



Theses and Dissertations

---

2018-12-01

## Analysis of the Effects of Inlet Distortion on Stall Cell Formation in a Transonic Compressor Using CREATE-AV Kestrel

Mikkel Andreas Unrau  
*Brigham Young University*

Follow this and additional works at: <https://scholarsarchive.byu.edu/etd>



Part of the [Engineering Commons](#)

---

### BYU ScholarsArchive Citation

Unrau, Mikkel Andreas, "Analysis of the Effects of Inlet Distortion on Stall Cell Formation in a Transonic Compressor Using CREATE-AV Kestrel" (2018). *Theses and Dissertations*. 7712.  
<https://scholarsarchive.byu.edu/etd/7712>

This Thesis is brought to you for free and open access by BYU ScholarsArchive. It has been accepted for inclusion in Theses and Dissertations by an authorized administrator of BYU ScholarsArchive. For more information, please contact [scholarsarchive@byu.edu](mailto:scholarsarchive@byu.edu), [ellen\\_amatangelo@byu.edu](mailto:ellen_amatangelo@byu.edu).

Analysis of the Effects of Inlet Distortion on Stall Cell Formation in a Transonic  
Compressor Using CREATE<sup>TM</sup>-AV Kestrel

Mikkel Andreas Unrau

A thesis submitted to the faculty of  
Brigham Young University  
in partial fulfillment of the requirements for the degree of  
Master of Science

Steven E. Gorrell, Chair  
Julie Crockett  
S. Andrew Ning

Department of Mechanical Engineering  
Brigham Young University

Copyright © 2018 Mikkel Andreas Unrau  
All Rights Reserved

## ABSTRACT

### Analysis of the Effects of Inlet Distortion on Stall Cell Formation in a Transonic Compressor Using CREATE<sup>TM</sup>-AV Kestrel

Mikkel Andreas Unrau  
Department of Mechanical Engineering, BYU  
Master of Science

Accurately predicting fan performance, including bounds of operation, is an important function of any Computational Fluid Dynamics (CFD) package. The presented research uses a CFD code developed as part of the Computational Research and Engineering Acquisition Tools and Environment (CREATE), known as Kestrel, to evaluate a single stage compressor at various operating conditions. Steady-state, single-passage simulations are carried out to validate capabilities recently added to Kestrel. The analysis includes generating speedlines of total pressure ratio and efficiency, as well as radial total temperature and total pressure profiles at two axial locations in the compressor at various operating conditions and fan speeds, and simulation data from the single-passage runs is compared to experimental data. Time-accurate, full annulus simulations are also carried out to capture and analyze the processes leading to stall inception for both uniform and distorted inlet conditions. The distortion profile used contains a 90° sector of lower total pressure at the inlet. The observed fan behavior at stall inception is compared to previous research, and it is concluded that the inlet distortion significantly changes the behavior of the part-span stall cells that develop after stall inception. Understanding the physical processes that lead to stall inception allows fan designers to design more robust fans that can safely take advantage of the better performance associated with operating closer to stall.

Keywords: CFD, Kestrel, turbomachinery, distortion, inlet distortion, stall, stall inception, rotor 4

## ACKNOWLEDGMENTS

My principle thanks goes to Dr. Steven E. Gorrell, who was willing to be my graduate advisor and provide me with valuable insight, direction, guidance, and especially patience. His timely and honest feedback pushed me to overcome challenges as they came up and learn in the process. I am grateful that he gave me the leeway to explore ideas and try new things.

I would also like to thank the other members of my graduate committee, Dr. Andrew Ning and Dr. Julie Crockett. Both were highly involved in my learning experience as an undergraduate student, and both provided me with research opportunities and experience prior to becoming a graduate student. My experiences working with them were instrumental in my deciding to obtain an advanced degree.

Simulations presented in this paper are a product of the CREATE-AV Element of Computational Research and Engineering for Acquisition Tools and Environments (CREATE) Program sponsored by the U.S. Department of Defense HPC Modernization Program Office. I would like to thank the CREATE-AV program for funding and making this research possible. I would like to acknowledge the HPC Modernization Program Office and the Brigham Young University Fulton Super Computing Laboratory for providing computing resources and support. I also acknowledge the support and contributions of Robert Nichols from Arnold Engineering and Development Complex. The author also would like to thank the other members of the Kestrel development team for their support. The author also acknowledges the support of Dr. Michael List and AFRL for supplying geometry and experimental data.

Finally I wish to thank my wonderful family members, especially my parents who always believed in me, and my lovely wife Sara for encouragement and patience, even when progress was slow.

## TABLE OF CONTENTS

<b>LIST OF TABLES</b> . . . . .	<b>vi</b>
<b>LIST OF FIGURES</b> . . . . .	<b>viii</b>
<b>Chapter 1 Introduction</b> . . . . .	<b>1</b>
<b>Chapter 2 Background &amp; Literature Review</b> . . . . .	<b>3</b>
2.1 Compressor Stall . . . . .	3
2.2 Stall Precursors . . . . .	5
2.3 Using CFD to Visualize Flow in Low-Speed Compressors . . . . .	8
2.4 High-Speed Compressors . . . . .	10
2.5 Using a Variable Nozzle Boundary Condition . . . . .	12
2.6 Sources of Inlet Distortion . . . . .	13
2.7 Testing the Effects of Inlet Distortion . . . . .	13
2.8 Modeling the Effects of Inlet Distortion . . . . .	15
2.9 Mixing Plane Analysis . . . . .	16
<b>Chapter 3 Methodology</b> . . . . .	<b>19</b>
3.1 The Kestrel Flow Solver . . . . .	19
3.2 Geometry . . . . .	20
3.3 Mixing Plane Validation Study Methods . . . . .	22
3.3.1 Computational Domain . . . . .	22
3.3.2 Mesh Generation . . . . .	22
3.3.3 Establishing Grid Independence . . . . .	23
3.3.4 Solver Inputs and Boundary Conditions . . . . .	26
3.4 Inlet Distortion Study Methods . . . . .	27
3.4.1 Computational Domain . . . . .	27
3.4.2 Mesh Generation . . . . .	28
3.4.3 Grid Independence . . . . .	31
3.4.4 Solver Inputs and Boundary Conditions . . . . .	32
<b>Chapter 4 Mixing Plane Validation Results</b> . . . . .	<b>35</b>
4.1 Unsteady Flow Across Mixing Planes . . . . .	35
4.2 Discontinuity Across Mixing Planes . . . . .	35
4.3 Performance Maps . . . . .	37
4.4 Radial Profiles . . . . .	43
<b>Chapter 5 Stall Inception Study Results</b> . . . . .	<b>51</b>
5.1 Stall Inception With Uniform Inlet Flow . . . . .	51
5.1.1 Performance Maps . . . . .	51
5.1.2 Stall Cell Formation and Characteristics . . . . .	53
5.2 Stall Inception With Distorted Inlet Flow . . . . .	62

5.2.1	Performance Maps . . . . .	63
5.2.2	Stall Cell Formation and Characteristics . . . . .	63
<b>Chapter 6</b>	<b>Conclusions and Future Work . . . . .</b>	<b>69</b>
6.1	Mixing Plane Validation . . . . .	69
6.2	Inlet Distortion Study . . . . .	70
<b>REFERENCES</b>	<b>. . . . .</b>	<b>73</b>
<b>Appendix A</b>	<b>Generating CAD Files for Single-passage Blade Rows . . . . .</b>	<b>77</b>
<b>Appendix B</b>	<b>Generating Pointwise Meshes for Single-passage Blade Rows . . . . .</b>	<b>91</b>
<b>Appendix C</b>	<b>Utility Scripts . . . . .</b>	<b>109</b>
C.1	Circumvolving a Single-passage Mesh Using Carpenter . . . . .	109
C.1.1	Submission Script . . . . .	109
C.1.2	Carpenter Inputs . . . . .	110
C.2	Monitoring Jobs . . . . .	111
C.2.1	Fortran Script . . . . .	111
C.2.2	Gnuplot Script . . . . .	119
C.3	User-defined Equations in FieldView . . . . .	122
C.4	Generating Distortion BC for Kestrel . . . . .	123

## LIST OF TABLES

3.1	Rotor 4 stage design parameters [27]. . . . .	21
3.2	Cell count before and after refinement. . . . .	26
3.3	Cell counts of old and new meshes. . . . .	30
3.4	Cell counts of full annulus region meshes. . . . .	31

## LIST OF FIGURES

2.1	A typical performance map indicating operating margin used to avoid stalled conditions.	4
2.2	An illustration of flow diversion caused by the presence of a stall cell. . . . .	5
2.3	Hot-wire data demonstrating a modal disturbance preceding stall [3]. . . . .	6
2.4	Hot-wire data demonstrating a spike disturbance preceding stall [3]. . . . .	7
2.5	Hot-wire data demonstrating both modal and spike disturbances preceding stall inception [3]. . . . .	8
2.6	Tip leakage vortex trajectory at design and near stall [7]. . . . .	9
2.7	The formation and propagation of a spike-type stall precursor [10]. . . . .	10
2.8	Pressure contours in rotor tip region at (a) stable operating conditions and (b) near-stall operating conditions [4]. . . . .	11
2.9	A schematic of the B2 propulsion system with S-duct inlet. Image courtesy of aviation.stackexchange.com. . . . .	13
2.10	Example of a one-per-rev distortion screen and the corresponding total pressure distortion [20]. . . . .	14
2.11	Example of a mixing plane simulation showing Mach number distribution for NASA Stage35 [32]. . . . .	17
3.1	Rotor 4 geometry. . . . .	21
3.2	The computational domains used in the mixing plane validation study. . . . .	23
3.3	Cross section of the rotor single-passage volume mesh. Blue indicates tets, orange indicates hexes, green indicates prisms, and yellow indicates pyramids. . . . .	24
3.4	Cross section of the stator single-passage volume mesh. Blue indicates tets, orange indicates hexes, green indicates prisms, and yellow indicates pyramids. . . . .	24
3.5	Cross section of the rotor single-passage refined volume mesh. Blue indicates tets, orange indicates hexes, green indicates prisms, and yellow indicates pyramids. . . . .	25
3.6	Cross section of the stator single-passage refined volume mesh. Blue indicates tets, orange indicates hexes, green indicates prisms, and yellow indicates pyramids. . . . .	26
3.7	The full annulus computational domain used for the inlet distortion study. . . . .	28
3.8	Cross section of the rotor single-passage volume mesh. Blue indicates tets, orange indicates hexes, green indicates prisms, and yellow indicates pyramids. . . . .	29
3.9	Cross section of the stator single-passage volume mesh. Blue indicates tets, orange indicates hexes, green indicates prisms, and yellow indicates pyramids. . . . .	30
3.10	Cross section of the blunted spinner volume mesh. Blue indicates tets, orange indicates hexes, green indicates prisms, and yellow indicates pyramids. . . . .	31
3.11	Contour plot of the distorted total pressure boundary condition used for this study. . . . .	33
4.1	Rotor 4 stage total pressure convergence demonstrating oscillatory behavior introduced at the mixing plane. . . . .	36
4.2	Rotor 4 stage total pressure convergence after v8.1 fix. . . . .	36
4.3	Rotor 4 stage performance results at 100% design speed. . . . .	39
4.4	Rotor 4 stage performance results at 95% design speed. . . . .	41
4.5	Rotor 4 stage performance results at 90% design speed. . . . .	42
4.6	Rotor 4 stage performance results at 100%, 95%, and 90% design speeds. . . . .	43



4.7	Stator leading edge profiles at 100% design speed for Rotor 4. . . . .	44
4.8	Stage exit profiles at 100% design speed for Rotor 4. . . . .	45
4.9	Stator leading edge profiles at 95% design speed for Rotor 4. . . . .	47
4.10	Stage exit profiles at 95% design speed for Rotor 4. . . . .	47
4.11	Stator leading edge profiles at 90% design speed for Rotor 4. . . . .	48
4.12	Stage exit profiles at 90% design speed for Rotor 4. . . . .	49
5.1	Rotor 4 stage performance results at 100% design speed. . . . .	52
5.2	Pressure traces at various circumferential locations demonstrating the movement of the rotating stall cell. . . . .	55
5.3	Negative axial velocity contours at 99.9% span. . . . .	57
5.4	Negative axial velocity contours at 99.9% span during stall inception. . . . .	58
5.5	Negative axial velocity contours during stalled conditions. . . . .	59
5.6	An axial view showing the extent of the stall cell. . . . .	60
5.7	Isosurface of the $\lambda_2$ variable identifying tip gap vortices and detached shocks. . . . .	60
5.8	An axial view showing the interaction of the $\lambda_2$ isosurface with the stall cell. . . . .	61
5.9	An axial view showing the interaction of the $\lambda_2$ isosurface with the stall cell. . . . .	62
5.10	Pressure ratios for clean and distorted inlet conditions. . . . .	64
5.11	Rotor 4 stage performance results at 100% design speed with distorted inlet BC. . . . .	64
5.12	Pressure traces at various circumferential locations demonstrating the movement of the rotating stall cells. . . . .	65
5.13	An axial view showing the extent of the stall cells. . . . .	66
5.14	Isosurface of the $\lambda_2$ variable with inlet distortion applied. . . . .	67
5.15	An axial view showing the interaction of the $\lambda_2$ isosurface with the four stall cells. . . . .	68

## CHAPTER 1. INTRODUCTION

When designing turbomachinery for air-breathing propulsion systems, a central issue is predicting performance parameters such as pressure ratio, efficiency, and mass flow rates for a given compressor or turbine design. In recent decades computational fluid dynamics (CFD) has become a common tool for predicting inlet, compressor, and turbine performance before a prototype is made and tested, saving time and money in this iterative process. Often, the engine and aircraft inlet are designed separately, with an assumption of uniform inlet flow. However, due to increasingly complex propulsion system configurations, the compressor performance changes when the engine and inlet are coupled, and the engine encounters flow with properties that vary circumferentially or radially, such as total pressure, total temperature, or velocity. Known as inlet distortion, this can lead to compressor stall during flight at higher flow rates than predicted, limiting thrust and putting high loads on the engine.

The Kestrel Computational Fluid Dynamics flow solver, or KCFD, is a cutting edge software tool being developed by the Department of Defense (DoD) to expedite the design and analysis of fixed-wing aircraft, including their propulsion systems. Due to the coupled nature of compressors and inlets, an intended functionality of the software is to allow for integrated analysis of an aircraft with its propulsion system, capturing the effects that the aircraft body and inlet design have on the distortion encountered by the propulsion system [1]. This allows for designers to identify and rectify potential issues before a prototype is made.

The purpose of this research is twofold: first is a validation of the mixing plane capabilities of KCFD. Mixing planes allow for steady simulations of turbomachinery to be run using single blade passages, greatly decreasing run time. This capability was introduced in a recent iteration of the solver and is important for preliminary performance analysis and in situations where a full annulus mesh cannot be simulated due to time or computational cost constraints. The second purpose is to examine the effects of distorted flow within an experimental compressor fan developed

by the Air Force Research Lab (AFRL) using KCFD, specifically looking at stall inception. In order for future gas turbine engines to be designed to accommodate distorted inlet flow, the appropriate mechanisms for stall inception, as well as the effect on the performance of the compression system, must be understood.

The scope of this research included an analysis of KCFD's mixing plane capabilities using single passage meshes of the rotor and stator rows of the experimental fan. To study the effects of inlet distortion on stall inception, a 90° sector of a total pressure deficit was applied as a boundary condition at the fan inlet. This was meant to approximate the distortion pattern observed in fans using modern inlet geometries. Pressure data at the casing and internal flow data allowed for observation of the flow mechanisms leading to stall inception under clean and distorted inlet condition. The performance of the clean and distorted inlet cases were compared, including an analysis of the flow mechanisms leading to stall inception.

The research will be presented in the following manner: first, a background will be given on stall inception and methods for detecting it. Inlet distortion will be discussed, including common sources and methods for predicting its effects. A chapter on Methodology will discuss the procedures followed for mesh generation and simulation setup. Following that, a chapter will discuss the results of both the mixing plane validation efforts as well as the inlet distortion study. Finally, conclusions and avenues for future work will be presented.

## **CHAPTER 2. BACKGROUND & LITERATURE REVIEW**

This chapter provides an overview of the relevant topics and background needed to understand the research. Compressor stall will be discussed, followed by a summary of research regarding stall precursors. The contributions of computational fluid dynamics will be presented as they apply to efforts in understanding stall precursors. Following that, research into stall inception in high-speed fans will be summarized. Subsequently, sources of inlet distortion will be presented, followed by a review of the methods that have historically been used to test and model compressor performance under distorted inlet conditions. This review will highlight the strengths and shortcomings of such methods. Finally, a review of mixing plane analysis will be given to provide context to the mixing plane validation portion of this research.

### **2.1 Compressor Stall**

Performance maps define the operating boundaries of stall and choke, and also show total pressure ratio and efficiency at various operating speeds and mass flow rates. Figure 2.1 shows an example of a performance map, with the black lines representing various rotor speeds. The compressor is designed to operate at or near the dotted operating line shown. The stall regime is colored pink, and the choked flow regime is colored green. The point of compressor choke is aptly named because it is at this point that the compressor is physically choked; increasing the throttle setting cannot increase the flow rate. Compressor stall occurs at low flow rates when the flow separates from the blades, inhibiting the blades' ability to do work on the fluid. Once the point of stall is ascertained, designers define an operating point at some margin away from stall. This creates a buffer zone so that an engine doesn't go into stall if the inlet flow experiences sudden changes, such as distorted flow (see section 2.5). While a large stall margin provides a safety net, Figure 2.1 demonstrates that higher pressure ratios—corresponding to greater thrust—can

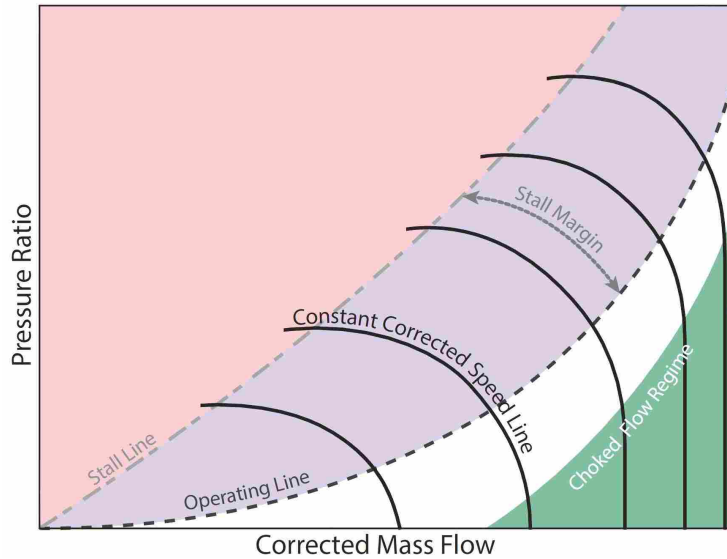


Figure 2.1: A typical performance map indicating operating margin used to avoid stalled conditions.

be obtained by operating closer to stall. Having an understanding of the events leading to stall inception would allow compressors to operate close to stall without risking stall inception.

A common form of stall in an axial flow compressor is rotating stall, where one or more stall cells form and rotate at some fraction of the rotor operating speed. Flow separation causes blockage, which diverts flow from one blade passage into the adjacent blade passages in a process illustrated in Figure 2.2a. Referring to Figure 2.2a, the flow diversion causes a higher incidence angle at the blade to the left of the stalled region and a lower incidence angle at the blade to the right. The blade passage to the left of the stall cell then goes into stall due to the higher incidence angle, diverting flow into the previously stalled blade passage at a lower incidence angle and bringing it out of stall. Thus, from the reference frame of the blades, the stall cell moves from right to left. A rotating stall cell can envelop multiple blade passages, as shown in Figure 2.2b. Extreme cases of stall can result in violent reversed flow through the compressor, known as surge [2]. The unsteady nature of stall and surge puts extreme loads on the compressor blades, reducing engine life.

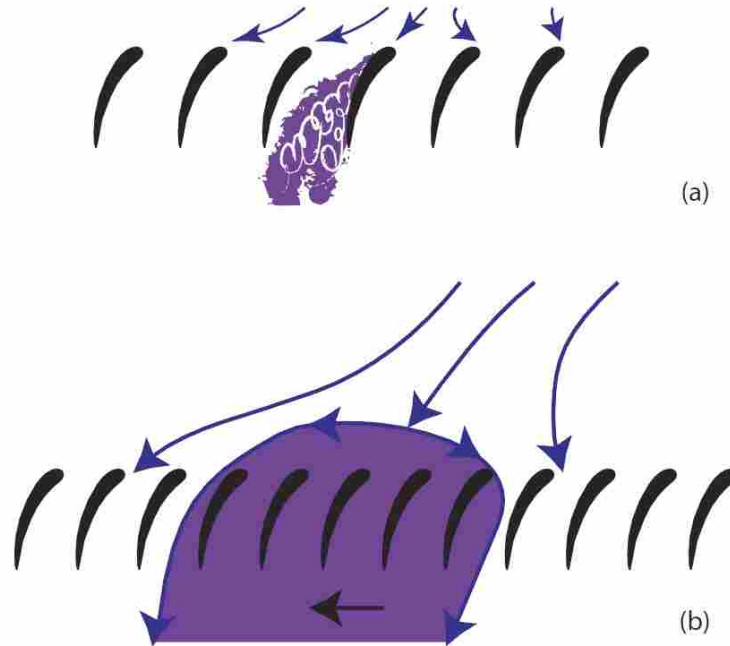


Figure 2.2: An illustration of flow diversion caused by the presence of a stall cell.

## 2.2 Stall Precursors

Because compressor stall defines the operating envelope and is so detrimental to the engine, decades of research have gone into identifying, predicting, and preventing the flow mechanisms that lead to stall inception [2]. Early on, two phenomena were observed in connection with stall in low-speed compressors, and have become known as modal and spike disturbances [3]. These phenomena have since been observed in high-speed compressors [4]. Since these phenomena preceded the development of rotating stall cells in a compressor, they became known as stall precursors. Modal disturbances are described by long length-scale tangential oscillations of flow properties, including axial velocity, upstream of the rotor [5]. Early studies observed that these tangential oscillations often grew in magnitude as the rotor approached stall, at which point stall cells began to form [6]. These stall cells formed gradually, not developing fully for several revolutions. An example of a modal disturbance generating a stall cell is seen in Figure 2.3. The six lines are six hot-wire data outputs at stations equally spaced around the annulus and located upstream of the rotor leading edge. The mode is detected as a slight oscillation in stations 1-6 and is shown by the dotted line. The modal disturbance rotates at approximately 20% of the fan speed. By the time that

it is picked up at station 2 for a second time, a stall cell is beginning to emerge inside the trough of the modal disturbance, shown as a more jagged data reading inside of the smooth trough. These observations led researchers to correctly conclude that modal disturbances can lead to rotating stall inception.

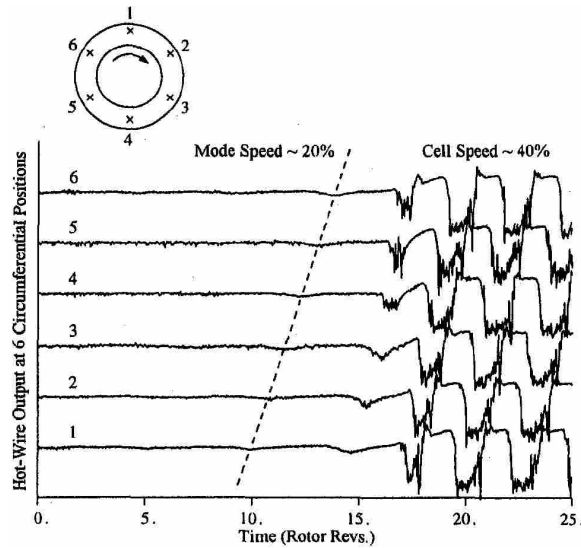


Figure 2.3: Hot-wire data demonstrating a modal disturbance preceding stall [3].

However, it was often the case that these modal oscillations were not observed in the moments leading to stall. Instead, stall cells were seen to develop rapidly (typically within 2-3 revolutions) following a sudden spike shaped drop in velocity measured on the casing near the rotor leading edge [6]. These spikes were of a much shorter length-scale than the modal oscillations, usually on the order of a few blade pitches. These rapid flow breakdown events subsequently became known as “spike” disturbances. An example of a spike disturbance generating a stall cell is seen in Figure 2.4. Here, an emerging spike is initially detected at hot-wire station 6, and subsequently observed at station 1. Based on the sensor data, the rotational speed of the spike is measured at about 70% of the rotational speed of the rotor. By the time it has been observed a third time at station 6, a stall cell is clearly beginning to form. By varying the tip clearance, Day discovered that tip clearance played a role in determining the type of precursor observed before stall cell formation [6]. Wider tip clearances resulted in modal precursors, while smaller tip clearances more commonly resulted in spike disturbances. Through detailed experiments, Camp

and Day formally defined the differences in physical characteristics between modal and spike disturbances [3]. In the same study, they determined that the type of stall precursor measured in a compressor could be affected by stage matching, which is affected by staggered blade rows and shaft speed. A mismatched stage occurs when shaft speed and/or staggered blades result in high rotor incidence, and high rotor incidence leads to the occurrence of spike disturbances. This study was also the first to show that spike and modal disturbances could both be present in the moments leading to compressor stall, as seen in Figure 2.5

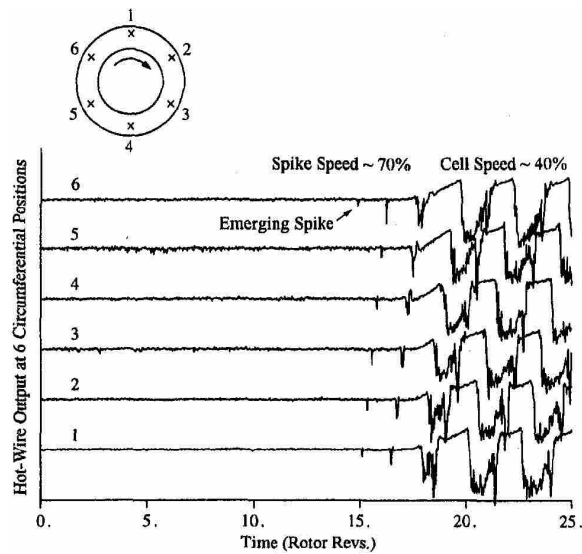


Figure 2.4: Hot-wire data demonstrating a spike disturbance preceding stall [3].

Although modal oscillations were effectively modeled and predicted early on, the flow mechanisms leading to spike disturbances proved difficult to explain. Part of the difficulty in understanding spike phenomena lay in the fact that early researchers were limited to observing the flow using pressure or hot-wire sensors at the casing, and could only make conjectures as to the three-dimensional structure of whatever was the cause of the observed spikes. Even so, a few facts pertaining to low-speed compressors were certain: spike disturbances originated at or near the leading edge of the tip of the rotor blade, they rotated more quickly than modal oscillations, and they rapidly developed into stall cells.



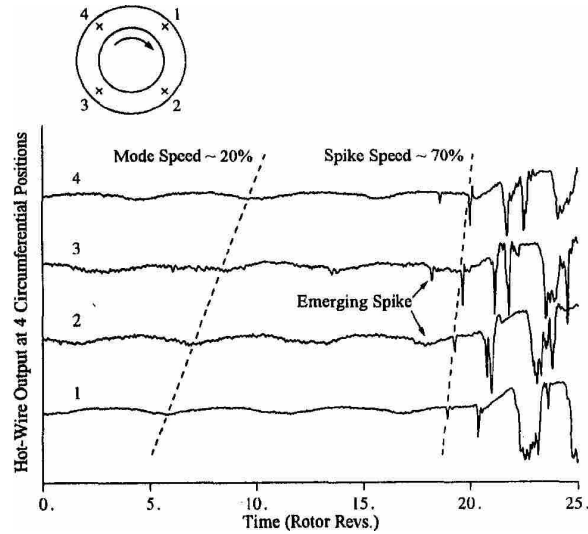


Figure 2.5: Hot-wire data demonstrating both modal and spike disturbances preceding stall inception [3].

### 2.3 Using CFD to Visualize Flow in Low-Speed Compressors

Beginning in the late 1990s it became feasible to simulate flow within a compressor stage using computational methods. With the advent of computational fluid dynamics (CFD), researchers have been able to peek into three-dimensional flow field and gain further insight into what causes spike disturbances. Early work done by Hoying, et al. used a rudimentary flow solver coupled with fluid models to observe the three-dimensional flow associated with spike disturbances in a low-speed compressor [7]. This early work established that, at least for the compressor simulated, the tip clearance flow resulted in flow blockages leading blades to stall. Tip clearance flow moves backwards (upstream) across the rotor tip, and occurs because of the high pressure gradient across the rotor row. This backward flow results in a vortex forming in the tip region at the boundary between positive and negative axial flow. The study observed that flow blockages happened when the tip clearance vortex became oriented perpendicular to the direction of axial flow, as shown in Figure 2.6, where the lines coming off of the blades represent the tip clearance vortex trajectories. At the design point, the vortex generated by tip clearance flow comes off of the blade leading edge at an angle that is not perpendicular to the axial flow direction. At stall, the orientation of the vortex is perpendicular to the axial flow direction, leading to blockage within the blade passage. Although this study was limited to single passage and partial annulus domains, this work

helped to establish that the perpendicular orientation of the tip clearance vortex as a criteria for the development of spike disturbances. This initial research was later expanded upon by Vo, et al., who used experimental and numerical results from a low-speed compressor to hypothesize that perpendicular tip clearance flow and the resulting leading edge spillage, as well as back flow near the tip of the trailing edge plane of the rotor, are the two criteria necessary to block the blade passage and cause spike-initiated rotating stall [8]. Leading edge spillage occurs when the tip clearance flow aligns with the leading edge plane resulting in some of the flow spilling across the leading edge of the adjacent blade.

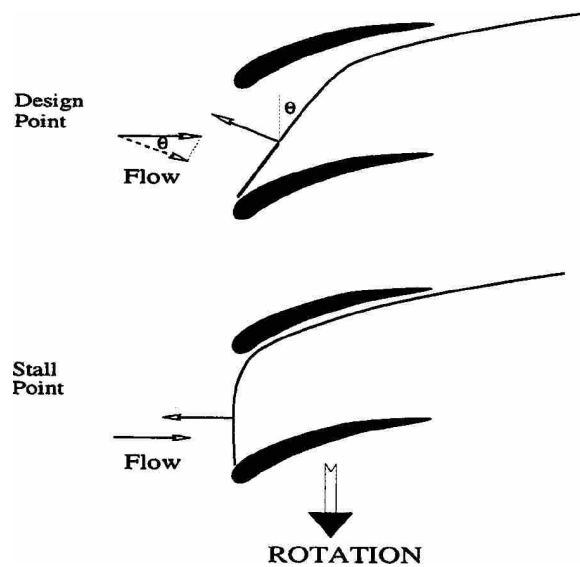


Figure 2.6: Tip leakage vortex trajectory at design and near stall [7].

Other studies have used CFD to demonstrate the existence of radial vortices at the leading edges of rotor blade tips in low-speed compressors, and effectively connected them with the experimental data used to identify spike disturbances [9, 10]. The existence of radial vortices was first theorized by Inoue, et al. based on experimental data and vortex theory [11]. Ultimately, the CFD simulations showed that spike disturbances are caused by flow separation due to high incidence at the leading edge of the rotor near the tip. This separated region develops into a vortical structure that attaches itself on one end to the suction side of the blade, and to the compressor casing on the other end. A diagram of this process is shown in Figure 2.7. The separated flow causes the blockage that is ultimately responsible for the pressure spike seen experimentally at the casing.

The point where the radial vortex attaches to the casing is responsible for the rapid pressure drop immediately adjacent to the spike. The researchers verified the existence of these radial vortices in simulations where there was no tip gap, leading them to conclude that although tip leakage flow has a role in spike stall inception when it is present, is not necessary for it [10]. This work disproved the earlier work done by Vo, et al. hypothesizing that the perpendicular orientation of the tip clearance flow was necessary for the appearance of spike disturbances [8], and established that the important metric is the incidence angle at the rotor leading edge.

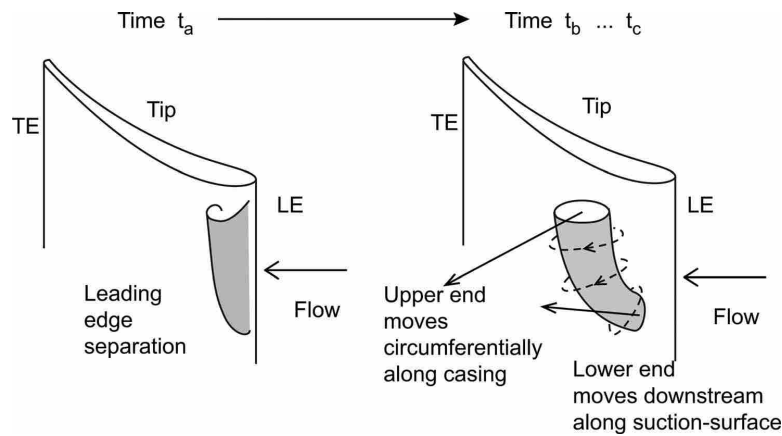


Figure 2.7: The formation and propagation of a spike-type stall precursor [10].

## 2.4 High-Speed Compressors

Thus far, the research presented has focused on low-speed compressors and fans. High-speed, highly loaded fans often exhibit different flow physics than low speed fans and compressors. For example, in high-speed, transonic compressors, oblique passage shocks form at the blade tips and add further complexity to the flow structures that exist there near stall. A general characteristic of high-speed fans is that these oblique shocks detach from the blade and become normal shocks as the compressor approaches stall. An example of this interaction is seen in Figure 2.8. In Figure 2.8a, the shock (shown by the red dotted line) is an oblique shock attached to the blade leading edge. In Figure 2.8b the shock has detached and moved upstream of the blade leading edge, becoming a normal shock. The flow phenomena present in transonic compressors may modify the previously established criteria for stall inception. This is especially likely since many of the flow

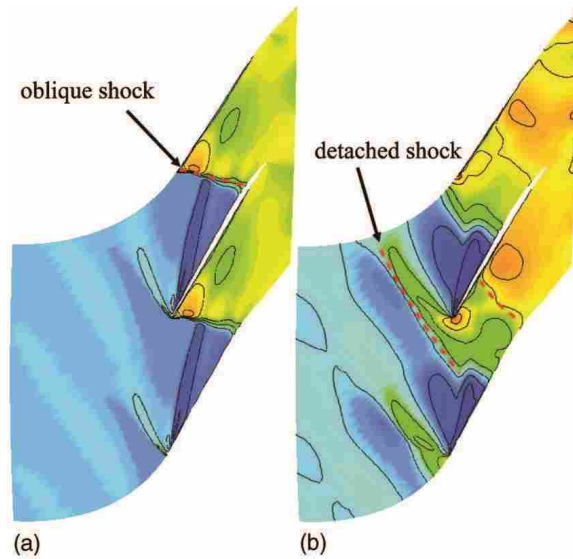


Figure 2.8: Pressure contours in rotor tip region at (a) stable operating conditions and (b) near-stall operating conditions [4].

structures attributed to stall inception in low-speed fans (tip gap flow, tip gap vortices, and radial vortices) occur in the vicinity of where a shock is located in a transonic fan.

Understandably, research in stall inception in transonic fans and compressors focuses on the role that a detached shock has in stall inception. In a groundbreaking study, Hah, et al. utilized CFD to study the role that tip clearance flow and passage shocks play in stall inception using a single-passage computational domain [12]. His analysis demonstrated the existence of a low momentum blockage region very close to the casing and immediately downstream of the detached shock on the pressure side of the blade. Near stall this low momentum region expanded across the leading edge and into the suction side of the blade, resulting in spillage into the adjacent blade passage and initiating stall inception. Although his results demonstrated that the tip clearance vortex moved toward the leading edge of the adjacent blade as the compressor approached stall, the tip clearance vortex ultimately did not become perpendicular to the axial flow direction to provide the mechanism for delivering tip clearance flow across the rotor leading edge. In a follow-up study, Hah, et al. used full annulus simulations to further define the role that the detached shock plays in spike-type stall inception [13]. Their findings showed that when the tip clearance vortex has moved sufficiently forward it interacts with the low momentum region caused by the shock and oscillates. It is established that when the oscillations become severe enough, the tip

clearance flow will move forward enough to initiate forward spillage of the tip clearance flow across the leading edge of the adjacent blade (leading edge spillage). Thus, a third criteria for spike-type stall inception was added for high-speed compressors: the detached shock had to move forward sufficiently to initiate this interaction. A study done by Chen, et al. also used CFD to establish the role played by both the detached shock and the tip clearance vortex in spike stall inception in a transonic compressor [4]. When simulating the NASA stage 35 compressor going into stall, he showed that spike disturbances leading to stall cell formation occurred shortly after the tip clearance vortex became perpendicular to the axial direction. Similar to Hah, et al. [13], Chen concluded that the interaction of the tip leakage vortex with the detached shock may play a part in the formation of a spike disturbance in transonic fans. Using experimental data from a high-speed compressor, Gannon and Hobson observed that low frequency flow instabilities had highest magnitudes where the normal shock interacted with the tip vortex and the blade suction surface, leading them to identify the tip vortex and the detached normal shock as flow phenomena responsible for causing the instabilities leading to stall cell formation [14]. They also attributed appearance of instabilities upstream of the rotor to the shock detaching from the rotor leading edge and moving upstream relative to the blade leading edge. To the author's knowledge the vortical structures theorized by Inoue, et al. and mentioned previously have not been observed in research involving transonic fans, indicating that the presence of a shock may prevent their formation.

## **2.5 Using a Variable Nozzle Boundary Condition**

Other efforts have been made to improve the capability of flow solvers to predict and simulate stall, including the breakdown in flow uniformity that leads to stall inception, the formation of stall cells, and the performance of the compressor during this process. These efforts included implementing correct boundary conditions, turbulence closure models, and grid types. Vahdati, et al. observed that single passage, steady-state simulations are less accurate near stalled conditions due to the unsteady nature of stall [15]. In the same study, he showed that implementing an extended variable nozzle geometry downstream of the compressor to throttle the flow prevented premature numerical stall and allowed for simulations at lower flow rates, more closely matching the flow rates seen experimentally. Davis and Yao also successfully implemented an extended domain with a nozzle to simulate the NASA stage 35 transonic compressor [16]. They showed that

CFD can predict stalled behavior within a compressor stage, and is capable of predicting both the long-wavelength (modal) and short-wavelength (spike) disturbances that lead to stall.

## 2.6 Sources of Inlet Distortion

Although many compressors are designed using an assumption of uniform inlet conditions, real operating conditions are often non-uniform. When aircraft house their propulsion system inside the fuselage, a pressure distortion is introduced in the inlet due to interactions with the frame of the aircraft. Furthermore, many modern aircraft implement S-duct inlets that generate flow non-uniformity that the propulsion system must deal with. An example of an S-duct is shown in figure 2.9, with the inlet on the left. S-ducts are known for contributing to swirl and unsteady flow distortions. The problem is further exacerbated when flow meets the inlet at an angle, such as during aggressive maneuvering, and takeoff and landing. This can cause inlet distortion even in commercial podded engine configurations without an S-duct. Thus, inlet distortion is something that all air-breathing propulsion systems must deal with. Since encountering inlet distortion affects the performance of a compressor, this problem is traditionally solved by operating with a large stall margin, limiting the work output of the compressor system.

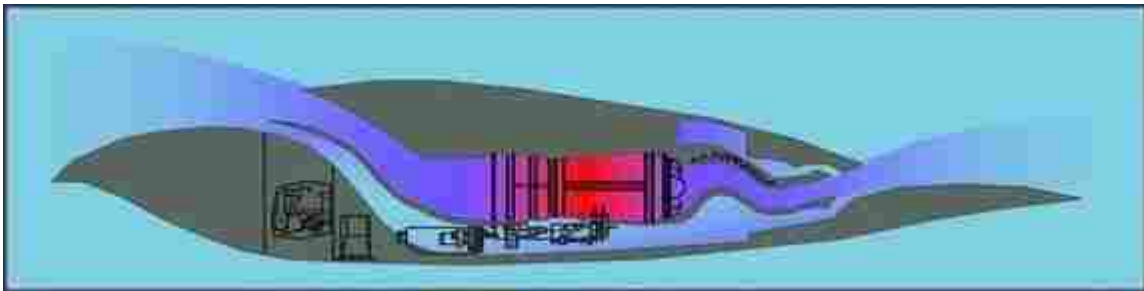


Figure 2.9: A schematic of the B2 propulsion system with S-duct inlet. Image courtesy of aviation.stackexchange.com.

## 2.7 Testing the Effects of Inlet Distortion

Though the causes of inlet distortion have been known for many decades, the methods and models used to analyze and design for distortion have evolved. Historically, distortion screens have

often been employed in test rigs to simulate total pressure distortion patterns [17]. An example of a distortion screen is shown in Figure 2.10. Though distortion profiles experienced during flight are often more complicated than the patterns created using screens, the use of screens has many advantages. Screens are often used as a “standard pattern” for evaluating compressor response to inlet distortion, allowing for much progress to be made in studying the effects of inlet distortion on compressor performance, stall margin, and stall inception. Tests using distortion screens have readily demonstrated the deleterious effects of inlet distortion on compressor performance and stall margin [18, 19]. Furthermore, the simplified distortion profiles generated by distortion screens are easier to define as a boundary condition in a flow solver while validating results. Although this is not the primary reason such screens are used, it is a beneficial consequence.

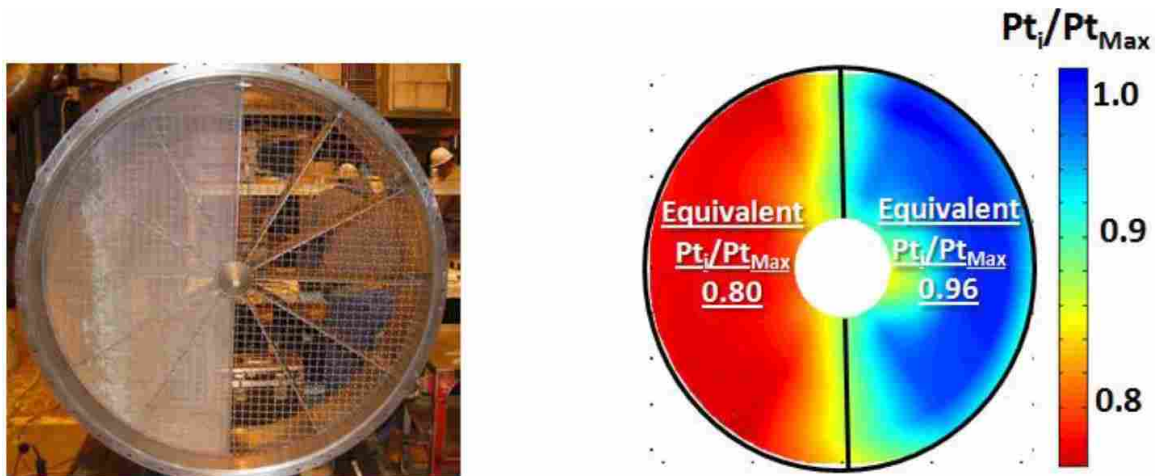


Figure 2.10: Example of a one-per-rev distortion screen and the corresponding total pressure distortion [20].

Distortion screens have also been used to define the effects of inlet distortion on stall precursors. As part of a study comparing various early warning approaches for stall avoidance, Höss, et al. utilized a screen consisting of a 90° sector to trip stall inception in a twin-spool turbofan engine [21]. This study found that in the presence of inlet distortion, the compressor exhibited spike-type stall inception at all operating speeds. Without the screen, spike-type stall precursors were only observed at lower speeds, and modal precursors were seen at higher speeds. In the presence of the screen, casing pressure data showed that flow instabilities were generated as the rotor blade passed from behind the screen into undistorted region, and that these were due to the higher

loading experienced by the blade as it passed into the undistorted region. These instabilities would be damped out within one revolution if the fan was operating far enough away from stall, but at the stall point the instabilities would grow into stall cells. This behavior in the presence of circumferential distortion screens was also observed by Day, et al. in the first rotor row of a multi-stage transonic compressor [22]. Furthermore, Spakovsky, et al. demonstrated that radial inlet distortion could also lead to spike-type stall precursors using a radial distortion screen [18].

## **2.8 Modeling the Effects of Inlet Distortion**

One of the earliest analytical models used to predict the effects of inlet distortion was the parallel compressor theory, which models a single compressor with a one-per-rev total pressure distortion as two parallel compressors, as shown on the right in figure 2.10, where each half of the inlet is assigned an equivalent total pressure. Each of the parallel compressors is modeled as operating with undistorted flow conditions, with the total pressure of one corresponding to the original compressor's distorted portion and the other corresponding to the undistorted portion [23]. Parallel compressor theory has seen various improvements and modifications since its inception in the 1950's and variations of this model are still in use today. While these simplified models allow for rapid analysis on the performance of the propulsion system, they have several limitations. For instance, they cannot be used to accurately model radial total pressure distortions [20, 24]. Furthermore, these models fail to provide insight as to how to design for robustness against such distortion.

High fidelity CFD has demonstrated the much needed capability of modeling the effects of inlet distortion on a propulsion system, including the formation of spike-type stall precursors [9,19,25–27]. CFD has also successfully predicted the secondary effects of inlet distortion, such as the generation of swirl and total temperature distortion within the system in response to distorted total pressure inlet boundary conditions, known as distortion generation [28–30]. Notably, CFD can accurately predict the phase shift that occurs between an inlet total pressure distortion and the total temperature distortion that is generated. However, current practices commonly involve at least two separate simulations: one including an entire airframe and a separate simulation for the propulsion system. These simulations are connected by an aerodynamic interface plane (AIP) where aerodynamic information from the airframe and inlet, including distortion, is passed to the



engine manufacturer. This distorted flow is then used as a boundary condition for the inlet plane when developing the gas turbine engine [17]. A central issue that comes from separating the two simulations in this way is that it becomes impossible for the coupled nature of the distortion to be captured. An engine designed and tested using current methods will often behave differently when integrated with the entire vehicle due to the system response propagating upstream as well as downstream, changing the flow conditions in the inlet to something that does not resemble the simulated flow conditions and resulting in performance different than the designers intended and possibly leading to premature fan or compressor stall. This coupling between the engine and airframe can likely only be predicted through more advanced numerical simulations or more sophisticated—and expensive—experimental setups.

## **2.9 Mixing Plane Analysis**

There are several methods that can be used for modeling internal flow in turbomachinery. The highest fidelity methods are time-accurate simulations, either periodic or full annulus. This high fidelity comes at a premium as full annulus methods are computationally demanding due to their large meshes, and time-accurate methods require very small time steps. An alternate approach utilizes mixing planes. Mixing plane simulations are 3D steady-state simulations and require only single-passage domains of each blade row, making them computationally much cheaper [31]. Figure 2.11 shows an example of a mixing plane simulation, where flow information from the rotor row is passed to the stator row through the mixing plane. At the mixing plane, flow properties such as total pressure, total temperature, and velocity profiles are circumferentially averaged and passed to the next blade row. Because flow properties are averaged in this way, mixing planes are usually not appropriate for analyzing inlet flow distortion, and also cannot simulate the unsteadiness associated with blade passing within the stage.

While not as accurate as a time-accurate, full annulus simulation, mixing plane analysis has a variety of uses. Since mixing planes can be used with single-passage grids, they represent a computationally cheap, medium-fidelity tool that facilitates the rapid analysis of compressor and turbine stages, and can be utilized to analyze multiple blade rows. Although mixing plane methods are typically not suited for analyzing inlet distortion, an exception to this is the case of radial distortion since radial variations are communicated across a mixing plane. Furthermore, a mixing

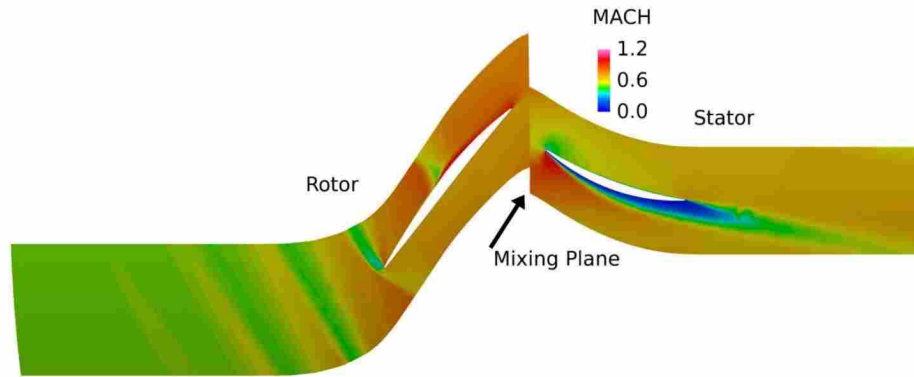


Figure 2.11: Example of a mixing plane simulation showing Mach number distribution for NASA Stage35 [32].

plane simulation can be used to establish grid independence for a single-passage mesh, which is subsequently circumvented to generate a full annulus mesh. Lastly, solutions from steady-state mixing plane simulations can be utilized to provide initial conditions for full annulus unsteady simulations, decreasing the overall number of iterations to reach a converged solution.

Kestrel’s mixing plane capabilities have been demonstrated by Nichols, et al [32]. He utilized mixing plane simulations in Kestrel to create performance maps for the NASA Stage35 compressor stage that closely match experimental and full annulus results. He also used mixing plane simulations to predict fan performance for a 3.5 stage military fan, demonstrating Kestrel’s ability to simulate several stages with mixing planes. Although the 3.5 stage performance predictions were not compared to experimental data, the observed performance trends were as expected, with the peak total pressure ratio occurring near stall and peak efficiency occurring close to the choked condition. Lastly, the study utilized mixing planes to analyze a compressor for a transonic wind tunnel. The static pressure rise vs. volume flow rate was predicted and the trends matched experimental data well, especially near the design point. All of these simulations assumed uniform inlet conditions.

The validation portion of this study seeks to add to the validation efforts that have already been made by providing a case using a high-performance compressor stage for comparison. Previously, only one of the published validation cases that used a high-performance compressor compared simulation data to experimental data. The current study serves as the second of such cases and strengthens the argument for wider use of Kestrel’s mixing plane capabilities. Lastly,

beyond plotting total pressure ratios and efficiencies, this study looks how radial profiles of total pressure and temperature generated using a mixing plane analysis compare to radial profiles from the time-averaged experimental data to provide a validation using more experimental data than previous validation studies. To the author's knowledge this is the first time that radial profiles resulting from Kestrel's mixing plane simulations have been compared to experimental data in this way.

## CHAPTER 3. METHODOLOGY

This chapter presents the methods used to perform this research. First the Kestrel solver will be presented, including its past and current capabilities. Next, the geometry used, known as Rotor 4, will be described. The final two sections will cover specific methods used for the validation study and the inlet distortion study, respectively. These sections will address mesh generation and grid independence, followed by a discussion of the inputs used in the Kestrel flow solver, including the boundary conditions implemented.

### 3.1 The Kestrel Flow Solver

The Computational Research and Engineering Acquisition Tools and Environments – Air Vehicles (CREATE-AV) Program was formed in 2008 as an initiative of the DoD’s High Performance Computing Modernization Program (HPCMP) to expedite the acquisition of aircraft through more efficient utilization of current and future computational resources and a streamlined work flow [33]. Out of that initiative came Kestrel, a multidisciplinary simulation tool for fixed wing aircraft. Key capabilities targeted during development included simulating multiple-body relative motion, fluid-structure interactions, complex aircraft maneuvers, and moving control surfaces [34]. Additionally, a central design goal of Kestrel since its inception has been to address the need to model and analyze an aircraft airframe, inlet, and engine simultaneously [1, 35].

The near-body flow solver that Kestrel utilizes, KCFD, was initially adapted from the Air Vehicle Unstructured Solver (AVUS) developed at Wright-Patterson Air Force Base in the 1990s. As its use cases diversified and required better robustness and engineering performance, its development diverged from AVUS, though the core algorithm is still similar. Furthermore, its developers added a propulsion integration component called Firebolt starting in 2013 with Kestrel v4 [31]. Kestrel v5, released in 2014, saw the addition of the capability to solve turbomachinery problems with rotating reference frames [1]. As part of this capability, the implementation of sliding inter-

face boundaries was updated. Although sliding interface boundaries had been implemented since Kestrel v2 [36], their intended uses were limited to modeling external control surfaces on aircraft, and thus did not stringently enforce conservation laws [37]. Additionally, interface boundaries were required to be planar. These limitations proved insufficient for modeling blade rows in turbomachinery, where performance predictions are very sensitive to the conservation of mass and total properties. The planar boundary requirement also proved unwieldy due to the close spacing between blade rows and swept geometries typically found in high-performance turbomachinery. The improved capabilities introduced in v5 rectified these issues by allowing for non planar interfaces boundaries. Further updates in Kestrel v6 allowed for data to be shared between different meshes, and interface planes could be defined between faces on two separate grids in the same simulation. The capability of KCFD to simulate the internal flow of turbomachinery using full annulus grids has been repeatedly demonstrated [1, 37, 38].

KCFD currently uses an unstructured, cell-centered, finite-volume solution scheme, with 2D and 3D capabilities [31]. Allowable cell types within the mesh include tetrahedra, prisms, pyramids, and hexahedra (quadrilaterals and triangles for a 2D mesh), with different cell types allowable inside the same mesh. Temporal and spatial accuracy can both be either first or second order. Available turbulence models include one-equation Spalart-Allmaras (SA) models, as well as Menter's baseline two-equation model (with and without the SST correction) [39]. A Delayed Detached-Eddy Simulation (DDES) variant is also available for all of the turbulence models.

All simulations were run using DoD supercomputing resources. Cores used per job ranged from 128 for the mixing plane simulations to 1024 for the full annulus simulations. Job times ranged from several hours for the mixing plane simulations to several weeks for the full annulus runs. Simulations closer to stall generally took more time to converge than simulations away from stall.

## **3.2 Geometry**

Because it is well documented experimentally, Rotor 4 was a prudent choice for use in these studies using the Kestrel flow solver. Its geometry is shown in Figure 3.1. Rotor 4 was developed in 1981 by GE Aircraft Engines as part of the Air Force Research Lab (AFRL) Parametric Blade Study (PBS) research program, which investigated methods of controlling shock strength, as well

as the coupling that exists between the cascade throat area and the suction surface shape [40, 41]. Its characteristics include a rotor tip speed of 1500 ft/s at design conditions, a notably low curvature on its suction surface, and a reduced throat area. Additional design parameters are outlined in Table 3.1.

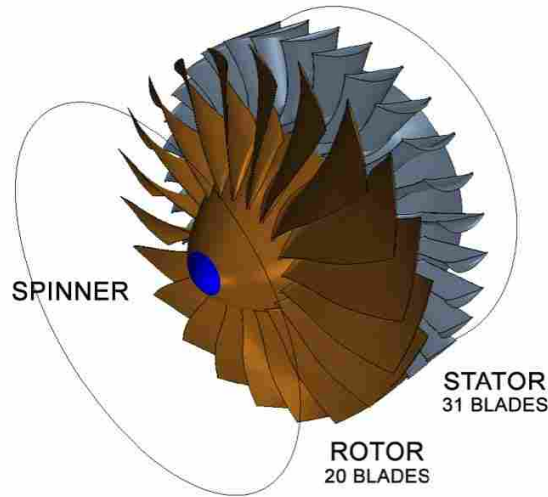


Figure 3.1: Rotor 4 geometry.

Table 3.1: Rotor 4 stage design parameters [27].

Design Parameter	Value
Running Tip Clearance	0.020-0.025 in
Flow Rate	60.77 lbm/s
Pressure Ratio	1.988
Rotor Efficiency	89.62%
Rotation Speed	20,200 RPM

The performance of Rotor 4 has been extensively documented experimentally at AFRL [40], and the experimental data obtained was used in this research. As part of the PBS report, the stage efficiency and total pressure ratios were measured at a range of corrected mass flow rates to create performance maps at various rotor speeds. In the same study, total pressure and temperature data were taken along the stator leading edge and again using rakes approximately 0.7 chord lengths downstream of the stator's trailing edge at various rotor speeds for near stall, peak

efficiency, and near choke conditions. This data was used as part of the mixing plane validation study to generate radial profiles of total pressure and temperature for comparison to KCFD results. Research at AFRL continues with Rotor 4 to understand its stall characteristics under diverse operating conditions, including with distortion caused by curved inlets [27].

### **3.3 Mixing Plane Validation Study Methods**

The methods specific to the mixing plane validation study are outlined below. The computational domain used is described in detail, followed by a description of the single-passage volume mesh generation process. Grid independence is verified. Later, the solver inputs and boundary conditions are discussed.

#### **3.3.1 Computational Domain**

The computational domain used for the mixing plane simulation included single passage grids of the rotor and stator rows. These domains are shown in Figure 3.2. After doing some preliminary simulations using a spinner mesh, total pressure ratio and efficiency values were seen to change by an average of less than 0.9% and 0.7%, respectively, along the entire speedline, indicating that it doesn't significantly affect the results. Since the purpose of utilizing single-passage domains with mixing planes is to generate rapid results, the spinner region shown in Figure 3.1 was excluded to simplify the simulations and decrease the time needed for results to be produced.

#### **3.3.2 Mesh Generation**

From the geometry files provided by AFRL, Pointwise was used to generate unstructured, single-passage volume meshes for the rotor and stator. Meshing best practices were followed, ensuring a denser grid resolution near the leading and trailing edges of the blades, as well as near the casing and hub. Quad-dominant structured surface grids comprised the upper and lower surfaces of the blades, the inflow and outflow boundaries, and the two periodic boundaries. All other surface grids consisted of triangles and quads, and were initialized using an advancing front orthogonal algorithm. Following the initialization of the surface grids, an advancing layer algorithm called "T-Rex" was used at the blade, casing, and hub surfaces to generate the anisotropic boundary layer

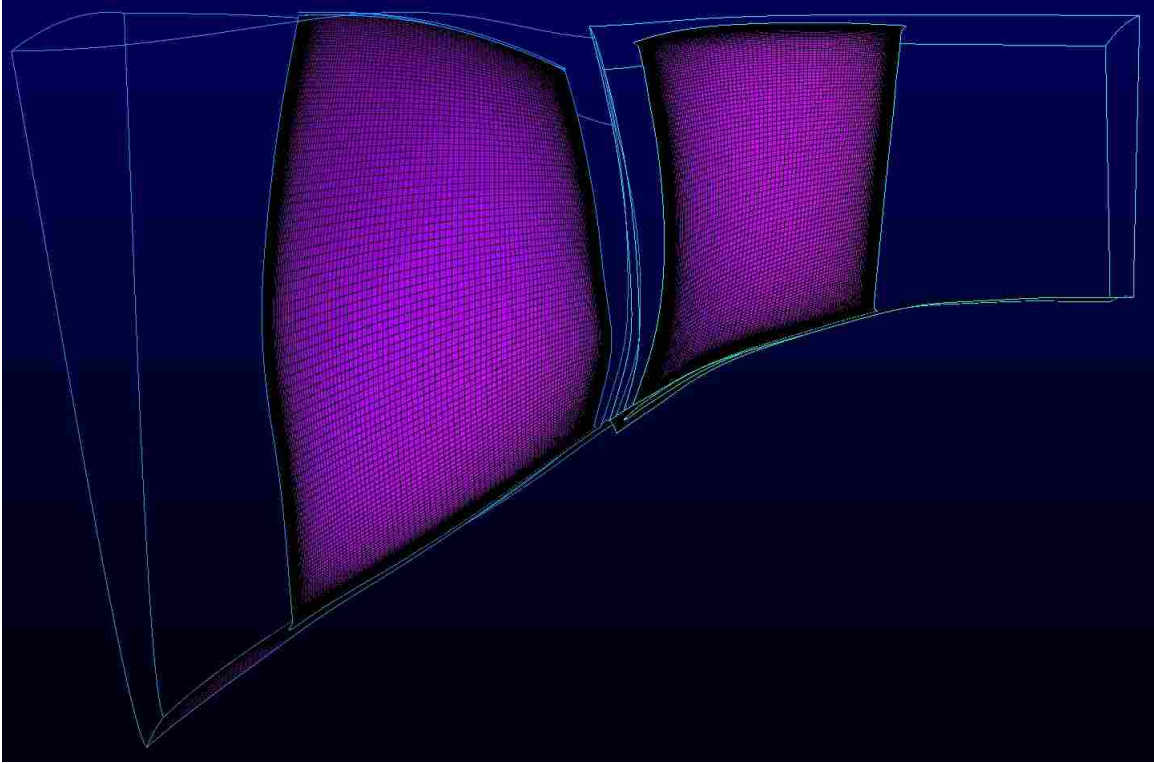


Figure 3.2: The computational domains used in the mixing plane validation study.

portion of the mesh. Using this boundary layer meshing utility allowed the anisotropic boundary layer, comprised of triangular prisms and hexahedra, to naturally transition to anisotropic tetrahedra away from the walls. Within the boundary layer a minimum of 15 prism layers was enforced in order to properly resolve the boundary layer at the walls and a prism layer growth rate of 1.2 was used. A maximum of 40 prism layers was also specified. A wall  $y^+$  value of less than one was achieved with an initial wall spacing of  $1.0E-4$  inches. Cross sections of the volume meshes are shown in Figures 3.3 and 3.4, where blue indicates tetrahedra, orange indicates hexahedra, green indicates prisms, and yellow indicates pyramids.

### 3.3.3 Establishing Grid Independence

A grid independence study was performed so that the results presented in this study could be considered meaningful. After creating the initial mesh, a refinement was done that increased the number of nodes along every edge by a factor of 1.5. Boundary layer spacing was also refined by a factor of 1.5. Cross sections of the refined meshes are shown in Figures 3.5 and 3.6, and the cell



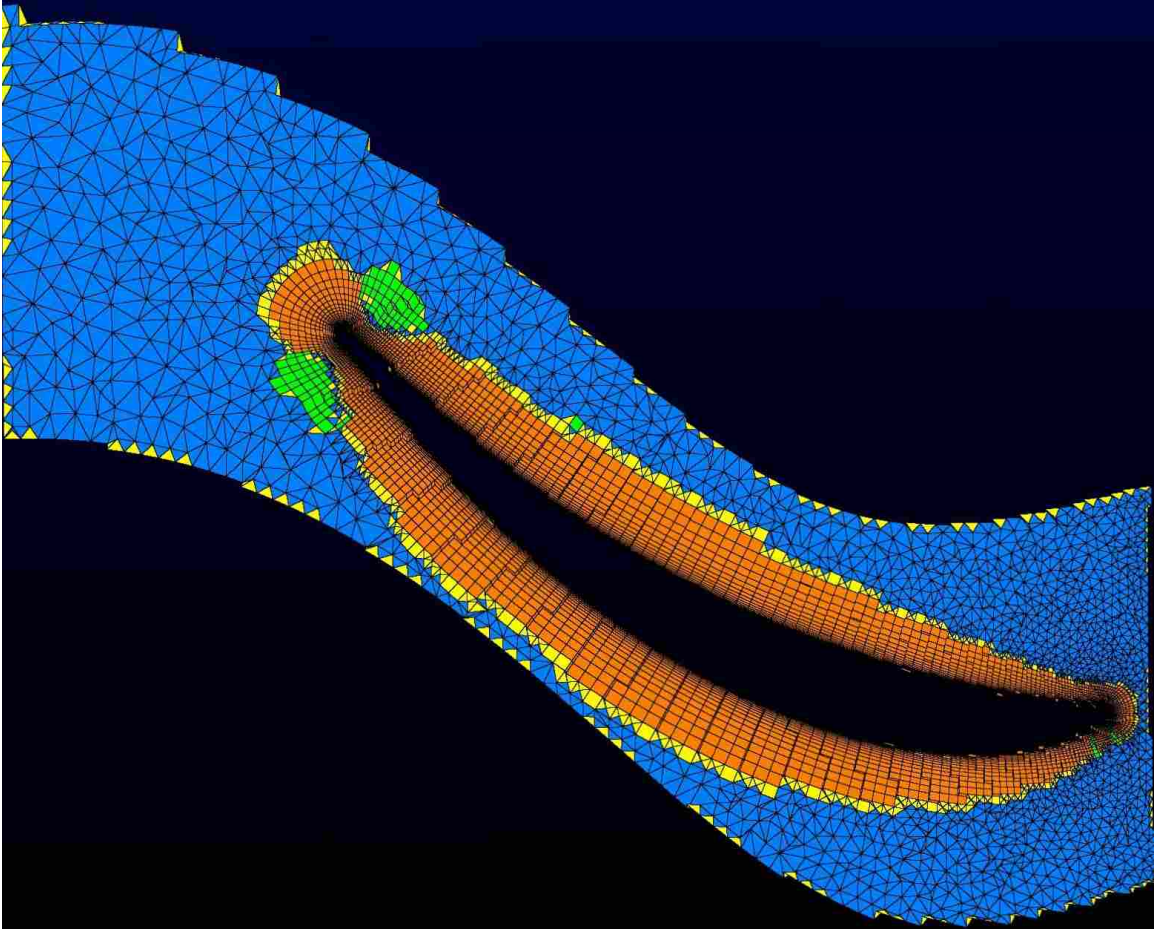


Figure 3.3: Cross section of the rotor single-passage volume mesh. Blue indicates tets, orange indicates hexes, green indicates prisms, and yellow indicates pyramids.

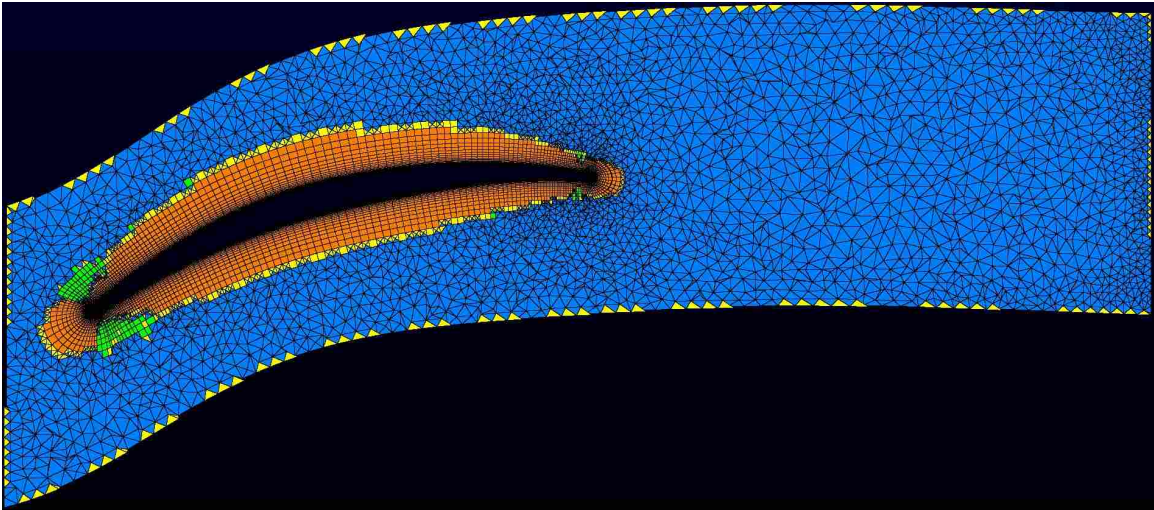


Figure 3.4: Cross section of the stator single-passage volume mesh. Blue indicates tets, orange indicates hexes, green indicates prisms, and yellow indicates pyramids.

counts for the rotor and stator before and after the refinement are given in Table 3.2. After running the simulation at a flow rate of 61 lbm/s (near design) using the refined mesh, the pressure ratio of the stage was shown to change by 0.41% and the efficiency changed by 0.70%, giving good reason to believe that the initial mesh spacing was sufficient. Although the change in efficiency between the two grids was not tracked, the small difference in total pressure ratio justified the continued use of the initial mesh for the remainder of the validation study.

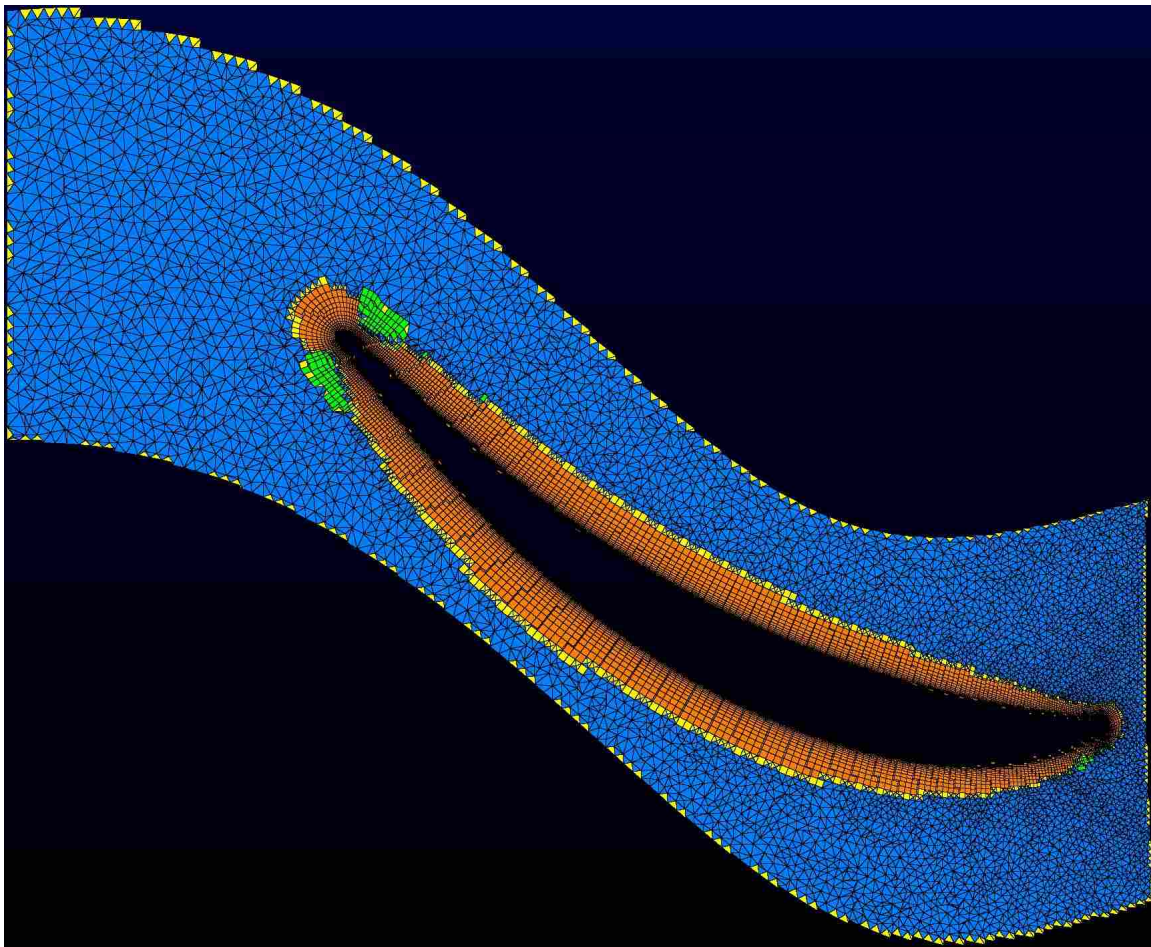


Figure 3.5: Cross section of the rotor single-passage refined volume mesh. Blue indicates tets, orange indicates hexes, green indicates prisms, and yellow indicates pyramids.

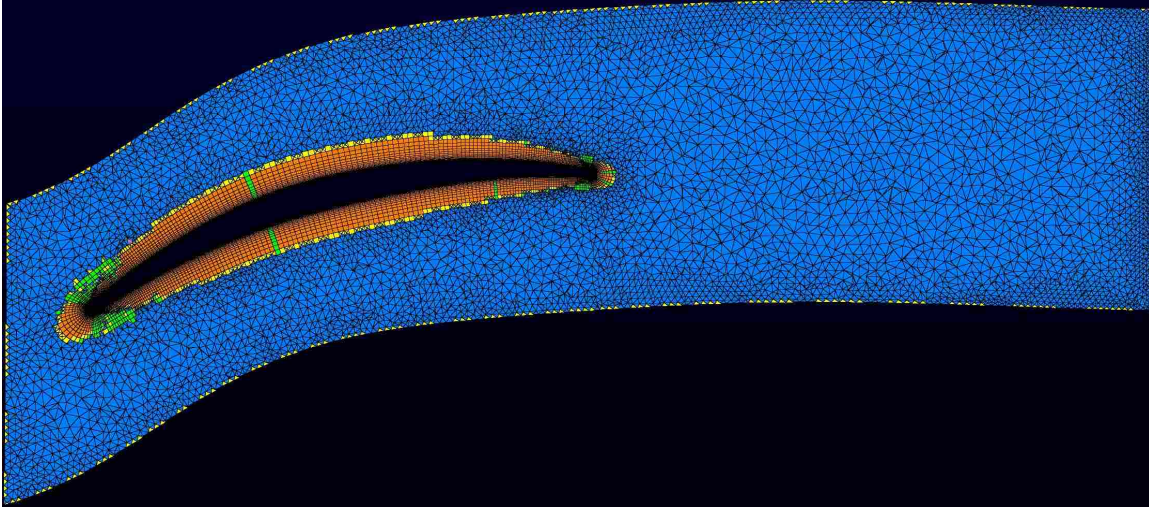


Figure 3.6: Cross section of the stator single-passage refined volume mesh. Blue indicates tets, orange indicates hexes, green indicates prisms, and yellow indicates pyramids.

Table 3.2: Cell count before and after refinement.

	Before Refinement	After Refinement
Rotor	2,795,687	8,522,025
Stator	3,108,957	8,180,399
Pressure Ratio at 61 lbm/s	2.0159	2.0242
Efficiency at 61 lbm/s	85.87%	85.18%

### 3.3.4 Solver Inputs and Boundary Conditions

The mixing plane runs were quasi-steady Reynolds-averaged Navier-Stokes (RANS) simulations, with the rotor row being computed in a non-inertial, rotating reference frame. The use of a rotating reference frame greatly reduces computation times compared to solving in an inertial reference frame with a rotational rigid-body motion. These quasi steady simulations were set up with a time step of 0.001 seconds per iteration, with two sub-iterations per recommendations made by the KCFD development team at AEDC. Simulations were run at 100%, 95%, and 90% design speed. The turbulence model used was Menter’s baseline two-equation model [39]. This decision was made based on previous studies done showing that the Menter model typically makes better predictions than the Spalart–Allmaras model in turbomachinery applications within KCFD [1]. Temporal accuracy was set to first-order accuracy, and spatial accuracy was set to second-order

accuracy, with advective temporal damping set to 0.05. The remaining KCFD parameters were left at their default settings.

The independent parameter used to vary stage performance was a mass flow rate boundary condition at both the rotor inlet and stator exit. The inlet boundary on the rotor was defined as a uniform source boundary condition, where total pressure, total temperature, and mass flow rate are specified by the user. For these simulations, sea level standard temperature and pressure were used at the inlet. The mass flow rate was calculated as a fraction of the full annulus flow rate that depended on the number of blades in the blade row. For a single-passage simulation:

$$[Flow\ Rate]_{single-passage} = \frac{[Flow\ Rate]_{full\ annulus}}{N_{blades}} \quad (3.1)$$

The outflow boundary of the stator row was defined as a sink with limited mass flow, allowing for the specification of an upper Mach number limit at the outflow boundary. Specifying an upper Mach number of 1.0 facilitated solver convergence by preventing supersonic flow at the exit while the solver determined the exit pressure needed for the mass flow rate desired. The mass flow rate for the sink boundary conditions was defined in the same manner as for the source. Varying the mass flow rates at the stage inlet and exit varied stage performance, allowing for the generation of a speed line.

### 3.4 Inlet Distortion Study Methods

The methods specific to the inlet distortion study are outlined below. The computational domain used is described. Furthermore, the full annulus volume mesh generation process is described and the justification for grid independence is discussed. Later, the solver inputs and boundary conditions are addressed.

#### 3.4.1 Computational Domain

The computational domain used for the distortion study was extended upstream and downstream from what was used for the mixing plane validation. The spinner region was added upstream of the rotor row. A blunted nose was used to more accurately mimic the experimental setup used for the PBS report [40]. Following the methodology of Vahdati, et al., a duct extension followed

by a variable nozzle was added downstream of the stator region [15]. The extended computational domain is shown in Figure 3.7, where the casing has been made transparent. Not only did including the nozzle more closely resemble the experimental setup, it also prevented premature numerical stall. This process is more fully covered in section 3.4.4 below.

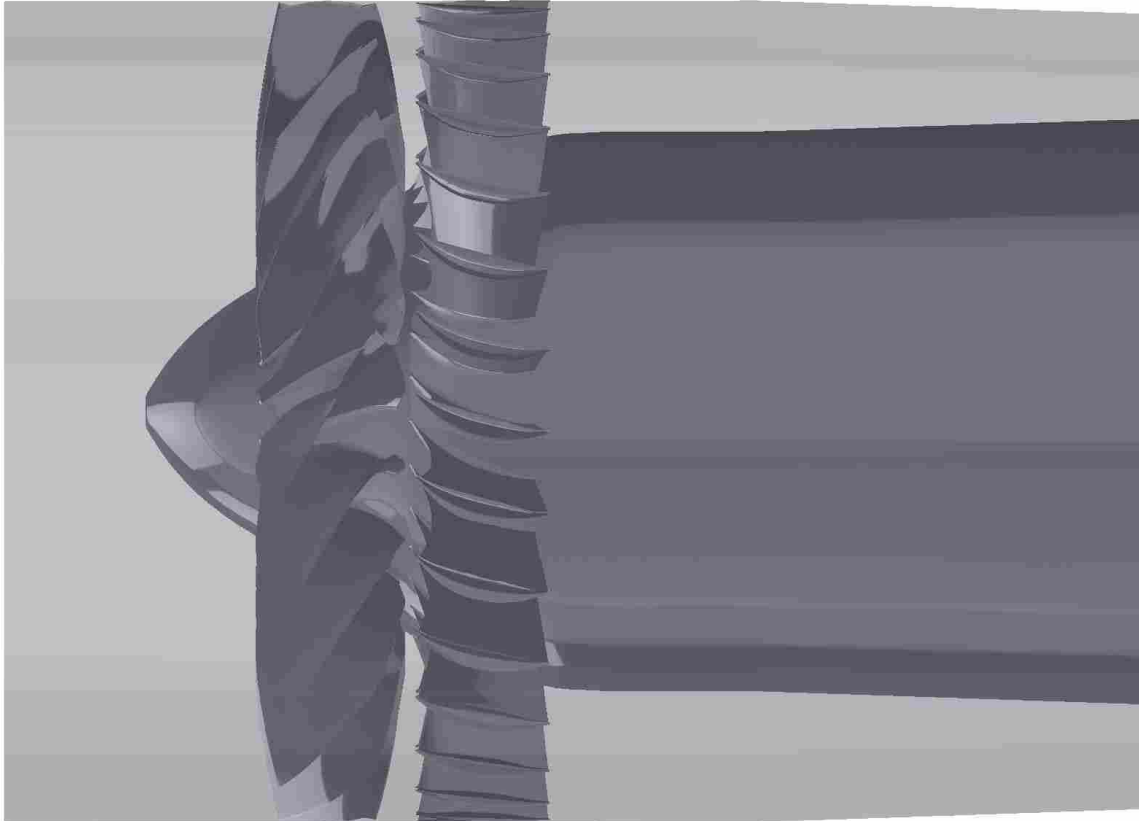


Figure 3.7: The full annulus computational domain used for the inlet distortion study.

### 3.4.2 Mesh Generation

A short time after beginning the inlet distortion study, it was discovered that the geometries used for the mixing plane validation study were reversed in orientation from the physical Rotor 4 geometry tested. While this would not affect the results of that study, the meshes were remade with the correct orientation for the full annulus simulations. The same meshing procedures were followed as with the mixing plane validation mesh generation, namely, a dense grid resolution was applied near the leading and trailing edges of the blades, as well as near the casing and hub. Within

the boundary layer a minimum of 15 prism layers was enforced in order to properly resolve the boundary layer at the walls and a prism layer growth rate of 1.2 was used. A  $y^+$  value of less than one was achieved with an initial wall spacing of  $1.0E-4$  inches. The maximum allowable prism layers was decreased to 30 in the boundary region, and other spacing adjustments were made to improve overall mesh quality. These adjustments slightly increased the cell count for both meshes. Cross sections of the single passage grids are shown in Figures 3.8 and 3.9. Once again, blue indicates tetrahedra, orange indicates hexahedra, green indicates prisms, and yellow indicates pyramids. For comparison, cell counts of the old and new single-passage grids are presented in Table 3.3.

Because of the simplicity of the extended nozzle domain, a structured mesh was used when creating the mesh for the nozzle. The grid spacing at the interface with the stator mesh was matched between the two domains, and a wall  $y^+$  of less than 1 was again achieved within the nozzle. Using

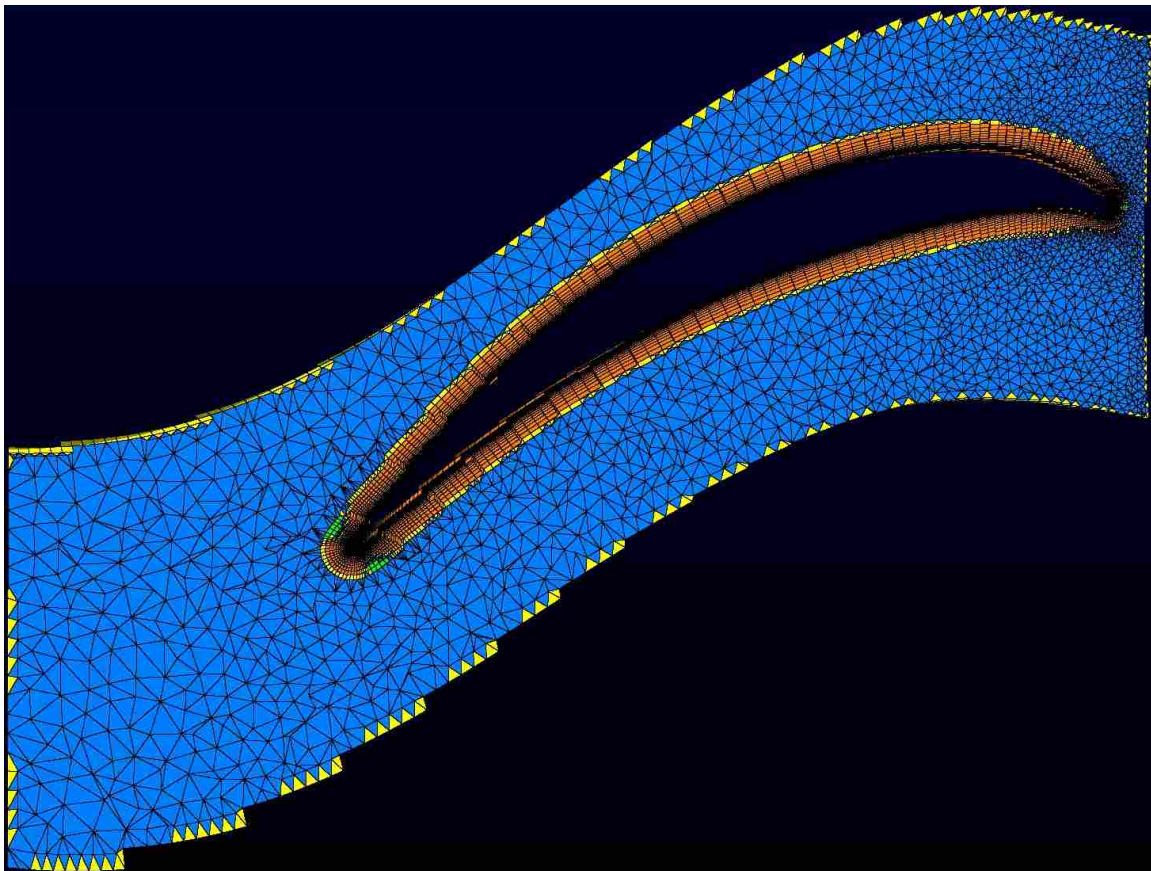


Figure 3.8: Cross section of the rotor single-passage volume mesh. Blue indicates tets, orange indicates hexes, green indicates prisms, and yellow indicates pyramids.

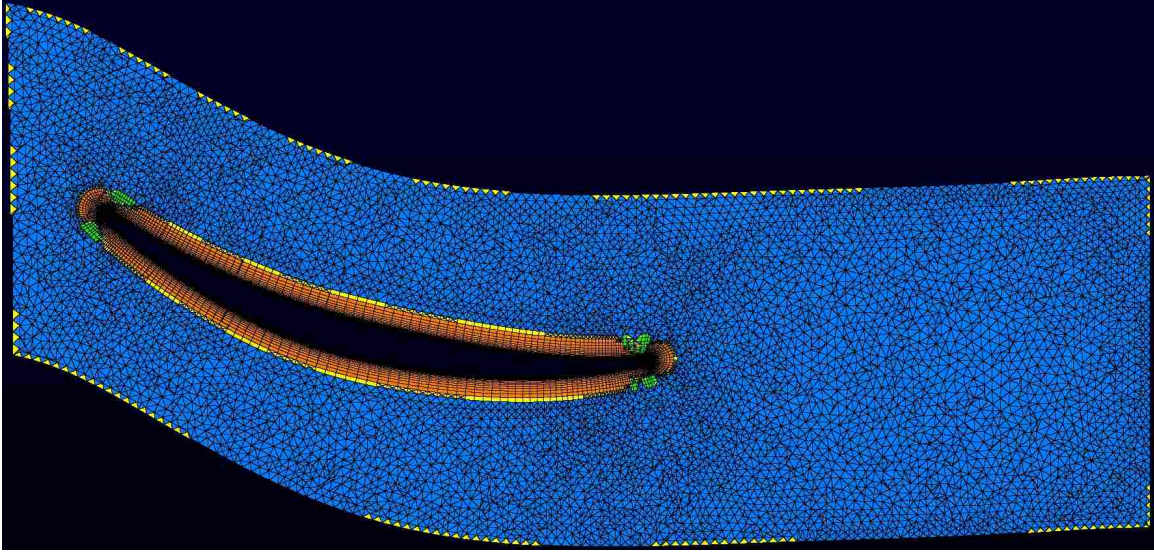


Figure 3.9: Cross section of the stator single-passage volume mesh. Blue indicates tets, orange indicates hexes, green indicates prisms, and yellow indicates pyramids.

Table 3.3: Cell counts of old and new meshes.

	Old Mesh	New Mesh
Rotor	2,795,687	3,053,946
Stator	3,108,957	4,461,720

a variable converging nozzle for the exit boundary conditions necessitated generating multiple nozzle of slightly differing exit areas. A script was used that could easily generate new meshes for the slightly different nozzle geometries.

The grid generation practices used for the spinner region emulated those used for the rotor and stator regions. At the walls a prism layer growth rate of 1.2 was enforced, as well as a  $y^+$  value of less than 1. Furthermore, cells near the stagnation point on the spinner were refined to capture the high gradients encountered there. A cross section of the spinner mesh is shown in Figure 3.10, where blue indicates tetrahedra, orange indicates hexahedra, green indicates prisms, and yellow indicates pyramids.

After the single-passage grids were made, full annulus meshes were generated using Carpenter, a mesh manipulation tool that is part of Kestrel [35]. A utility within Carpenter allows the user to circumvolve the mesh of a partial annulus, in this case a single blade passage, to generate

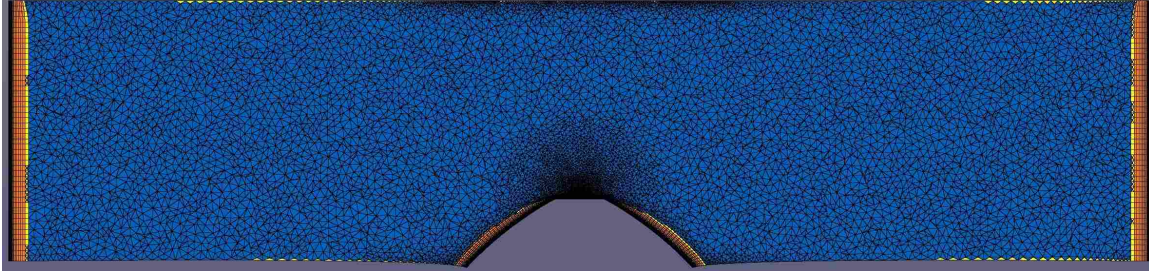


Figure 3.10: Cross section of the blunted spinner volume mesh. Blue indicates tets, orange indicates hexes, green indicates prisms, and yellow indicates pyramids.

a full annulus mesh. Utilizing Carpenter after creating a high quality single blade passage mesh in Pointwise ensured that the full annulus mesh would also be of high quality. For this reason, Carpenter was used to generate the rotor and stator rows, as well as the nozzle, for this study. The spinner region was not generated using the Carpenter utility because it was generated in Pointwise as a full annulus region. The full annulus cell counts for the spinner, rotor, stator, and nozzle regions are shown in Table 3.4.

Table 3.4: Cell counts of full annulus region meshes.

	Spinner	Rotor	Stator	Nozzle	Total
Cell Count	7,886,290	61,078,920	138,313,320	21,873,600	229,152,130

### 3.4.3 Grid Independence

Grid independence for the rotor and stator grids used in this study was established as part of the mixing plane validation study. Although the grids used in this study were changed to reflect proper blade orientation as discussed, similar meshing procedures were followed as before, and the grids used for the rotor and stator rows in this study ultimately were slightly denser than those used in the mixing plane validation study. The grid density in the spinner and nozzle regions was consistent with the rotor and stator regions. Furthermore, the flow regions of import in this study exist only in the rotor and stator regions. Lastly, the cell counts for the grids used in this



study are consistent with cell counts in grids used in other studies with full annulus computational domains [4, 10]. For these reasons, no grid independence study was performed as part of this study.

### 3.4.4 Solver Inputs and Boundary Conditions

The time-accurate, unsteady Reynolds-averaged Navier-Stokes (RANS) simulations were initialized from steady RANS simulations employing the rotating reference frame (RRF) methodology used in the mixing plane validation study. Using RRF simulations initializes the flow field quickly and decreases the required time steps needed in the unsteady simulations to reach convergence. The unsteady simulations were set up with a time step of  $1.03135313531e-6$  seconds per iteration (8 iterations per degree and 2880 iterations per revolution), with 1 sub-iteration. All simulations in this study were run at the design speed (20,200 RPM). The turbulence model used was Menter's baseline two-equation model [39]. This decision was made based on previous studies done showing that the Menter model typically makes better predictions than the Spalart-Allmaras model in turbomachinery applications within KCFD [1]. Temporal and spacial accuracy were both set to second-order accuracy, with advective temporal damping set to 0.05. The remaining KCFD parameters were left at their default settings.

The inlet boundary condition used for the clean inlet flow simulations was a uniform source boundary condition, where total pressure and total temperature were specified at sea level conditions ( $518.7^{\circ}$  R and 14.69 psi, respectively). A mass flow rate was not specified. The distorted inlet flow simulations also used a source boundary condition, where the distortion input method was used.

The distortion file specified a  $90^{\circ}$  sector with a 15% total pressure deficit, meaning that the total pressure within the distorted region is 85% of the total pressure in the undistorted region. This is a pattern similar to those observed in experiments using curved inlet ducts [27]. A contour plot of the distorted total pressure boundary used is shown in Figure 3.11. The total temperature at the inlet was uniformly  $518.7^{\circ}$  R. Similar distortion patterns have been employed in previous inlet distortion research using Rotor 4 [42, 43]. A mass flow rate was also not specified at the inlet boundary for the distorted inlet simulations.

The exit employed a static pressure outlet following an extended nozzle domain. The static pressure at the nozzle exit is set equal to ambient conditions at sea level (14.69 psi). At mass

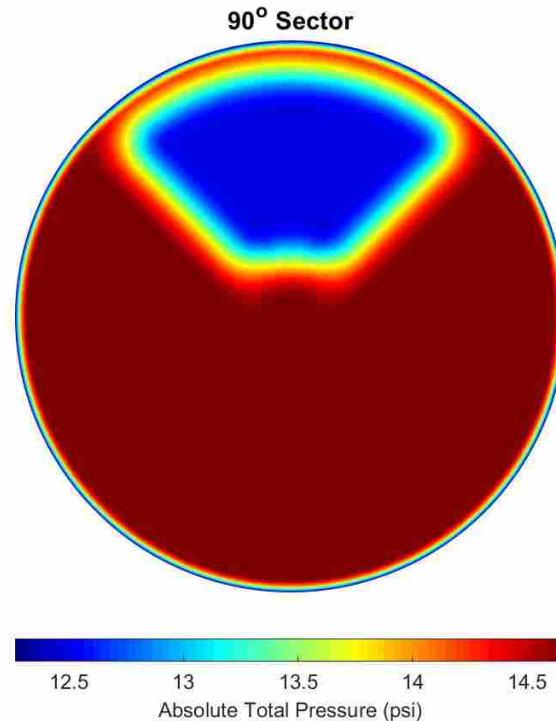


Figure 3.11: Contour plot of the distorted total pressure boundary condition used for this study.

flow rates away from stall, the nozzle is choked, but at lower flow rates near stall the nozzle does not choke. In the cases where the nozzle does not choke, the flow rate adjusts so that the static pressure drop that occurs through the nozzle matches the exit static pressure at the nozzle exit. In this way, the exit area determines the flow rate through the stage whether or not the nozzle is choked. Utilizing an extended nozzle domain in this manner more closely mimics experimental rigs and has been shown to better approximate test conditions and avoid premature “numerical stall”, allowing for CFD to predict near-stall performance at flow rates closer to what is measured experimentally [15, 16].

Sliding interface planes were used at the spinner-rotor interface, the rotor-stator interface, and the stator-nozzle interface. Within KCFD, the sliding interface boundary condition allows two co-located boundary patches to share information regardless of any relative motion between the two connected boundaries. A prescribed rotational motion of 121,200 deg/sec (20,200 RPM) was applied to the grid containing the rotor blade row. Although the same motion could have been

applied to the spinner grid, it was physically equivalent and computationally cheaper to instead apply a rotational wall velocity of 20,200 RPM to the wall boundary representing the spinner nose.

## **CHAPTER 4. MIXING PLANE VALIDATION RESULTS**

This chapter will focus on the results from the mixing plane validation study. A bug in the mixing plane implementation that introduced unsteadiness at the mixing plane interface is demonstrated and rectified. An issue involving a discontinuity in the values extrapolated across the mixing plane is addressed. Performance maps showing the total pressure rise across the stage and the efficiency at various flow rates are displayed and discussed. Finally, the span-wise profiles of the total pressures and total temperatures at two axial locations generated using KCFD are shown and compared to experimental data for three distinct operating points at 100%, 95%, and 90% design speeds.

### **4.1 Unsteady Flow Across Mixing Planes**

Initial mixing plane simulations run using Kestrel v8.0 showed unsteadiness being introduced at the mixing plane that propagated downstream, demonstrated in Figure 4.1. This unsteadiness is introduced at the Stator Inflow boundary, which is the downstream face of the mixing plane pair. The oscillatory behavior persists downstream and is noticeable at the Stator Outflow boundary. The issue was brought to the attention of Robert Nichols and fixed starting with Kestrel v8.1. The improved mixing plane implementation is demonstrated in Figure 4.2.

### **4.2 Discontinuity Across Mixing Planes**

When obtaining total pressure and temperature ratios, a slight discontinuity in total pressure across the mixing plane was noted. An example of this can be observed in the convergence plot in Figure 4.2. Here, the data representing Rotor Outflow and Stator Inflow total pressures should ideally be equivalent, but there is a small jump in total pressure across the mixing plane that needs to be accounted for. Although the difference is within 1% for all cases, a solution recommended by Robert Nichols at AEDC was used to calculate the pressure ratio for the stage using Equation

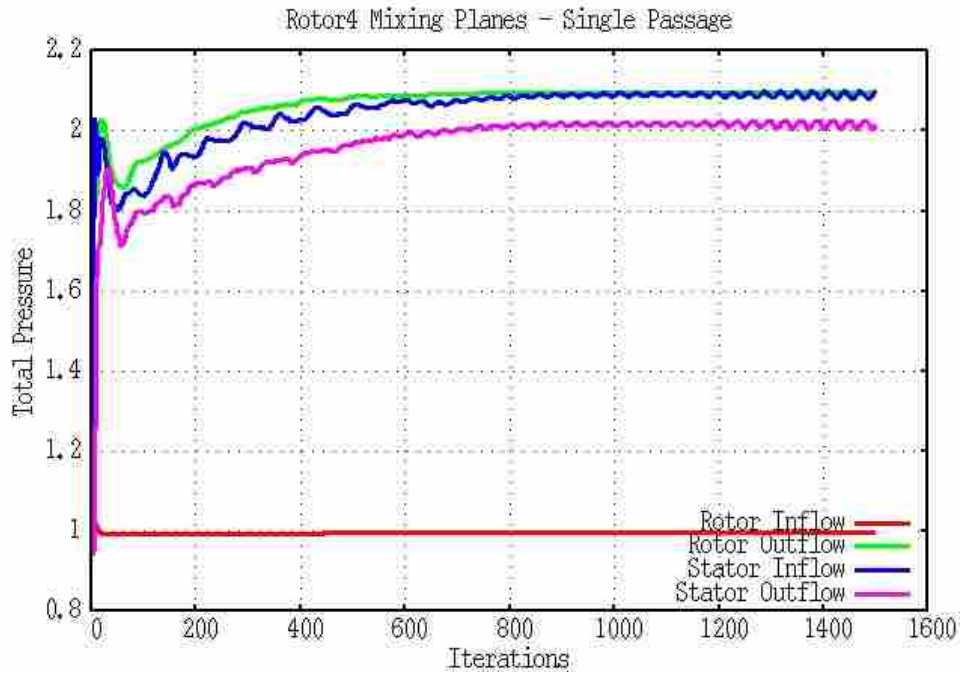


Figure 4.1: Rotor 4 stage total pressure convergence demonstrating oscillatory behavior introduced at the mixing plane.

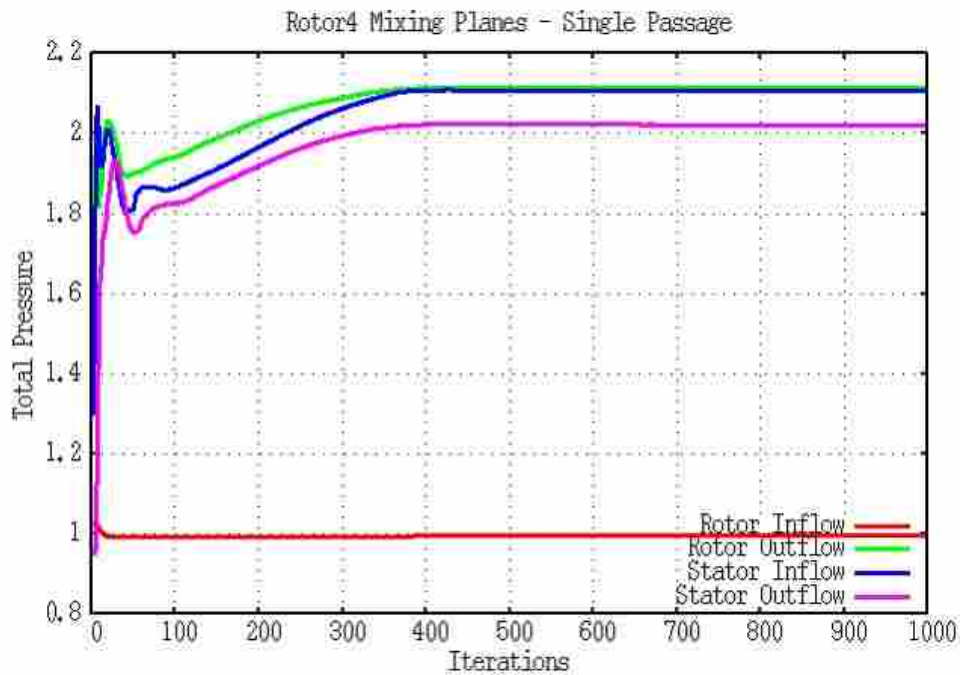


Figure 4.2: Rotor 4 stage total pressure convergence after v8.1 fix.

4.1. This allowed for the solution to be unaffected by any discrepancies across the mixing plane. A similar discontinuity was seen in total temperature and accounted for in the same manner. This issue was attributed to the data reduction that takes place when replacing a 3D flow field with a radial profile at the mixing plane boundaries, and has typically been less noticeable in past mixing plane applications using Kestrel [32]. Furthermore, when using KCFD jumps are typically worse for mixing planes that have flow going in both directions across the same mixing plane, a situation that may exist in the present study due to the close spacing between the rotor and stator rows. This issue has since been brought to the attention of those developing Kestrel's capabilities.

$$PR_{stage} = PR_{rotor} * PR_{stator} \quad (4.1)$$

### 4.3 Performance Maps

As part of the validation effort, performance maps were produced using simulations of the Rotor 4 stage for several operating points at 100%, 95%, and 90% design speeds. The operating points simulated included near choke, at peak efficiency, and near stall. To predict an approximate stall point, the simulated flow rate was decreased from a previous solution until KCFD could no longer converge to a steady solution. Similarly, the point of choke was found by increasing the flow rate until Kestrel could not converge on a steady solution. Where possible, simulations were run at the same corrected mass flow rate as the experiment, and also at some intermediate flow rates for better curve resolution. The predicted performance maps are compared with the experimental data at 100%, 95%, and 90% design corrected speeds and shown in Figures 4.3, 4.4, and 4.5 respectively. All results are mass-averaged. Error bars for measured efficiency and corrected mass flow rate, calculated from the instrumentation accuracy reported in the PBS study, are also plotted. The PBS study reports the mass flow rate measurement error as plus or minus 0.5%. The total pressure measurement error of the downstream rakes is reported as plus or minus 0.0046 psi, and the plenum total pressure measurement error as plus or minus 0.003 psi. Using these numbers the calculated error bounds for pressure ratio were calculated using Equation 4.2, where  $PR$  is the pressure ratio and  $Pt$  is the total pressure. Calculating the error for pressure ratio in this way resulted in an error of less than 0.06% of the reported results and thus is not plotted, being too small to see on the plots. The total temperature measurement error was reported as plus or minus

0.25° R, and the error bounds for total temperature were calculated using Equation 4.3, where  $TR$  is the temperature ratio, and  $Tt$  is the total temperature. Being thus equipped with upper and lower bounds for total temperatures and pressures, Equation 4.4 was used to calculate the error range for the measured efficiency, where  $\eta_c$  is compressor efficiency,  $PR_c$  is total pressure ratio,  $TR_c$  is the total temperature ratio, and  $\gamma$  is the ratio of specific heats.

$$PR_{bounds} = \frac{PR * Pt_{plenum} \pm 0.0046}{Pt_{plenum} \mp 0.003} \quad (4.2)$$

$$TR_{bounds} = \frac{TR * Tt_{plenum} \pm 0.25}{Tt_{plenum} \mp 0.25} \quad (4.3)$$

$$\eta_c = \frac{PR_c^{\frac{\gamma-1}{\gamma}} - 1}{TR_c - 1} \quad (4.4)$$

For comparison to mixing plane accuracy of other flow solvers, simulation results produced by Dr. Michael List [27] of the same fan using STAR-CCM+ are also shown in Figure 4.3. Although List used a slightly different computational domain, the authors feel that the comparison is warranted because of his use of single-passage grids and mixing planes. It is also noted that List used a Spalart-Allmaras turbulence model, whereas the turbulence model used in KCFD was Menter's baseline two-equation model.

The performance maps at 100% design speed are shown in Figure 4.3. It is worth addressing the capability of Kestrel's mixing plane analysis to predict the bounds of operation for 100% design speed, as this is a major design consideration. Importantly, the mixing plane model chokes at a higher flow rate than experiment, with the KCFD solver predicting choked flow above 61.68 lbm/s, 0.51% greater than the 61.37 lbm/s obtained experimentally. This is consistent with the observations made by Nichols et al in his simulation of the NASA stage35 [32]. Additionally, the mixing plane model fails to converge below 58.5 lbm/s, indicating stalled behavior. This is 2.9% higher than the experimental point of stall for this fan of 56.85 lbm/s.

At 100% design speed, the mixing plane analysis predicts a peak efficiency of 87.26% at a corrected mass flow rate of 61.5 lbm/s, whereas the measured peak efficiency was 88.62% at 60.77 lbm/s. Since the experimental data chokes at a lower flow rate than the simulation, the measured

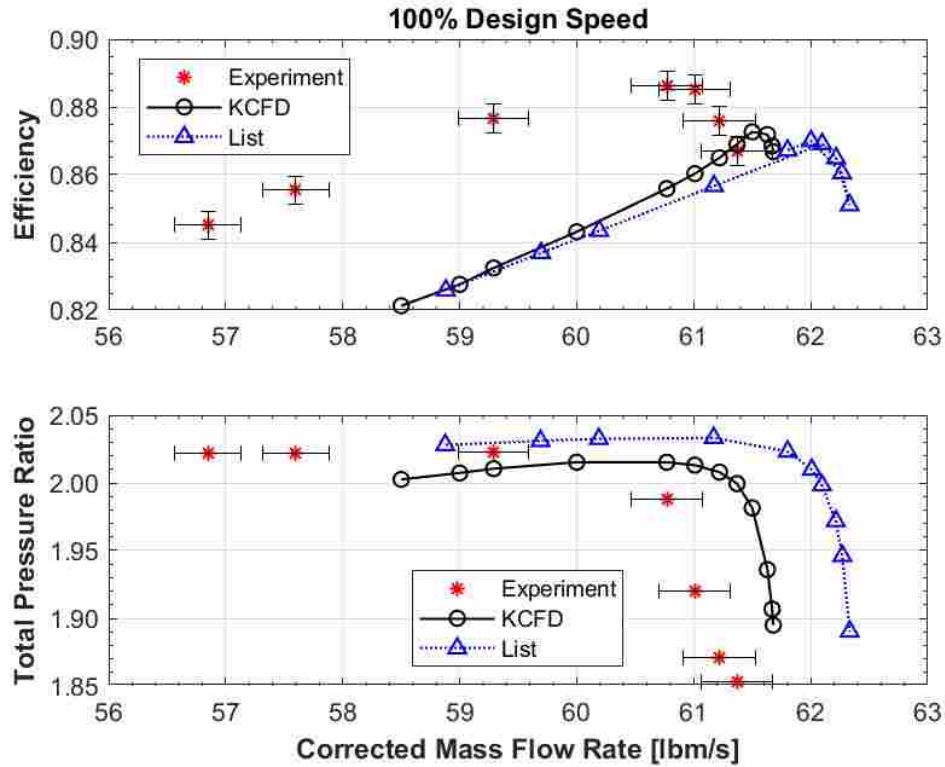


Figure 4.3: Rotor 4 stage performance results at 100% design speed.

efficiencies peak and begin to drop with increasing flow rates while the predicted efficiency line is still rising. Thus, the under prediction is most severe at flow rates below 60.77 lbm/s, where KCFD predicts efficiencies about 4% lower than the measured data. These results are opposite of the trend seen in the mixing plane analysis done by Nichols using the NASA Stage35 compressor geometry, where the efficiency was over predicted [32]. This under prediction is likely due to a higher predicted total temperature ratio for the stage than was seen experimentally, which would contribute to a lower calculated efficiency. Further insight is presented in the discussion of the total temperature profiles at the stage exit.

There are some differences between the predicted total pressure ratios and the experimental values shown in Figure 4.3. The total pressure ratio rolls off at a higher mass flow rate than experiment, leading to KCFD over predicting the total pressure ratio near choke by as much as 7.9%. Away from choked conditions, the predictions are within 0.64% experiment. A peak total pressure ratio of 2.015 is predicted by KCFD at 60.77 lbm/s. By comparison, the peak total pressure ratio measured experimentally was 2.023 at 59.29 lbm/s. The difference between the experimentally



measured peak and the peak predicted by KCFD is less than 0.4%. Lastly, KCFD predicts a noticeable drop in total pressure ratio as the fan approaches stall, whereas the experimental pressure ratio remains relatively even at flow rates close to stall.

The trends present in the KCFD simulations in Figure 4.3 are visually similar to the results produced by List using mixing planes in STAR-CCM+ [27]. Notably, these other simulations also under predicted the efficiency of the stage, predicted a higher corrected mass flow rate at choke than the experimental data showed, and stalled at a higher mass flow rate than was achieved by experiment. This gives good reason to believe that discrepancies between the KCFD performance predictions and experiment can be attributed to inaccuracies inherently found in the CFD models, such as some poorly modeled physical phenomenon, rather than a bad implementation of mixing planes within KCFD. Some important differences between the results generated by List and the results from the current study are that the mass flow rates for choked and stalled conditions predicted by KCFD are closer to those determined experimentally. Furthermore, KCFD predicts slightly lower total pressure ratios along the whole speedline than STAR-CCM+. Near choke, this means that KCFD is closer to the experimental values.

Simulations were also run at 95% design speed, and the performance maps generated using Kestrel are compared to the experimental data in Figure 4.4. The stall and choke points were determined in the same manner as for the 100% design speed performance maps. Similar to what was observed at 100% design speed, the mixing plane analysis predicts choke and stall behavior at higher mass flow rates than obtained experimentally. At 95% design speed, the KCFD solver predicts choked flow above 60.25 lbm/s, 1.1% higher than the highest flow rate obtained experimentally of 59.61 lbm/s. Stall is predicted to occur below 56.13 lbm/s, 4.8% higher than the 53.54 lbm/s that was measured.

At 95% speed KCFD again under predicts the efficiency along the speedline. Except for where the experiment choked, KCFD predicts efficiencies 3-4% lower than what was measured. Because the experimental data chokes sooner, the experimental efficiency peaks and drops off while the predicted efficiency continues to rise. The peak predicted efficiency is 89.50% at a flow rate of 60 lbm/s compared to the measured peak efficiency of 91.27% measured at a flow rate of 58.84 lbm/s.

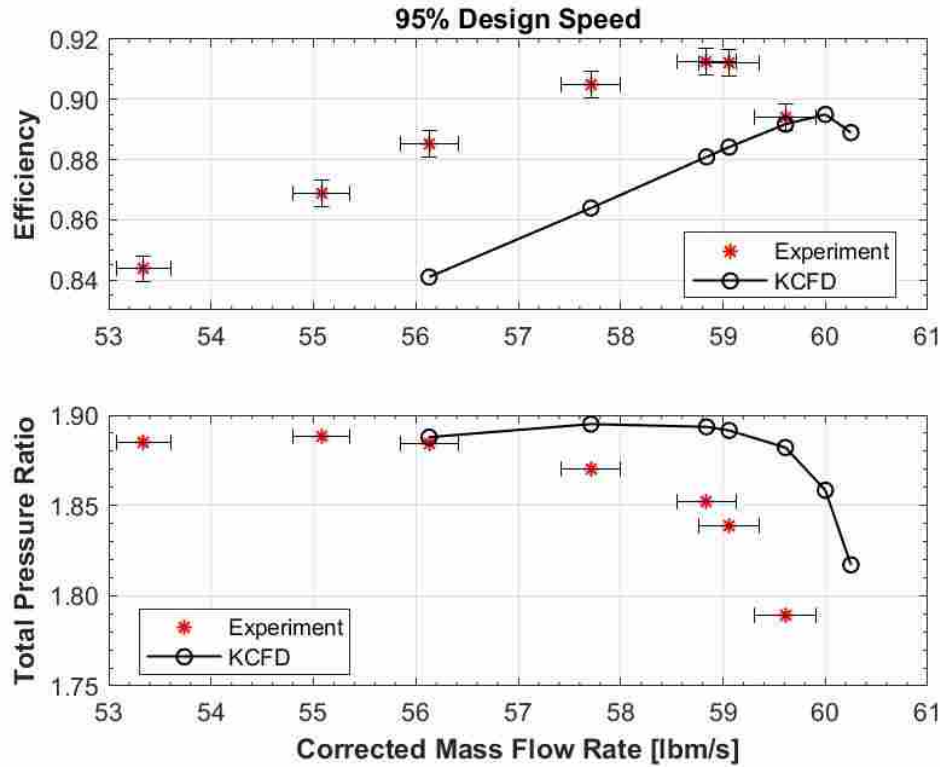


Figure 4.4: Rotor 4 stage performance results at 95% design speed.

Away from choked conditions, the predicted total pressure ratio is within 0.21% of the values obtained experimentally. The predicted peak pressure ratios are also similar, with KCFD predicting a peak pressure ratio of 1.895 at a flow rate of 57.71 lbm/s compared to the experimentally measured peak of 1.888 at 55.08 lbm/s. Thus at 95% design speed KCFD over predicts the peak pressure ratio by less than 0.4%. The experimental speedline for the total pressure ratios rolls off more gradually at 95% than at 100%, which is commonly seen in transonic rotors due to a weaker shock existing at lower speeds. This more gradual decline is not well predicted by KCFD.

Figure 4.5 shows the performance maps for 90% design speed. As with 100% and 95%, the flow rates for choked and stalled flow at 90% are predicted higher than what was measured experimentally. Choked flow is predicted above 58.5 lbm/s, 3.4% higher than the measured 56.59 lbm/s. Stalled flow is predicted below 52.72 lbm/s, while experimental data was measured at flow rates as low as 49.92 lbm/s before stall was detected, constituting a difference of 5.6% between the measured and predicted stall points.

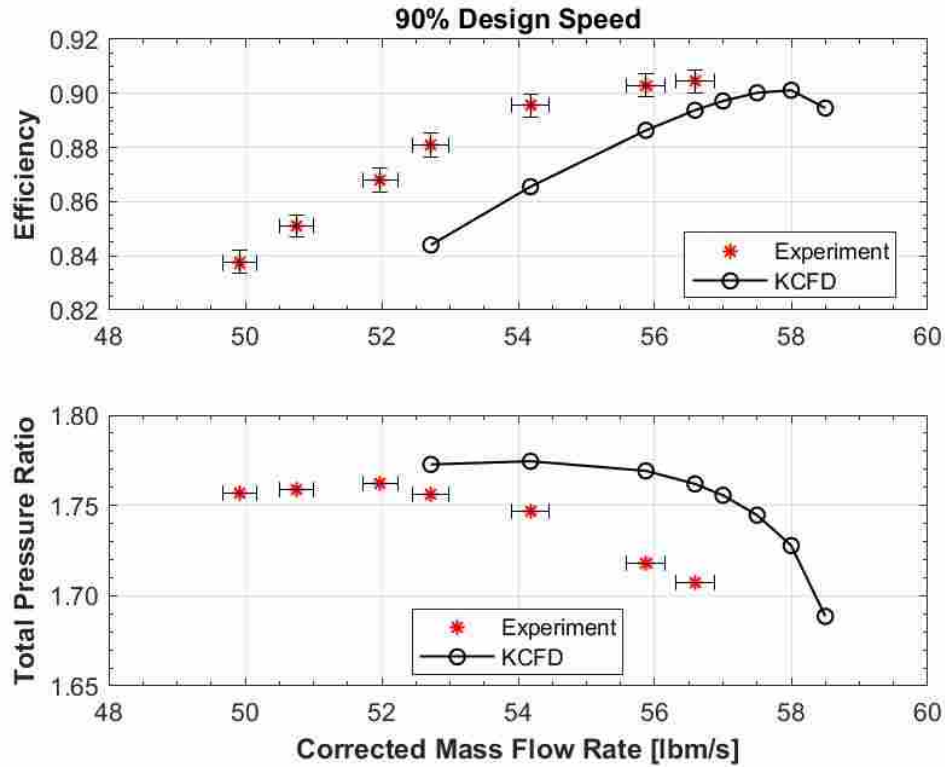


Figure 4.5: Rotor 4 stage performance results at 90% design speed.

The predicted efficiency speedline in Figure 4.5 under predicts the measured efficiencies. Away from where the experiment choked, KCFD predicts efficiencies 3-4% lower than the measured data. Near choke the efficiencies predicted by KCFD approach the efficiencies measured. The peak efficiency predicted by KCFD is 90.12% at 58 lbm/s. Although the experimental efficiency data doesn't exhibit a peak and steep decline near choke, the data does begin to level out as it approaches the point of choke. The highest efficiency measured was 90.44% at 56.59 lbm/s.

There are some important differences between the measured total pressure ratios and the profile predicted by KCFD in Figure 4.5. Unlike at 95% design speed, KCFD does predict a more gradual drop off in performance as the mass flow rate increases. Along the whole speedline, KCFD over predicts the total pressure ratio, but predicts to within 1% of the experimental data at lower flow rates. The peak total pressure ratio predicted by KCFD is 1.774 at a flow rate of 54.18 lbm/s. The measured peak total pressure ratio was 1.762 at a flow rate of 51.98 lbm/s. The difference between these peak pressure ratios is within 0.7%.

There are some consistent trends in the performance maps for 100%, 95%, and 90% design speeds. At all three speeds, Kestrel predicts stalled and choked conditions at higher flow rates than were measured. Furthermore, Kestrel under predicts the efficiencies along the speedline by 3-4% at the three speeds. Lastly, Kestrel closely predicts the peak pressure ratio for all three speeds, and away from choke the predicted pressure ratios show good agreement with the data. Pressure ratio plots for the three speeds are shown in Figure 4.6. Upon comparison to Figure 2.1 it is apparent that both the experimental and predicted speedlines follow the expected trends at the three corrected rotor speeds.

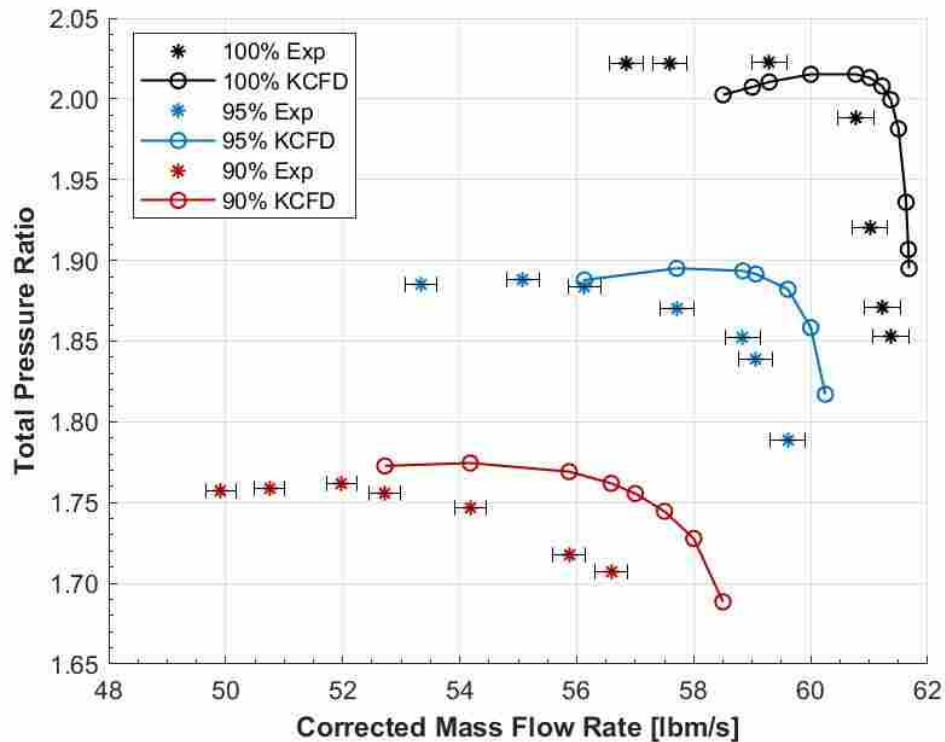


Figure 4.6: Rotor 4 stage performance results at 100%, 95%, and 90% design speeds.

#### 4.4 Radial Profiles

Radial profiles of total pressure and total temperature at the stator leading edge and stage exit were generated at design speed and compared to the experimental data. This was done for conditions closest to choke, at peak efficiency, and near stall for the mixing plane analysis and

the experimental data. Thus, the run that was closest to stall in the experiment is compared to the simulation that was closest to stall, the experiment with the highest efficiency is compared to the simulation with the highest efficiency, and the highest mass flow operating point achieved experimentally is compared to the highest mass flow simulation (near choke). These profiles are shown in Figures 4.7 and 4.8.

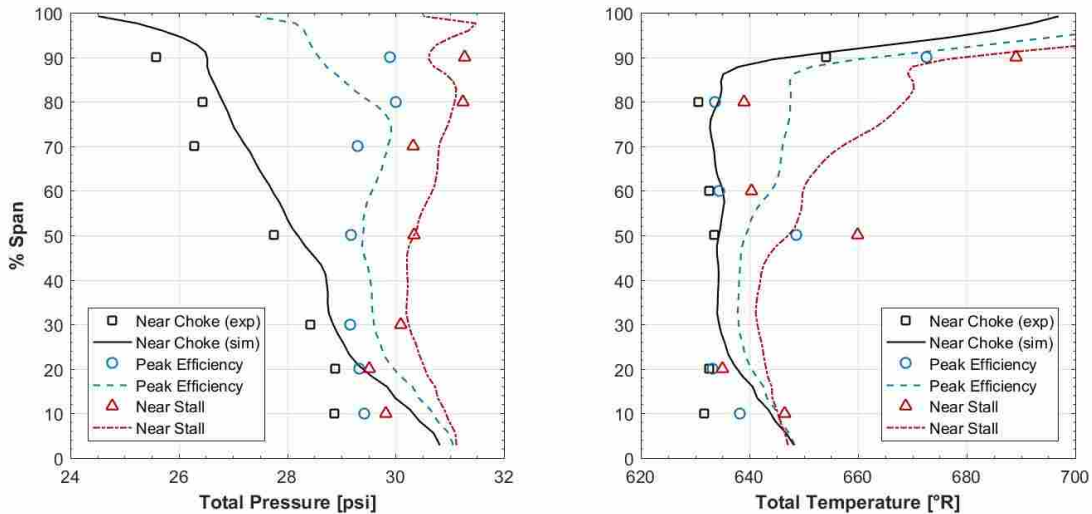


Figure 4.7: Stator leading edge profiles at 100% design speed for Rotor 4.

The total pressure profiles in Figure 4.7 show good agreement to the data, especially away from the hub region. Below 20-30% span, the total pressures are over predicted by 5% or less for all three operating conditions. Above 20% span, the predicted near choke profile follows the same trend as the experimental data, but over predicts the total pressures by about 2%. Similarly, the predicted total pressure profile at peak efficiency is within 3% of the experimental data between 20% and 80% span, but under predicts the pressures by about 4% above 80% span. Lastly, the predicted near stall profile stays within 2% of the experimental data above 30% span.

The total temperature profiles at the stator leading edge also show good agreement to the data. Above 20% span the predicted near choke profile is within 1% of the data. Even at 10% span the predicted near choke profile only over predicts the data by less than 2%. The predicted peak efficiency profile shows a similar trend to what was measured experimentally, but over predicts the temperatures along the span by about 1.5%. Lastly, the predicted near stall profile does not seem to

match very well the measure data, except at 10% and 90% span. The simulation profile increases in magnitude from 650° R to approximately 670° R above 50% span, whereas the experimental data remains near 640° R until 90% span where it increases rapidly. Furthermore, the total temperature data for both the peak efficiency data and the near stall data exhibit a spike in temperature at 50% span that is not present in the KCFD prediction. This may be due flow physics that aren't captured by the mixing plane model. Alternatively, it could be bad data taken in the experiment. Although data was taken radially in 10% increments, some data points were thrown out if it was believed the instrumentation was faulty at those locations. For this reason there are some gaps in the experimental data shown in Figure 4.7, such as at 30%, 40%, and 70% span in the total temperature profiles. This brings up the question of whether or not the data at 50% span may also have been bad.

The total pressure profiles at the stage exit are shown in Figure 4.8. The predicted near choke profile is within 1% of the measured data above 50%, but then over predicts by about 4% below that. The predicted peak efficiency profile under predicts the total pressure by as much as 5% above 40% span, and then over predicts by 1-5% below that. Lastly, the predicted near stall profile shows a trend that matches what was measured over the whole span, but under predicts the data by about 2.5%.

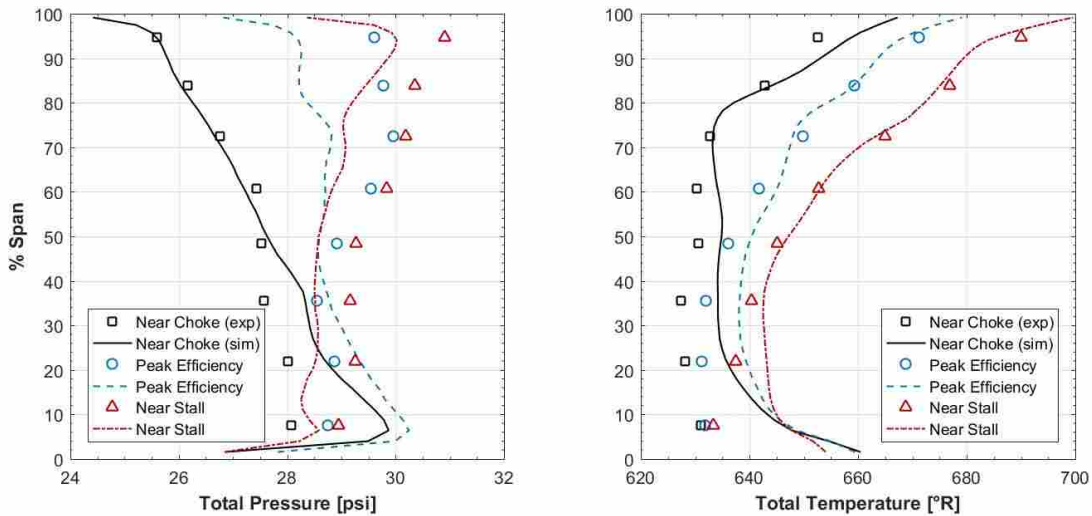


Figure 4.8: Stage exit profiles at 100% design speed for Rotor 4.

For the total temperature profiles at the stage exit, the profile shapes of all three operating conditions are well predicted, and the magnitudes are within 1% of the experimental data above 40% span. Below this point, all the predicted profiles over predict the experimental data by as much as 2%, indicating that higher total temperatures are predicted downstream of the rotor. This may explain the lower efficiency predicted by the mixing plane model, as a higher temperature ratio across the stage will lead to a lower efficiency, as shown in Equation 4.4.

Figure 4.9 shows the radial profiles at the stator leading edge at 95% speed. Here, the total pressure profiles exhibit the same trends as with the 100% design speed analysis. Notably, the predictions are within about 2% of the experimental measurements above 20% span, and below that the predicted total pressure profiles deviate but stay within 5% of what was measured experimentally.

The total temperature predictions seen in Figure 4.9 for near choke and peak efficiency show the best agreement to the data, staying within 2% of the experimental measurements. Similar to the 100% data they slightly over predict the total temperature below 20% span by as much as 2%. The near stall predicted total temperature profile shows an increase in temperature above 70% span that is not present in the experimental data, but is still within 5% of what was measured. As with the 100% design speed runs there is again a spike in temperature at 50% span that is not predicted. Again, this may be due flow physics that aren't captured by the mixing plane model or bad data taken in the experiment.

The radial profiles for 95% speed at the stage exit are shown in Figure 4.10. Looking first at the total pressure profiles, the predicted near choke profile is within 1% of the experimental data above 40% span, but over predicts total pressures by as much as 7% below that. As with 100% design speed, the total pressure at peak efficiency is under predicted by 1-4% above 50% span, and then over predicted by 1-6% below that. The total pressure predictions for near stall remain within 2% along the entire span, and capture the trends very well. Of the three profiles, the near stall profile is closest to the experimental data near the hub.

Similar to the 100% design speed, the predicted total temperatures at stage exit are very close to the experimental data, especially above 30% span. The predictions stay within 1% of measured above 30% span. Below 30% span the temperatures are slightly over predicted, but remain within 2% of the data at all three operating points.

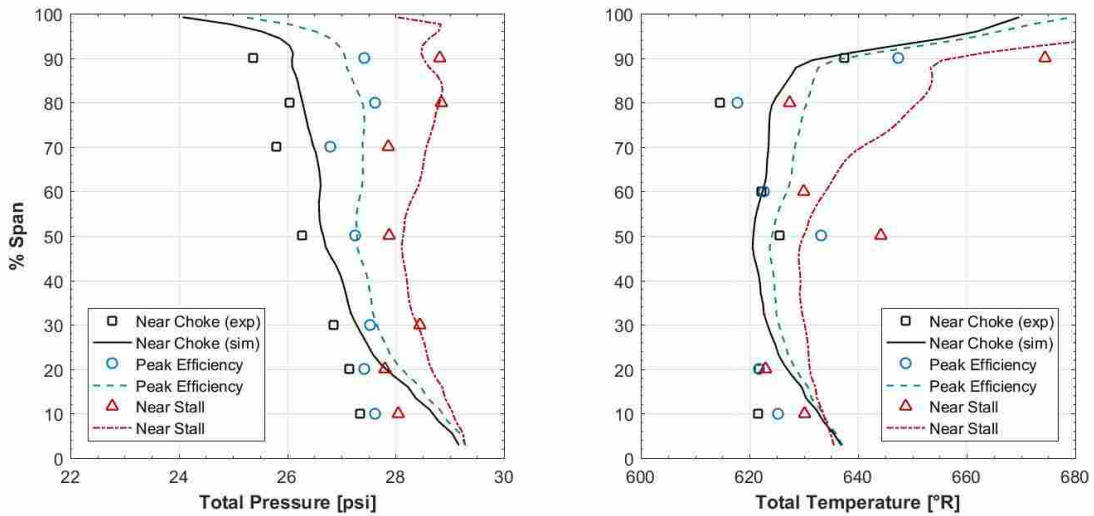


Figure 4.9: Stator leading edge profiles at 95% design speed for Rotor 4.

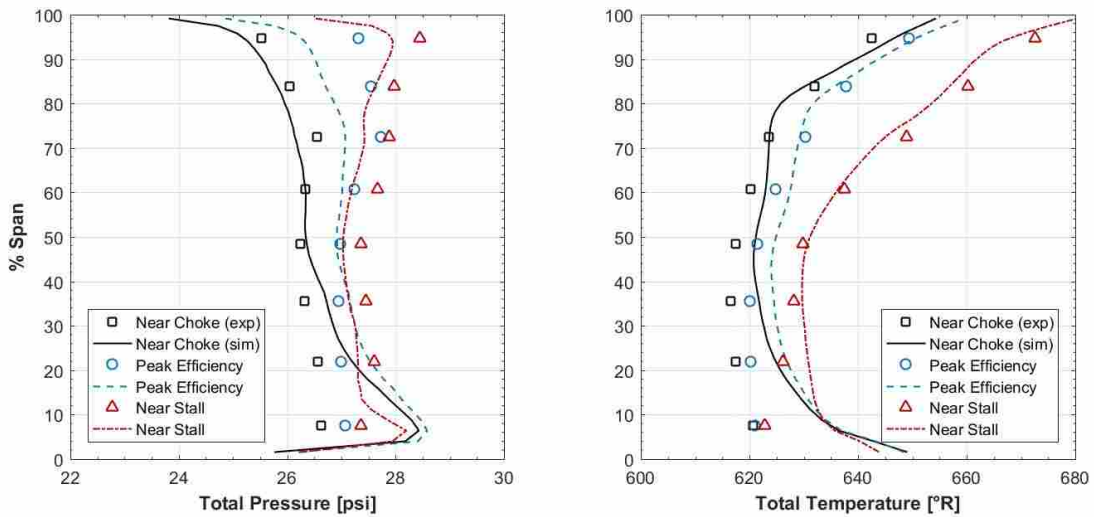


Figure 4.10: Stage exit profiles at 95% design speed for Rotor 4.

Figure 4.11 shows the radial profiles at the stator leading edge at 90% speed. The highest flow rate achieved experimentally for 90% speed also had the highest efficiency, and there was not any data at higher flow rates to use as near choke data for comparison. The total pressure profile for peak efficiency shows good agreement with the experimental data, largely staying within 2% of the experimental data, except at 70% span and 10% span, where the total pressure is over predicted by about 4%. The near stall total pressure profile also stays within 2% of the experimental data



along most of the span, but over predicts by as much as 4% at 70%, 20%, and 10% span. The peak efficiency total temperature profile matches the trends seen experimentally very well, but over predicts by about 3% at 10% span. As seen for 100% and 95%, the near stall predicted profile noticeably over predicts the temperature at 80% span, but is still within 4% of what was measured.

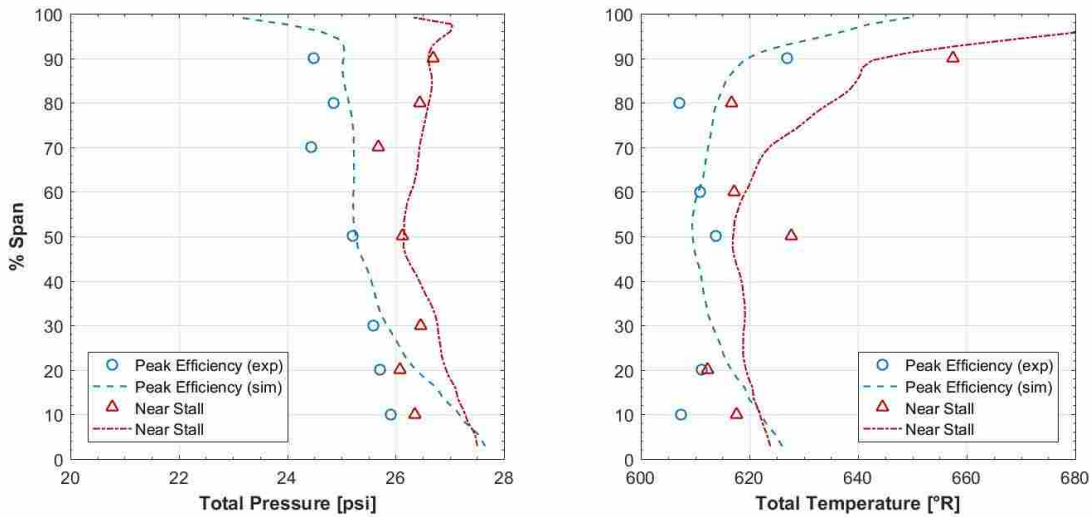


Figure 4.11: Stator leading edge profiles at 90% design speed for Rotor 4.

The stage exit radial profiles at 90% design speed are shown in Figure 4.12. The exit profiles for both total temperature and total pressure match the best at this rotation speed, likely because of the absence of a shock at lower rotational speeds. The total pressure profiles for both peak efficiency and near stall are within 1-2% of the experimental data above 10% span, but near 10% span over predict the pressure by 8% and 4% respectively. The total temperature predictions are also very accurate, predicting to within 1% of the experimental measurements above 30% span. Below 30% span the total temperatures are over predicted, but stay within 3% of what was measured.

The possible sources for the differences between the radial profiles are many. Efforts were made to probe the simulated flow field at approximately the same axial locations that experimental data was taken. However, due to poor documentation regarding the exact axial location of the stage exit rakes, discrepancies may exist that could explain some differences between the data sets. Additionally, some heat transfer is likely to have occurred into the hub and casing sections

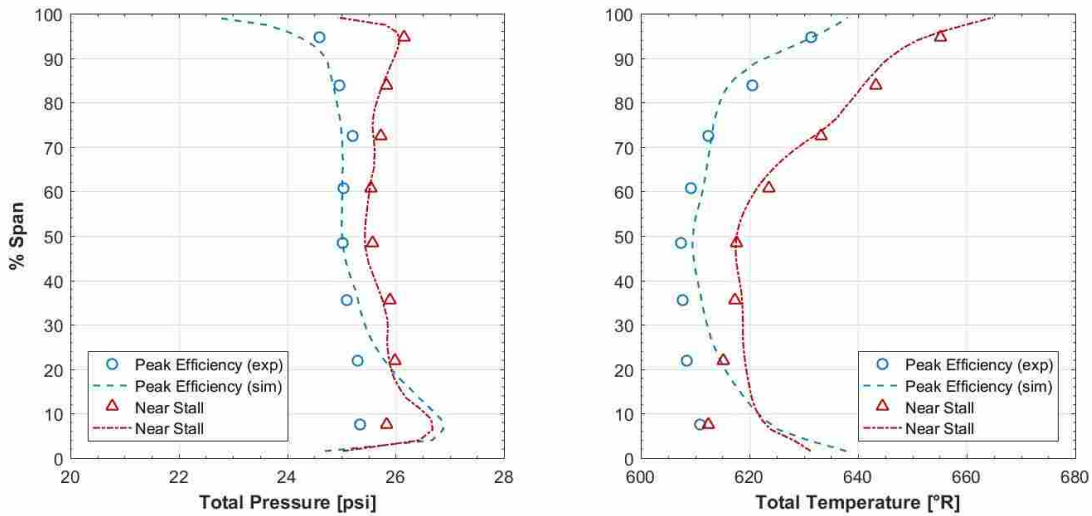


Figure 4.12: Stage exit profiles at 90% design speed for Rotor 4.

during experiment, whereas this mixing plane simulation assumed adiabatic wall boundaries. This may explain the generally higher predicted total temperatures near the hub. This possibility is also pointed out by List in his mixing plane analysis of the same fan [27].

This validation study demonstrated the ability of Kestrel's mixing plane simulations to predict very closely the performance of this fan without necessitating the use of time-accurate runs and full annulus grids. The near choke point was predicted at all three corrected rotation speeds to within 3.4%, and the near stall point was predicted to within 5.6%. Peak efficiency and total pressure ratios were predicted to within 2% and 1% respectively. Furthermore, the radial profile plots correctly captured the trends of the profiles. The changes in magnitude between near choke, peak efficiency, and near stall operating are well captured. Also, the magnitudes of the profiles is generally very good, in every case staying within 4% of the experimental data along most of the span and not varying by 8% at any point.

## **CHAPTER 5. STALL INCEPTION STUDY RESULTS**

This chapter presents the results from the inlet distortion study. The first section focuses on results obtained while simulating stall inception under uniform (clean) inlet conditions. The results presented in the uniform inlet section are used as a point of comparison in the following section, which addresses stall inception with the inlet distortion boundary condition applied.

### **5.1 Stall Inception With Uniform Inlet Flow**

In this section the full annulus performance results are compared to the mixing plane predictions and the experimental data. Then various flow data are compared as the compressor approaches and goes into stall, including casing pressure data, and various iso-surfaces and contour plots.

#### **5.1.1 Performance Maps**

The performance maps obtained with the full annulus runs using a uniform inlet boundary condition are plotted in Figure 5.1. All values are mass-averaged. Error bars for measured efficiency and corrected mass flow rate were calculated using the process described in section 4.3. Due to the focus of this work on stall inception mechanisms, most of the simulations done were close to stall. Runs near choke were not performed with the full annulus grid. For the clean inlet simulations, the point of stall was discovered by iteratively decreasing the exit area of the extended nozzle, which gradually decreased the mass flow rate through the compressor, until the appearance of a part-span rotating stall cell. The flow field for each nozzle configuration was initialized using the restart data from the converged simulation of the previous nozzle configuration. The run that contained the rotating stall cell is connected via a dotted line to the stall point (the lowest mass flow operating point prior to rotating stall occurring). When a fan goes into stall, the blades can no longer do as much work on the fluid. This results in a drop in total pressure ratio as well as

efficiency. Furthermore, the stall cell represents a flow blockage region, which also limits the mass flow rate through the fan. Thus, a compressor is expected to experience a drop in efficiency, total pressure ratio, and mass flow rate as it enters into the stalled regime. The lowest mass flow rate obtained using the full annulus simulations before stall inception occurred was 57.27 lbm/s, which is 0.74% higher than the 56.85 lbm/s obtained experimentally. By contrast, the mixing plane simulations over predicted the stall point by 2.9%. The process for determining the point of stall using mixing planes is described in section 4.3. The improved predictions of the stall point made by the full annulus runs is likely due to the use of the extended nozzle domain, which has been shown to improve stall point prediction in single-passage and full annulus simulations [15, 44]. However, it is apparent from these plots that the full annulus simulations would choke at a higher flow rate than those predicted using mixing planes, which over predicted the flow rate associated with compressor choke by 0.51%.

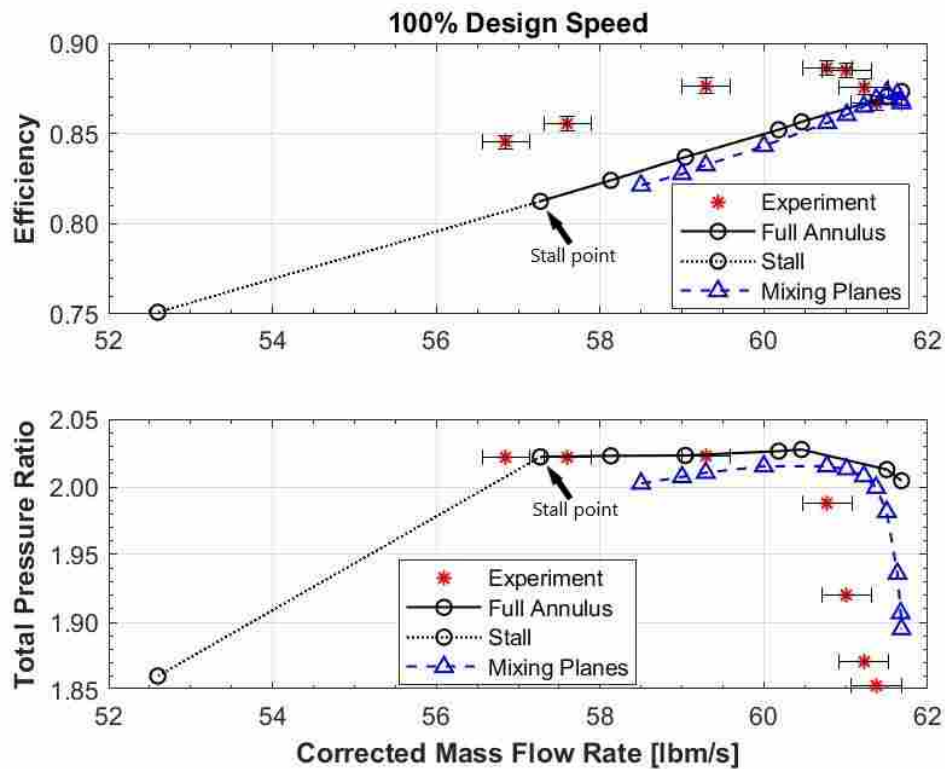


Figure 5.1: Rotor 4 stage performance results at 100% design speed.

Similar to the mixing plane simulations, the full annulus simulations under predict stage efficiency at most flow rates, though they are slightly closer than the mixing plane simulations. The highest efficiency predicted by the full annulus simulations was 87.33% at a flow rate of 61.68 lbm/s, slightly higher than the peak efficiency of 87.26% predicted by the mixing plane analysis, but still lower than the experimentally measured peak efficiency of 88.62%, which occurred at 60.77 lbm/s. No full annulus simulations were run at flow rates higher than 61.68 lbm/s, making it difficult to know if 87.33% is the peak efficiency that would have been predicted using the full annulus grid. Since the experimental data chokes at a lower flow rate than the full annulus simulations predict, the measured efficiencies peak at a flow rate of 60.77 lbm/s and begin to drop with increasing flow rates while the efficiency line simulated by KCFD is still increasing. Thus, the full annulus efficiency predictions differ most from experiment at flow rates below 60.77 lbm/s, where the simulations predict efficiencies about 3-4% lower than the measured data.

The total pressure ratios predicted by the full annulus simulations are very close to those obtained experimentally, especially away from choked conditions. Since the experimental data reaches choke at lower flow rates than predicted by the simulations, the measured and predicted total pressure ratios differ by as much as 8% near choke. A peak total pressure ratio of 2.028 was predicted by the full annulus simulations at 60.64 lbm/s, whereas the measured peak total pressure ratio was 2.023 at 59.29 lbm/s. These predictions are noticeably better than those made by the mixing plane simulations. Below flow rates of 60.77 lbm/s, the full annulus simulations predict the pressure ratios within 0.3% of the measured data. The improved predictions made using the full annulus simulations are unsurprising given the higher fidelity associated with time-accurate simulations.

### **5.1.2 Stall Cell Formation and Characteristics**

More important than performance prediction, the main purpose of this study was to utilize CFD to examine the flow physics relating to stall cell formation in Rotor 4, and understand how the presence of inlet distortion affects the stall inception process. This meant examining the flow as the fan approached stall, and getting time-accurate data during stall inception. Several aspects of the flow were tracked through the stall inception process to understand the flow mechanisms involved. Since it is believed that stall inception occurs in the tip region of the rotor, this was the

region examined most particularly. Data analyzed includes casing tap data, contour plots of various flow variables at radial locations near the rotor tip, and isosurfaces of velocity and  $\lambda_2$  values.

In our simulations, the fan was numerically instrumented using tap files that specified tap locations at 6 equally spaced locations around the annulus at 99.9% span, essentially at the casing. The first researchers to observe spike-type stall formation instrumented their compressors at an axial location at or just ahead of the leading edge plane of the rotor, and this has since been the modus operandi for researchers examining stall inception [3, 6, 9, 10]. This is because leading edge separation is one of the first events to occur in stall inception. This tap data was taken at 50% axial chord upstream of the rotor leading edge. Five revolutions of static pressure data from the simulation that stalled is shown in Figure 5.2. Although the moment of stall inception is not shown, some useful observations regarding the behavior of the stall cell can be made. A sudden drop in pressure representing the stall cell is clearly discernible as it passes the tap locations. Furthermore, the tap data shows a building up of pressure in the moments before the low pressure region passes through, which is consistent with previous studies [10]. This pressure build up is due to the blockage caused by the stall cell. The tap data also shows that the fully formed stall cell rotates at approximately 60% rotor speed.

A well known criterion for spike stall inception in low-speed compressors is leading edge spillage typically associated with tip clearance flow that is perpendicular to the axial direction [4, 7, 8]. This study sought to establish whether the same can be said of Rotor 4, a high-speed compressor. Because tip clearance flow leaks upstream from the pressure surface to the suction surface through the tip gap, it is easily visualized by identifying regions of negative axial velocity in cut planes inside of the tip gap. Side-on views of instantaneous axial velocity contour plots at 99.9% span are shown in Figure 5.3 for flow rates between 60.46 and 57.27 lbm/s. Flow through the stage is moving from top to bottom, and blade movement is from right to left (see Figure 5.3a). To view the tip clearance flow, the contours are truncated to show only velocity regions with a negative axial component (positive axial velocities are white). The axial positions of the blade leading and trailing edges are marked by the dotted lines. The detached shock is shown to cause negative axial flow at the casing, indicated by regions of negative axial velocity just ahead of the blade leading edge. This is consistent with findings made by Hah, et al., which showed that regions of low momentum fluid form just downstream of where the shock impinges on the casing, and that

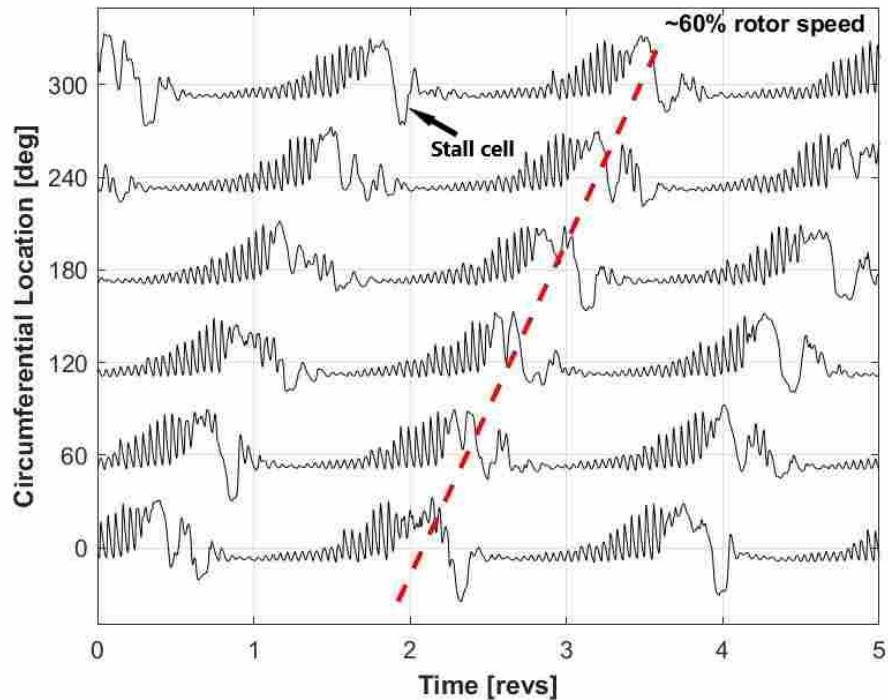


Figure 5.2: Pressure traces at various circumferential locations demonstrating the movement of the rotating stall cell.

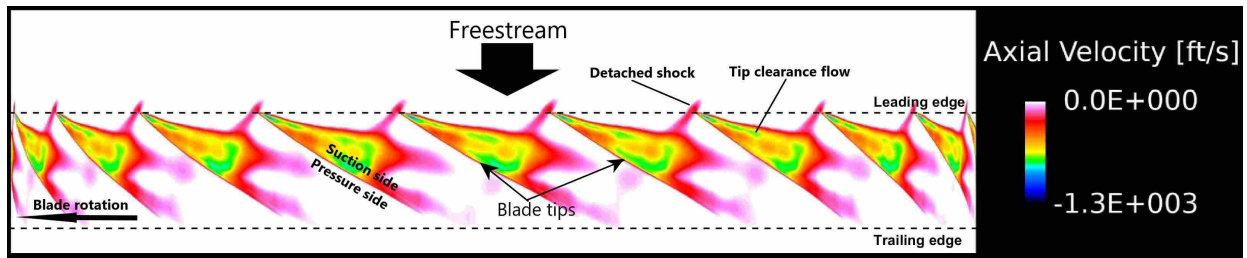
these low momentum regions can lead to flow separation at the casing [12, 13]. Figure 5.3 takes those findings further and demonstrates that these low momentum regions cause negative axial flow in the tip region. As expected, the shock is seen to move further away from the blade leading edge as the flow rate decreases, evidenced by a larger region of recirculating flow behind the shock at lower flow rates. This is most apparent when comparing Figures 5.3a and 5.3e. As flow rates decrease, the unsteadiness associated with the shock and tip vortex interaction becomes visually more apparent. This is especially true in Figure 5.3e, where the leading edge of the reversed tip gap flow, along with the shock position, visually change position from the blade passage labeled number 1 to the blade passage labeled number 2, whereas in Figure 5.3a the flow features in each blade passage are visually more similar. Although the tip clearance flow was not observed to be perpendicular to the axial direction at any of these flow rates, it is apparent that the region of reversed flow caused by the normal shock provides an avenue for leading edge spillage, much like the process described by Hah, et al [12, 13], and summarized in section 2.4. At progressively lower flow rates, the leading edge spillage becomes more severe. The second criterion for spike stall

inception in low-speed compressors is reversed flow in the tip region at the trailing edge plane [8]. Figure 5.3 does not show a noticeable increase of reversed flow at the blade trailing edge at lower flow rates.

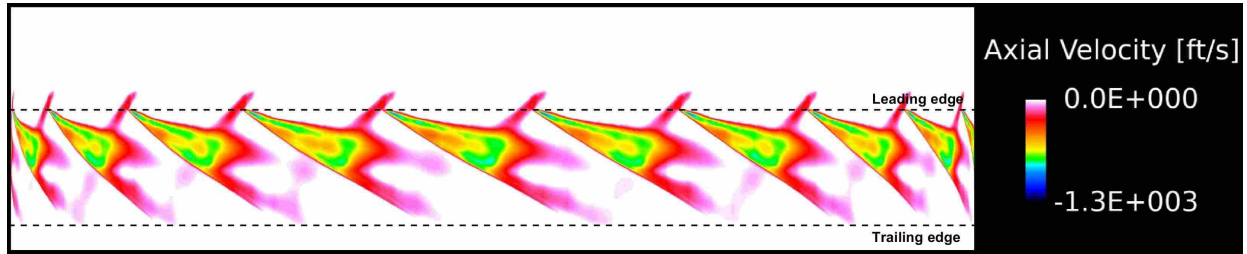
Stall occurred at corrected flow rates lower than 57.27 lbm/s. Figure 5.4 shows side-on views of instantaneous axial velocity contour plots at 99.9% span at several time steps after the nozzle area was decreased past the stall point, allowing for the observation of the flow in this region during stall inception. In Figure 5.4a the flow structures have not changed much compared to Figure 5.3e. In Figure 5.4b the reversed flow inside the blade passages become visually more severe and almost completely fills the regions between blade tips in some of the blade passages. Furthermore, the regions of negative axial velocity caused by tip clearance flow and the detached shock start to join together into one region as the negative axial flow region behind the shock grows in size due to the shock moving further upstream from the blade leading edge. Finally, the negative axial flow region also extends downstream, toward the trailing edge plane marked by the dotted line. By the time step shown in Figure 5.4c the tip clearance flow and the negative axial velocity region caused by the shock have largely combined into one region of negative axial flow in all of the blade passages, and many blade passages are completely blocked by negative axial flow. In Figure 5.4d the upstream edges of the negative axial flow regions have broken down. This is likely due to the breakdown of the detached shocks upstream of the leading edges of the blades caused by flow blockages in each blade passage. Furthermore, the region of negative axial velocity has extended downstream past the trailing edge plane, and the two criteria for stall inception are satisfied: there is leading edge spillage of tip clearance flow caused by the detached normal shock, and there is negative axial flow extending downstream past the trailing edge plane [8]. The blockages present in each blade passage will combine into a single stall cell as the flow stabilizes to its new operating point.

Figure 5.5 shows a side-on view of an instantaneous axial velocity contour plot at 99.9% span after the formation of a stable stall cell. The stall cell is seen on the left of the figure. Since the stall cell rotates more slowly than the rotor, the stall cell in this figure would be moving from left to right relative to the rotor blades (see Section 2.1 and Figure 2.2). This knowledge allows us to observe what happens in the tip regions of the blade passages to the right of the stall cell just before they become stalled, allowing for an analysis similar to what was done in Figure 5.4. Figure

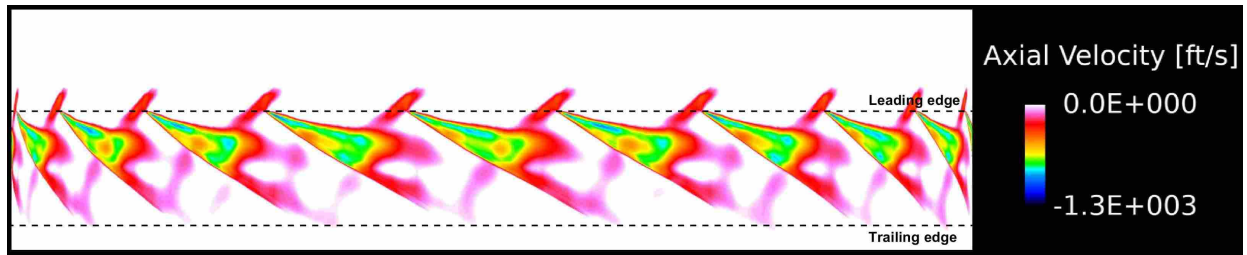




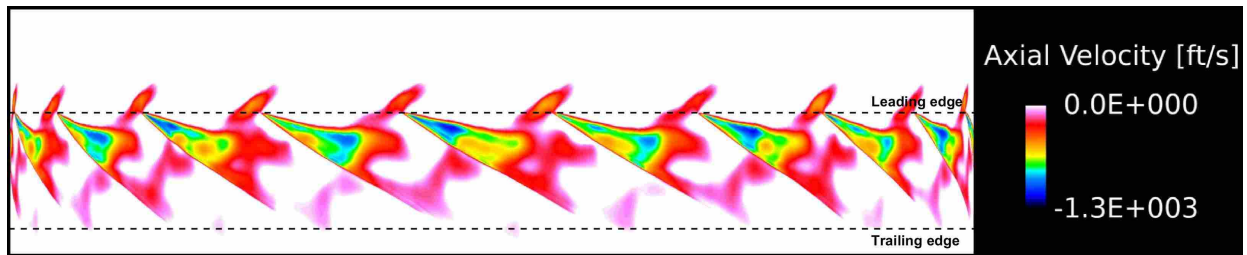
(a) 60.46 lbm/s



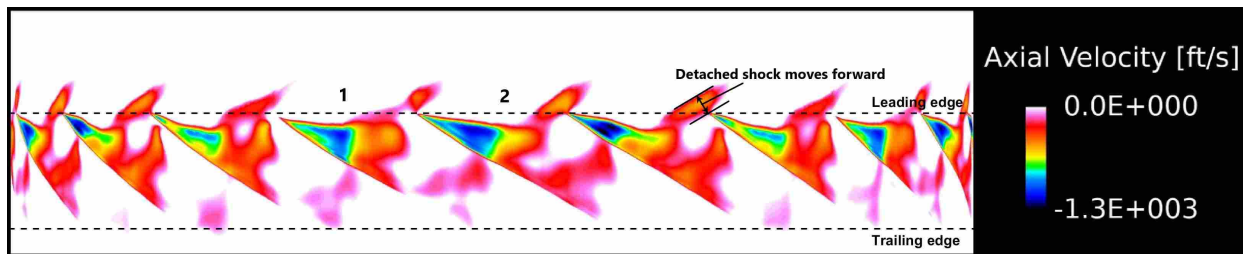
(b) 60.18 lbm/s



(c) 59.04 lbm/s



(d) 58.13 lbm/s



(e) 57.27 lbm/s

Figure 5.3: Negative axial velocity contours at 99.9% span.

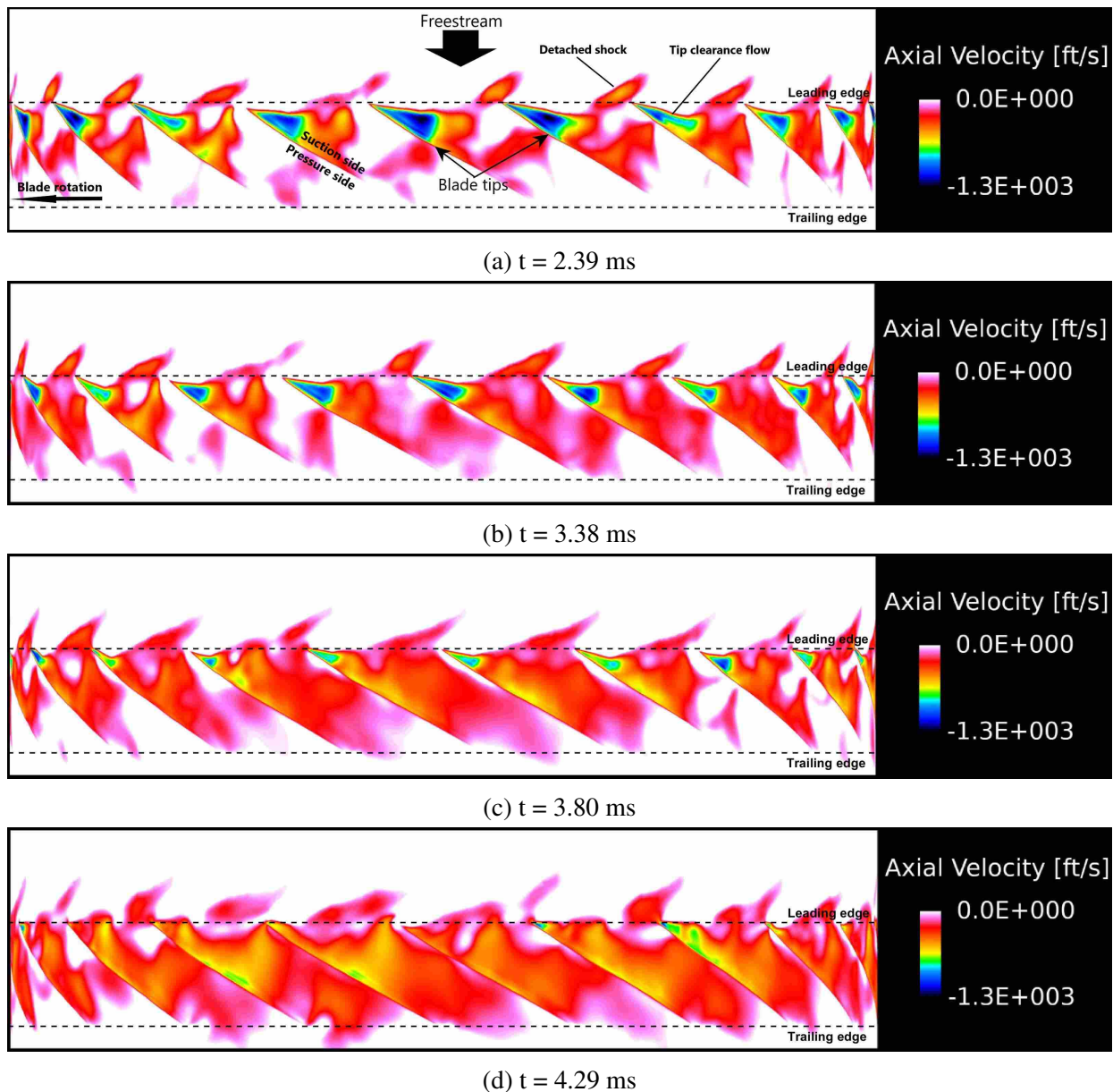


Figure 5.4: Negative axial velocity contours at 99.9% span during stall inception.

5.5 does not show the leading edge of the reversed tip clearance flow becoming perpendicular to the axial direction as the blades move closer to the stall cell, as would be expected in a low-speed compressor. However, reversed flow due to detached shocks near the leading edges of the blades closest to the stall cell is noticeable, supporting the idea that they provide an avenue for leading edge spillage to occur. Furthermore, near the stall cell, reversed flow occurs at or near the trailing edge plane, indicating a blocked blade passage.

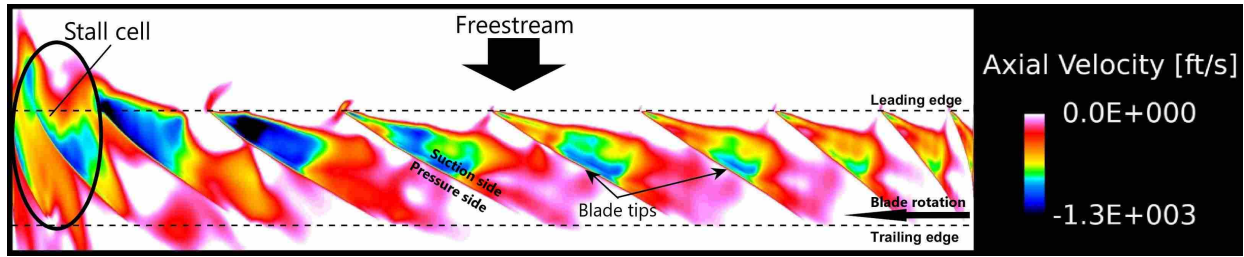


Figure 5.5: Negative axial velocity contours during stalled conditions.

Figure 5.6 is an axial view of the stall cell at the rotor midchord and shows the radial and circumferential extents of the rotating stall cell. The stall cell appears to envelop the outer 10-20% of the span of the rotor and be contained within four blade passages, although there are some other areas of reversed flow outside of where the main stall cell is located. The total cross-sectional area of the stall cell at this axial location, measured using the area of reversed flow associated with the stall cell, is approximately  $7.7 \text{ in}^2$ . Whether a stall cell will become a part- or a full-span stall cell can be predicted by radial profiles of the total pressure after the rotor [45]. Part-span stall cells are predicted by higher total pressures at the rotor tips than at the rotor mid-span, whereas a full-span stall cell is predicted by higher total pressures near the rotor mid-span. Figure 4.7 shows predicted and measured radial profiles of total temperature and total pressure at the stator leading edge. Near stall the highest total pressures are measured near the tip, evidenced by higher total pressure ratios on the outer 20% of the span. This was measured experimentally and predicted by the mixing plane simulation, and the development of a part-span stall cell is consistent with predictions that would be made from the radial profile data.

The  $\lambda_2$ -criterion is a measure of local vorticity similar to the Q-criterion and is useful for visualizing areas of high vorticity, including shocks and vortices. Its derivation and definition were first put forth by Jeong and Hussain [46]. Figure 5.7 shows the  $\lambda_2$  isosurface during stalled conditions. Away from the stall cell, tip clearance vortices are clearly defined, as are the detached shocks near the blade leading edges. These flow structures are present at every flow rate simulated with uniform inlet conditions.

Near the stall cell these structures break down, as shown in Figure 5.8. The dark blue surface is an isosurface of 0 axial velocity, and aids in the visualization of the stall cell. In the bottom right corner region where the stall cell is visible, the  $\lambda_2$  isosurfaces are truncated and don't

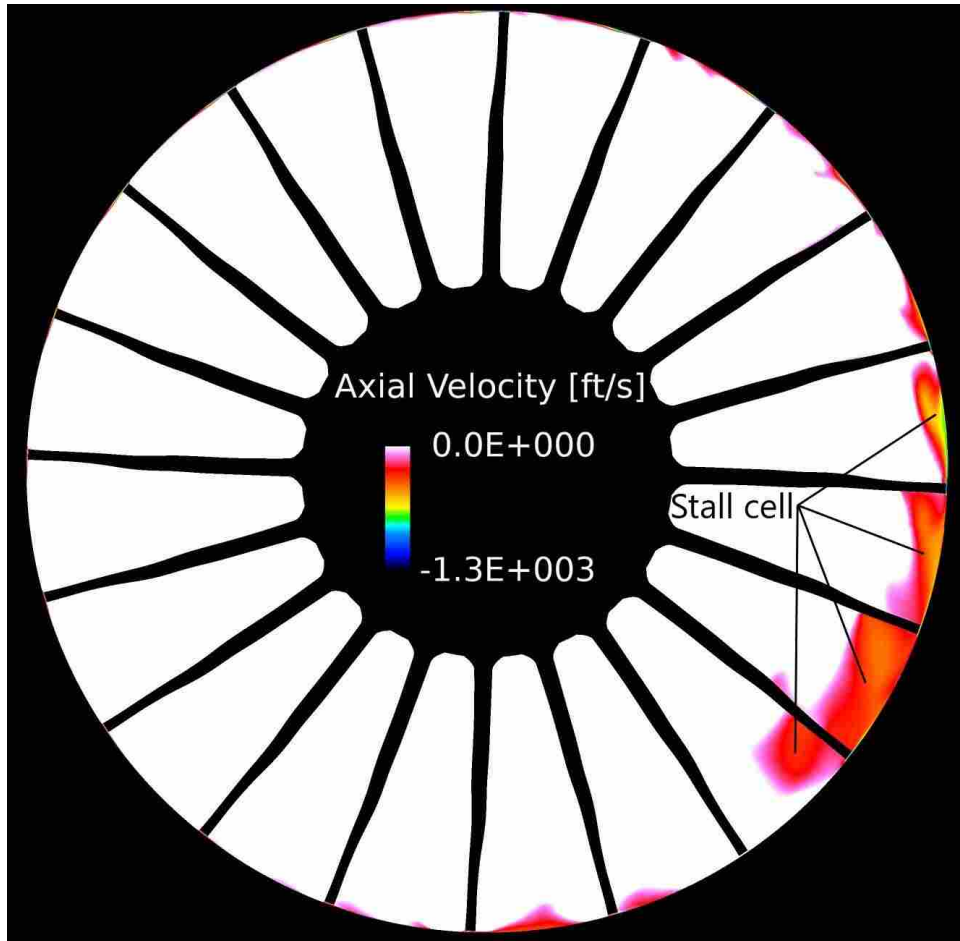


Figure 5.6: An axial view showing the extent of the stall cell.

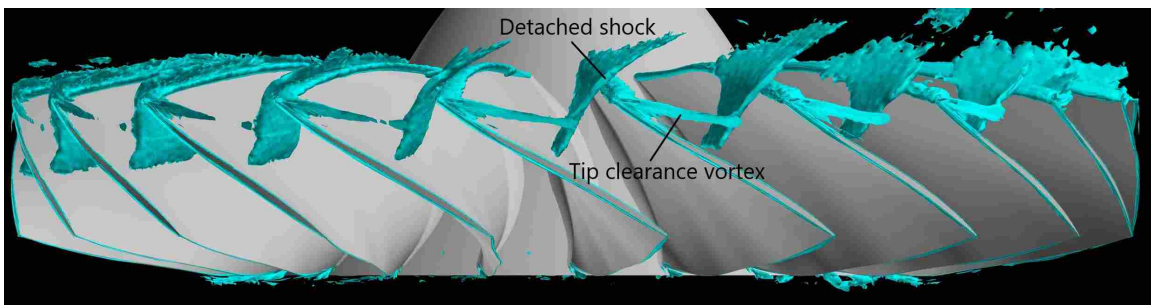


Figure 5.7: Isosurface of the  $\lambda_2$  variable identifying tip gap vortices and detached shocks.

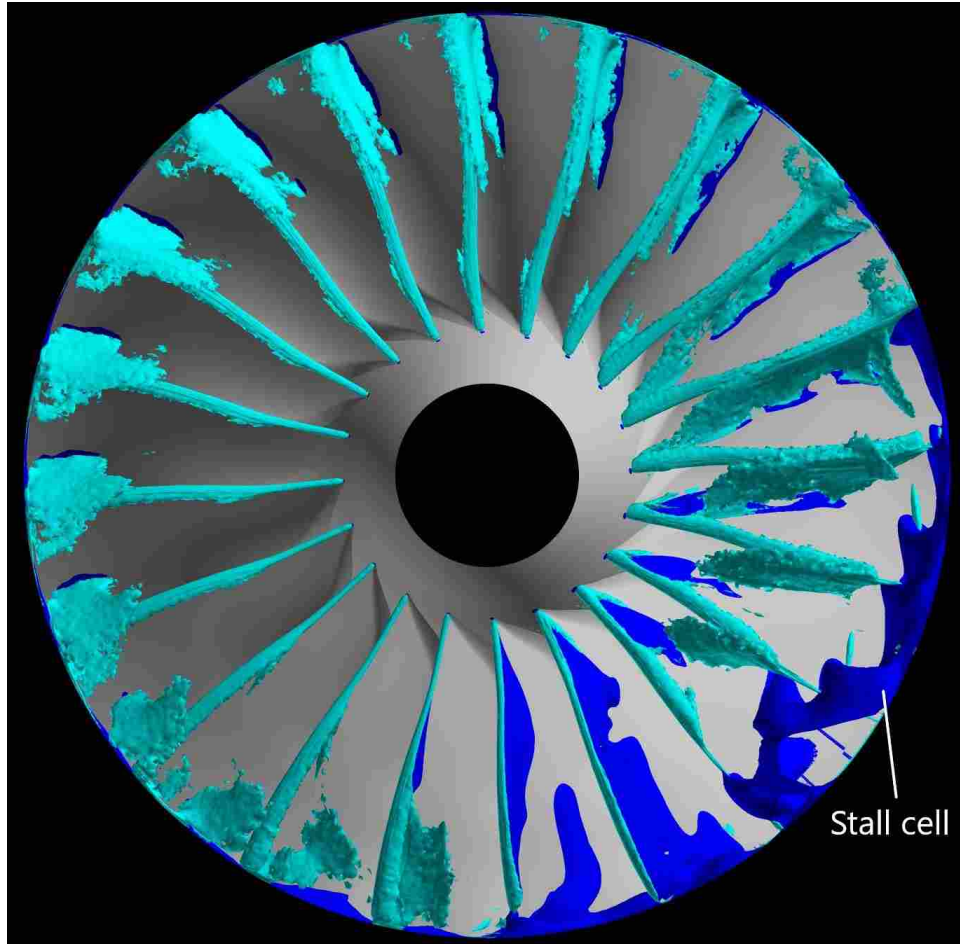


Figure 5.8: An axial view showing the interaction of the  $\lambda_2$  isosurface with the stall cell.

extend into the stall cell. Thus, inside of the stall cell, normal shocks and tip clearance vortices are not formed. This is to be expected, as the stall cell represents a flow blockage, and flow inside of a stall cell is highly chaotic.

Low-speed compressor research has established that stall inception is caused by flow separation due to the high incidence angle at the blade leading edge. If severe enough, the resulting separation region can form a radial vortex that attaches to the blade and the casing, which develops into a rotating stall cell. Simulations of low-speed compressors have utilized the  $\lambda_2$  isosurface to demonstrate the presence of these radial vortices at stall inception [9, 10]. Similar radial vortices were also seen in many of the time steps in the present study after the formation of a stable stall cell, shown on the right side of Figure 5.9. Also shown in the figure is a radial contour plot showing negative axial velocities, which locates the radial vortex in relation to the stall cell. Figure 5.9

demonstrates that these radial vortices can exist in high-speed compressors during stall, and form as the the blade moves out of the stall cell. However, they were not observed to occur before or during stall inception, and thus are not thought to be part of the stall inception process in Rotor 4. In other words, these radial vortices only formed after the commpressor had gone into stall. To the author’s knowledge, this is the first time that radial vortices have been confirmed to form in high-speed compressors using CFD.

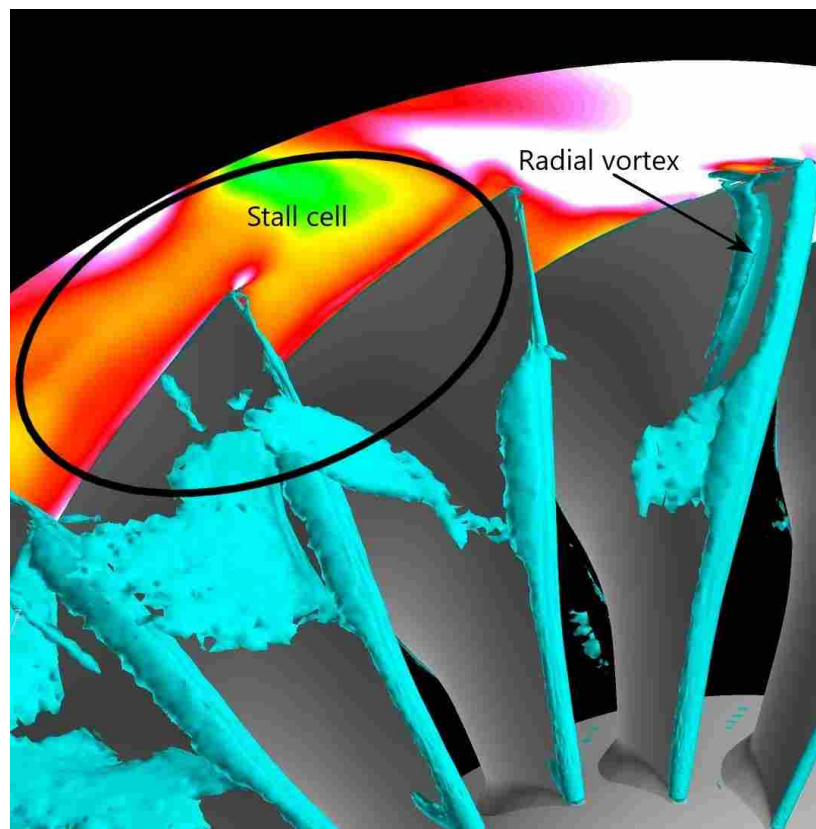


Figure 5.9: An axial view showing the interaction of the  $\lambda_2$  isosurface with the stall cell.

## 5.2 Stall Inception With Distorted Inlet Flow

In this section the results from the distortion inlet simulations are presented and contrasted with the clean inlet simulations. Performance maps of the distortion inlet simulations are shown. Various flow data is examined as the compressor experiences stall, including casing pressure tap data, and various iso-surfaces and contour plots.

### **5.2.1 Performance Maps**

The distorted inlet simulations were carried out by applying the inlet distortion boundary shown in Figure 3.11 to three converged clean inlet simulations. As a reminder, the variable nozzle exit area determined the operating point for the clean inlet simulations, and so by restarting from three different clean operating points, three nozzle geometries were being utilized for the distortion runs. All three of the simulations went into stall, meaning there were no simulations with an unstalled compressor with inlet distortion. This is unfortunate because it limits the comparisons that can be made between the clean and the distorted inlet simulations to just those simulations that were in the stalled regime, and a goal of this research was to compare the flow in clean inlet and distortion inlet simulations as the fan approached stall. Due to time constraints there was not time to attempt further distortion inlet simulations using more nozzle configurations to achieve simulations that were not in stall. Figure 5.10 shows plots the operating points using the distortion boundary condition together with the clean inlet operating points. Dotted lines connect the converged clean inlet configurations with the performance results after applying the distortion boundary condition. Stalled performance with distorted inlet boundary conditions is noticeably worse than stalled performance with a clean inlet. Figure 5.11 shows the performance maps for just the simulations using a stalled inlet boundary condition. Understandably, the efficiency of rotor 4 is also drastically decreased by the presence of inlet distortion. However, without data from inlet distortion simulations that aren't stalled, it is difficult to make any firm conclusions as to how much efficiency is lost due to inlet distortion, and how much is attributable to stalled conditions alone.

### **5.2.2 Stall Cell Formation and Characteristics**

The stalled behavior of the compressor in the event of inlet distortion is noticeably different from the clean inlet simulation. Five revolutions of tap data gathered during a stalled run with inlet distortion is shown in Figure 5.12. These taps were in the same locations as the taps used for the clean inlet simulations (99.9% span and 50% axial chord length upstream of the rotor leading edge). The plot shows that there are more stall cells present than in the clean inlet simulations. Furthermore, the transition from low to high pressure is more abrupt than for the clean inlet simu-

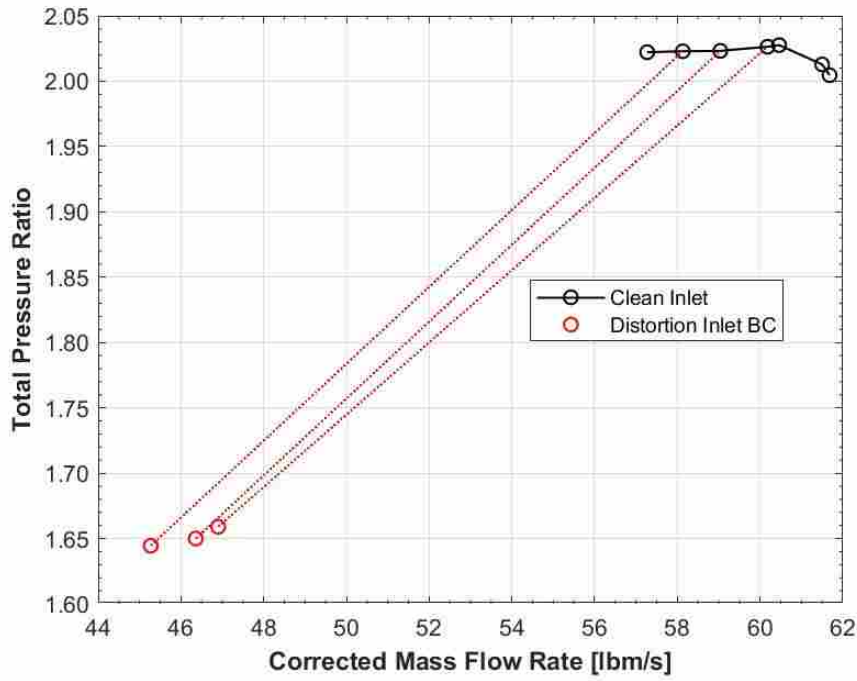


Figure 5.10: Pressure ratios for clean and distorted inlet conditions.

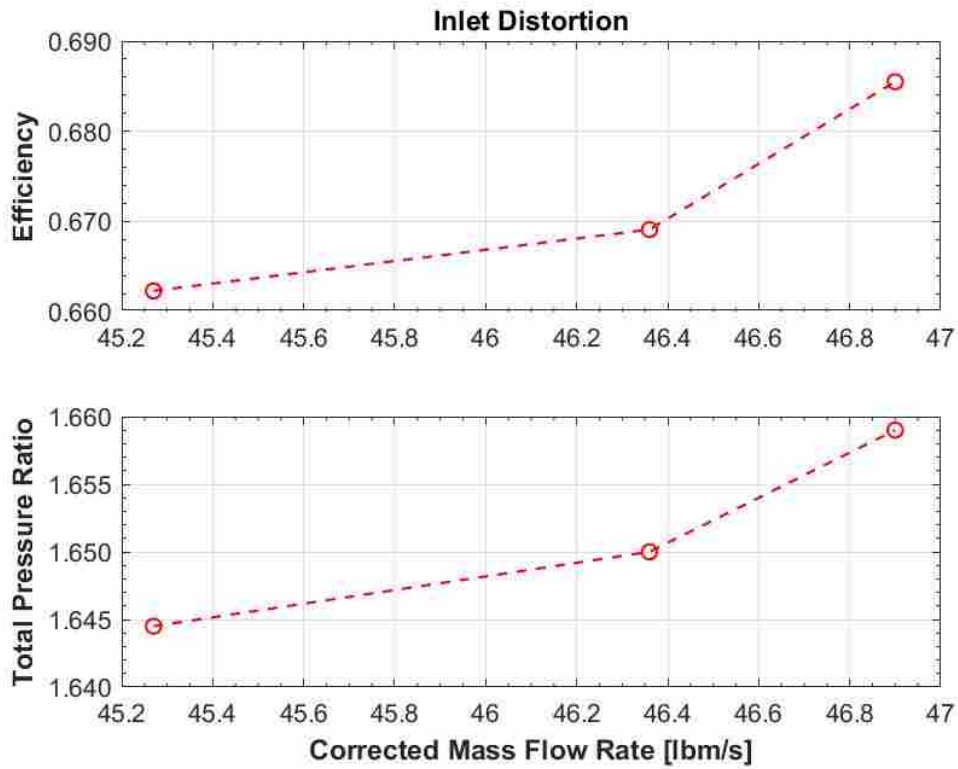


Figure 5.11: Rotor 4 stage performance results at 100% design speed with distorted inlet BC.



lation, likely because of the closer spacing of the stall cells. The stall cells rotate at approximately 70% rotor speed, somewhat higher than the 60% rotor speed observed in the clean inlet simulation. From the pressure tap data it appears that the presence of inlet distortion affects stalled behavior in at least two ways: it effects the number of stall cells as well as their rotation speed. In the absence of more simulations, it is difficult to make the conclusion that inlet distortion directly caused four stall cells to form. Typically, as a fan goes deeper into stall the number of stable stall cells that form varies, often increasing from one stall cell to multiple stall cells as the operating point goes deeper into the stalled regime. It may be that the simulations with inlet distortion are deeper into the stalled regime than the stalled clean inlet simulation, resulting in the formation of more stable stall cells.

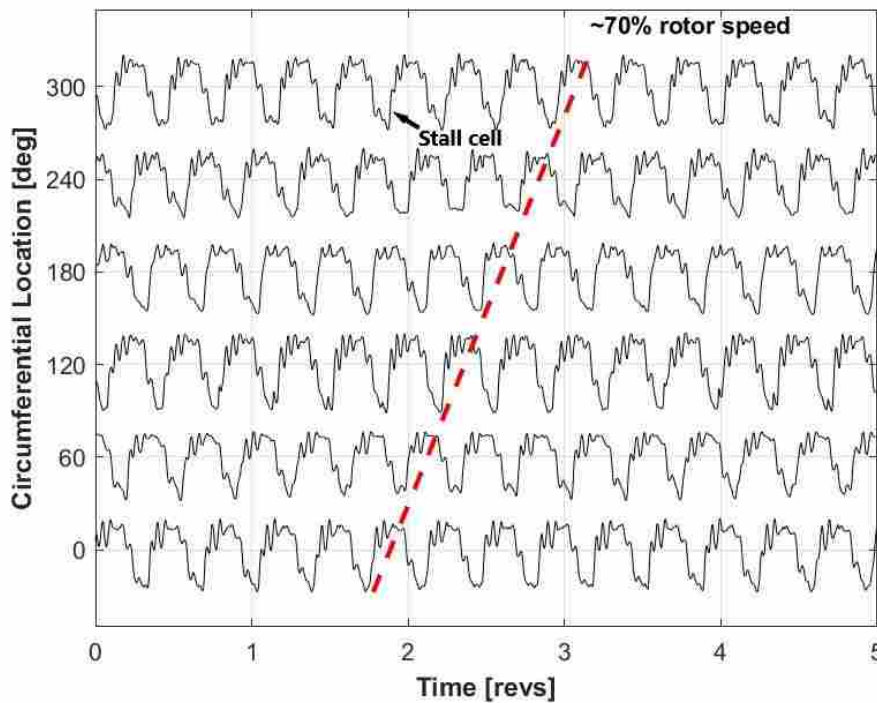


Figure 5.12: Pressure traces at various circumferential locations demonstrating the movement of the rotating stall cells.

Figure 5.13 is an axial view of the axial velocity at the rotor midchord and shows that there are four distinct rotating stall cells that are spaced approximately equidistant from each other. These stall cells are contained within about three blade passages and each have a cross-sectional

area of approximately  $4.0 \text{ in}^2$ . By comparison, the single rotating stall cell formed under clean inlet conditions spanned four blade passages and had a cross-sectional area of  $7.7 \text{ in}^2$ , making each of these stall cells 48% smaller than the single stall cell that forms under uniform inlet conditions. It has been observed that smaller stall cells will rotate at a speed closer to the rotational speed of the rotor [45]. Furthermore, spike-type stall precursors rotate quickly initially and then settle to a slower rotational rate after the rotating stall cell fully forms [3]. The simulation results from this study agree with these previous observations, and the smaller stall cells that develop with distorted inlet flow have a higher rotational speed than the stall cell that develops under clean inlet conditions. Despite the individual stall cells being smaller, the total blockage created by the stall cells in the presence of distorted flow is greater because there are four stall cells ( $16 \text{ in}^2$  vs  $7.7 \text{ in}^2$ ).

$\lambda_2$  isosurfaces were also examined for the distortion simulations. Figure 5.14 shows a side view of the compressor with the  $\lambda_2$  isosurface visible. As with the clean inlet simulations, the presence of the shocks at the leading edge of the blade is discernable. Unlike for the clean inlet simulations, the  $\lambda_2$  isosurface does not show the trajectory or existence of tip clearance vortices. The reason for this is most easily seen in Figure 5.15, which simultaneously shows the  $\lambda_2$  and 0 axial velocity isosurfaces. As with the clean inlet flow, the presence of the stall cell near the tips of the rotor blades disrupts the vortical flow structures shown by the  $\lambda_2$  isosurface, and so the isosurface is truncated near the blade tips around the full annulus. Thus, a large portion of the tip gap region around the annulus are inside of stall cells, and so the tip gap vortices cannot form. Unlike the clean inlet simulations, no radial vortices were identified during or after stall inception at any of the time steps sampled for the distorted inlet boundary condition. This may similarly be due to presence of multiple stall cells. Recall that when there is only one stall cell, the radial vortices were seen on some of the blades as they exited the stall cell region. Due to the close spacing of the stall cells in distorted inlet simulations, the conditions do not exist for radial vortices to form on the blades as they exit the stall cell.

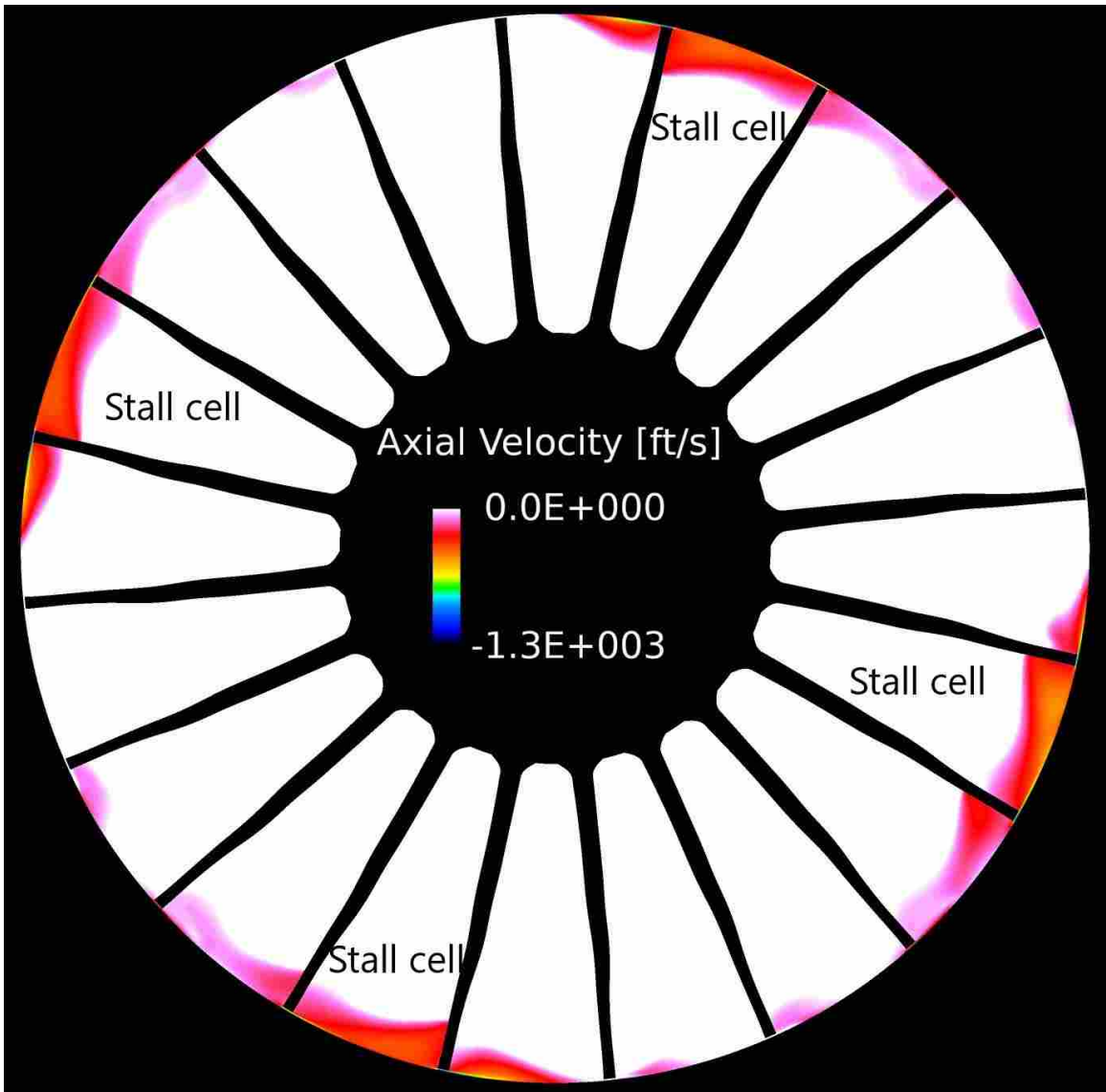


Figure 5.13: An axial view showing the extent of the stall cells.

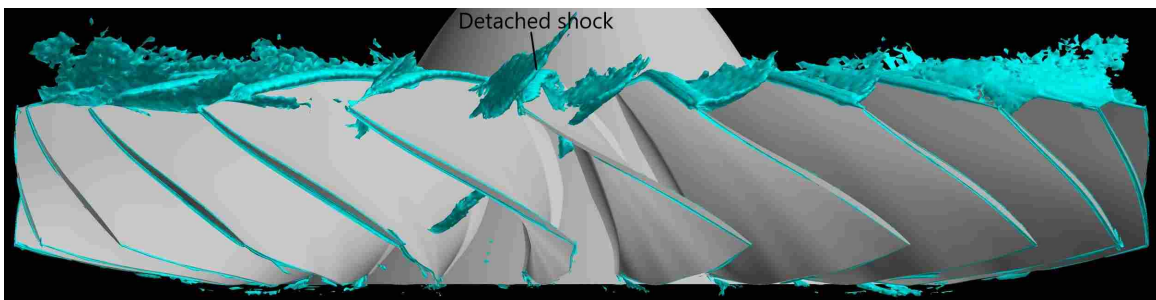


Figure 5.14: Isosurface of the  $\lambda_2$  variable with inlet distortion applied.

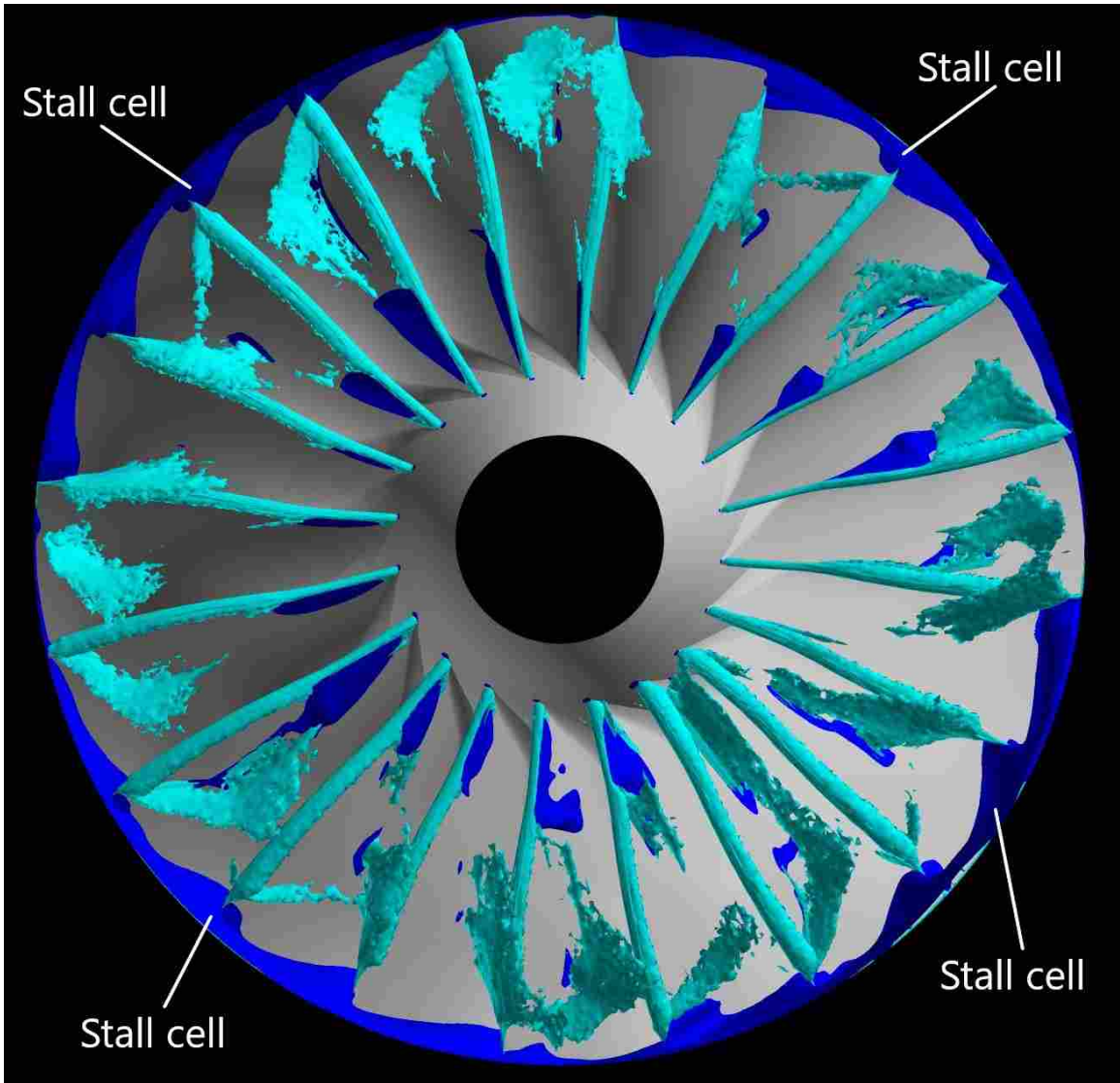


Figure 5.15: An axial view showing the interaction of the  $\lambda_2$  isosurface with the four stall cells.

## **CHAPTER 6. CONCLUSIONS AND FUTURE WORK**

This chapter summarizes the conclusions made from the two parts of this research and outlines recommendations for avenues of further investigation. Because there are two parts to the research, there are two sections of conclusions made.

### **6.1 Mixing Plane Validation**

The mixing plane validation study aimed to provide a validation for Kestrel's mixing plane model using a high-performance research fan. To that end, performance maps were generated using Kestrel for the fan at 100%, 95%, and 90% design speed over a variety of flow rates. These performance maps were compared to experimental data. The performance maps at 100% design speed were also compared to other CFD mixing plane data from the same fan. Additionally, radial profiles of total pressures and total temperatures at the stator leading edge and downstream of the stator were generated from the simulation data and compared to experimental data at the three rotational speeds. Differences between the simulation data and experimental data were discussed.

From the comparison between data and simulation in the performance maps, it was apparent that Kestrel's mixing plane simulations are quite capable of predicting the pressure rise over the stage, though accuracy diminished close to choked conditions as observed in previous validation studies. It was also observed that at all three rotation speeds, the mixing plane model predicted stall at higher flow rates than were observed in experimentation. Also, the mixing plane model under predicted stage efficiency, and this under prediction has been observed in other mixing plane simulations of the same fan using other CFD packages. Because past mixing plane simulations of high-performance fans in Kestrel have not shown a tendency to under predict stage efficiency, future work should be done to identify if this phenomenon is seen in other fans, or is due to characteristics specific to Rotor 4.

The total pressure and temperature profiles suggested that the mixing plane implementation, while unable to capture unsteady effects, can be used to predict trends and magnitudes in time-averaged radial data with a high degree of accuracy. At the stator leading edge, the profiles showed best agreement above 20% span. For the total temperature profiles at the stator leading edge, the predicted profile for the near stall operating point was shown to be further away from experimental data than the other operating points, but was still within 5% of the data. This is true for all three rotational speeds.

At the stage exit, the operating points that showed best agreement between simulation and data for the total pressure profiles were near choke and near stall. Total temperature profiles for all three operating points showed good agreement with the data, but below 40% span the mixing plane simulations began to over predict the total temperatures. These trends were observed at all three rotational speeds. This over prediction of total temperatures was identified as contributing to the under prediction of stage efficiency seen in the performance maps.

Because of their speed and low demand for computational resources, mixing plane simulations will continue to be a vital tool for engine developers for years to come. Kestrel, an important tool used by the DoD for aircraft acquisition, allows for high fidelity simulations to be run that can predict the complex interactions between an aircraft, inlet, and propulsion system in a way that has never been achieved before. As a cutting edge analysis tool, it is necessary that Kestrel's mixing plane capabilities also be thoroughly developed and demonstrated. This research provided valuable data about the mixing plane capabilities of Kestrel as part of that effort.

## **6.2 Inlet Distortion Study**

The inlet distortion study successfully utilized the Kestrel flow solver, KCFD, to predict the stall characteristics of Rotor 4 with and without an inlet distortion boundary condition applied. Stage performance under clean conditions was accurately modeled at the design speed over a range of corrected mass flow rates, and the predicted pressure ratios near stalled conditions were within 0.3% of measured data. Furthermore, the lowest predicted mass flow rate prior to stall was within 1% of the lowest mass flow rate achieved experimentally. Given the focus of modern research on predicting boundaries of operation, and the challenges associated with modeling stall in high-speed compressors, this fidelity to experimental data is impressive.

The clean inlet simulations also provided insight into the mechanisms for stall inception. Unsteady interaction between tip clearance flow and the detached normal shock inside of the blade passages became more prevalent at flow rates close to stall, and this unsteadiness likely provided a way for leading edge spillage to occur and initiate the stall process. Although reversed flow at the trailing edge plane was not observed to be more prevalent in simulations near stall than in simulations away from stall, the phenomenon was shown to occur in blade passages that were approaching a stall cell.

Kestrel was also shown to accurately depict flow structures present in the fan before, during, and after stall inception.  $\lambda_2$  isosurfaces were used to visualize the detached normal shock associated with high-speed fans, as well as tip clearance vortices associated with tip gap flow. Furthermore, radial vortices were also identified, although they were not associated with stall inception as has been done in low-speed compressors. Isosurfaces of zero axial velocity, as well as contour plots, proved useful in visualizing the extent of the rotating stall cells. The stall cells were shown to be part-span stall cells in all cases. The part-span radial extent of the stall cells predicted by the time-accurate simulations is congruent with previous work establishing that higher blade loading at the rotor tips is more likely to result in part-span stall cells.

Differences in stall cell size and behavior were identified between clean and distorted inlet simulations. Most notable was the number of stall cells. All distorted inlet simulations, after converging to stable behavior, exhibited the presence of four part-span stall cells, spaced approximately evenly around the annulus. By contrast, the clean inlet simulations exhibited only one rotating stall cell. It was unclear whether the formation of multiple stall cells is attributable to the inlet distortion or to the compressor being deeper into stall. Another major difference in stall cell behavior was the rotation speed of the stall cells. In clean inlet flow, the single stall cell rotated at approximately 60% of the rotor speed, whereas the stall cells present in the distortion simulations rotated at roughly 70% of the rotor speed. A difference in individual stall cell size is put forth as a plausible explanation for these differences in rotational speed. Furthermore,  $\lambda_2$  isosurface visualizations in the simulations with inlet distortion showed that the close spacing of the multiple stall cells prevented tip gap vortices and radial vortices from forming, the reason for this being that the presence of a stall cell disrupts the flow that leads to the formation of these flow structures.

All simulations utilizing a distortion inlet boundary condition quickly went into stall. This may signify a high sensitivity of Rotor 4 to inlet distortion. Alternatively, it may also signify issues with the solver. These could be inherent in the solver, or a result of user inputs. For example, it may be that utilizing a smaller time step or extending the exit duct prior to the converging nozzle would allow for distortion simulations to be run without going into stall. Future work should be done to obtain data for simulations with inlet distortion that are not stalled. Additional runs of this nature would allow for observation of the effects of inlet distortion on the interaction of the shock and tip clearance flow near stall, as well as properly establish the effect of inlet distortion on stall margin for Rotor 4.



## REFERENCES

- [1] Nichols, R. H., McDaniel, D. R., Tuckey, T., Koomullil, R., Ito, Y., and Klepper, J., 2015. “Firebolt c2.0—Unstructured Grid Navier-Stokes Code for Airframe/Propulsion Integration.” In *53rd AIAA Aerospace Sciences Meeting*, no. 2015-0043. 1, 19, 20, 26, 32
- [2] Day, I., 2016. “Stall, Surge, and 75 Years of Research.” *Journal of Turbomachinery*, **138**(011001-1). 4, 5
- [3] Camp, T., and Day, I., 1998. “A Study of Spike and Modal Stall Phenomena in a Low-Speed Axial Compressor.” *Journal of Turbomachinery*, **120**(3), pp. 393–401. 5, 6, 7, 8, 54, 66
- [4] Chen, J.-P., Hathaway, M. D., and Herrick, G. P., 2008. “Prestall Behavior of a Transonic Axial Compressor Stage via Time-Accurate Numerical Simulation.” *Journal of Turbomachinery*, **130**(4). 5, 11, 12, 32, 54
- [5] McDougall, N. M., Cumpsty, N., and Hynes, T., 1990. “Stall Inception in Axial Compressors.” *Journal of Turbomachinery*, **112**(1), pp. 116–123. 5
- [6] Day, I., 1991. “Stall Inception in Axial Flow Compressors.” *Journal of Turbomachinery*, **115**, pp. 1–9. 5, 6, 54
- [7] Hoying, D. A., Tan, C. S., Vo, H. D., and Greitzer, E. M., 1998. “Role of Blade Passage Flow Structures in Axial Compressor Rotating Stall Inception.” In *ASME 1998 International Gas Turbine and Aeroengine Congress and Exhibition*, no. 98-GT-588, American Society of Mechanical Engineers. 8, 9, 54
- [8] Vo, H. D., Tan, C. S., and Greitzer, E. M., 2008. “Criteria for Spike Initiated Rotating Stall.” *Journal of Turbomachinery*, **130**(1). 9, 10, 54, 56
- [9] Lin, F., Zhang, J., Chen, J., and Nie, C., 2008. “Flow Structure of Short-Length-Scale Disturbance in an Axial-Flow Compressor.” *Journal of Propulsion and Power*, **24**(6), pp. 1301–1308. 9, 15, 54, 61
- [10] Pullan, G., Young, A., Day, I., Greitzer, E., and Spakovszky, Z., 2015. “Origins and Structure of Spike-Type Rotating Stall.” *Journal of Turbomachinery*, **137**(5). 9, 10, 32, 54, 61
- [11] Inoue, M., Kuromaru, M., Tanino, T., Yoshida, S., and Furukawa, M., 2001. “Comparative Studies on Short and Long Length-Scale Stall Cell Propagating in an Axial Compressor Rotor.” *Journal of Turbomachinery*, **123**(1), pp. 24–30. 9
- [12] Hah, C., Rabe, D. C., and Wadia, A. R., 2004. “Role of Tip-Leakage Vortices and Passage Shock in Stall Inception in a Swept Transonic Compressor Rotor.” In *ASME Turbo Expo 2004: Power for Land, Sea, and Air*, no. GT2004-53867, American Society of Mechanical Engineers. 11, 55

- [13] Hah, C., Bergner, J., and Schiffer, H.-P., 2006. “Short Length-Scale Rotating Stall Inception in a Transonic Axial Compressor: Criteria and Mechanisms.” In *ASME Turbo Expo 2006: Power for Land, Sea, and Air*, no. GT2006-90045, American Society of Mechanical Engineers. 11, 12, 55
- [14] Gannon, A., and Hobson, G., 2009. “Pre-stall Instability Distribution Over a Transonic Compressor Rotor.” *Journal of Fluids Engineering*, **131**(5). 12
- [15] Vahdati, M., Sayma, A., Freeman, C., and Imregun, M., 2005. “On The Use of Atmospheric Boundary Conditions for Axial-Flow Compressor Stall Simulations.” *Journal of Turbomachinery*, **127**(2), pp. 349–351. 12, 28, 33, 52
- [16] Davis, R. L., and Yao, J., 2006. “Prediction of Compressor Stage Performance From Choke Through Stall.” *Journal of Propulsion and Power*, **22**(3), pp. 550–557. 12, 33
- [17] Cousins, W. T., 2004. “History, Philosophy, Physics, and Future Directions of Aircraft Propulsion System/Inlet Integration.” In *ASME Turbo Expo 2004: Power for Land, Sea, and Air*, no. GT2004-54210, American Society of Mechanical Engineers. 14, 16
- [18] Spakovszky, Z., Weigl, H., Paduano, J., Van Schalkwyk, C., Suder, K., and Bright, M., 1999. “Rotating Stall Control in a High-Speed Stage With Inlet Distortion: Part I—Radial Distortion.” *Journal of Turbomachinery*, **121**(3), pp. 510–516. 14, 15
- [19] Shaw, M., Hield, P., and Tucker, P., 2014. “The Effect of Inlet Guide Vanes on Inlet Flow Distortion Transfer and Transonic Fan Stability.” *Journal of Turbomachinery*, **136**(2). 14, 15
- [20] Davis, M. W., and Cousins, W. T., 2011. “Evaluating Complex Inlet Distortion with a Parallel Compressor Model: Part 2—Applications to Complex Patterns.” In *ASME 2011 Turbo Expo: Turbine Technical Conference and Exposition*, no. GT2011-45068, American Society of Mechanical Engineers. 14, 15
- [21] Höss, B., Leinhos, D., and Fottner, L., 1998. “Stall Inception in the Compressor System of a Turbofan Engine.” In *ASME 1998 International Gas Turbine and Aeroengine Congress and Exhibition*, no. 98-GT-475, American Society of Mechanical Engineers. 14
- [22] Day, I., Breuer, T., Escuret, J., Cherrett, M., and Wilson, A., 1999. “Stall Inception and the Prospects for Active Control in Four High-Speed Compressors.” *Journal of turbomachinery*, **121**(1), pp. 18–27. 15
- [23] Pearson, H., and McKenzie, A., 1959. “Wakes in axial compressors.” *Journal of the Royal Aeronautical Society*, **63**(583), pp. 415–416. 15
- [24] Cousins, W. T., and Davis, M. W., 2011. “Evaluating Complex Inlet Distortion With a Parallel Compressor Model: Part 1—Concepts, Theory, Extensions, and Limitations.” In *ASME 2011 Turbo Expo: Turbine Technical Conference and Exposition*, no. GT2011-45067, American Society of Mechanical Engineers. 15
- [25] Chima, R. V., 2006. “A Three-Dimensional Unsteady CFD Model of Compressor Stability.” In *ASME Turbo Expo 2006: Power for Land, Sea, and Air*, no. GT2006-90040, American Society of Mechanical Engineers. 15

- [26] Fidalgo, V. J., Hall, C., and Colin, Y., 2012. “A Study of Fan-Distortion Interaction Within the NASA Rotor 67 Transonic Stage.” *Journal of Turbomachinery*, **134**(5). 15
- [27] List, M. G., 2014. “Numerical Quantification of Interaction Effects in a Closely-Coupled Diffuser-Fan System.” PhD thesis, University of Cincinnati. 15, 21, 22, 32, 38, 40, 49
- [28] Yao, J., Gorrell, S., and Wadia, A., 2007. “A Time-Accurate CFD Analysis of Inlet Distortion Induced Swirl in Multistage Fans.” In *43rd AIAA/ASME/SAE/ASEE Joint Propulsion Conference & Exhibit*, no. 2007-5059. 15
- [29] Yao, J., Gorrell, S. E., and Wadia, A. R., 2010. “High-Fidelity Numerical Analysis of Per-Rev-Type Inlet Distortion Transfer in Multistage Fans—Part I: Simulations With Selected Blade Rows.” *Journal of Turbomachinery*, **132**(041014). 15
- [30] Yao, J., Gorrell, S. E., and Wadia, A. R., 2010. “High-Fidelity Numerical Analysis of Per-Rev-Type Inlet Distortion Transfer in Multistage Fans—Part II: Entire Component Simulation and Investigation.” *Journal of Turbomachinery*, **132**(041015). 15
- [31] McDaniel, D. R., Nichols, R. H., Eymann, T. A., Starr, R. E., and Morton, S. A., 2016. “Accuracy and Performance Improvements to Kestrel’s Near-Body Flow Solver.” In *54th AIAA Aerospace Sciences Meeting, San Diego, CA*, no. 2016-1051. 16, 19, 20
- [32] Nichols, R. H., McDaniel, D. R., and Klepper, J., 2016. “Mixing Plane Multi-Stage Turbomachinery Simulation Capability for Kestrel/Firebolt.” In *54th AIAA Aerospace Sciences Meeting, San Diego, CA*, no. 2016-0566. 17, 37, 38, 39
- [33] Meakin, R., Atwood, C., and Hariharan, N., 2011. “Development, Deployment, and Support of a Set of Multi-Disciplinary, Physics-Based Simulation Software Products.” In *49th AIAA Aerospace Sciences Meeting, Orlando, FL*, no. 2011-1104. 19
- [34] Morton, S., and McDaniel, D., 2015. “HPCMP CREATE<sup>TM</sup>-AV Kestrel Current Capabilities and Future Direction for Fixed-Wing Aircraft Simulations.” In *53rd AIAA Aerospace Sciences Meeting*, no. 2015-0039. 19
- [35] Morton, S. A., and Meakin, R. L., 2016. “HPCMP CREATE<sup>TM</sup>-AV Kestrel Architecture, Capabilities, and Long Term Plan for Fixed-Wing Aircraft Simulations.” In *54th AIAA Aerospace Sciences Meeting, San Diego, CA*, no. 2016-1051. 19, 30
- [36] McDaniel, D. R., Sears, D. R., Tuckey, T. R., Tillman, B., and Morton, S. A., 2011. “Aerodynamic Control Surface Implementation in Kestrel v2.0.” In *49th AIAA Aerospace Sciences Meeting, Orlando, FL*, no. 2011-1175. 20
- [37] McDaniel, D. R., Nichols, R. H., and Klepper, J. B., 2016. “Unstructured Sliding Interface Boundaries in Kestrel.” In *54th AIAA Aerospace Sciences Meeting, San Diego, CA*, no. 2016-1299. 20
- [38] Klepper, J., Nichols, R. H., and Jenkins, J., 2015. “A-10 Analysis Using HPCMP CREATE<sup>TM</sup>-AV Kestrel Product Utilizing the Firebolt Propulsion Component.” In *53rd AIAA Aerospace Sciences Meeting*, no. 2015-1043. 20

- [39] Menter, F. R., 1994. “Two-Equation Eddy-Viscosity Turbulence Models for Engineering Applications.” *AIAA journal*, **32**(8), pp. 1598–1605. 20, 26, 32
- [40] Law, C., and Puterbaugh, S., 1988. Parametric Blade Study Test Report Rotor Configuration No. 4 Tech. rep., AFWAL-TR-88-2110, Wright-Patterson AFB, OH, Nov. 21, 27
- [41] Cunningham, C., 1988. Swept Rotor Study Tech. rep., AFRL-PR-WP-TR-1998-2040, Wright-Patterson AFB, OH. 21
- [42] Peterson, M. W., Gorrell, S. E., List, M. G., and Custer, C., 2016. “Implementation of Fourier Methods in CFD to Analyze Distortion Transfer and Generation Through a Transonic Fan.” In *52nd AIAA/SAE/ASEE Joint Propulsion Conference*, no. 2016-4746. 32
- [43] Soderquist, D. R., Gorrell, S. E., and List, M. G., 2018. “Radial Variation in Distortion Transfer and Generation Through a Highly Loaded Fan Stage.” In *ASME Turbo Expo 2018: Turbomachinery Technical Conference and Exposition*, no. GT2018-77221, American Society of Mechanical Engineers. 32
- [44] Marshall, M. L., 2014. “Validation of a Modified Version of OVERFLOW 2.2 for Use with Turbomachinery Under Clean and Total Pressure Distorted Conditions and a Study of Blade Loading in Distortion.” Master’s thesis, Brigham Young University-Provo. 52
- [45] Kim, S., Pullan, G., Hall, C., Grewe, R., Wilson, M., and Gunn, E., 2018. “Stall Inception in Low Pressure Ratio Fans.” In *ASME Turbo Expo 2018: Turbomachinery Technical Conference and Exposition*, no. GT2018-75153, American Society of Mechanical Engineers. 59, 66
- [46] Jeong, J., and Hussain, F., 1995. “On the identification of a vortex.” *Journal of Fluid Mechanics*, **285**, pp. 69–94. 59

## **APPENDIX A. GENERATING CAD FILES FOR SINGLE-PASSAGE BLADE ROWS**

The following walkthrough documents the steps taken to generate single-passage geometry files for the rotor and stator rows from the files that were provided by Dr. Michael List at AFRL.

## Create Single Passage Compressor CAD Model

### SolidWorks

Nutshell: Create Basic Compressor file by importing coordinate data and IGES surfaces. Create assembly with cut planes to slice into rotor and stator sections.

#### Create the hub surface

We used the method described for creating our hub geometry. However, if an IGES model for the hub is provided then this method is unnecessary as the IGES file may be imported and used directly.

1. Open new part
2. Features toolbar > Curves > Curve through XYZ points
  - a. In the window that appears click browse and find the hub.txt file in the CD\_ROTOR4 DAT folder
  - b. Click ok
3. Convert to sketch
  - a. Select Front Plane from the part tree and Show it
  - b. Create sketch on that plane
  - c. Click convert entities and select the DAT curve
  - d. Click green check
  - e. Close sketch
4. Create Axis
  - a. Select Right Plane from part tree and Show it
  - b. Features > Reference Geometry > Axis
    - i. Select the Right Plane and the tip of the curve that is farthest from plane
    - ii. Click green check
5. Surface Revolve
  - a. Insert > Surface > Revolved
    - i. Select the sketch and then the axis
    - ii. Click green check
6. Save As

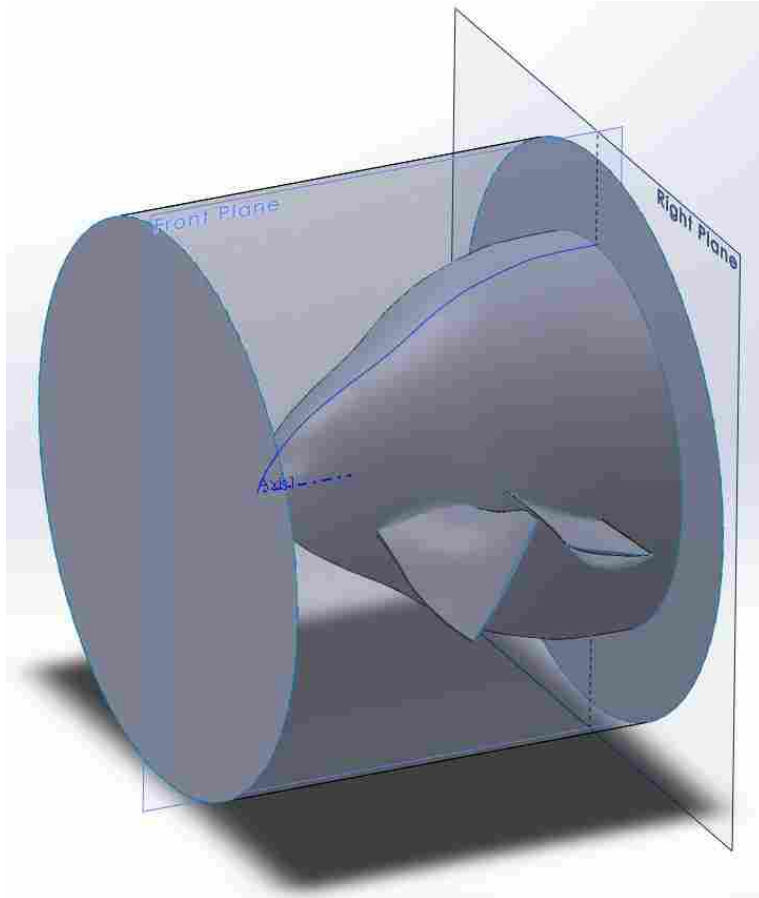
#### Create Blade surfaces by importing blade surfaces

1. Import the Rotor
  - a. Insert > Features > Imported...
    - i. Change file type to IGES and select the rotor\_fillets.igs file from the **fillets** folder
      1. It should automatically import to the correct position
      2. Check to make sure the rotor is oriented to spin clockwise when looking from the front. If the rotor is oriented in the opposite direction then delete the imported feature and import the rotor\_fillets.flipped.igs file instead.
  - b. In the part tree right click the newly imported surface and select feature properties from the drop down menu
    - i. Rename the surface "Rotor"
      1. You should rename every file you import and every operation you perform, it makes hiding and suppressing things a lot easier
2. Import the Tip Gap
  - a. Same steps as above except it is in the **igs** folder, be sure to rename it as well

- b. If Rotor is replaced with the Tip Gap
        - i. Tools > Options > System Options > Import
          - 1. Uncheck "Enable 3D Interconnect"
  - 3. Trim Rotor
    - a. Insert > Surface > Trim
      - i. Select the Tip gap as the trim tool and the rotor as the surface to be trimmed
        - 1. Make sure you select the correct option to keep or remove selection based on what part of the rotor you selected
          - a. Hide the Tip Gap after you have used it to trim the rotor
      - b. Repeat again except select the hub as the trim tool
4. Mirror Rotor (optional)
  - a. Be sure the Axis you created is showing
  - b. Features > Linear Pattern drop down arrow > Circular Pattern
    - i. Click the Bodies option in order to select the rotor
    - ii. Select the axis as the axis of rotation
    - iii. Specify the number of degrees to be # of rotors/360
    - iv. Click the green check
5. Import the Stator
  - a. Same steps as Import Rotor, again be sure to check direction and rename it
6. Import the Casing
  - a. Same steps as Import Rotor except in the igs folder, rename it.
  - b. To improve visibility, right click on the Casing and select Change Transparency
7. Trim the Stator
  - a. Same as rotor trim except select the casing as the trim tool
  - b. Repeat with hub as trim tool
8. Mirror Stator (optional)
  - a. Same as Mirror Stator
9. Save As

### **Create AIP and exit plane, mixing plane, and spinner split plane**

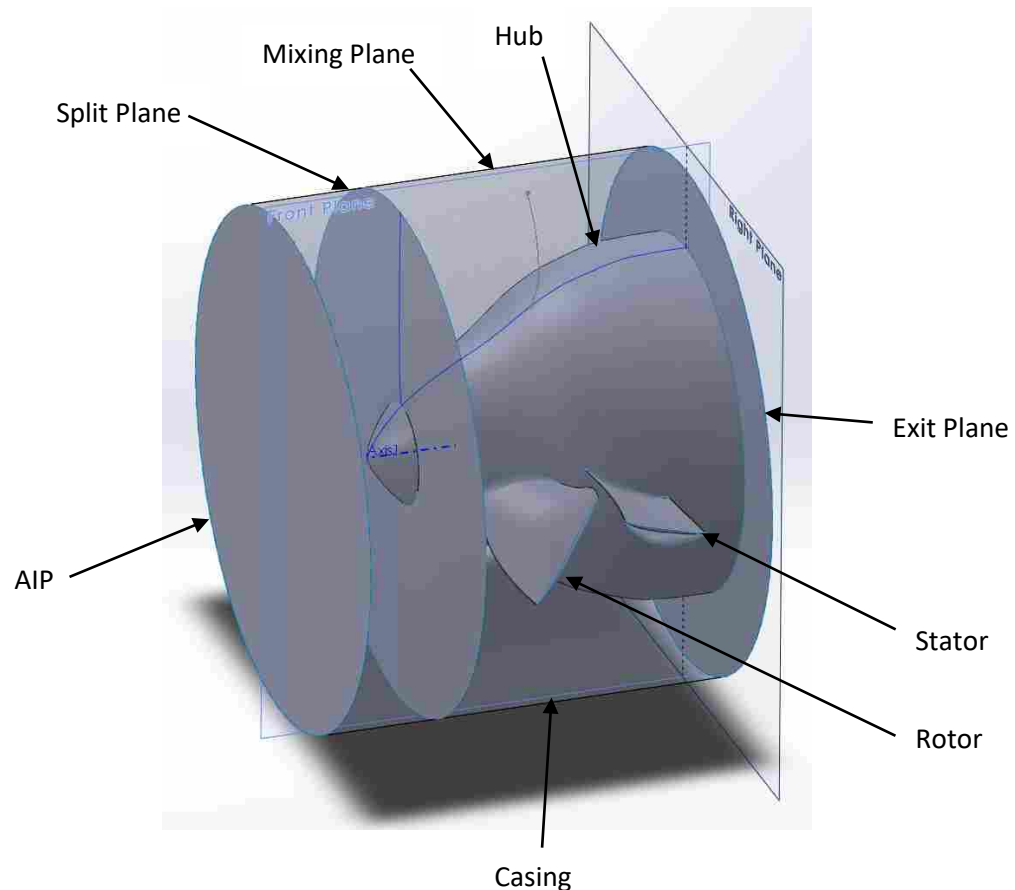
- 1. Create AIP
  - a. Insert > Features > Imported...
    - i. Select aip.igs from igs folder
  - b. Trim the AIP using the casing, then trim the casing using the AIP so there are no exposed ridges
- 2. Create Exit Plane
  - a. Insert > Features > Imported...
    - i. Select outlet.igs from igs folder
  - b. Trim the Exit Plane using the casing and hub, then trim the casing and hub with the Exit Plane so there are no exposed ridges.
  - c.



3. Create a split plane that will define the mixing plane between the rotor and stator rows
  - a. Features toolbar > Curves > Curve through XYZ points
    - i. In the window that appears click browse and find the splitline.txt file in the CD\_ROTOR4 DAT folder
    - ii. Click ok
    - iii. Make sure that this generates a curve that is in between the rotor and the stator rows (it may create the curve in the wrong place and wrong size)
      1. It may be that it has imported it using incorrect units. If this is the case:
        - a. Tools->Options
        - b. Click on the Document Properties tab
        - c. Click on Units
        - d. Select IPS (inch, pound, second) as the Unit system
  - b. Convert to sketch
    - i. Select Front Plane from the part tree and Show it
    - ii. Create sketch on that plane
    - iii. Click convert entities and select the DAT curve
    - iv. Click green check
    - v. Close sketch
2. Create a split plane that will divide the spinner and rotor regions
  - a. Features toolbar > Curves > Curve through XYZ points
    - i. In the window that appears click browse and find the split1.txt (or split2.txt) file in the CD\_ROTOR4 DAT folder

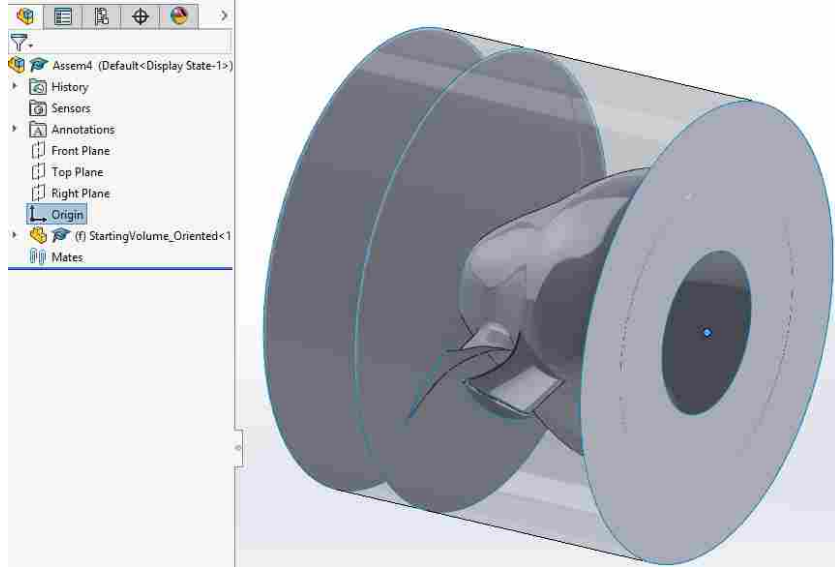


1. The choice of split1.txt or split2.txt may affect the creation of periodic boundaries later on. If one causes difficulties, come back and use the other.
    - ii. Click ok
  - b. Convert to sketch
    - i. Select Front Plane from the part tree and Show it
    - ii. Create sketch on that plane
    - iii. Click convert entities and select the DAT curve
    - iv. Click green check
    - v. Close sketch
  - c. Create Axis (if not already done)
    - i. Select Right Plane from part tree and Show it
    - ii. Features > Reference Geometry > Axis
      1. Select the Right Plane and the tip of the hub
      2. Click green check
  - d. Surface Revolve
    - i. Insert > Surface > Revolved
      1. Select the sketch corresponding to the split plane between the spinner and rotor regions, and then the axis
      2. Click green check
  - e. Trim using the casing and hub as needed
3. Save
  4. The part should look like this:

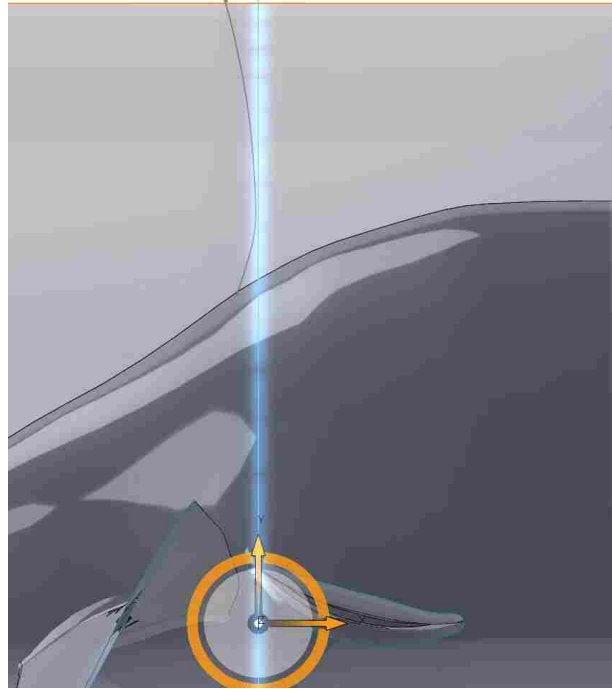


## Create Assembly with Split Plane Part and Basic stage Part

1. Create Rotor part
  - a. Create Assembly
    - i. File > Make Assembly From Part (**You must do this to keep the correct coordinate system and orientation**)
      1. Select the Compressor you created
      2. Click green check
      3. Verify that the Origin is still located in the right spot:



4.
  - ii. Insert the AIP to use as a split plane between the rotor and stator rows for now (this makes generating the periodic boundary surfaces using STAR-CCM+ easier)
    1. Assembly Toolbar > Insert Components
      - a. Select AIP.SLDPRT or hit browse if not shown in open documents box
      - b. Click in the viewing area to place the part
    2. Mate the split Plane and then put it between blade rows
    3. Assembly > Mate
      - a. Select Concentric Mate
      - b. Select the outer edge of this last split plane or the outer edge of the casing
      - c. Click green check
  1. Manual Move: Right click the middle split plane, select Move with Triad from the drop down list
    - a. Change view to a side view and drag the plane in between the rotor and stator
    - b. Make sure that it is placed just barely downstream of the mixing plane curve, but as close to it as possible
      - i. The process of creating the periodic surfaces using Star-CCM+ works better if the downstream boundary is planar
      - ii. The mixing plane curve will be revolved later and used to define the boundary between the rotor and stator region

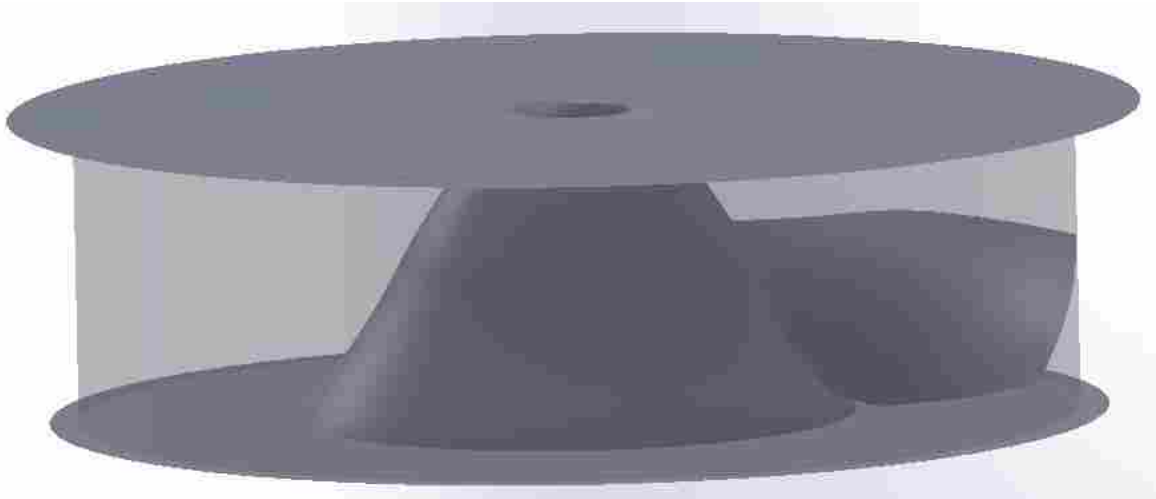


- iii.
- c. When you have it in a good place, right click and select fix so you don't move it by accident later
- c. Save As sldprt (indicate that it is for the rotor)
  - i. Chose "Save the document without rebuilding" if you are given the choice
- 10. Create Stator part
  - a. Move the AIP split plane just upstream of the mixing plane curve
  - b. Save As sldprt (indicate that it is for the stator)

### Make Rotor and Stator Volumes

- 1. Make Rotor Volume
  - a. Open the file for the rotor
  - b. It may help to right click the casing and set it to be transparent
  - c. Trim off the spinner row
    - i. Right Click the compressor part in the part tree and select edit part
      - 1. Insert > Surface > Trim
        - a. Select Front Split Plane as Trim tool
        - b. Select Casing and Hub nose to be trimmed
        - c. Click the green check
      - ii. Get rid of the AIP by right clicking -> delete
    - ii. Get rid of the AIP by right clicking -> delete
  - b. Trim off the stator row
    - i. Same as trimming the front compressor except select the Middle Split Plane as the trimming tool
    - ii. Delete the stator
    - iii. Delete the exit plane
  - c. Trim the Split Planes
    - i. Insert > Surface > Trim
      - 1. Select the Casing as the trim tool
      - 2. Select the Split Plane to be trimmed
      - 3. Do this until nothing extends past the casing

- d. Trim the blade root
  - i. First use the blade surface as a tool to cut out the part of the hub that is inside the blade
  - ii. Then use the hub to trim off the part of the blade surface that extends past the hub
- e. Save As an IGES file
  - i. A box may come up asking if you want to resolve the hidden and suppressed parts before saving, select no



2. Make Stator Volume
  - a. Open the file for the stator
  - b. It may help to right click the casing and set it to be transparent
  - c. Repeat the trimming process from the Rotor Volume except trim away and delete everything forward of the Middle Split Plane
  - d. Save As an IGES file
    - i. A box may come up asking if you want to resolve the hidden and suppressed parts before saving, select no

### **Star-CCM+**

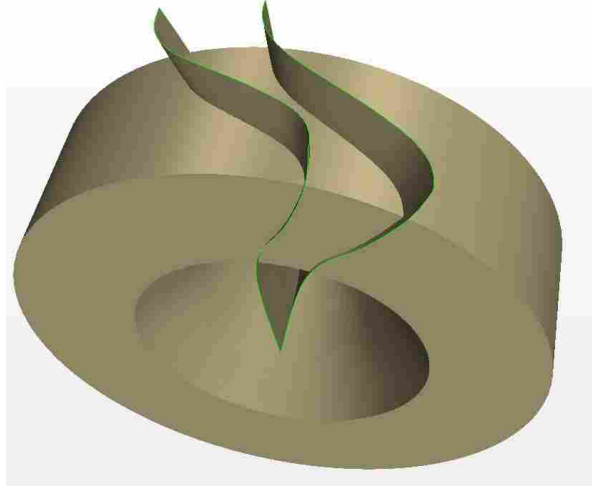
Nutshell: Create single passage volumes by importing the rotor and stator rows into Star and using Turbo Slicing Tool to generate periodic boundaries that follow the curvature of the blade. This will basically follow the steps outlined in the star-ccm+ User Guide (User Guide > Modeling Geometry > Using 3D-CAD > The Turbo Slicing Tool)

### **Rotor Single Passage Volume (Untrimmed)**

1. Launch 3D-CAD in Star-CCM+
  - a. Open Star-CCM+
  - b. File>New Simulation
  - c. In the Work tree on the left, right click Geometry>3D-CAD Models
  - d. select "New"
2. Import Rotor Row volume
  - a. Right click 3D-CAD Model 1 in the work tree

- b. Import>CAD Model
- c. Choose the IGS file that contains the rotor row volume
- 3. Create the Neutral Surface
  - a. Place “Create Turbo Slicing Surface” icon on the toolbar (See “Customizing a Toolbar” in the user guide for how to do this).
  - b. “Create Turbo Slicing Surface” Icon.
  - c. The Turbo Slicing Surface panel appears.
  - d. Specify the type of geometry:
    - i. For axial geometries, set Blade Type to Axial.
  - e. Use the “Select From Scene Tool” to select the faces that make up a single blade.
    - i. To help with the selection, you can hide any obstructing faces (See “Hiding Faces” in the user guide).
  - f. Select the faces that make up the hub.
    - i. You are advised to only select the hub faces that envelop the selected blade, and that define the hub surface of revolution. It is not necessary to select all the faces that make up the hub. Do not select faces that are coincident with upstream or downstream boundaries.
  - g. Select the faces that make up the casing.
    - i. If you hid any faces on the domain, restore them by right-clicking on a blank area in the scene and selecting Restore Hidden Faces.
  - h. Select the faces that correspond to the inlet. The inlet must be planar.
  - i. Select the faces that correspond to the outlet.
  - j. If the neutral surface can be calculated, a preview appears.
  - k. Set the turbo slicing surface parameters so that the neutral surface fits through the center of the blade.
    - i. Also make sure that the areas where the neutral surface meet the inlet and outlet are normal to these boundaries.
    - ii. The surface must fully intersect all domain walls so that you can use it to slice the single blade passage from the overall fluid domain.
    - iii. Parameters that have worked well in the past:
      - 1. Surface Resolution: Very Course (7)
      - 2. Leading Edge/Inlet Transition Shift: 0.5
      - 3. Trailing Edge/Outlet Transition Shift: 0.5
  - l. Click “OK”
- 4. Position Neutral surface on either side of the blade
  - a. Rotate the neutral surface in one direction
    - i. In the part tree, right click on Bodies->Body 3 (or whichever body is the neutral surface just created).
    - ii. Select Transform->Rotate
    - iii. Make sure the neutral surface is selected
    - iv. Change axis direction to 1, 0, 0
    - v. Rotation angle is calculated as  $\frac{360/N_{blades}}{2}$ 
      - 1.  $N_{blades}$  is 20 for the rotor
      - 2. 31 for the stator
    - vi. Set Action to “Copy + Rotate Bodies”
    - vii. Push ok
  - b. Rotate the original neutral surface in the opposite direction

- i. Follow the steps above to rotate by the same angle, only negative
- ii. Make sure Action is set to "Rotate Bodies"
- c. Final product should look like this:



- d.
- 5. Export as a .x\_b file (crashes when I do .igs)
  - a. Right click "3D-CAD Model 1"
  - b. Select "Export..."

#### **Stator Single Passage Volume (Untrimmed)**

- 1. Follow the same steps outlined above, only using the stator volume

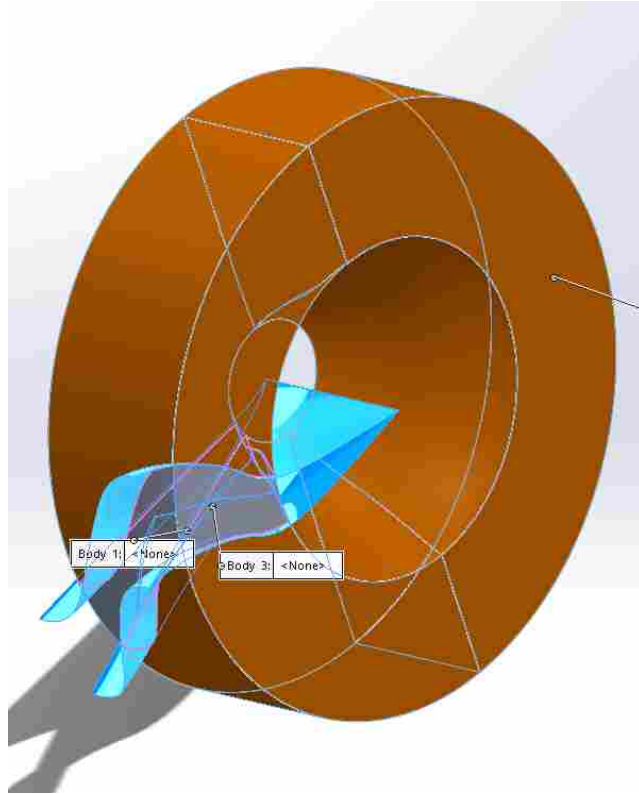
#### **SolidWorks**

Nutshell: Import the untrimmed parasolid (.x\_b) files created using Star-CCM+ and trim the volume into a single passage volume

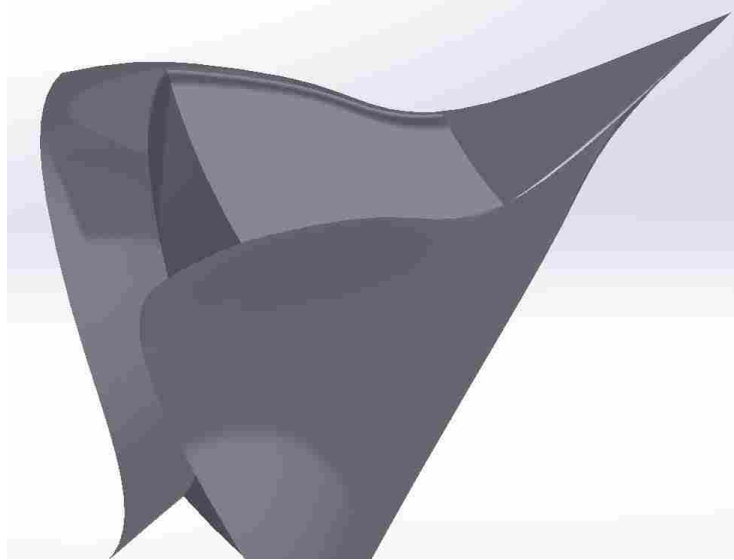
#### **Trim the Rotor and Stator Volumes**

- 1. Open the untrimmed parasolid (.x\_b) file
  - a. You will be asked if you want to run Import Diagnostics hit "Yes"
    - i. Select "Attempt to Heal All"
    - ii. There might be some gaps it cannot heal, that is ok.
  - b. Do not attempt feature recognition
- 2. Create a mixing plane that will divide rotor and stator rows
  - a. Change document units to inches
    - i. Tools->Options
    - ii. Click on the Document Properties tab
    - iii. Click on Units
    - iv. Select IPS (inch, pound, second) as the Unit system
    - v. Ok
  - b. Features toolbar > Curves > Curve through XYZ points
    - i. In the window that appears click browse and find the splitline.txt file in the CD\_ROTOR4 DAT folder
    - ii. Click ok
  - c. Convert to sketch
    - i. Select Front Plane from the part tree and Show it
    - ii. Create sketch on that plane

- iii. Click convert entities and select the DAT curve
    - iv. Click green check
    - v. Exit sketch
  - d. Create Axis (if not already done)
    - i. Select Top Plane from part tree and Show it
    - ii. Features > Reference Geometry > Axis
      - 1. Create using Two Planes
      - 2. Select the Top Plane and the Front Plane
      - 3. Click green check
  - e. Surface Revolve
    - i. Insert > Surface > Revolved
      - 1. Select the sketch and then the axis
      - 2. Click green check
  - f. Split off everything on the wrong side of the mixing plane surface
    - i. Select Insert->Features->Split
    - ii. Select the new revolved surface as the cutting tools
    - iii. Under "Target Bodies" make sure "All bodies" is selected
    - iv. Click "Cut Bodies"
    - v. Click onto the area downstream of the mixing plane (the AIP used to approximate the MP for creating the periodic surfaces)
    - vi. Click the green checkmark at the top left
- 3. Split the full annulus geometry using the periodic surfaces
  - a. The two periodic surfaces should show up as one imported surface
    - i. If they don't: Select Insert->Surface->Knit to knit them both together
  - b. Select Insert->Features->Split
    - i. Select the periodic surfaces as the cutting tools
    - ii. Under "Target Bodies" make sure "All bodies" is selected
    - iii. Click "Cut Bodies"
    - iv. Click on part of the full annulus outside of the cutting surfaces, as well as the new mixing plane surface. They should turn orange:

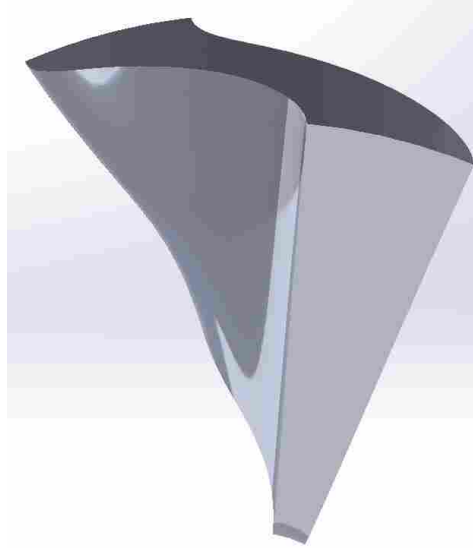


- v.
  - vi. Click the green checkmark at the top left
4. Get rid of unneeded parts of the periodic surfaces



- a.
- b. Insert->Surface->Trim
  - i. Do this on all four faces that aren't the periodic surfaces.






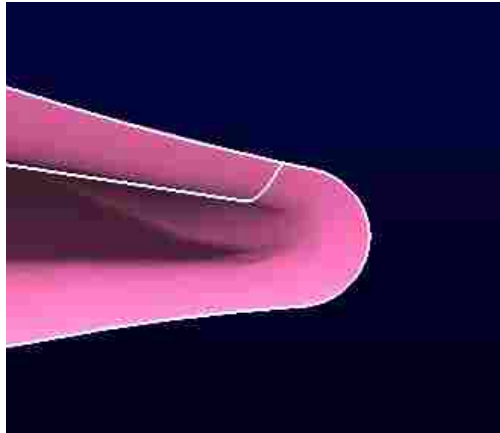
- ii.
5. Save as a .IGES file
6. Repeat these steps for the rotor

## **APPENDIX B. GENERATING POINTWISE MESHES FOR SINGLE-PASSAGE BLADE ROWS**

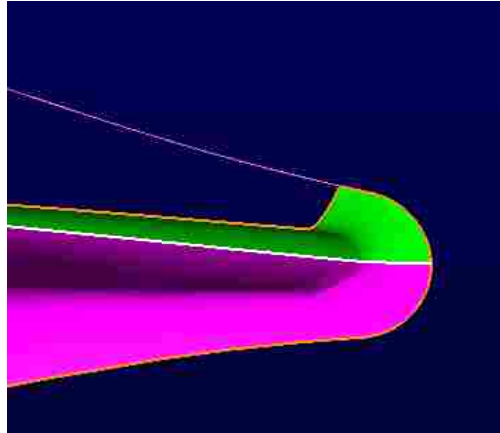
The following walkthrough documents the steps taken to generate single-passage volume meshes from the CAD files generated previously.

## Meshing The Rotor

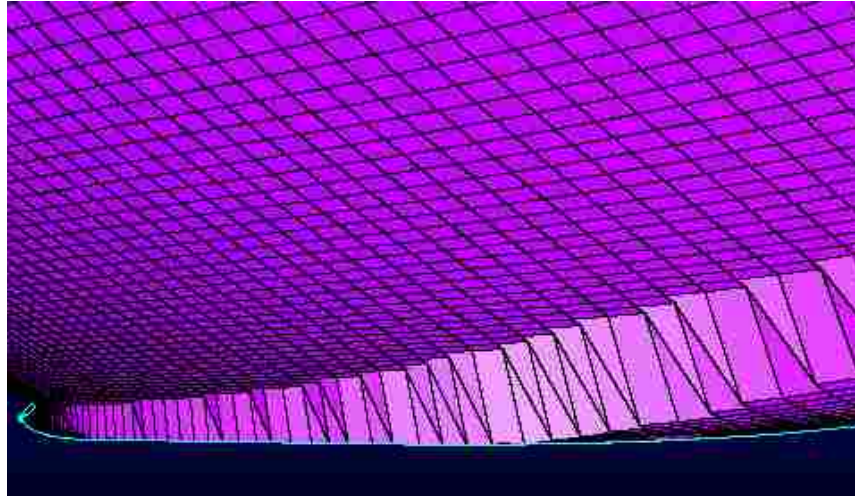
1. Clean up the Database Entities
  - a. Disclaimer: This process is very dependent on how the geometry was created using CAD software, and will likely not be exactly the same for everybody.
  - b. Combine all Models into one Model
    - i. Within the Database group select all “models”
    - ii. Select “Assemble Models” 
    - iii. This should combine all of them into 1 model. If it does not, you may have to combine a few at a time and gradually combine them all together
    - iv. It may help to switch to a shaded view
  - c. Go through the list of Quilts, combine/delete any repeats
    - i. There should be a separate Quilt for each surface/boundary of the computational domain, including one each for the upper and lower surfaces of the blade
2. Create an appropriate leading edge division
  - a. You will notice that the division between the upper and lower blade surfaces near the leading edge is not right at the edge. We will need to change this to have better control over refinement at the leading edge. Essentially, you will split off part of the lower surface quilt at the leading edge and combine it to the upper edge instead



- b.
- c. Create a line on the database that sticks to the leading edge
  - i. Create->Draw Curves->Line on Database
  - ii. Make sure “Entity Type” is set to Database
  - iii. Select a point on the root that is at the leading edge. You can do this visually
    1. Make sure that the quilt that is highlighted is the one the blade
  - iv. Select a point at the leading edge at the tip
  - v. Apply
  - vi. OK
- d. Trim the quilt using the line you just drew
  - i. Select the quilt surface that you just drew a line on
  - ii. Edit->Trim By Curves
  - iii. Select the curve you drew
  - iv. Imprint
  - v. You should get something like this:



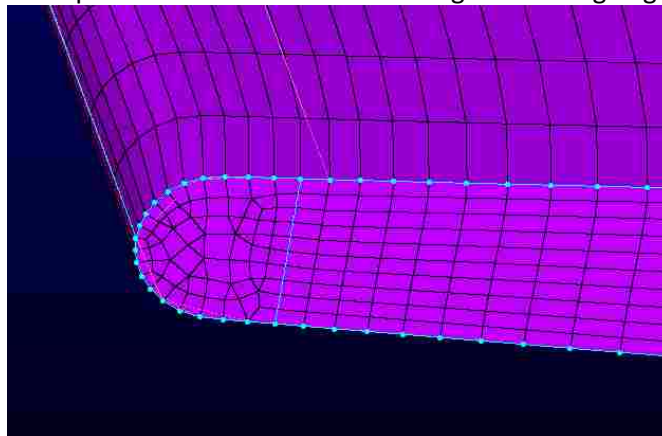
- vi.
- vii. OK
- viii. Now if you look at the Databases, there will be a Quilt Surface that is just that tiny piece you trimmed off
- e. Combine the two quilts that will make up the upper surface of the blade
  - i. "Assemble Quilts"
  - ii. They should now be one quilt
- 3. Create a surface mesh on the blade
  - a. With the blade surfaces selected (not the tip), select Create->On Database Entities...
  - b. Ensure that Connectors is selected
  - c. Check Connector Join Angle and set it to 30
    - i. This makes sure that only one connector is made per edge
  - d. Dimension the connectors
    - i. 121 nodes for the leading and trailing edge connectors
    - ii. 151 for the base and tip connectors
  - e. Distribute nodes on the connectors
    - i. On the root/tip connectors, use a spacing of 0.0015 and tanh distribution
    - ii. On the leading and trailing edge connectors use a spacing of 0.01 at the tip and 0.005 at the root (there is a lot of curvature at the root due to the fillet)
  - f. Create Domains (surface meshes) on the blade surfaces
    - i. Make sure the selected mesh type is Structured
    - ii. Select the connectors on the blade
    - iii. Select "Assemble Domains"
    - iv. You may notice that near the root, the meshes look really bad




- v.
- vi. Select the domains

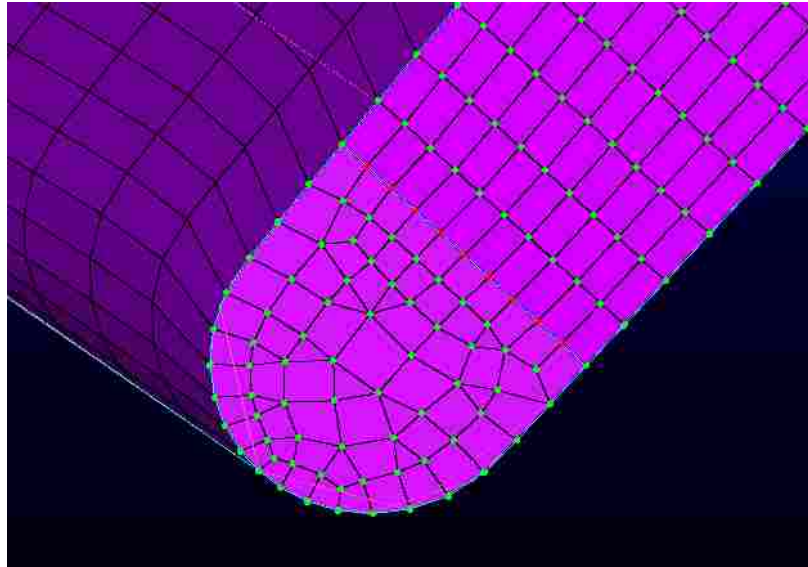


- vii. Hit "Start Solve"
- viii. Let it go for a few iterations until the mesh looks good
- g. Mesh the surface on the tip of the blade
  - i. The tip will look like this at the leading and trailing edge:



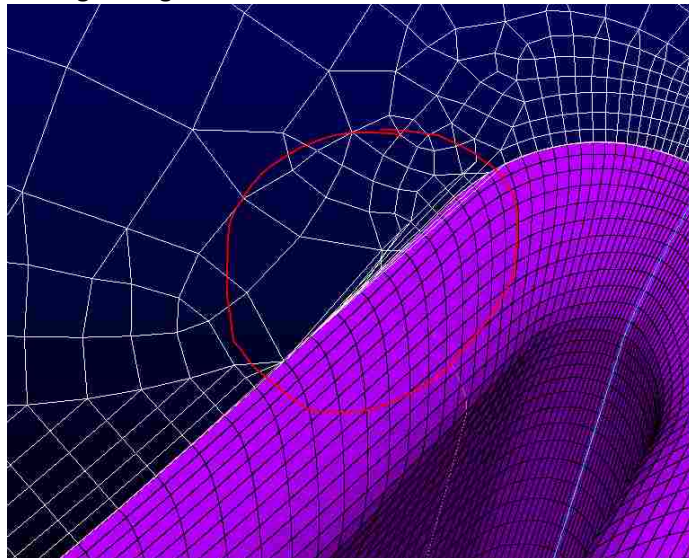
- ii. Draw connectors as 2 point curves at the leading and trailing edges, as shown above (  )
  - 1. This allows most of the grid at the blade tip to be a structured grid
  - 2. Go 10 or 11 nodes back from the leading/trailing edges
  - 3. Dimension the new connectors with 11 nodes each
- iii. After you have created these two new connectors, split the connectors at the tip where they intersect with the new connectors
  - 1. Select the top connector Edit->Split select the two nodes where the two new connectors intersect with the top connector (one at the leading edge, one at the trailing edge)
  - 2. OK
  - 3. Do the same with the lower connector
- iv. Create the unstructured domains in the small areas at the leading/trailing edges
  - 1. Make sure to set them as "Advancing Front Ortho" with quads and triangles


- v. Create the structured mesh in the middle area making up most of the tip
  - 1. Make sure that the structured grid mode is selected
- vi. Check that all three domains making up the blade tip are constrained to the database surface
  - 1. Select all three domains
  - 2. View -> Database Associativity
  - 3. All points should be green except for the new connectors you just made:



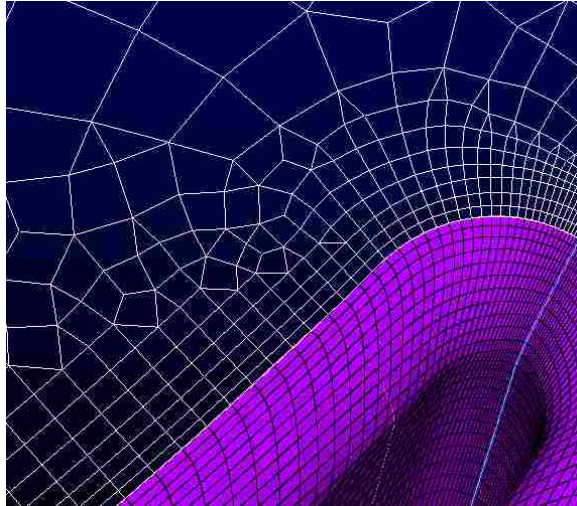
- 4. That should be ok. One way to remedy this is to use “Draw Curves on Database Entities” when creating the connectors, but I had trouble getting this to work.
4. Create surface mesh on hub
    - a. With the hub surface selected, click “Connectors on Database Entities”
      - i. This will create connectors where the hub meets the blade, where connectors already exist. Select the new connectors that were created there and get rid of them
    - b. Dimension connectors
      - i. You may dimension these however you see fit, but the following values are what worked for me
      - ii. Upstream edge: 21
      - iii. Downstream edge: 41
      - iv. Periodic edges: 101
    - c. Distribute nodes
      - i. Do this as you see fit
    - d. Generate the mesh
      - i. Make sure that Unstructured Domains is selected
      - ii. Create->Assemble Special->Domain...
      - iii. Click on one of the outside connectors of the hub
        1. If Auto Complete is checked, it should have automatically selected all 4 outside connectors
        2. If not, make sure they are all selected
      - iv. Save Edge
      - v. Select both of the inner connectors (where the blade meets the hub)

- vi. Make sure that the orientations of the inner and outer edge are opposite (one should be clockwise and one counter)
  - 1. Click “Flip Edge Orientation” if you need to
- vii. Save edge
- viii. Apply
- ix. Make the mesh look good
  - 1. Mesh type
    - a. I like “Advancing Front Ortho” with Triangles and Quads selected, but I have no idea if one kind of mesh gives better results than another
    - b. The guys I worked with at AEDC used it
  - 2. Use T-Rex cells to grow some layers away from the blade root
    - a. This improves cell quality
    - b. Set the two connectors at the blade root as type “Wall”, set  $\Delta x$  to be the same as (or close to) the  $\Delta x$  at the root of the leading and trailing edge connectors on the blade.
    - c. This makes it so the surface cell sizes don’t abruptly change where the hub meets the blade, improving mesh quality
    - d. 0 full layers
    - e. As many max layers as looks good to you (I did 12)
    - f. Growth Rate: 1.2
    - g. Cell Type: Triangles and quads
  - 3. Initialize Mesh
  - 4. You may get some weird behavior where the T-Rex cells were supposed to be growing:



- 5. This is a bug in the current version of pointwise (18.0 R4) and has been reported. It will hopefully be fixed in future iterations
- 6. Constrain the domain (surface mesh) to a BSurf instead of to a quilt
  - a. While viewing the list of databases, select View->Show Hidden
  - b. Find which BSurf corresponds to the hub surface
  - c. With it selected, click Show (  )
  - d. Un-show hidden

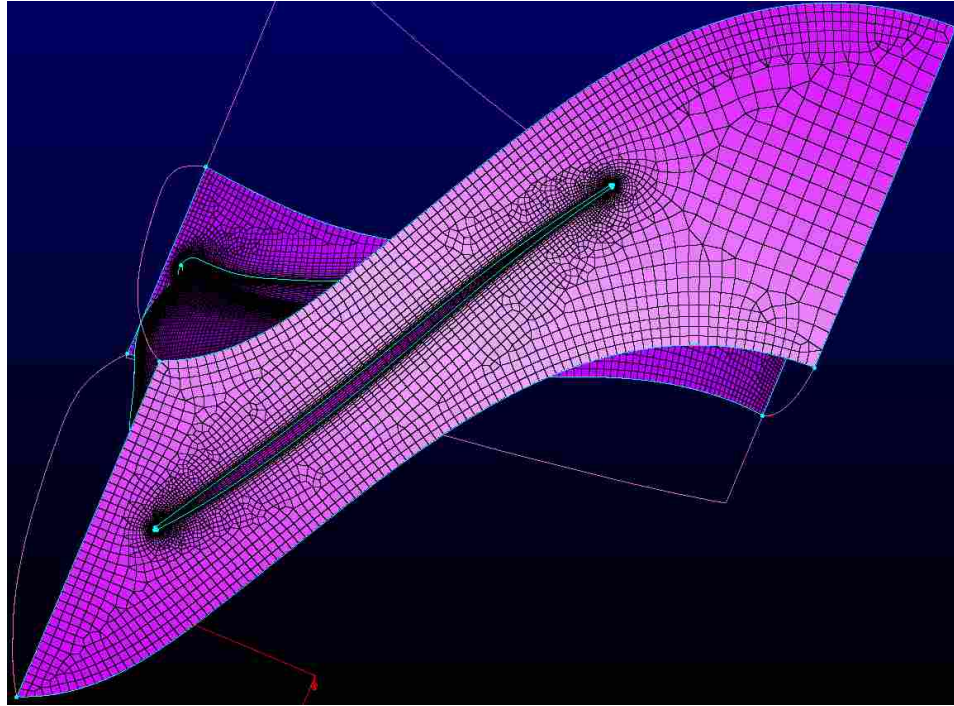
- e. Now with the problem domain selected, select Grid->Solve
- f. In the attributes tab, under Target Database Selection, select Begin, choose the BSurf, select end
- g. Initialize the mesh. It should look all better:



- 5. Create Domain on Case Surface
  - a. Copy and paste the domains on the tip to project them onto the casing to get good cell size matching
    - i. Select the three tip domains
    - ii. Ctrl+c
    - iii. Ctrl+v
    - iv. Select Project...
    - v. Uncheck "Interior Only"
    - vi. Check "Target Database Selection"
    - vii. Begin
    - viii. Choose the quilt that is the case
    - ix. End
    - x. Click "Project"
    - xi. You may play with the Type, but Closest Point seems to work fine
    - xii. OK
    - xiii. OK
    - xiv. You should now have copies of the three tip domains on the casing surface
  - b. Create Connectors on the 4 edges of the casing surface
    - i. Select the quilt making up the casing
    - ii. Connectors on Database Entities
  - c. Dimension the connectors
    - i. On the two periodic edges: 101 nodes (same as on the hub edges)
    - ii. On the upstream edge: 21 nodes
    - iii. On the downstream edge: 41 nodes
  - d. Generate the mesh
    - i. Make sure that Unstructured Domains is selected
    - ii. Create->Assemble Special->Domain...
    - iii. Click on one of the outside connectors of the casing
      - 1. If Auto Complete is checked, it should have automatically selected all 4 outside connectors



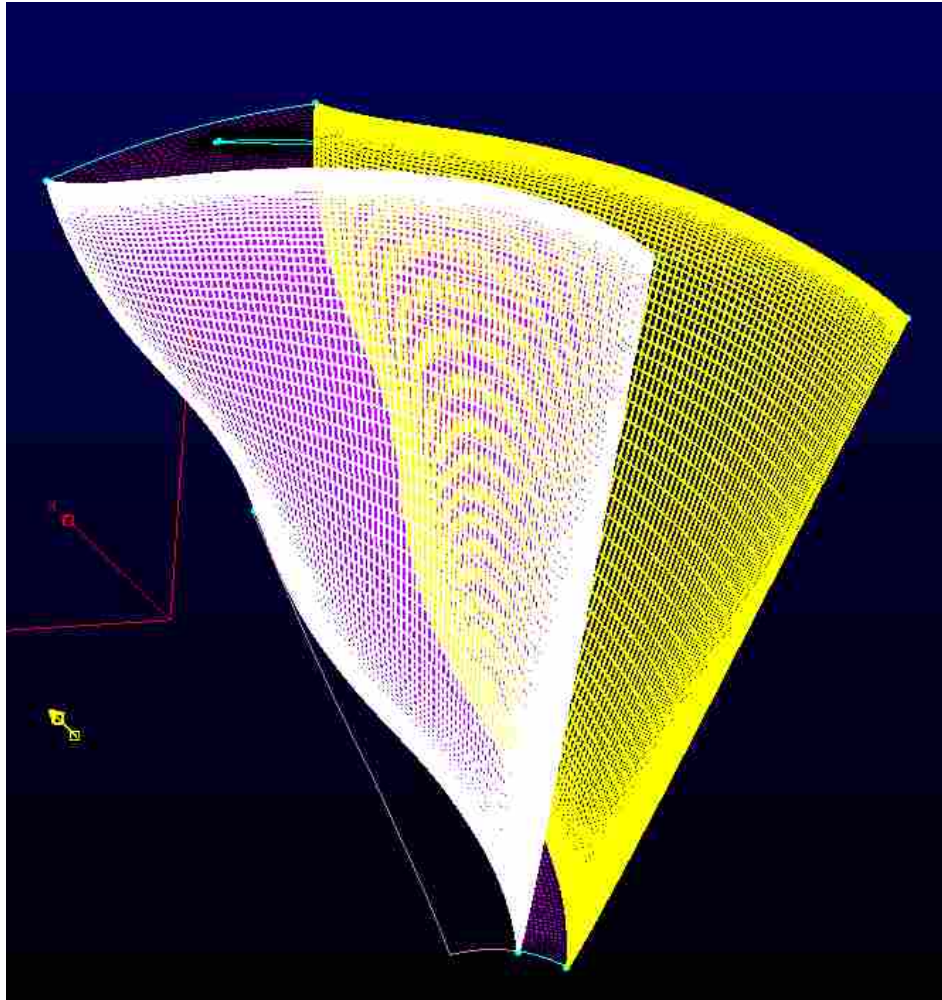
2. If not, make sure they are all selected
- iv. Save Edge
- v. Select the inner connectors that make up the tip blade profile
  1. You may need to unselect Auto complete so it doesn't select one of the two connectors inside the airfoil profile
- vi. Make sure that the orientations of the inner and outer edge are opposite (one should be clockwise and one counter)
  1. Click "Flip Edge Orientation" if you need to
- vii. Save edge
- viii. Apply
- ix. Make sure that the domain is Advancing Front Ortho, with Triangles and Quads
- e. Make the mesh look good
  - i. Similar to what was done on the hub, we will use T-Rex cells for good transition from the refined region near the blade tip to the rest of the casing
  - ii. With the casing domain selected (not the three inner domains), Grid->Solve->T-Rex
    1. Select the 6 inner connectors of the domain that trace the tip outline
    2. New
    3. Type -> Wall
    4. Set  $\Delta s$  to something similar to the grid spacing of the inner structured domain, so the cells growing off of the connector are of similar size on both sides of the connector (for me it was about 0.002)
    5. Go to the T-Rex tab
      - a. These settings are not set in stone: use your best judgement. The following are what I used, and what worked for me
      - b. Max Layers: 15
      - c. Full Layers: 3
      - d. Growth Rate: 1.3
      - e. Triangles and Quads
    6. In the Attributes tab, set boundary decay to 0.9 if you so desire
    7. Initialize the mesh



- f.
6. Create one of the periodic domains
  - a. Create Connectors on the edges of the periodic surface
    - i. Select one of the quilts representing the periodic surface
    - ii. Create -> On Database Entities...
    - iii. Connector Join Angle: 30
    - iv. OK
    - v. Delete the duplicate connectors that were created on the hub and casing edges
  - b. Dimension the two remaining connectors: 121
  - c. Select the 4 connectors making up the edges
  - d. Make sure Structured Domain is selected
  - e. Assemble Domains
  - f. Distribute the inflow and outflow connectors so that there is good boundary layer spacing at the hub and casing surfaces
    - i. Choose the connectors at the edges of the inflow and outflow boundaries
    - ii. Grid->Distribute
    - iii. Set Spacing to 0.0001
    - iv. In the Functions tab, change Function to Growth
    - v. In Growth Distribution Parameters make sure that Begin and End are set to Number of Layers and Growth Rate
    - vi. Set number of layers to 30
    - vii. Growth Rate to 1.2
    - viii. Apply

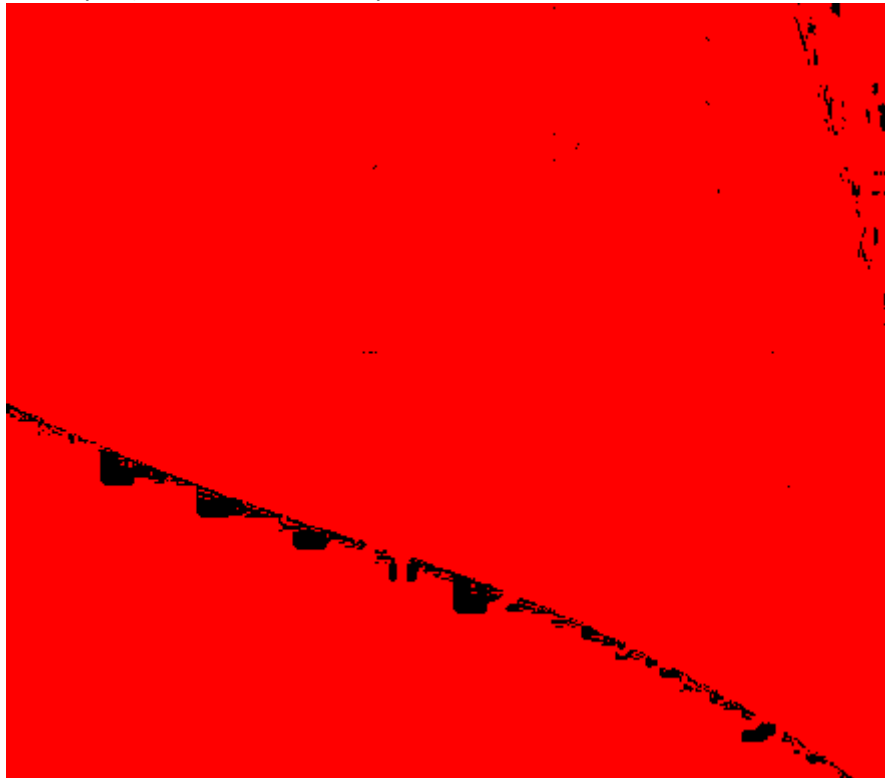


- e. The domain to be created will be previewed in yellow:




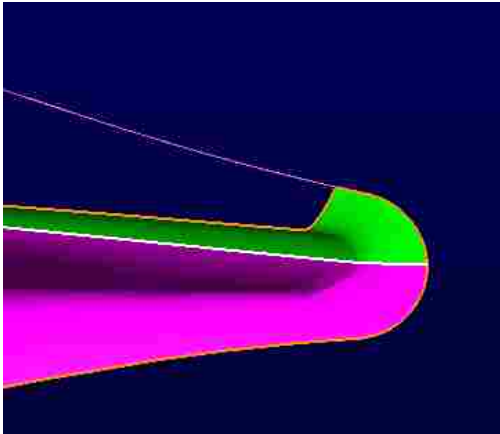
- f. Make sure that there are no duplicate connectors where the new periodic domain is (at the hub and casing edges)
- i. If there are, use Grid->Merge
  - ii. Check Merge by Picking
  - iii. Replace 1<sup>st</sup> with 2<sup>nd</sup>
  - iv. Sometimes this isn't necessary (pointwise automatically gets rid of the duplicate connectors)
8. Create the domains on the inflow and outflow boundaries
- a. At this point, the connectors at these boundaries should all exist
  - b. Select all of these connectors (8 in total)
  - c. Make sure Structured Mesh is selected
  - d. Assemble Domains
9. Create the Volume Mesh
- a. Make sure Unstructured is selected
  - b. Create->Assemble Special->Block...
  - c. Choose the Automatic Tab
  - d. Draw a selection box around everything to select all domains
  - e. Assemble Faces
  - f. OK


- g. Select the Block
- h. Grid->T-Rex
- i. Max Layers: 30
- j. Full Layers: 12
- k. Growth Rate: 1.2
- l. Check Push Attributes
- m. Select All – Tets, Pyramids, Prisms and Hexes
- n. In Boundary Conditions tab, create 2 new boundary conditions
  - i. Name them “wall” and “fluid”
  - ii. Select all wall surfaces (blade surfaces, hub, and casing - should be 10 in total) and make them the “wall” bc. Change its type to Wall with a spacing of 0.0001.
  - iii. Select the 4 remaining surface grids and put them under the “fluid” bc. Change its type to “Match”.
- o. Initialize
- p. You may run into errors. Many of them stem from the same problems that caused the bad surface meshes on the hub (not being constrained to a good surface)
- q. Constrain the problem surfaces to the appropriate Bsurfs as you did before (I don’t think it is necessary to do this with the copied periodic surface)
- r. I also ran into problems where the T-rex layers would halt early (before the specified Full Layers), often within 1-3 layers:



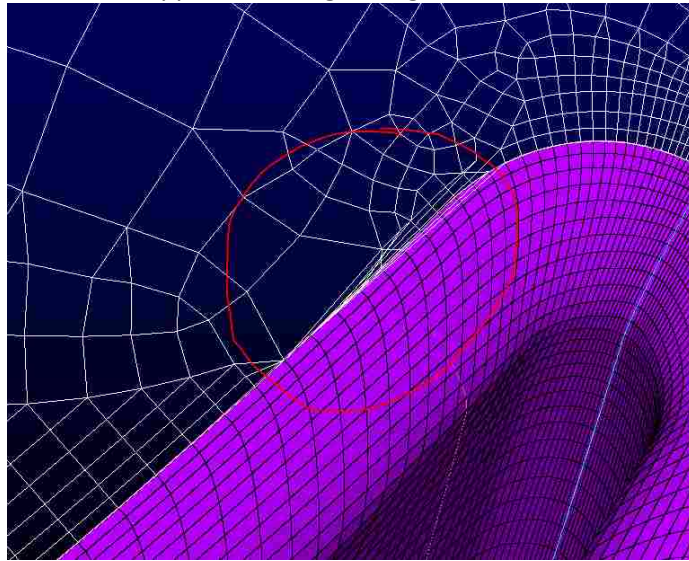
- s. I had to change the number of nodes on those edges until it worked (I went lower so that there could be fewer overall cells in the grid). I had to change those edges to have 95 nodes instead of 101, your experience may be different (you only need to change the number of nodes on the top and bottom edges of the first periodic surface, since the second surface mirrors it). As long as it can get to about 10-12 full layers before it has to stop, it is probably sufficient to resolve the boundary layer. 15 would be better though.

## Meshing The Stator

1. Clean up the Database Entities
  - a. Combine all Models into one Model
    - i. Within the Database group select all “models”  

    - ii. Select “Assemble Models”
    - iii. This should combine all of them into 1 model. If it does not, you may have to combine a few at a time and gradually combine them all together
    - iv. It may help to switch to a shaded view
  - b. Go through the list of Quilts, combine/delete any repeats
    - i. There should be a separate Quilt for each surface/boundary of the computational domain, including one each for the upper and lower surfaces of the blade
2. Create an appropriate leading edge division
  - a. You will notice that the division between the upper and lower blade surfaces near the leading edge is not right at the edge. We will need to change this to have better control over refinement at the leading edge. Essentially, you will split off part of the lower surface quilt at the leading edge and combine it to the upper edge instead
  - b. Create a line on the database that sticks to the leading edge
    - i. Create->Draw Curves->Line on Database
    - ii. Make sure “Entity Type” is set to Database
    - iii. Select a point on the root that is at the leading edge. You can do this visually
      1. Make sure that the quilt that is highlighted is the one the blade
    - iv. Select a point at the leading edge at the tip
    - v. Apply
    - vi. OK
  - c. Trim the quilt using the line you just drew
    - i. Select the quilt surface that you just drew a line on
    - ii. Edit->Trim By Curves
    - iii. Select the curve you drew
    - iv. Imprint
    - v. You should get something like this:  

    - vi.
    - vii. OK
    - viii. Now if you look at the Databases, there will be a Quilt Surface that is just that tiny piece you trimmed off
  - d. Combine the two quilts that will make up the upper surface of the blade

- i. "Assemble Quilts"
    - ii. They should now be one quilt
- 3. Create a surface mesh on the blade
  - a. With the blade surfaces selected (not the tip), select Create->On Database Entities...
  - b. Ensure that Connectors is selected
  - c. Check Connector Join Angle and set it to 30
    - i. This makes sure that only one connector is made per edge
  - d. Dimension the connectors
    - i. 121 nodes for the leading and trailing edge connectors
    - ii. 151 for the base and tip connectors
  - e. Distribute nodes on the connectors
    - i. On the root/tip connectors, use a spacing of 0.0025 and tanh distribution
    - ii. On the leading and trailing edge connectors use a spacing of 0.01 at the tip and at the root
  - f. Create Domains (surface meshes) on the blade surfaces
    - i. Make sure the selected mesh type is Structured
    - ii. Select the connectors on the blade
    - iii. Select "Assemble Domains"
    - iv. Select the blade domains
    - v. Hit "Start Solve" 
    - vi. Let it go for a few iterations until the mesh looks good
- 4. Create surface mesh on hub
  - a. With the hub surface selected, click "Connectors on Database Entities"
    - i. This will create connectors where the hub meets the blade, where connectors already exist. Select the new connectors that were created there and get rid of them
  - b. Dimension connectors
    - i. You may dimension these however you see fit, but the following values are what worked for me
    - ii. Upstream edge: 31
    - iii. Downstream edge: 41
    - iv. Periodic edges: 81
  - c. Distribute nodes
    - i. Do this as you see fit
  - d. Generate the mesh
    - i. Make sure that Unstructured Domains is selected
    - ii. Create->Assemble Special->Domain...
    - iii. Click on one of the outside connectors of the hub
      - 1. If Auto Complete is checked, it should have automatically selected all 4 outside connectors
      - 2. If not, make sure they are all selected
    - iv. Save Edge
    - v. Select both of the inner connectors (where the blade meets the hub)
    - vi. Make sure that the orientations of the inner and outer edge are opposite (one should be clockwise and one counter)
      - 1. Click "Flip Edge Orientation" if you need to
    - vii. Save edge

- viii. Apply
- ix. Make the mesh look good
  - 1. Mesh type
    - a. "Advancing Front Ortho" with Triangles and Quads selected
    - b. The guys I worked with at AEDC used it
  - 2. Use T-Rex cells to grow some layers away from the blade root
    - a. This improves cell quality
    - b. Set the two connectors at the blade root as type "Wall", set  $\Delta x$  to be the same as (or close to) the  $\Delta x$  at the root of the leading and trailing edge connectors on the blade (.01).
    - c. This makes it so the surface cell sizes don't abruptly change where the hub meets the blade, improving mesh quality
    - d. 1 full layer
    - e. As many max layers as looks good to you (I did 11)
    - f. Growth Rate: 1.2
    - g. Cell Type: Triangles and quads
  - 3. Initialize Mesh
  - 4. You may get some weird behavior as with the rotor where the T-Rex cells were supposed to be growing:



- a. Fix the same way as was done for the rotor
- 5. Create Domain on Case Surface
  - a. With the casing surface selected, click "Connectors on Database Entities"
    - i. This will create connectors where the hub meets the blade, where connectors already exist. Select the new connectors that were created there and get rid of them
  - b. Dimension connectors
    - i. Use the same dimensions that you used for the hub domain
  - c. Distribute nodes
    - i. Do this as you see fit
  - d. Generate the mesh
    - i. Make sure that Unstructured Domains is selected
    - ii. Create->Assemble Special->Domain...



- iii. Click on one of the outside connectors of the casing
    - 1. If Auto Complete is checked, it should have automatically selected all 4 outside connectors
    - 2. If not, make sure they are all selected
  - iv. Save Edge
  - v. Select both of the inner connectors (where the blade meets the casing)
  - vi. Make sure that the orientations of the inner and outer edge are opposite (one should be clockwise and one counter)
    - 1. Click "Flip Edge Orientation" if you need to
  - vii. Save edge
  - viii. Apply
  - ix. Make the mesh look good
    - 1. Mesh type
      - a. "Advancing Front Ortho" with Triangles and Quads selected
    - 2. Use T-Rex cells to grow some layers away from the blade root
      - a. This improves cell quality
      - b. Set the two connectors at the blade root as type "Wall", set  $\Delta x$  to be the same as (or close to) the  $\Delta x$  at the root of the leading and trailing edge connectors on the blade (.01).
      - c. This makes it so the surface cell sizes don't abruptly change where the hub meets the blade, improving mesh quality
      - d. 1 full layer
      - e. As many max layers as looks good to you (I did 11)
      - f. Growth Rate: 1.2
      - g. Cell Type: Triangles and quads
    - 3. Initialize Mesh
    - 4. You may get the same weird behavior as before where the T-Rex cells were supposed to be growing. Fix in the same way as before.
6. Create one of the periodic domains
- a. Create Connectors on the edges of the periodic surface
    - i. Select one of the quilts representing the periodic surface
    - ii. Create -> On Database Entities...
    - iii. Connector Join Angle: 30
    - iv. OK
    - v. Delete the duplicate connectors that were created on the hub and casing edges
  - b. Dimension the two remaining connectors: 121
  - c. Select the 4 connectors making up the edges
  - d. Make sure Structured Domain is selected
  - e. Assemble Domains
  - f. Distribute the inflow and outflow connectors so that there is good boundary layer spacing at the hub and casing surfaces
    - i. Choose the connectors at the edges of the inflow and outflow boundaries
    - ii. Grid->Distribute
    - iii. Set Spacing to 0.0001
    - iv. In the Functions tab, change Function to Growth
    - v. In Growth Distribution Parameters make sure that Begin and End are set to Number of Layers and Growth Rate
    - vi. Set number of layers to 30

- vii. Growth Rate to 1.2
  - viii. Apply
- 7. Use periodic rotate to create the other periodic domain that is a point-to-point match with the first periodic domain
  - a. Select the first periodic domain
  - b. Create->Periodic->Rotate
  - c. Use the same parameters as with the rotor, only the angle should be 360/31 (11.61290323)
  - d. You may have to do -11.61290323 for the angle depending on which domain you created initially.
  - e. The domain to be created will be previewed in yellow
  - f. Make sure that there are no duplicate connectors where the new periodic domain is (at the hub and casing edges)
    - i. If there are, use Grid->Merge
    - ii. Check Merge by Picking
    - iii. Replace 1<sup>st</sup> with 2<sup>nd</sup>
    - iv. Sometimes this isn't necessary (pointwise automatically gets rid of the duplicate connectors)
- 8. Create the domains on the inflow and outflow boundaries
  - a. At this point, the connectors at these boundaries should all exist
  - b. Select all of these connectors (8 in total)
  - c. Make sure Structured Mesh is selected
  - d. Assemble Domains
- 9. Create the Volume Mesh
  - a. Make sure Unstructured is selected
  - b. Create->Assemble Special->Block...
  - c. Choose the Automatic Tab
  - d. Draw a selection box around everything to select all domains
  - e. Assemble Faces
  - f. OK
  - g. Select the Block
  - h. Grid->T-Rex
  - i. Max Layers: 40
  - j. Full Layers: 20
  - k. Growth Rate: 1.2
  - l. Check Push Attributes
  - m. Select All – Tets, Pyramids, Prisms and Hexes
  - n. In Boundary Conditions tab, create 2 new boundary conditions
    - i. Name them "wall" and "fluid"
    - ii. Select all wall surfaces (blade surfaces, hub, and casing) and make them the "wall" bc. Change its type to Wall with a spacing of 0.0001.
    - iii. Select the 4 remaining surface grids and put them under the "fluid" bc. Change its type to "Match".
  - o. Initialize
  - p. You may run into errors. Many of them stem from the same problems that caused the bad surface meshes on the hub (not being constrained to a good surface)
  - q. Constrain the problem surfaces to the appropriate Bsurfs as you did before (I don't think it is necessary to do this with the copied periodic surface)

- r. I also ran into problems where the T-rex layers would halt early (before the specified Full Layers), often within 1-3 layers:



- s. I had to change the number of nodes on those edges until it worked (I went lower so that there could be fewer overall cells in the grid). As long as it can get to about 10-12 full layers before it has to stop, it is probably sufficient to resolve the boundary layer. 15 is best though.

## APPENDIX C. UTILITY SCRIPTS

Various scripts were used throughout the process of running simulations and post-processing results. These scripts, along with a description of what they were used for, are included below.

### C.1 Circumvolving a Single-passage Mesh Using Carpenter

The following scripts were used to submit a job to circumvolve a single-passage mesh to a full annulus mesh. Submitting as a job was necessary because the login node often did not have enough memory to create the full annulus grid, so a compute node needed to be requested.

#### C.1.1 Submission Script

```
1 #!/bin/bash
2
3 #PBS -N jobname
4 #PBS -l walltime=00:15:00
5 #PBS -l select=1:mpiprocs=32:ncpus=32
6 #PBS -l bigmem=1
7 #PBS -l ccm=1
8 #PBS -A [projectID]
9 #PBS -q standard
10
11 module load create
12 module load ccm
13 echo start script
14
15 kiedir=/usr/cta/restricted/create/av/kestrel/current/bin
16
```

```
17 echo kiedir is $kiedir
18
19 cd /path/to/singlepassage/grids/.
20
21 ccmrun /p/home/elynch/carp921_for_ccmrun.sh < inputs.txt
```

## C.1.2 Carpenter Inputs

This script contains the inputs that Carpenter uses while creating the full annulus mesh from the single-passage mesh. The user will need to modify file names and blade numbers appropriately. More information about using Carpenter to create a full annulus blade row can be found in the “Full-Annulus, Rotor 67” tutorial for Kestrel.

```
1 2
2 1
3 1
4 singlepassagegrid.avm
5 3
6 0
7 15
8 4
9 5
10 6
11 31
12 31
13 1 0 0
14 0 0 0
15 n
16 0
17 2
18 5
19 1
20 1
21 fullannulusgrid.avm
```

```
22 0
23 0
```

## C.2 Monitoring Jobs

The process for monitoring a job as it was running involved first executing the compiled Fortran script below. This would write several \*.out files that could be read and plotted using the Gnuplot script below.

### C.2.1 Fortran Script

The following script reads the \*.out files created by the Fortran executable.

```
1      program massflow
2
3      c      extract information from surf.out for plotting
4      c      variables in surf.out file:
5      c      1 - Iteration
6      c      2 - Time
7      c      3 - Total Temperature
8      c      4 - Total Pressure
9      c      5 - Static temperature
10     c      6 - Static pressure
11     c      7 - Density
12     c      8 - U velocity
13     c      9 - V velocity
14     c      10 - W velocity
15     c      11 - Massflow
16
17     implicit none
18
19     integer :: idum,m,n,nfile,ntot,hlines
20     integer,allocatable :: ncnt (:),ITER(:)
21     real :: d1,d2,d3,d4,d5,d6,d7,d8,d9,d10,d11
```

```

22  real :: d12,d13,d14,d15,d16,d17,d18,d19,d20,d21
23  real :: cc,vv,ratp,ratt
24  real :: err1,err2,err3
25  real :: ptra,ttra,effa,gml,mout
26  real,allocatable :: TT(:, :),PT(:, :),TS(:, :),PS(:, :),RHO(:, :)
27  real,allocatable :: U(:, :),V(:, :),W(:, :),MASS(:, :),MACH(:, :)
28  real,allocatable :: TIME(),AREA(:, :)
29  real,allocatable :: MERR(),MECORR()
30  real :: rhoa,vxa,vya,vza,pa,ta,pta,tta,vna
31  character*160,allocatable :: filename()
32  character*160 :: file_prefix
33  character*4 :: txt
34  real,allocatable :: nblades()
35  real :: mtarget
36  real, PARAMETER :: GAMMA=1.4,PTINF=14.69,TTINF=518.7
37  real, PARAMETER :: PTSTD=14.6959449525,TTSTD=518.67
38  logical :: exist
39
40 c  average stuff
41
42  integer :: istart,isamp
43  real,allocatable :: mavg(),ptavg(),ttavg()
44
45 c  get file names
46  nfile = 8
47  allocate(ncnt(nfile))
48  allocate(filename(nfile))
49  allocate(mavg(nfile),ptavg(nfile),ttavg(nfile))
50
51  allocate(nblades(nfile))
52  mtarget = 0.158040618*32.17*12.0
53  nblades(1) = 1.0
54  nblades(2) = 1.0
55  nblades(3) = 1.0
56  nblades(4) = 1.0
57  nblades(5) = 1.0

```

```

58     nblades(6) = 1.0
59     nblades(7) = 1.0
60     nblades(8) = 1.0
61
62     file_prefix =
63     & '/path/to/run/directory/tracking/'
64     filename(1) = trim(file_prefix)//'spinner/aip/stage4.props'
65     filename(2) = trim(file_prefix)//'spinner/exit/stage4.props'
66     filename(3) = trim(file_prefix)//'rotor/inflow/stage4.props'
67     filename(4) = trim(file_prefix)//'rotor/rotorOut/stage4.props'
68     filename(5) = trim(file_prefix)//'stator/inflow/stage4.props'
69     filename(6) = trim(file_prefix)//'stator/outflow/stage4.props'
70     filename(7) = trim(file_prefix)//'nozzle/inlet/stage4.props'
71     filename(8) = trim(file_prefix)//'nozzle/outlet/stage4.props'
72
73 c     check if files exist
74
75     do n=1,nfile
76         inquire(file=filename(n), exist=exist)
77         if( exist ) then
78             print *, 'The file exists! ',n,trim(filename(n))
79         else
80             print *, 'The file does not exist ',n,trim(filename(n))
81             stop
82         endif
83     enddo
84
85 c     get max number of iterations
86     do n=1,nfile
87         ncnt(n) = 0
88         open(1,file=filename(n),form='formatted')
89
90 c     skip header lines
91         hlines=24
92         do m=1,hlines
93             read(1,*,END=98,ERR=98) txt

```



```

94     enddo
95     goto 99
96  98   print *, 'Bad file header for file ', trim(filename(n))
97     stop
98  99   continue
99
100     do m=1, 999999999
101         read(1, *, END=101, ERR=101) idum,
102     &             d1, d2, d3, d4, d5, d6, d7, d8, d9, d10,
103     &             d11, d12, d13, d14, d15, d16, d17, d18, d19, d20, d21
104         ncnt(n) = ncnt(n)+1
105     enddo
106  101   continue
107     close(1)
108     enddo
109  100   format (i10, 1x, 11(1x, 1pe12.5))
110     ntot = 999999999
111     do n=1, nfile
112         ntot = min(ntot, ncnt(n))
113     enddo
114     print *, 'Number of samples = ', ntot
115
116  c    averaging stuff
117
118  c    istart = ntot-irev
119     istart = ntot-1
120     isamp = 0
121     mavg(:) = 0.0
122     ptavg(:) = 0.0
123     ttavg(:) = 0.0
124
125
126  c    read files
127     allocate(TT(ntot, nfile), PT(ntot, nfile), TS(ntot, nfile))
128     allocate(PS(ntot, nfile), RHO(ntot, nfile), U(ntot, nfile))
129     allocate(V(ntot, nfile), W(ntot, nfile), MASS(ntot, nfile))

```

```

130 allocate(MACH(ntot,nfile),ITER(ntot),TIME(ntot),AREA(ntot,nfile))
131 allocate(MECORR(ntot))
132 do n=1,nfile
133     open(1,file=filename(n),form='formatted')
134 c     skip header lines
135     do m=1,hlines
136         read(1,*) txt
137     enddo
138     do m=1,ntot
139         read(1,*) ITER(m),TIME(m),Area(m,n),MASS(m,n),rhoa,vxa,vya,vza,
140 &         pa,ta,pta,tta,vna,
141 &         RHO(m,n),U(m,n),V(m,n),W(m,n),
142 &         PS(m,n),TS(m,n),PT(m,n),TT(m,n)
143         cc = sqrt(GAMMA*PS(m,n)/RHO(m,n))
144         vv = sqrt(U(m,n)**2+V(m,n)**2+W(m,n)**2)
145         MACH(m,n) = vv/cc
146         MASS(m,n) = abs(MASS(m,n))
147         MASS(m,n) = MASS(m,n)*32.17*12.0*nblades(n)
148 c     averaging
149         if( m.gt.istart ) then
150             if( n.eq.1 ) isamp = isamp+1
151             mavg(n) = mavg(n)+MASS(m,n)
152             ptavg(n) = ptavg(n)+PT(m,n)
153             ttavg(n) = ttavg(n)+TT(m,n)
154         endif
155     enddo
156     close(1)
157 enddo
158
159 c     output some easy stuff
160     open(1,file='rho.out',form='formatted')
161     do m=2,ntot
162         write(1,100) ITER(m),TIME(m),(RHO(m,n),n=1,nfile)
163     enddo
164     close(1)
165     open(1,file='ps.out',form='formatted')

```

```

166     do m=2,ntot
167         write(1,100) ITER(m),TIME(m),(PS(m,n),n=1,nfile)
168     enddo
169     close(1)
170     open(1,file='ps.out',form='formatted')
171     do m=2,ntot
172         write(1,100) ITER(m),TIME(m),(PS(m,n),n=1,nfile)
173     enddo
174     close(1)
175     open(1,file='ts.out',form='formatted')
176     do m=2,ntot
177         write(1,100) ITER(m),TIME(m),(TS(m,n),n=1,nfile)
178     enddo
179     close(1)
180     open(1,file='tt.out',form='formatted')
181     do m=2,ntot
182         write(1,100) ITER(m),TIME(m),(TT(m,n),n=1,nfile)
183     enddo
184     close(1)
185     open(1,file='pt.out',form='formatted')
186     do m=2,ntot
187         write(1,100) ITER(m),TIME(m),(PT(m,n),n=1,nfile)
188     enddo
189     close(1)
190     open(1,file='u.out',form='formatted')
191     do m=2,ntot
192         write(1,100) ITER(m),TIME(m),(u(m,n),n=1,nfile)
193     enddo
194     close(1)
195     open(1,file='v.out',form='formatted')
196     do m=2,ntot
197         write(1,100) ITER(m),TIME(m),(v(m,n),n=1,nfile)
198     enddo
199     close(1)
200     open(1,file='w.out',form='formatted')
201     do m=2,ntot

```

```

202     write(1,100) ITER(m), TIME(m), (w(m,n), n=1, nfile)
203 enddo
204 close(1)
205
206 C   output some stuff
207     open(1, file='mach.out', form='formatted')
208     open(2, file='merr.out', form='formatted')
209     open(3, file='ratio.out', form='formatted')
210     open(8, file='pterr.out', form='formatted')
211     open(9, file='tterr.out', form='formatted')
212     allocate(merr(nfile))
213     do m=2, ntot
214         write(1,100) ITER(m), TIME(m), (MACH(m,n), n=1, nfile)
215     c   get massflow error
216         do n=1, nfile
217             merr(n) = 100.0*(mass(m,n)-mtarget)/mtarget
218         enddo
219         write(2,100) ITER(m), TIME(m), (merr(n), n=1, nfile)
220         err1 = 100.0*(pt(m,1)-pt(m,2))/max(pt(m,1), 1.0e-12)
221         write(8,100) ITER(m), TIME(m), err1
222         err1 = 100.0*(tt(m,1)-tt(m,2))/max(tt(m,1), 1.0e-12)
223         write(9,100) ITER(m), TIME(m), err1
224
225     c   write(6,100) ITER(m), TIME(m), mass(m,1), mass(m,6),
226     c   &           ratt, ratp, err1, ps(m,6)
227         ratt = TT(m,4)/TT(m,1)
228         ratp = PT(m,4)/PT(m,1)
229         MECORR(m) = mass(m,4)*sqrt(TT(m,4)/TTSTD)/(PT(m,4)/PTSTD)
230         write(6,100) ITER(m), (mass(m,n), n=1, nfile),
231     &           (merr(n), n=1, nfile), ratt, ratp
232         write(3,100) ITER(m), TIME(m), ratt, ratp
233     enddo
234     open(1, file='mass.out', form='formatted')
235     do m=2, ntot
236         write(1,100) ITER(m), TIME(m), (MASS(m,n), n=1, nfile), MECORR(m)
237     enddo

```

```

238     close(1)
239
240 c   averaging stuff out
241
242     print *, ' '
243     print *, 'isamp = ', isamp
244     do n=1,nfile
245         mavg(n) = mavg(n)/float(isamp)
246         ptavg(n) = ptavg(n)/float(isamp)/PTINF
247         ttavg(n) = ttavg(n)/float(isamp)/TTINF
248         if( n.eq.3 ) then
249             print *, 'Rotor Inflow      ', mavg(n), ptavg(n), ttavg(n),
250             &          Mach(ntot,n)
251         endif
252         if( n.eq.4 ) then
253             print *, 'Rotor Outflow     ', mavg(n), ptavg(n), ttavg(n),
254             &          Mach(ntot,n)
255         endif
256         if( n.eq.5 ) then
257             print *, 'Stator Inflow      ', mavg(n), ptavg(n), ttavg(n),
258             &          Mach(ntot,n)
259         endif
260         if( n.eq.6 ) then
261             print *, 'Stator Outflow     ', mavg(n), ptavg(n), ttavg(n),
262             &          Mach(ntot,n)
263         endif
264     enddo
265
266     gm1 = gamma-1.0
267     mout = mtarget
268 c   Rotor performance
269     ptra = ptavg(4)/ptavg(1)
270     ttra = ttavg(4)/ttavg(1)
271     effa = (ptra**(gm1/gamma)-1.0)/(ttra-1.0)
272     print *, ' '
273     print *, 'Rotor Performance'

```

```

274     write(6,600)mout,ptr,ttra,effa
275 c     Stator performance
276     ptr = ptavg(6)/ptavg(5)
277     ttra = ttavg(6)/ttavg(5)
278     print *,' '
279     print *,'Stator Performance'
280     write(6,600)mout,ptr,ttra
281 c     Stage performance
282     ptr = ptavg(6)/ptavg(1)
283     ttra = ttavg(6)/ttavg(1)
284     effa = (ptr**(gml/gamma)-1.0)/(ttra-1.0)
285     print *,' '
286     print *,'Stage Performance'
287     write(6,600)mout,ptr,ttra,effa
288
289 600 format(8(1x,1pe14.7))
290
291     stop
292     end

```

## C.2.2 Gnuplot Script

```

1 # set terminal postscript color solid "Helvetica" 24
2 # set output "mass.ps"
3 set title 'Rotor4'
4 set xlabel 'Iterations'
5 set grid
6 set pointsize 3
7 #set key 2100,0.026
8 #set nokey
9 set ylabel 'Massflow (lbm/s)'
10 p [][] \
11     'mass.out' using 1:3 title 'Spinner Inflow' with lines lw 3, \
12     'mass.out' using 1:4 title 'Spinner Outflow' with lines lw 3, \

```

```

13 'mass.out' using 1:5 title 'Rotor Inflow' with lines lw 3, \
14 'mass.out' using 1:6 title 'Rotor Outflow' with lines lw 3, \
15 'mass.out' using 1:7 title 'Stator Inflow' with lines lw 3, \
16 'mass.out' using 1:8 title 'Stator Outflow' with lines lw 3, \
17 'mass.out' using 1:9 title 'Nozzle Inflow' with lines lw 3, \
18 'mass.out' using 1:10 title 'Nozzle Outflow' with lines lw 3
19 pause -1
20 set ylabel 'Massflow error (%)'
21 p [][-4:4] \
22 'merr.out' using 1:3 title 'Spinner Inflow' with lines lw 3, \
23 'merr.out' using 1:4 title 'Spinner Outflow' with lines lw 3, \
24 'merr.out' using 1:5 title 'Rotor Inflow' with lines lw 3, \
25 'merr.out' using 1:6 title 'Rotor Outflow' with lines lw 3, \
26 'merr.out' using 1:7 title 'Stator Inflow' with lines lw 3, \
27 'merr.out' using 1:8 title 'Stator Outflow' with lines lw 3, \
28 'merr.out' using 1:9 title 'Nozzle Inflow' with lines lw 3, \
29 'merr.out' using 1:10 title 'Nozzle Outflow' with lines lw 3
30 pause -1
31 set key bottom
32 set ylabel 'Total Pressure'
33 p [[]] \
34 'pt.out' using 1:($3/14.7) title 'Spinner Inflow' with lines lw 3, \
35 'pt.out' using 1:($4/14.7) title 'Spinner Outflow' with lines lw 3, \
36 'pt.out' using 1:($5/14.7) title 'Rotor Inflow' with lines lw 3, \
37 'pt.out' using 1:($6/14.7) title 'Rotor Outflow' with lines lw 3, \
38 'pt.out' using 1:($7/14.7) title 'Stator Inflow' with lines lw 3, \
39 'pt.out' using 1:($8/14.7) title 'Stator Outflow' with lines lw 3, \
40 'pt.out' using 1:($9/14.7) title 'Nozzle Inflow' with lines lw 3, \
41 'pt.out' using 1:($10/14.7) title 'Nozzle Outflow' with lines lw 3
42 pause -1
43 set ylabel 'Total Temperature'
44 p [[]] \
45 'tt.out' using 1:($3/518.0) title 'Spinner Inflow' with lines lw 3, \
46 'tt.out' using 1:($4/518.0) title 'Spinner Outflow' with lines lw ...
    3, \
47 'tt.out' using 1:($5/518.0) title 'Rotor Inflow' with lines lw 3, \

```

```

48 'tt.out' using 1:($6/518.0) title 'Rotor Outflow' with lines lw 3, \
49 'tt.out' using 1:($7/518.0) title 'Stator Inflow' with lines lw 3, \
50 'tt.out' using 1:($8/518.0) title 'Stator Outflow' with lines lw 3, \
51 'tt.out' using 1:($9/518.0) title 'Nozzle Inflow' with lines lw 3, \
52 'tt.out' using 1:($10/518.0) title 'Nozzle Outflow' with lines lw 3
53 pause -1
54 set ylabel 'Mach Number'
55 p [][] \
56 'mach.out' using 1:3 title 'Spinner Inflow' with lines lw 3, \
57 'mach.out' using 1:4 title 'Spinner Outflow' with lines lw 3, \
58 'mach.out' using 1:5 title 'Rotor Inflow' with lines lw 3, \
59 'mach.out' using 1:6 title 'Rotor Outflow' with lines lw 3, \
60 'mach.out' using 1:7 title 'Stator Inflow' with lines lw 3, \
61 'mach.out' using 1:8 title 'Stator Outflow' with lines lw 3, \
62 'mach.out' using 1:9 title 'Nozzle Inflow' with lines lw 3, \
63 'mach.out' using 1:10 title 'Nozzle Outflow' with lines lw 3
64 pause -1
65 set ylabel 'Static Pressure'
66 p [][] \
67 'ps.out' using 1:3 title 'Spinner Inflow' with lines lw 3, \
68 'ps.out' using 1:4 title 'Spinner Outflow' with lines lw 3, \
69 'ps.out' using 1:5 title 'Rotor Inflow' with lines lw 3, \
70 'ps.out' using 1:6 title 'Rotor Outflow' with lines lw 3, \
71 'ps.out' using 1:7 title 'Stator Inflow' with lines lw 3, \
72 'ps.out' using 1:8 title 'Stator Outflow' with lines lw 3, \
73 'ps.out' using 1:9 title 'Nozzle Inflow' with lines lw 3, \
74 'ps.out' using 1:10 title 'Nozzle Outflow' with lines lw 3
75 pause -1
76 set ylabel 'Angle'
77 p [][] \
78 '/path/to/run/directory/tracking/rotor/stage4.mmotion' using 1:15 ...
    title 'Rotor' with lines lw 3
79 pause -1

```



### C.3 User-defined Equations in FieldView

Within FieldView it was often necessary to create user-defined functions to aid in post-processing. A .frm script was used to define any functions not defined by default in FieldView, such as the  $\lambda_2$  variable shown below.

```
1 formula_restart_version: 1
2 theta
3 PI/2-"atan(Y/Z) "
4 Vel
5 "VelocityVec [rel] "
6 U_x
7 "VelocityVec [axial][rel] "
8 Vaxial_ft
9 "VelocityVec [axial][rel]"/12
10 U_y
11 cos("theta")*"VelocityVec [radial][rel]"-sin("theta")*"VelocityVec ...
    [tangential][rel] "
12 U_z
13 sin("theta")*"VelocityVec [radial][rel]" +cos("theta")*"VelocityVec ...
    [tangential][rel] "
14 VecZ(curl('Vel'))
15 VecZ(curl("Vel"))
16 vort
17 curl("Vel")
18 vort_x
19 VecX("vort")
20 vort_y
21 VecY("vort")
22 vort_z
23 VecZ("vort")
24 mag('vort')
25 mag("vort")
26 L1
27 sqrt("vort_x"*"vort_x"+"vort_y"*"vort_y"+"vort_z"*"vort_z")
```

```

28 gUx^2
29 grad("U_x") dot grad("U_x")
30 gUy^2
31 grad("U_y") dot grad("U_y")
32 gUz^2
33 grad("U_z") dot grad("U_z")
34 lambda2
35 sqrt("gUx^2"+"gUy^2"+"gUz^2")-"mag('vort') "

```

## C.4 Generating Distortion BC for Kestrel

The following Matlab script was used to generate and plot the distortion boundary condition that was used for this research.

```

1 clear;clc;close all;
2
3 res = [50,180]; %resolution of profile [radial, theta]
4
5 Rmax = 8.5; % Max radius, in user-specified units
6 Rmin = 0;
7 p0 = 14.69; % Inlet total pressure
8 % p0=0; %Use this to be standard atmospheric ...
   pressure in the undistorted region
9
10 BL = 0.01*39.3701; % Boundary layer thickness, in ...
   user-specified units
11 TL = 0.03*39.3701; % Transition layer thickness, ...
   in user-specified units
12 rmax = Rmax*0.9; % Max radius for the 90 degree sector
13 rmin = Rmin+0.2*Rmax; % Min radius for the 90 degree sector
14 modecombo = [0,90]; % Modes to combine to form the new ...
   pattern (90 for 90-degree sector)
15 pDist = 15; % Percent distortion
16 A = (p0)*pDist/100; % Total pressure distortion amplitude

```

```

17 %BLA = 16000; % Boundary layer amplitude (Pa)
18 BLA = 2.31966; % Boundary layer amplitude (psi)
19
20 r = linspace(Rmin,Rmax,res(1));
21 t = linspace(-pi,pi,res(2)+1);
22 [theta,radius] = meshgrid(t,r);
23
24 P0 = zeros(res(1),res(2)+1);
25 if ismember(0,modecombo)
26     for i = 1:res(1)
27         if r(i)>Rmax-BL
28             a=-BLA/BL^2;
29             b=2*BLA/BL^2*(Rmax-BL);
30             c=p0-BLA-BLA/BL^2*Rmax^2+2*BLA/BL*Rmax;
31             P0(i,:)=a*r(i)^2+b*r(i)+c;
32         elseif r(i)<Rmin+BL && Rmin≠0
33             a=-BLA/BL^2;
34             b=2*BLA/BL^2*(BL+Rmin);
35             c=p0-BLA-BLA/BL^2*Rmin^2-2*BLA/BL*Rmin;
36             P0(i,:)=a*r(i)^2+b*r(i)+c;
37         else
38             P0(i,:) = p0;
39         end
40     end
41 end
42
43 P90 = zeros(res(1),res(2)+1);
44 if ismember(90,modecombo)
45     for i = 1:res(1)
46         for j = 1:res(2)+1
47             if t(j)>pi/4+TL/r(i) && t(j)<3*pi/4-TL/r(i) && r(i)>rmin+TL ...
48                 && r(i)<rmax-TL
49                 P90(i,j) = -A;
50             elseif t(j)>pi/4-TL/r(i) && t(j)≤pi/4+TL/r(i) && ...
51                 r(i)>rmin+TL && r(i)<rmax-TL
52                 P90(i,j)=-A/2*(sin(pi*r(i)/2/TL*(t(j)-pi/4))+1);

```

```

51     elseif t(j) ≥ 3*pi/4-TL/r(i) && t(j) < 3*pi/4+TL/r(i) && ...
        r(i) > rmin+TL && r(i) < rmax-TL
52         P90(i,j) = -A/2 * (-sin(pi*r(i)/2/TL*(t(j)-3*pi/4))+1);
53     elseif t(j) > pi/4+TL/r(i) && t(j) < 3*pi/4-TL/r(i) && ...
        r(i) > rmin-TL && r(i) ≤ rmin+TL
54         P90(i,j) = -A/2 * sin(pi/2/TL*(r(i)-(rmin)))-A/2;
55     elseif t(j) > pi/4+TL/r(i) && t(j) < 3*pi/4-TL/r(i) && ...
        r(i) > rmax-TL && r(i) ≤ rmax+TL
56         P90(i,j) = A/2 * sin(pi/2/TL*(r(i)-(rmax)))-A/2;
57     elseif t(j) > pi/4-TL/r(i) && t(j) ≤ pi/4+TL/r(i) && ...
        r(i) > rmin-TL && r(i) ≤ rmin+TL
58         P90(i,j) = -A/4 * ((-sin(pi*r(i)/2/TL*(t(j)-pi/4))-1) * ...
            (-sin(pi/2/TL*(r(i)-(rmin)))-1));
59     elseif t(j) > pi/4-TL/r(i) && t(j) ≤ pi/4+TL/r(i) && ...
        r(i) > rmax-TL && r(i) ≤ rmax+TL
60         P90(i,j) = -A/4 * ((-sin(pi*r(i)/2/TL*(t(j)-pi/4))-1) * ...
            (sin(pi/2/TL*(r(i)-(rmax)))-1));
61     elseif t(j) ≥ 3*pi/4-TL/r(i) && t(j) < 3*pi/4+TL/r(i) && ...
        r(i) > rmin-TL && r(i) ≤ rmin+TL
62         P90(i,j) = -A/4 * ((sin(pi*r(i)/2/TL*(t(j)-3*pi/4))-1) * ...
            (-sin(pi/2/TL*(r(i)-(rmin)))-1));
63     elseif t(j) ≥ 3*pi/4-TL/r(i) && t(j) < 3*pi/4+TL/r(i) && ...
        r(i) > rmax-TL && r(i) ≤ rmax+TL
64         P90(i,j) = -A/4 * ((sin(pi*r(i)/2/TL*(t(j)-3*pi/4))-1) * ...
            (sin(pi/2/TL*(r(i)-(rmax)))-1));
65     end
66     end
67     end
68 end
69
70 P = P0+P90;
71 Pabs = P;
72 Pt = reshape(P(:,1:end-1)', res(1)*res(2), 1);
73 R = reshape(radius(:,1:end-1)', res(1)*res(2), 1);
74 T = reshape(theta(:,1:end-1)', res(1)*res(2), 1);
75

```

```

76 % Plot distortion profile
77 [X,Y] = pol2cart(theta,radius);
78 % colors = doubleRainbow();
79 figure('Position', [50, 50, 900, 900]);
80 [h,a]=contourf(X,Y,Pabs,127);
81 colormap(jet);
82 set(a,'LineStyle','none');
83 b=colorbar('Location','southoutside');
84 axis square;
85 axis off;
86 xlabel(b,'Absolute Total Pressure (psi)','FontSize',25);
87 title('90^o Sector','FontSize',30);
88 set(gca,'FontSize',20);
89 hold off
90
91 [X,Y] = pol2cart(T,R);
92 Z = zeros(res(1)*res(2),1);
93
94 % Write to a csv file formatted for use in STAR-CCM+
95 ncomb = size(modecombo,2);
96 filename = 'AIP';
97 for i=1:ncomb
98     filename = strcat(filename,'_',num2str(modecombo(i)));
99 end
100 filename = strcat(filename,'.csv');
101 delete (filename);
102 fid = fopen(filename,'w');
103 fprintf(fid,'"X (in)","Y (in)","Z (in)","Total Pressure (psi)"\n');
104 fclose(fid);
105 if Rmin==0
106     m = [X(res(2):end),Y(res(2):end),Z(res(2):end),Pt(res(2):end)];
107 else
108     m = [X,Y,Z,Pt];
109 end
110 dlmwrite(filename,m,'-append','precision',8);
111

```

```

112 % Write to a file formatted for use in Kestrel
113 D = csvread('AIP_0_90.csv',1,0);
114
115 x = D(:,1);
116 y = D(:,2);
117 z = D(:,4);
118
119 xlin = linspace(min(x), max(x),51);
120 ylin = linspace(min(y), max(y),51);
121 [X,Y] = meshgrid(xlin,ylin);
122
123 F = scatteredInterpolant(x,y,z);
124 Z = F(X,Y);
125
126 % Plot Kestrel distortion profile
127 figure('Position', [50, 50, 900, 900]);
128 [h,a]=contourf(X,Y,Z,127);
129 colormap(jet);
130 set(a,'LineStyle','none');
131 b=colorbar('Location','southoutside');
132 axis square;
133 axis off;
134 caxis([min(z) max(z)])
135 xlabel(b,'Absolute Total Pressure (psi)','FontSize',25);
136 title('90^o Sector','FontSize',30);
137 set(gca,'FontSize',20);
138 hold off
139
140 x_aip = -13.4723;
141
142 X_AIP = ones(867,3)*x_aip;
143 Y_AIP = reshape(X,867,3); %X
144 Z_AIP = reshape(Y,867,3); %Y
145 Pt = reshape(Z,867,3); %Z
146 x_vel = ones(867,3);
147 y_vel = zeros(867,3);

```

```
148 z_vel = zeros(867,3);
149 Tt = ones(867,3)*518.7;
150 Turb1 = ones(867,3)*-1;
151 Turb2 = ones(867,3)*-1;
152
153 facebc = vertcat([51 51 ...
    1],X_AIP,Y_AIP,Z_AIP,Pt,x_vel,y_vel,z_vel,Tt,Turb1,Turb2);
154
155 dlmwrite('facebc.input',facebc,'delimiter','\t','precision','%.6E');
```