

University of Windsor

Scholarship at UWindor

Electronic Theses and Dissertations

Theses, Dissertations, and Major Papers

2016

A Novel Numerical Approach for Generation and Propagation of Rotor-Stator Interaction Noise

Krishna Rajendra Patel
University of Windsor

Follow this and additional works at: <https://scholar.uwindsor.ca/etd>

Recommended Citation

Patel, Krishna Rajendra, "A Novel Numerical Approach for Generation and Propagation of Rotor-Stator Interaction Noise" (2016). *Electronic Theses and Dissertations*. 5860.
<https://scholar.uwindsor.ca/etd/5860>

This online database contains the full-text of PhD dissertations and Masters' theses of University of Windsor students from 1954 forward. These documents are made available for personal study and research purposes only, in accordance with the Canadian Copyright Act and the Creative Commons license—CC BY-NC-ND (Attribution, Non-Commercial, No Derivative Works). Under this license, works must always be attributed to the copyright holder (original author), cannot be used for any commercial purposes, and may not be altered. Any other use would require the permission of the copyright holder. Students may inquire about withdrawing their dissertation and/or thesis from this database. For additional inquiries, please contact the repository administrator via email (scholarship@uwindsor.ca) or by telephone at 519-253-3000ext. 3208.

A Novel Numerical Approach for Generation and Propagation of Rotor-Stator Interaction Noise

by

Krishna Patel

A Thesis

Submitted to the Faculty of Graduate Studies
through the Department of Mechanical, Automotive & Materials Engineering
in Partial Fulfillment of the Requirements for
the Degree of Master of Applied Science at the
University of Windsor

Windsor, Ontario, Canada

2016

© 2016 Krishna Patel

A Novel Numerical Approach for Generation and Propagation of Rotor-Stator Interaction Noise

by

Krishna Patel

APPROVED BY:

Narayan Kar
Electrical & Computer Engineering

Gary Rankin
Mechanical, Automotive & Materials Engineering

Colin Novak, Advisor
Mechanical, Automotive & Materials Engineering

Jeff Defoe, Advisor
Mechanical, Automotive & Materials Engineering

September 16, 2016

Declaration of Previous Publication

This thesis includes 1 original paper that has been previously published/submitted for publication in conference proceedings, as follows:

Thesis Chapter	Publication title/full citation	Publication status
<i>Chapter 3</i>	A Novel Numerical Approach for Generation and Propagation of Rotor-Stator Interaction Noise. Patel, K., Novak, C., Defoe, J. <i>Presented at the 22nd AIAA/CEAS Aeroacoustics Conference</i> , AIAA Paper 2016-2821, 2016.	Published

I certify that I have retained copyright to the above paper. I certify that the above material describes work completed during my registration as graduate student at the University of Windsor.

I declare that, to the best of my knowledge, my thesis does not infringe upon anyone's copyright nor violate any proprietary rights and that any ideas, techniques, quotations, or any other material from the work of other people included in my thesis, published or otherwise, are fully acknowledged in accordance with the standard referencing practices. Furthermore, to the extent that I have included copyrighted material that surpasses the bounds of fair dealing within the meaning of the Canada Copyright Act, I certify that I have obtained a written permission from the copyright owner(s) to include such material(s) in my thesis.

I declare that this is a true copy of my thesis, including any final revisions, as approved by my thesis committee and the Graduate Studies office, and that this thesis has not been submitted for a higher degree to any other University or Institution.

Abstract

As turbofan engine designs move towards bypass ratios ≥ 12 and corresponding low pressure ratios, fan rotor blade tip Mach numbers are reduced, leading to rotor-stator interaction becoming an important contributor to tonal fan noise. For future aircraft configurations employing boundary layer ingestion, non-uniform flow enters the fan. The impact of such non-uniform flows on the generation and propagation of rotor-stator interaction tones has yet to be assessed. In this thesis, a novel approach is proposed to numerically predict the generation and propagation of rotor-stator interaction noise with distorted inflow. The approach enables a 42% reduction in computational cost compared to traditional approaches employing a sliding interface between the rotor and stator. Such an interface may distort rotor wakes and can cause non-physical acoustic wave reflections if time steps are not sufficiently small. Computational costs are reduced by modelling the rotor using distributed, volumetric body forces. This eliminates the need for a sliding interface and thus allows a larger time step size. The force model responds to local flow conditions and thus can capture the effects of long-wavelength flow distortions. Since interaction noise is generated by the incidence of the rotor wakes onto the stator vanes, the key challenge is to produce the wakes using a body force field since the rotor blades are not directly modelled. It is shown that such an approach can produce wakes by concentrating the viscous forces along streamtubes in the last 15% chord. The new approach to rotor wake generation is assessed on the GE R4 fan from NASA's Source Diagnostic Test, for which the computed overall aerodynamic performance matches the experiment to within 1%. The rotor blade wakes are generated with widths in excellent agreement and depths in fair agreement with the experiment. An assessment of modal sound power levels computed in the exhaust duct indicates that this approach can be used for predicting downstream propagating interaction noise.

Acknowledgments

There are many people who I must thank that made this work possible. I would like to first thank each and every member of my thesis committee for all of their efforts and accepting a tight timeline for the timely completion of my thesis defense. Dr. Narayan Kar and Dr. Gary Rankin, I thank you both for your time and invaluable feedback. To my co-supervisors, Dr. Colin Novak and Dr. Jeff Defoe, I cannot begin to explain my appreciation to the both of you for your patience with me and ensuring that I continue to expand my knowledge and understanding in the vast field of aeroacoustics.

Insightful feedback provided by Dr. Claudio Lettieri of TU Delft cannot be forgotten and is greatly appreciated.

Thank you to those members of the Computational Fluid Dynamics Laboratory that helped me through the great complexities of CFD. I am also grateful to the remaining members of the Turbomachinery and Unsteady Flows Research Group for the great times we've shared.

This work was funded by NSERC Discovery Grants program. Computational resources were provided by the facilities of the Shared Hierarchical Academic Research Computing Network (SHARCNET:www.sharcnet.ca) and Compute/Calcul Canada.

Last but not least, to my family and friends, I would like extend my sincerest gratitude for motivating me to pursue a graduate degree and for being by my side the entire way.

Without those mentioned here, my success would not have been possible. Thank you all.

Contents

Declaration of Previous Publication	iii
Abstract	iv
Acknowledgments	v
List of Figures	ix
List of Tables	xii
Nomenclature	xiii
1 Introduction	1
1.1 Objective	2
1.2 Challenges	2
1.3 Major Findings and Conclusions	4
1.4 Scope of Thesis	4
2 Literature Review	6
2.1 The Nature of Interaction Noise in Uniform Inflow	6
2.2 Current Numerical Prediction Capability for Interaction Noise	8
2.3 Body Force Fan Model	11
2.4 Acoustics in Uniform and Non-Uniform Flow	15

2.5	Acoustic Buffer Zones	20
2.6	Acoustic Propagation in Non-Uniform Flows	21
2.7	Overview of Current State-of-the-Art	23
3	Approach	24
3.1	Experimental Acoustic Results	25
3.2	Experimental Aerodynamic Results	25
3.3	Computational Setup	28
3.4	Body Force Fan Model Innovations	29
3.4.1	Rotor-Alone Numerical Details	29
3.4.2	Parallel Force Model Updates	31
3.4.3	Camber Line Alteration	35
3.4.4	Rotor Wake Generation	40
3.5	Stator Geometry Development	42
3.6	Aeroacoustic Numerical Details	47
3.6.1	Computational Domain Splitting Method	49
3.6.2	Split Domain Method Validation	52
4	Uniform Inflow Assessment	55
4.1	Inlet Flow Computation	55
4.2	Source Generation Computation	56
4.3	Future Non-Uniform Inflow Computational Details	66
5	Conclusions and Future Work	69
5.1	Summary	69
5.2	Key Outcomes and Conclusions	71
5.3	Future Work	71
	Bibliography	73

A	Body Force Model Fluent User-Defined Function	78
B	Modal Decomposition in Uniform Flow	102
B.1	κ Function in Matlab	102
B.2	κ Matrix Generation in Matlab	102
B.3	E_{mn} Function in Matlab	104
B.4	Sutliff's Modal Decomposition Approach in Matlab	105
	Vita Auctoris	110

List of Figures

1-1	Comparison of traditional engine configuration and BLI engine configuration.	1
2-1	Rotor-stator interaction pattern showing a negative mode (inspired by Tyler and Sofrin).	7
2-2	Flow over rotor in both the stationary and relative frame.	9
2-3	Wave attenuation by co-swirling streamwise vortex in the outer span (from Defoe and Spakovszky, used with permission).	22
2-4	Reduction in sound power decay rate due to counter-swirling, lifted off streamwise vortex (from Defoe and Spakovszky, used with permission).	22
3-1	High-level approach flow chart. Green boxes: completed items. Dashed box: partially completed item. Red box: future work.	24
3-2	Experimental spanwise profiles of stagnation pressure ratio, stagnation temperature ratio, and adiabatic efficiency of the GE R4 rotor at takeoff (adapted from Hughes et al.).	27
3-3	Duct geometry and rotor/stator profiles.	28
3-4	Computational domain for rotor-alone computations.	30
3-5	Meridional view of rotor-alone grid.	31
3-6	Comparison of viscous model behavior with flight condition.	32
3-7	Velocity triangle at rotor trailing edge.	36

3-8	Comparison of blade camber line at 50% span.	39
3-9	Circumferentially-averaged stagnation temperature ratio profile at rotor trailing edge at takeoff.	39
3-10	Blade trailing edge viscous regions.	40
3-11	Non-dimensionalized axial velocity at 25%, 50%, and 75% span one chord downstream of the rotor trailing edge at takeoff.	42
3-12	Example of MISES generated grid showing two stator passages. . . .	44
3-13	MISES isentropic Mach number plot at the pitchline location.	46
3-14	Side view and perspective view of stator vane.	47
3-15	Computational domain for aeroacoustic computations.	48
3-16	Fan nacelle geometry showing planes used to define computational domains. The rotor and stator leading and trailing edge profiles are also shown.	50
3-17	Meridional view of the axisymmetric grid upstream of the stator and perspective view of a single passage grid around a stator vane from the source generation domain.	52
3-18	Split domain approach validation method.	53
4-1	Flow quantities one rotor chord upstream of the rotor leading edge. . .	56
4-2	Flow direction components one rotor chord upstream of the rotor leading edge.	56
4-3	Instantaneous contours of stagnation pressure non-dimensionalized by inlet stagnation pressure at 75% span after 1/4 rotor revolution. . . .	57
4-4	Time- and circumferentially-averaged axial Mach number downstream of the stator trailing edge.	58
4-5	Moving-average analysis of the acoustic modes of interest two stator chords downstream of the stator trailing edge.	59
4-6	Tone sound power level comparison at takeoff.	60

4-7	Modal sound power distribution $0.5c_{stator}$ downstream of stator trailing edge at takeoff.	61
4-8	Modal power distribution $1.0c_{stator}$ downstream of stator trailing edge at takeoff.	62
4-9	Modal power distribution $2.0c_{stator}$ downstream of stator trailing edge at takeoff.	63
4-10	Non-dimensionalized stagnation pressure profile for the non-uniform inflow computation (from Defoe).	67
4-11	Axial Mach number profile for the non-uniform inflow computation (from Defoe).	67
4-12	Radial Mach number profile for the non-uniform inflow computation (from Defoe).	68
4-13	Tangential Mach number profile for the non-uniform inflow computation (from Defoe).	68

List of Tables

3.1	Power levels of cut-on downstream propagating modes from the NASA SDT (from Heidelberg).	26
3.2	Overall rotor-alone and stage performance of the GE R4 fan at takeoff (from Hughes et al.).	26
3.3	Design parameters of the GE R4 rotor at takeoff (from Hughes et al.).	27
3.4	Overall mass-averaged rotor-alone performance comparison at takeoff.	38
3.5	Blade camber distribution coefficients for the R4 rotor.	38
3.6	Leading and trailing edge coefficients for the R4 rotor.	38
3.7	Summary of stator geometry specifications (from Hughes et al.). . . .	45
3.8	Sound power level at the acoustic measurement line.	54

Nomenclature

Symbols

a speed of sound

A area

B number of rotor blades

c blade chord

c_p specific heat at constant pressure of air

D diameter

E disturbance internal energy

f frequency

F body force per unit mass

h staggered spacing in blade row

H enthalpy

J Bessel function of the first kind

k wavenumber

K body force coefficient

l linear momentum per unit volume

m circumferential mode/order

M Mach number

n radial mode/order

N harmonic index

p pressure
 P viscous stress tensor
 q heat flux
 r radial coordinate
 R specific gas constant
 s entropy
 t time
 T temperature
 v velocity
 V number of stator vanes
 w relative velocity
 W disturbance energy flux
 x axial coordinate
 Y Bessel function of second kind
 α local blade camber angle
 γ ratio of specific heats
 δ Kronecker delta matrix
 ζ cut-off ratio
 η efficiency
 Ψ transverse eigenfunction of annular duct
 θ circumferential coordinate
 Θ azimuthal part of transverse eigenfunction of annular duct
 κ eigenvalue
 λ wavelength
 μ dynamic viscosity
 Ξ source term in Equation 2.33
 ϖ_p sound power

ρ density
 σ solidity
 ϕ hub-to-tip radius ratio
 ω loss coefficient
 Ω fan rotational speed

Subscripts

0 time-averaged quantity
 i, j vector/matrix component (tensor notation)
 LE leading edge
 t stagnation quantity
 TE trailing edge

Superscripts

' fluctuating portion of a quantity
 M mass-averaged quantity

Abbreviations

BLI Boundary Layer Ingestion/Ingesting
BPF Blade Passing Frequency
CFD Computational Fluid Dynamics
PPW Points Per Wavelength
RANS Reynolds-Averaged Navier-Stokes
SDT Source Diagnostic Test

Chapter 1

Introduction

Fan design is moving towards very high bypass ratios (e.g. 12:1 for the PW1500G [1]), with corresponding reduced pressure ratios and rotor blade tip Mach numbers to improve propulsive efficiency. These changes in engine design lead to different noise sources becoming important. Rotor shock noise is not a concern for fans with subsonic tip relative Mach numbers, yet tonal noise caused by rotor-stator interaction can still be significant. Analytical predictions of the cut-on acoustic modes are possible for axisymmetric flow, but the behavior of downstream propagating rotor-stator interaction modes in circumferentially non-uniform flow has not been studied extensively. Such flow fields arise in boundary-layer-ingesting (BLI) configurations such as those proposed for future commercial aircraft, e.g. the SAX-40 [2] and D8 [3]. An example of how BLI gives rise to non-uniform flow is illustrated in Figure 1-1.

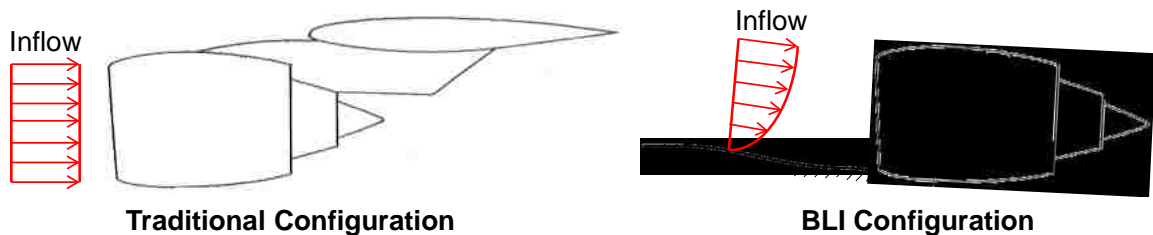


Figure 1-1: Comparison of traditional engine configuration and BLI engine configuration.

1.1 Objective

The objective of this thesis is to develop an innovative, computationally cost-effective approach to numerically model such rotor-stator interaction noise using direct computational aeroacoustics. The turbomachine of interest in this thesis is the fan rotor and stator from the NASA Source Diagnostics Test (SDT) [4, 5]. Though the purpose of developing this approach is to eventually investigate behaviour in circumferentially non-uniform flows, this thesis deals only with uniform flow and has the aim of assessing the capabilities of the approach. Future work will utilize this approach to investigate the effects of non-uniform inflow on the generation and propagation of rotor-stator interaction tones.

1.2 Challenges

Challenges arise when attempting to numerically model the generation and propagation of rotor-stator interaction noise in non-uniform flow. These challenges are:

1. the full wheel of the rotor and stator must be included in the computation,
2. a sliding interface between the rotor and stator is required,
3. non-physical wave reflections must be prevented at the flow inlet and outlet boundaries of the computational grid, and
4. it is not obvious how to determine what constitutes sound power in non-uniform flow.

The first point is a challenge because, for aeroacoustic computations, required grid resolutions are typically at least an order of magnitude higher than for purely aerodynamic computations. Therefore, the larger the region of interest, the more resources needed, since both computation run times and memory requirements scale with the

number of grid points. The second point is a challenge due to the small time step requirement for accurate acoustic wave propagation through the interface; this will be discussed in Section 2.2; therefore the challenge is again associated with the computational cost. The third challenge stems from the fact that the implementations of non-reflecting boundary conditions in commercial computational fluid dynamics (CFD) solvers are only effective at absorbing plane waves [6], while rotor-stator interaction noise gives rise to complex spinning modes. The final challenge is extracting the sound power spectra in non-uniform background flow.

To address the first and second challenges, the fan rotor in this work is modelled by a body-force-based blade row model. Body forces replace the rotor swept volume with a force field that achieves the same flow turning and pressure rise as the actual blade row. The force depends on the local flow conditions such that it responds appropriately to non-uniform inflow having wavelengths that are much larger than the blade pitch. Such models have been widely used to model turbomachinery performance. The use of this model lowers computational cost because (1) the grid count is reduced by avoiding the need for boundary layer grids on the blade surfaces and (2) a sliding interface between the rotor and stator is not needed, loosening the time step size requirements. By construction, body force blade row models typically yield a circumferentially-averaged flow field in uniform inlet flow. Rotor-stator interaction arises due to the incidence of discrete rotor blade wakes onto the stator vanes. Therefore, to generate interaction tones, the body force model must produce these wakes. In this thesis, the rotor wakes are produced via local concentrations of viscous forces at each blade trailing edge.

Non-physical wave reflections from the inlet and outlet boundaries are avoided using active damping of waves near the boundaries; the details will be introduced later in Section 2.5.

Finally, determining what constitutes sound power in non-uniform flow is not trivial as traditional modal decomposition techniques assume uniform background flow. A method to determine sound power spectra in non-uniform background flow must be implemented to assess the outcomes of the computations.

1.3 Major Findings and Conclusions

The key outcomes of this thesis are:

1. discrete rotor wakes are successfully demonstrated to be generated by body forces, and
2. the body-force generated rotor wakes give rise to the appropriate rotor-stator interaction modes.

The discrete rotor wakes generated by the body forces are found to have fair agreement with the experimental wake shapes. Due to the blade geometry not being available in the open literature, this level of agreement is accepted for this work. The appropriate rotor-stator interaction modes are predicted using this approach, however, their amplitudes are under-predicted. The rotor-locked mode at blade passing frequency (BPF) is under-predicted by 15 dB and the rotor-locked and interaction modes at 2BPF are under-predicted by 27 dB and 19 dB, respectively. Though the modes are under-predicted, this approach can be used to assess the impact of non-uniform flow on interaction noise as the changes in individual mode amplitudes can be assessed on a relative basis.

1.4 Scope of Thesis

The organization of the remainder of the thesis is as follows. Pertinent past literature is reviewed in Chapter 2. The approach taken in this work is discussed in detail

in Chapter 3. The results of the uniform inflow assessment are given in Chapter 4. Lastly, the conclusions and plans for future work are presented in Chapter 5.

Chapter 2

Literature Review

In this chapter, relevant background is presented to highlight the state of the art and identify gaps in the available literature.

2.1 The Nature of Interaction Noise in Uniform Inflow

The flow mechanisms responsible for the generation of interaction noise in uniform inflow were discussed by Tyler and Sofrin [7]. It was shown that rotor-stator interaction produces numerous modes, or rotating pressure patterns, corresponding to a single frequency. The different sources of interaction noise are (1) cutting of wakes of upstream stators by rotor blades, (2) impingement of rotating blade wakes on downstream stators, and (3) interruption of the rotating periodic pressure field of the rotor by the proximity of reflecting objects, apart from wake effects. For integer multiples of the BPF, the interaction modes present in a given configuration can be predicted using the following expression provided by Tyler and Sofrin,

$$m = NB + kV \tag{2.1}$$

where N is the harmonic index, B is the number of rotor blades, k is any integer value, and V is the number of stator vanes. A harmonic of a wave is a component frequency of the signal that is an integer multiple of the fundamental frequency. The interaction modes rotate at an angular velocity of $nB\Omega/m$, where Ω is the rotational speed of the rotor. Tyler and Sofrin's approach assumes uniform background flow and is not able to predict modal amplitudes.

In turbomachinery aeroacoustics, it is possible that modes with $m < 0$ exist. This is caused by the rotating rotor wakes impinging on different stator blades as shown in the example illustrated in Figure 2-1. This example consists of a 3 blade rotor (shown as blue lines) and 4 vane stator (shown as red lines). The dashed blue line allows the tracking of a single rotor blade. Using the expression by Tyler and Sofrin, with $k = -1$, a interaction mode $m = -1$ exists. As the rotor rotates clockwise, the generated single-lobed interaction pressure pattern rotates counter-clockwise. All negative modes rotate opposite the direction of the rotor.

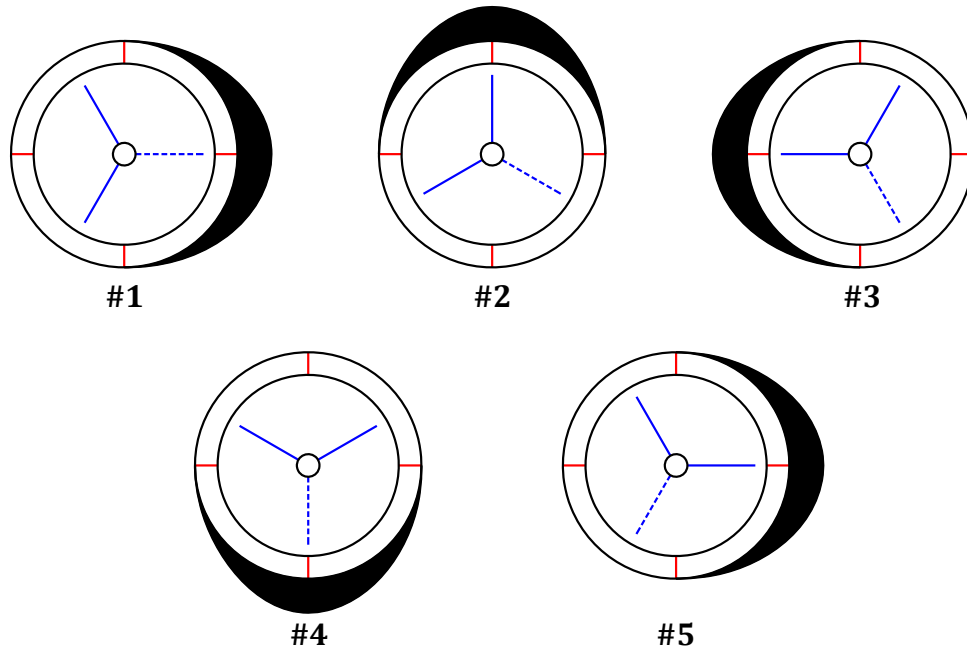


Figure 2-1: Rotor-stator interaction pattern showing a negative mode (inspired by Tyler and Sofrin [7]).

As part of the NASA Source Diagnostic Test (SDT), Heidelberg [8] experimentally compared the effects of different stator geometries on interaction noise. The different geometries included a 54-vane radial design, a 26-vane radial design, and a 26-vane swept design. Comparing the two radial designs, the general trend was that the 26-vane design generated higher noise levels for both upstream and downstream propagating tones at all flight conditions considered. The 26-vane swept design reduced the total tone sound power by 3-13 dB and individual interaction mode powers by over 20 dB. At higher power settings, the 26-vane swept design had lower tone power levels than the 54-vane radial design, however, at lower power settings, it was the other way around.

Lui et al. [9] studied the effects of stator lean on interaction noise and concluded that for the first BPF, stator vanes leaned in the direction of rotation at an angle greater than 10° reduced the interaction noise more than vanes leaned less than 10° . This was determined experimentally and it was also shown that the predominant component of the interaction noise is tonal.

This work will investigate the tonal component of rotor-stator interaction noise that is generated due to the impingement of the rotating blade wakes on downstream radial stators for the same fan studied by Heidelberg [8].

2.2 Current Numerical Prediction Capability for Interaction Noise

Computationally modeling the generation and propagation of rotor-stator interaction tones in non-uniform flow is challenging because the full wheel of the rotor and stator must be included in the numerical simulation. Figure 2-2 illustrates that while the flow is non-uniform in space in the non-rotating frame of reference, it becomes non-uniform in time (unsteady) in a frame of reference fixed to the rotor. Since there

is no frame of reference in which the mean flow is steady for the rotor, traditional approaches require a sliding interface between the rotor and stator for finite volume method CFD.

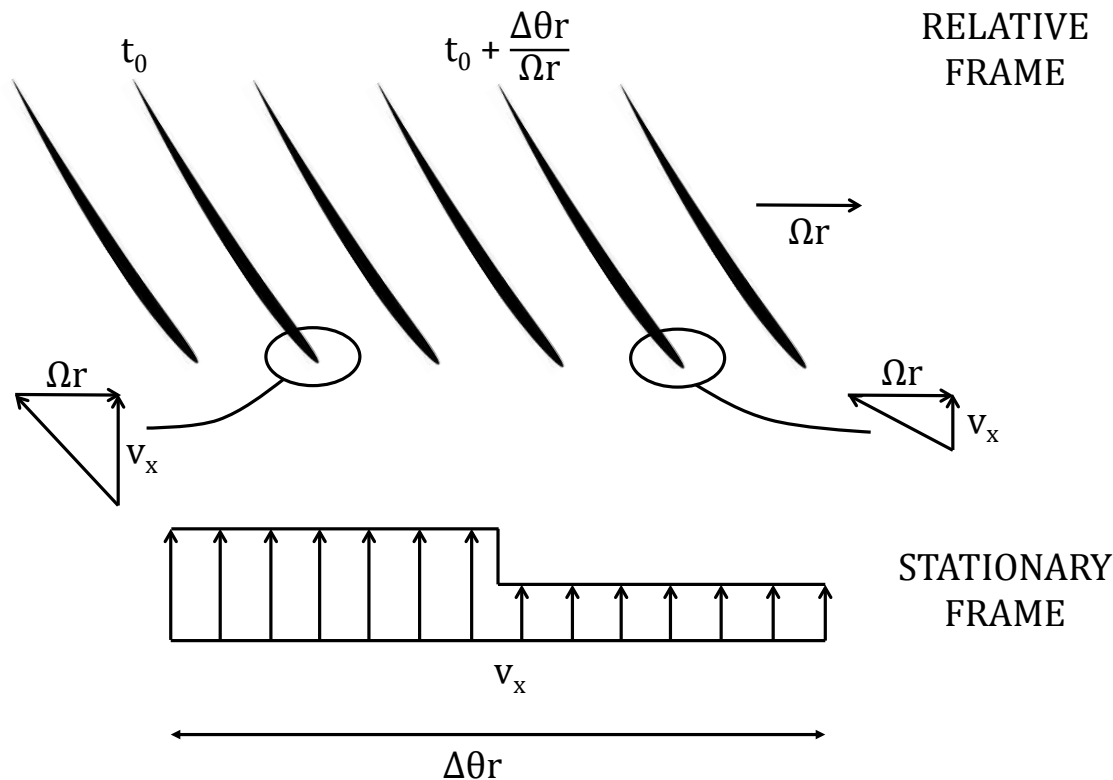


Figure 2-2: Flow over rotor in both the stationary and relative frame.

For accurate acoustic propagation through such an interface, the time step must be sufficiently small to minimize possible wake distortion and non-physical acoustic wave reflections. Using time step studies, Rumsey [10] provides a guideline to determine the maximum time step size for second-order time accurate Navier-Stokes codes as,

$$\Delta t_{max} = \frac{\theta_{period}}{40 |\Omega - \Omega_{mode}|} \quad (2.2)$$

where θ_{period} and Ω_{mode} are the angular period and the angular velocity of the acoustic mode of interest, respectively. Accurate propagation throughout the computational

grid, not just at the interfaces, must also be considered. For a second-order time accurate Navier-Stokes code, another study done by Rumsey et al. [11] demonstrated that a minimum of 60 time steps per period for the highest frequency of interest is required. Therefore, the maximum allowable time step is limited either by the interface or the grid zones, whichever requires the smaller time step.

In computational aeroacoustics, it is important to maintain sufficiently small grid cell sizes to minimize numerical dissipation. Chen et al. [12] showed that a spatial resolution of 25 grid points per wavelength (PPW) is necessary in the axial, circumferential, and radial directions for accurate propagation of acoustic waves in a second-order spatially accurate Navier-Stokes code. Though no dissipation rate was provided, Defoe [13] provided a best fit expression for wave decay for the inviscid, second-order, density-based ANSYS Fluent solver. This expression is,

$$Decay = 100 \cdot PPW^{-2.7} \frac{dB}{cell} \quad (2.3)$$

For 25 PPW, this expression predicts approximately a 0.2 dB/cell decay. The results predicted in this work are not corrected using this expression because this rate of decay is predicted for an inviscid case while this work uses a viscous model.

Numerical predictions of the exhaust tonal noise for the NASA SDT fan and duct were obtained by Sharma et al. [14, 15]. Two approaches were used. The first approach involved solving the Reynolds Averaged Navier-Stokes (RANS) equations to resolve the fan wake and consequently using linearized Euler equations to compute the acoustic response of the stator. This iterative method consisted of determining the inviscid stator mean flow by approximately matching the vane loading using the viscous solution. This resulted in an average under-prediction of the experimental sound power levels by about 5 dB. To shorten the computational process, a second approach was proposed which involved replacing the linearized Euler calculation with a linearized RANS calculation to eliminate the flow matching step. This improved

method simplifies the prediction process with no clear change in accuracy. To consider non-uniform inflow, this approach would have to be extended to full wheel, however, doing this would increase the computation cost dramatically. Rather than doing this, it would be more effective to simply use a traditional aeroacoustic approach.

2.3 Body Force Fan Model

Body-force-based fan models have been used to investigate various turbomachinery phenomena including rotor noise generation and propagation, as well as the impact of flow distortion on rotor performance. These models replace a blade row with a volumetric force field that generates the same stagnation pressure changes and flow turning within the swept volume of the blade row. In this section, the development of the model used as the starting point for the present work is discussed.

The use of a body force approach to represent the overall characteristics of a blade row was first introduced by Marble [16]. Marble developed the formulations that were required to generate the same axisymmetric effects on the flow as the blade row. This was done by producing a given change in swirl and entropy along a streamline.

Gong [17] developed an approach to obtain body forces corresponding to a given rotor that respond to local flow properties. The body forces were tested in non-uniform inflows for NASA stage 35 and it was shown that the loss coefficient and deviation across the span were well predicted compared to experimentally measured values. This showed that distortion transfer through the body force blade row is accurately captured. To model the effect of the blade row on the flow field, Gong's approach added source terms to the momentum and energy equations within the swept volume of the rotor and stator that are dependent on the local flow properties. The approach is based on a two-dimensional blade row model and assumes negligible radial force, which is a reasonable assumption for machines without strongly changing

annulus area. One major limitation to this approach is that it is unable to capture changes in the parallel (viscous) force due to changes in machine operating condition or flow coefficient. The full derivation of the formulation is given in [17] but a general overview of the final expressions are given here.

The source term added to the energy equation is,

$$\vec{F} \cdot \vec{v} = F_\theta \Omega r \quad (2.4)$$

where \vec{F} is the body force per unit mass, \vec{v} is the absolute velocity vector in the blade row, Ω is the rotational speed of the blade row, and r is the radial coordinate.

To accurately capture the effect of the frame of reference moving with the blade row, the force acting on the flow at a given location is split into components normal (\vec{F}_n) and parallel (\vec{F}_p) to the local relative flow direction. An additional term that acts in the normal direction is included to model the effects of the cross-passage pressure gradient in a staggered channel and is given by,

$$F_{n,\nabla p} = \frac{1}{\rho} \frac{\partial p}{\partial x} \sin \alpha \quad (2.5)$$

where α is the local blade camber angle relative to the meridional direction. The remainder of the normal force component is,

$$F_n = \frac{K_n(\alpha, x, r)}{h} (w_x \cos \alpha + w_\theta \sin \alpha) (w_\theta \cos \alpha - w_x \sin \alpha) \quad (2.6)$$

where h is the staggered spacing between the blades, and is given by,

$$h = \frac{2\pi r \sqrt{\sigma} \cos \alpha}{B} \quad (2.7)$$

σ is the blade solidity (chord/pitch) and B is the number of rotor blades. As mentioned above, there is no radial normal force. The normal force in component form

is:

$$F_{n,x} = F_n \frac{w_\theta}{w} \quad (2.8)$$

$$F_{n,\theta} = -F_n \frac{w_x}{w} \quad (2.9)$$

where x and θ are the axial and circumferential directions, respectively.

The parallel force component is due to viscous effects within the blade row and is given by,

$$F_p = -\frac{K_p(\alpha, x, r)}{h} w^2 \quad (2.10)$$

and has components,

$$F_{p,x} = F_p \frac{w_x}{w} \quad (2.11)$$

$$F_{p,r} = F_p \frac{w_r}{w} \quad (2.12)$$

$$F_{p,\theta} = F_p \frac{w_\theta}{w} \quad (2.13)$$

Note that the radial (r) component is non-zero for the viscous force.

To accurately model the specific performance characteristics of a given blade row, two body force coefficients, K_n and K_p , are introduced in the above equations and are determined empirically. K_n is a scaling function due to changes in deviation and K_p is related to the viscous effects. Gong determined these expressions using loss and deviation correlations by Lieblein [18] at the blade midspan and tip for a given blade geometry. The blade geometry must also be known because the number of blades B and the spanwise and chordwise distributions of α and σ are required.

With the individual components, the overall body force per unit mass is deter-

mined by summing the components:

$$F_x = F_{n,\nabla p} \cos \alpha + F_{n,x} + F_{p,x} \quad (2.14)$$

$$F_r = F_{p,r} \quad (2.15)$$

$$F_\theta = F_{n,\nabla p} \sin \alpha + F_{n,\theta} + F_{p,x\theta} \quad (2.16)$$

A transformation to Cartesian coordinates is needed for use of these forces in most computational codes:

$$\begin{bmatrix} F_x \\ F_y \\ F_z \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \theta & -\sin \theta \\ 0 & \sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} F_x \\ F_r \\ F_\theta \end{bmatrix} \quad (2.17)$$

Lastly, the force per unit mass is multiplied by the local fluid density ρ to obtain the force per unit volume.

Defoe [19, 20, 21] used Gong's approach, but determined K_n and K_p using single-passage, three-dimensional RANS calculations of the blade row being modelled. The expressions were determined using results at the midspan and tip locations. Defoe's study of upstream propagating rotor shock noise led to the extension of the body force approach to include shock generation. This was done by adding a circumferentially-varying force field to the axial body force rotating at the angular speed of the rotor with average value zero (to minimize changes to the overall performance). The other components of the body force formulation were unchanged; however, due to their dependence on local flow conditions, the actual forces adjust accordingly.

To use Gong's approach for different combinations of fan rotational speed and mass flow, K_p values corresponding to each configuration need to be determined a priori using RANS calculations to ensure the parallel force scales correctly. Peters [22]

corrected this by altering Gong’s parallel force formulation (Equation 2.10) to capture the increase in the blade losses at off-design operating conditions by introducing a quadratic dependence on the relative Mach number at the blade row inlet. The resulting improved parallel force (F_p) expression is given by,

$$F_p = -\frac{K_{p1}}{h} \left[\left(\overline{M_{rel}}^M \right)^2 + K_{p2} \left(\overline{M_{rel}}^M - \overline{M_{ref}}^M \right)^2 \right] w^2 \quad (2.18)$$

Here, K_{p1} and K_{p2} are body force coefficients specific to a given blade row and $\overline{M_{rel}}^M$ and $\overline{M_{ref}}^M$ are the mass-averaged relative Mach numbers at blade row inlet at a particular operating condition and at peak efficiency, respectively. $\overline{M_{rel}}^M$ is dependent on the flow coefficient and blade tip Mach number, the two metrics which define a particular flight condition. Furthermore, Peters altered Gong’s normal force formulation by introducing a radial force component to account for the effects of blade lean by rotating the normal force by the local blade lean angle. Changes made by Peters decreased the level of approximation for body force modelling for fans with significant lean and radial flow.

In this work, the body force model is adapted from Defoe [19, 20, 21] which is based on Gong’s blade passage model. To better suit the nature of this research, the improved parallel force model by Peters is implemented and rotor wakes are generated using a similar approach to Defoe’s rotor shock generation.

2.4 Acoustics in Uniform and Non-Uniform Flow

In a circular or annular duct, sound power is typically determined by assuming uniform flow. Candel and Poinot [23] provide the solution to the Helmholtz equation for the unsteady pressure field using this assumption as,

$$p'(x, r, \theta, t) = \Psi(r, \theta) e^{ik_{\parallel}x - i\omega t} \quad (2.19)$$

where the transverse eigenfunction $\Psi(r, \theta) = R(r)\Theta(\theta)$ and k_{\parallel} is the axial wavenumber. As seen by this description, the solution separates the pressure field into circumferential and radial components. The axial wavenumber determines whether a given acoustic mode is propagating (cut-on) in the axial direction, decaying exponentially (cut-off) in axial direction, or if the mode is a standing wave. A real axial wavenumber results in the cyclic variation of the exponential term resulting in a cut-on mode. An imaginary axial wavenumber results in a real exponent leading to a cut-off mode. Lastly, a zero axial wavenumber results in a standing wave.

For uniform flow, Sutliff [24] presented an approach that can be used to analyze the acoustic spectra on a given plane for which detailed derivations and background theory are provided in the paper. The high-level approach and final formulations are presented here. This method assumes a uniform Mach number in both the circumferential and radial direction in a constant area cylindrical or annular duct, therefore, it cannot be used for the non-uniform flow case. The first step in this approach is to take the Fourier transform in time and space of the unsteady pressure on the plane so that the unsteady pressure is a function of frequency (f), circumferential mode (m), and radial coordinate (r). To determine the radial mode orders (n), the eigenvalues (κ) of the cylindrical wave equation must first be determined. For hardwall boundary conditions, i.e. duct walls with no acoustic treatment, Sutliff provides the following equation that can be solved iteratively to obtain the values of κ for all (m, n) modes of interest,

$$\frac{mJ_m(\kappa_{mn}) - \kappa_{mn}J_{m+1}(\kappa_{mn})}{mY_m(\kappa_{mn}) - \kappa_{mn}Y_{m+1}(\kappa_{mn})} = \frac{\frac{m}{\phi}J_m(\kappa_{mn}\phi) - \kappa_{mn}J_{m+1}(\kappa_{mn}\phi)}{\frac{m}{\phi}Y_m(\kappa_{mn}\phi) - \kappa_{mn}Y_{m+1}(\kappa_{mn}\phi)} \quad (2.20)$$

where J_m and Y_m are the Bessel functions of the first and second kind of order m , respectively, and ϕ is the duct hub-to-tip radius ratio. Bessel functions are the

solutions to Bessel's differential equation,

$$z^2 \frac{d^2 y}{dz^2} + z \frac{dy}{dz} + (z^2 - m^2) y = 0 \quad (2.21)$$

where m is a real constant. J_m forms a fundamental set of solutions of order m [25],

$$J_m(z) = \left(\frac{z}{2}\right)^m \sum_{(k=0)}^{\infty} \frac{\left(\frac{-z^2}{4}\right)^k}{k! \Gamma(m+k+1)} \quad (2.22)$$

where $\Gamma(x)$ is the gamma function,

$$\Gamma(x) = \int_0^{\infty} e^{-t} t^{x-1} dt \quad (2.23)$$

Lastly, Y_m is a second solution of the Bessel's equation [25],

$$Y_m(z) = \frac{J_m(z) \cos(m\pi) - J_{-m}(z)}{\sin(m\pi)} \quad (2.24)$$

To determine if a given (m,n) mode propagates, Sutliff provides a cut-off ratio (ζ) given by,

$$\zeta = \frac{2\pi fr}{\kappa_{mn} a \sqrt{1 - (\overline{M}_x^M)^2}} \quad (2.25)$$

where a is the speed of sound and \overline{M}_x^M is the mass-averaged axial Mach number on the plane. $\zeta > 1$ indicates the mode is cut-on, $\zeta < 1$ indicates the mode is cut-off, and $\zeta = 0$ indicates it is a standing mode. The modal pressures are re-created from a least-squares-fit of the radial Bessel functions to the actual pressure profile. The least-squares analysis requires the curve to be fit to $n_{max} + 1$ radial modes, where n_{max} is the maximum number of radial modes being considered. For a given frequency, after applying the least-squares analysis, the modal amplitudes $[P_n]$ are determined using,

$$[P_n] = [F_{vn}]^{-1} [E_{vr}] [p_r] \quad (2.26)$$

where $[p_r]$ are the measured pressures,

$$[F_{vn}] = \begin{bmatrix} \sum_{i=1}^{N_p} E_{m0}^2(\kappa_{m0} r_i) & \cdots & \sum_{i=1}^{N_p} E_{m0}(\kappa_{m0} r_i) E_{mn_{max}}(\kappa_{mn_{max}} r_i) \\ \vdots & & \vdots \\ \sum_{i=1}^{N_p} E_{mn_{max}}(\kappa_{mn_{max}} r_i) E_{m0}(\kappa_{m0} r_i) & \cdots & \sum_{i=1}^{N_p} E_{mn_{max}}^2(\kappa_{mn_{max}} r_i) \end{bmatrix} \quad (2.27)$$

and,

$$[E_{vr}] = \begin{bmatrix} E_{m0}(\kappa_{m0} r_1) & \cdots & E_{m0}(\kappa_{m0} r_{max}) \\ \vdots & & \vdots \\ E_{mn_{max}}(\kappa_{mn_{max}} r_1) & \cdots & E_{mn_{max}}(\kappa_{mn_{max}} r_{max}) \end{bmatrix} \quad (2.28)$$

Here, E is the duct profile function and is given by,

$$E_{mn}(\kappa_{mn} r) = C_{mn} [J_m(\kappa_{mn} r) + Q_{mn} Y_m(\kappa_{mn} r)] \quad (2.29)$$

where C is a normalizing coefficient and Q is the weighting function:

$$\frac{1}{C_{mn}^2} = \frac{1}{2} \left\{ \left[1 - \frac{m^2}{\kappa_{mn}^2} \right] [J_m(\kappa_{mn}) + Q_{mn} Y_m(\kappa_{mn})]^2 - \left[\phi^2 - \frac{m^2}{\kappa_{mn}^2} \right] [J_m(\kappa_{mn} \phi) + Q_{mn} Y_m(\kappa_{mn} \phi)]^2 \right\} \quad (2.30)$$

$$Q_{mn} = - \frac{\left[\left(\frac{m}{\kappa_{mn}} \right) J_m(\kappa_{mn}) - J_{m+1}(\kappa_{mn}) \right]}{\left[\left(\frac{m}{\kappa_{mn}} \right) Y_m(\kappa_{mn}) - Y_{m+1}(\kappa_{mn}) \right]} \quad (2.31)$$

Once the modal pressure amplitudes have been determined, the modal power

amplitude (ϖ_p) for each (m,n) mode is,

$$\varpi_p = \mp \frac{\pi r^2 (1 - \phi^2)}{\rho a} \left| \sqrt{\left(1 - (\overline{M_x^M})^2\right)} \right|^4 \Re \left\{ \frac{\sqrt{1 - \frac{1}{\zeta^2}}}{\left|1 \pm (\overline{M_x^M}) \sqrt{1 - \frac{1}{\zeta^2}}\right|^2} \right\} |P_n|^2 \quad (2.32)$$

Myers [26] developed a formulation to determine the disturbance energy intensity field in non-uniform flow from the unsteady flow field. Myers' approach is based on work by Morfey [27], for which the governing equations take the form,

$$\frac{\partial E}{\partial t} + \frac{\partial W_i}{\partial x_i} = \Xi \quad (2.33)$$

Here, E is the acoustic energy, W_i is the acoustic energy flux, and Ξ is a source term. Myers' approach does not require either linearizing the equations of motion or the decomposition of the flow field; therefore, the generalized disturbance energy is given by,

$$E = \rho [H_t - H_{t,0} - T_0 (s - s_0)] - l_o \cdot (v - v_0) - (p - p_0) \quad (2.34)$$

where H_t is the total enthalpy and p , T , s , ρ , and v are the pressure, temperature, entropy, density, and velocity, respectively. The generalized disturbance energy flux is given by,

$$\begin{aligned} W_i = & (l_i - l_{i0}) [H_t - H_{t,0} - T_0 (s - s_0)] + l_{i0} (T - T_0) (s - s_0) \\ & - (l_j - l_{j0}) \left(\frac{P_{ij}}{\rho} - \frac{P_{ij0}}{\rho_0} \right) + (T - T_0) \left(\frac{q_i}{T} - \frac{q_{i0}}{T_0} \right) \end{aligned} \quad (2.35)$$

where $l_i = \rho v_i$ is the linear momentum per unit volume. The subscript 0 represents the time-averaged value of a quantity. P_{ij} is the viscous stress tensor and is given by,

$$P_{ij} = -\frac{2}{3}\mu \left(\frac{\partial v_k}{\partial x_k} \right) \delta_{ij} + \mu \left(\frac{\partial v_j}{\partial x_i} + \frac{\partial v_i}{\partial x_j} \right) \quad (2.36)$$

where μ is the dynamic viscosity and δ_{ij} is a Kronecker delta matrix. q_i is the heat flux vector which is zero when no heat is added to the system. Once the disturbance energy intensity field is determined, a Fourier transform can be performed to obtain frequency spectra.

In this work, Sutliff's approach will be used for the uniform inflow case. In future work, Myers' approach is best used for non-uniform inflow cases. Myers' approach was also used by Defoe [13] while studying upstream propagating rotor shock noise with non-uniform inflows. These two acoustic analysis approaches can be compared using the uniform inflow case. Sutliff's approach is employed for the uniform flow case to allow for a direct comparison between the computed and experimentally-measured results.

2.5 Acoustic Buffer Zones

When propagating acoustic waves numerically, non-physical reflections of outward-traveling waves from the inflow and outflow boundaries must be avoided to prevent spurious acoustic results. To solve this problem, acoustic buffer zones such as those based on the work of Freund [28] can be implemented. These buffer zones are located adjacent to the inflow and outflow boundaries. In these zones, additional terms are added to the governing equations which actively damp the unsteady oscillations in the flow. The damping terms scale as $\sin^4\left(\frac{x^*}{L^*}\right)$, where x^* is the axial distance within a buffer zone which extends from $x^* = 0$ (where acoustic waves are incident) to $x^* = L^*$, where L^* is the axial length of the buffer zone. This results in waves being attenuated sufficiently such that reflections back into the computational region of interest are negligible.

Chen et al. [12] studied the performance of various sizes of buffer zones and concluded that for 33 PPW, the smallest buffer zone where no apparent reflections were observed was 5 grid points. In this work, with 25 PPW, a smaller buffer zone may be sufficient, however, 5 grid points are used to be conservative.

An assessment of the combination of a buffer zone and a grid stretching region performed by Defoe [13] showed that transmitted wave amplitudes were reduced by approximately 60 dB. Though the grid stretching region is not included in this work, similar performance of the buffer zone is expected due to the reflected waves having to travel back through the buffer zone and being attenuated a second time.

2.6 Acoustic Propagation in Non-Uniform Flows

The effects of swirl distortion on in-duct acoustic propagation needs to be considered when investigating spatially non-uniform flow. Defoe and Spakovszky [21] studied these effects. BLI at low flight speed, where flow accelerates into an aircraft's engines, can cause the formation of a horseshoe vortex due the interaction of the boundary-layer vorticity and the inlet lip. The two vortices, counter-swirling (rotating in the opposite direction as the fan) and co-swirling (rotating in the same direction as the fan), affect the wave propagation depending on their location relative to the duct wall.

If these vortices are located near the wall, the co-swirling vortex creates a region that decreases the local relative Mach number which causes a decay in wave amplitude as shown in Figure 2-3. This flow mechanism affects wavelengths on the order of the extent of the subsonic relative flow. If these vortices have lifted off the duct wall, the counter-swirling vortex creates a region that increases the local relative Mach number which causes a reduced attenuation rate as can be seen in Figure 2-4. This flow mechanism affects waves with short wavelengths (higher frequencies).

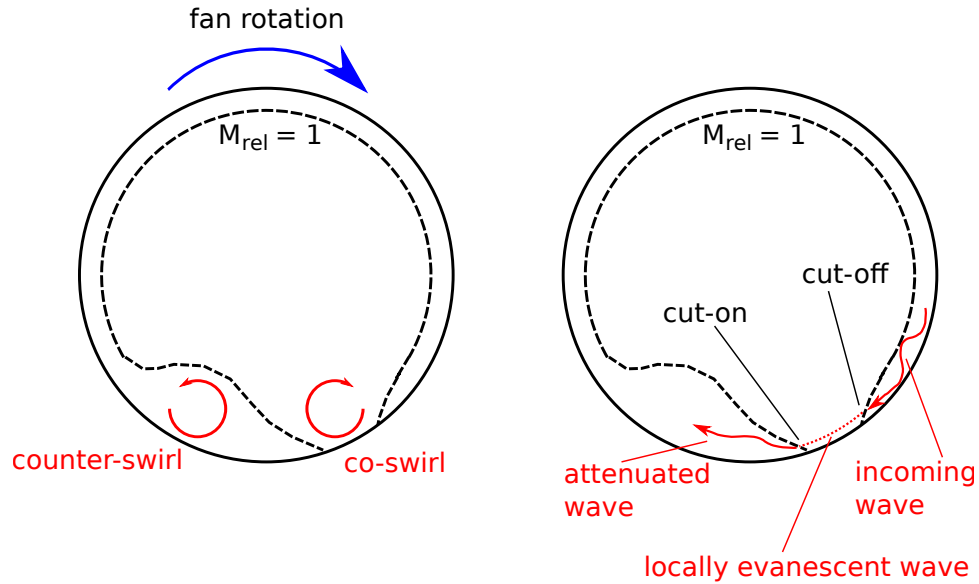


Figure 2-3: Wave attenuation by co-swirling streamwise vortex in the outer span (from Defoe and Spakovszky [21], used with permission).

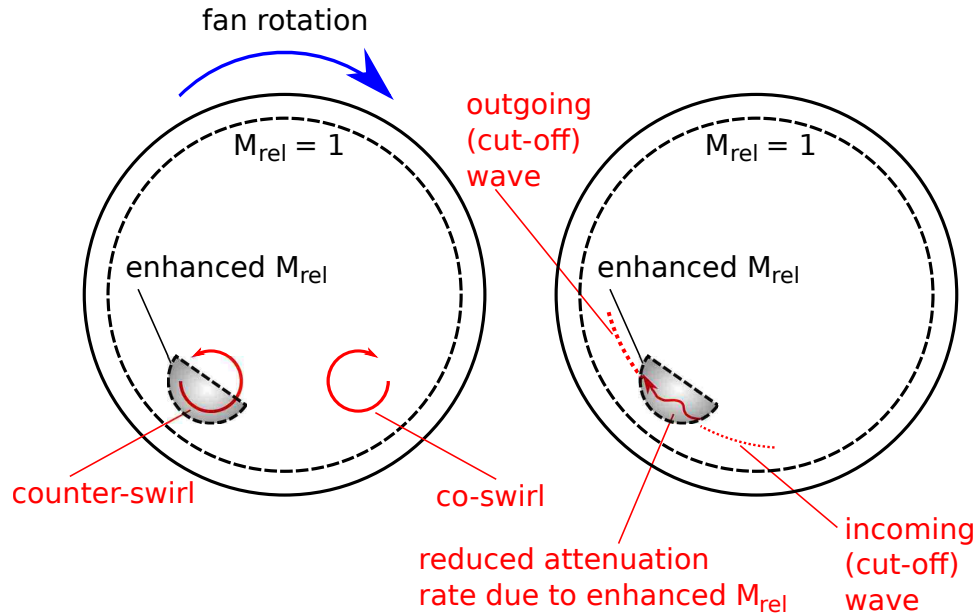


Figure 2-4: Reduction in sound power decay rate due to counter-swirling, lifted off streamwise vortex (from Defoe and Spakovszky [21], used with permission).

Si [29] investigated the effects of four different non-uniform mean flow fields on the propagation of acoustic waves. This study was performed numerically using two different acoustic propagation models; linearized Euler equations (LEE) and acoustic perturbation equations (APE). Results indicate that non-uniform mean flows modify

both the amplitude and shape of the acoustic waves comparing to that of the uniform mean flow. The two numerical models compared yielded similar results with some differences in amplitude depending on the location.

In planned future work that will consider non-uniform inflow, the acoustic results and resultant flow field will reveal the effects of inlet distortions on in-duct propagation of downstream-travelling rotor-stator interaction noise.

2.7 Overview of Current State-of-the-Art

The current state-of-the-art to numerically predict rotor-stator interaction noise involves including both rotor and stator geometry in the computations. The drawbacks of this approach are that a boundary layer grid is required for both blade rows, sufficient grid resolution is needed to accurately resolve the rotor wakes, and a sliding interface is required between the rotor and stator. As modelling non-uniform inflow requires the full annulus, this causes this approach to be very computationally expensive. Though it is possible to use this approach for non-uniform inflow, the available literature shows that it has not been attempted.

This work adds to the state-of-the-art by developing a computationally-cost effective approach to model rotor-stator interaction noise in both uniform and non-uniform flow. This approach reduces computational cost by replacing the rotor blade row with a body force field which eliminates the fine boundary layer grid (for the rotor) as well as the sliding interface. If insufficient computational resources are available to run the resulting simulations, a method to split the computation into smaller regions while retaining accurate acoustic propagation is also developed.

Chapter 3

Approach

In this chapter, the details of the approach used to computationally predict the generation and propagation of rotor-stator interaction tones are described in detail. The high-level approach is shown in Figure 3-1. First, the machine for which the approach is assessed and the experimental results from the literature are presented. The details of the computational modelling follow.

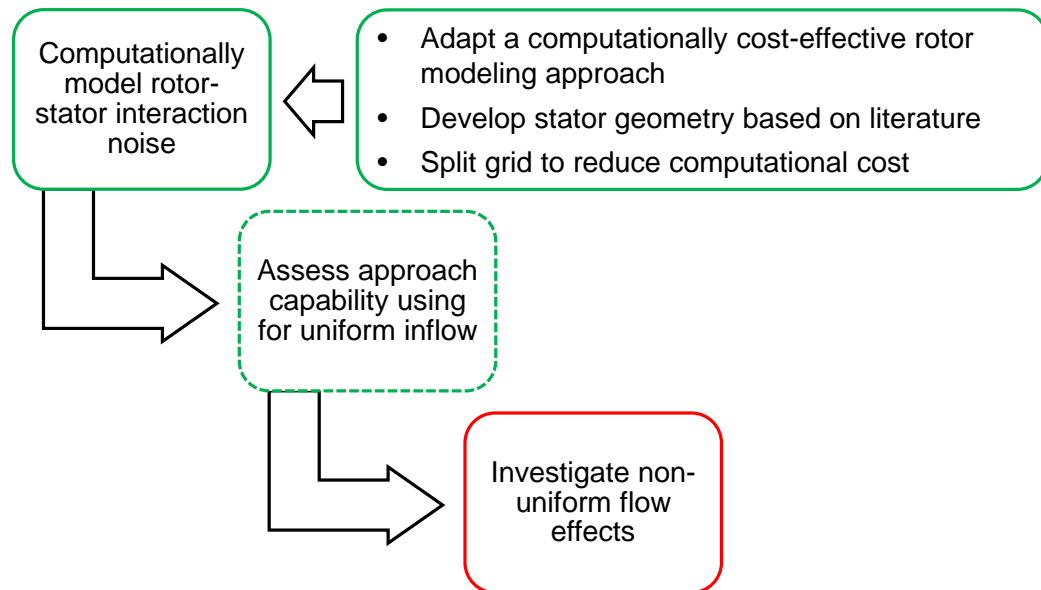


Figure 3-1: High-level approach flow chart. Green boxes: completed items. Dashed box: partially completed item. Red box: future work.

3.1 Experimental Acoustic Results

To assess the ability of the body-force-based approach to generate rotor wakes and tonal turbomachinery noise, the 22-blade GE R4 rotor and associated 54-vane radial (baseline) stator from the NASA SDT series are used. Extensive experimental data is available for both the aerodynamic [4, 5, 30] and acoustic performance [8] of this transonic fan in uniform flow. For this rotor/stator combination, experimental inlet (upstream propagating) and exhaust (downstream propagating) tone mode measurements were performed for three flight conditions: approach (61.7% corrected rotor speed), cutback (87.5% corrected rotor speed), and takeoff (100% corrected rotor speed). The focus in this thesis is on downstream propagating noise due to the overall sound power in the exhaust being higher than that in the inlet. An examination of the cut-on modes at each operating point, given in Table 3.1, indicates that takeoff is the most interesting for the current work. This is because at this condition both rotor-alone ($m = 44$) and a first interaction mode ($m = -10$) are cut-on at twice the BPF and the rotor-alone mode ($m = 22$) is cut-on at the BPF. Equation 2.1 by Tyler and Sofrin [7] can be used to show that the additional cut-on mode at $m = -10$ is an interaction mode (for $k = -1$ and $n = 2$).

3.2 Experimental Aerodynamic Results

All the pertinent fan details and experimental aerodynamic results are provided in this section. Overall rotor-alone and stage performance is given in Table 3.2. Fan design parameters from Hughes et al. [4] are listed in Table 3.3. The fan considered in this work has supersonic blade tip speeds and a high pressure ratio. The characteristics of this fan do not effect the rotor-stator interaction noise mechanisms and the approach developed in this work is not limited to fans of this type. The experimental spanwise profiles of the stagnation pressure ratio, stagnation temperature ratio, and adiabatic

efficiency for the GE R4 rotor are plotted in Figure 3-2. These profiles are included here as they are later used to assess the accuracy of the body force fan model. The reduction in efficiency near the tip is a result of shock losses stemming from supersonic blade tip speeds.

Table 3.1: Power levels of cut-on downstream propagating modes from the NASA SDT (from Heidelberg [8]).

Sound Power Level at Approach, 61.7% Speed (dB)		
Circumferential Mode	BPF	2BPF
$m = -10$	—	115

Sound Power Level at Cutback, 87.5% Speed (dB)		
Circumferential Mode	BPF	2BPF
$m = -10$	—	127

Sound Power Level at Takeoff, 100% Speed (dB)		
Circumferential Mode	BPF	2BPF
$m = -10$	—	131
$m = 22$	131	—
$m = 44$	—	133

Table 3.2: Overall rotor-alone and stage performance of the GE R4 fan at takeoff (from Hughes et al. [4, 5]).

	Stagnation Pressure Ratio		Stagnation Temperature Ratio		Adiabatic Efficiency	
	Rotor-alone	Stage	Rotor-alone	Stage	Rotor-alone	Stage
	Takeoff	1.488	1.470	1.130	1.128	0.924

Table 3.3: Design parameters of the GE R4 rotor at takeoff (from Hughes et al. [4]).

Number of Fan Blades	22
Fan Tip Diameter	0.559 m
Corrected Tip Speed	370.3 m/s
Corrected RPM	12,657
Corrected Fan Mass Flow	45.586 kg/s
Stage Pressure Ratio	1.47
Bypass Ratio	8.85

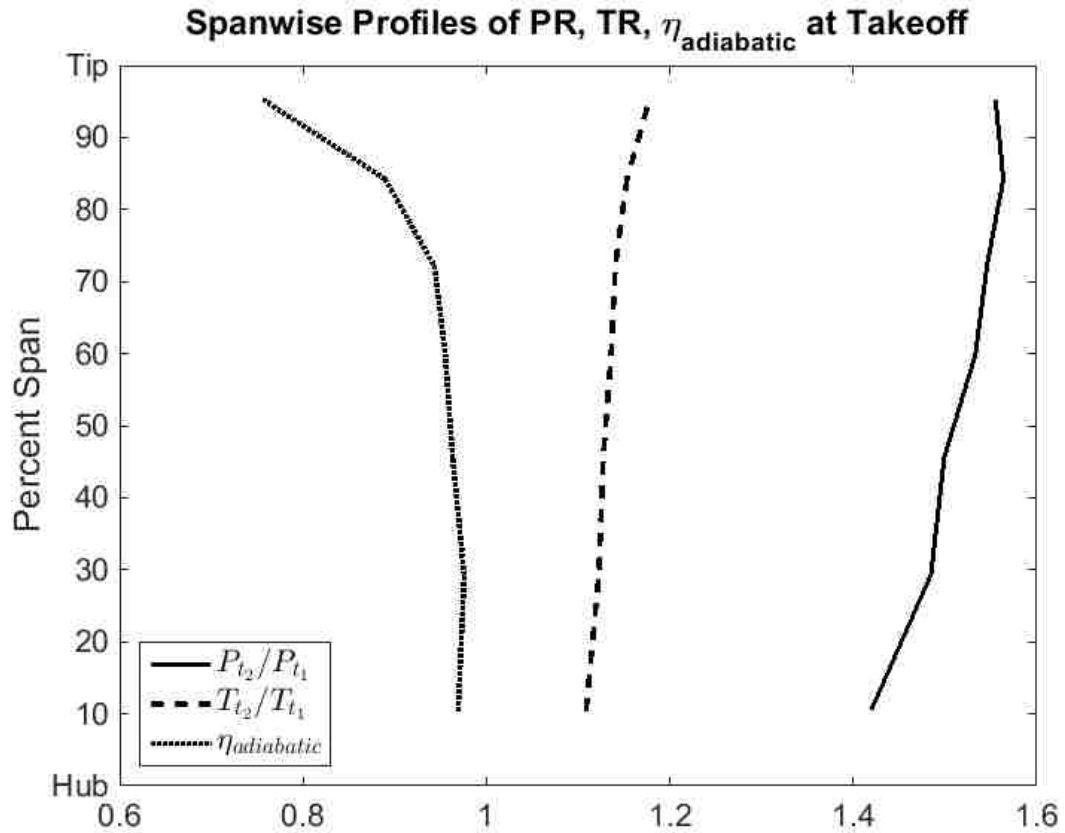


Figure 3-2: Experimental spanwise profiles of stagnation pressure ratio, stagnation temperature ratio, and adiabatic efficiency of the GE R4 rotor at takeoff (adapted from Hughes et al. [4]).

3.3 Computational Setup

The commercial CFD software ANSYS Fluent 15 [31] is used in this work. The code is second-order accurate in both space and time. Turbulence closure is achieved using the one-equation Spalart-Allmaras (SA) model [32]. A recent paper by Gunn and Hall [33] contains a review of past studies which indicate that non-linear distortion transfer is well-captured using the SA model, suggesting it is appropriate for the phenomena of interest here. In ANSYS Fluent 15, the SA model has been extended with a y^+ -insensitive wall treatment, however, a range of $1 < y^+ < 30$ is recommended to maintain the integrity of the model. In this work, a y^+ of ~ 10 is used to ensure the near wall grid resolution is within the recommended range.

The operating condition is achieved by setting a stagnation pressure and stagnation temperature at the inlet and varying the static pressure at the outlet to yield the takeoff corrected mass flow. The hub upstream of the rotor trailing edge rotates for consistency with the experimental configuration. Figure 3-3 illustrates the duct geometry as well as the rotor and stator profiles from Hughes et al. [5].

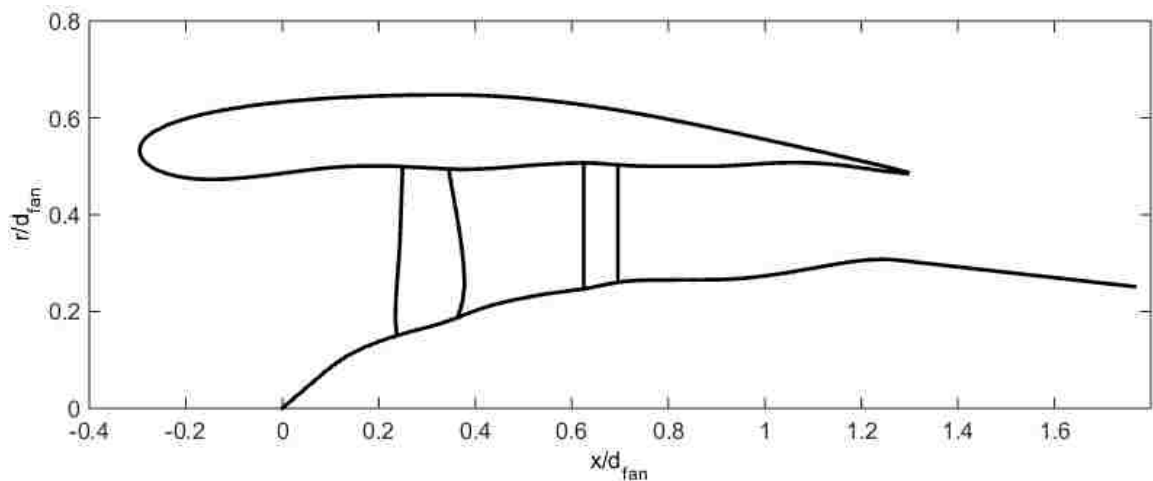


Figure 3-3: Duct geometry and rotor/stator profiles (from Hughes et al. [5]). Flow is from left to right.

3.4 Body Force Fan Model Innovations

Rather than developing a new model of the GE R4, the rotor body force model has been adapted from Defoe et al. [13, 19] for use in this work since the blade geometry data is not publicly available while the body force model is. The model was originally designed for the cutback flight condition and the pressure ratio achieved is accurate only for the corresponding flow coefficient and corrected speed. As Defoe’s model had been developed using Gong’s approach, the viscous model was set such that the losses were correct only for the cutback flight condition. This is determined by an assessment of the model at the takeoff flight condition.

To employ this model in the current work, changes are made to the body force model from Defoe et al. [13, 19] to correctly capture the adiabatic efficiency and work input of the GE R4 rotor at flight conditions other than cutback. The changes made to the model are:

1. the efficiency is corrected by modifying the viscous (loss-generating) force to capture typical turbomachinery loss bucket behavior using Peters’ approach [22],
2. the blade camber profiles are adjusted to obtain stagnation pressure and temperature ratios in agreement with the experimental data, and
3. rotor wakes need for the generation of interaction noise are produced via local concentrations of viscous forces at each blade trailing edge.

Each of these changes are discussed in detail in the remainder of this section.

3.4.1 Rotor-Alone Numerical Details

All of the body force innovations are verified using three-dimensional rotor-alone computations. The computational region of interest or domain for these calculations is

illustrated in Figure 3-4. The inlet of this calculation is located 2 rotor diameters upstream from the inlet throat (minimum outer radius) and the outlet is located 2 rotor diameters downstream of the stator leading edge. The duct past the stator leading edge is not included in the computations because in the rotor-alone configuration, the swirl imparted by the rotor causes the flow to choke prior to the duct outlet.

The grid size is set based on a grid convergence study performed by Defoe [13]. The grid consists of 60 radial cells and the axial cell length is set to ensure that cell aspect ratios in the meridional plane are all approximately 1, as can be seen in Figure 3-5. Since the body force computations with uniform inflow are axisymmetric, the circumferential extent of the grid is set to a single cell spanning one-eighth of a blade passage (approximately 2°). This keeps the computational costs to a minimum while ensuring accurate results. This grid is used to update the parallel force formulation as well as to adjust the blade camber profiles as will be described later in this Chapter.

Assessing the generation of rotor wakes requires a full blade passage in the computational domain. The axial and radial components of the grid remain the same, while the circumferential resolution is iterated upon to find the minimum required to resolve the wake. This yielded 13 cells per passage (about 1.25° per cell).

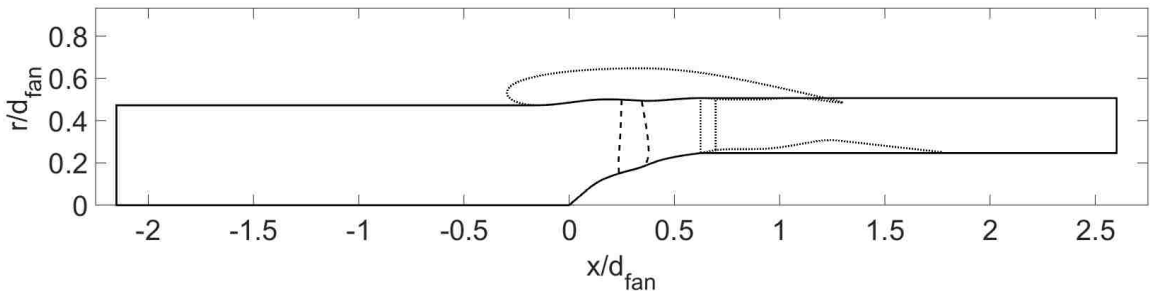


Figure 3-4: Computational domain for rotor-alone computations.

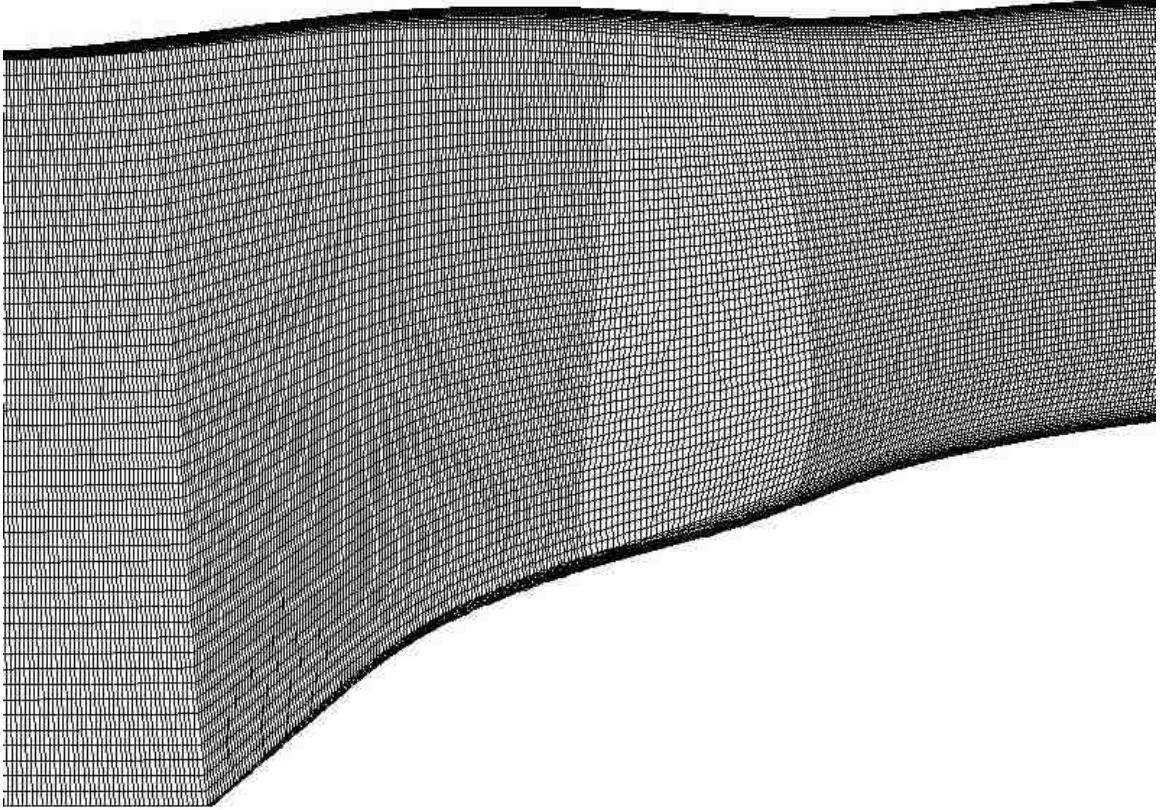


Figure 3-5: Meridional view of rotor-alone grid.

3.4.2 Parallel Force Model Updates

The parallel force formulation used in the body force model adapted from Defoe's [13, 19] work yields a nearly linear variation in efficiency with inlet relative Mach number which does not represent the typical "loss bucket" behavior of turbomachines. This previous parallel force formulation was accurate only for the cutback condition, however, since this work focuses on the takeoff flight condition, changes are needed. To accurately capture changes in efficiency at different flight conditions, the viscous model is updated using Peters' [22] improved formulation:

$$F_p = -\frac{K_{p1}}{h} \left[\left(\overline{M_{rel}}^M \right)^2 + K_{p2} \left(\overline{M_{rel}}^M - \overline{M_{ref}}^M \right)^2 \right] w^2 \quad (3.1)$$

The difference between the previous parallel force formulation and the updated Peters formulation is schematically illustrated in Figure 3-6. It can be seen that the improved model places the peak efficiency at takeoff while also capturing the cutback efficiency correctly.

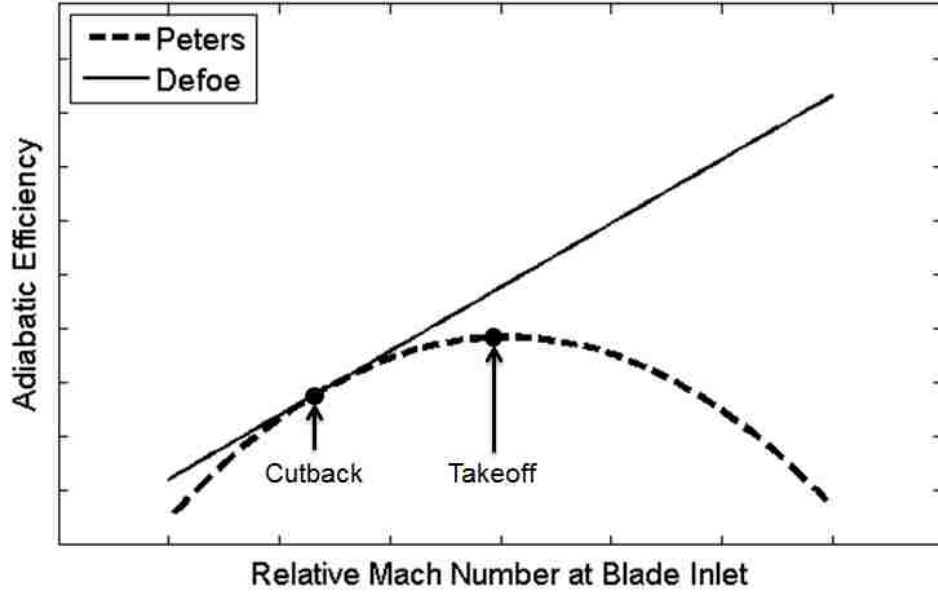


Figure 3-6: Comparison of viscous model behavior with flight condition.

To implement this model, the body force coefficients (K_{p1} and K_{p2}) for this blade row must be determined. Typically, single-passage RANS calculations are used to determine these coefficients, however, without the exact rotor geometry available, these values are determined analytically.

Peters' formulation introduces a quadratic dependence on the mass-averaged relative Mach number at blade row inlet ($\overline{M_{rel}}^M$). $\overline{M_{ref}}^M$ is the mass-averaged relative Mach number at blade row inlet at the design flight condition. In this work, it is assumed that the takeoff flight condition is the design flight condition as the adiabatic efficiency is the highest at this operating point. Knowing the vertex of the curve (takeoff) and another point (cutback) is sufficient to fit the coefficients.

To determine the body force coefficients K_{p1} and K_{p2} , the mass-averaged relative Mach numbers need to be calculated. To determine the mass-averaged relative Mach

numbers, the local relative Mach numbers need to be calculated first. The local relative Mach number is given by,

$$M_{rel} = \sqrt{M_x^2 + M_u^2} \quad (3.2)$$

where M_x is the local axial Mach number at blade inlet and M_u is the local wheel Mach number at blade inlet. Due to the original body force model not being accurate at the takeoff flight condition, M_x is determined analytically using the corrected flow equation,

$$\frac{\dot{m}\sqrt{RT_t}}{AP_t\sqrt{\gamma}} = \frac{M_x}{\left[1 + \left(\frac{\gamma-1}{2}\right) M_x^2\right]^{\frac{\gamma+1}{2(\gamma-1)}}} \quad (3.3)$$

where \dot{m} is the mass flow rate and A is the duct cross-sectional area. M_u is calculated as,

$$M_u = \frac{\Omega r}{\sqrt{\gamma RT}} \quad (3.4)$$

For ease of implementation, the full duct cross-sectional area is divided into 10 radially-equal bands to calculate the mass-averaged Mach numbers. The mass flow through each piece, needed to determine M_x , is determined as,

$$\dot{m}_i = \dot{m} \frac{A_i}{A} \quad (3.5)$$

where the subscript i represents each piece. Once M_x and M_u are determined for each band, M_{rel} can be determined for each band using Equation 3.2. Lastly, the overall mass-averaged relative Mach number is calculated from,

$$\overline{M_{rel}}^M = \frac{\sum_{i=1}^n \dot{m}_i M_{rel,i}}{\sum_{i=1}^n \dot{m}_i} \quad (3.6)$$

This process is performed separately for both the takeoff and cutback flight conditions.

The body force coefficients for Peters' model can be determined knowing that the body force coefficient for Defoe's model was correct for the cutback flight condition. Defoe's model used Gong's parallel force formulation which is given here again for clarity:

$$F_p = -\frac{K_p}{h} w^2 \quad (3.7)$$

Setting Gong's formulation equal to Peters' formulation at cutback yields,

$$K_{p,cutback} = K_{p1} \left[\left(\overline{M_{rel,cutback}}^M \right)^2 + K_{p2} \left(\overline{M_{rel,cutback}}^M - M_{ref} \right)^2 \right] \quad (3.8)$$

$K_{p,cutback} = 0.05$ for Defoe's model [13] and can be scaled using the relative velocities at the blade row inlet and the polytropic efficiencies (η_p) for the different flight conditions. The square of the relative velocity is used to scale the coefficient because it is directly proportional to the parallel force as seen in Equation 3.7. Polytropic efficiency is also used to scale the coefficient because it allows for the comparison of processes that generate different amounts of work. In this case, more work is being done by the blade row at takeoff than at cutback. Polytropic efficiency is calculated as,

$$\eta_p = \left[\frac{\gamma - 1}{\gamma} \right] \left[\frac{\ln \left(\frac{P_{t,out}}{P_{t,in}} \right)}{\ln \left(\frac{T_{t,out}}{T_{t,in}} \right)} \right] \quad (3.9)$$

Therefore, K_p at takeoff can be determined using the scaling relation developed in this work,

$$K_{p,takeoff} = K_{p,cutback} \left(\frac{\eta_{p,cutback} w_{cutback}^2}{\eta_{p,takeoff} w_{takeoff}^2} \right) \quad (3.10)$$

With K_p at two flight conditions determined and knowing $M_{rel} = M_{ref}$ at takeoff,

K_{p1} and K_{p2} can be determined as follows,

$$K_{p1} = \frac{K_{p,takeoff}}{M_{rel,takeoff}^2} \quad (3.11)$$

$$K_{p2} = \frac{K_{p,cutback} - K_{p1} \left(\overline{M_{rel,cutback}}^M \right)^2}{K_{p1} \left(\overline{M_{rel,cutback}}^M - \overline{M_{ref}}^M \right)^2} \quad (3.12)$$

For the R4 rotor, K_{p1} is found to be 0.0336 and K_{p2} is found to be 0.6321. With this updated model implementation, rotor-alone body force computations yielded adiabatic efficiencies at both cutback and takeoff within 1% of the experimental values. Adiabatic efficiency is used for this comparison because they were provided in the GE R4 rotor experimental data at both flight conditions [4].

3.4.3 Camber Line Alteration

Rotor-alone body force computations showed that, despite the corrections to the efficiency, the stagnation pressure and temperature ratios are lower than the experimentally measured values. To correct this, the blade camber profiles are adjusted using the experimentally measured spanwise distribution of stagnation temperature ratio at takeoff given in Hughes et al. [4], the Euler Turbine equation, flow data from a rotor-alone body force computation using the original camber profile, and velocity triangles.

The Euler Turbine equation is used to transform this data into a spanwise tangential velocity distribution at rotor outlet knowing that the inflow is axial. With axial inflow and uniform inlet stagnation temperature, the Euler Turbine equation can be rearranged to calculate the required absolute tangential velocity at rotor trailing edge:

$$v_{\theta,TE}(r) = \left(\frac{c_p T_{t,LE}}{\Omega r} \right) \left(\frac{T_{t,TE}(r)}{T_{t,LE}} - 1 \right) \quad (3.13)$$

Here c_p is the specific heat at constant pressure of air, Ω is the angular velocity of the rotor, r is the radius, and $T_{t,LE}$ and $T_{t,TE}(r)$ are the stagnation temperatures at rotor inlet and outlet, respectively.

Using the velocity triangle at the rotor outlet drawn in Figure 3-7, the required relative flow angle (β_{TE}) is determined to be

$$\beta_{TE} = \sin^{-1} \left(\frac{\Omega r - v_{\theta,TE}}{v_{rel,TE}} \right) \quad (3.14)$$

where $v_{rel,TE}$ is taken from a rotor-alone body force computation. The deviation (δ_{TE}) is subtracted from β_{TE} to get the required blade camber angle at the trailing edge (α_{TE}). It is assumed that the deviation across the span will not change significantly due to the re-camber, so $\delta_{TE}(r)$ is taken from the rotor-alone body force computation with the original blade camber angle distribution.

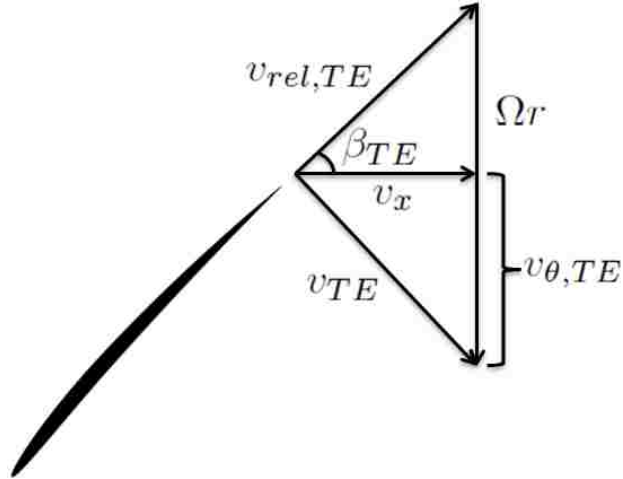


Figure 3-7: Velocity triangle at rotor trailing edge.

The additional trailing edge camber is included by adding a new component to the original blade camber profile. The implementation is such that the change in camber at the leading edge is zero and increases to the required camber at the trailing edge. Mathematically,

$$\alpha_{new}(x, r) = \alpha_{old}(x, r) + \frac{[\alpha_{TE}(r) - \alpha_{old}(x, r)](x - x_{LE})}{x_{TE} - x_{LE}} \quad (3.15)$$

where x is the axial coordinate. The best fit for the additional term can be represented in terms of (x, r) as,

$$\begin{aligned} \frac{[\alpha_{TE}(r) - \alpha_{old}(x, r)](x - x_{LE})}{x_{TE} - x_{LE}} = & b_0 + b_1(x) + b_2(r) + b_3(x^2) + b_4(xr) + b_5(r^2) \\ & + b_6(x^2r) + b_7(xr^2) + b_8(r^3) \end{aligned} \quad (3.16)$$

The original blade camber profile (α_{old}) and the axial locations of the leading and trailing edge (x_{LE} and x_{TE}) are given in Defoe [13] by,

$$\alpha_{old}(x, r) = a_0 + a_1(r) + a_2(x) + a_3(xr) + a_4(r^2) + a_5(x^2) \quad (3.17)$$

$$x_{LE} = c_0 + c_1(r) + c_2(r^2) + c_3(r^3) + c_4(r^4) + c_5(r^5) + c_6(r^6) \quad (3.18)$$

$$x_{LE} = d_0 + d_1(r) + d_2(r^2) + d_3(r^3) + d_4(r^4) + d_5(r^5) + d_6(r^6) \quad (3.19)$$

Table 3.5 lists the coefficients a_i from Defoe's model and b_i determined in this work and Table 3.6 lists the coefficients c_i and d_i for the axial locations of the leading and trailing edge. The resulting change in the blade camber line is shown at 50% span in Figure 3-8. The overall mass-averaged stagnation temperature ratio, stagnation pressure ratio, and adiabatic efficiency from both the experiments and the computation following the re-camber are given in Table 3.4. The computed overall performance values are

in agreement with the experimental data to within less than 1%. In Figure 3-9, the computed stagnation temperature ratio profiles are shown alongside the experimental results. The computed stagnation temperature ratio in the tip region does not closely agree with the experimental data. This was initially hypothesized to be caused by an absence of a tip gap in the body force model; however, an assessment with the addition of a tip gap of 1% span resulted in no significant changes. The level of agreement in the tip region can be investigated in future work, however, the agreement of the mass-averaged values is considered good enough for the purposes of this thesis.

Table 3.4: Overall mass-averaged rotor-alone performance comparison at takeoff. [5]

	Experimental	Computation
Stagnation Pressure Ratio	1.49	1.49
Stagnation Temperature Ratio	1.13	1.13
Adiabatic Efficiency	0.92	0.92

Table 3.5: Blade camber distribution coefficients for the R4 rotor.

a_0	1.63	a_3	54.9	b_0	5.799	b_5	32.26
a_1	0.46	a_4	-12.2	b_1	-64.40	b_6	-765.5
a_2	-14.3	a_5	0.4	b_2	-38.11	b_7	-413.4
				b_3	139.1	b_8	35.42
				b_4	423.4		

Table 3.6: Leading and trailing edge coefficients for the R4 rotor.

c_0	0.181	c_4	-72.9	d_0	0.0403	d_4	961
c_1	-1.21	c_5	359	d_1	2.62	d_5	-2695
c_2	9.45	c_6	-402	d_2	-4.46	d_6	2764
c_3	-19.1			d_3	-129		

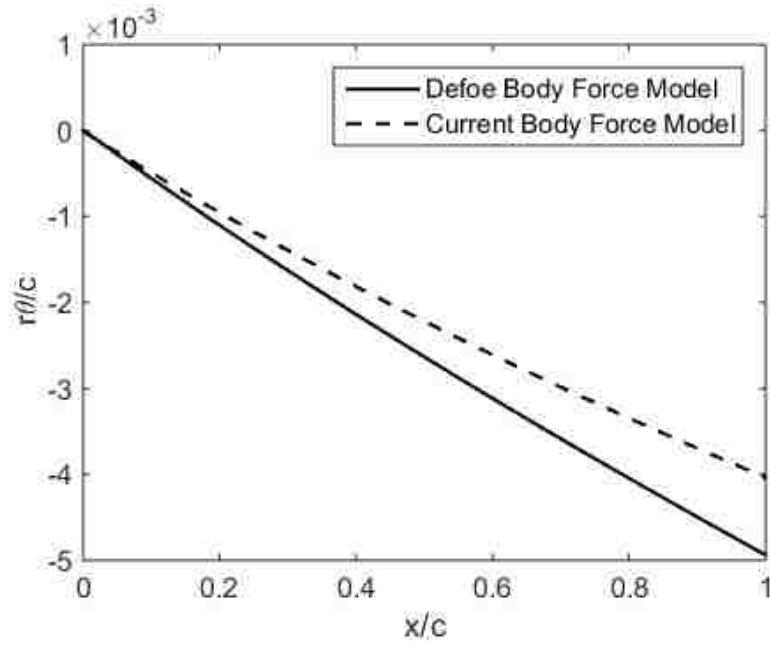


Figure 3-8: Comparison of blade camber line at 50% span.

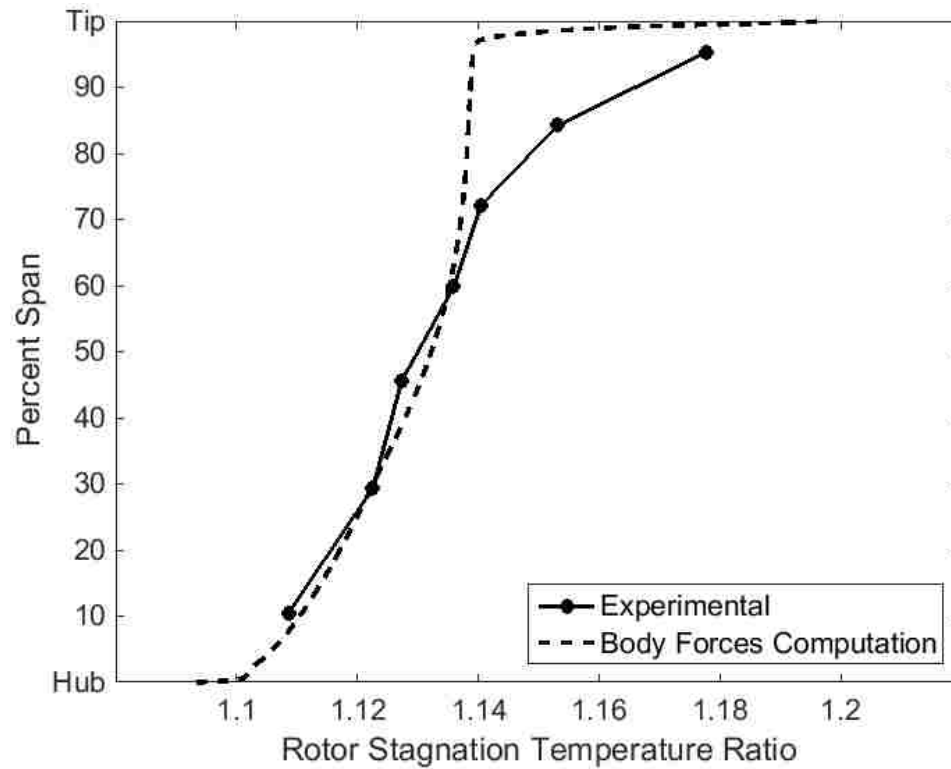


Figure 3-9: Circumferentially-averaged stagnation temperature ratio profile at rotor trailing edge at takeoff. [4]

3.4.4 Rotor Wake Generation

The method used to generate blade wakes from the body force model of the rotor in this work is conceptually similar to the production of rotor shock noise via body force perturbations in Defoe et al. [19], however, in this work the rotor wakes are generated via a local concentration of viscous forces at each blade trailing edge.

By design, the body force provides a circumferentially-averaged flow field within the swept volume, therefore, if the viscous force is not concentrated in a region notionally associated with each (absent) blade trailing edge, it would be effectively smeared out. All the viscous force in the rotor swept volume in this work is concentrated in the last 15% chord, in a rotating circumferential region for each blade approximately equal to the boundary layer width at the blade trailing edge. The size of the boundary layer at the blade trailing edge is approximated using flat plate theory. For future work, the viscous force can be concentrated to whatever extent is needed to get accurate wake shapes and magnitudes. The body force model is thus inviscid everywhere except in these discrete trailing edge regions shown in red in Figure 3-10.

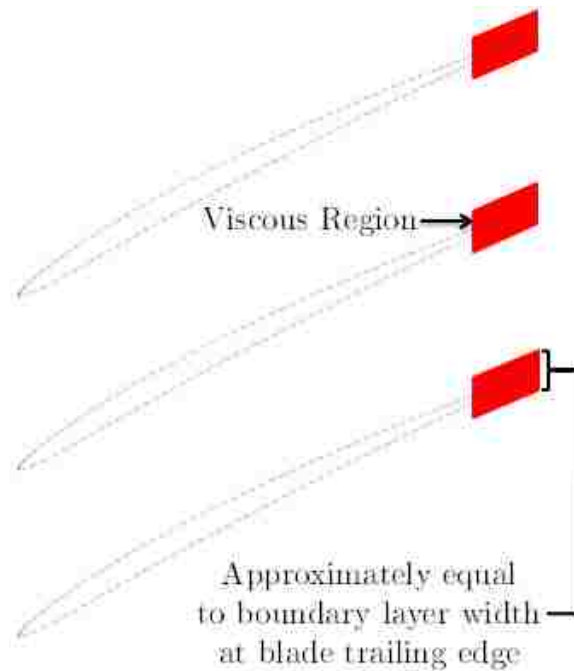


Figure 3-10: Blade trailing edge viscous regions.

In an iterative process, the width and the magnitude of the viscous region is varied while maintaining the same amount of mass-averaged entropy generation until the wake size and depth is in satisfactory agreement with experimental data. This is done by introducing an amplification factor to the parallel force used to control the strength, or depth, of the wake. This amplification factor is needed because the relative velocities are smaller near the trailing edge than near the leading edge where some of the viscous force is normally located. The width of the wake is controlled by the circumferential extent of the viscous force region. After each iteration, the wake shape and overall performance of the rotor is compared to the experimental results.

Using this method, fair agreement with the experimental wake shape is obtained while maintaining a less than 1% error on the overall performance shown by Table 3.4. Podboy et al. [30] provided contours of the axial velocity on a measurement plane one chord length downstream of the rotor trailing edge. Figure 3-11 depicts the local axial velocity normalized by the mass-averaged axial velocity at this measurement plane for both the experiments and computations, for 25%, 50%, and 75% span. It can be seen that in the computations, the depth of the wake is slightly under-predicted but that the general character of the wake is captured. This wake shape is acceptable because an approximation of the stator geometry is used for this work (discussed in Section 3.5), therefore, the goal of this work is not to replicate the experimental data exactly but to quantify the differences between computation and experiment.

The amplification factor used here is constant across the span, however, in future work, it would be possible to implement a spanwise-varying amplification factor to improve the level of agreement. This result demonstrates that it is possible to produce discrete blade wakes using body forces, avoiding the use of computational interfaces between blade rows. This allows for less expensive computations that can still capture acoustics without the time step size penalties associated with the use of sliding interfaces.

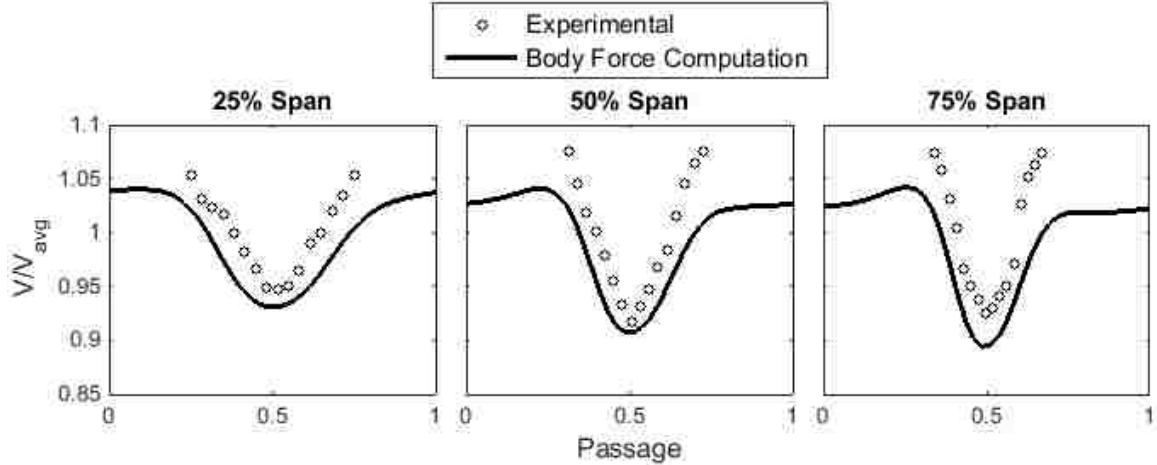


Figure 3-11: Non-dimensionalized axial velocity at 25%, 50%, and 75% span one chord downstream of the rotor trailing edge at takeoff. [30]

3.5 Stator Geometry Development

Since rotor-stator interaction is the noise source of interest in this work, a solid model of the stator vanes (not a body force field) are needed in the computation. The detailed geometry of the stator vanes from the NASA SDT is not available in the literature so an approximate stator vane design is developed based on the limited specifications provided by Hughes [5]. This data is reproduced for reference in Table 3.7.

As a part of the NASA SDT, Heidelberg [8] provided tonal modal results for the GE R4 rotor combined with three different stator vanes. The radial baseline stator vane is chosen for this work because the highest interaction sound power level is generated with this vane.

Information such as the shape of the camberline, leading and trailing edge radii, etc. are not given. Using the available information, SolidWorks 2014 [34] is used to develop the vane cross-sections at the hub, pitchline, and tip locations. Circular-arc vane sections are assumed and the leading and trailing edge radii are iterated upon until the flow rounding the leading edge remains attached, the outlet swirl is near

zero, and the experimental pressure ratio is matched. The three cross-sections are stacked at 50% chod. Using blade solidity,

$$\sigma = \frac{cV}{2\pi r} \quad (3.20)$$

where c is blade chord and V is the number of stator vanes, the radius at the pitchline location is determined to be 60% span.

The blade-to-blade flow field analysis code MISES [35] is used to iterate upon stator geometries. Figure 3-12 shows an example of a grid generated by MISES. This process ensured that the stagnation pressure loss/entropy generation across the stator is accurately captured since the rotor-alone stagnation pressure ratio is already known to be in good agreement as given in Table 3.4. The flow over vane sections at the hub, pitchline, and tip radii are first computed using MISES. To obtain accurate inlet and outlet boundary conditions, experimental data from Hughes [4] is used. Hughes presented the spanwise Mach number distribution at the duct outlet from an experiment which included the rotor and the stator as well as the spanwise swirl angle at duct outlet from a rotor-alone experiment. Using this data, the Mach number at stator outlet and the swirl angle at stator inlet are approximated at each location to be used as boundary conditions. Since there is essentially zero swirl at the stator outlet, the Mach number at that position is approximated using the corrected flow equation,

$$\frac{\dot{m}\sqrt{RT_t}}{AP_t\sqrt{\gamma}} = \frac{M_x}{\left[1 + \left(\frac{\gamma-1}{2}\right) M_x^2\right]^{\frac{\gamma+1}{2(\gamma-1)}}} \quad (3.21)$$

Between the stator outlet and the duct outlet, all the flow properties on the left side of the equation can be assumed to remain constant except the duct cross-sectional area. Therefore, if the right side of the equation is represented as a function $f(M_x)$, at any two locations (here the stator outlet and duct outlet):

$$[f(M_x)(A)]_{stator\ outlet} = [f(M_x)(A)]_{duct\ outlet}$$

Once $f(M_x)_{stator\ outlet}$ is determined, the approximate Mach number at stator outlet can be computed. This process is done separately for all three radial locations. The swirl angle at the stator inlet is assumed to be the same as at the duct outlet in the rotor-alone case based on the conservation of angular momentum. With these two boundary conditions for each of the three radial locations, MISES is used to analyze the flow; a MISES blade surface isentropic Mach number result at pitchline radius is given in Figure 3-13. The primary outputs of interests are the outlet flow slope s_2 and the total loss coefficient ω , where

$$\omega = \frac{p_{t,out}^{isen} - p_{t,out}}{p_{t,in} - p_{in}} \quad (3.22)$$

The superscript *isen* represents the isentropic stagnation pressure.

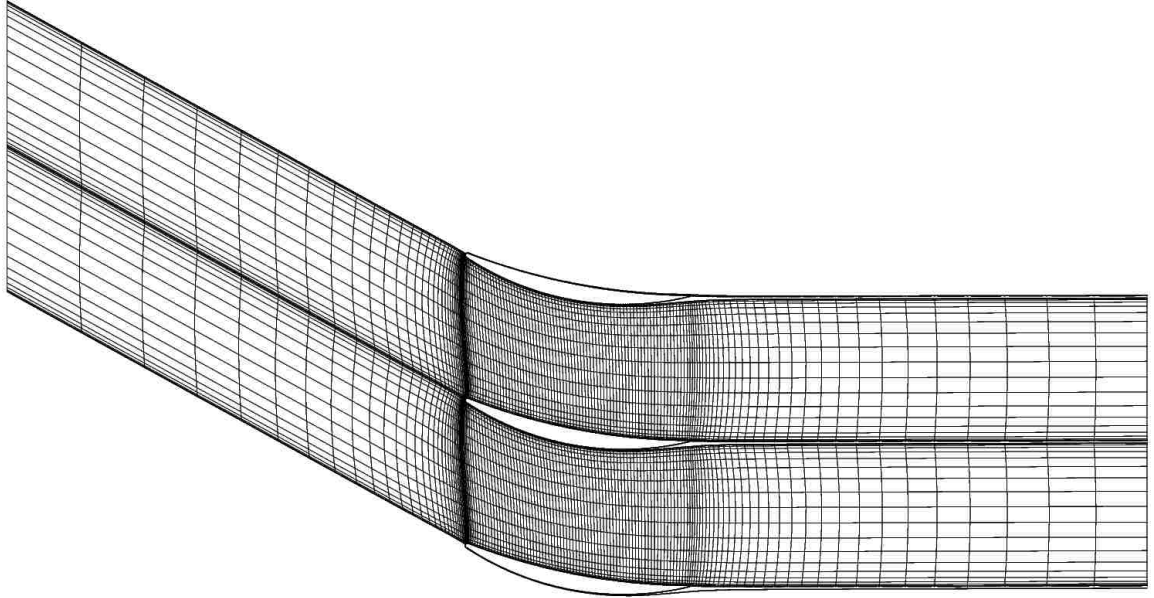


Figure 3-12: Example of MISES generated grid showing two stator passages. [35]

Table 3.7: Summary of stator geometry specifications (from Hughes et al. [5]).

	Span Location	Radial Baseline
No. of Blades	—	54
Aspect Ratio	Pitchline ¹	3.51
Chord (in)	Pitchline ¹	1.57
Solidity	Hub	2.25
	Pitchline ¹	1.52
	Tip	1.23
Stagger (deg)	Hub	12.56
	Pitchline ¹	10.29
	Tip	10.65
Camber (deg)	Hub	38.40
	Pitchline ¹	34.56
	Tip	40.49
Maximum thickness/chord	Hub	0.0707
	Pitchline ¹	0.0702
	Tip	0.0698

¹ Pitchline radius determined to be 60% span as explained in text.

The leading and trailing edge radii are iterated upon until the outlet swirl slopes are close to zero (hub: 0.014, pitchline: -0.007, tip: -0.013) and the stagnation pressure ratio neared the reported value of 0.988. The profiles at the three radial locations are then used to generate a 3D model of the approximate vane. 10 cross-sections of the 3D vane are tested within MISES at equally spaced spanwise locations to verify that the cross-sections are joined appropriately and to obtain a more accurate overall performance prediction for the assumed vane.

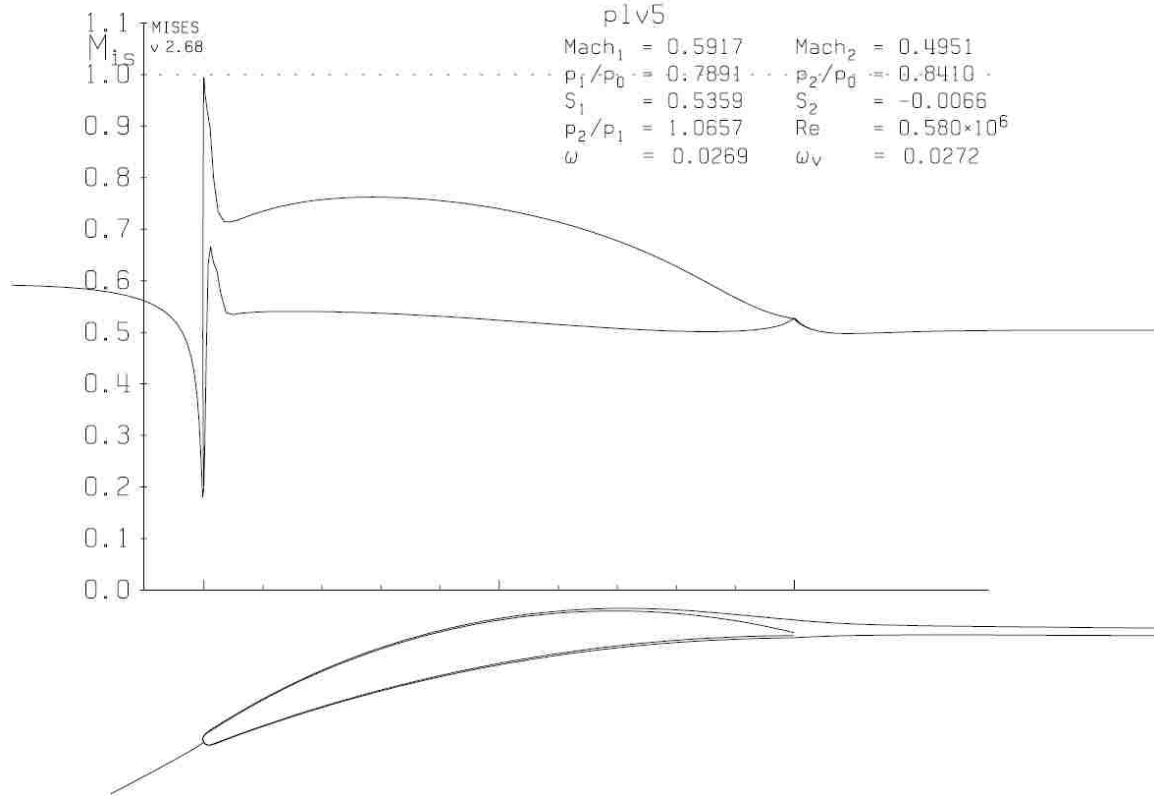


Figure 3-13: MISES isentropic Mach number plot at the pitchline location. [35]

With 1.0% chord leading edge and 0.5% chord trailing edge radii, the stage adiabatic efficiency with the assumed geometry is 90.7% compared to a reported value by Hughes [5] of 89.0%. The stagnation pressure ratio of the assumed vane geometry is 0.993 compared to the reported value of 0.988. The reported value of the stagnation pressure ratio for the stator is determined by dividing the stage stagnation pressure ratio by the rotor-alone stagnation pressure ratio from Hughes [5]. The mass-averaged outlet swirl slope of the assumed geometry is determined to be 0.0037 which corresponds to an outlet swirl angle of 0.2°. Therefore, despite only achieving fair agreement with actual stage performance, this assumed geometry is considered to be acceptable because exact stator geometry is not essential to generate the interaction noise and a uniform inflow assessment is performed to quantify the differences between the experimental and computational results. The final stator vane developed is depicted in Figure 3-14.

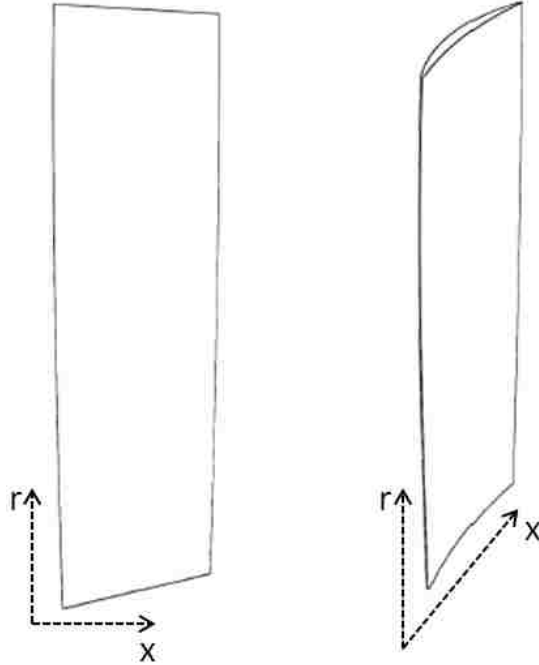


Figure 3-14: Side view (left) and perspective view (right) of stator vane.

3.6 Aeroacoustic Numerical Details

A uniform inflow assessment is performed on the NASA SDT geometry to determine the accuracy of the approach described in this thesis. The uniform inflow assessment is carried out using a half annulus grid. Since the rotor and stator have 22 and 54 blades, respectively, a half annulus is the minimum sector required to capture the periodicity of the rotor wake-stator interaction. Future non-uniform inflow computations will be full annulus. The purpose of the uniform inflow assessment is not to see if the results match the experimental data, but to ensure that the correct modes are generated and to quantify the difference in the predicted sound power so that a useful comparison can be made for future non-uniform inflow cases.

Since the focus of this work is on the downstream propagated sound power, the computations consider only the internal duct flow. The region of interest (domain) for the computations is illustrated in Figure 3-15. The inlet of the domain is located

at the inlet throat of the nacelle. The outlet is located at the downstream end of the nacelle and is the location at which the sound power is assessed in the experimental data.

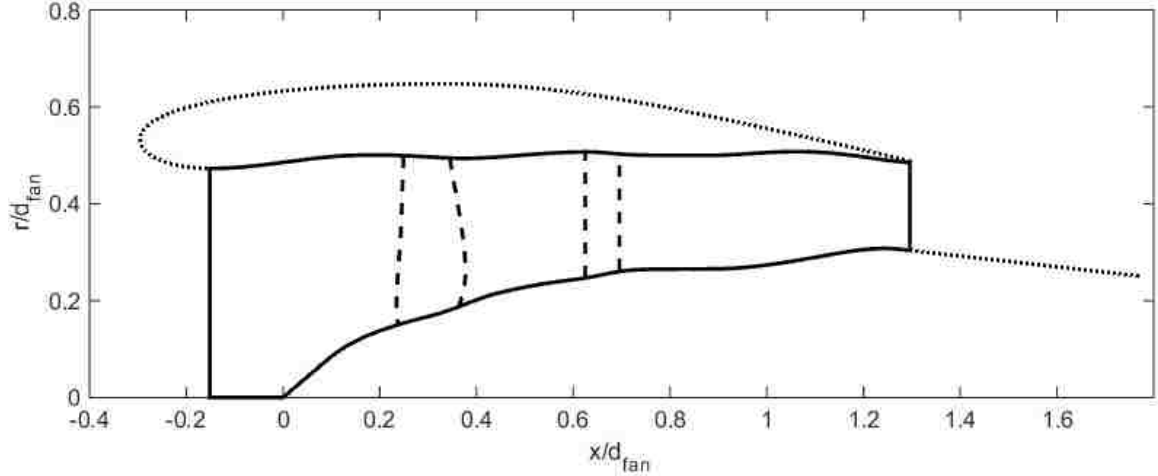


Figure 3-15: Computational domain for aeroacoustic computations (solid line). Flow is from left to right. Swept rotor and stator volumes are shown by dashed lines; the nacelle outer surface is shown by the dotted line.

The maximum cell size is set to 0.0015 m to allow for accurate propagation of acoustic waves in the second-order accurate finite volume solver based on previous studies [12]. This cell size corresponds to 25 PPW. For the half and full annulus computational grids, this results in approximately 52 million and 104 million hexahedral cells, respectively.

The time step size in the unsteady computations is set to 1/1320 of half the rotor revolution period ($\sim 1.796 \times 10^{-6}$ s). This is set based on time step studies [11] which determined that 60 time steps per period for the highest frequency of interest (2BPF) are required for accurate acoustic propagation. Since the rotor is represented by a body force field, a sliding interface is not needed which enables this larger time step and thus reduces the computational cost. The reduction in computational cost can be determined by comparing the maximum time step allowable with and without the interface. Using the interface, Equation 2.2 can be used to calculate the maximum

possible time step. Recall that for this work, $m = -10$ is the acoustic interaction mode of interest. For this mode, $\theta_{period} = 2\pi/10$ and $\Omega_{mode} = -44\Omega/10$. Using Equation 2.2, it can be determined that approximately 2287 time steps per half rotor revolution are required for accurate acoustic wave propagation through the interface. Therefore, the use of a sliding interface would be 73% more computationally expensive than the body force approach based purely on allowable time step size.

3.6.1 Computational Domain Splitting Method

For this work, insufficient computational resources were available to simulate the entire domain at once. SHARCNET, the high-performance computing cluster used, permits the use of a maximum of 64 processes of ANSYS Fluent in a single computation. For the half-annulus computational grid with 52 million cells, it is estimated that at least 256 processes would be required. A method of splitting the domain had to be developed and validated to reduce the cost of a single computation to a level which could be run with the available resources.

The full domain is axially split into three partially-overlapping computational domains: (1) the inlet flow section, (2) the source generation section and (3) the noise propagation section. In the following paragraphs, the planes referred to are labeled in Figure 3-16 (based on the geometry from Hughes et al. [5]). For the uniform inflow assessment computations, all domains are half annulus while for the future non-uniform inflow computations, all domains will be full annulus (with associated increased computational costs).

The first domain (inlet flow section) is used to determine the flow redistribution at plane B caused by the upstream influence of the rotor. It begins at plane A and ends at plane C. Plane A is the inlet throat (minimum outer radius) of the nacelle and plane C is midway between the rotor and the stator. A steady computation on a relatively coarse grid (~ 7 million cells for the full wheel), which includes the

steady rotor body forces (no wakes), is used to obtain the flow direction, stagnation pressure, and stagnation temperature profiles at plane B. The stagnation temperature is expected to vary only at the hub due to the work done by the rotating hub. Plane B is located one rotor chord upstream from the rotor leading edge. This location is chosen based on where the upstream influence of the individual wakes will become negligible such that an axisymmetric inlet condition is reasonable to apply. The non-axisymmetric upstream influence decays exponentially with the characteristic length scale of $2\pi r_{tip}/B$. At one rotor chord upstream of the rotor leading edge, the upstream potential field magnitude will be decayed by at least 90% of its initial value.

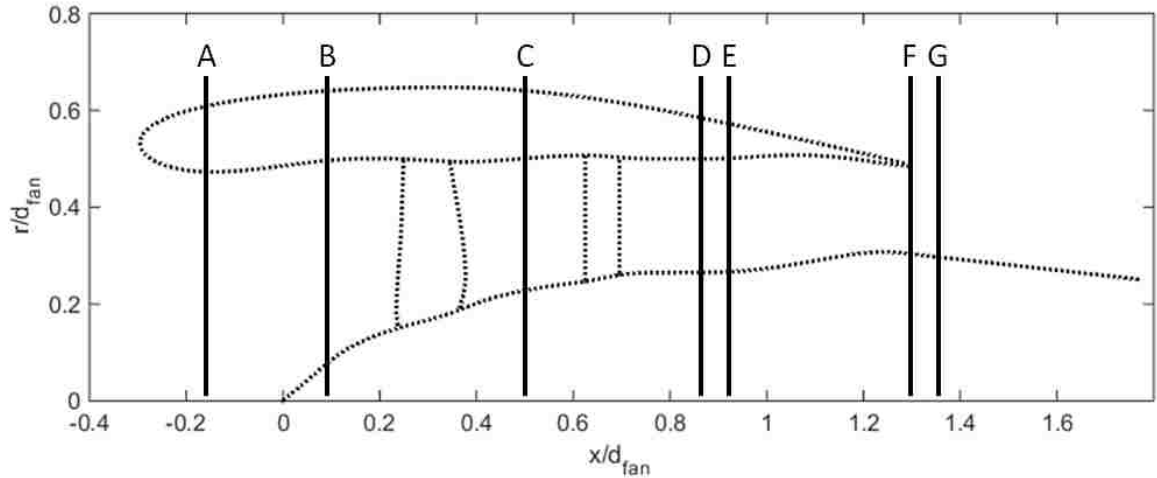


Figure 3-16: Fan nacelle geometry showing planes used to define computational domains [5]. The rotor and stator leading and trailing edge profiles are also shown. Flow is from left to right.

The acoustic source generation occurs in the second domain. It spans from plane B to E; the intermediate plane D is located two stator chords downstream of the stator trailing edge. The location of plane D is chosen to ensure that the interaction source generation is complete prior to the end of the domain and that the rotor wakes have decayed sufficiently. Raj et al. [36] provide a relation to estimate the decay rate of the rotor wakes which is governed by viscous mechanisms. The rate at which the wake decays differs in the region immediately following the trailing edge and further

downstream. It is estimated that 80% to 90% of the wake is decayed within 10% of the blade spacing. Further downstream the wake decays at a slower rate. Plane D is located almost 3.5 blade spacings (based on rotor tip radius) downstream of the rotor trailing edge to ensure the rotor wakes have almost completely mixed out. At the inlet of this domain, the profiles obtained from the first domain are imposed as the boundary conditions. An unsteady computation on an aeroacoustic grid (~ 68 million cells for the full wheel) is carried out on this domain to generate rotor blade wakes and their interactions with the stator vanes. The sole source of unsteadiness is the rotating wakes in the rotor body force zone. The time-resolved flow field at plane D is the key output of this computation. In addition, it will be shown that time-resolved flow field data between the stator trailing edge and plane D can be used to provide an assessment of the acoustic source generation. To prevent artificial reflections at the boundaries of this domain, acoustic buffer zones are used at both the inlet and outlet; E is located downstream of D to accommodate the outlet buffer zone. These buffer zones were discussed in Section 2.5. Figure 3-17 illustrates the resultant computational grid. The left part of the figure depicts a meridional view of the axisymmetric grid upstream of and in the rotor swept volume for the source generation domain. The right part of the figure is a perspective view of a single passage of the grid around a stator vane, also for the source generation domain.

The third and final domain is used to propagate the acoustic signals to the downstream end of the nacelle. It spans from plane D to plane G. Another unsteady computation on an aeroacoustic grid (~ 25 million cells for the full wheel) is carried out on this domain using the time-varying flow profiles at plane D from the second domain as the inlet boundary condition. The acoustic signature at plane F can be directly compared to the experimental data. To prevent artificial reflections, an acoustic buffer zone is used at the outlet boundary located in an artificial constant-area extension of the fan duct at G.

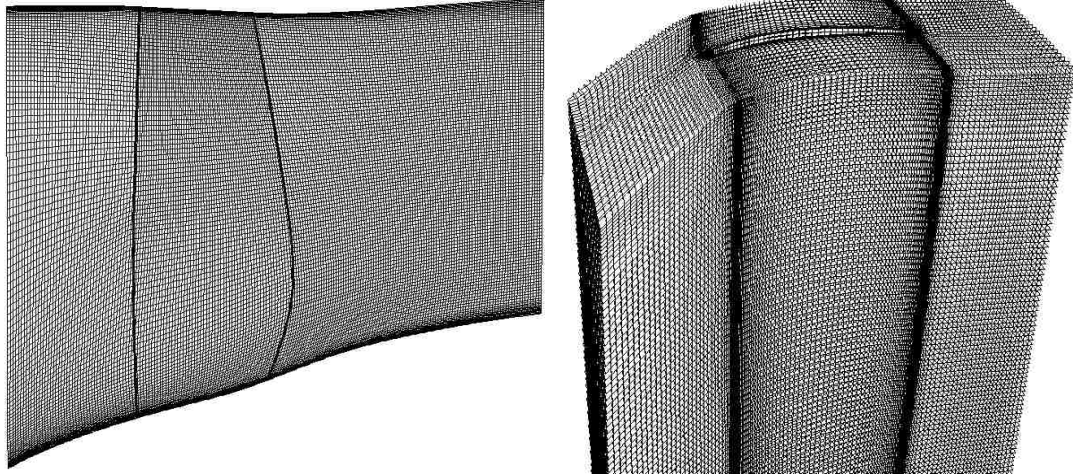


Figure 3-17: Meridional view of the axisymmetric grid upstream of the stator (left) and perspective view of a single passage grid around a stator vane (right) from the source generation domain.

3.6.2 Split Domain Method Validation

The split domain approach is validated using low cost two-dimensional computations with an axial vane and a temporally- and spatially-varying perturbation provided at the inlet. The computation is carried out for the entire domain as well as for two domains, split two vane chords downstream of the vane trailing edge. Both the full and split domains are depicted in Figure 3-18.

The inlet boundary condition of both simulations includes a time-varying stagnation pressure profile to simulate rotor wakes while keeping the stagnation temperature constant. The stagnation pressure profile is arbitrarily selected and has an amplitude (A) of 100 Pa, a wavelength (λ) of 0.25 m (also the height of the domains), and a frequency (f) of 2 Hz. Mathematically,

$$P_t = 101325 + A \sin \left[\left(\frac{2\pi}{\lambda} \right) y - (2\pi f) t \right] \text{ Pa} \quad (3.23)$$

where y is the spatial coordinate along the height of domain and $y = 0$ at the lower periodic boundary.

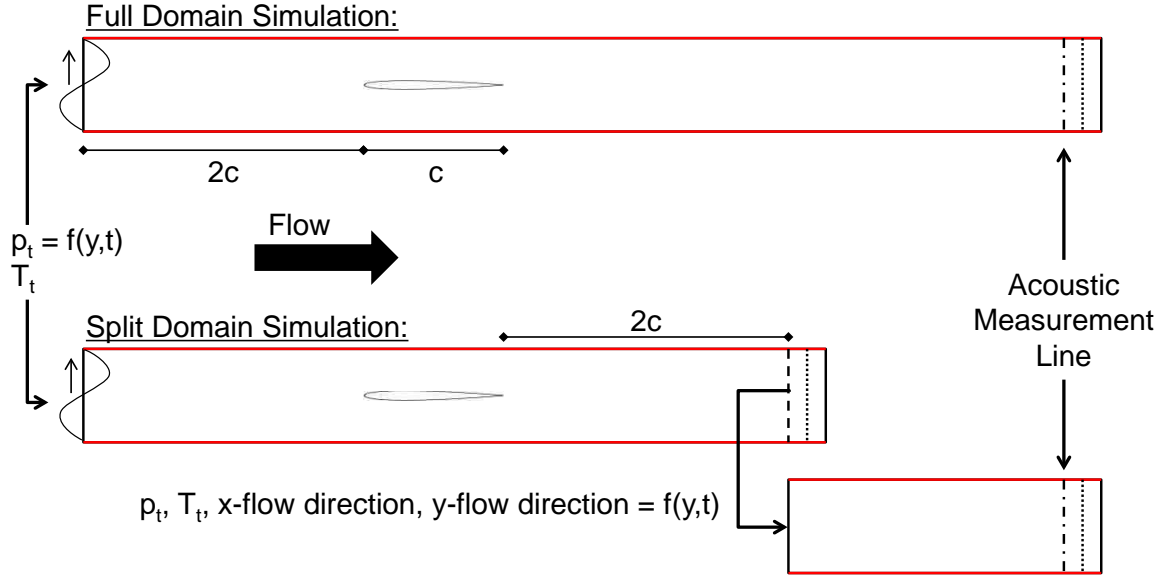


Figure 3-18: Split domain approach validation method. Red lines indicate a periodic boundary.

The maximum cell size is determined to be 0.01 m using the wavelength of the inlet perturbation and the guideline of 25 PPW for accurate wave propagation. For the split domain calculation, the grid points at the split plane are matched between the upstream and downstream domains. Furthermore, with a period (T) of 0.5 s, a time step of $T/60$ is used for these computations. The mass flow through the two domains is chosen to be 8.0 kg/s to yield an approximate inlet Mach number of 0.1, the same as the freestream Mach number used in the SDT experiments. All computations include a downstream buffer zone to avoid non-physical wave reflections at the outlet.

To verify convergence for both simulations, the time-varying mass-averaged pressure at the at the acoustic measurement line is compared for each period. The solution is deemed complete once the pressure reaches periodic steady-state.

Once the upstream domain computation reaches periodic steady-state in the split approach, the time and space varying stagnation pressure/temperature and flow direction at the split plane are used to specify the inlet boundary conditions of the downstream computation.

The total sound power of both cases are compared at the acoustic measurement line. The sound power (ϖ_p) propagating through a surface A is given by,

$$\varpi_p = \int_A \vec{I} \cdot d\vec{A} \quad (3.24)$$

where \vec{I} is the sound intensity vector. The sound intensity vector can be determined using,

$$\vec{I} = \frac{1}{T} \int_0^T p' \vec{v}' dT \quad (3.25)$$

where T is the period, p' is the fluctuating pressure, and \vec{v}' is the fluctuating velocity vector.

The results are given in Table 3.8 and it can be seen that for the split domain computations, the total sound power level at the acoustic measurement line of the downstream (noise propagation) domain is within 0.3 dB of the sound power level at the same location for the full domain. This validates the split-domain approach for acoustic propagation.

Table 3.8: Sound power level at the acoustic measurement line.

Simulation	Sound Power Level (dB/passages)
Full	88.6
Split	88.3

Chapter 4

Uniform Inflow Assessment

Results which allow the capabilities of the rotor-stator interaction noise generation approach to be assessed are presented in this chapter. Due to time constraints imposed by significant downtime incurred on the Sharcnet cluster [37], only the computations on the first and second domains of the uniform inflow assessment could be completed. Nevertheless, it will be shown that the approach shows great promise for accurately generating tonal turbomachinery noise.

4.1 Inlet Flow Computation

As discussed in Section 3.6.1, the first domain of the computation spans from the inlet throat to midway between the rotor and the stator. In this computation, the flow variable profiles one rotor chord upstream of the rotor leading edge are of interest. These profiles are used as the inlet boundary condition of the source generation computation. All of the pertinent profiles are given in Figures 4-1 and 4-2. The stagnation pressure profile shows losses at the hub and casing walls as expected due to viscous effects. The stagnation temperature profile shows increases at the hub due to the viscous work associated with the rotating wall upstream of the rotor. The losses seen in the stagnation temperature profile at the casing is caused by a non-physical

numerical error due to the large turning angles for the flow, however, it is expected to have a negligible effect on the overall flow field because it is a small deviation confined to a very small region. Lastly, the flow direction components are as expected; the tangential flow direction shows flow in the negative θ -direction near the hub due the rotor spinning in the same direction.

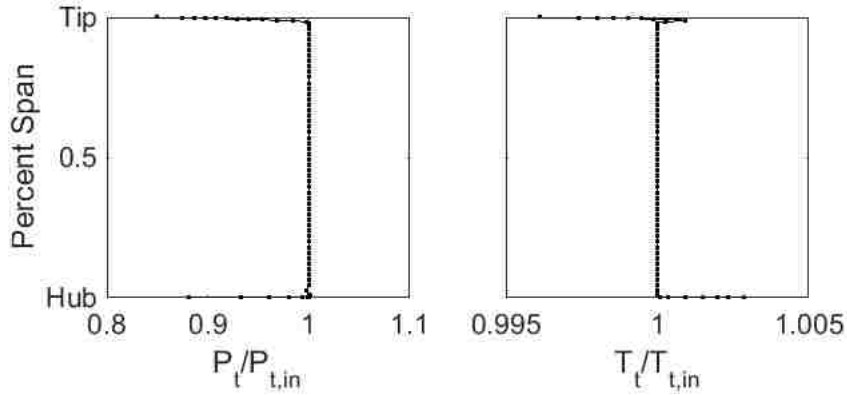


Figure 4-1: Flow quantities one rotor chord upstream of the rotor leading edge.

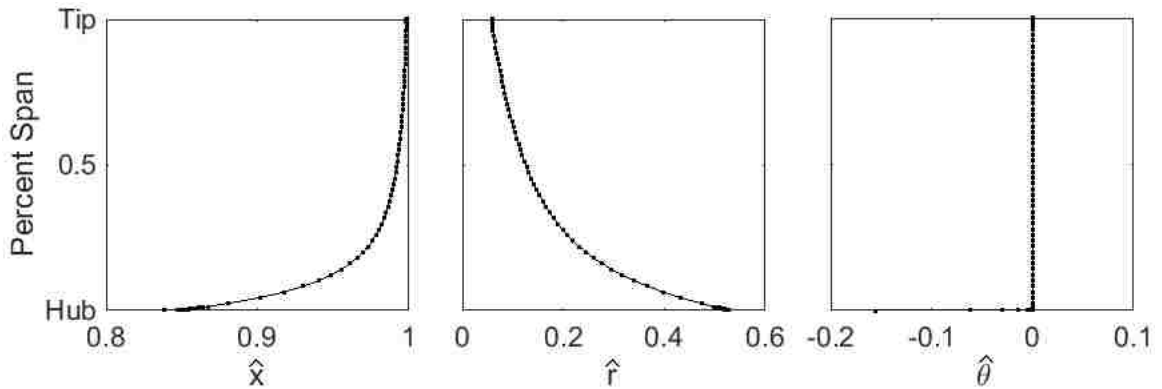


Figure 4-2: Flow direction components one rotor chord upstream of the rotor leading edge.

4.2 Source Generation Computation

Recall, the source generation computation goes from one rotor chord upstream of the rotor leading edge to two stator chords downstream of the stator trailing edge. Figure

4-3 shows the generation of the rotor wakes by the body forces and their incidence onto the stator vanes.

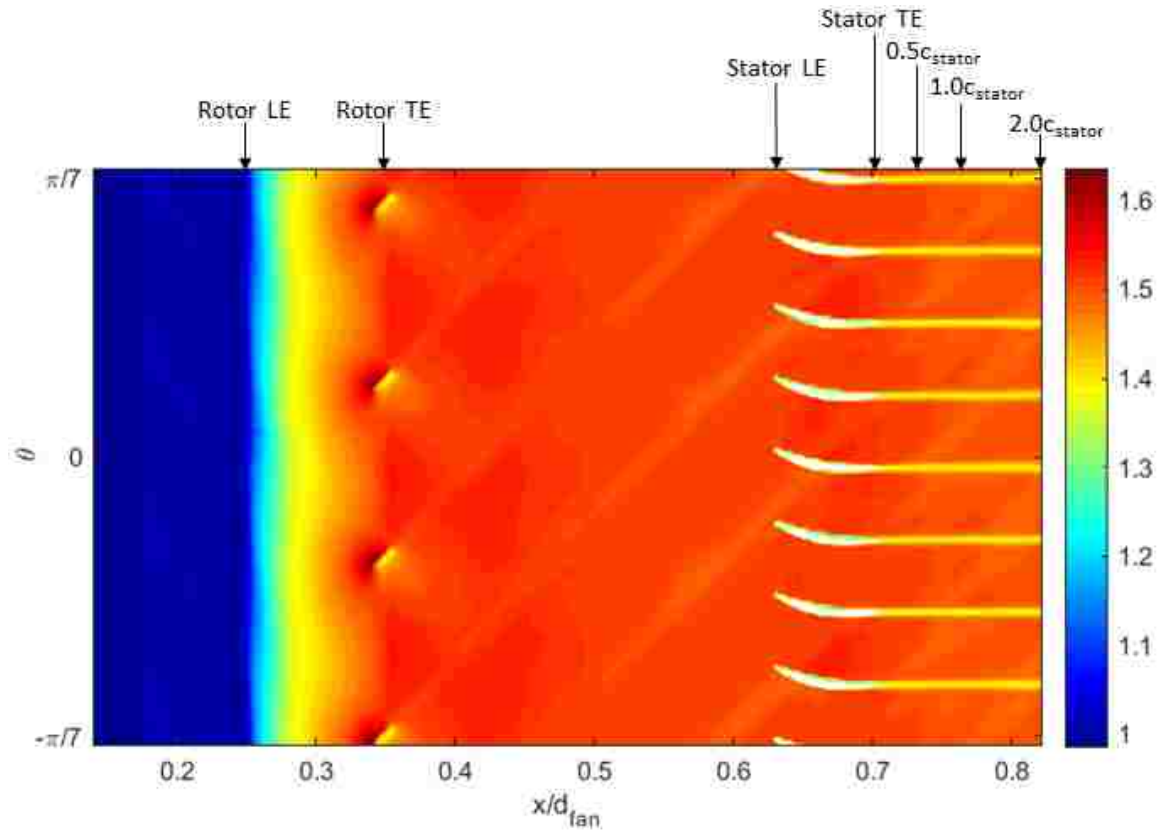


Figure 4-3: Instantaneous contours of stagnation pressure non-dimensionalized by inlet stagnation pressure at 75% span (based on stator trailing edge) after 1/4 rotor revolution. Flow is from left to right and the rotor is rotating in the negative θ direction.

Modal decompositions of the acoustic field are carried out on axial planes one-half stator chord, one stator chord, and two stator chords downstream of the stator trailing edge spanning from 2.5 to 3 rotor revolutions. The resulting frequency resolution of the analysis is 1/10 of the shaft frequency. The modal decomposition approach used is described by Sutliff [24]; it is the same approach used to analyze the experimentally measured acoustics in the literature. This method assumes a uniform flow on the plane of interest. The axial Mach number across the span, averaged in time and then circumferentially, is shown in Figure 4-4 for the three planes for which modal

decompositions are performed. While the mean flow is seen to be non-uniform, the experimental data from the SDT is analyzed using the same method. A direct comparison between the computation and the experimental data can therefore be made with confidence despite the approximation inherent in neglecting the radial variations in the mean flow.

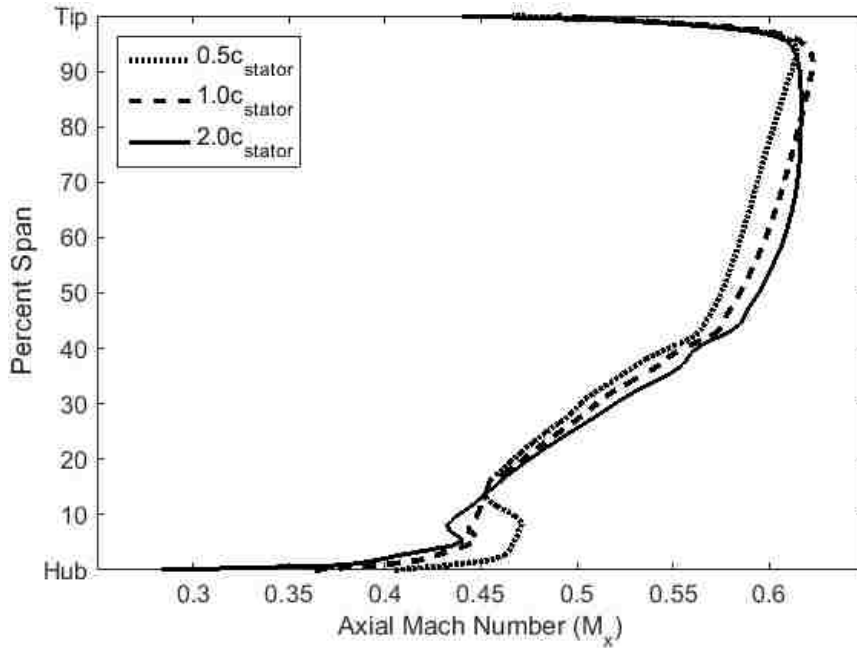


Figure 4-4: Time- and circumferentially-averaged axial Mach number downstream of the stator trailing edge.

To assess when the computation reaches a periodic steady-state, a moving-average analysis of the sound power for modes of interest at two stator chords downstream of the stator trailing edge is conducted. This analysis is conducted by performing modal decompositions on successive partially overlapping half rotor revolutions of unsteady acoustic data. 24 moving averages are computed for each rotor revolution. The results are illustrated in Figure 4-5. It is clear that the amplitude of each of the three modes of interest has become steady to within 1 dB by the end of two rotor revolutions.

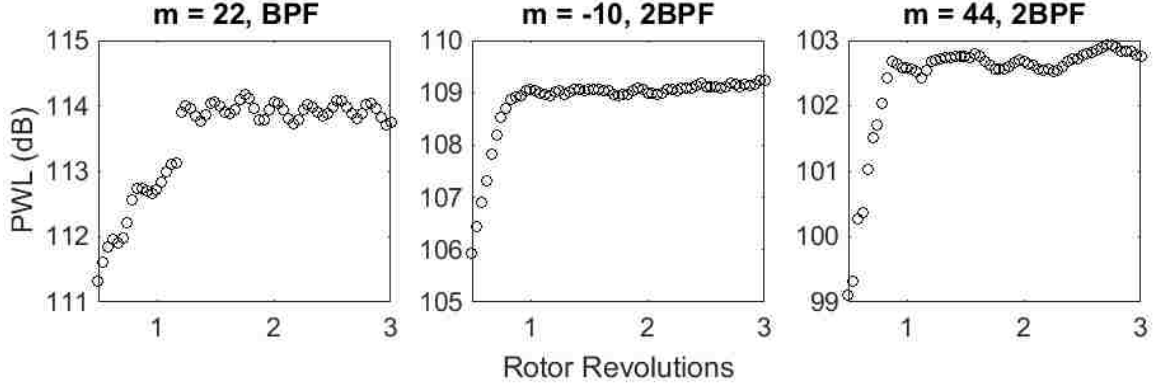


Figure 4-5: Moving-average analysis of the acoustic modes of interest two stator chords downstream of the stator trailing edge.

The dominant cut-on modes measured experimentally are shown in Table 3.1 and more detailed modal power distribution plots are found in reference [8]. The noise floor of the experimental results for BPF and 2BPF is about 105 dB. Most circumferential modes for both BPF and 2BPF have sound power levels between 100 and 110 dB. The computational noise floor is much lower as a result of the lack of broadband noise in the numerical simulations. This reveals additional cut-on modes which cannot be clearly seen in the experimental data. In the experimental data at BPF, the largest peak is the rotor-locked mode ($m = 22$) with a sound power level of 131 dB. For 2BPF, the largest peaks are the rotor locked mode ($m = 44$) and the interaction mode ($m = -10$), with sound power levels of 133 dB and 131 dB, respectively. In that work, the power was summed across all cut-on radial modes for each circumferential mode.

In Figure 4-6, the sound power levels of the acoustic modes of interest at BPF and 2BPF are compared between the experimental data (at plane F from Figure 3-16) and computational results at the three axial planes considered. The values in this figure are summed across all radial modes. This figure reveals relatively small changes between the two furthest downstream planes which suggests that sound power generation is essentially complete by two chords downstream of the stator and that

comparing the upstream computational results to the experimental results at duct exit is useful.

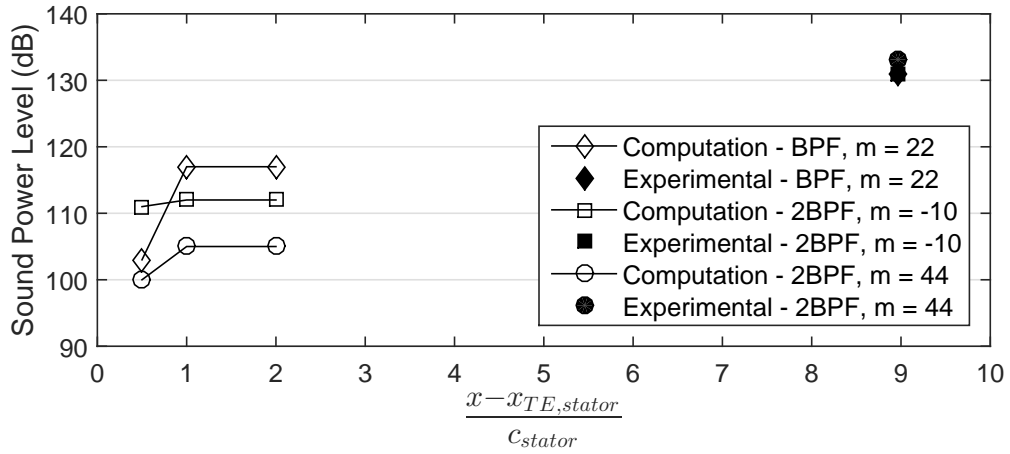


Figure 4-6: Tone sound power level comparison at takeoff. [8]

The modal power distributions from the computations determined in the same way at the three axial locations identified above are shown in Figures 4-7, 4-8, and 4-9 for the planes one-half stator chord, one stator chord, and two stator chords downstream of the stator, respectively. This data in these figures are based on data from 2.5 to 3 rotor revolutions and include distributions at both BPF and at 2BPF.

At the BPF, the rotor-locked mode ($m = 22$) is cut-on at the three axial locations as in the experiments and is under-predicted by 14 dB. At 2BPF, the interaction mode ($m = -10$) and the rotor-locked mode ($m = 44$) observed are also cut-on at all three axial locations as in the experiments and are under-predicted by 19 dB and 28 dB, respectively. The cut-on mode amplitudes are all under-predicted, but recall that the uniform inflow assessment is performed to obtain a baseline against which to compare the effects of non-uniform inflow.

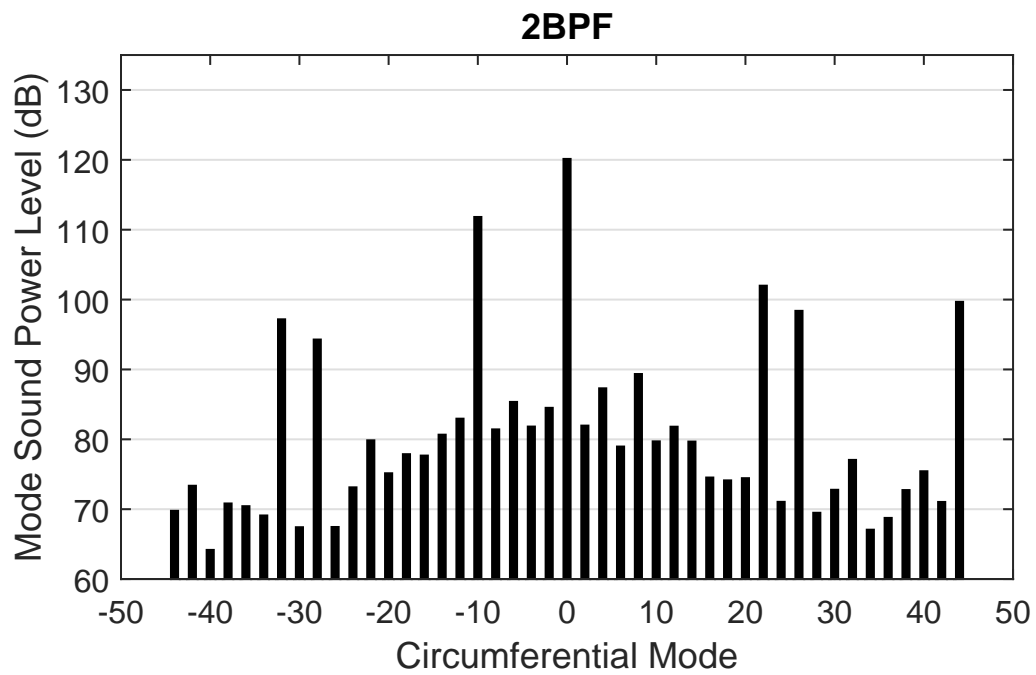
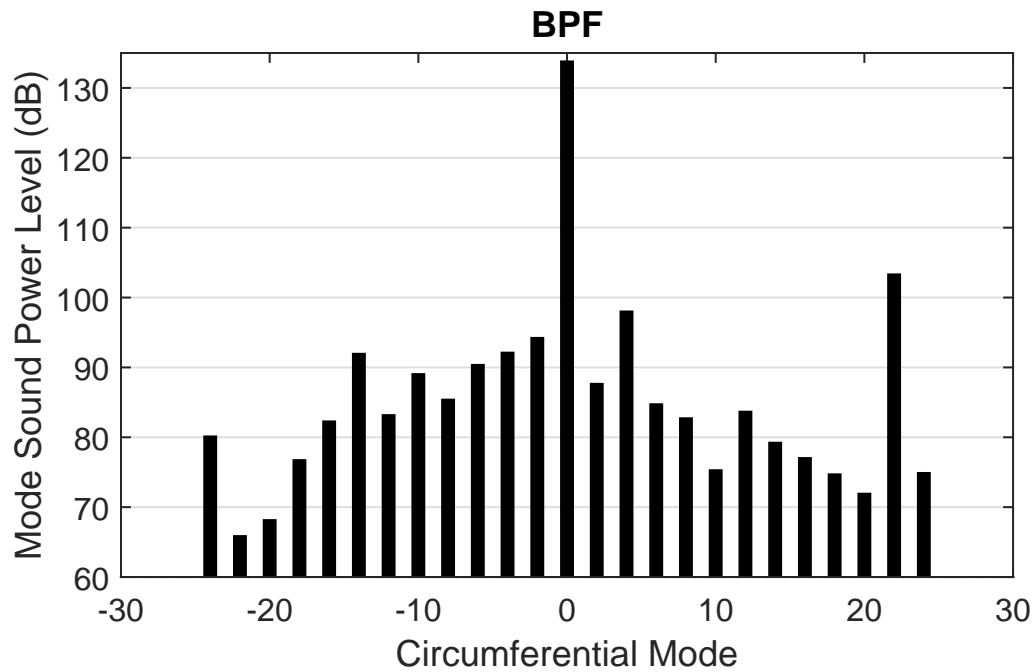


Figure 4-7: Modal sound power distribution $0.5c_{stator}$ downstream of stator trailing edge at takeoff.

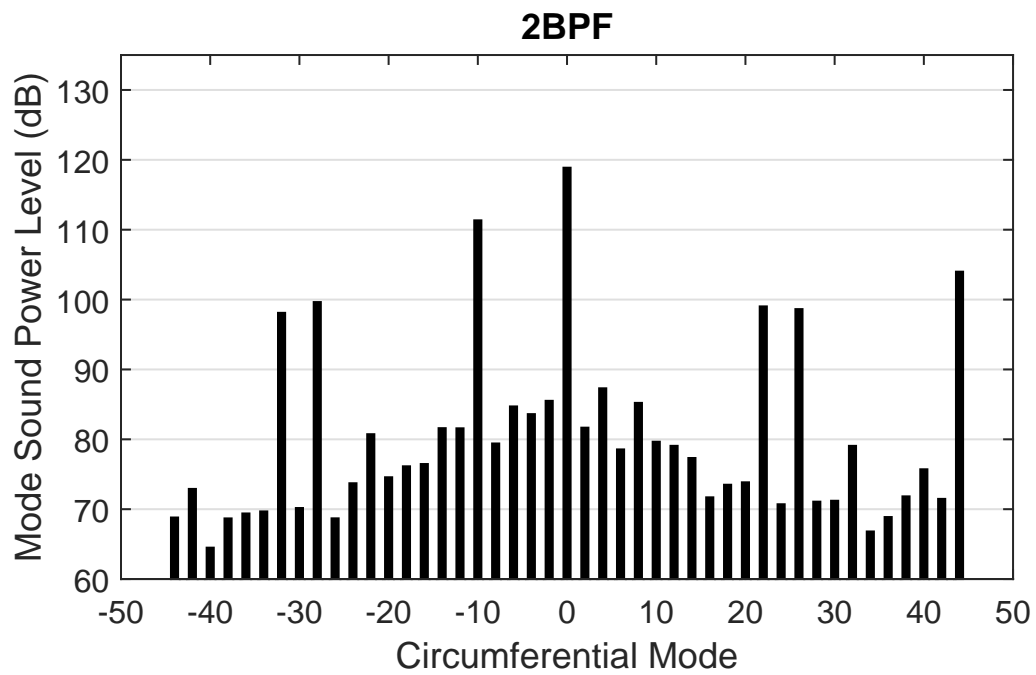
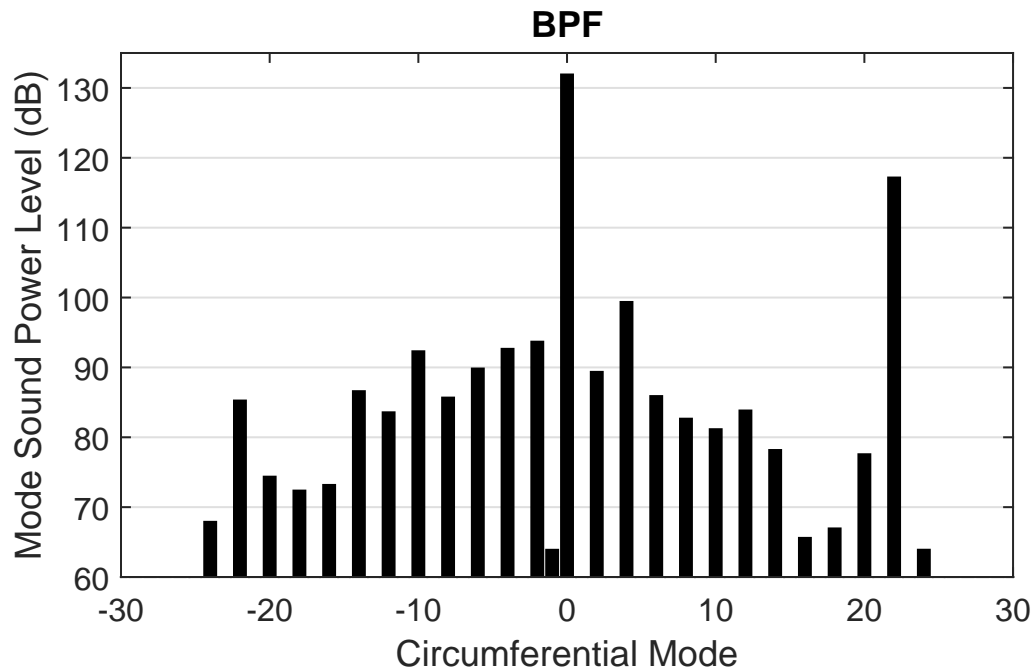


Figure 4-8: Modal power distribution $1.0c_{stator}$ downstream of stator trailing edge at takeoff.

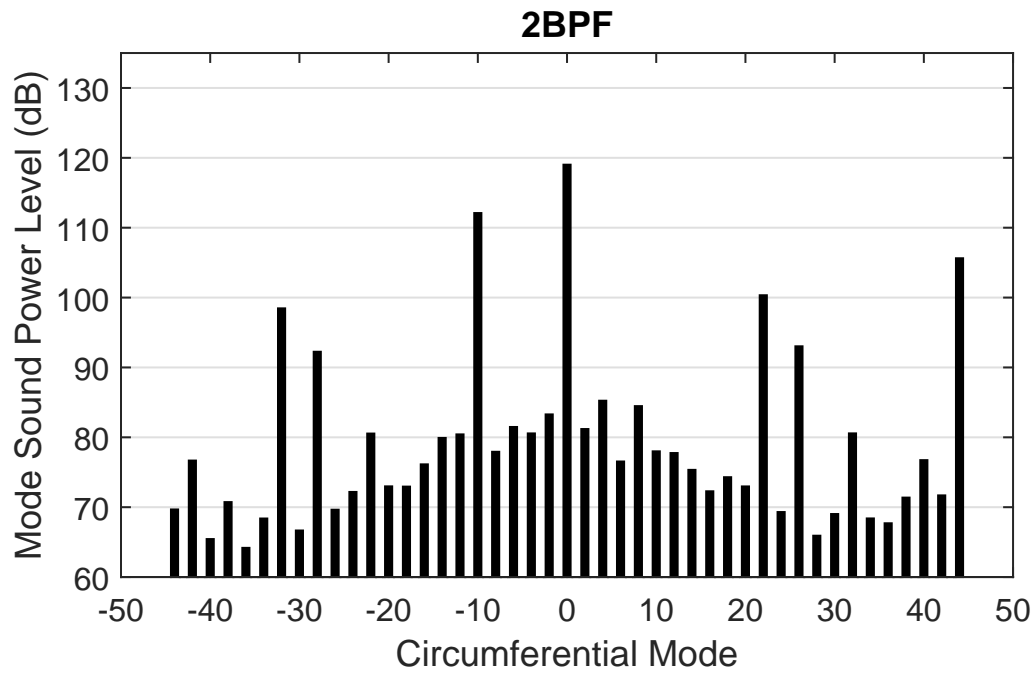
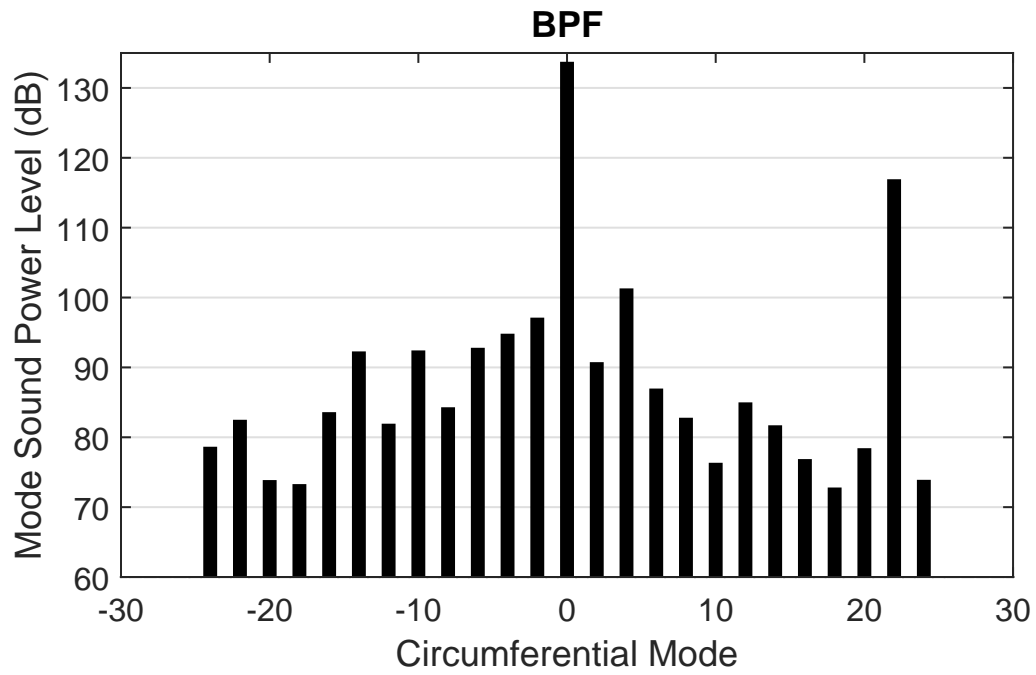


Figure 4-9: Modal power distribution $2.0c_{stator}$ downstream of stator trailing edge at takeoff.

On each of the modal distributions shown, there a strong plane wave mode ($m = 0$) cut-on. Using the moving-average analysis previously discussed, the amplitude of this plane wave mode at both BPF and 2BPF is found to be periodic. In the unsteady computations, the static pressure is set as the outlet boundary condition to achieve the desired corrected mass flow rate. As time advances, due to the imposed constant pressure at the outlet, there are small periodic changes in the mass flow due to compressibility effects. The strong artificial plane wave mode is thought to be a result of the changing mass flow rate of the unsteady computations.

At 2BPF, there are other modes in the sound power distributions that could be cut-on. These modes are $m = -32, -28, 22,$ and 26 . Equation 2.1 can be used to verify that at 2BPF, only $m = -10$ and 44 are Tyler-Sofrin modes. As stated earlier, these modes stand out due to the low noise floor in the computations caused by a lack of broadband noise. The sound powers of the $m = -28$ and 26 attenuate by about 8 dB between the one and two stator chord planes and the sound powers of the $m = -32$ and 22 increase by about 2 dB between the same two planes. Therefore, it is hypothesized that the $m = -28$ and 26 will continue to attenuate down the duct. It is not clear how $m = -32$ and 22 will behave down the duct or whether they correspond to something physical or not. To determine exactly how these modes will propagate down the duct, the data from noise propagation computation at the downstream acoustic measurement plane would be needed.

The overall under-prediction of the modal amplitudes indicates that there is an issue in the source generation rather than propagation. These amplitudes may be under-predicted due to the strength of the wake. The wake generated by the body forces has a lower strength than that of the experiments. The lower the rotor wake strength, the lower the acoustic strength of the fan wake and stator interaction. This applies for all acoustic modes of interest in this work. Aside from the strength being under-predicted, is it possible that the rate of mixing of the rotor wakes in the

computations is not correct. As the evolution of the rotor wakes in the experiments is not provided, the rotor wakes could only be matched at one location. Therefore one possible explanation for the under-prediction of the amplitudes is that the rotor wakes in the computations are mixing out more quickly than in the experiments. Another likely cause of under-prediction of the modal amplitudes is the assumed stator geometry. Though the presence of the assumed stator gave rise to the correct interaction mode, the amplitude of the mode could be under-predicted due to unknown discrepancies from the exact geometry.

Only even-numbered modes are predicted at the three planes considered. This is because the uniform inflow computation is performed on a half annulus grid and it is impossible to have an odd-numbered mode with a half annulus periodicity. The lack of odd-numbered modes is not of concern as for the blade counts considered in this work, it is impossible to have a odd-numbered turbomachinery mode of interest.

At the time of writing of this thesis, the third (acoustic propagation) domain computation is still underway. Once the final domain of the uniform inflow assessment is complete, the analysis of the accuracy can be performed again using the results at the acoustic measurement plane. The current analysis, however, is sufficient to indicate that downstream propagating rotor-stator interaction noise can be generated using the approach described in this thesis. Changes in the acoustics caused by circumferentially non-uniform background flow are expected to be well-captured since the wakes generated by body forces are dependent on the local flow. Knowing the difference between the computationally predicted and the experimental uniform inflow results, the effects of the non-uniform inflow can be assessed accurately on a relative basis.

4.3 Future Non-Uniform Inflow Computational Details

All of the ground work needed for the non-uniform inflow computation has been completed. No changes need to be made to the rotor body model as it depends on local flow conditions. The required grids have been generated and the non-uniform inflow profile has been determined, as will be discussed in this section.

The grid domains for the non-uniform inflow computation follow the split domain approach introduced in Section 3.6.1. The only difference between the uniform inflow computation grids and non-uniform inflow grids is that the latter are full annulus.

The non-uniform inflow profile to be used in this work is from Defoe [13]. Defoe's work includes contour plots for the non-dimensionalized stagnation pressure profile and the three components of the Mach number for the cutback flight condition as seen in Figures 4-10 and 4-11 to 4-13, respectively. Though the focus of the current work is at the takeoff flight condition, the flow profile at cutback will be sufficient as the resulting changes in the profiles between the two conditions is expected to be negligible since the flight Mach number is low for both conditions. These contour plots are digitized using Matlab 2014b [38] by comparing the colour of each pixel to the colour map and assigning a corresponding value.

Aside from the full annulus grid and the non-uniform inflow profile, no other aspect of the computational details from the uniform inflow assessment will be changed. This will allow an accurate analysis of the effect of non-uniform flow on the interaction noise.

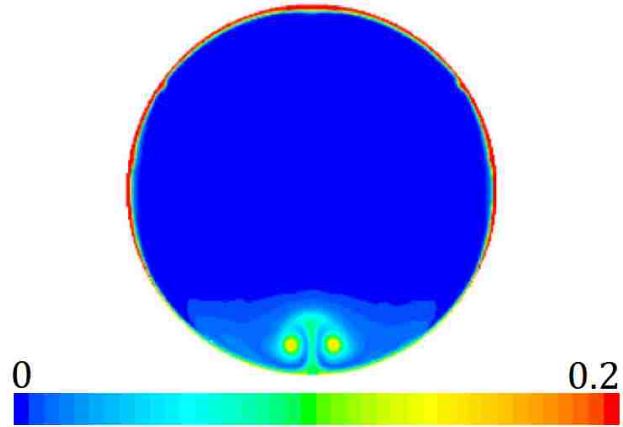


Figure 4-10: Non-dimensionalized stagnation pressure profile $\left(\frac{p_{t,\infty}-p_t}{p_t-p}\right)$ for the non-uniform inflow computation (from Defoe [13]).

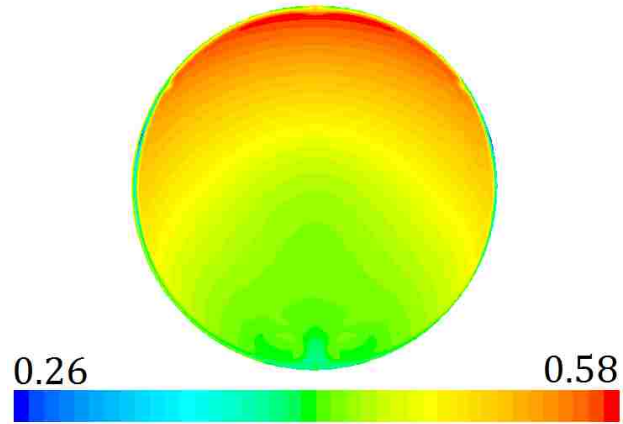


Figure 4-11: Axial Mach number profile for the non-uniform inflow computation (from Defoe [13]).

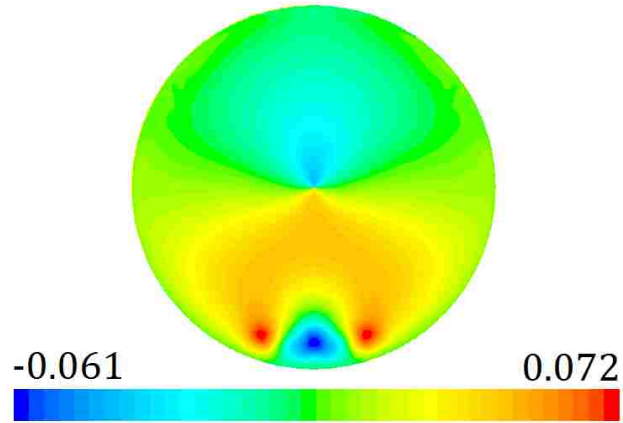


Figure 4-12: Radial Mach number profile for the non-uniform inflow computation (from Defoe [13]).

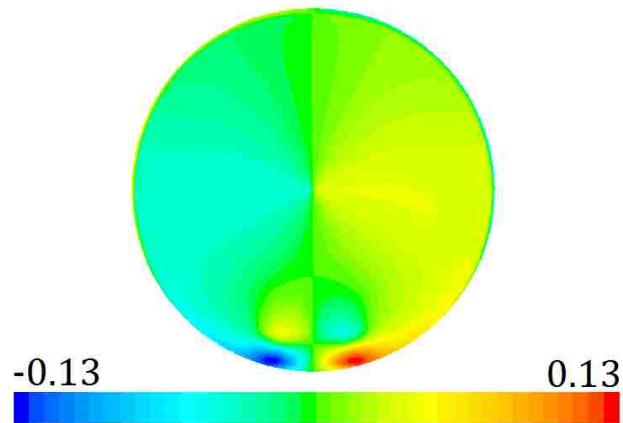


Figure 4-13: Tangential Mach number profile for the non-uniform inflow computation (from Defoe [13]).

Chapter 5

Conclusions and Future Work

The objectives of this thesis are to determine whether body forces can be used to generate rotor-stator interaction noise, assess the accuracy of the approach, and determine whether the approach is suitable for analyzing the non-uniform flow effects. A body-force-based fan model was used in an internal flow aeroacoustic computation with uniform inflow in an attempt to meet these objectives. This chapter provides a summary of the work, the limitations of the body force approach, the key outcomes, and suggestions for future work.

5.1 Summary

A new approach to predicting downstream propagating tonal rotor-stator interaction noise has been developed with the use of a body-force-based fan model. This approach is assessed using experimental data from the NASA Source Diagnostic Test.

The body force model for this work is adapted from previous work. Changes are successfully made to improve the performance of the blade row model. The efficiency is corrected by modifying the viscous (loss-generating) force to capture typical turbomachinery loss bucket behavior. To obtain stagnation pressure and temperature ratios in agreement with the experimental data, the blade camber profiles

are adjusted. Lastly, for the generation of interaction noise, the rotor wakes are produced via local concentrations of viscous forces at each blade trailing edge.

The computed overall aerodynamic performance is within 1% of the experimental stagnation pressure ratio, stagnation temperature ratio, and adiabatic efficiency measured for the GE R4 rotor from the SDT. Rotor wakes are successfully generated by the body forces and are found to slightly under-predict wake strength while producing wake widths in excellent agreement with the experimental aerodynamic results.

For the generation of rotor-stator interaction noise, stator geometry is approximated based on limited information available in the literature. The approximated geometry yields a stagnation pressure ratio of 0.993 compared to an experimental value of 0.988.

A method of splitting the computation domain is developed to reduce the computational cost of a single computation based on limited available resources. This consists of splitting the domain into an inlet flow section, a source generation section, and a noise propagation section. The inlet flow section is used to determine the flow profiles at one chord upstream of the rotor inlet. The rotor-stator interaction noise source generation occurs in the second domain and the noise is propagated to the acoustic measurement plane in the final domain. The approach is validated on a similar two-dimensional case for which split simulations yield sound power levels that are within 0.3 dB of the full-domain simulation.

Due to unforeseen downtime on the computational network (Sharcnet), only the inlet flow and the source generation computations are completed. This allows for the interaction noise to be predicted upstream of the acoustic measurement plane. However, an analysis showed that, though amplitudes are under-predicted by 14-28 dB, the correct cut-on modes are produced by the computations. In addition, modal and overall sound power levels are near-constant beyond one chord downstream of the stator trailing edge, enabling a direct assessment of the available computational results

and the experimental data. With the difference known between the computations and the experiment, the future non-uniform calculations can be completed to determine the effects of non-uniform inflow on interaction noise.

5.2 Key Outcomes and Conclusions

From the results, it can be concluded that computationally cost-saving body forces can be used to generate discrete rotor wakes that give rise to rotor-stator interaction tones. The approach enables a cost savings of approximately 42%. The accuracy of the amplitudes of the predicted interaction tones depend heavily on the accuracy of the rotor wake strength and shape, accurate mixing of rotor wakes, and the detailed stator geometry. An artificial plane wave mode is predicted from the computations that is hypothesized to be caused by the small changes in mass flow due to the imposed outlet static pressure. Other modes predicted by the computations are expected to attenuate down the duct but cannot be verified until the noise propagation computation is completed.

As the body-force-based rotor model depends on local flow conditions, the effects of circumferentially non-uniform flow on the downstream propagating interaction tones can be determined using this approach. Overall, all objectives set out by this thesis have been met in full.

5.3 Future Work

In the near future, the following work will be completed, though it does not form part of the current thesis:

1. The noise propagation section of the uniform inflow computation will be completed and differences between the experimental and numerical results will be

assessed. If the computational results at the acoustic measurement plane are similar to the current results at 2 stator chords downstream of the stator trailing edge, it does not necessarily mean the acoustic propagation computation can be dropped in the non-uniform flow case. This is because the behavior of different acoustic modes and their amplitudes during propagation through a non-uniform mean flow is of interest.

2. The non-uniform inflow computation will be completed and the effects of the non-uniform flow on the generation and propagation of tonal rotor-stator interaction noise will be assessed.

Bibliography

- [1] Pratt & Whitney. PurePower Engine Family Specifications Chart. Online. Accessed on October 7, 2015 at https://www.pw.utc.com/Content/Press_Kits/pdf/ce_pw1500g_pCard.pdf.
- [2] Hileman, J., Spakovszky, Z., Drela, M., Sargeant, M. Airframe design for "Silent Aircraft". *Presented at the 45th AIAA Aerospace Sciences Meeting and Exhibit*, AIAA Paper 2007-453, 2007.
- [3] Drela, M. Development of the D8 Transport Configuration. *Presented at the 29th AIAA Applied Aerodynamics Conference*, AIAA Paper 2011-3970, 2011.
- [4] Hughes, C., Jeracki, R., Woodward, R., Miller, C. Fan Noise Source Diagnostic Test - Rotor Alone Aerodynamic Performance Results. *Presented at the 8th AIAA/CEAS Aeroacoustics Conference*, AIAA Paper 2002-2426, 2002.
- [5] Hughes, C. Aerodynamic Performance of Scale-Model Turbofan Outlet Guide Vanes Designed for Low Noise. *Presented at the 40th AIAA Aerospace Sciences Meeting and Exhibit*, AIAA Paper 2002-0374, 2002.
- [6] Torregrosa, A. J., Fajardo, P., Gil, A., Navarro, R. Development of Non-Reflecting Boundary Condition for Application in 3D Computational Fluid Dynamics Codes. *Engineering Applications of Computations Fluid Mechanics*, 6 (3), pp. 447-460, 2012.

- [7] Tyler, J. M., Sofrin, T. G. Axial Flow Compressor Noise Studies. *SAE Technical Papers*, 1962.
- [8] Heidelberg, L. J. Fan Noise Source Diagnostic Test - Tone Modal Structure Results. *Presented at the 8th AIAA/CEAS Aeroacoustics Conference*, AIAA Paper 2002-2428, 2002.
- [9] Liu, H., Ouyang, H., Wu, Y., Tian, J., Du, Z. An analytical investigation of stator lean on rotor-stator interaction noise. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, 229 (1), pp. 96-109, 2015.
- [10] Rumsey, C. Computation of Acoustic Waves Through Sliding-Zone Interfaces Using an Euler/Navier-Stokes Code. *Presented at the 2nd AIAA/CEAS Aeroacoustics Conference*, AIAA Paper 1996-1752, 1996.
- [11] Rumsey, C., Biedron, R., Farassat, F. Ducted-Fan Engine Acoustic Predictions Using a Navier-Stokes Code. *Journal of Sound and Vibration*, 213(4):643-664, 1998.
- [12] Chen, X., Zhang, X., Morfey, C., Nelson, P. A Numerical Method for Computation of Sound Radiation from an Unflanged Duct. *Journal of Sound and Vibration*, 270:573-586, 2004.
- [13] Defoe, J. *Inlet Swirl Distortion Effects on the Generation and Propagation of Fan Rotor Shock Noise*. Ph.D Thesis, MIT, September 2011.
- [14] Sharma, A., Richards, S. K., Wood, T. H., Shieh, C. M. Numerical Prediction of Exhaust Fan Tone Noise from High Bypass Aircraft Engines. *Presented at the 13th AIAA/CEAS Aeroacoustics Conference*, AIAA Paper 2007-3700, 2007.

- [15] Sharma, A., Chen, H., Shieh, C. M. Linearized Navier-Stokes Analysis for Rotor-Stator Interaction Tone Noise Prediction. *Presented at the 16th AIAA/CEAS Aeroacoustics Conference*, AIAA Paper 2010-3744, 2010.
- [16] Marble, F. *Three-dimensional flow in turbomachines*, volume X of *High Speed Aerodynamics and Jet Propulsion*, Hawthorne, W. R., ed. Princeton University Press, Princeton, NJ, pp. 83-166 edition, 1964.
- [17] Gong, Y. *A Computational Model for Rotating Stall and Inlet Distortions in Multistage Compressors*. Ph.D. Thesis, MIT, Department of Aeronautics and Astronautics, March 1999.
- [18] Lieblein, S. *Experimental Flow in Two-Dimensional Cascades*. Number SP-36. 1965.
- [19] Defoe, J., Narkaj, A., Spakovszky, Z. A Novel MPT Noise Prediction Methodology for Highly-Integrated Propulsion Systems with Inlet Flow Distortion. *Presented at the 15th AIAA/CEAS Aeroacoustics Conference*, AIAA Paper 2009-3366, 2009.
- [20] Defoe, J., Narkaj, A., Spakovszky, Z. A Body Force-Based Method for Prediction of Multiple-Pure-Tone Noise: Validation. *Presented at the 16th AIAA/CEAS Aeroacoustics Conference*, AIAA Paper 2010-3747, 2010.
- [21] Defoe, J., Spakovszky, Z. Effects of Boundary-Layer Ingestion on the Aeroacoustics of Transonic Fan Rotors. *Journal of Turbomachinery*, 135(3):051013, 2013.
- [22] Peters, A. *Ultra-Short Nacelles for Low Fan Pressure Ratio Propulsors*. Ph.D Thesis, MIT, Department of Aeronautics and Astronautics, February 2014.

- [23] Candel, S. and Poinso, T. A Tutorial on Acoustics. Laboratoire d’Énergétique Moléculaire et Macroscopique, Combustion, CNRS, Ecole Central des Art et Manufactures, Chatenay-Malabry, France, 1987.
- [24] Sutliff, D. Rotating Rake Turbofan Duct Mode Measurement System. NASA/TM-2005-2138328, 2005.
- [25] Abramowitz, M., Stegun, I.A. Handbook of Mathematical Functions. *National Bureau of Standards, Applied Math. Series #55*, Dover Publications. Sections 9.1.1, 9.1.89, and 9.12. Formulas 9.1.10 and 9.2.5. 1965.
- [26] Myers, M. Transport of Energy by Disturbances in Arbitrary Steady Flows. *Journal of Fluid Mechanics*, 226:383-400, 1991.
- [27] Morfey, C. Acoustic Energy in Non-Uniform Flows. *Journal of Sound and Vibration*, 14(2):159–170, 1971.
- [28] Freund, J. Proposed Inflow/Outflow Boundary Condition for Direct Computation of Aerodynamic Sound. *AIAA Journal*, 35(4):740-742, 1997.
- [29] Si, H. Shen, W., Zhu, W. Effect of non-uniform mean flow field on acoustic propagation problems in computational aeroacoustics. *Aerospace Science and Technology*, 28(1):145–153, 2013.
- [30] Podboy, G., Krupar, M., Helland, S., Hughes, C. Steady and Unsteady Flow Field Measurements Within a NASA 22-Inch Fan Model. *Presented at the 40th AIAA Aerospace Sciences Meeting and Exhibit*, AIAA Paper 2002-1033, 2002.
- [31] Ansys, Inc., Canonsburg, PA. *ANSYS FLUENT User’s Guide*, 15.0 edition, 2013.
- [32] Spalart, P., Allmaras, S. A One-Equation Turbulence Model for Aerodynamic Flows. *Technical Report*, AIAA Paper 1992-0439, 1992.

- [33] Gunn, E. J., Hall, C. A. Aerodynamics of Boundary Layer Ingesting Fans. *ASME Turbo Expo 2014: Turbine Technical Conference and Exposition*. American Society of Mechanical Engineers, 2014.
- [34] Dassault Systèmes SolidWorks Corporation, Waltham, MA. *SolidWorks User's Guide*, 2014 edition, 2014.
- [35] Drela, M. *A Users Guide to MISES 2.63*. MIT Aerospace Computational Design Laboratory, 2008.
- [36] Raj, R., Lakshminarayana, B. On the Investigation of Cascade and Turbomachinery Rotor Wake Characteristics. *Prepared for the National Aeronautics and Space Administration*, NASA CR-134680, 1975.
- [37] SHARCNET. Notice Board for Global B. Online. Accessed on July 8, 2016 at https://www.sharcnet.ca/my/systems/full_status/72.
- [38] MathWorks, Inc., Natick, MA. *MathWorks MATLAB Documentation*, 2014b edition, 2014.

Appendix A

Body Force Model Fluent

User-Defined Function

```
/* **** */
/* Fluent UDF which adds source terms to modify the Euler equations to
   model the presence of a blade row with body forces. */
   /* Alex Narkaj and Jeff Defoe */
   /* July/2008–March/2010 */
   /* later edits by Jeff Defoe */
   /* March/2010–October/2010 */
   /* later edits by Krishna Patel */
   /* March/2015–August/2016 */

/* THIS VERSION WORKS WITH THE DENSITY-BASED SOLVER IN FLUENT. */
/* Buffer Zone sizes for Source Generation Computation */
/* **** */

#include "udf.h"

#define omega -1325.4 /* rad/sec — needs to be negative for kn/kp and
   negative for free-vortex fan */
#define P_OP 0 /* operating pressure, must be set equal to value used in
   GUI */
#define Nblade 22 /* number of blades */
#define Kp1 0.0336 /* constant Kp1 */
#define Kp2 0.6321 /* constant Kp2 */
#define Mref 1.0052 /* reference Mach number for loss model (relative
   Mach number at design) */
#define Mrel 1.0052 /* reference relative Mach number at design */

#define pert_const 300 /* perturbation constant */
```

```

#define dsxstart 0.4671502411 /* inside boundary of downstream damping
zone */
#define dsxbound 0.4745351374 /* outside boundary of downstream damping
zone */

#define duxstart 0.07737116516 /* inside boundary of upstream damping
zone */
#define duxbound 0.0700000003 /* outside boundary of upstream damping
zone */

#define beta_sigma_in 4.0 /* parameters for buffer zones */
#define beta_sigma_out 4.0

#define pudm 0 /* pressure udm */
#define uudm 1 /* x-velocity udm */
#define vudm 2 /* y-velocity udm */
#define wudm 3 /* z-velocity udm */
#define rudm 4 /* density udm */
#define hudm 5 /* enthalpy udm */
#define tudm 6 /* temperature udm */
#define cudm 7 /* speed of sound udm */

#define xudm 8 /* steady x-bodyforce udm */
#define yudm 9 /* steady y-bodyforce udm */
#define zudm 10 /* steady z-bodyforce udm */
#define eudm 11 /* steady e-bodyforce udm */

#define xudm2 12 /* unsteady x-bodyforce udm */

#define rludm 13 /* r/L value for buffer zones udm */

#define BFvudm 14 /* viscous bodyforce udm */
#define BFnudm 15 /* normal bodyforce udm */
#define BFrudm 16 /* radial bodyforce udm */
#define PGxudm 17 /* x pressure gradient udm */
#define PGnudm 18 /* normal pressure gradient udm */

#define th_shiftudm 19 /* unsteady theta shift udm */

/*****
/* File-wide variable declaration section */
*****/

static long unsigned int Nsteps = 1; /* number of time-steps used to
averaging so far */

/*****
/* Function declaration and utility function section */
*****/

double calcr1(real x[]);

/* This version uses many UDMs... they are all named below for
convenience */

```

```

/*****
/* Buffer zone section */
*****/

DEFINE_ON_DEMAND(steadyfieldbuffer)
{
/* This UDF gets the steady flow field and stores it to user-defined
memory */

    real gamma = 1.4, cv;
    real pref, rhoref;

    #if !RP_HOST
    Domain *d;
    Thread *t;
    cell_t c;

    pref = 101325;
    rhoref = 1.225;

    d = Get_Domain(1); /* returns fluid domain pointer */

    #endif

    node_to_host_int_1(Nsteps); /* Send the current Nsteps value to
the host process */

    #if !RP_NODE

    Message("Getting value of Nsteps counter. Previous value was
Nsteps = %lu\n...\n",Nsteps);
    Nsteps = RP_Get_Integer("nsteps");
    Message("Nsteps has been set. Nsteps = %lu\n",Nsteps);

    #endif

    host_to_node_int_1(Nsteps); /* distribute Scheme variables to
compute nodes */

    #if !RP_NODE
    if(Nsteps == 1)
        {Message("Storing flow field to UDMs...\n");}
    else
        {Message("Nsteps > 1... not storing anything!\n");}
    #endif

    #if !RP_HOST

    if(Nsteps == 1)
    {
        thread_loop_c(t,d) /* Loop through all cells in domain:
*/

```



```

    {
        begin_c_loop_int(c,t)
        {
            C_UDMI(c,t,pudm)=C_P(c,t); /* pressure
            */
            C_UDMI(c,t,uudm)=C_U(c,t); /* x-velocity
            */
            C_UDMI(c,t,vudm)=C_V(c,t); /* y-velocity
            */
            #if RP_3D
            C_UDMI(c,t,wudm)=C_W(c,t); /* z-velocity
            */
            #endif
            C_UDMI(c,t,rudm)=C_R(c,t); /* density */
            C_UDMI(c,t,hudm)=C_H(c,t); /* enthalpy
            */
            C_UDMI(c,t,tudm)=C_T(c,t); /*
            temperature */

            C_UDMI(c,t,cudm)=sqrt(gamma*C_T(c,t)*
            C_RGAS(c,t)); /* speed of sound */
        }
        end_c_loop_int(c,t)
    }
}
#endif
}

```

```

DEFINE_ON_DEMAND(rdivl)
{
    /* This UDF does a purely geometrical calculation, getting the fraction
    of the distance
    along the buffer zone (r/L) for each cell in the buffer zones.

    The buffer zones go from +/- 3 to 8 m */

    /* DEFINE VARIABLES USEABLE IN ALL PROCESSES */
    real x[ND_ND];
    int n;

    /* DEFINE VARIABLES ONLY USEABLE FOR NODE PROCESS */
    #if !RP_HOST
    Domain *d;
    Thread *t;
    Thread *tf;
    cell_t c;
    #endif /* !RP_HOST */

    #if !RP_HOST
    d = Get_Domain(1); /* returns fluid domain pointer */

    /* Loop through all cells in the buffer zones (or threads): */
    thread_loop_c(t,d)
    {

```

```

        begin_c_loop_int(c,t)
        {
            C_UDMI(c,t,rludm) = 0.0; /* initialize to 0 */
            C_CENTROID(x,c,t);
            C_UDMI(c,t,rludm) = calcul(x); /* get r/L */
            if(C_UDMI(c,t,rludm) > 1.0)
                {C_UDMI(c,t,rludm) = 1.0;} /* cannot
                    have a value greater than 1 */
        }
    end_c_loop_int(c,t)
}
#endif /* !RP_HOST */
}

double calcul(real x[ND_ND])
{
    real rl = 0.0; /* variable to be returned */

    #if !RP_HOST

    if(x[0] >= dsxstart) /* downstream buffer zone */
    {
        rl = (x[0]-dsxstart)/(dsxbound-dsxstart);
    }
    else if(x[0] <= duxstart) /* upstream buffer zone */
    {
        rl = (duxstart-x[0])/(duxstart-duxbound);
    }
    else /* not in a buffer zone */
    {
        rl = 0.0;
    }

    #endif

    return rl; /* return the ratio */
}

DEFINE_ON_DEMAND(rdivlreset)
{
    /* This UDF resets r/l values. */

    /* DEFINE VARIABLES ONLY USEABLE FOR NODE PROCESS */
    #if !RP_HOST
    Domain *d;
    Thread *t;
    cell_t c;
    face_t f;
    #endif /* !RP_HOST */

    #if !RP_HOST
    d = Get_Domain(1); /* returns fluid domain pointer */

    /* Loop through domain */

```

```

thread_loop_c(t,d)
{
    begin_c_loop_int(c,t)
    {
        C_UDMI(c,t,rludm) = 0.0; /* initialize to 0 */
    }
    end_c_loop_int(c,t)

    begin_f_loop(f,t)
    {
        F_UDMI(c,t,rludm) = 0.0;
    }
    end_f_loop(f,t)
}
#endif /* !RP_HOST */
}

/* buffer UDFs - source terms - artificial damping */
DEFINE_SOURCE(buffer_m_out_sigma,c,t,dS,eqn)
{
    #if !RP_HOST
    real U, Uterm, sigma, sigma0in, sigma0out, sigmaterm, a, U0in,
        U0out, lambda, f, fin, fout, gradrho1;
    real gamma = 1.4;

    a = sqrt(gamma*300.0*C_RGAS(c,t));
    f = fabs(omega)/2.0/M_PI;
    lambda = a/f;

    sigma0out = C1out *a/lambda;
    sigma = sigma0out*pow(sin(C_UDMI(c,t,rludm)),beta_sigma_out);

    sigmaterm = sigma*(C_R(c,t)-C_UDMI(c,t,rudm));

    dS[eqn] = 0.0;

    return (-sigmaterm);
    #endif
}

DEFINE_SOURCE(buffer_x_out_sigma,c,t,dS,eqn)
{
    #if !RP_HOST
    real U, Uterm, sigma, sigma0in, sigma0out, sigmaterm, a, U0in,
        U0out, lambda, f, fin, fout, gradrho1;
    real rhou[ND_ND];
    real gamma = 1.4;
    rhou[0] = C_R_RG(c,t)[0]*C_U(c,t)+C_R(c,t)*C_U_RG(c,t)[0];
    rhou[1] = C_R_RG(c,t)[1]*C_U(c,t)+C_R(c,t)*C_U_RG(c,t)[1];
    rhou[2] = C_R_RG(c,t)[2]*C_U(c,t)+C_R(c,t)*C_U_RG(c,t)[2];

    a = sqrt(gamma*300.0*C_RGAS(c,t));
    f = fabs(omega)/2.0/M_PI;
    lambda = a/f;

```

```

sigma0out = C1out*a/lambda;
sigma = sigma0out*pow(sin(C_UDMI(c,t,rludm)),beta_sigma_out);

sigmaterm = sigma*(C_R(c,t)*C_U(c,t)-C_UDMI(c,t,rudm)*C_UDMI(c,t
,uudm));

dS[eqn] = 0.0;

return (-sigmaterm);
#endif
}

DEFINE_SOURCE(buffer_y_out_sigma,c,t,dS,eqn)
{
#if !RP_HOST
real U, Uterm, sigma, sigma0in, sigma0out, sigmaterm, a, U0in,
U0out, lambda, f, fin, fout, gradrhov1;
real rhov[ND_ND];
real gamma = 1.4;
rhov[0] = C_R_RG(c,t)[0]*C_V(c,t)+C_R(c,t)*C_V_RG(c,t)[0];
rhov[1] = C_R_RG(c,t)[1]*C_V(c,t)+C_R(c,t)*C_V_RG(c,t)[1];
rhov[2] = C_R_RG(c,t)[2]*C_V(c,t)+C_R(c,t)*C_V_RG(c,t)[2];

a = sqrt(gamma*300.0*C_RGAS(c,t));
f = fabs(omega)/2.0/M_PI;
lambda = a/f;

sigma0out = C1out*a/lambda;
sigma = sigma0out*pow(sin(C_UDMI(c,t,rludm)),beta_sigma_out);

sigmaterm = sigma*(C_R(c,t)*C_V(c,t)-C_UDMI(c,t,rudm)*C_UDMI(c,t
,vudm));

dS[eqn] = 0.0;

return (-sigmaterm);
#endif
}

DEFINE_SOURCE(buffer_z_out_sigma,c,t,dS,eqn)
{
#if !RP_HOST
real U, Uterm, sigma, sigma0in, sigma0out, sigmaterm, a, U0in,
U0out, lambda, f, fin, fout, gradrhov1;
real rhov[ND_ND];
real gamma = 1.4;
rhov[0] = C_R_RG(c,t)[0]*C_W(c,t)+C_R(c,t)*C_W_RG(c,t)[0];
rhov[1] = C_R_RG(c,t)[1]*C_W(c,t)+C_R(c,t)*C_W_RG(c,t)[1];
rhov[2] = C_R_RG(c,t)[2]*C_W(c,t)+C_R(c,t)*C_W_RG(c,t)[2];

a = sqrt(gamma*300.0*C_RGAS(c,t));
f = fabs(omega)/2.0/M_PI;
lambda = a/f;

```

```

sigma0out = C1out*a/lambda;
sigma = sigma0out*pow(sin(C_UDMI(c,t,rludm)),beta_sigma_out);

sigmaterm = sigma*(C_R(c,t)*C_W(c,t)-C_UDMI(c,t,rudm)*C_UDMI(c,t,
,wudm));

dS[eqn] = 0.0;

return (-sigmaterm);
#endif
}

DEFINE_SOURCE(buffer_e_out_sigma,c,t,dS,eqn)
{
#if !RP_HOST
real U, Uterm, sigma, sigma0in, sigma0out, sigmaterm, a, U0in,
U0out, lambda, f, fin, fout, gradel;
real e, ess, grade[ND_ND], gradp[ND_ND];
real gamma = 1.4;

gradp[0] = C_RGAS(c,t)*(C_R(c,t)*C_T_RG(c,t)[0]+C_T(c,t)*C_R_RG(
c,t)[0]); /* dP/dx */
gradp[1] = C_RGAS(c,t)*(C_R(c,t)*C_T_RG(c,t)[1]+C_T(c,t)*C_R_RG(
c,t)[1]); /* dP/dy */
gradp[2] = C_RGAS(c,t)*(C_R(c,t)*C_T_RG(c,t)[2]+C_T(c,t)*C_R_RG(
c,t)[2]); /* dP/dz */

e = C_R(c,t)*(C_H(c,t)+(1.0/2.0)*(C_U(c,t)*C_U(c,t)+C_V(c,t)*C_V
(c,t)+C_W(c,t)*C_W(c,t))-C_P(c,t)/C_R(c,t));
ess = C_UDMI(c,t,rudm)*(C_UDMI(c,t,hudm)+(1.0/2.0)*(C_UDMI(c,t,
uudm)*C_UDMI(c,t,uudm)+C_UDMI(c,t,vudm)*C_UDMI(c,t,vudm)+
C_UDMI(c,t,wudm)*C_UDMI(c,t,wudm))-C_UDMI(c,t,pudm)/C_UDMI(c
,t,rudm));
grade[0] = C_R_RG(c,t)[0]*e+C_R(c,t)*(C_H_RG(c,t)[0]+C_U(c,t)*
C_U_RG(c,t)[0]+C_V(c,t)*C_V_RG(c,t)[0]+C_W(c,t)*C_W_RG(c,t)
[0]-gradp[0]/C_R(c,t)+C_P(c,t)*C_R_RG(c,t)[0]/C_R(c,t)/C_R(c
,t));
grade[1] = C_R_RG(c,t)[1]*e+C_R(c,t)*(C_H_RG(c,t)[1]+C_U(c,t)*
C_U_RG(c,t)[1]+C_V(c,t)*C_V_RG(c,t)[1]+C_W(c,t)*C_W_RG(c,t)
[1]-gradp[1]/C_R(c,t)+C_P(c,t)*C_R_RG(c,t)[1]/C_R(c,t)/C_R(c
,t));
grade[2] = C_R_RG(c,t)[2]*e+C_R(c,t)*(C_H_RG(c,t)[2]+C_U(c,t)*
C_U_RG(c,t)[2]+C_V(c,t)*C_V_RG(c,t)[2]+C_W(c,t)*C_W_RG(c,t)
[2]-gradp[2]/C_R(c,t)+C_P(c,t)*C_R_RG(c,t)[2]/C_R(c,t)/C_R(c
,t));

a = sqrt(gamma*300.0*C_RGAS(c,t));
f = fabs(omega)/2.0/M_PI;
lambda = a/f;

sigma0out = C1out*a/lambda;
sigma = sigma0out*pow(sin(C_UDMI(c,t,rludm)),beta_sigma_out);

```

```

        sigmaterm = sigma*(e-ess);

        dS[eqn] = 0.0;

        return (-sigmaterm);
    #endif
}

/*****
/* Body Forces Section */
*****/

/* Steady version */
DEFINE_SOURCE(x_bf_knkp, c, t, dS, eqn)
{
    #if !RP_HOST
        real source;
        int condition1, condition2, condition3, i;
        real M1rel, xle35;
        double phi, thetau, th_shift;

        /* definition of geometry and flow variables */

        real xc[ND_ND]; /* ND_ND = 3 for 3D, 2 for 2D */
        C_CENTROID(xc, c, t); /* returns cell centroid array */
        real x = xc[0]; /* Cell xyz coordinates */
        real y = xc[1];
        real z = xc[2];
        real r = sqrt(y*y+z*z); /* Cell radial coordinate */
        real theta = atan2(z, y); /* Cell theta coordinate */
        real sinq = z/r; /* sin(theta) */
        real cosq = y/r; /* cos(theta) */
        real Uwheel = r*omega; /* Wheel speed in m/s */

        real rho = C_R(c, t); /* Cell Density */
        real p = C_P(c, t); /* Cell Pressure */
        real T = C_T(c, t); /* Cell Temperature */
        real u = C_U(c, t);
        real v = C_V(c, t); /* Cell xyz velocities */
        real w = C_W(c, t);
        real Rgas = C_RGAS(c, t);
        real gamma = 1.4;

        real Vr = v*cosq+w*sinq; /* Radial Velocity */
        real Vt = -v*sinq+w*cosq; /* Tangential Velocity */
        real Vtrel = Vt-Uwheel; /* Rel. Tang. Velocity */
        real Vrel = sqrt(u*u+Vr*Vr+Vtrel*Vtrel); /* Relative Vel. Mag. */
        real alpha = (1.63+0.46*r-14.3*x+54.9*r*x-12.2*r*r+0.4*x*x)
            +(5.799-64.4*x-38.11*r+139.1*x*x+423.4*x*r+32.26*r*r-765.5*x*x*r
            -413.4*x*r*r+35.42*r*r*r); /* Local Metal Angle */
        real sol = 51.235*r*r-28.57*r+4.888; /* solidity, linear from hub
            (1.77) to tip (1.29) */
        real Kn = (4.2-3.3*alpha)*(-89.3572*r*r+39.3732*r-2.0773);
    #endif
}

```

```

    real gap = 2*M_PI*r*cos(alpha)*sqrt(sol)/Nblade; /* Staggered Spacing,
        solidity^(1-Ncarter), Ncarter=0.5 */

/* definition of gradients in y,z */

    real rho_x = C_R_RG(c,t)[0]; /* d(rho)/dx */
    real rho_y = C_R_RG(c,t)[1]; /* d(rho)/dy */
    real rho_z = C_R_RG(c,t)[2]; /* d(rho)/dz */
    real T_x = C_T_RG(c,t)[0]; /* dT/dx */
    real T_y = C_T_RG(c,t)[1]; /* dT/dy */
    real T_z = C_T_RG(c,t)[2]; /* dT/dz */
    real p_x = Rgas*(rho*T_x+T*rho_x); /* dP/dx */
    real p_y = Rgas*(rho*T_y+T*rho_y); /* dP/dy */
    real p_z = Rgas*(rho*T_z+T*rho_z); /* dP/dz */

/* Compute Body Forces (x,r,theta), convert to (x,y,z) */

/* 1. Viscous Body Force */
    real BFx = -(Kp1+Kp2*(Mrel-Mref))*(Mrel-Mref)*Vrel*u/gap;

/* 2. Turning Force */
    real BFn = Kn*(Vtrel*cos(alpha)-u*sin(alpha))*(u*cos(alpha)+Vtrel*sin(
        alpha))/gap;
    BFx = BFx+BFn*Vtrel/Vrel;

/* 3. BF due to Pressure Gradient in Staggered Passage */
    real PG_n = p_x*sin(alpha)/rho; /* pressure gradient N/m^3 */
    real PG_x = PG_n*sin(alpha);

/* Source Term */

    source = rho*BFx+PG_x; /* source in N/m^3 */

    C_UDMI(c,t,xudm)=BFx; /* store the steady body force */

    dS[eqn] = 0.0;

    return source;
        #endif
}

/* Unsteady version with Wakes */
DEFINE_SOURCE(x_bf_wake,c,t,dS,eqn)
{
    #if !RP_HOST
        real source;
        int condition1, condition2, condition3, i;
        real M1rel,xle35;
        double phi, thetau, th_shift;

/* definition of geometry and flow variables */

        real xc[ND_ND]; /* ND_ND = 3 for 3D, 2 for 2D */
        C_CENTROID(xc,c,t); /* returns cell centroid array*/

```

```

real x = xc[0]; /* Cell xyz coordinates */
real y = xc[1];
real z = xc[2];
real r = sqrt(y*y+z*z); /* Cell radial coordinate */
real theta = atan2(z,y); /* Cell theta coordinate */
real sinq = z/r; /* sin(theta) */
real cosq = y/r; /* cos(theta) */
real Uwheel = r*omega; /* Wheel speed in m/s */

real rho = C_R(c,t); /* Cell Density */
real p = C_P(c,t); /* Cell Pressure */
real T = C_T(c,t); /* Cell Temperature */
real u = C_U(c,t);
real v = C_V(c,t); /* Cell xyz velocities */
real w = C_W(c,t);
real Rgas = C_RGAS(c,t);
real gamma = 1.4;

real Vr = v*cosq+w*sinq; /* Radial Velocity */
real Vt = -v*sinq+w*cosq; /* Tangential Velocity */
real Vtrel = Vt-Uwheel; /* Rel. Tang. Velocity */
real Vrel = sqrt(u*u+Vr*Vr+Vtrel*Vtrel); /* Relative Vel. Mag. */
real alpha = (1.63+0.46*r-14.3*x+54.9*r*x-12.2*r*r+0.4*x*x)
+ (5.799-64.4*x-38.11*r+139.1*x*x+423.4*x*r+32.26*r*r-765.5*x*x*r
-413.4*x*r*r+35.42*r*r*r); /* Local Metal Angle */
real sol = 51.235*r*r-28.57*r+4.888; /* solidity, linear from hub
(1.77) to tip (1.29) */
real Kn = (4.2-3.3*alpha)*(-89.3572*r*r+39.3732*r-2.0773);
real gap = 2*M_PI*r*cos(alpha)*sqrt(sol)/Nblade; /* Staggered Spacing,
solidity^(1-Ncarter), Ncarter=0.5 */

/* define local blade coordinate - poly fits are used to match SDT
geometry */
real xle = 0.181-1.21*r+9.45*r*r-19.1*r*r*r-72.9*r*r*r*r+359*r*r*r*r*r
-402*r*r*r*r*r*r; /* LE definition, m */
real xte = 0.0403+2.62*r-4.46*r*r-129*r*r*r+961*r*r*r*r-2695*r*r*r*r*r
+2764*r*r*r*r*r*r; /* TE definition, m */
real twist = -(-0.5596+10.63*x-4.086*r-62.89*x*x+50.63*x*r-1.262*r*r
+203.9*x*x*r-236.5*x*r*r+52.93*r*r*r); /* blade twist */
real chord = xte-xle;
real pct_chord = (x-xle)/chord; /* % chord */

phi = omega*CURRENT_TIME; /* radian measure of blade movement starting
at time = 0 */
phi = (phi/(2.0*M_PI)-floor(phi/(2.0*M_PI)))*2.0*M_PI; /* convert phi
so it is always between 0 and 2pi */
theta = theta+M_PI; /* atan2() above returns [-pi,pi], so we add pi so
the range is [0,2pi] */
thetau = theta-phi; /* thetau is the "unsteady" theta for the force
perturbation - moves with speed omega*time */
thetau = thetau+twist; /* TWIST */
thetau = (thetau/(2.0*M_PI)-floor(thetau/(2.0*M_PI)))*2.0*M_PI; /*
convert thetau so it is always between 0 and 2pi */

```



```

th_shift = (thetau/(2.0*M_PI/Nblade)-floor(thetau/(2.0*M_PI/Nblade)))
          *2.0*M_PI/Nblade; /* local theta in each passage, range = [0, 2pi/
          N] = [0,0.28560] */

/* definition of gradients in y,z */

real rho_x = C_R_RG(c,t)[0]; /* d(rho)/dx */
real rho_y = C_R_RG(c,t)[1]; /* d(rho)/dy */
real rho_z = C_R_RG(c,t)[2]; /* d(rho)/dz */
real T_x = C_T_RG(c,t)[0]; /* dT/dx */
real T_y = C_T_RG(c,t)[1]; /* dT/dy */
real T_z = C_T_RG(c,t)[2]; /* dT/dz */
real p_x = Rgas*(rho*T_x+T*rho_x); /* dP/dx */
real p_y = Rgas*(rho*T_y+T*rho_y); /* dP/dy */
real p_z = Rgas*(rho*T_z+T*rho_z); /* dP/dz */

/* Compute Body Forces (x,r,theta), convert to (x,y,z) */

/* 1. Viscous Body Force */
real BFX = 0;

if ( (pct_chord > 0.85) && (th_shift < 0.0074) )
{
    BFX = -pert_const*(Kp1+Kp2*(Mrel-Mref))*(Mrel-Mref))*Vrel*u/gap;
}

/* 2. Turning Force */
real BFn = Kn*(Vtrel*cos(alpha)-u*sin(alpha))*(u*cos(alpha)+Vtrel*sin(
alpha))/gap;
BFX = BFX + BFn*Vtrel/Vrel;

/* 3. BF due to Pressure Gradient in Staggered Passage */
real PG_n = p_x*sin(alpha)/rho; /* pressure gradient N/m^3 */
real PG_x = PG_n*sin(alpha);

/* Source Term */

source = rho*BFX+PG_x; /*source in N/m^3 */

C_UDMI(c,t,xudm2)=BFX; /* store the steady body force */
C_UDMI(c,t,th_shiftudm)=th_shift; /* store the th_shift */

dS[eqn] = 0.0;

return source;
#endif
}

DEFINE_SOURCE(y_bf_knkp,c,t,dS,eqn)
{
    #if !RP_HOST
    real source;

/* definition of geometry and flow variables */

```

```

real xc[ND_ND]; /* ND_ND = 3 for 3D, 2 for 2D */
C_CENTROID(xc,c,t); /* returns cell centroid array*/
real x = xc[0]; /* Cell xyz coordinates */
real y = xc[1];
real z = xc[2];
real r = sqrt(y*y+z*z); /* Cell radial coordinate */
real theta = atan2(z,y); /* Cell theta coordinate */
real sinq = z/r; /* sin(theta) */
real cosq = y/r; /* cos(theta) */
real Uwheel = r*omega; /* Wheel speed in m/s */

real rho = C_R(c,t); /* Cell Density */
real p = C_P(c,t); /* Cell Pressure */
real T = C_T(c,t); /* Cell Temperature */
real u = C_U(c,t);
real v = C_V(c,t); /* Cell xyz velocities */
real w = C_W(c,t);
real Rgas = C_RGAS(c,t);
real gamma = 1.4;

real Vr = v*cosq+w*sinq; /* Radial Velocity */
real Vt = -v*sinq+w*cosq; /* Tangential Velocity */
real Vtrel = Vt-Uwheel; /* Rel. Tang. Velocity */
real Vrel = sqrt(u*u+Vr*Vr+Vtrel*Vtrel); /* Relative Vel. Mag. */
real alpha = (1.63+0.46*r-14.3*x+54.9*r*x-12.2*r*r+0.4*x*x)
+ (5.799-64.4*x-38.11*r+139.1*x*x+423.4*x*r+32.26*r*r-765.5*x*x*r
-413.4*x*r*r+35.42*r*r*r); /* Local Metal Angle */
real sol = 51.235*r*r-28.57*r+4.888; /* solidity, linear from hub
(1.77) to tip (1.29) */
real Kn = (4.2-3.3*alpha)*(-89.3572*r*r+39.3732*r-2.0773);
real gap = 2*M_PI*r*cos(alpha)*sqrt(sol)/Nblade; /* Staggered Spacing,
solidity^(1-Ncarter), Ncarter=0.5 */

/* definition of gradients in y,z */

real rho_x = C_R_RG(c,t)[0]; /* d(rho)/dx */
real rho_y = C_R_RG(c,t)[1]; /* d(rho)/dy */
real rho_z = C_R_RG(c,t)[2]; /* d(rho)/dz */
real T_x = C_T_RG(c,t)[0]; /* dT/dx */
real T_y = C_T_RG(c,t)[1]; /* dT/dy */
real T_z = C_T_RG(c,t)[2]; /* dT/dz */
real p_x = Rgas*(rho*T_x+T*rho_x); /* dP/dx */
real p_y = Rgas*(rho*T_y+T*rho_y); /* dP/dy */
real p_z = Rgas*(rho*T_z+T*rho_z); /* dP/dz */

/* Compute Body Forces (x,r,theta), convert to (x,y,z) */

/* 1. Viscous Body Force*/
real BFv = -(Kp1+Kp2*(Mrel-Mref))*(Mrel-Mref)*Vrel*Vt/gap;
real BFr = -(Kp1+Kp2*(Mrel-Mref))*(Mrel-Mref)*Vrel*Vr/gap;

/* 2. Turning Force */

```

```

    real BFn = Kn*(Vtrel*cos(alpha)-u*sin(alpha))*(u*cos(alpha)+Vtrel*sin(
        alpha))/gap;
    real BFt = BFv-BFn*u/Vrel;

/* 3. BF due to Pressure Gradient in Staggered Passage */
    real PG_x = p_x; /* pressure gradient N/m^3 */
    real PG_n = PG_x*sin(alpha)/rho;
    real PG_t = -PG_n*cos(alpha);
    real PG_y = -PG_t*sinq;
    real PG_z = PG_t*cosq;

/****** Y & Z Body Forces *****/
    real BFy = BFr*cosq - BFt*sinq;
    real BFz = BFr*sinq + BFt*cosq;

/* Source Term */

    source = rho*BFy +PG_y; /*source in N/m^3 */

    C_UDMI(c,t,uudm) = u; /* store the x-velocity */
    C_UDMI(c,t,vudm) = v; /* store the y-velocity */
    C_UDMI(c,t,wudm) = w; /* store the z-velocity */
    C_UDMI(c,t,rudm) = rho; /* store the density */

    C_UDMI(c,t,BFvudm) = BFv; /* store the viscous body force */
    C_UDMI(c,t,BFnudm) = BFn; /* store the normal body force */
    C_UDMI(c,t,BFrudm) = BFr; /* store the radial body force */
    C_UDMI(c,t,PGxudm) = PG_x; /* store the pressure gradient */
    C_UDMI(c,t,PGnudm) = PG_n; /* store the normal pressure gradient */

    C_UDMI(c,t,yudm) = BFy; /* store the steady body force */

    dS[eqn] = 0.0;

    return source;
#endif
}

/* Unsteady version with Wakes */
DEFINE_SOURCE(y_bf_wake,c,t,dS,eqn)
{
    #if !RP_HOST
    real source;
    int i;
    double phi, theta, th_shift;

/* definition of geometry and flow variables */

    real xc[ND_ND]; /* ND_ND = 3 for 3D, 2 for 2D */
    C_CENTROID(xc,c,t); /* returns cell centroid array*/
    real x = xc[0]; /* Cell xyz coordinates */
    real y = xc[1];
    real z = xc[2];
    real r = sqrt(y*y+z*z); /* Cell radial coordinate */

```

```

real theta = atan2(z,y); /* Cell theta coordinate */
real sinq = z/r; /* sin(theta) */
real cosq = y/r; /* cos(theta) */
real Uwheel = r*omega; /* Wheel speed in m/s */

real rho = C_R(c,t); /* Cell Density */
real p = C_P(c,t); /* Cell Pressure */
real T = C_T(c,t); /* Cell Temperature */
real u = C_U(c,t);
real v = C_V(c,t); /* Cell xyz velocities */
real w = C_W(c,t);
real Rgas = C_RGAS(c,t);
real gamma = 1.4;

real Vr = v*cosq+w*sinq; /* Radial Velocity */
real Vt = -v*sinq+w*cosq; /* Tangential Velocity */
real Vtrel = Vt-Uwheel; /* Rel. Tang. Velocity */
real Vrel = sqrt(u*u+Vr*Vr+Vtrel*Vtrel); /* Relative Vel. Mag. */
real alpha = (1.63+0.46*r-14.3*x+54.9*r*x-12.2*r*r+0.4*x*x)
+ (5.799-64.4*x-38.11*r+139.1*x*x+423.4*x*r+32.26*r*r-765.5*x*x*r
-413.4*x*r*r+35.42*r*r*r); /* Local Metal Angle */
real sol = 51.235*r*r-28.57*r+4.888; /* solidity, linear from hub
(1.77) to tip (1.29) */
real Kn = (4.2-3.3*alpha)*(-89.3572*r*r+39.3732*r-2.0773);
real gap = 2*M_PI*r*cos(alpha)*sqrt(sol)/Nblade; /* Staggered Spacing,
solidity^(1-Ncarter), Ncarter=0.5 */

/* define local blade coordinate - poly fits are used to match SDT
geometry */
real xle = 0.181-1.21*r+9.45*r*r-19.1*r*r*r-72.9*r*r*r*r+359*r*r*r*r*r
-402*r*r*r*r*r*r; /* LE definition, m */
real xte = 0.0403+2.62*r-4.46*r*r-129*r*r*r+961*r*r*r*r-2695*r*r*r*r*r
+2764*r*r*r*r*r*r; /* TE definition, m */
real twist = -(-0.5596+10.63*x-4.086*r-62.89*x*x+50.63*x*r-1.262*r*r
+203.9*x*x*r-236.5*x*r*r+52.93*r*r*r); /* blade twist */
real chord = xte-xle;
real pct_chord = (x-xle)/chord; /* % chord */

phi = omega*CURRENT_TIME; /* radian measure of blade movement starting
at time = 0 */
phi = (phi/(2.0*M_PI)-floor(phi/(2.0*M_PI)))*2.0*M_PI; /* convert phi
so it is always between 0 and 2pi */
theta = theta+M_PI; /* atan2() above returns [-pi,pi], so we add pi so
the range is [0,2pi] */
thetau = theta-phi; /* thetau is the "unsteady" theta for the force
perturbation - moves with speed omega*time */
thetau = thetau+twist; /* TWIST */
thetau = (thetau/(2.0*M_PI)-floor(thetau/(2.0*M_PI)))*2.0*M_PI; /*
convert thetau so it is always between 0 and 2pi */
th_shift = (thetau/(2.0*M_PI/Nblade)-floor(thetau/(2.0*M_PI/Nblade)))
*2.0*M_PI/Nblade; /* local theta in each passage, range = [0, 2pi/
N] = [0,0.28560] */

/* definition of gradients in y,z */

```

```

real rho_x = C_R_RG(c,t)[0]; /* d(rho)/dx */
real rho_y = C_R_RG(c,t)[1]; /* d(rho)/dy */
real rho_z = C_R_RG(c,t)[2]; /* d(rho)/dz */
real T_x = C_T_RG(c,t)[0]; /* dT/dx */
real T_y = C_T_RG(c,t)[1]; /* dT/dy */
real T_z = C_T_RG(c,t)[2]; /* dT/dz */
real p_x = Rgas*(rho*T_x+T*rho_x); /* dP/dx */
real p_y = Rgas*(rho*T_y+T*rho_y); /* dP/dy */
real p_z = Rgas*(rho*T_z+T*rho_z); /* dP/dz */

/* Compute Body Forces (x,r,theta), convert to (x,y,z) */

/* 1. Viscous Body Force */
real BFv = 0;
real BFr = 0;

if ( (pct_chord > 0.85) && (th_shift < 0.0074) )
{
    BFv = -pert_const*(Kp1+Kp2*(Mrel-Mref))*(Mrel-Mref)*Vrel
        *Vt/gap;
    BFr = -pert_const*(Kp1+Kp2*(Mrel-Mref))*(Mrel-Mref)*Vrel
        *Vr/gap;
}

/* 2. Turning Force */
real BFn = Kn*(Vtrel*cos(alpha)-u*sin(alpha))*(u*cos(alpha)+Vtrel*sin(
    alpha))/gap;
real BFt = BFv-BFn*u/Vrel;

/* 3. BF due to Pressure Gradient in Staggered Passage */
real PG_x = p_x; /* pressure gradient N/m^3 */
real PG_n = PG_x*sin(alpha)/rho;
real PG_t = -PG_n*cos(alpha);
real PG_y = -PG_t*sinq;
real PG_z = PG_t*cosq;

/****** Y & Z Body Forces *****/
real BFy = BFr*cosq-BFt*sinq;
real BFz = BFr*sinq+BFt*cosq;

/* Source Term */

source = rho*BFy+PG_y; /*source in N/m^3 */

dS[eqn] = 0.0;

return source;
#endif
}

DEFINE_SOURCE(z_bf_knkp, c, t, dS, eqn)
{

```

```

#if !RP_HOST
real source;

/* definition of geometry and flow variables */

real xc[ND_ND]; /* ND_ND = 3 for 3D, 2 for 2D */
C_CENTROID(xc,c,t); /* returns cell centroid array*/
real x = xc[0]; /* Cell xyz coordinates */
real y = xc[1];
real z = xc[2];
real r = sqrt(y*y+z*z); /* Cell radial coordinate */
real theta = atan2(z,y); /* Cell theta coordinate */
real sinq = z/r; /* sin(theta) */
real cosq = y/r; /* cos(theta) */
real Uwheel = r*omega; /* Wheel speed in m/s */

real rho = C_R(c,t); /* Cell Density */
real p = C_P(c,t); /* Cell Pressure */
real T = C_T(c,t); /* Cell Temperature */
real u = C_U(c,t);
real v = C_V(c,t); /* Cell xyz velocities */
real w = C_W(c,t);
real Rgas = C_RGAS(c,t);
real gamma = 1.4;

real Vr = v*cosq+w*sinq; /* Radial Velocity */
real Vt = -v*sinq+w*cosq; /* Tangential Velocity */
real Vtrel = Vt-Uwheel; /* Rel. Tang. Velocity */
real Vrel = sqrt(u*u+Vr*Vr+Vtrel*Vtrel); /* Relative Vel. Mag. */
real alpha = (1.63+0.46*r-14.3*x+54.9*r*x-12.2*r*r+0.4*x*x)
+ (5.799-64.4*x-38.11*r+139.1*x*x+423.4*x*r+32.26*r*r-765.5*x*x*r
-413.4*x*r*r+35.42*r*r*r); /* Local Metal Angle */
real sol = 51.235*r*r-28.57*r+4.888; /* solidity, linear from hub
(1.77) to tip (1.29) */
real Kn = (4.2-3.3*alpha)*(-89.3572*r*r+39.3732*r-2.0773);
real gap = 2*M_PI*r*cos(alpha)*sqrt(sol)/Nblade; /* Staggered Spacing,
solidity^(1-Ncarter), Ncarter=0.5 */

/* definition of gradients in y,z */

real rho_x = C_R_RG(c,t)[0]; /* d(rho)/dx */
real rho_y = C_R_RG(c,t)[1]; /* d(rho)/dy */
real rho_z = C_R_RG(c,t)[2]; /* d(rho)/dz */
real T_x = C_T_RG(c,t)[0]; /* dT/dx */
real T_y = C_T_RG(c,t)[1]; /* dT/dy */
real T_z = C_T_RG(c,t)[2]; /* dT/dz */
real p_x = Rgas*(rho*T_x+T*rho_x); /* dP/dx */
real p_y = Rgas*(rho*T_y+T*rho_y); /* dP/dy */
real p_z = Rgas*(rho*T_z+T*rho_z); /* dP/dz */

/* Compute Body Forces (x,r,theta), convert to (x,y,z) */

/* 1. Viscous Body Force */
real BFv = -(Kp1+Kp2*(Mrel-Mref))*(Mrel-Mref)*Vrel*Vt/gap;

```

```

    real BFr = -(Kp1+Kp2*(Mrel-Mref))*(Mrel-Mref)*Vrel*Vr/gap;

/* 2. Turning Force */
    real BFn = Kn*(Vtrel*cos(alpha)-u*sin(alpha))*(u*cos(alpha)+Vtrel*sin(
        alpha))/gap;
    real BFt = BFv-BFn*u/Vrel;

/* 3. BF due to Pressure Gradient in Staggered Passage */

    real PG_x = p_x; /* pressure gradient N/m^3 */
    real PG_n = PG_x*sin(alpha)/rho;
    real PG_t = -PG_n*cos(alpha);
    real PG_y = -PG_t*sinq;
    real PG_z = PG_t*cosq;

/****** Y & Z Body Forces *****/
    real BFy = BFr*cosq-BFt*sinq;
    real BFz = BFr*sinq+BFt*cosq;

/* Source Term */

    source = rho*BFz+PG_z; /*source in N/m^3 */

    C_UDMI(c,t,zudm) = source; /* store the steady body force */

    dS[eqn] = 0.0;

    return source;
        #endif
}

/* Unsteady version with Wakes */
DEFINE_SOURCE(z_bf_wake,c,t,dS,eqn)
{
    #if !RP_HOST
        real source;
        int i;
        double phi, theta, th_shift;

/* definition of geometry and flow variables */

        real xc[ND_ND]; /* ND_ND = 3 for 3D, 2 for 2D */
        C_CENTROID(xc,c,t); /* returns cell centroid array*/
        real x = xc[0]; /* Cell xyz coordinates */
        real y = xc[1];
        real z = xc[2];
        real r = sqrt(y*y+z*z); /* Cell radial coordinate */
        real theta = atan2(z,y); /* Cell theta coordinate */
        real sinq = z/r; /* sin(theta) */
        real cosq = y/r; /* cos(theta) */
        real Uwheel = r*omega; /* Wheel speed in m/s */

        real rho = C_R(c,t); /* Cell Density */
        real p = C_P(c,t); /* Cell Pressure */

```

```

real T = C_T(c,t); /* Cell Temperature */
real u = C_U(c,t);
real v = C_V(c,t); /* Cell xyz velocities */
real w = C_W(c,t);
real Rgas = C_RGAS(c,t);
real gamma = 1.4;

real Vr = v*cosq+w*sinq; /* Radial Velocity */
real Vt = -v*sinq+w*cosq; /* Tangential Velocity */
real Vtrel = Vt-Uwheel; /* Rel. Tang. Velocity */
real Vrel = sqrt(u*u+Vr*Vr+Vtrel*Vtrel); /* Relative Vel. Mag. */
real alpha = (1.63+0.46*r-14.3*x+54.9*r*x-12.2*r*r+0.4*x*x)
  +(5.799-64.4*x-38.11*r+139.1*x*x+423.4*x*r+32.26*r*r-765.5*x*x*r
  -413.4*x*r*r+35.42*r*r*r); /* Local Metal Angle */
real sol = 51.235*r*r-28.57*r+4.888; /* solidity, linear from hub
  (1.77) to tip (1.29) */
real Kn = (4.2-3.3*alpha)*(-89.3572*r*r+39.3732*r-2.0773);
real gap = 2*M_PI*r*cos(alpha)*sqrt(sol)/Nblade; /* Staggered Spacing,
  solidity^(1-Ncarter), Ncarter=0.5 */

/* define local blade coordinate - poly fits are used to match SDT
  geometry */
real xle = 0.181-1.21*r+9.45*r*r-19.1*r*r*r-72.9*r*r*r*r+359*r*r*r*r*r
  -402*r*r*r*r*r*r; /* LE definition, m */
real xte = 0.0403+2.62*r-4.46*r*r-129*r*r*r+961*r*r*r*r-2695*r*r*r*r*r
  +2764*r*r*r*r*r*r; /* TE definition, m */
real twist = -(-0.5596+10.63*x-4.086*r-62.89*x*x+50.63*x*r-1.262*r*r
  +203.9*x*x*r-236.5*x*r*r+52.93*r*r*r); /* blade twist */
real chord = xte-xle;
real pct_chord = (x-xle)/chord; /* % chord */

phi = omega*CURRENT_TIME; /* radian measure of blade movement starting
  at time = 0 */
phi = (phi/(2.0*M_PI)-floor(phi/(2.0*M_PI)))*2.0*M_PI; /* convert phi
  so it is always between 0 and 2pi */
theta = theta+M_PI; /* atan2() above returns [-pi,pi], so we add pi so
  the range is [0,2pi] */
thetau = theta-phi; /* thetau is the "unsteady" theta for the force
  perturbation - moves with speed omega*time */
thetau = thetau+twist; /* TWIST */
thetau = (thetau/(2.0*M_PI)-floor(thetau/(2.0*M_PI)))*2.0*M_PI; /*
  convert thetau so it is always between 0 and 2pi */
th_shift = (thetau/(2.0*M_PI/Nblade)-floor(thetau/(2.0*M_PI/Nblade)))
  *2.0*M_PI/Nblade; /* local theta in each passage, range = [0, 2pi/
  N] = [0,0.28560] */

/* definition of gradients in y,z */

real rho_x = C_R_RG(c,t)[0]; /* d(rho)/dx */
real rho_y = C_R_RG(c,t)[1]; /* d(rho)/dy */
real rho_z = C_R_RG(c,t)[2]; /* d(rho)/dz */
real T_x = C_T_RG(c,t)[0]; /* dT/dx */
real T_y = C_T_RG(c,t)[1]; /* dT/dy */
real T_z = C_T_RG(c,t)[2]; /* dT/dz */

```



```

    real p_x = Rgas*(rho*T_x+T*rho_x); /* dP/dx */
    real p_y = Rgas*(rho*T_y+T*rho_y); /* dP/dy */
    real p_z = Rgas*(rho*T_z+T*rho_z); /* dP/dz */

/* Compute Body Forces (x,r,theta), convert to (x,y,z) */

/* 1. Viscous Body Force */
real BFv = 0;
real BFr = 0;

if ( (pct_chord > 0.85) && (th_shift < 0.0074) )
{
    BFv = -pert_const*(Kp1+Kp2*(Mrel-Mref))*(Mrel-Mref)*Vrel
        *Vt/gap;
    BFr = -pert_const*(Kp1+Kp2*(Mrel-Mref))*(Mrel-Mref)*Vrel
        *Vr/gap;
}

/* 2. Turning Force */
real BFn = Kn*(Vtrel*cos(alpha)-u*sin(alpha))*(u*cos(alpha)+Vtrel*sin(
    alpha))/gap;
real BFt = BFv-BFn*u/Vrel;

/* 3. BF due to Pressure Gradient in Staggered Passage */

real PG_x = p_x; /* pressure gradient N/m^3 */
real PG_n = PG_x*sin(alpha)/rho;
real PG_t = -PG_n*cos(alpha);
real PG_y = -PG_t*sinq;
real PG_z = PG_t*cosq;

/****** Y & Z Body Forces *****/
real BFy = BFr*cosq-BFt*sinq;
real BFz = BFr*sinq+BFt*cosq;

/* Source Term */

source = rho*BFz+PG_z; /* source in N/m^3 */

dS[eqn] = 0.0;

return source;
#endif
}

DEFINE_SOURCE(e_bf_knkp, c, t, dS, eqn)
{
    #if !RP_HOST
    real source;

/* definition of geometry and flow variables */

    real xc[ND_ND]; /* ND_ND = 3 for 3D, 2 for 2D */
    C_CENTROID(xc, c, t); /* returns cell centroid array*/

```

```

real x = xc [0]; /* Cell xyz coordinates */
real y = xc [1];
real z = xc [2];
real r = sqrt(y*y+z*z); /* Cell radial coordinate */
real theta = atan2(z,y); /* Cell theta coordinate */
real sinq = z/r; /* sin(theta) */
real cosq = y/r; /* cos(theta) */
real Uwheel = r*omega; /* Wheel speed in m/s */

real rho = C_R(c,t); /* Cell Density */
real p = C_P(c,t); /* Cell Pressure */
real T = C_T(c,t); /* Cell Temperature */
real u = C_U(c,t);
real v = C_V(c,t); /* Cell xyz velocities */
real w = C_W(c,t);
real Rgas = C_RGAS(c,t);
real gamma = 1.4;

real Vr =v*cosq+w*sinq; /* Radial Velocity */
real Vt =-v*sinq+w*cosq; /* Tangential Velocity */
real Vtrel = Vt-Uwheel; /* Rel. Tang. Velocity */
real Vrel = sqrt(u*u+Vr*Vr+Vtrel*Vtrel); /* Relative Vel. Mag. */
real alpha = (1.63+0.46*r-14.3*x+54.9*r*x-12.2*r*r+0.4*x*x)
  +(5.799-64.4*x-38.11*r+139.1*x*x+423.4*x*r+32.26*r*r-765.5*x*x*r
  -413.4*x*r*r+35.42*r*r*r); /* Local Metal Angle */
real sol = 51.235*r*r-28.57*r+4.888; /* solidity , linear from hub
  (1.77) to tip (1.29) */
real Kn = (4.2-3.3*alpha)*(-89.3572*r*r+39.3732*r-2.0773);
real gap = 2*M_PI*r*cos(alpha)*sqrt(sol)/Nblade; /* Staggered Spacing,
  solidity^(1-Ncarter), Ncarter=0.5 */

/* definition of gradients in y,z */

real rho_x = C_R_RG(c,t) [0]; /* d(rho)/dx */
real rho_y = C_R_RG(c,t) [1]; /* d(rho)/dy */
real rho_z = C_R_RG(c,t) [2]; /* d(rho)/dz */
real T_x = C_T_RG(c,t) [0]; /* dT/dx */
real T_y = C_T_RG(c,t) [1]; /* dT/dy */
real T_z = C_T_RG(c,t) [2]; /* dT/dz */
real p_x = Rgas*(rho*T_x+T*rho_x); /* dP/dx */
real p_y = Rgas*(rho*T_y+T*rho_y); /* dP/dy */
real p_z = Rgas*(rho*T_z+T*rho_z); /* dP/dz */

/* Compute Body Forces (x,r,theta), convert to (x,y,z) */

/* 1. Viscous Body Force */
real BFv = -(Kp1+Kp2*(Mrel-Mref))*(Mrel-Mref)*Vrel*Vt/gap;

/* 2. Turning Force */
real BFn = Kn*(Vtrel*cos(alpha)-u*sin(alpha))*(u*cos(alpha)+Vtrel*sin(
  alpha))/gap;
real BFt = BFv-BFn*u/Vrel;

/* 3. BF due to Pressure Gradient in Staggered Passage */

```

```

real PG_x = p_x; /* pressure gradient N/m^3 */
real PG_n = PG_x*sin(alpha)/rho;
real PG_t = -PG_n*cos(alpha);

real rhoBFt = rho*BFt+PG_t;

/* Source Term */

source = rhoBFt*Uwheel; /* source in W/m^3 */

C_UDMI(c,t,eudm) = source; /* store the steady body force */

dS[eqn] = 0.0;

return source;
    #endif
}

/* Unsteady version with Wakes */
DEFINE_SOURCE(e_bf_wake,c,t,dS,eqn)
{
    #if !RP_HOST
        real source;
        int i;
        double phi, theta, th_shift;

/* definition of geometry and flow variables */

        real xc[ND_ND]; /* ND_ND = 3 for 3D, 2 for 2D */
        C_CENTROID(xc,c,t); /* returns cell centroid array */
        real x = xc[0]; /* Cell xyz coordinates */
        real y = xc[1];
        real z = xc[2];
        real r = sqrt(y*y+z*z); /* Cell radial coordinate */
        real theta = atan2(z,y); /* Cell theta coordinate */
        real sinq = z/r; /* sin(theta) */
        real cosq = y/r; /* cos(theta) */
        real Uwheel = r*omega; /* Wheel speed in m/s */

        real rho = C_R(c,t); /* Cell Density */
        real p = C_P(c,t); /* Cell Pressure */
        real T = C_T(c,t); /* Cell Temperature */
        real u = C_U(c,t);
        real v = C_V(c,t); /* Cell xyz velocities */
        real w = C_W(c,t);
        real Rgas = C_RGAS(c,t);
        real gamma = 1.4;

        real Vr = v*cosq+w*sinq; /* Radial Velocity */
        real Vt = -v*sinq+w*cosq; /* Tangential Velocity */
        real Vtrel = Vt-Uwheel; /* Rel. Tang. Velocity */
        real Vrel = sqrt(u*u+Vr*Vr+Vtrel*Vtrel); /* Relative Vel. Mag. */
        real alpha = (1.63+0.46*r-14.3*x+54.9*r*x-12.2*r*r+0.4*x*x)
            +(5.799-64.4*x-38.11*r+139.1*x*x+423.4*x*r+32.26*r*r-765.5*x*x*r

```

```

    -413.4*x*r*r+35.42*r*r*r); /* Local Metal Angle */
real sol = 51.235*r*r-28.57*r+4.888; /* solidity, linear from hub
    (1.77) to tip (1.29) */
real Kn = (4.2-3.3*alpha)*(-89.3572*r*r+39.3732*r-2.0773);
real gap = 2*M_PI*r*cos(alpha)*sqrt(sol)/Nblade; /* Staggered Spacing,
    solidity^(1-Ncarter), Ncarter=0.5 */

/* define local blade coordinate - poly fits are used to match SDT
    geometry */
real xle = 0.181-1.21*r+9.45*r*r-19.1*r*r*r-72.9*r*r*r*r+359*r*r*r*r*r
    -402*r*r*r*r*r; /* LE definition, m */
real xte = 0.0403+2.62*r-4.46*r*r-129*r*r*r+961*r*r*r*r-2695*r*r*r*r*r
    +2764*r*r*r*r*r; /* TE definition, m */
real twist = -(-0.5596+10.63*x-4.086*r-62.89*x*x+50.63*x*r-1.262*r*r
    +203.9*x*x*r-236.5*x*r*r+52.93*r*r*r); /* blade twist */
real chord = xte-xle;
real pct_chord = (x-xle)/chord; /* % chord */

phi = omega*CURRENT_TIME; /* radian measure of blade movement starting
    at time = 0 */
phi = (phi/(2.0*M_PI)-floor(phi/(2.0*M_PI)))*2.0*M_PI; /* convert phi
    so it is always between 0 and 2pi */
theta = theta+M_PI; /* atan2() above returns [-pi,pi], so we add pi so
    the range is [0,2pi] */
thetau = theta-phi; /* thetau is the "unsteady" theta for the force
    perturbation - moves with speed omega*time */
thetau = thetau+twist; /* TWIST */
thetau = (thetau/(2.0*M_PI)-floor(thetau/(2.0*M_PI)))*2.0*M_PI; /*
    convert thetau so it is always between 0 and 2pi */
th_shift = (thetau/(2.0*M_PI/Nblade)-floor(thetau/(2.0*M_PI/Nblade)))
    *2.0*M_PI/Nblade; /* local theta in each passage, range = [0, 2pi/
    N] = [0,0.28560] */

/* definition of gradients in y,z */

real rho_x = C_R_RG(c,t)[0]; /* d(rho)/dx */
real rho_y = C_R_RG(c,t)[1]; /* d(rho)/dy */
real rho_z = C_R_RG(c,t)[2]; /* d(rho)/dz */
real T_x = C_T_RG(c,t)[0]; /* dT/dx */
real T_y = C_T_RG(c,t)[1]; /* dT/dy */
real T_z = C_T_RG(c,t)[2]; /* dT/dz */
real p_x = Rgas*(rho*T_x+T*rho_x); /* dP/dx */
real p_y = Rgas*(rho*T_y+T*rho_y); /* dP/dy */
real p_z = Rgas*(rho*T_z+T*rho_z); /* dP/dz */

/* Compute Body Forces (x,r,theta), convert to (x,y,z) */

/* 1. Viscous Body Force */
real BFv = 0;

if ( (pct_chord > 0.85) && (th_shift < 0.0074) )
{
    BFv = -pert_const*(Kp1+Kp2*(Mrel-Mref))*(Mrel-Mref)*Vrel
        *Vt/gap;
}

```

```

}

/* 2. Turning Force */
real BFn = Kn*(Vrel*cos(alpha)-u*sin(alpha))*(u*cos(alpha)+Vrel*sin(alpha))/gap;
real BFt = BFv-BFn*u/Vrel;

/* 3. BF due to Pressure Gradient in Staggered Passage */
real PGx = px; /* pressure gradient N/m3 */
real PGn = PGx*sin(alpha)/rho;
real PGt = -PGn*cos(alpha);

real rhoBFt = rho*BFt+PGt;

/* Source Term */
source = rhoBFt*Uwheel; /*source in W/m3 */

dS[eqn] = 0.0;

return source;
#endif
}

```

Appendix B

Modal Decomposition in Uniform Flow

B.1 κ Function in Matlab

```
function f = kappafunc(m,k,sigma)
    f = ((m.*besselj(m,k)-k.*besselj(m+1,k))./(m.*bessely(m,k)-k.*
        bessely(m+1,k))) - (((m./sigma).*besselj(m,k.*sigma)-k.*besselj(m
        +1,k.*sigma))./(m./sigma).*bessely(m,k.*sigma)-k.*bessely(m+1,k
        .*sigma)));
```

B.2 κ Matrix Generation in Matlab

```
%% Kappa Matrix Generator — Krishna Patel

clear
clc

load datamatrix.mat sigma

%% Setup

m = -44:44; % Circumferential Mode Range
Nmax = 20; % Maximum Number of Radial Modes to consider

%% Fill Kappa Matrix — kappafunc(m,k,sigma)

kappa = zeros(length(m),Nmax);
```

```

for m_ind = 1:length(m)
    kp = [];
    disp([ 'm = ' num2str(m(m_ind)) ])
    ktestin = 0.5:0.001:500;
    ktestout = kappafunc(m(m_ind),ktestin,sigma);
    % Check for NaN/infs:
    kt = find(abs(ktestout)<inf,1);
    ktestin = ktestin(kt:end);
    ktestout = ktestout(kt:end);

    % Now have the proper range for the function...
    % start at the beginning; find a zero; then increment by pi and
      start
    % looking again; repeat

    % first interval: search for a zero but give up on a find of a
      negative
    % slope (indicates an asymptote)

    jf = 1; % 'jump factor' - how much to skip forward normally
    fk = @(k)kappafunc(m(m_ind),k,sigma); % function handle
    k0 = ktestin(kt); % initial value
    kptemp = 0;

    while (length(kp) < Nmax)
        ktempold = kptemp;
        [kptemp, fval, exitflag] = fzero(fk,k0);
        disp(k0)
        if exitflag>0
            if isempty(kp)
                kp = [kp kptemp];
                k0 = kptemp + jf;
                logic = 1;
            elseif (~isempty(kp) && (kptemp-kp(end))>0.1)
                kp = [kp kptemp];
                k0 = kptemp + jf/2;
                logic = 2;
            else
                k0=k0+jf/3; % make sure the algorithm doesn't get stuck
                logic = 3;
            end
        elseif exitflag == -5
            if (fk(kptemp+0.1)-fk(kptemp-0.1))>0 % positive slope - a
              false negative!
                if isempty(kp)
                    k0 = k0 + jf/2;
                    logic = 4;
                elseif (~isempty(kp) && (kptemp-kp(end))>0.1)
                    k0 = k0 + jf/3;
                    logic = 5;
                else
                    k0 = k0 + jf/3;
                    logic = 6;
                end
            end
        end
    end

```

```

        end
    else
        k0 = k0 + jf/3;
        logic = 7;
    end
else
    k0 = k0 + jf;
    logic = 8;
end
disp(['exitflag = ' num2str(exitflag) ', kptemp = ' num2str(
    kptemp) ', length of kp is ' num2str(length(kp)) ', logic =
    ' num2str(logic)])
end
disp(['m = ' num2str(m(m_ind)) ' done'])
kappa(m_ind,:) = kp;
end

%% Fill Q Matrix
Qmn = zeros(size(kappa));

for m_ind = 1:length(m)
    for n = 1:Nmax
        Qmn(m_ind,n) = -((m(m_ind)./kappa(m_ind,n)).*besselj(m(m_ind),
            kappa(m_ind,n))-besselj(m(m_ind)+1,kappa(m_ind,n)))./((m(
            m_ind)./kappa(m_ind,n)).*bessely(m(m_ind),kappa(m_ind,n))-
            bessely(m(m_ind)+1,kappa(m_ind,n))));
    end
end

%% Fill C Matrix
Cmn = zeros(size(kappa));

for m_ind = 1:length(m)
    for n = 1:Nmax
        Cmn(m_ind,n) = 1 / (sqrt(0.5*((1 - (m(m_ind))^2)/((kappa(m_ind),
            n))^2))*((besselj(m(m_ind),kappa(m_ind,n)) + Qmn(m_ind,n)*
            bessely(m(m_ind),kappa(m_ind,n)))^2 - (sigma.^2 - ((m(
            m_ind)).^2)./((kappa(m_ind,n)).^2)).*((besselj(m(m_ind),
            kappa(m_ind,n).*sigma) + Qmn(m_ind,n).*bessely(m(m_ind),
            kappa(m_ind,n).*sigma)).^2))));
    end
end

%% Save Matrix
save kappa.mat kappa Qmn Cmn

```

B.3 E_{mn} Function in Matlab


```

function f = Emn_func(m,r,kappa,C,Q)
f = C.*(besselj(m,kappa.*r)+Q.*bessely(m,kappa.*r));

```

B.4 Sutliff's Modal Decomposition Approach in Matlab

```

%% Get sound power for SDT (validation)
%
% Krishna Patel
% April 2016

clear
clc

%% Setup

load datamatrix.mat PPert YS ZS sigma
dt = 0.00000179562872144; % Time Step Size
omega = 1325.4; % Rotor Angular Velocity

Fs = 1/dt;
RS = sqrt(YS.^2+ZS.^2);
THETAS = atan2(ZS,YS);

clear YS ZS

Nblade = 22; % blades
r_spacing = 114; % radial cell spacing
theta_spacing = 1133; % tangential cell spacing, odd number

Ro = max(RS);
rs = linspace(min(min(RS)),max(max(RS)),r_spacing+1);

m_min = -44; % Minimum Circumferential Mode of Interest
m_max = 44; % Maximum Circumferential Mode of Interest
Nmax = 20; % chosen - maximum number of radial modes to consider

M_D = 0.57064939; % Axial Duct Mach number (mass-flow average)

k = 1.4; % ratio of specific heats
Rgas = 287;
T = 319.33249; % Mass-Averaged Static Temperature on Plane
rho = 1.2915137; % Mass-Averaged Density on Plane

%% Number of Points

La = size(PPert,1); % number of angular points
Lt = size(PPert,3); % number of temporal points

```

```

%% 1. Compute circumferential modes
disp('Determining circumferential modeshapes... ')

% PPert has indices (theta,r,time) - so operate on indices 1 and 3,
% looping over index 2

Paf = zeros(size(PPert,1),size(PPert,3));
NFFT = Lt;

for i=1:size(PPert,2)
    Pmf{i} = zeros([size(PPert,1) NFFT]); % initialize
end

for i=1:size(PPert,2)

    disp(['Computing FFT for radial location ' num2str(i) ' of ' num2str
          (size(PPert,2))])
    Paf = reshape(PPert(:,i,:),size(PPert,1),size(PPert,3));
    Paf = (fft(Paf',NFFT))'/Lt; % Fourier transform only in time -
                                % now have a frequency-dependent vector at
                                % each angular position
                                % with indices (angular position, frequency)
    Pmf{i} = fft(Paf)/La; % Fourier transform only in space -
                          % now have a P(m,f)
    % The indices on Pmf{i} are (m,f) (mode, frequency)
    Pmf{i} = fftshift(Pmf{i}(:,:)); % Shift the data so m=0 is at center
                                    % Also put f = 0 at center (wants +
                                    % and
                                    % - frequencies)
    % Because of the earlier subtraction of the time mean of each
    % p(theta,r) from 'PPert', the DC part should be ~0.

end
clear Paf

%% Create m and f vectors
% (no assumptions about frequency content being made):

mref = linspace(fix(-theta_spacing/2),fix(theta_spacing/2),La);
m_min_ind = find(mref <= m_min,1,'last');
m_max_ind = find(mref >= m_max,1);

m = m_min:m_max; % linspace(fix(-theta_spacing/2),fix(theta_spacing/2),
La);
% floor functions ensure that the m = 0 element is at the correct
% location, regardless of whether there are an even or odd number of
% theta points.

f = Fs/2*linspace(0,1,ceil(NFFT/2));
fn = -f(2):f(1)-f(2):-max(f)-f(2);
fn = fliplr(fn);
if mod(NFFT,2)
    fn = fn(2:end);
end

```

```

f = [fn f];

[mM,fM] = ndgrid(m,f);

%% 2. Compute radial modes

% Reshape the data:

disp('Computing radial modes and mode powers...')

P = zeros(size(PPert,1),size(PPert,2),resolution);
for i = 1:(r_spacing)
    P(:,i,:) = reshape(Pmf{i}(:,,:),[size(Pmf{i},1) 1 size(Pmf{1},2)]);
end

P = P(m_min_ind:m_max_ind,:,:) ; % Reshape to only circumferential modes
    of concern

    % P has indices (m,r,f)

clear Pmf

% Relevant equations (J is Bessel function of 1st kind):
% -----
%
%
%  $P(f,m,r) = \sum_{n=0}^{\infty} (E_{mn}(\kappa_{mn} * r))$ 
%
%  $\kappa_{m,n} * (J_{m+1}(\kappa_{m,n}) / J_m(\kappa_{m,n})) - m = 0$ 
%
%  $(1/C_{m,n})^2 = (1/2) (1 - (m/\kappa_{m,n})^2) (J_m(\kappa_{m,n}))^2$ 
%
%  $E_{m,n}(r) = J_m(\kappa_{m,n} * r)$ 
%
%  $P_{m,n}(r) = C_{m,n} * E_{m,n}(r)$ 
%
%  $P_n = (F_{v,n})^{-1} * E_{v,r} * P_r$  (matrix form for a given frequency)
% (least squares fit)

%% Setup

Nmeas = r_spacing;

r = rs/Ro; % normalized radius
W=ones(size(r)); % Weighting vector

P_done = zeros(length(m),Nmax,size(P,3)); % modal pressures
Wp = zeros(size(P_done)); % mode power
zeta = zeros(length(m),Nmax); % cut-off ratios (reset for each frequency)
)
Nmaxlocal = zeros(length(m),length(f)); % number of propagating radial
    modes

%% Fill kappa matrix:

```

```

load kappa.mat kappa Qmn Cmn % load in (doesn't depend on the results at
    all)

%% Solve matrix equation:

beta = sqrt(1-M_D^2);

for i=find(f<-2*Nblade*(omega/2/pi),1,'last'):find(f>2*Nblade*(omega/2/
    pi),1) %1:length(f) % loop over all RELEVANT frequencies

    % Figure out what modes will propagate, based on cut-off ratio:
    % (frequency must be above the cut-off frequency for the m,n mode)
    % this means zeta>1 means the mode propagates

    zeta = 2*pi*abs(f(i))*Ro./kappa./sqrt(k*Rgas*T)./beta;
    % print how many elements of zeta are >1

    disp(['propagating modes = ' num2str(length(find(zeta>1))])
    % Want use (propagating radial modes + 1) as the number to use in
    % the curve fit for a given circumferential mode

    for m_ind=1:length(m) % find(m <= m_min,1,'last') : find(m >= m_max
        ,1) % loop over modes
        disp(['mode m = ' num2str(m(m_ind)) ', frequency f = ' num2str(f
            (i)/(omega/2/pi)) ' times shaft freq.'])

        % Determine how many radial modes will propagate here:
        if isempty(find(zeta(m_ind,:) >1,1,'last'))
            q = 0;
        else
            q = find(zeta(m_ind,:) >1,1,'last');
        end

        if length(q) == Nmax
            disp('Need to increase Nmax!')
            break
        end

        disp(['Number of radial modes propagating at this f,m: ' num2str
            (q)])
        Nmaxlocal(m_ind,i)=1+q; % includes the first non-propagating
            mode

        % Initialize matrices:
        Fvn = zeros(Nmaxlocal(m_ind,i),Nmaxlocal(m_ind,i));
        Evr = zeros(Nmaxlocal(m_ind,i),Nmeas);

        for v=1:min([Nmaxlocal(m_ind,i) Nmax])
            for n=1:min([Nmaxlocal(m_ind,i) Nmax])
                Fvn(v,n) = sum( (W(n) .* Emn_func( m(m_ind) , r , kappa(
                    m_ind,v) , Cmn(m_ind,v) , Qmn(m_ind,v) )) .* (W(n).*
                    Emn_func( m(m_ind) , r , kappa(m_ind,n) , Cmn(m_ind
                    ,n) , Qmn(m_ind,n) )) );
            end
        end
    end
end

```

```

end

for v=1:min([Nmaxlocal(m_ind, i) Nmax])
    for j=1:Nmeas
        Evr(v, j) = W(j)^2 * Emn_func ( m(m_ind) , r(j) , kappa(
            m_ind, v) , Cmn(m_ind, v) , Qmn(m_ind, v) );
    end
end

% Perform least-squares fit where it applies:
P_done(m_ind, 1:Nmaxlocal(m_ind, i), i) = inv(Fvn)*Evr*(P(m_ind, :, i)
    )');
% Now P_done has indices (m, n, f)

% Compute modal power:
Nmaxlocal(m_ind, i)=Nmaxlocal(m_ind, i)-2;
Wp(m_ind, 1:Nmaxlocal(m_ind, i)+1, i) = pi.*Ro.^2.*(1-sigma^2)./
    sqrt(k.*Rgas.*T)./(rho).*(abs(beta)).^4.*real(sqrt(1-(1./
    zeta(m_ind, 1:Nmaxlocal(m_ind, i)+1)).^2)./(abs(1-M_D.*sqrt
    (1-(1./zeta(m_ind, 1:Nmaxlocal(m_ind, i)+1)).^2)).^2)).*(abs(
    P_done(m_ind, 1:Nmaxlocal(m_ind, i)+1, i)).^2); % cut-on modes
end
end

n=0:1:(Nmax-1);

f_min_ind = find(f<-2*Nblade*(omega/2/pi), 1, 'last');
f_max_ind = find(f>2*Nblade*(omega/2/pi), 1);

Wp_mf = sum(Wp(1:length(m), :, :), 2);
Wp_mf = Wp_mf(:, :, f_min_ind:f_max_ind);
Wp_mf = reshape(Wp_mf, [length(m) , size(Wp_mf, 3)]);

WpL = 10*log10(Wp_mf/(1e-12)); % Convert to dB

%% Save results:

save soundpower.mat Wp WpL Wp_mf m f f_max_ind f_min_ind n

```

Vita Auctoris

Name: Krishna Patel

Place of Birth: Windsor, Ontario

Year of Birth: 1992

Education: M.A.Sc in Mechanical Engineering
University of Windsor, 2014-2016

B.A.Sc in Mechanical Engineering
University of Windsor, 2010-2014