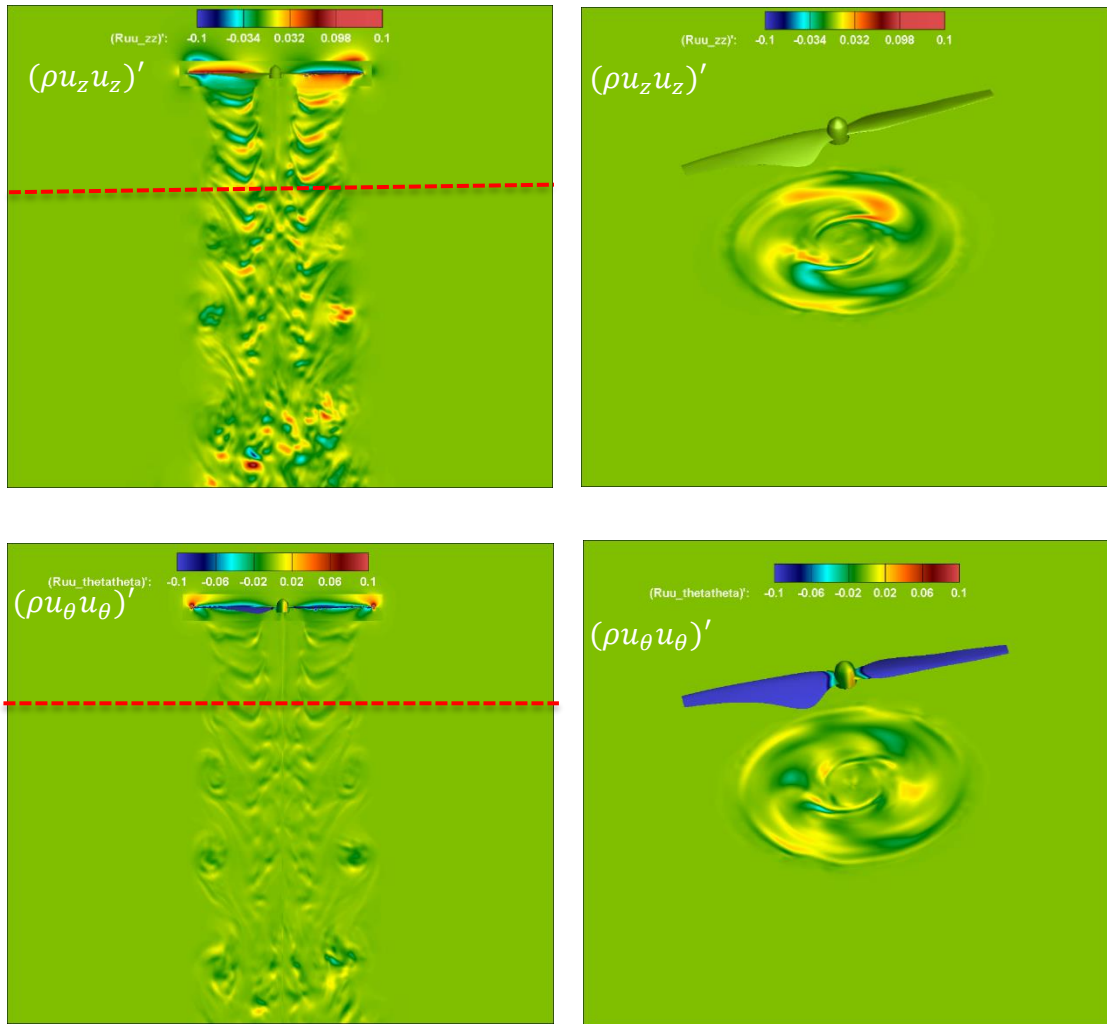
Figure 4.18 Vertical and Horizontal Snapshots of u_r , u_θ , and u_z

From the snapshots above, we can draw a few conclusions. On the blade surface, the azimuthal component of the velocity is noted to have the greatest magnitude with a value of approximately 75 m/s. For uniformity, the plots are scaled with the same range. The radial velocity is seen to be marginally greater than the axial velocity component. This is usually true for non-translational rotors (hovering condition). The axial component has no contribution on the blade surface. Looking at the wake, we can notice the downward convection of the flow is the strongest, which is in the axial component of the velocities, with a downward convection speed of about 12 m/s. The swirling nature of the wake is

seen predominantly in the azimuthal component of the wake.

Total Fluctuating Component plots

We now plot out the total fluctuating components $(\rho u_i u_j)'$:



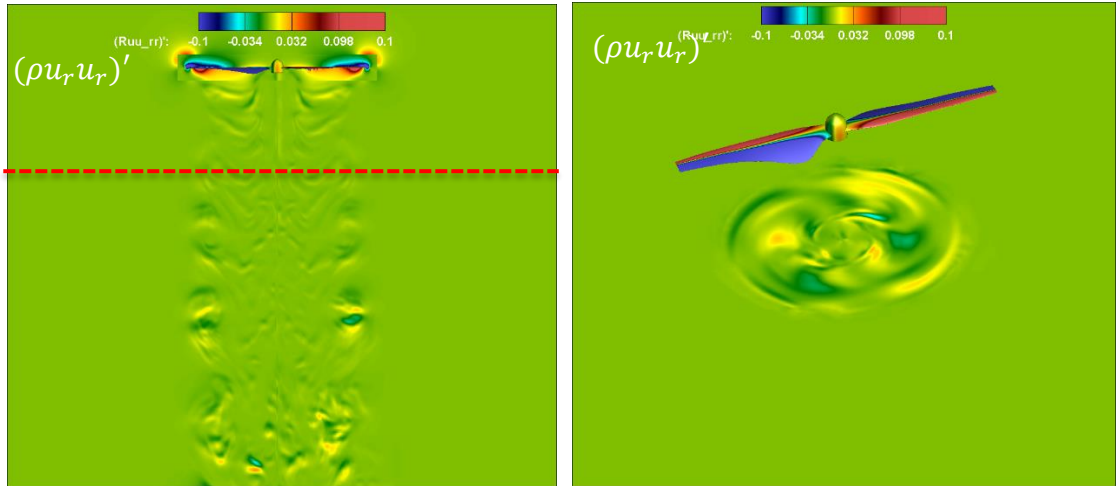
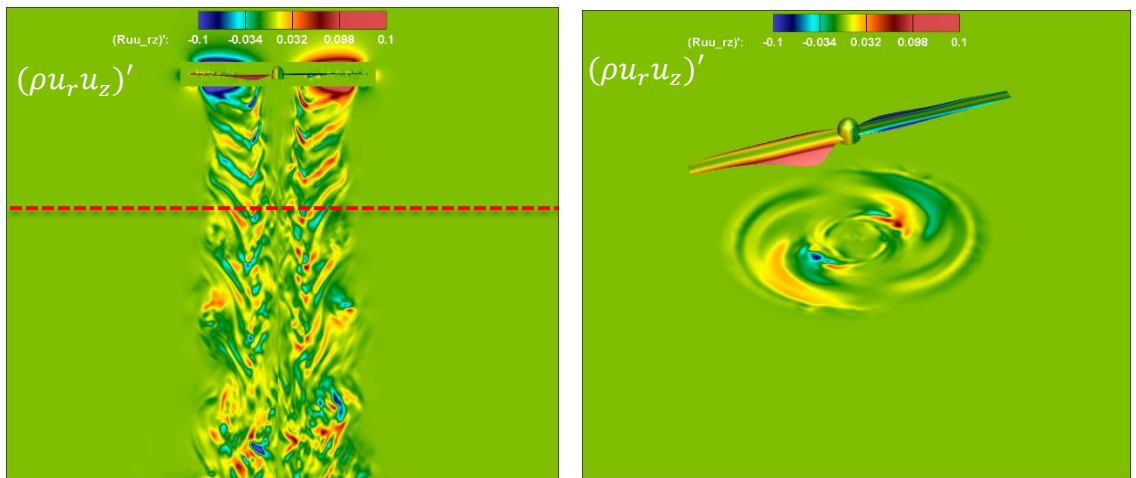


Figure 4.19 Snapshots of the normal components of $(\rho u_i u_j)'$



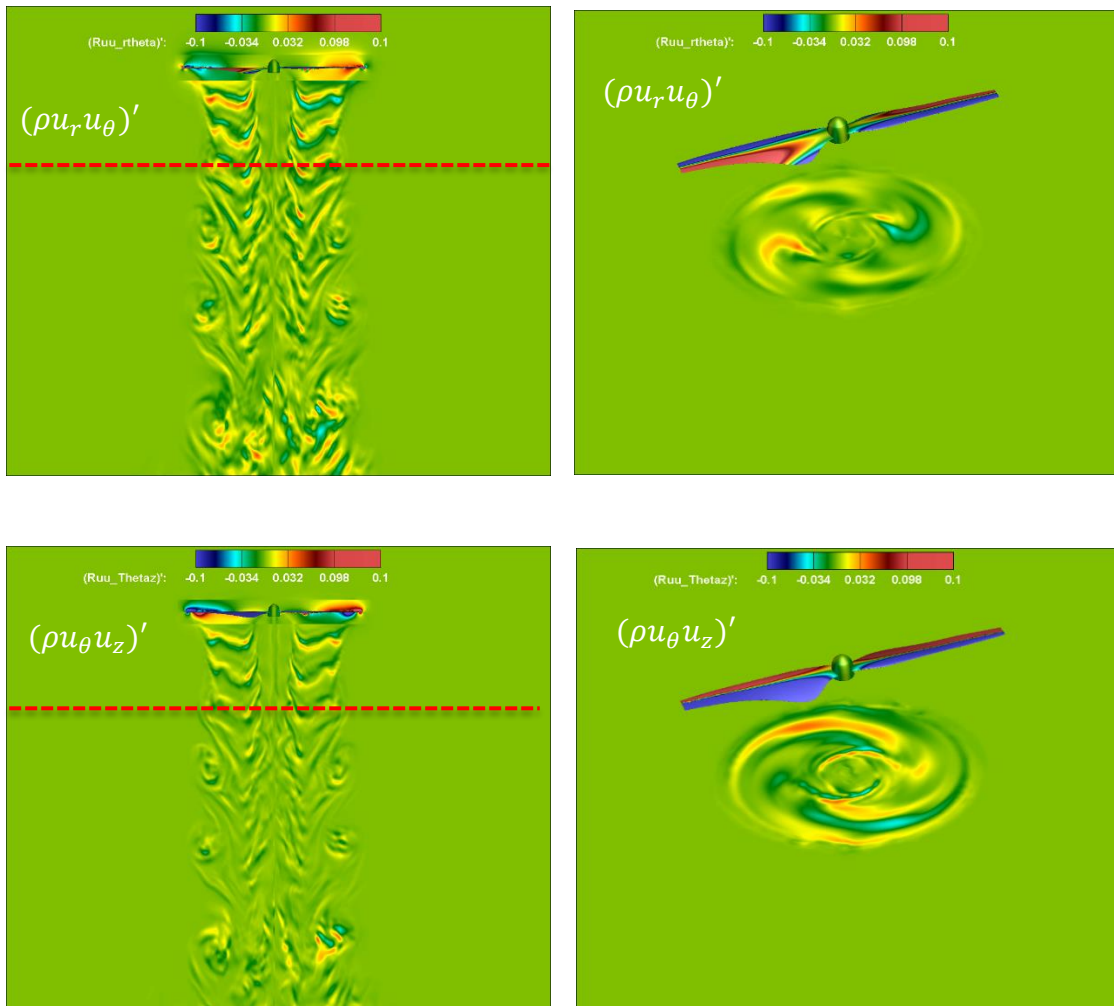


Figure 4.20 Snapshots of the shear components of $(\rho u_i u_j)'$

Contributions of Lighthill Tensors on blade surface

A few observations can be made based on the above plots. On the blade surface, the total fluctuation components are greatly represented, except for the $(\rho u_z u_z)'$. This can be attributed to the convection of the flow downstream.

Contributions of Lighthill Tensors in the wake

Most of the representation is seen in the wake structure. We see the physics of the swirl of the wake in the θ components of the total fluctuating part of the tensors. The

$(\rho u_z u_z)'$ component is seen to have the strongest magnitudes among the normal components. This further validates the downward convection of the flow as a strong source in the Lighthill tensors. Looking at the shear components, we observe the radial components being the strongest contributors to the noise generation from the wake.

We can conclude that the shearing effects of rotor rotation contributes significantly to the turbulence noise in the wake. All these components are well represented in the wake region, which would be included in the surface integral terms of the FWH equations as fluid variables, their fluctuations, and their time derivatives. Neglecting Lighthill's Stress tensors on the blade surface would impede the accuracy of the far-field noise predictions.

4.4. Far-Field Acoustics

As previously discussed in chapter 3, the Farassat's 1A formulation of the Ffowcs-William Hawking's equations are utilized in this study. The equations are simplified for stationary control surfaces and reiterated below. The third-party library "libAcoustics" is implemented in OpenFOAM v4.1. Cylindrical control surfaces are used for all the far-field computations from here on. The computed data on the cylinder provides input to the acoustic solver. The following unsteady flow data is needed as input: the velocity of the FW-H surface if it is moving (in our case, the surface is stationary, so it is zero), the speed of sound, the free-stream values of pressure and density, the fluid pressure and density variables, the observer location, and the sampling frequency. A sampling rate of 1.52×10^{-5} seconds, comparable to experimental data by Intaratep et al (2016) is used. Data is collected after 30 rotor rotations. All subsequent results are compared with this experimental data. The resulting Power Spectral Density (PSD) of the far-field noise frequency spectrum obtained based on the pressure fluctuation time history on the FW-H

surface is shown in the figures below. The results reveal that our computations accurately predict the dominant noise harmonic at the blade passage frequency (BPF) 200 Hz.

$$4\pi p(x, t) = \int_{f=0} \left[\frac{\rho_0 \dot{U}_n}{r} \right]_{ret} dS + \frac{1}{c} \int_{f=0} \left[\frac{\dot{L}_r}{r} \right]_{ret} dS + \int_{f=0} \left[\frac{L_r}{r^2} \right]_{ret} dS \quad (4.11)$$

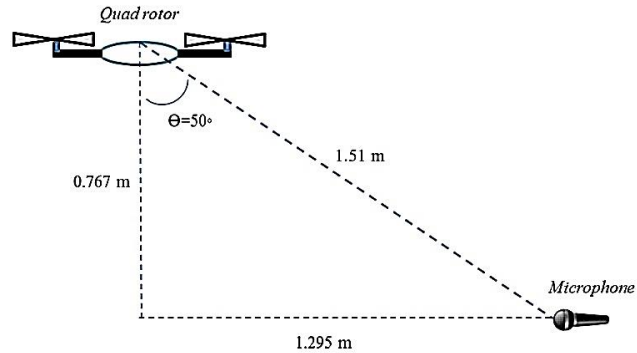


Figure 4.21 Experimental setup showing Far-field location (Intaratep et al. 2016)

4.4.1. FWH Results on Coarser Mesh

As discussed in chapter 2.3, a coarser grid was generated with only the acoustic resolution in mind. Due to the computational cost of running a very fine grid for acoustic data sampling, this coarser grid was built to avert this issue to obtain enough data sample size for analysis. To justify the grid quality of the coarser grid for acoustic predictions, we look at the resolution in the near field. The near field region spans $2.5R$ from the center of rotation. The number of cells in this region is approximately 6 million cells. An average cell size of $0.003m$ is maintained in this region. This corresponds to a conservative acoustic resolvable frequency of $\sim 1.5kHz$. This is more than enough to resolve our frequencies of interest in the spectrum. As such we use this grid for parametric studies of the ideal location of the FWH control surface.

Shape of FWH Control Surface

Three control surfaces are considered:

1. Cylinder with no caps
2. Cylinder with only top cap
3. Cylinder with both caps

According to Lockard and Casper (2005), there might be some introduction of errors in the surface integration when non-acoustic disturbances or convecting vortices pass through the FWH surface. This is due to the fact that the quadrupole source term is not included in the surface integrals. Ikeda et al (2017) proposed an approximated quadrupole surface integral to rectify this problem. This correction term is not considered in this formulation.

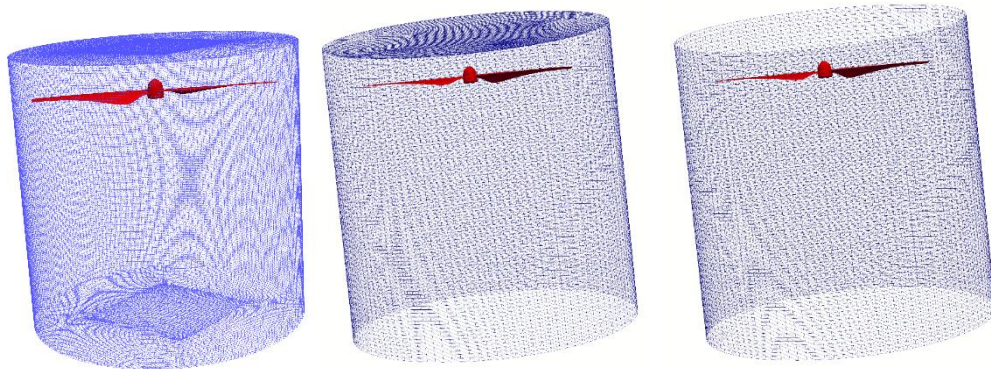


Figure 4.22 FWH control surfaces considered

The FWH cylinders are placed at 1.2 Rotor radii away from the center of rotation. Given the location of the FWH surface, the maximum cell size in the region is seen to be about 0.0025m, which corresponds to about 1600Hz cut off frequency. With this grid size, approximately 33,000 data samples are collected. Hanning windowing with 50% overlap amounting to 7 windows of 8192 samples is used. The results are shown below:

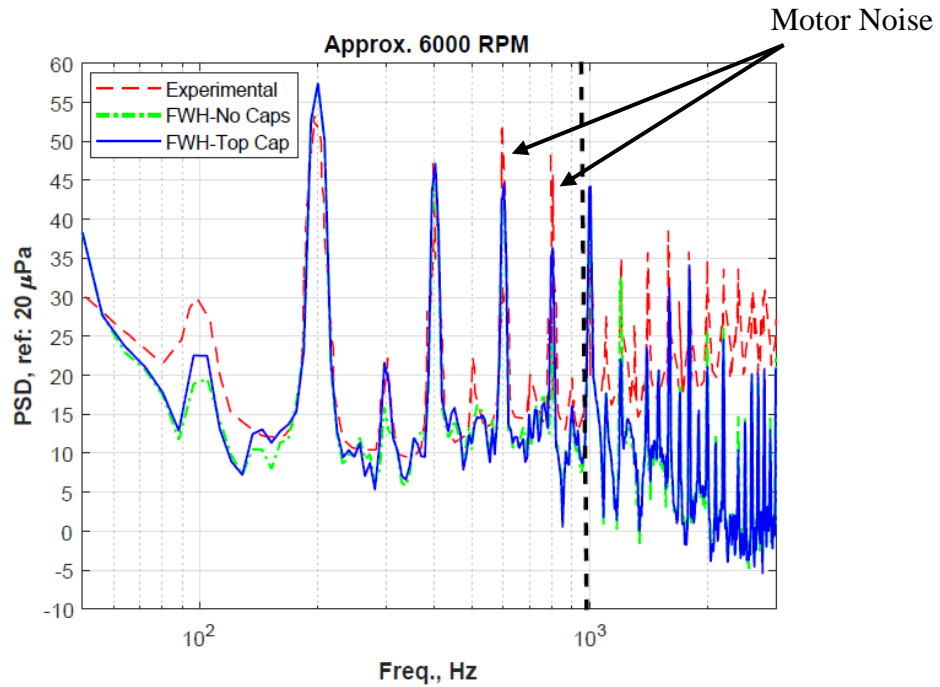


Figure 4.23 Power Spectral Density of top-capped, and open-capped control surfaces

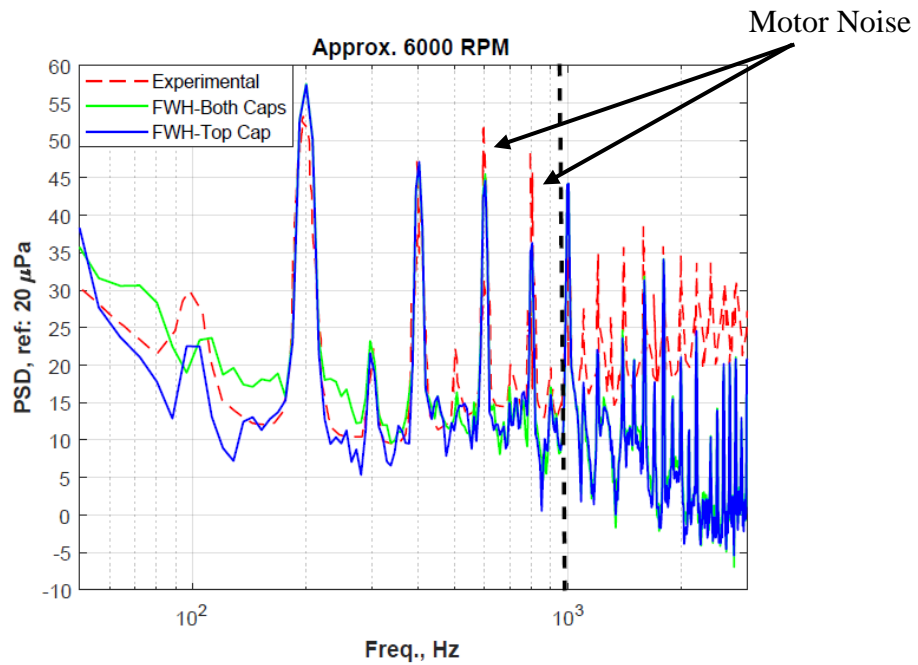


Figure 4.24 Power Spectral Density of top-capped, and both-capped control surfaces

From the results, we can see that the capped FWH surfaces give better peak to peak predictions compared with the open surface. We also notice some over-prediction of the

broadband aspects with the bottom capped surface. This phenomenon is also reported by Meng et al (1996), as well as Ikeda et al (2017). This is due to the introduction of errors from the passage of fluid momentum through the bottom surface. In such cases, a quadrupole source correction term is included in the formulations by approximating the volume integral as a surface integral in the direction normal to the rotor plane (Ianniello, 1999). This correction is implemented in codes such as the WOPWOP2+ code. Moving forward, we focus only on the top-capped FWH surface as our ideal control surface.

Location of the FWH Surface – Radial Location Effect

A degree of accuracy can be achieved by simply repositioning the FWH control surface closer to the rotor. On the other hand, moving away from the center of source closer to the far-field should in theory improve accuracy of prediction, provided the grid quality stays same. By extending the cylinder to a distance of 1.4 Rotor radii, the largest mesh size becomes $\sim 0.004\text{m}$, corresponding to a conservative cut-off frequency of $\sim 1000\text{Hz}$. We should expect a reduction in accuracy because of the reduction in resolution.

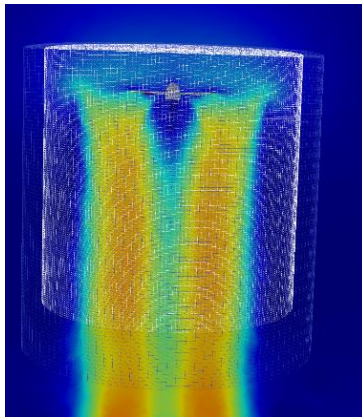


Figure 4.25 FWH surfaces at different Radii distances

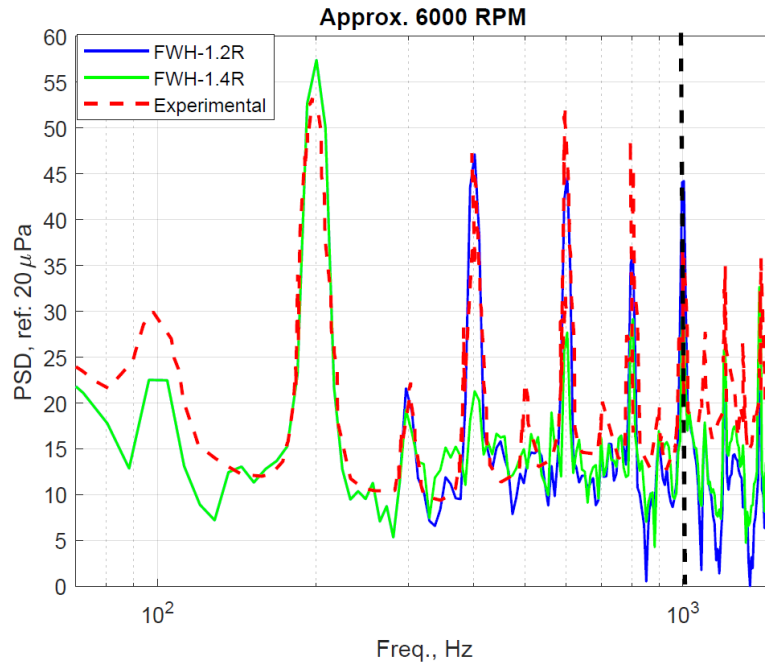


Figure 4.26 Power Spectral Density plots of the two FWH locations.

The figure above shows that extending the FWH control surface farther from the rotor decreases the accuracy of the prediction. It should be noted, however that this might not hold true for a very fine mesh case. This is because for the coarse grid, the farther you move away from the center of rotation, the coarser the grid becomes, unable to meet the grid resolution requirements for the acoustics. For a sufficiently fine grid, the grid would still be fine enough for the acoustic resolution, as such would marginally improve the prediction. The optimal location of the FWH for this grid is seen to be at 1.2R with only the top cap of the cylinder. To better visualize the prediction of the broadband levels, we only show the spectra up to ~ 1.1 kHz.

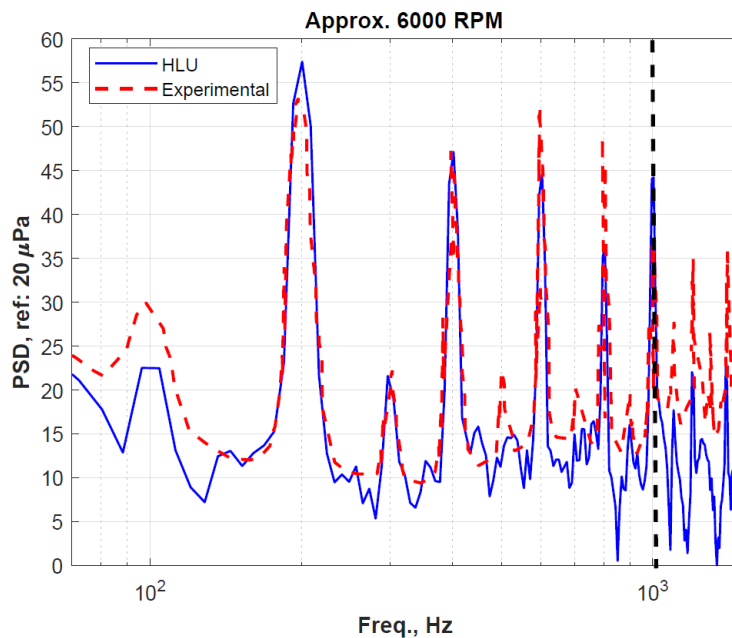


Figure 4.27 Power Spectral Density of HLU compared with Experimental Data

Permeable FWH Surface Vs Impermeable Surface

We investigate the effects of using the blade surface as our control surface for far field noise prediction as opposed to using our cylindrical permeable off-body surface.

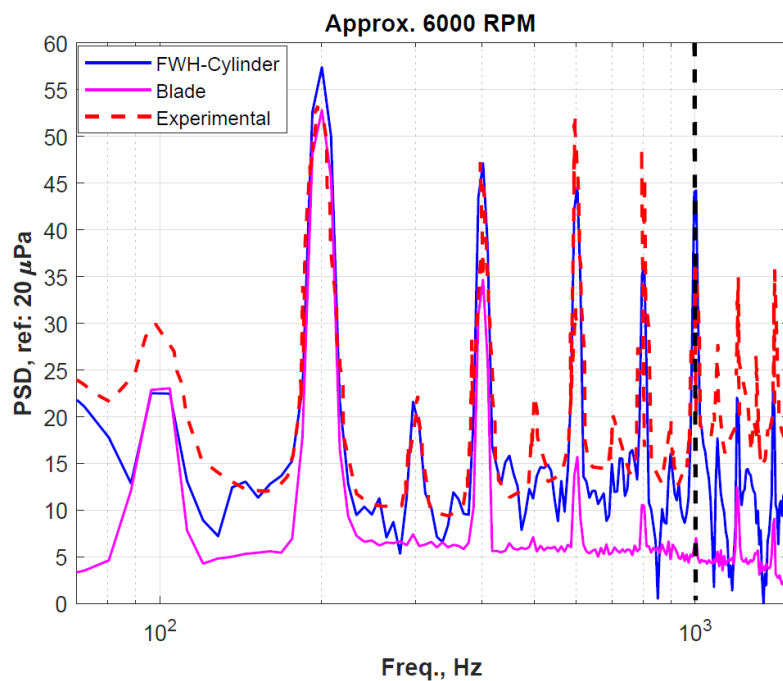


Figure 4.28 Power Spectral Density of top-capped, and Blade surface control surfaces

We can see that using the blade surface as the source does not do well to predict the peaks after the first harmonic. We also notice a reduction in the overall noise levels of the broadband aspects of the spectrum in the low to mid-frequency range. This is due to the fact that the wake noise source is not included. The effects of the noise from the wake is clearly seen to contribute greatly to the broadband noise.

Directivity of Propeller Noise

The directivity of the FWH-1.2R control surface is shown below. Good correspondence is seen in the predictions compared with works by Zawodny et al (Zawodny, Boyd, and Burley, 2016) (see *Figure 1.6* for experimental setup).

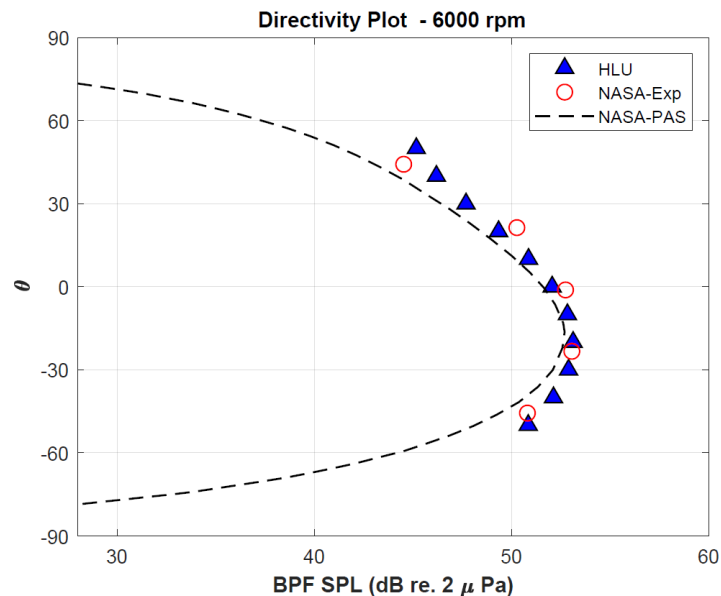


Figure 4.29 Directivity plot at 1st Blade passage frequency.

4.4.2. FWH Results on Fine Mesh

The fine grid is used to predict the far-field noise using the optimal FWH location of 1.2R. Due to the size of the grid, and computational cost, three windows of 8192 samples are obtained and used for the spectral processing.

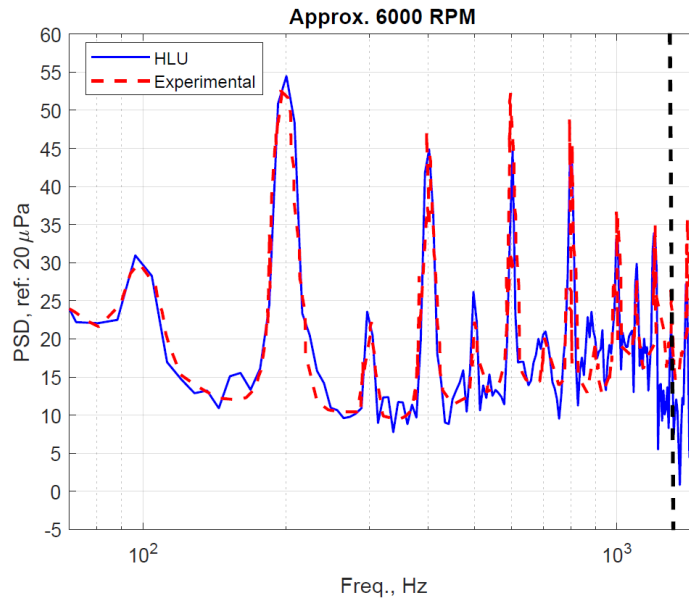


Figure 4.30 Fine Grid Power Spectral Density plot compared with experimental data.

Looking at the above results, we see matching broadband levels in the low to mid frequency range. We also notice an increase in resolvable frequency from the coarser grid. The jagged nature of the spectra is attributed to the relatively smaller number of samples compared with experimental data.

4.4.3. Remarks

From the obtained spectrum comparison, we notice jaggedness of the broadband spectra. This could be due to insufficient sampling size. The experimental data uses 511 samples, each with 8192 data points. Numerically, this is not feasible. The data presented in our study only uses 3 samples of 8192 data points for the fine case and 7 windows for the coarser case. Overall, we do observe the good agreement in the broadband noise levels in the low-to-mid frequency range. Another factor of concern is that we use URANS (not LES) at the blade surface, as demonstrated in *Figure 4.31* below. Thus, the high-frequency content is not resolved in the near-wall region. Finally, the turbulence

model introduces certain dissipation that can also introduce some errors in the predicted levels of the high-frequency noise sources.

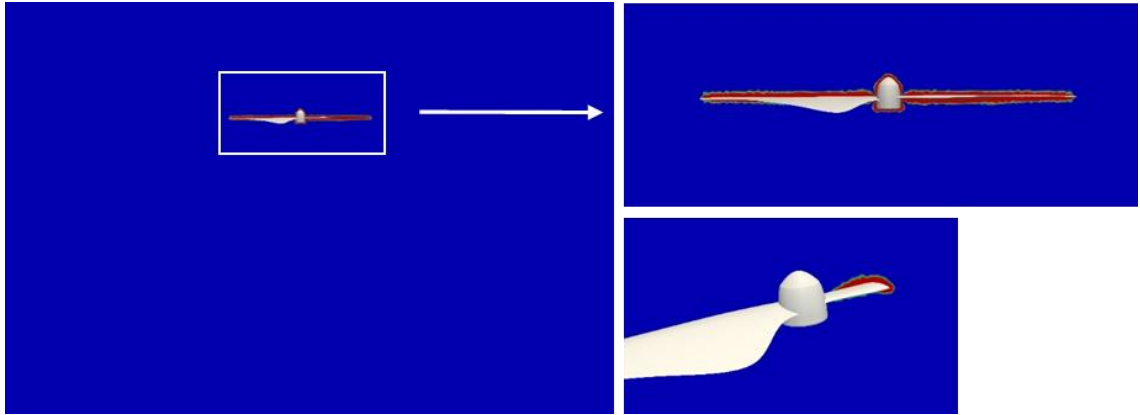


Figure 4.31 Regions of RANS (red) and LES (blue) in computational domain

4.5. Contributions to the Far-Field Noise

We ask the question, what contributes to the far-field noise generation? To address this question, we attempt to plot the variables on the FWH control surface contributing to the far-field. The simplified FWH equation for a stationary control surface is reiterated:

$$4\pi p(x, t) = \int_{f=0} \left[\frac{\rho_o \dot{U}_n}{r} \right]_{ret} dS + \frac{1}{c} \int_{f=0} \left[\frac{\dot{L}_r}{r} \right]_{ret} dS + \int_{f=0} \left[\frac{\dot{L}_r}{r^2} \right]_{ret} dS$$

The point on the enclosed FWH surface is defined by (r, θ, z) . The observer (microphone) located at (r_o, θ_o, z_o) . For clarity, we use R to denote the radiation vector and its unit vector is \hat{R} . Note that the subscript (R) denotes projection of any vector in the radiation direction. The surface integral equations can be re-written more compactly as:

$$4\pi p'(x, t) = \int_{f=0} \left[\frac{\dot{m}}{R} + \frac{\dot{L}_R}{cR} + \frac{\dot{L}_R}{R^2} \right]_{ret} dS \quad (4.12)$$

Here,

$$m = \rho u_n$$

$$L_R = (P_{ij} \hat{n}_j + \rho u_n \mathbf{u}_i) \cdot \hat{R}$$

The above equation is reformulated in the cylindrical coordinate system and is plotted on the cylindrical FWH control surface. The following simplifications and relations are used:

On the sides of the cylinder, we use:

$$u_n = u_r; \quad m = \rho u_r$$

$$P_{ij} \hat{n}_j = \tilde{p} \hat{r},$$

$$\tilde{p} = (p - p_0),$$

$$L_R = \tilde{p} (\hat{r} \cdot \hat{R}) + m (u_r \hat{r} + u_\theta \hat{\theta} + u_z \hat{z}) \cdot \hat{R}$$

On the caps of the cylinder, we use:

$$u_n = u_z$$

$$m = \rho u_z$$

$$P_{ij} \hat{n}_j = \tilde{p} \hat{z}$$

$$L_R = \tilde{p} (\hat{z} \cdot \hat{R}) + m (u_r \hat{r} + u_\theta \hat{\theta} + u_z \hat{z}) \cdot \hat{R}$$

Contributions of thickness (mass) noise on far-field.

To investigate the influence of the thickness (mass flux) noise compared to the pressure term, we plot the following components on the entire cylindrical surface for visual analysis.

$$\frac{\partial m}{\partial t}; \quad \frac{1}{c} \frac{\partial \tilde{p}}{\partial t}$$

Noting that m is the mass flux normal to the surface.

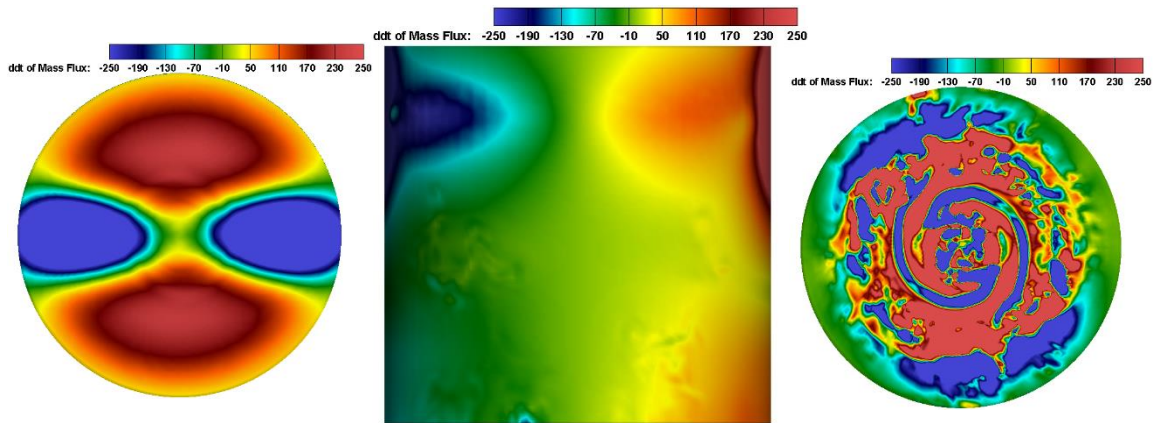


Figure 4.32 Time derivative of Mass flux distribution on the FWH surface: (a) Top cap
(b) cylinder surface, (c) bottom cap

This term contributes to the integrand of the thickness noise terms. The direction of the strongest magnitudes is observed to be in the radial direction in the rotor plane.

The Pressure term is also plotted similarly:

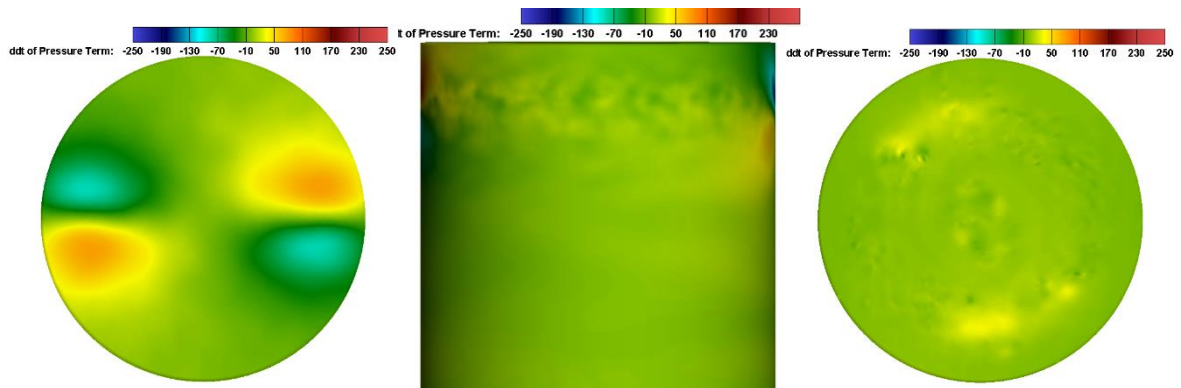


Figure 4.33 The Time derivative of the Pressure term distribution on the FWH surface:
(a) Top cap (b) cylinder surface, (c) bottom cap

Based on the visual representation of the above images, we can notice that the mass flux derivative is much higher than the pressure derivative. To further investigate these terms, linear plots at $Z=0$, -0.05 , and $-0.1m$ are taken along the circumference of the

cylinder, as well as at the surface of the top and bottom caps at 1.1 Radii.

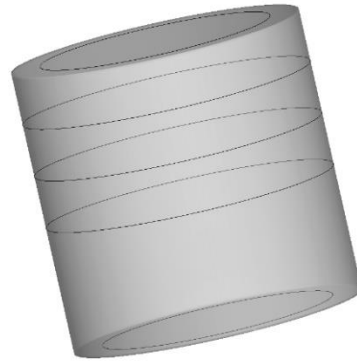
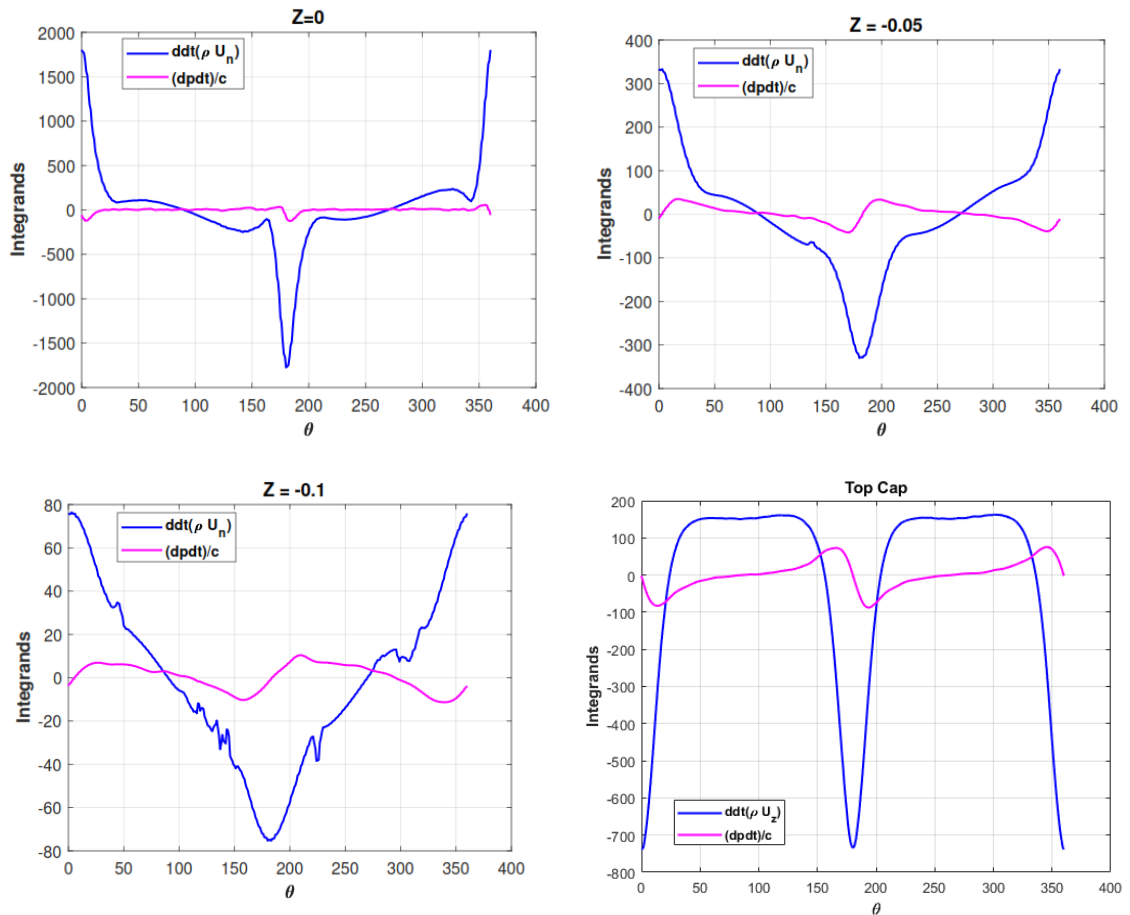


Figure 4.34 Location of linear slices around the cylinder



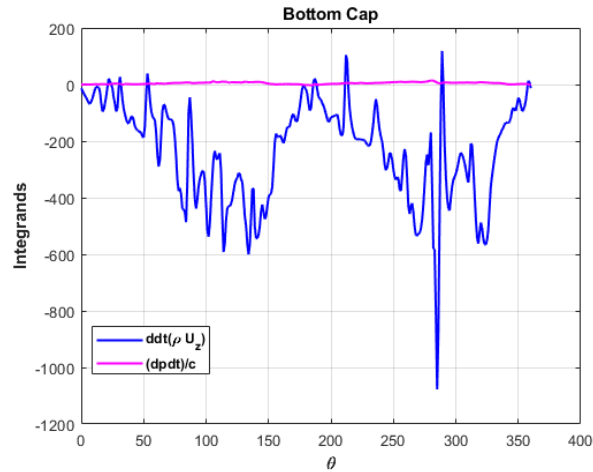


Figure 4.35 Linear Plots comparison between mass flux and pressure derivatives

From the above plots, we see a trend in the magnitudes of the mass flux derivatives as we move downward of the wake. We do see, however that the pressure derivatives are smaller in magnitude than those of the mass flux derivatives. This may be counterintuitive as generally; the pressure term should be greater. To understand what is going on, we look at how time derivatives are computed. A forward or backward temporal scheme would calculate the derivative as the difference between the quantity of two consecutive timesteps, divided by dt . In typical CFD code, this dt is usually very small, of orders of $\sim 10^{-7}$ s. Dividing by such a small number makes the derivatives quite large, as we see in the mass flux derivative. The same is seen in pressure derivatives but is divided by speed of sound as per the formulation used, thus has a reduced magnitude.

Typically, in FWH solvers, data is sampled at a much higher dt , and thus might not represent the results seen above.

Contributions of momentum fluxes on far-field

We attempt to answer the question, is the momentum term significant relative to the pressure term? To do this, we plot \dot{L}_R .

For the sides of the cylinder:

$$\frac{\partial \bar{p}}{\partial t} (\hat{r} \cdot \hat{R}) + \frac{\partial (\rho u_r u_r)}{\partial t} (\hat{r} \cdot \hat{R}) + \frac{\partial (\rho u_r u_\theta)}{\partial t} (\hat{\theta} \cdot \hat{R}) + \frac{\partial (\rho u_r u_z)}{\partial t} (\hat{z} \cdot \hat{R}) \quad (4.13)$$

And for the caps of the cylinder:

$$\frac{\partial \bar{p}}{\partial t} (\hat{z} \cdot \hat{R}) + \frac{\partial (\rho u_z u_r)}{\partial t} (\hat{r} \cdot \hat{R}) + \frac{\partial (\rho u_z u_\theta)}{\partial t} (\hat{\theta} \cdot \hat{R}) + \frac{\partial (\rho u_z u_z)}{\partial t} (\hat{z} \cdot \hat{R}) \quad (4.14)$$

Before we show these components, we first show the flow field in the vicinity of the FWH surface. We cut a horizontal slice at $Z = -0.02m$ below the propeller.

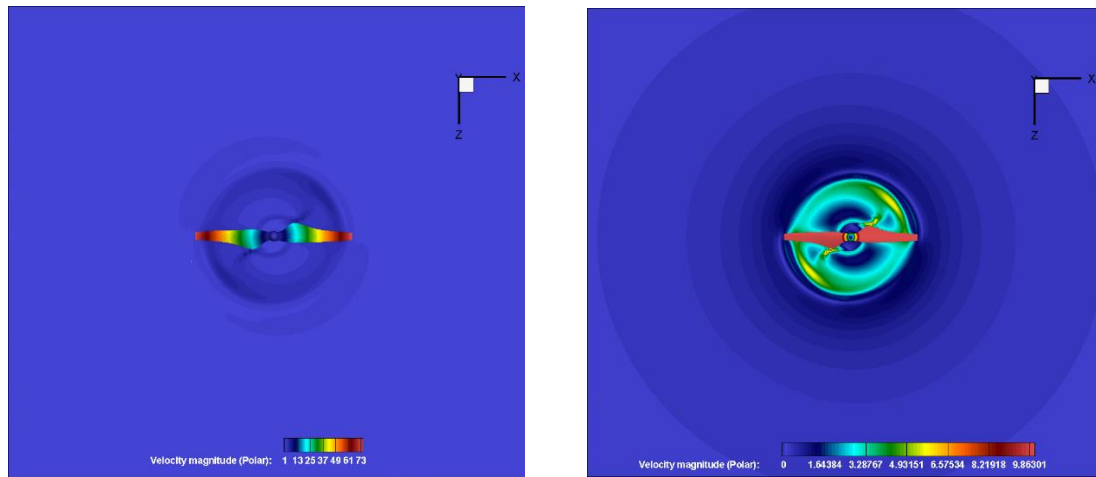


Figure 4.36 Horizontal slice of Velocity magnitudes unscaled (left) and scaled (right)

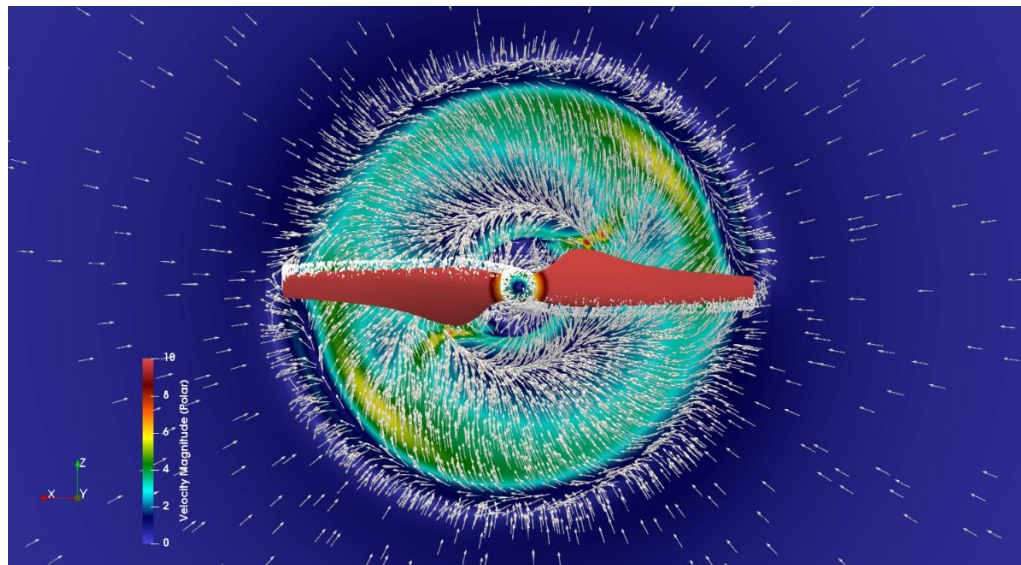


Figure 4.37 Velocity Magnitude contours with polar coordinate vectors

In the above image, we see the physics of the flow in the vicinity of the rotating propeller. We see the suction effects and the rotation of the flow. The components of the velocities are similarly plotted:

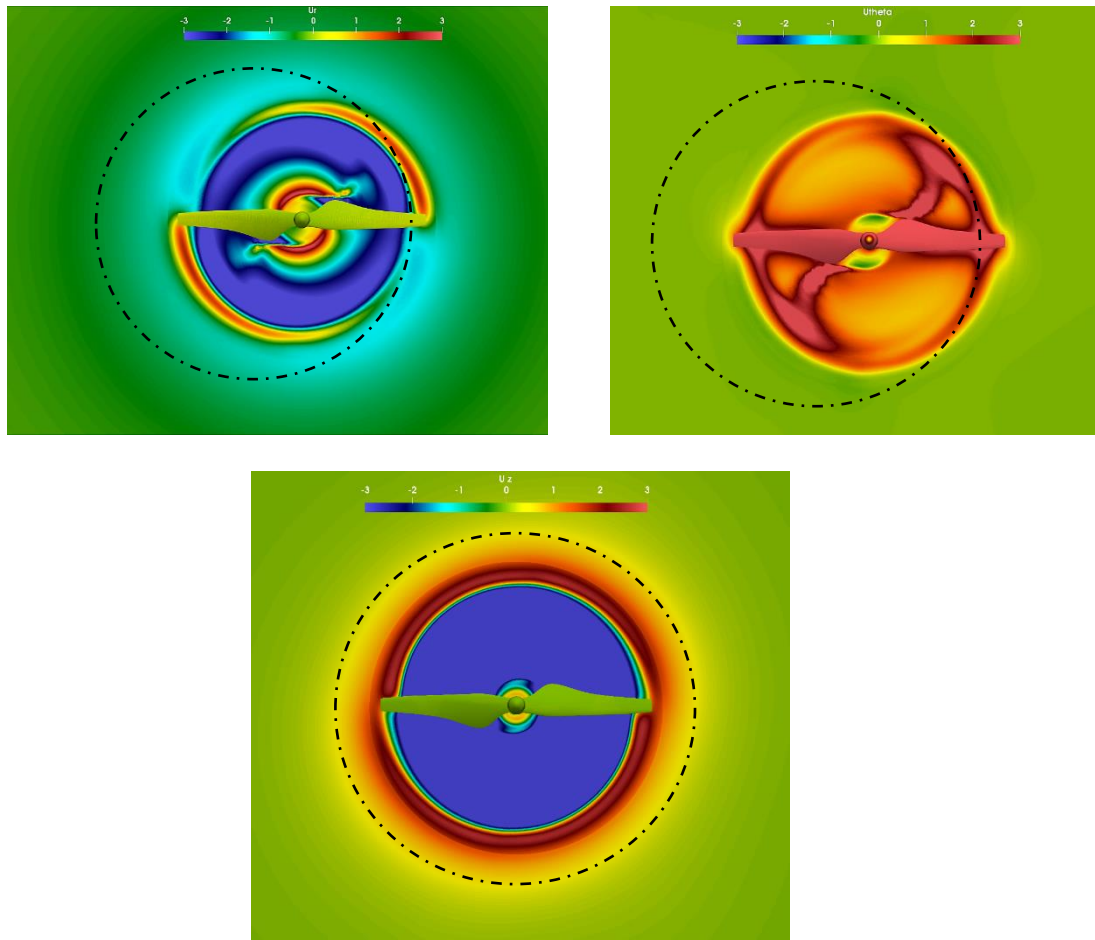


Figure 4.38 Velocity components in polar coordinate system.

From the above plots, the location of the FWH cylinder is shown with the dashed circle. The largest contributing component as seen above is the radial component with $\sim 1.5\text{m/s}$. Next is the axial component. This is due to the fact that most of the azimuthal velocity is driven downwards and outwards (radially).

The time derivatives of the momentum flux terms are plotted:

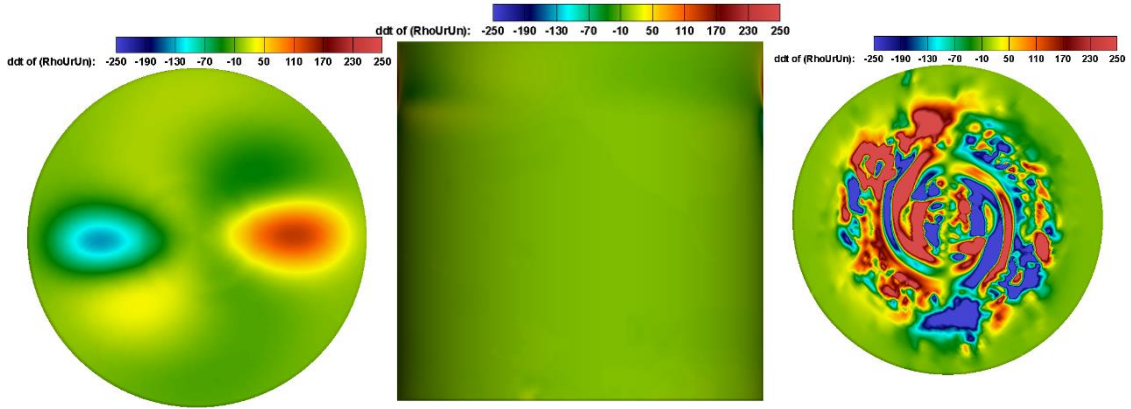


Figure 4.39 Time Derivative of Momentum Flux $\rho u_r(u_n)$ distribution on FWH surface

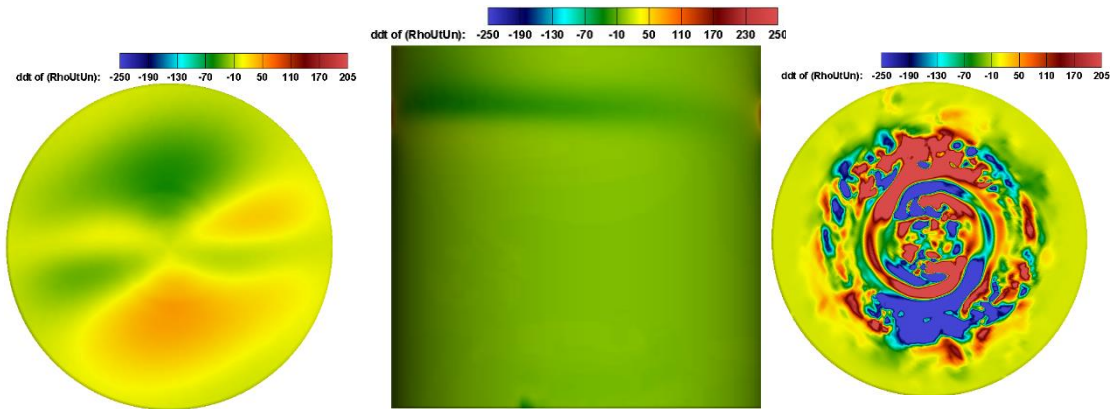


Figure 4.40 Time derivative of Momentum Flux $\rho u_\theta(u_n)$ distribution on FWH surface

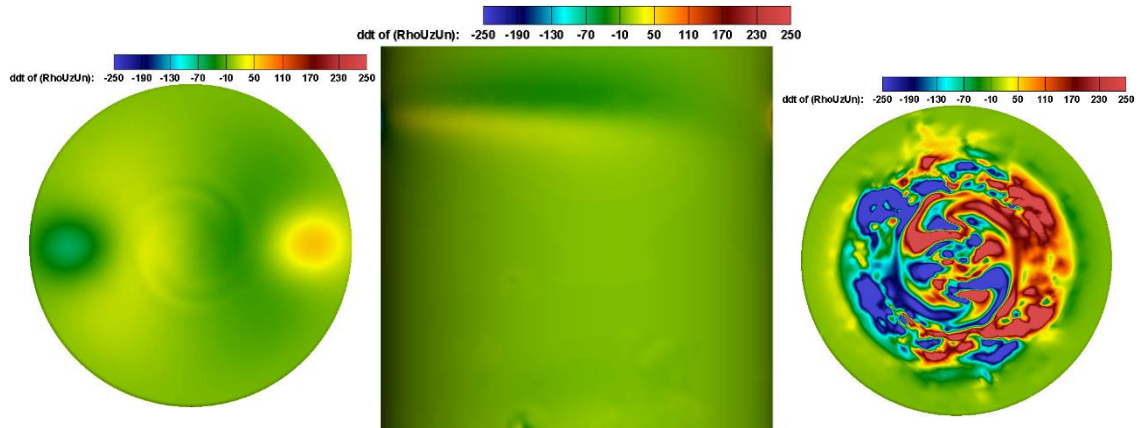


Figure 4.41 Time derivative of Momentum Flux $\rho u_z(u_n)$ distribution on FWH surface

Again, the above plots might not give a good picture of which term is largest or more significant. Thus, we look at the linear plots below:

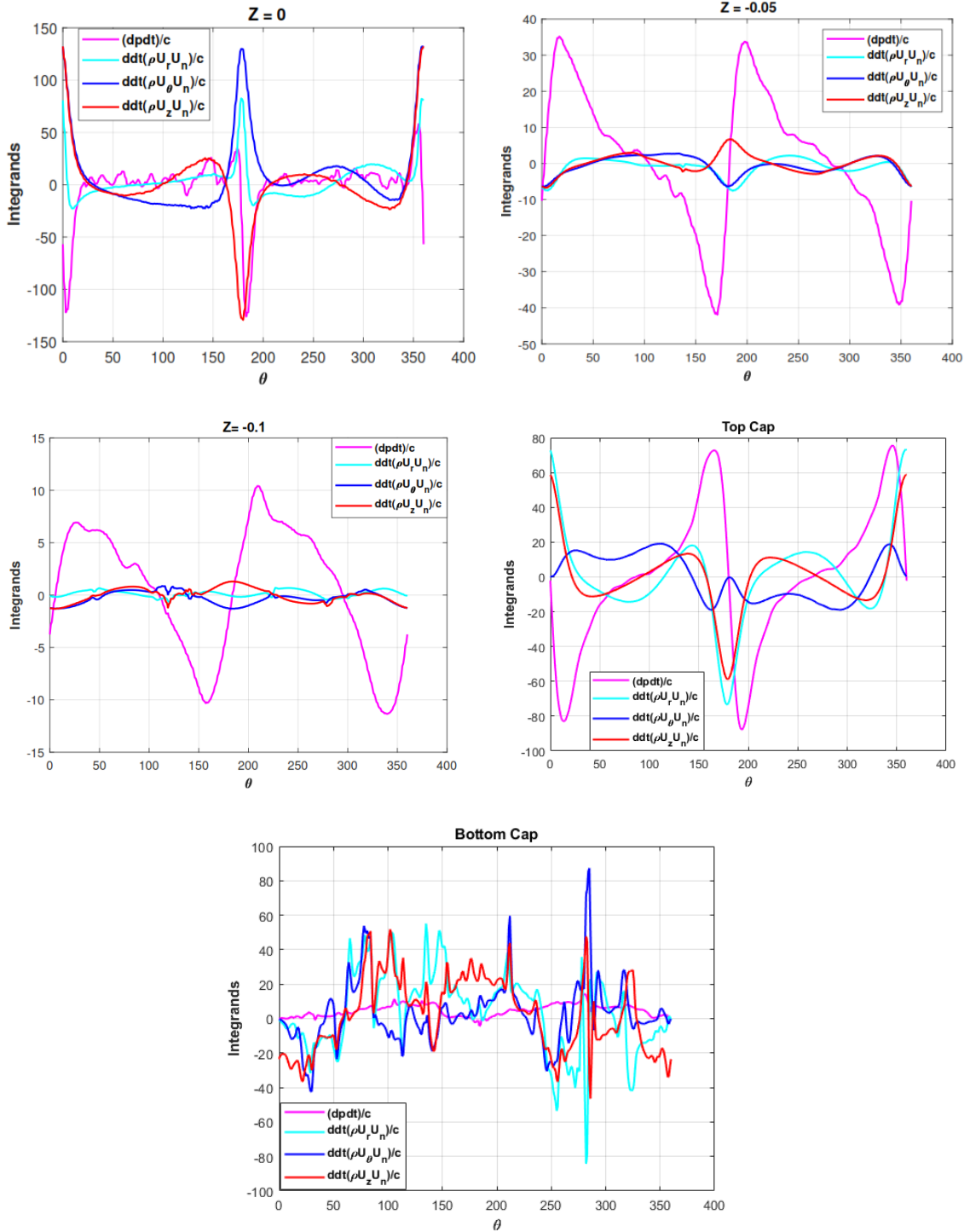


Figure 4.42 Linear plots of Time derivatives comparing Loading terms

It can be observed from the linear plots that the dominant term in the integrands is the pressure term except at the rotor plane, $Z=0$, and the top cap, where contributions of the

momentum flux derivatives are significant, and bottom cap, where the momentum flux derivative overshadows the pressure term. This is due to the convection of the flow through the bottom cap.

An analysis of the above linear plots shows that there is a general dominance of the pressure terms, if we do not account for the discrepancies of the time derivatives. The momentum fluxes, however, are seen to be quite significant in the bottom cap mainly the “rz” component. This can be construed to be in part the effects of the downward convection of the flow as well as the radial translation of the fluid momentum impacted by the rotating blade.

Generally, as seen previously, inclusion of the end cap does have significant effect on the far-field noise prediction. This is due to the introduction of high momentum terms on the control surface. Inclusion of the end cap does have a detrimental effect on the far-field noise prediction if a correction term is not implemented.

5. Conclusions

Simulation of an isolated propeller and its noise radiation was performed. A Hybrid LES-URANS (HLU) is adopted where URANS is used only near the solid surfaces, and LES is used elsewhere. This enabled the prediction of the unsteady flow structure, and the acoustic field with a realistic resolution requirement. A spherical grid structure with grid stretching is used as it seen to better dissipate the acoustic waves generated by the propeller. A rotation/curvature correction modification is implemented in the turbulence models in the open source CFD code OpenFOAM v4.1.

Current methods of predicting far-field noise of propellers utilize the blade surface as the control surface, by assuming an impermeable surface. A different approach is used in this study. The Farassat's 1A formulation ignores the quadrupole source terms, but by utilizing a control surface that encases both the blade surface, as well as the wake, little to no correction is needed to account for the quadrupole source terms as they influence the physical variables on the FWH to be integrated (Pressure, momentum fluxes, and the mass fluxes). To avoid over-prediction of the noise levels, it is recommended to use a control surface that does not have an end cap that intersects with the flow field. If that is used, a quadrupole correction term must be utilized in the formulation to avoid this.

The following observations and conclusions were obtained based on the present simulations of an isolated propeller:

- The acoustic spectra of the isolated propeller were dominated by high and sustained tones at the blade passage frequencies and their harmonics.
- The predicted sound sources can be split into two regimes - one associated with the propeller blade surface and the other associated with the unconfined, swirling,

wake-like flow below the propeller.

- On the propeller bottom surface, the pressure fluctuations seem to be intense at the mid-to inner radius of the propeller. This is attributed to the blade steady loading, while for the top surface it is more intense towards the outer radius.
- The pressure fluctuations in the wake region, while less intense than that on the blade surface, still contributes significantly to the radiated sound as broadband noise.

5.1.Limitations and Challenges

Several challenges were encountered in the presented research. Grid generation for numerical simulations involving acoustic resolution required care and attention. Multiple grids were generated in an attempt to dissipate the spurious waves. This generated a large grid cell size.

The current research was performed on Embry-Riddle Aeronautical University's VEGA cluster. The simulations were typically run on 360 CPU cores in parallel for the fine-grid case. A run time of approximately 650 CPU hours is typically needed for flow convergence. This corresponds to approximately one rotation per day.

Another challenge is data collection. To match experimental data, or to get comparable results, a large enough acoustic pressure sample size must be obtained. The experimental data compared in this thesis used 511 windows of 8192 samples each. This is unfeasible numerically. It takes approximately three weeks of simulation run time to generate one window of 8192 samples of data.

Another limiting factor was data storage. A typical time step reached a file size of approximately 100GB of data. To process acoustics and perform far-field acoustic

predictions requires a time history of fluid variables. This becomes an issue for the large grid size and limited the amount of data to be collected for better spectral analysis as well as flow structures.

5.2. Recommendations and Future Work

When generating a grid to resolve the acoustics as well as flow fields, it is recommended not to utilize a hybrid structured and unstructured grid. This tends to generate non-physical waves and disturbances in the transition points between the structured and unstructured part of the grid, which would destroy the acoustic field. It is recommended to use either a fully unstructured grid with good grid stretching, or a fully structured grid, especially for any form of LES or HLU simulation.

Acoustic damping region should be placed far enough such that no convective flow interacts with the region. This region tends to allow and dissipate acoustic waves but does not allow vortices or actual flow to pass through it and behaves like a solid wall.

This thesis explored the noise radiation of an isolated propeller in hover condition. With the insights observed, some future work would include further studies on the effects of multi-rotor interactions, fluid-structure interactions of propellers, as well as noise of propellers in translational motion.

Also, future work on the development of techniques for noise suppression of the propeller should be performed by exploring both passive and active methods as well as some geometric optimization for overall performance and acoustic gains.

REFERENCES

- Brentner, K. S, and Farassat, F., “Analytical Comparison of the Acoustic Analogy and Kirchhoff Formulation for Moving Surface,” *AIAA*, Vol. 36, No. 8, pp. 1379–1386, 1998.
- Crank, J., Nicolson, P., “A practical method for numerical evaluation of solutions of partial differential equations of the heat conduction type”. *Proc. Camb. Phil. Soc.* **43** (1): 50–67. (1947). doi:10.1007/BF02127704
- Di Francescantonio, P. 1997. “A New Boundary Integral Formulation for the Prediction of Sound Radiation.” *Journal of Sound and Vibration* 202 (4). Elsevier: 491–509.
- DJI, “DJI Self-Tightening Propeller set for Phantom 2 Vision”, 2018. URL https://www.bhphotovideo.com/c/product/1013324-REG/dji_part3_phanton_2_self_tightening_propeller.html
- DJI, “Phantom 3 Standard,” URL: <https://www.dji.com/phantom-3-standard>
- Farassat, F, and M K Myers. 1988. “Extension of Kirchhoff’s Formula to Radiation from Moving Surfaces.” *Journal of Sound and Vibration* 123 (3). Elsevier: 451–60.
- Federal Aviation Regulations, “Code of Federal Regulations,” *Part 14, Subpart H & Subpart K*, 2011.
- Feight, J. Jacob, J. D., Gaeta, R. J. “Acoustic Characterization of a Multi-Rotor UAS as a First Step Towards Noise Reduction”, *55th AIAA Aerospace Sciences Meeting Jan 2017*, DOI: 10.2514/6.2017-1174.
- Hubbard, H. H., “Aeroacoustics of Flight Vehicles: Theory and Practice”, *NASA Reference Publication*, Vol. 1: Noise Sources, No. 1258, 1991, pp. 1-145.
- Hubbard, H. H., and Regier, A. A., “Free Space Oscillating Pressures Near the Tips of Rotating Propellers” *NACA Report 996*, Washington, D.C., 1950.
- Goldstein, M. E., “Aeroacoustics.” *McGraw-Hill International Book Co.* 1976. ISBN: 9780070236851.
- Gutin, L. “On the Sound Field of a Rotating Propeller.” *NACA TM 1195*, 1948.
- Ianniello, S., “Quadrupole Noise Predictions Through the Ffowcs Williams–Hawkings equation”, *AIAA* Vol. 37, No. 9, September 1999.
- Ikeda T., Enomoto S., Yamamoto K., and Amemiya K., “Quadrupole Corrections for the Permeable-Surface Ffowcs Williams–Hawkings equation”, *AIAA* Vol. 55, No. 7, July 2017.

- Intaratep, N., Nathan, A., Devenport, W., Grace, S. M., and Dropkin, A. “Experimental Study of Quadcopter Acoustics and Performance at Static Thrust Conditions,” *22nd AIAA/CEAS Aeroacoustics Conference*, DOI: 10.2514/6.2016-2873.
- Israeli, M., and Orszag, S.A. “Approximation of radiation boundary conditions”, *Journal of Computational Physics*, 41:115–135. (1981).
- Lighthill, Michael J. 1952., “On Sound Generated Aerodynamically. I. General Theory.” *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 211:564–87. The Royal Society.
- Lim, T. B., Sankar, L. N., Hariharan, N., and Reddy, N. N., “A Technique for the Prediction of Propeller Induced Acoustic Loads on Aircraft Structures,” *AIAA-93-4340*, 15th AIAA Aeroacoustics Conference, Long Beach CA, 1993.
- Lockard D. P., and Casper H. J., “Permeable Surface Corrections for Ffowcs Williams and Hawkings Integrals” *NASA Langley Research Center, AIAA Paper 2005-2995*.
- Lu, Z. Liu, Y., Debiassi, M., Khoo, B. C. “Acoustic Characteristics of a Multi-Rotor MAV and its Noise Reduction Technology.” *INTER-NOISE*, 45th International Congress and Exposition of Noise Control Engineering, Hamburg, 2016.
- Lyrantzis, Anastasios S. 1994. “Review: The Use of Kirchhoff’s Method in Computational Aeroacoustics.” *Journal of Fluids Engineering* 116 (4). American Society of Mechanical Engineers: 665–76.
- Lyrantzis, Anastasios. 2003. “Surface Integral Methods in Computational aeroacoustics- From the (CFD) near-Field to the (Acoustic) Far-Field.” *International Journal of Aeroacoustics*, Multi Science Publishing: 95–128.
- Mankbadi, R. R. “Transition, Turbulence, and Noise,” *Kluwer Academic Press*, Boston, 1994 (republished by Springer).
- Mankbadi R., Lyrantzis AS., “Prediction of the far-field jet noise using Kirchhoff’s formulation. *AIAA J.* 34, 413–416. (doi:10.2514/3.13079)1996.
- Mankbadi, R. R., Shih, S. H., Hixon, R., Stuart, J. T., and Povinelli, L. A., “Extension of Near Field to Far Field Jet Noise Prediction,” *AIAA Paper 96-2651*, 32nd Joint Propulsion Conference, Lake Buena Vista, FL, July 1996
- Mankbadi, R. R., Hixon, D. R., Shih, S-H, and Povinelli, L. A., “Use of Linearized Euler Equations for Supersonic Jet Noise Prediction,” *AIAA Journal*, Vol. 36, No.2, February 1998, pp. 140-147, 1998.

- Mankbadi, R. R., Shih, S-H., Hixon, D. R., Stuart, J. T. and Povinelli, L. A., "A Surface-Integral Formulation for Jet Noise Prediction Based on The Pressure Signal Alone," *Journal of Computational Acoustics*, Vol. 6, No. 3, pp. 307-320, 1998.
- Mankbadi, R. R., Hixon, R., Povinelli, L. A., "Very Large Eddy Simulations of Jet Noise," *AIAA 2000-2008*, 6th AIAA/CEAS Aeroacoustics Conference, Lahaina, Hawaii, 12-14 June 2000.
- Mankbadi, R., Lo, S. C., Lyrintzis, A., Golubev, V., Dewan, Y., and Kurbatski, K., "Hybrid LES-RANS simulations of a jet impinging on a flat plate," *International Journal of Aeroacoustics*, vol. 15, pp. 535–553, 2016.
- Marte, J. E., and Kurtz, D. W., "A Review of Aerodynamic Noise from Propellers, Rotors, and Lift Fans." *NASA Technical Report 32-1462*, Jan 1970.
- Perez, A. M., and Lopez, O. D. "Computational Study of the Wake of a Quadcopter Propeller in Hover." 23rd *AIAA Aviation Forum* June 2017, DOI: 10.2514/6.2017-3961.
- Pierce, A. D., & Beyer, R. T. *Acoustics: An Introduction to Its Physical Principles and Applications*. 1989.
- Pilon, A. R., and Lyrintzis, A. S., "Integral Methods for Computational Aeroacoustics," *AIAA Paper 97-0020*, Jan 1997.
- Poinsot, T. J., and S. K. Lele, "Boundary Conditions for Direct Simulations of Compressible Viscous Reacting Flows," *Journal of Computational Physics*, vol. 101, pp. 104-129, 1992.
- Shih, S. H., Hixon, D. R., and Mankbadi, R. R., "A Zonal Approach for the Prediction of Jet Noise," *CEAS/AIAA paper 95-144* CEAS/AIAA Aeroacoustics Conference (16th AIAA Aeroacoustics Conference), Munich Germany, June 1995.
- Shih, S-H, Hixon, D. R., and Mankbadi, R. R., "Zonal Approach for Computational Aeroacoustics," *AIAA Journal of Propulsion and Power*, Vol. 13, No.6, pp. 745-758, 1997.
- Shur, M. L., Strelets, M. K., Travin, A. K., Spalart, P. R., "Turbulence Modeling in Rotating and Curved Channels: Assessing the Spalart-Shur Correction," *AIAA Journal* Vol. 38, No. 5, pp. 784-792, 2000.
- Simons, I. A., "Oscillatory Aerodynamic Loads on Helicopter Rotor Blades in Hover." *University of Southampton Internal Rept. ISVR*, Feb. 1966.
- Spalart, P. R., and Allmaras, S. R., "A One-Equation Turbulence Model for Aerodynamic Flows," *AIAA Paper 1992-0439*, Jan 1992.

- Spalart, P. R., Jou, W. H., Strelets, M., and Allmaras, S. R., “Comments on the feasibility of LES for Wings and on a Hybrid RANS/LES Approach,” *Advances in DNS/LES*, pp. 137–147, 1997.
- Spalart, P. R., Deck, S., Shur, M. L., Squires, K. D., Strelets, M. Kh., and Travin, A., “A New Version of Detached-Eddy Simulation, Resistant to Ambiguous Grid Densities,” *Theoretical Computational Fluid Dynamics Journal*, pp 181-195, 2006.
- Strawn, Roger C, and Rupak Biswas. 1995. “Computation of Helicopter Rotor Acoustics in Forward Flight.” *Journal of the American Helicopter Society*40 (3). AHS International: 66–72.
- Textron, B. H., “Air Taxi”, 2018. URL: <https://www.bellflight.com/company/innovation/air-taxi>.
- Uber, “Uber Elevate.” URL: <https://www.uber.com/elevate.pdf>
- UniCFDLab, “libAcoustics Library for Far-Field Noise Computation,” 2018. URL: <https://github.com/unicfdlab/libAcoustics>.
- Viswanathan, K., and Sankar, L. N., “Toward the Direct Calculation of Noise: Fluid/Acoustic Coupled Simulation,” *AIAA Journal*, Vol. 33, No. 12, pp. 2271-2279.
- Wagner, C., Huttl, T., and Sagaut, P., “Large-Eddy Simulations for Acoustics”, *Cambridge University Press*, New York, pp 219, 2007
- Wilcox, D. C. (2004). *Turbulence modeling for cfd*. DCW Industries.
- Williams, J E Ffowcs, and David L Hawkings. 1969. “Sound Generation by Turbulence and Surfaces in Arbitrary Motion.” *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences* 264 (1151). The Royal Society: 321–42.
- Yoon, S., and Diaz P. V., “High-Fidelity Computational Aerodynamics of Multi-Rotor Unmanned Aerial Vehicles” *AIAA SciTech*, AIAA 2018-1266, Jan 2018, DOI: 10.2514/6.2018-1266.
- Zawodny, N. S., Boyd, D. D., and Burley, C. L., “Acoustic Characterization and Prediction of Representative, Small-Scale Rotary-Wing Unmanned Aircraft System Components,” *72nd American Helicopter Society*, Paper #4, May 2016.
- Zhang, Q. and Yang, Y., “A New Simpler Rotation/Curvature Correction Method for Spalart-Allmaras Turbulence Model,” *Chinese Journal of Aeronautics*, Vol. 26, 2013, pp. 326-333.
- Zipline, “Lifesaving Deliveries by Drone,” URL: <https://flyzipline.com>

A. Modification of the Spalart-Allmaras Turbulence Model

The following is a code snippet of the modification to OpenFOAM's Spalart-Allmaras Turbulence Model to include the rotation/curvature correction. These codes are compatible only with version 4.1 of OpenFOAM.

SpalartAllmaras.H

```

1. /*-----*\
2.  =====
3.  \ \ / / F i e l d      | OpenFOAM: The Open Source CFD Toolbox
4.  \ \ / / O p e r a t i o n
5.  \ \ / / A n d          | Copyright (C) 2011-2016 OpenFOAM Foundation
6.  \ \ / / M a n i p u l a t i o n
7.  -----*\
8.  License
9.      This file is part of OpenFOAM.
10.
11.      OpenFOAM is free software: you can redistribute it and/or modify it
12.      under the terms of the GNU General Public License as published by
13.      the Free Software Foundation, either version 3 of the License, or
14.      (at your option) any later version.
15.
16.      OpenFOAM is distributed in the hope that it will be useful, but WITHOUT
17.      ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or
18.      FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License
19.      for more details.
20.
21.      You should have received a copy of the GNU General Public License
22.      along with OpenFOAM. If not, see <http://www.gnu.org/licenses/>.
23.
24.  Class
25.      Foam::RASModels::SpalartAllmaras
26.
27.  Group
28.      grpRASTurbulence
29.
30.  Description
31.      Spalart-Allmaras one-eqn mixing-length model for incompressible and
32.      compressible external flows.
33.
34.      Reference:
35.      \verbatim
36.          Spalart, P.R., & Allmaras, S.R. (1994).
37.          A one-equation turbulence model for aerodynamic flows.
38.          La Recherche Aeronautique, 1, 5-21.
39.      \endverbatim
40.
41.      The model is implemented without the trip-term and hence the ft2 term is
42.      not needed.
43.
44.      It is necessary to limit the Stilda generation term as the model generates
45.      unphysical results if this term becomes negative which occurs for complex
46.      flow. Several approaches have been proposed to limit Stilda but it is not

```

```

47.   clear which is the most appropriate. Here the limiter proposed by Spalart
48.   is implemented in which Stilda is clipped at Cs*Omega with the default value

49.   of Cs = 0.3.
50.
51.   The default model coefficients are
52.   \verbatim
53.       SpalartAllmarasCoeffs
54.       {
55.           Cb1      0.1355;
56.           Cb2      0.622;
57.           Cw2      0.3;
58.           Cw3      2.0;
59.           Cv1      7.1;
60.           Cs       0.3;
61.           sigmaNut 0.66666;
62.           kappa    0.41;
63.           Cr1      1.0;
64.           Cr2      12.0;
65.           Cr3      1.0;
66.       }
67.   \endverbatim
68.
69. SourceFiles
70.   SpalartAllmaras.C
71.
72. \*-----
73. */
74. #ifndef SpalartAllmaras_H
75. #define SpalartAllmaras_H
76.
77. #include "RASModel.H"
78. #include "eddyViscosity.H"
79.
80. // * * * * * //

81.
82. namespace Foam
83. {
84. namespace RASModels
85. {
86.
87. /*-----*\
88.           Class SpalartAllmaras Declaration
89. \*-----
90. */
91. template<class BasicTurbulenceModel>
92. class SpalartAllmaras
93. :
94.     public eddyViscosity<RASModel<BasicTurbulenceModel>>
95. {
96.     // Private Member Functions
97.
98.     // Disallow default bitwise copy construct and assignment
99.     SpalartAllmaras(const SpalartAllmaras&);
100.    void operator=(const SpalartAllmaras&);
101.
102.
103. protected:

```

```

104.
105.     // Protected data
106.
107.         // Model coefficients
108.
109.             dimensionedScalar sigmaNut_;
110.             dimensionedScalar kappa_;
111.
112.             dimensionedScalar Cb1_;
113.             dimensionedScalar Cb2_;
114.             dimensionedScalar Cw1_;
115.             dimensionedScalar Cw2_;
116.             dimensionedScalar Cw3_;
117.             dimensionedScalar Cv1_;
118.             dimensionedScalar Cs_;
119.
120.             dimensionedScalar Cr1_;
121.             dimensionedScalar Cr2_;
122.             dimensionedScalar Cr3_;
123.
124.         // Optional flag to activate the Rotation/Curvature Correction
125.         Switch RCMCorrection_;
126.
127.         // Fields
128.
129.             volScalarField nuTilda_;
130.             const volScalarField& y_;
131.
132.
133.         // Protected Member Functions
134.
135.             tmp<volScalarField> chi() const;
136.
137.             tmp<volScalarField> fv1(const volScalarField& chi) const;
138.
139.             tmp<volScalarField> fv2
140.             (
141.                 const volScalarField& chi,
142.                 const volScalarField& fv1
143.             ) const;
144.
145.             tmp<volScalarField> Stilda
146.             (
147.                 const volScalarField& chi,
148.                 const volScalarField& fv1
149.             ) const;
150.
151.             tmp<volScalarField> fw(const volScalarField& Stilda) const;
152.
153.         // Rotation/Correction Term declaration
154.         tmp<volScalarField> fr1(const volTensorField& gradU) const;
155.
156.         void correctNut(const volScalarField& fv1);
157.         virtual void correctNut();
158.
159.
160. public:
161.
162.     typedef typename BasicTurbulenceModel::alphaField alphaField;
163.     typedef typename BasicTurbulenceModel::rhoField rhoField;
164.     typedef typename BasicTurbulenceModel::transportModel transportModel;

```

```

165.
166.
167.     //- Runtime type information
168.     TypeName("SpalartAllmaras");
169.
170.
171.     // Constructors
172.
173.         //- Construct from components
174.         SpalartAllmaras
175.         (
176.             const alphaField& alpha,
177.             const rhoField& rho,
178.             const volVectorField& U,
179.             const surfaceScalarField& alphaRhoPhi,
180.             const surfaceScalarField& phi,
181.             const transportModel& transport,
182.             const word& propertiesName = turbulenceModel::propertiesName,
183.             const word& type = typeName
184.         );
185.
186.
187.     //- Destructor
188.     virtual ~SpalartAllmaras()
189.     {}
190.
191.
192.     // Member Functions
193.
194.         //- Read RASProperties dictionary
195.         virtual bool read();
196.
197.         //- Return the effective diffusivity for nuTilda
198.         tmp<volScalarField> DnuTildaEff() const;
199.
200.         //- Return the turbulence kinetic energy
201.         virtual tmp<volScalarField> k() const;
202.
203.         //- Return the turbulence kinetic energy dissipation rate
204.         virtual tmp<volScalarField> epsilon() const;
205.
206.         //- Solve the turbulence equations and correct the turbulence viscosity
207.
208.         virtual void correct();
209. };
210.
211. // * * * * *
212.
213. } // End namespace RASModels
214. } // End namespace Foam
215.
216. // * * * * *
217.
218. #ifdef NoRepository
219.     #include "SpalartAllmaras.C"
220. #endif
221.

```

```

222.// * * * * * //
223.
224.#endif
225.
226.// ***** //

```

SpalartAllmaras.C

```

1. /*-----*\
2.  ===== |
3.  \ \ / F i e l d | OpenFOAM: The Open Source CFD Toolbox
4.  \ \ / O p e r a t i o n |
5.  \ \ / A n d | Copyright (C) 2011-2016 OpenFOAM Foundation
6.  \ \ / M a n i p u l a t i o n |
7.  -----*\
8. License
9. This file is part of OpenFOAM.
10.
11. OpenFOAM is free software: you can redistribute it and/or modify it
12. under the terms of the GNU General Public License as published by
13. the Free Software Foundation, either version 3 of the License, or
14. (at your option) any later version.
15.
16. OpenFOAM is distributed in the hope that it will be useful, but WITHOUT
17. ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or
18. FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License
19. for more details.
20.
21. You should have received a copy of the GNU General Public License
22. along with OpenFOAM. If not, see <http://www.gnu.org/licenses/>.
23.
24. \*-----*\
    */
25.
26. #include "SpalartAllmaras.H"
27. #include "fvOptions.H"
28. #include "bound.H"
29. #include "wallDist.H"
30. #include "fvcDdt.H"
31. // * * * * * //
32.
33. namespace Foam
34. {
35. namespace RASModels
36. {
37.
38. // * * * * * Protected Member Functions * * * * * //
39.
40. template<class BasicTurbulenceModel>
41. tmp<volScalarField> SpalartAllmaras<BasicTurbulenceModel>::chi() const
42. {
43.     return nuTilda_/this->nu();
44. }

```

```

45.
46.
47. template<class BasicTurbulenceModel>
48. tmp<volScalarField> SpalartAllmaras<BasicTurbulenceModel>::fv1
49. (
50.     const volScalarField& chi
51. ) const
52. {
53.     const volScalarField chi3(pow3(chi));
54.     return chi3/(chi3 + pow3(Cv1_));
55. }
56.
57.
58. template<class BasicTurbulenceModel>
59. tmp<volScalarField> SpalartAllmaras<BasicTurbulenceModel>::fv2
60. (
61.     const volScalarField& chi,
62.     const volScalarField& fv1
63. ) const
64. {
65.     return 1.0 - chi/(1.0 + chi*fv1);
66. }
67.
68.
69. template<class BasicTurbulenceModel>
70. tmp<volScalarField> SpalartAllmaras<BasicTurbulenceModel>::Stilda
71. (
72.     const volScalarField& chi,
73.     const volScalarField& fv1
74. ) const
75. {
76.     volScalarField Omega(::sqrt(2.0)*mag(skew(fvc::grad(this->U_))));
77.
78.     return
79.     (
80.         max
81.         (
82.             Omega
83.             + fv2(chi, fv1)*nuTilda_/sqr(kappa_*y_),
84.             Cs_*Omega
85.         )
86.     );
87. }
88.
89.
90. template<class BasicTurbulenceModel>
91. tmp<volScalarField> SpalartAllmaras<BasicTurbulenceModel>::fw
92. (
93.     const volScalarField& Stilda
94. ) const
95. {
96.     volScalarField r
97.     (
98.         min
99.         (
100.            nuTilda_
101.            /(
102.                max
103.                (
104.                    Stilda,
105.                    dimensionedScalar("SMALL", Stilda.dimensions(), SMALL)

```

```

106.         )
107.         *sqr(kappa_*y_)
108.     ),
109.     scalar(10.0)
110. )
111. );
112. r.boundaryFieldRef() == 0.0;
113.
114. const volScalarField g(r + Cw2_*(pow6(r) - r));
115.
116. return g*pow((1.0 + pow6(Cw3_))/(pow6(g) + pow6(Cw3_)), 1.0/6.0);
117.}
118.
119.// Main code for rotation correction term fr1
120.template<class BasicTurbulenceModel>
121.tmp<volScalarField> SpalartAllmaras<BasicTurbulenceModel>::fr1(const volTensorFi
    eld& gradU) const
122.{
123.    if (RCMCorrection_)
124.    {
125.
126.        const volVectorField& U = this->U_;
127.        tmp<volTensorField> tgradU = fvc::grad(U);
128.
129.        volScalarField Omg1((2.0)*magSqr(skew(tgradU())));
130.        volScalarField Omg(pow(Omg1, 0.5));
131.
132.
133.        volScalarField OmgS
134.            (
135.                max
136.            (
137.                Omg,
138.                dimensionedScalar("Omg", dimensionSet(0, 0, -
139.                    1, 0, 0, 0, 0), 1e-16)
140.            ));
141.
142.        volScalarField S1((2.0)*magSqr(symm(tgradU())));
143.        volScalarField S(pow(S1, 0.5));
144.
145.        volScalarField SS
146.            (
147.                max
148.            (
149.                S,
150.                dimensionedScalar("S", dimensionSet(0, 0, -
151.                    1, 0, 0, 0, 0), 1e-16)
152.            ));
153.        volScalarField rStar(SS/OmgS);
154.        volScalarField rTilda((1-rStar)/sqr(rStar));
155.        return
156.            (1 + Cr1_)*2.0*rStar/(1.0 + rStar)*(1.0 - Cr3_*atan(Cr2_*rTilda))
157.            - Cr1_;
158.    }
159.    else
160.    {
161.        return tmp<volScalarField>
162.            (
163.                new volScalarField

```



```

164.         (
165.             IObject
166.             (
167.                 "fr1",
168.                 this->mesh_.time().timeName(),
169.                 this->mesh_,
170.                 IObject::NO_READ,
171.                 IObject::NO_WRITE
172.             ),
173.             this->mesh_,
174.             dimensionedScalar("fr1", dimless, 1),
175.             zeroGradientFvPatchScalarField::typeName
176.         )
177.     );
178. }
179.}
180.
181.
182.
183.
184. template<class BasicTurbulenceModel>
185. void SpalartAllmaras<BasicTurbulenceModel>::correctNut
186. (
187.     const volScalarField& fv1
188. )
189. {
190.     this->nut_ = nuTilda_*fv1;
191.     this->nut_.correctBoundaryConditions();
192.     fv::options::New(this->mesh_).correct(this->nut_);
193.
194.     BasicTurbulenceModel::correctNut();
195. }
196.
197.
198. template<class BasicTurbulenceModel>
199. void SpalartAllmaras<BasicTurbulenceModel>::correctNut()
200. {
201.     correctNut(fv1(this->chi()));
202. }
203.
204.
205. // * * * * * Constructors * * * * * //
206.
207. template<class BasicTurbulenceModel>
208. SpalartAllmaras<BasicTurbulenceModel>::SpalartAllmaras
209. (
210.     const alphaField& alpha,
211.     const rhoField& rho,
212.     const volVectorField& U,
213.     const surfaceScalarField& alphaRhoPhi,
214.     const surfaceScalarField& phi,
215.     const transportModel& transport,
216.     const word& propertiesName,
217.     const word& type
218. )
219. :
220.     eddyViscosity<RASModel<BasicTurbulenceModel>>
221.     (
222.         type,
223.         alpha,

```

```

224.     rho,
225.     U,
226.     alphaRhoPhi,
227.     phi,
228.     transport,
229.     propertiesName
230. ),
231.
232. sigmaNut_
233. (
234.     dimensioned<scalar>::lookupOrAddToDict
235.     (
236.         "sigmaNut",
237.         this->coeffDict_,
238.         0.66666
239.     )
240. ),
241. kappa_
242. (
243.     dimensioned<scalar>::lookupOrAddToDict
244.     (
245.         "kappa",
246.         this->coeffDict_,
247.         0.41
248.     )
249. ),
250.
251. Cb1_
252. (
253.     dimensioned<scalar>::lookupOrAddToDict
254.     (
255.         "Cb1",
256.         this->coeffDict_,
257.         0.1355
258.     )
259. ),
260. Cb2_
261. (
262.     dimensioned<scalar>::lookupOrAddToDict
263.     (
264.         "Cb2",
265.         this->coeffDict_,
266.         0.622
267.     )
268. ),
269. Cw1_(Cb1_/sqrt(kappa_) + (1.0 + Cb2_)/sigmaNut_),
270. Cw2_
271. (
272.     dimensioned<scalar>::lookupOrAddToDict
273.     (
274.         "Cw2",
275.         this->coeffDict_,
276.         0.3
277.     )
278. ),
279. Cw3_
280. (
281.     dimensioned<scalar>::lookupOrAddToDict
282.     (
283.         "Cw3",
284.         this->coeffDict_,

```

```

285.         2.0
286.     )
287. ),
288. Cv1_
289. (
290.     dimensioned<scalar>::lookupOrAddToDict
291.     (
292.         "Cv1",
293.         this->coeffDict_,
294.         7.1
295.     )
296. ),
297. Cs_
298. (
299.     dimensioned<scalar>::lookupOrAddToDict
300.     (
301.         "Cs",
302.         this->coeffDict_,
303.         0.3
304.     )
305. ),
306.
307. Cr1_
308. (
309.     dimensioned<scalar>::lookupOrAddToDict
310.     (
311.         "Cr1",
312.         this->coeffDict_,
313.         1.0
314.     )
315. ),
316. Cr2_
317. (
318.     dimensioned<scalar>::lookupOrAddToDict
319.     (
320.         "Cr2",
321.         this->coeffDict_,
322.         2.0
323.     )
324. ),
325. Cr3_
326. (
327.     dimensioned<scalar>::lookupOrAddToDict
328.     (
329.         "Cr3",
330.         this->coeffDict_,
331.         1.0
332.     )
333. ),
334. //Making a switching function for selection of the mode
335. RCMCorrection_
336. (
337.     Switch::lookupOrAddToDict
338.     (
339.         "RCMCorrection",
340.         this->coeffDict_,
341.         true
342.     )
343. ),
344.
345.

```

```

346.     nuTilda_
347.     (
348.         IObject
349.         (
350.             "nuTilda",
351.             this->runTime_.timeName(),
352.             this->mesh_,
353.             IObject::MUST_READ,
354.             IObject::AUTO_WRITE
355.         ),
356.         this->mesh_
357.     ),
358.
359.
360.     y_(wallDist::New(this->mesh_).y())
361. {
362.     if (type == typeName)
363.     {
364.         this->printCoeffs(type);
365.     }
366. // Adding information to the display
367.     if (RCMCorrection_)
368.     {
369.         Info<< "      Employing Modified Rotation/Curvature correction. " << endl;
370.     }
371.
372. }
373.
374.
375. // * * * * * Member Functions * * * * * //
376.
377. template<class BasicTurbulenceModel>
378. bool SpalartAllmaras<BasicTurbulenceModel>::read()
379. {
380.     if (eddyViscosity<RASModel<BasicTurbulenceModel>>::read())
381.     {
382.         sigmaNut_.readIfPresent(this->coeffDict());
383.         kappa_.readIfPresent(this->coeffDict());
384.
385.         Cb1_.readIfPresent(this->coeffDict());
386.         Cb2_.readIfPresent(this->coeffDict());
387.         Cw1_ = Cb1_/sqr(kappa_) + (1.0 + Cb2_)/sigmaNut_;
388.         Cw2_.readIfPresent(this->coeffDict());
389.         Cw3_.readIfPresent(this->coeffDict());
390.         Cv1_.readIfPresent(this->coeffDict());
391.         Cs_.readIfPresent(this->coeffDict());
392.         Cr1_.readIfPresent(this->coeffDict());
393.         Cr2_.readIfPresent(this->coeffDict());
394.         Cr3_.readIfPresent(this->coeffDict());
395.
396.
397.         RCMCorrection_.readIfPresent
398.         (
399.             "RCMCorrection", this->coeffDict()
400.         );
401.
402.         return true;
403.     }
404.     else

```

```

405.     {
406.         return false;
407.     }
408. }
409.
410.
411. template<class BasicTurbulenceModel>
412. tmp<volScalarField> SpalartAllmaras<BasicTurbulenceModel>::DnuTildaEff() const
413. {
414.     return tmp<volScalarField>
415.     (
416.         new volScalarField("DnuTildaEff", (nuTilda_ + this->nu())/sigmaNut_)
417.     );
418. }
419.
420.
421. template<class BasicTurbulenceModel>
422. tmp<volScalarField> SpalartAllmaras<BasicTurbulenceModel>::k() const
423. {
424.     return tmp<volScalarField>
425.     (
426.         new volScalarField
427.         (
428.             IOobject
429.             (
430.                 "k",
431.                 this->runTime_.timeName(),
432.                 this->mesh_
433.             ),
434.             this->mesh_,
435.             dimensionedScalar("0", dimensionSet(0, 2, -2, 0, 0), 0)
436.         )
437.     );
438. }
439.
440.
441. template<class BasicTurbulenceModel>
442. tmp<volScalarField> SpalartAllmaras<BasicTurbulenceModel>::epsilon() const
443. {
444.     WarningInFunction
445.     << "Turbulence kinetic energy dissipation rate not defined for "
446.     << "Spalart-Allmaras model. Returning zero field"
447.     << endl;
448.
449.     return tmp<volScalarField>
450.     (
451.         new volScalarField
452.         (
453.             IOobject
454.             (
455.                 "epsilon",
456.                 this->runTime_.timeName(),
457.                 this->mesh_
458.             ),
459.             this->mesh_,
460.             dimensionedScalar("0", dimensionSet(0, 2, -3, 0, 0), 0)
461.         )
462.     );
463. }
464.
465.

```

