

NONLINEAR GROWTH AND BREAKDOWN
OF THE HYPERSONIC CROSSFLOW INSTABILITY

A Dissertation

Submitted to the Faculty

of

Purdue University

by

Joshua B. Edelman

In Partial Fulfillment of the

Requirements for the Degree

of

Doctor of Philosophy

August 2019

Purdue University

West Lafayette, Indiana

THE PURDUE UNIVERSITY GRADUATE SCHOOL
STATEMENT OF DISSERTATION APPROVAL

Dr. Steven P. Schneider, Chair

School of Aeronautics and Astronautics

Dr. Gregory A. Blaisdell

School of Aeronautics and Astronautics

Dr. Jonathan Poggie

School of Aeronautics and Astronautics

Dr. Matthew P. Borg

Air Force Research Laboratory

Approved by:

Dr. Weinong Wayne Chen

Head of the School Graduate Program

For Mom and Dad, Granddad, and Uncle David

It is far easier to start something than it is to finish it.

Amelia Earhart

ACKNOWLEDGMENTS

This dissertation is the culmination of half a decade of work, and though it is a cliché to say so, it would not be possible without the encouragement and aid of several others. I would of course like to thank my committee, Drs. Schneider, Borg, Poggie, and Blaisdell, for their advice and support. I would especially like to thank Matt Borg, who took the time out of his tremendously busy schedule to have several one-on-one technical discussions with me, and Dr. Schneider, who provided invaluable advice throughout my work.

I could not have made it all these years in West Lafayette without the support of the rest of the lab, past and present. Thanks to all of you for keeping me happy and (mostly) sane. And a special shout-out goes to Kat Casper for being an amazing boss, mentor, and friend.

This dissertation would be entirely theoretical without the help of the three AAE machinists, Jim Younts, Robin Snodgrass, and Jerry Hahn. The tunnel would not operate without their hard work, and the lab is eternally indebted to them.

Thanks are also due to the AFOSR for funding via contract FA9550-17-1-0419, and indirectly to every tax-paying viewer like you, for funding me through the NDSEG fellowship (administered by ASEE).

Neal Bitter at Sandia National Laboratories provided computations for this work, and I am very grateful for his help. Sandia National Laboratories is a multi-mission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC., a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

Finally, thanks to my family. You know why.

TABLE OF CONTENTS

	Page
LIST OF TABLES	viii
LIST OF FIGURES	ix
NOMENCLATURE	xviii
ABSTRACT	xxi
1 INTRODUCTION	1
1.1 Transition on a Cone at Angle of Attack	2
1.2 Secondary Instability of Low-Speed Crossflow Waves	6
1.3 High-Speed Crossflow Boundary Layers	11
1.3.1 Recent and Ongoing Crossflow Work	17
1.3.2 Previous Work by the Author and Current Objectives	19
1.4 Approaches to Generating Stationary Vortices	20
1.5 Heat Flux from Infrared Images	21
2 FACILITY AND MODELS	24
2.1 Boeing/AFOSR Mach-6 Quiet Tunnel	24
2.1.1 Tunnel Pressure Measurement	25
2.1.2 Run Characteristics	26
2.2 Modular Cone	27
2.2.1 Model Steps, Gaps, and Flaws	31
2.2.2 Roughness Elements	34
2.2.3 RIM Roughness Properties	36
2.2.4 Sandblasted Roughness Properties	39
3 INSTRUMENTATION AND PROCESSING	42
3.1 PCB Piezotronics Pressure Sensors	42
3.2 Kulite Pressure Sensors	43
3.3 Oscilloscopes	43
3.4 Calculation of Flow Conditions	44
3.5 Computational Fluid Dynamics Model	44
3.6 Infrared Thermography	46
3.6.1 Algorithms for Inferring Heat-Transfer from Temperature Images	48
3.6.2 Heat-Transfer Reduction Process	52
3.7 Pressure Data Reduction	55
3.7.1 Spectral Quantities and Fluctuation Amplitude	55
3.7.2 Calculation of the Phase Velocity	56
3.7.3 Quantifying Nonlinear Interaction	57

	Page
3.7.4 Turbulence Metric	60
4 DEVELOPMENT OF CONTROLLED ROUGHNESS	65
4.1 Demonstrating Control	66
4.1.1 200 μm Elements	66
4.1.2 127 μm Elements	71
4.2 Roughness Reynolds Number	77
4.3 Roughness Spacing	84
4.4 Direct Metal Laser Sintering Roughness Inserts	85
4.5 Summary of RIM Development	85
5 MEASUREMENTS USING DISCRETE ROUGHNESS ELEMENTS	89
5.1 Repeatability	89
5.1.1 Primary Instability Repeatability	89
5.1.2 Secondary Instability Repeatability	92
5.2 Primary Instability Measurements	99
5.2.1 Stationary Crossflow Vortices	99
5.2.2 Traveling Crossflow Vortices	110
5.3 Secondary Instability Measurements	112
5.3.1 Peak Frequencies	113
5.3.2 Amplitude Development	120
5.3.3 Nonlinear Interactions	127
5.4 Measurements After a Small Leeward Shift of the Roughness	133
5.5 Summary of RIM Measurements	133
6 MEASUREMENTS USING SMALL-AMPLITUDE DISTRIBUTED ROUGHNESS	138
6.1 Repeatability	138
6.2 Primary Instability Measurements	149
6.2.1 Stationary Vortices	149
6.2.2 Traveling Vortices	155
6.3 Secondary Instability Measurements	162
6.3.1 Nonlinear Interactions with Traveling Crossflow	171
6.4 Summary of Measurements with Distributed Roughness	172
7 SUMMARY AND CONCLUSIONS	178
7.1 Open Questions and Future Work	180
REFERENCES	182
A SELECTED RUN CONDITIONS	189
B PCB REPEATABILITY SPECTRA	195
C SUPPLEMENTARY MEASUREMENTS USING SMALL DISCRETE ROUGHNESS	207
D THEORY OF THE IR PROCESSING TECHNIQUES	210
D.1 Direct Method	210
D.2 Indirect Method	214

	Page
D.2.1 A More Efficient Algorithm	217
D.3 Fourier-Based Method	220
D.4 Two-Step Method	226
E HEAT TRANSFER CODES	227
E.1 Heat Transfer Calculation	227
E.2 Image Processing	262
F BICOHERENCE CODE AND VALIDATION	264
F.1 Code Listing	264
F.2 Code Validation	270
F.3 Performance of Bicoherence on Hidden Signals	270
F.3.1 Single Harmonic in the Presence of Noise	271
F.3.2 Wide-Band Quadratic Interaction With Noise	273
G NONLINEAR PCB RESPONSE	275
H ENGINEERING DRAWINGS	280
I BAM6QT INFRARED MEASUREMENT SYSTEM HANDBOOK	298

LIST OF TABLES

Table	Page
2.1 Summary of Contraction Kulite calibrations over several years.	26
2.2 PCB sensor positions for the Modular Cone Sensor Frustum.	31
2.3 Positions of the Kulites in the Modular Cone.	32
2.4 Wind ray Re_{kk} of the steps at the component interfaces, compared with a $152\ \mu\text{m}$ reference (the maximum element height on the RIM insert shown in Figure 2.8). From the CFD computation discussed in Section 3.5, $Re_{\infty} = 11.6 \times 10^6\ \text{m}^{-1}$	34
2.5 Properties of the roughness inserts used in the present experiments.	35
3.1 US3D CFD computation conditions.	45
3.2 Conditions and mesh points for the 0° angle of attack validation case presented in Figure 3.2.	50
3.3 Conditions and mesh points for the 0° angle of attack validation case presented in Figure 3.3.	51
3.4 Conditions and mesh points for the 0° angle of attack validation case presented in Figure 3.4.	52
5.1 Variation in flow conditions for the 37 runs used in Figure 5.1.	90
5.2 Axial stations of the PCB repeatability pairs.	96
A.1 Entry 9.	189
A.2 Entry 13.	189
A.3 Entry 14.	190
A.4 Entry 16.	191
A.5 Entry 17.	193
E.1 Method to trace peaks in heat transfer images.	263

LIST OF FIGURES

Figure	Page
1.1 Diagram of a cone at an angle of attack.	3
1.2 A notional crossflow boundary-layer mass flux profile. Based on a similarity solution, from Craig and Saric [3].	3
1.3 Amplitudes of the stationary crossflow wave disturbance velocities, u' , v' , and w' , digitized and redrawn from Figure 17(b) in Malik et al. [14]. The abscissa, R , is a Reynolds number based on a characteristic length scale defined in [14].	5
1.4 Drawing of the eigenfunction locations of the two types of secondary instabilities. Based on data from Malik et al. [22].	8
1.5 Comparison of a cone with and without added roughness. Adding roughness results in large stationary waves near the sensor, and a measured secondary instability. Reproduced from Henderson [37], with permission.	13
1.6 Power spectra from a single PCB rotated under a crossflow vortex. Note that the roughness elements moves as well, which changes the vortex. Reproduced from Ward [38], with permission.	14
1.7 Isocontours of the normalized pressure fluctuation amplitudes $ \hat{p} $, isolines of basic-state mass flux $\bar{\rho}\bar{u}^*$ (black lines), critical layer (blue line), relative sonic line (green line), and azimuthal angles of $\theta = 115^\circ, 117.5^\circ, 120^\circ, 122.5^\circ$ (the vertical magenta lines) at $x = 405$ mm. (a) Mode I at 15 kHz, (b) Mode II at 125 kHz, (c) Mode III at 375 kHz, and (d) Mode IV at 140 kHz. Reproduced from [40] with permission. Caption from [19]. Flow conditions can be found in [40].	16
2.1 Schematic of the BAM6QT.	24
2.2 Examples of the measured stagnation pressure from the contraction Kulite and the uncalibrated hot-film voltage, from Run 1680.	27
2.3 The Modular Cone System.	28
2.4 A schematic of the Modular Cone.	29
2.5 Magnified view of the sharp nosetip, showing the radius $r = 30.4 \mu\text{m}$. Taken with the Zygo Zegage white-light interferometer.	30
2.6 Illustration of the steps between the Modular Cone components, as well as the nosetip radius. Not to scale.	33

Figure	Page
2.7 A photograph of the two roughness inserts used in the present experiments: Sandblasted (left) and RIM (right).	36
2.8 Stitched Zygo measurements of all 12 elements on the RIM roughness. A contour map of the elements, with the cone axis pointed down (top), and a slice along the $x = 50.8$ mm plane (bottom). The red dashed line is the nominal roughness height, $127 \mu\text{m}$ (0.005 inch).	37
2.9 Comparison of measured height across a single roughness element using the Zygo optical profilometer and a confocal microscope.	38
2.10 The discrete Fourier amplitudes of the roughness at wavenumbers up to 200 per circumference. The nominal roughness height is $127 \mu\text{m}$	39
2.11 Roughness height maps for the nominally smooth and sandblasted inserts. Note the color scales are different.	40
2.12 Histograms for the nominally smooth and Sandblasted roughness heights. Note that the smooth insert has a lower RMS height, as expected, but it has a fat left tail due to the machining grooves.	41
2.13 Probability density function of the Sandblasted roughness heights, shifted to have a mean of zero. The PDF is well modeled by a normal distribution with $\sigma = 3.87 \mu\text{m}$	41
3.1 Contour of the crossplane density gradient at $x = 300$ mm from the Navier-Stokes solution. This view highlights the rapid azimuthal change in boundary-layer thickness and the mushroom-like upwelling at the lee ray.	46
3.2 Comparison of the three methods against the theoretical solution for heat transfer on a cone at 0° angle of attack. Note that the direct solution (orange) is obscured by the indirect solution (green); they are nearly identical. The computation time for each method is listed in seconds for the author's PC. $\text{Re}_\infty = 8.9 \times 10^6 \text{ m}^{-1}$	50
3.3 Accuracy of the heat-transfer reduction process near the nosetip. The dip in heat transfer at $\text{Re} \approx 0.7 \times 10^6$ is coincident with the interface between the roughness insert and the Upper Frustum, and is likely the result of inaccuracies in image registration. $\text{Re}_\infty = 11.3 \times 10^6 \text{ m}^{-1}$	51
3.4 Comparison between the different 1D heat-transfer reduction methods and the Zaccara results [68] on a cone at angle of attack. Note that the direct and indirect results are nearly identical. $\text{Re}_\infty = 7.96 \times 10^6 \text{ m}^{-1}$	52
3.5 Example computational mesh for IR processing.	53
3.6 Example of the results of the image registration process.	54

Figure	Page
3.7 The orientation of the wave angle coordinate system with respect to the Kulite array used in this work. The outer circles at each sensor location indicate the approximate diameter of the Kulite sensing element. The green lines indicate the propagation direction of the wave crests. Note that typically for crossflow $\Psi < 0$, as the traveling waves move downstream from lee to wind.	57
3.8 An example of a bicoherence calculation to illustrate its features and to orient the reader. The 95% significance level, $b_{95\%}^2$, is provided as the orange line on the colorbar. The maximum bicoherence is given to two decimal places at the right of the colorbar. In every bicoherence plot, the maximum will be located with dashed lines to the top and right sides, which will label the frequency location of the maximum (170 kHz and 160 kHz in this example).	61
3.9 An example entropy calculation for three spectra at different axial stations. The spectra clearly show the growth of an instability at 200 kHz and a nearly turbulent flow at 379 mm. The entropy H correspondingly increases from 0.6 at 303 mm to 0.92 at 379 mm. $Re_\infty = 10.4 \times 10^6 \text{ m}^{-1}$	64
4.1 Full heat transfer maps for the baseline and rotated element cases. $Re_\infty = 11 \times 10^6 \text{ m}^{-1}$. Run conditions are provided in Appendix A.	67
4.2 Computational vortex paths from Moyes et al. [40] superimposed on the experimental streaks in Figure 4.1(a).	68
4.3 The relative downstream shift of the streaks for a unit upstream rotation of the roughness elements.	70
4.4 Comparison of heat transfer slices at 150 mm for the baseline and rotated cases. $Re_\infty = 11 \times 10^6 \text{ m}^{-1}$. The run conditions are provided in Appendix A. Baseline: Run 1403; Baseline, Prev. Entry: Run 1321; Rotated: Run 1404.	72
4.5 A comparison between the baseline and rotated cases at 150 mm. The x-axis has been interpolated to be the estimated azimuthal angle at the roughness axial location, given the mapping provided by TAMU. The rotated case has been artificially shifted windward by 3.6°	73
4.6 Comparison of slices at $x = 120$ mm for the baseline case and a case with the roughness rotated 2.2° leeward.	75
4.7 The heat transfer perturbation at 120 mm interpolated to the roughness position (50.8 mm) plotted over the measured roughness heights. Each element produces a pair of streaks, with the leeward streak at a higher amplitude. The relative amplitudes of the streaks seem to be correlated with the individual element heights.	76
4.8 Heat-transfer images of the aft portion of the cone with different upstream roughness heights. The white line is an average, qualitative transition front, described in the text. $Re_\infty \approx 11.1 \times 10^6 \text{ m}^{-1}$	80

Figure	Page
4.9 Images of the heat flux near the roughness elements (the dark circles at 50.8 mm) with increasing heights k . $Re_\infty = 11.1 \times 10^6 \text{ m}^{-1}$	81
4.10 Images of the heat flux on the forward portion of the cone using roughness elements with heights nominally $k = 406 \mu\text{m}$. Images from three increasing unit Reynolds numbers are shown, illustrating how the turbulent wedge near the wind ray moves forward as the roughness Re_{kk} increases. At the maximum unit Reynolds number, the Re_{kk} of the roughness causing the wedge is $Re_{kk} \approx 6900$	82
4.11 CFD-estimated Re_{kk} along the center of the nominally $127 \mu\text{m}$ RIM roughness elements, $Re_\infty = 11.6 \times 10^6 \text{ m}^{-1}$	83
4.12 Comparison of different wavenumbers of the input roughness on the downstream vortex pattern. For (a)–(c), $Re_\infty = 10.8 \times 10^6 \text{ m}^{-1}$; for (d) $Re_\infty = 9.8 \times 10^6 \text{ m}^{-1}$. (a) Run 1415, $k_{\text{RMS}} \approx 1.3 \mu\text{m}$; (b) Run 1417, $k \approx 254 \mu\text{m}$; (c) Run 1408, $k \approx 127 \mu\text{m}$; (d) Run 936, $k \approx 127 \mu\text{m}$	87
4.13 A height map of two of the test DMLS roughness elements.	88
5.1 Slices of the IR-derived heat transfer at $x = 344 \text{ mm}$ for 37 runs. The azimuthal location of the sensors ranged from 112° to 144° . The gray lines are the individual slices and the blue line is the mean. The upper plot shows the standard deviation as a percentage of the local mean. $Re_\infty = 10.5 \times 10^6 \text{ m}^{-1}$	91
5.2 Comparison of the heat transfer at $x = 344 \text{ mm}$ between Entry 16 and Entry 17. The model was completely uninstalled and disassembled, and the camera was repositioned and recalibrated between these measurements. $Re_\infty = 10.5 \times 10^6 \text{ m}^{-1}$	92
5.3 PSDs of measurements at 122° for three PCBs, several runs apart.	93
5.4 The bicoherence for PCB 8 ($x = 303 \text{ mm}$) at 122° , 40 runs apart. These correspond to the two PSDs for PCB 8 in Figure 5.3. $Re_\infty = 10.5 \times 10^6 \text{ m}^{-1}$	94
5.5 PSDs for measurements at the same azimuthal and axial locations, with different sensors. Note that PCB 21 and 22 exhibit large sensor resonances which obscure the repeatability at higher frequencies. $Re_\infty = 10.5 \times 10^6 \text{ m}^{-1}$	96
5.6 Several PSDs for PCBs 13 and 21 at the same axial and azimuthal stations. The dashed gray line is the estimated frequency of the second mode from the CFD solution. $Re_\infty = 10.5 \times 10^6 \text{ m}^{-1}$. See Appendix A for run numbers and conditions.	97
5.7 Pressure fluctuation amplitudes for the three repeatability pairs. The low and high frequency bands are $50 - f_{\text{CFD}}$ kHz and $f_{\text{CFD}} - 1000$ kHz, respectively, where f_{CFD} is the local second mode frequency estimated using the laminar CFD solution as described in Section 3.5. $Re_\infty = 10.5 \times 10^6 \text{ m}^{-1}$	98
5.8 Surface heating caused by the stationary vortices. $Re_\infty = 11.2 \times 10^6 \text{ m}^{-1}$. Images from three runs, left to right: Run 1603, Run 1602, Run 1601.	101
5.9 Wishbone-type streak merging when using the RIM roughness.	102

Figure	Page
5.10 DNS of flow over isolated roughness in a 3D boundary layer. The main vortex (blue) and the secondary vortex (orange) merge in the similar way to the wishbone merging in Figure 5.9. Reproduced from Figure 12 in Kurz and Kloker [99], with permission. Color not in original.	103
5.11 Wavenumber spectrum normalized by the maximum power. The wavenumber is the number of waves over the entire circumference of the cone, extrapolated from the imaged portion. The green line is the computed wavenumber from Moyes et al. [19] along a particular vortex for an upstream wavenumber of 50 (the present experiments used an upstream wavenumber of 40).	105
5.12 Amplitude growth of the streaks.	107
5.13 Comparison of two different classes of streak growth.	108
5.14 heat flux along Vortex 4 at several different unit Reynolds numbers (during the same run). The insets show more detail in the region of the first peak.	109
5.15 Locations of the four Kulite sensors. Their placement relative to the stationary vortices is visible in the zoomed inset.	111
5.16 Spectral properties of the Kulite measurements. $Re_\infty = 10.5 \times 10^6 \text{ m}^{-1}$	112
5.17 The measurement points for the data in this section overlaid on an IR heat-transfer image. The dots are not the same size as the PCB sensors.	113
5.18 PSDs of the secondary instability for several different azimuthal angles at $x = 347 \text{ mm}$. The open circles mark the approximate peak frequency, while the closed circles are the estimated second-mode frequency. $Re_\infty = 10.5 \times 10^6 \text{ m}^{-1}$	114
5.19 Variation in peak frequency of the measured instabilities at two axial stations. Both U_e and δ were extracted from the Navier-Stokes solution as described in Section 3.5.	116
5.20 Contour plot of the experimental peak frequency over the entire measurement range. The peak frequency decreases for all instabilities in the streamwise direction, as the boundary layer thickens both axially and azimuthally. The flow in the upper-right corner is beginning to break down, so the peak identification is not as reliable. The data in this region have been omitted. $Re_\infty = 10.5 \times 10^6 \text{ m}^{-1}$	117
5.21 The ratio of the measured peak frequency to the estimated second-mode frequency, from the data in Figure 5.20. This ratio accounts for the changing boundary layer thickness, so the spanwise modulation in frequency is more distinct.	118
5.22 Contour plot of the peak frequency from Figure 5.20 overlaid on an IR image. The high-frequency modes are concentrated under the hot streaks, while the low-frequency modes grow in the cold spaces in between.	119
5.23 Pressure fluctuation amplitudes in the low and high frequency bands.	122

Figure	Page	
5.24	Contours of amplitude in the HF band overlaid on an IR image. The lobes of HF growth are almost exactly aligned with the hot streaks in the IR.	123
5.25	Contours of both secondary instability modes superimposed on an IR image of the same flow. LF is the low-frequency mode (Fig. 5.23(a)) and HF is the high-frequency mode (Fig. 5.23(b)).	124
5.26	Individual measurement points following a stationary vortex, corresponding to the spectra in Figure 5.27.	124
5.27	Spectra of the measurements along a single vortex, for both instability modes. The increase in power at low frequencies for the HF mode may be spectral filling, a consequence of turbulent breakdown, or it could represent growth of a separate low frequency mode. The Reynolds numbers are in millions based on the axial distance of the measurement.	125
5.28	Comparison of the amplitude growth in the primary and secondary instabilities. Vortex 4 is physically closest to the high-frequency lobe while Vortex 5 is closest to the low-frequency one. The Shannon entropy H was calculated at three axial stations of interest, marked on the plot: (1) $H_{LF} \approx 70\%$, $H_{HF} \approx 80\%$; (2) $H_{HF} = 99\%$; (3) $H_{LF} = 99\%$. Note that the heat-flux measurements were taken at $Re_\infty = 11.2 \times 10^6 \text{ m}^{-1}$ whereas the pressure fluctuation measurements are from $Re_\infty = 10.5 \times 10^6 \text{ m}^{-1}$. See Figure 5.14 for the effect of small changes in unit Reynolds number.	126
5.29	Harmonic generation in the LF mode. The filled dot on the spectrum is the CFD-estimated second-mode frequency. Run 1613.	128
5.30	Harmonic generation in the HF mode. Run 1618.	129
5.31	Possible nonlinear interaction between the two modes. Run 1640.	130
5.32	Nonlinear interaction between the LF and HF mode. Run 1623.	131
5.33	Nonlinear interaction between unidentified modes. Run 1632.	132
5.34	Unwrapped images of the Stanton number for both roughness configurations.	134
5.35	Detailed view of Figure 5.34, showing the forward movement of the wishbone structures when the roughness is rotated. The numbers indicate the same structure as it moves.	135
5.36	Comparison of the secondary instability amplitudes for the baseline and shifted cases. $Re_\infty = 10.2 \times 10^6 \text{ m}^{-1}$	137
6.1	Azimuthal cross-sections of the heat flux from 15 runs at two different axial stations. Each gray line is a slice from one run; the dark blue line is the mean for all the runs. The upper sub-plot shows the standard deviation at each point as a percentage of the local mean.	141

Figure	Page
6.2 Comparison between the mean data from Entry 16 (Figure 6.1(b)) and data from after a change in the model and camera position (refer to Figure 5.2 for the same comparison with the RIM insert). The stationary vortices generated using the Sandblasted insert are evidently far more sensitive to upstream conditions than those from the RIM insert.	142
6.3 Repeatability of the traveling crossflow wave measurements from the Kulite sensors. The origin of the discrepancies between runs 1711 and 1718 is unclear. The narrow peaks at 225 kHz and 300 kHz are the resonances of the sensors. $Re_\infty = 11 \times 10^6 \text{ m}^{-1}$.	143
6.4 Repeatability of the traveling crossflow and secondary instability measurements for two runs at the same azimuthal angle, recorded by PCB sensors. $\theta = 137^\circ$ and $Re_\infty = 11.1 \times 10^6 \text{ m}^{-1}$.	145
6.5 Repeatability of the bicoherence for PCB 18, from the same data as Figure 6.4.	146
6.6 Repeat spectra from PCBs 13 and 21, from the same azimuthal angles. Recall PCB 21 is offset 6° windward from PCB 13, so the measurements at each azimuthal angle are from different runs. The small peak in the offset ray (orange line) at around 425 kHz is likely a sensor resonance.	147
6.7 Pressure fluctuation amplitudes for the three repeatability pairs. The low and high frequency bands are 25–75 kHz and 80–250 kHz, respectively. $Re_\infty = 11.1 \times 10^6 \text{ m}^{-1}$.	148
6.8 Comparison between heat flux images using the nominally smooth insert and the Sandblasted insert.	149
6.9 Surface heating caused by the stationary vortices. $Re_\infty = 11 \times 10^6 \text{ m}^{-1}$. Images from three runs, left to right: Run 1705, Run 1710, Run 1709.	152
6.10 Wavenumber spectrum normalized by the maximum power. The wavenumber is the number of waves over the entire circumference of the cone, extrapolated from the imaged portion. The green line is the computed wavenumber from Moyes et al. [19] along a particular vortex for an upstream wavenumber of 220 (the present experiments have an unknown upstream wavenumber).	153
6.11 Amplitude growth of the streaks.	154
6.12 Spectral quantities of the Kulite measurements at a low azimuthal angle ($\theta = 90^\circ$, $x = 330 \text{ mm}$, $Re_\infty = 11 \times 10^6 \text{ m}^{-1}$). Run 1714.	156
6.13 Wave properties for traveling crossflow at 90° and $Re_\infty = 11 \times 10^6 \text{ m}^{-1}$. Runs 1714 and 1715 were performed consecutively, with no rotation of the Sensor Frustum. The Ward data is from Reference [38], and is at $\theta = 90^\circ$ and $Re_\infty = 11.5 \times 10^6 \text{ m}^{-1}$.	157
6.14 Spectral quantities of the Kulite measurements at a high azimuthal angle ($\theta = 142^\circ$, $x = 330 \text{ mm}$, $Re_\infty = 11 \times 10^6 \text{ m}^{-1}$). There are two apparent modes of instability. The activity at around 50 kHz is traveling crossflow. The peak at 125 kHz may be a type-I secondary instability. Run 1713.	160

Figure	Page
6.15 Wave properties for the measured instabilities at $\theta = 142^\circ$ and $\text{Re}_\infty = 11 \times 10^6 \text{ m}^{-1}$	161
6.16 Propagation angles of the measured high-frequency instabilities compared to the stationary vortex paths. The line $y = 0$ is centered at $\theta = 142^\circ$. $\Psi_{\text{trav}} = -65^\circ$ and $\Psi_{\text{SI}} = -30^\circ$	161
6.17 Pressure fluctuation amplitudes of the two high-frequency instabilities.	165
6.18 Contours of the instability pressure fluctuation amplitudes superimposed on a heat-transfer image. The image is the average of all 15 runs with the Sandblasted insert in Entry 16.	166
6.19 Discrete sensor locations used to measure the growth of the traveling crossflow and secondary instabilities. The plotted points correspond to the spectra in Figure 6.20.	167
6.20 Spectra along the paths indicated in Figure 6.19. The legend indicates axial Reynolds numbers in millions. Note the increasing peak frequency as the secondary instability grows downstream.	168
6.21 Growth of the secondary instabilities against the Strouhal number f/f_{CFD} , illustrating the difference in behavior between the two roughness cases.	169
6.22 Pressure fluctuation amplitudes compared to neighboring stationary streaks.	170
6.23 Bicoherence showing traveling crossflow wave harmonic generation. Run 1675.	172
6.24 Bicoherence showing traveling crossflow harmonic generation and interaction with a high-frequency instability. Run 1680.	173
6.25 Bicoherence showing harmonic generation within the secondary instability band. Run 1673.	174
6.26 Bicoherence showing low levels of nonlinear interaction between traveling crossflow, the secondary instability, and a third peak of unknown origin. Run 1679.	175
6.27 Comparison of power spectra from the Kulites and similarly situated PCB sensors. The Kulite data is from Entry 17, after the model change, so the repeatability of the stationary vortices is uncertain. $\theta = 142^\circ$, $\text{Re}_\infty = 11 \times 10^6 \text{ m}^{-1}$. PCB data, Run 1680; Kulite data, Run 1713.	176
6.28 Bicoherence from a Kulite at a high azimuthal angle. Several regions of nonlinear interaction are evident, including harmonic generation within the traveling crossflow and secondary instability, and interaction between these two instabilities. Run 1713.	177
B.1 PCBs 13 & 21, set 1.	195
B.2 PCBs 13 & 21, set 2.	196
B.3 PCBs 13 & 21, set 3.	197
B.4 PCBs 13 & 21, set 4.	198

Figure	Page
B.5 PCBs 16 & 22, set 1.	199
B.6 PCBs 16 & 22, set 2.	200
B.7 PCBs 16 & 22, set 3.	201
B.8 PCBs 16 & 22, set 4.	202
B.9 PCBs 19 & 23, set 1.	203
B.10 PCBs 19 & 23, set 2.	204
B.11 PCBs 19 & 23, set 3.	205
B.12 PCBs 19 & 23, set 4.	206
C.1 PSD and bicoherence of data taken with a dimpled Torlon insert, $k \approx 38 \mu\text{m}$. Reproduced from the original data in Reference [21].	208
C.2 PSD and bicoherence of data taken using a RIM insert with nominal height $k = 51 \mu\text{m}$	209
F.1 Power spectrum and bicoherence for the validation case.	271
F.2 Power spectra from the first test case: a single sinusoid and its phase-locked first harmonic, with and without additive noise.	272
F.3 The bicoherence of the first test case with additive noise. Note that the maximum theoretical bicoherence is 1.0 at (100 kHz, 100 kHz).	273
F.4 Power spectra for individual components of a quadratically-nonlinear signal and their sum.	274
F.5 The bicoherence of the wide-band test case accurately captures the nonlinearity in the signal.	274
G.1 The sensor voltage response to the mallet hit.	276
G.2 Spectrum of the PCB's mechanical response.	276
G.3 Bicoherence of the PCB's mechanical response.	277
G.4 Sensor response to an acoustic impulse (sensor in pitot mode in a shock tube, see Ref. [80]). The orange section was used for the following analysis. Amplitude units are arbitrary.	278
G.5 Bicoherence of the PCB's response to the shock.	278
G.6 Bicoherence of the PCB's response to the shock, refined view.	279

NOMENCLATURE

Acronyms

BAM6QT Boeing/AFOSR Mach-6 Quiet Tunnel

CFD Computational Fluid Dynamics

DMLS Direct Metal Laser Sintering

DNS Direct Numerical Simulation

FFT Fast Fourier Transform

HF High-Frequency

IR Infrared

LF Low-Frequency

LST Linear Stability Theory

PEEK Poly-Ether Ether Ketone

PSD Power Spectral Density

RIM Rod Insertion Method

RMS Root Mean Square

SLST Secondary Linear Stability Theory

TSP Temperature Sensitive Paint

Symbols

- α Angle of attack, deg
- δ Boundary-layer thickness, mm
- H Shannon information entropy
- Ψ Traveling wave propagation angle, deg
- Re_∞ Unit Reynolds number, U_∞/ν_∞ , m^{-1}
- Re_{kk} Roughness Reynolds number, $U_k k/\nu_k$
- θ Azimuthal angle from wind ray, deg
- A_0 Disturbance initial amplitude
- b^2 Bicoherence
- c Wave phase speed, $m\ s^{-1}$
- f Frequency, Hz or kHz
- m Azimuthal wavenumber, waves/360°
- p Pressure, Pa or psia
- Q Heat transfer, $W\ m^{-2}$ or $kW\ m^{-2}$
- T Temperature, K
- x Axial distance downstream from nosetip, m or mm
- Re Axial Reynolds number, $U_\infty x/\nu_\infty$
- St Stanton number

Subscripts

- 0 Stagnation quantity

- ∞ Freestream quantity
- e Edge quantity
- i Quantity at the beginning of a run
- w Wall quantity
- CFD Quantity from CFD computation
- exp Experimental quantity

ABSTRACT

Edelman, Joshua B. Ph.D., Purdue University, August 2019. Nonlinear Growth and Breakdown of the Hypersonic Crossflow Instability. Major Professor: Steven P. Schneider.

A sharp, circular 7° half-angle cone was tested in the Boeing/AFOSR Mach-6 Quiet Tunnel at 6° angle of attack, extending several previous experiments on the growth and breakdown of stationary crossflow instabilities in the boundary layer. Measurements were made using infrared imaging and surface pressure sensors. Detailed measurements of the stationary and traveling crossflow vortices, as well as various secondary instability modes, were collected over a large region of the cone.

The Rod Insertion Method (RIM) roughness, first developed for use on a flared cone, was adapted for application to crossflow work. It was demonstrated that the roughness elements were the primary factor responsible for the appearance of the specific pattern of stationary streaks downstream, which are the footprints of the stationary crossflow vortices. In addition, a roughness insert was created with a high RMS level of normally-distributed roughness to excite the naturally most-amplified stationary mode.

The nonlinear breakdown mechanism induced by each type of roughness appears to be different. When using the discrete RIM roughness, the dominant mechanism seems to be the modulated second mode, which is significantly destabilized by the large stationary vortices. This is consistent with recent computations. There is no evidence of the presence of traveling crossflow when using the RIM roughness, though surface measurements cannot provide a complete picture. The modulated second mode shows strong nonlinearity and harmonic development just prior to breakdown. In addition, pairs of hot streaks merge together within a constant azimuthal band, leading to a peak in the heating simultaneously with the peak amplitude of the measured secondary instability. The heating then decays before rising again to turbulent levels. This non-monotonic heating pattern is reminiscent of experiments on a flared cone and earlier computations of crossflow on an elliptic cone.

When using the distributed roughness there are several differences in the nonlinear breakdown behavior. The hot streaks appear to be much more uniform and form at a higher wavenumber, which is expected given computational results. Furthermore, the traveling crossflow waves become very prominent in the surface pressure fluctuations and weakly nonlinear. In addition there appears in the spectra a higher-frequency peak which is hypothesized to be a type-I secondary instability under the upwelling of the stationary vortices. The traveling crossflow and the secondary instability interact nonlinearly prior to breakdown.

1. INTRODUCTION

WHEN moving through a fluid, any vehicle develops a thin boundary layer over which the vehicle's momentum is diffused into the fluid. Whether this boundary layer is laminar or turbulent can have a significant impact on the vehicle's design and performance. A laminar boundary layer has lower skin-friction drag and heating, but is less resistant to separation than a turbulent boundary layer. On hypersonic vehicles, a turbulent boundary layer can have an order-of-magnitude higher heating rates than a laminar one, which drives the design of thermal protection systems. Failure to accurately predict the state of the boundary layer during the design process could lead to an over-designed thermal protection system, which reduces performance, or an under-designed one, which could result in the loss of the vehicle. Accurate prediction or control of the boundary-layer state is also important for the design of hypersonic engine inlets and control surfaces. However, current transition prediction methodologies are mostly empirical, and often do not incorporate all of the relevant mechanisms of transition, so they are difficult to apply outside of the conditions from which they were created. To improve upon the existing empirical correlations, there is a need for a set of robust methods for prediction of boundary-layer transition which are based on a physical understanding of the mechanisms involved [1].

The transition process starts with receptivity, by which disturbances enter the boundary layer. Disturbances take many forms, from those generated on the body (like added vorticity from roughness) to freestream fluctuations (like acoustic noise). The receptivity of the boundary layer to specific disturbances is dependent on nearly every aspect of the vehicle geometry and the surrounding flow. Once a disturbance has entered the boundary layer, it will either amplify or decay as a result of instabilities inherent to the specific boundary layer in question. These instabilities are also determined by the overall environment, and depending on the instability the initial disturbances will manifest in various different forms, from acoustic waves to stationary vortices. The unstable disturbances will continue to amplify downstream, possibly interacting

with one another, until they grow large enough that the flow in the boundary layer begins to break down into turbulence.

As with any problem in fluid mechanics, one can study boundary layer transition via experimental methods. However, no single wind tunnel can simulate all aspects of hypersonic flight. Ground-based experiments to study transition require unique facilities because any disturbances in the freestream of the wind tunnel will not necessarily be representative of those found in flight. If the boundary layer is turbulent on the wall of the diverging wind-tunnel nozzle, the fluctuating displacement thickness results in large acoustic noise in the core of the flow. High levels of noise can result in earlier transition or in some cases change the mechanism responsible for transition [2]. In conventional “noisy” wind tunnels the acoustic fluctuations can be 2% of the mean pitot pressure or greater, which is orders of magnitude larger than what is seen in flight. To study transition, it is often necessary to use “quiet” tunnels which are specifically designed to maintain a laminar boundary layer on the nozzle wall. Quiet wind tunnels have noise levels less than 0.05% of the mean pitot pressure, which is more similar to a flight environment.

Despite their advantages, quiet tunnels are not a complete solution to understanding the transition problem. Other phenomena of hypervelocity flight, like chemistry and non-equilibrium effects, cannot at present be replicated in quiet tunnels, nor can the operational quiet facilities reach flight-like Reynolds numbers. In addition, the only currently operational quiet hypersonic tunnels are all Mach 6, so an understanding of Mach number effects on transition can only be experimentally examined in noisy facilities. It is therefore important that experimental efforts in quiet tunnels are partnered with computational studies and experiments in conventional tunnels to provide a more complete picture of the transition process on a real vehicle.

1.1 Transition on a Cone at Angle of Attack

A cone at an angle of attack in a supersonic freestream is a canonical geometry to study boundary-layer transition in a three-dimensional flowfield. Figure 1.1 provides a diagram of such a geometry, showing the coordinate system used when discussing the flowfield. The axial distance downstream from the nose is x and the azimuthal angle from the wind ray is θ .

The windward side of the cone creates a larger turning angle than the leeward side, so a circumferential pressure gradient is developed, driving the flow from the wind ray to the lee ray. This pressure gradient affects the low-momentum fluid near the wall more than the freestream, creating a component of the boundary-layer velocity that is transverse to the outer flow, called crossflow. The crossflow component of the velocity must be zero at the wall and smoothly become zero in the freestream, and therefore it is inflectional and inviscidly unstable. Figure 1.2 shows a notional crossflow mass-flux profile illustrating the inflectional nature of the crossflow component.

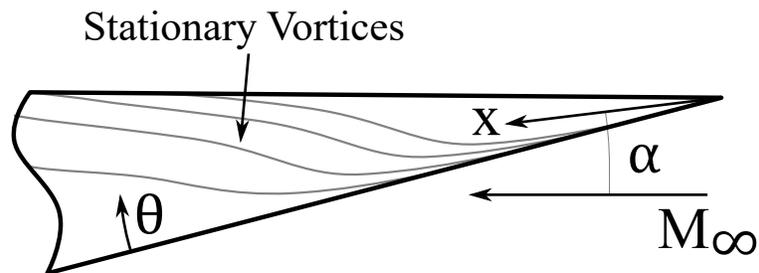


Figure 1.1. Diagram of a cone at an angle of attack.

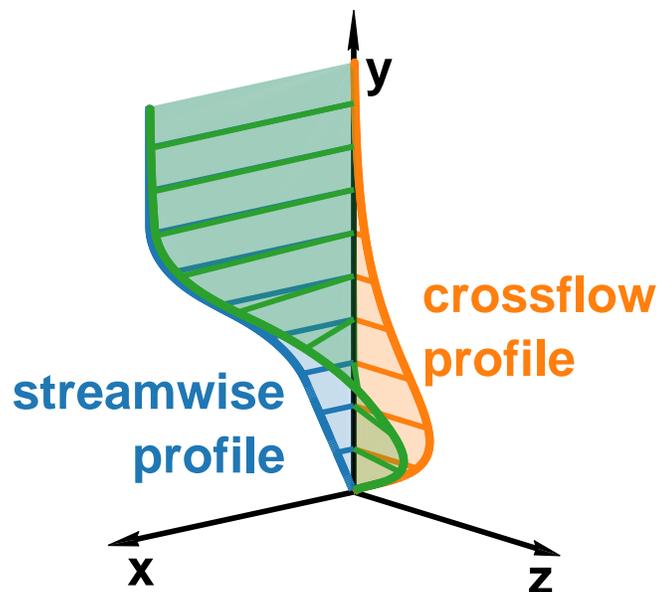


Figure 1.2. A notional crossflow boundary-layer mass flux profile. Based on a similarity solution, from Craig and Saric [3].

The crossflow instability manifests in two forms: stationary and traveling vortices. The stationary vortices are fixed with respect to the surface and nearly aligned with the outer flow. Their axes are thought to pass through the generalized inflection points of the undisturbed boundary layer. These vortices can be readily observed via their modulation of the surface shear and heat transfer, or by measuring the velocity distortion through the boundary layer. Traveling vortices are packets of vortex-like disturbances which move obliquely downstream along the surface, from the lee side to the wind side. The traveling vortices can be easily measured as pressure or velocity fluctuations at a fixed sensor position. Several sensors in an array can be used to determine the vortices' phase speed and propagation angle [4, 5].

Bippes [6] and Saric et al. [7] provide thorough reviews of low-speed crossflow experiments and computations up to the year 2003. At low speeds, traveling waves dominate transition for high freestream turbulence levels, while stationary waves are dominant in the lower-noise environment of low-disturbance tunnels and flight [8]. However, it is not necessarily true that traveling crossflow waves will dominate transition in noisy hypersonic tunnels. Some recent evidence suggests that the traveling waves either do not appear or are buried in high-amplitude, low-frequency noise of unknown origin [9, 10]. However, low-speed computations show that nonlinear interaction between the stationary and traveling crossflow modes can have a significant effect on transition, so understanding the traveling mode is critical to the development of mechanism-based transition prediction methods, even in quiet flow.

Stationary crossflow vortices are sensitive to roughness height and diameter [11], as well as roughness spacing [12, 13]. However, stationary crossflow waves are strongly stabilized by nonlinear growth effects, leading to amplitude saturation once they have grown large enough [14]. Figure 1.3 shows nonlinear computations of the stationary vortex disturbance velocities with initial stationary mode amplitudes $A_{0, \text{stat}} = 0.1\%$ and initial traveling mode amplitudes $A_{0, \text{trav}} = 0.01\%$ [14]. The saturation in the amplitudes is evident at Reynolds numbers above about $R = 500$. The saturation occurs for cases both with and without traveling waves; for instance see Figures 17(a) and 18 in Malik et al [14]. This saturation means that it is impractical to create an amplitude criterion for transition based on the stationary waves alone, as growth of the waves could stop long before transition.

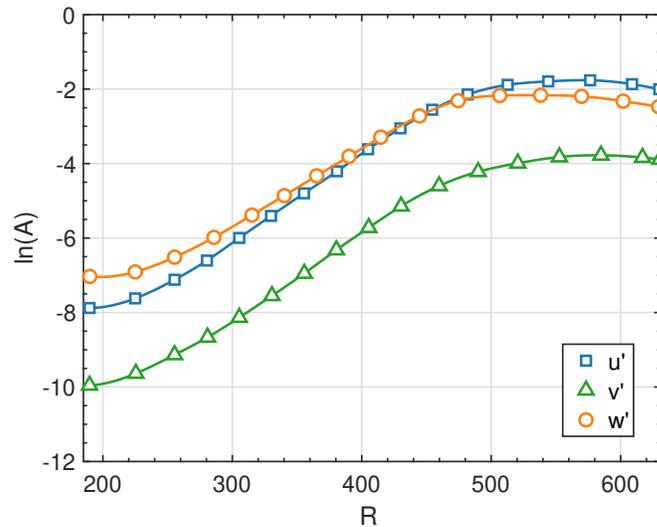


Figure 1.3. Amplitudes of the stationary crossflow wave disturbance velocities, u' , v' , and w' , digitized and redrawn from Figure 17(b) in Malik et al. [14]. The abscissa, R , is a Reynolds number based on a characteristic length scale defined in [14].

At low speeds in stationary-crossflow-dominated transition, the stationary crossflow wave usually breaks down to turbulence by way of a number of secondary instabilities [15, 16]. As the stationary wave grows, it modulates the mean flow of the boundary layer, creating strong shear layers in both the spanwise and wall-normal directions. These inflectional profiles generate two distinct types of secondary instabilities. Type-I instabilities are associated with extrema of transverse gradients of velocity within the stationary wave, and are sometimes referred to as z or θ modes. Type-II instabilities are associated with extrema of wall-normal velocity gradients, and can therefore be called y or r modes.

Both types occur at a frequency an order of magnitude higher than that of the traveling crossflow vortices, but the type-II instabilities are typically higher frequency than the type I [17]. At low speeds, the relative importance of each type of secondary instability to transition is related to the forced wavenumber of the stationary crossflow vortices [16]. The secondary instabilities tend to appear when the stationary wave has quasi-saturated, quickly followed by turbulent breakdown.

Measurements at high speeds have shown that the hypersonic crossflow instability behaves similarly to its low-speed counterpart. Measurements and computations have shown that the

stationary vortices saturate in amplitude, and that secondary instabilities appear with the same mode shapes as the low-speed case [18–20]. However, an additional instability is present in high-speed boundary layers. At hypersonic speeds, the second-mode instability also appears to play a role in boundary-layer transition under crossflow, at least on some geometries. The second mode is like an acoustic wave propagating through the boundary layer between the wall and the relative sonic line. It is usually the dominant transition mechanism in two-dimensional flow-fields like cones at zero degrees angle of attack. Some recent computations [19, 20] and experiments [10, 21] suggest that the second mode is modulated and destabilized by the stationary crossflow vortices, and that this modulated acoustic wave is the ultimate mechanism for nonlinear breakdown to turbulence. Note that for the purposes of this work, the modulated second mode will be referred to as a “secondary instability” akin to the type I and II “true” secondary instabilities. This is because the second mode would not exist in its modulated form without the presence of the primary instability, the stationary crossflow vortices. An important question which is not addressed by the present experiments is whether a change in the nonlinear breakdown mechanism actually affects the location of turbulent onset in a significant way.

The importance of secondary instabilities to transition prediction, whatever their form, is exemplified by Malik et al. [22], who showed that at low speeds an e^N correlation with the amplitudes of the secondary instabilities performed better at predicting transition than using the amplitudes of the stationary vortices. It is therefore important to understand the growth and breakdown of the secondary instabilities at high speeds in the hopes that a mechanism-based prediction method can be created for crossflow in general.

1.2 Secondary Instability of Low-Speed Crossflow Waves

A majority of the experimental and computational work on the secondary instability of the stationary crossflow waves has been at subsonic speeds. Bippes [6] provides an excellent review of low-speed crossflow experiments, including secondary instability measurements. The first experimental evidence of the secondary instability seems to be from Michel et al. [23] in 1984 on a swept wing and Poll [24] in 1985 on a swept cylinder. Note that this is thirty years after Gregory et al. [25] first studied crossflow vortices on a swept wing and rotating disk. Hot-wire

measurements by Michel and Poll showed an instability at a frequency an order of magnitude higher than the measured travelling wave frequency. Poll credited these results to intermittent turbulence.

However, in 1991 Kohama et al. [15] argued that the higher frequency was actually a secondary instability, which has become the consensus. In their article, Kohama et al. provide measurements taken on a swept wing in the Arizona State University (ASU) Unsteady Wind Tunnel. They describe the secondary instability as co-rotating vortices aligned nearly perpendicularly to the stationary crossflow waves, and propagating in the streamwise direction. They write that the breakdown to turbulence of the crossflow vortices is the result of the secondary instability growing, interacting with the primary vortex, and forming multiple other inflectional instabilities. The entire breakdown takes place over a very limited distance on the order of the boundary-layer thickness, and between the middle and the edge of the boundary layer where the secondary instability forms.

The crossflow vortices in the Kohama et al. experiments were naturally excited. To extend these results, Lerche used several vibrating membranes on a swept wing as a disturbance generator [26]. Using this system, he excited a single travelling crossflow mode. He also examined the results when both a travelling crossflow mode and a stationary crossflow mode were excited. In both cases he observed a secondary instability at a frequency an order of magnitude higher than the excitation. The instability occurred at a particular phase between the travelling wave and the excitation signal. This seems to suggest that Lerche's high-frequency measurements were secondary instabilities of the travelling waves, as opposed to secondary instabilities of the stationary waves measured by Kohama et al. [15].

In the mid-1990's, there was a great deal of computational work studying the low-speed crossflow instability and associated secondary instabilities. An analogous secondary instability analysis had already been performed on the Görtler instability, a similar streamwise vortex-based instability [27, 28]. In 1996, Malik et al. [17] computed the secondary instabilities in the 1991 ASU swept wing experiment described by Kohama et al. [15]. They found two modes, which they called mode I and mode II. The mode II frequency was slightly less than double the mode I frequency. The mode I instability grew on the leeward side of the primary stationary wave and is caused by

high spanwise shear. The mode II instability grew over the crest of the wave as a result of high wall-normal shear.

In 1999, Malik et al. [22] published a more extensive secondary instability analysis for a different swept-wing experiment at ASU in 1996 [12]. This computational crossflow study produced a few key results. First, the computations show that in some places as many as seven secondary instability modes exist. For some modes, the growth rate curves have multiple frequency peaks. All these modes begin to grow once the stationary wave has saturated. In general, each of the modes can be classified as a z or y mode (equivalent to the type-I and type-II nomenclature used throughout the present work), based on the dominant energy production mechanism. The z and y designation seems to have replaced the ‘mode I’ and ‘mode II’ names used in the previous paper. A stylized depiction of the locations of the z and y modes on the stationary waves is shown in Figure 1.4. This was drawn based on the results in Figures 7 and 8 of Malik et al. [22].

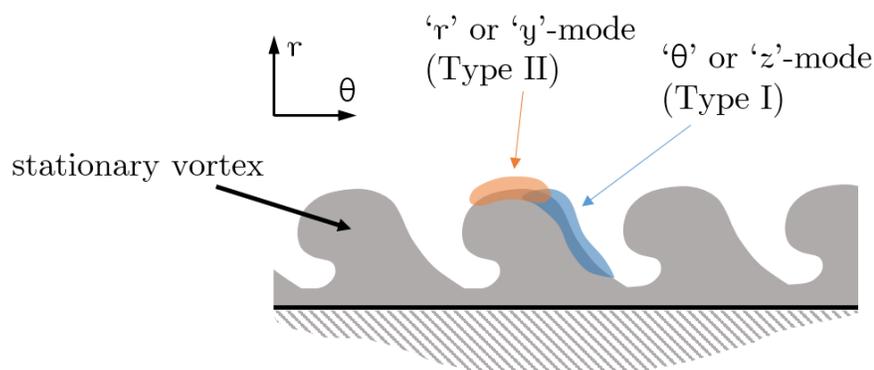


Figure 1.4. Drawing of the eigenfunction locations of the two types of secondary instabilities. Based on data from Malik et al. [22].

In addition, Malik et al. provide an estimate for the secondary instability frequency. They use $f_{est} \approx U_e / \lambda_p$, where U_e is the edge velocity and λ_p is the stationary crossflow wavelength. This assumes that the phase speed of the secondary instability is the same as the edge velocity.

However, the most important result from Malik et al. is that an N-factor transition correlation based on the growth of the secondary instabilities is more successful than a correlation based on the stationary wave amplitude. This is because of the nonlinear amplitude saturation of the

stationary vortices. As discussed previously, the amplitude saturation makes a criterion based on the stationary wave amplitude impractical. The secondary instabilities, however, grow quickly just prior to transition.

Around the same time as the Malik et al. work, White and Saric conducted an extensive hot-wire study of crossflow on a swept wing in the low-disturbance ASU Unsteady Wind Tunnel, which was published in 2005 [16]. Though they credit Malik et al. for the type-I and type-II nomenclature used in the present work, it appears this wording was first used by Koch et al. [29].

White and Saric present measurements at several streamwise locations for a few different roughness configurations. At the farthest upstream locations, no secondary instabilities are detected. There are several peaks in the hot-wire power spectra, but these are explained as traveling crossflow waves (at low frequencies) and a Tollmien-Schlichting-like wave (at a higher frequency).

Farther downstream, several secondary instabilities appear. Type-I modes exist on the shoulder of the stationary vortex. White and Saric also found that multiple frequencies of type-I modes appeared in the same location, but were not harmonics. The lowest-frequency mode, a type I, was found to have the largest amplitude of all the secondary instabilities that were measured.

In several cases White and Saric also measured a type-II mode on the crest of the stationary vortex. The type-II frequency was approximately double the fundamental type-I frequency, as predicted by Malik et al. [17]. They report that the low-amplitude type-II mode is often difficult to measure and can be masked by the growth of a high-frequency harmonic of the type-I mode, which sometimes appears in a spatially overlapping region.

Malik et al. [22] showed that the type-II mode initially has a much higher growth rate than type-I modes. However, experimental evidence [15, 16] shows the type-I modes appear before the type-II modes and are much larger. Note that these results are at slightly different but comparable conditions (Reynolds number, angle of attack, and stationary vortex wavelength). Both White and Saric, and Malik et al. use a receptivity argument to explain this disparity. The experimental results would make sense if the type-I modes had larger initial amplitudes. White and Saric posit that this may be because the freestream turbulence level decreases with increasing frequency.

However, they found that by supercritically forcing the stationary waves (using roughness elements to force a spanwise wavelength longer than the naturally dominant one) the type-I

growth was suppressed because the region of high spanwise shear was reduced. In this case type-II modes were easier to identify and more important in transition.

One of the most recent numerical explorations of the low-speed crossflow secondary instability is by Bonfigli and Kloker in 2007 [30]. They compared the results of Secondary Linear Stability Theory (SLST) with Direct Numerical Simulation (DNS) of a swept flat plate. Unlike many previous computations, Bonfigli and Kloker examine secondary instabilities of both stationary and traveling crossflow waves. The DNS forced a specific stationary crossflow wavelength, and once these had saturated used wall blowing to force secondary instabilities. Their work produced several important results.

First, Bonfigli and Kloker computed the phase speeds of each mode of the secondary instability. They found that the phase and group velocities of the secondary instability were nearly equal, and the type-I phase speeds were slightly lower than the type-II speeds — $0.9 U_e$ and $1.1 U_e$, respectively. For secondary instabilities of the travelling waves, the phase speeds are slightly lower than for the stationary waves.

Second, the DNS results showed no sign of type-II instabilities, despite their appearance in experiments. The SLST results indicate type-II instabilities should be the most amplified. Bonfigli and Kloker attribute the discrepancy to inaccuracies in the SLST and in the base state from which secondary instabilities are calculated. Indeed, they found that the growth rates calculated by SLST were highly sensitive to the cross-plane velocity components in the base flow.

Finally, Bonfigli and Kloker determined that the base state (the stationary or travelling crossflow waves) need not be periodic for the secondary instabilities to appear. Secondary instabilities on neighboring crossflow vortices develop independently.

Li and Choudhari [31] examined spatially growing secondary instabilities from the swept-wing experiments of Reibert et al. [12]. Previously, most secondary instability computations used the temporal instability approach instead. Li and Choudhari's work generally corroborated previous computational efforts. In addition, it was found that increasing the roughness height causes the secondary instabilities to appear earlier (as this increases the initial amplitude of the primary stationary vortex), but with lower growth rates. This leads to roughly the same total amplification of the secondary instabilities at breakdown regardless of initial stationary wave amplitudes. This confirms that an e^N approach based on secondary instability amplification is useful. In fact, their

findings showed that either the y modes (type II) or the z modes (type I) could be used for such a correlation.

1.3 High-Speed Crossflow Boundary Layers

The body of work for the secondary instability of high-speed stationary crossflow waves is much smaller than that for low speeds. In 1996, Malik et al. performed computations for the crossflow instability over a swept cylinder at Mach 3.5 [17]. The computations revealed three secondary instability modes, each with different phase velocities slightly lower than the edge velocity. They found that the frequency range of the secondary instability spans from tens of kilohertz to more than 1 MHz.

To the author's knowledge, no work focused on secondary instabilities of high-speed crossflow vortices was published during the next two decades, though many experiments studied the primary crossflow instabilities at high speeds. One reason for the dearth of high-speed secondary instability experiments is the requirement for high-frequency, low-amplitude pressure or velocity measurements. In the last decade, however, pressure sensors manufactured by PCB Piezotronics have enabled the measurement of high-frequency instabilities in the hundreds of kilohertz. These sensors were first used by Fujii [32] to study second-mode waves at Mach 7. In addition, Mach-6 quiet tunnels at Purdue University [33] and Texas A&M [34] (formerly at NASA Langley) became operational in 2006 and 2008, respectively. These new quiet tunnels allowed the experimental study of stationary-crossflow-dominated transition at hypersonic speeds.

In 2008, Li and Choudhari [31] used a spatial instability framework to compute secondary instabilities on a Mach 2.4 swept wing. They studied two cases, the most unstable stationary wave spanwise wavelength ($\lambda = 3$ mm) and its first harmonic ($\lambda = 1.5$ mm). They found that for the dominant case ($\lambda = 3$ mm), all of the computed secondary instabilities were y modes (type II). For the harmonic case ($\lambda = 1.5$ mm), a z mode (type I) appears. The z mode grows rapidly initially, but is soon overtaken by the y modes.

In 2014, Ward and Henderson reported the first measurements of potential hypersonic secondary instabilities [35] (described in more detail in Reference [36]), made in the Mach-6 quiet tunnel at Purdue University using surface pressure sensors. These measurements renewed interest

in high-speed secondary instability research. High-frequency instabilities were measured by Ward and Henderson only when stationary vortices were near the sensor.

Henderson [36,37] performed experiments on a sharp 7° half-angle cone at low angles of attack (2° – 4°). Figure 1.5 shows heat flux inferred from Temperature Sensitive Paint (TSP) images and the Power Spectral Density (PSD) from a set of two runs conducted by Henderson. These are reproduced with permission from Figures 5, 6, and 8 in Reference [36]. Figure 1.5(a) shows the heat flux from a run at 4° angle of attack with a smooth cone. There are no visible stationary vortices below the 90° ray (which is indicated in the figure). Figure 1.5(b) shows the heat flux from a run with the same configuration, except a ring of nail polish has been added upstream of the paint. The nail polish ring had an RMS roughness of about $50\ \mu\text{m}$. With the added roughness, a number of large stationary vortices are evident below the 90° ray. One large wave passes over the PCB sensor at $x = 360\ \text{mm}$ on the 60° ray, at a spanwise reference of about $0.015\ \text{m}$ in the figure. Figure 1.5(c) shows the PSDs from that PCB for both the smooth and roughness-added runs. When the nail polish roughness is added, an instability appears at about $450\ \text{kHz}$, where there was nothing in the smooth case. This is believed to be a secondary instability, associated with the large streak passing over the sensor in the roughness-added case.

Ward [36,38] made similar measurements on the same cone as Henderson at 6° angle of attack. In addition, Ward shows the result of rotating the cone by a few degrees. The vortices are likely fixed to the exciting-roughness location (Ward used a Torlon dimpled roughness in the style of Schuele et al. [39]), and the roughness was fixed relative to the cone. When the cone is rolled the roughnesses are in a different part of the flow field, and so the vortices take a different path. Therefore, rolling the cone by small angles displaces the stationary vortex relative to the sensor. It is important to note that this is not a perfect measurement across the vortex. Once the cone is rolled, the measurements are of a different vortex, though presumably for small rotations the difference is small.

Ward shows power spectra from rolling a particular sensor under a stationary vortex. His Figure 35 in [36] (or Figure 8.6 in [38]) has been redrawn from the original data in Figure 1.6. The sensor is rotated from 115° to 122.5° . At 117 – 120° , a high-frequency instability appears at about $400\ \text{kHz}$. When the cone is rotated to move the sensor farther leeward, it no longer records a high-frequency instability but instead captures one at around $150\ \text{kHz}$. At the windward edge, 115° ,

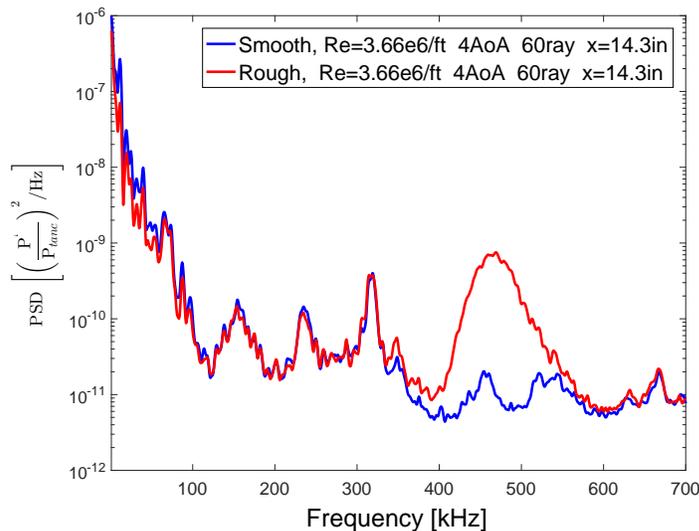
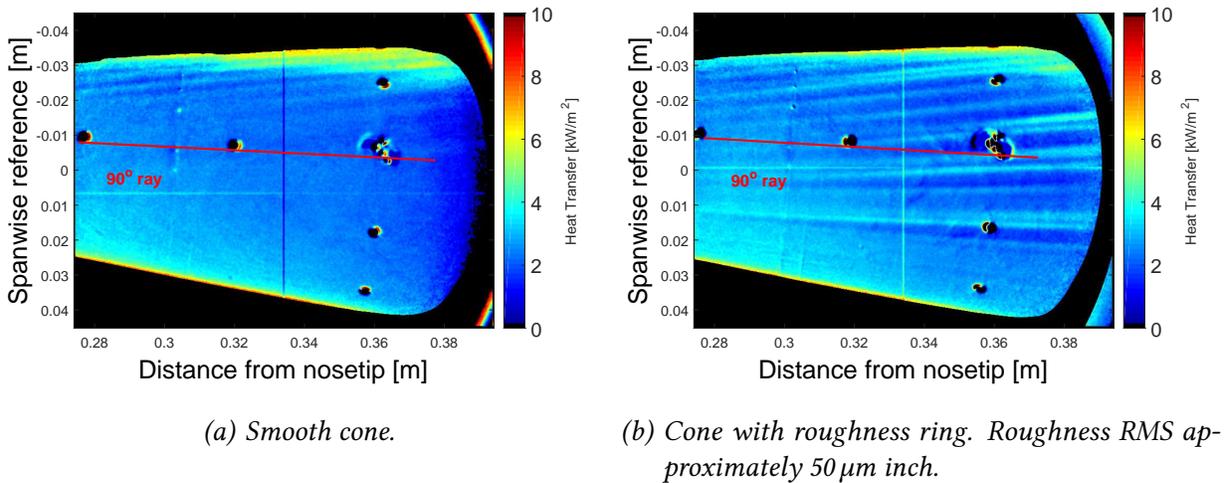


Figure 1.5. Comparison of a cone with and without added roughness. Adding roughness results in large stationary waves near the sensor, and a measured secondary instability. Reproduced from Henderson [37], with permission.

no instability is observed, showing the instability is spatially confined. However, this particular test was conducted without TSP, so it is impossible to clarify the sensor location relative to the vortex.

The secondary instabilities were only observed when a streak in the heat flux passed near a sensor. However, Ward and Henderson [36] note that in many cases a streak was observed

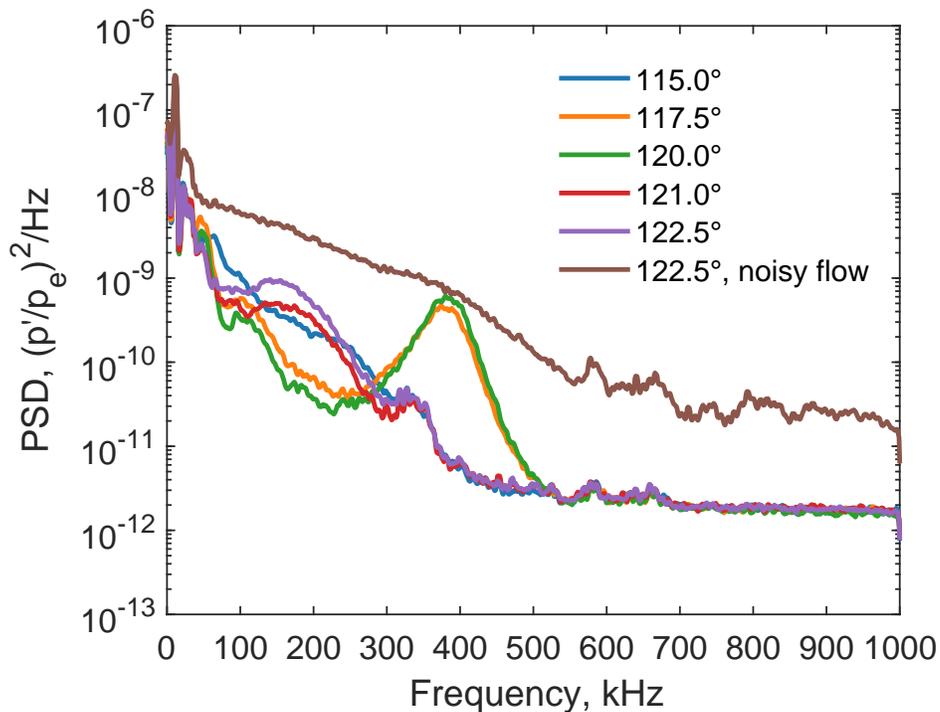


Figure 1.6. Power spectra from a single PCB rotated under a crossflow vortex. Note that the roughness elements moves as well, which changes the vortex. Reproduced from Ward [38], with permission.

passing over a sensor, but without an instability being recorded. They write that in these cases the roughness used was likely not sufficient and the vortices had not achieved the appropriate amplitude before reaching the sensors.

In 2015, Craig and Saric [3,18] reported hot-wire measurements of secondary instabilities on the same geometry as Ward and Henderson at Mach 6. The model was tested in the Texas A&M Mach-6 Quiet Tunnel at 5.6° angle of attack. The hot-wire frequency response was approximately 180 kHz, which is too low to measure the high-frequency signals measured by Ward and Henderson.

Craig and Saric measure significant fluctuations in two frequency bands, 15–60 kHz and 80–130 kHz. The lower frequency band is attributed to the traveling waves, or at upstream locations the first mode. The 80-130 kHz band is located on the leeward shoulder of the stationary waves, indicating it is a type-I secondary instability. The measurements show the amplitude of the secondary instability saturating instead of growing to turbulence. Craig and Saric suggest this is because their data were not taken far enough downstream. In addition, the axial distance

over which the amplitude is near-constant before explosive growth is much larger than for the low-speed type-I waves, which is attributed to the higher edge velocity.

Craig and Saric write that at low speeds the expected type-I secondary instability frequency is $f_I \approx U_e/(2\delta)$, which for their conditions is about 120 kHz, in the middle of the measured band. They also write that a type-II instability would have a frequency approximately double the type I frequency, which is beyond the hot-wire's frequency response for the experiment.

Until recently, there were few high-speed secondary-instability computations. The work by Moyes et al. in 2016 [19,40] is among the most thorough. Moyes et al. performed LST and spatial biglobal analysis on a sharp 7° half-angle cone at 6° angle of attack. They adjusted the initial amplitudes of the stationary crossflow base flow until the mass flux contour and RMS amplitudes agreed with Craig and Saric's [18] experimental results at $x = 380$ mm and $\theta = 118^\circ$. Six separate secondary instability modes were identified using the spatial biglobal analysis. Three of these modes appear to be low-frequency traveling crossflow (<100 kHz) modulated by the stationary waves. One of the modes has two growth rate peaks, with the higher frequency peak identified as a type-I secondary instability. It is concentrated on the shoulder of the stationary vortex. Two modes are type-II, concentrating on the crest. As expected from low-speed results, the type-II frequencies are about double that of the type-I instability.

One calculated instability shows similarities with the second mode. The eigenfunction of this mode has a maximum between the wall and the sonic line, and a second, smaller local maximum above the critical layer. The growth rate of this instability also has two peaks. The lower frequency is concentrated under the thick shoulder of the stationary wave, whereas the higher frequency peak is found in the thinner trough. As with second-mode waves, the frequency seems to scale inversely with boundary layer thickness.

The Moyes et al. computations show secondary instabilities in the same frequency bands measured by Craig and Saric [18]. Craig and Saric measured a modulated travelling wave at 15–60 kHz (mode II from Moyes et al.) and a higher-frequency type I secondary instability at 80–130 kHz (computed as mode II, peak II by Moyes et al.).

According to Moyes et al. the discrete roughness element wavelength of 7.2° (wavenumber of 50) used by Ward and Henderson [36] results in a different shape of the crossflow vortices in the spanwise and wall-normal plane [40]. The computational data could not be compared to

experimental profiles (as Ward and Henderson do not have measurements within the boundary layer), so the initial amplitudes were taken from a correlation with roughness amplitude by Balakumar and Owens [41]. The computed secondary-instability pressure modes for this case, provided in Figure 1.7, shows that the secondary instabilities are azimuthally confined. This agrees with the measurements by Ward and the present experiment. For the pressure modes illustrated in Figure 1.7, mode I is a modulated traveling wave and mode II is a type-I instability. Mode III is the type-II instability, and mode IV is the second mode. The computed frequencies of these modes are close to the measured instability frequencies from Ward and the present experiment. Interestingly, the computations show that only the mode IV instability has a significant portion of its mode shape at the wall.

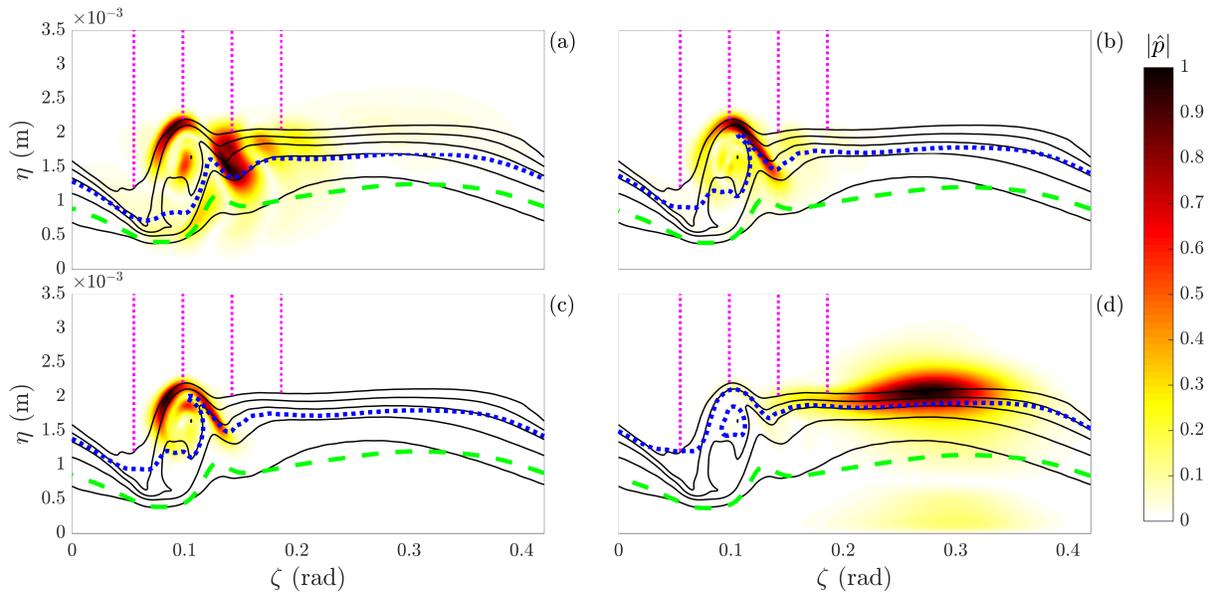


Figure 1.7. Isocontours of the normalized pressure fluctuation amplitudes $|\hat{p}|$, isolines of basic-state mass flux $\bar{\rho}\bar{u}^*$ (black lines), critical layer (blue line), relative sonic line (green line), and azimuthal angles of $\theta = 115^\circ, 117.5^\circ, 120^\circ, 122.5^\circ$ (the vertical magenta lines) at $x = 405$ mm. (a) Mode I at 15 kHz, (b) Mode II at 125 kHz, (c) Mode III at 375 kHz, and (d) Mode IV at 140 kHz. Reproduced from [40] with permission. Caption from [19]. Flow conditions can be found in [40].

Li et al. have also performed several computations of high-speed secondary instabilities [42]. A quasi-parallel method was used with a wavenumber of 60 waves per circumference. The most unstable instability was found to be a modified second-mode wave, similar to the results of Moyes

et al. Li et al. also found a very broad-band, type-I secondary instability. Growth-rate curves show that the modified second-mode consistently reaches higher N-factors regardless of the vortex on which it occurs. The most unstable second-mode frequency decreases with increasing azimuthal angle, an effect of the increasing boundary-layer thickness. However, the peak frequency of the computed secondary instability increases with increasing azimuthal angle (on three different vortices). At the highest computed azimuthal angle (which appears to be about 150° but is never explicitly stated), the peak frequencies are about 200 kHz for the second mode and about 300 kHz for the secondary instability.

1.3.1 Recent and Ongoing Crossflow Work

In recent years, research efforts on the high-speed crossflow instability have expanded as quickly as interest in hypersonics in general. A number of recent experiments and computations on the nonlinear breakdown of crossflow are now summarized.

The bulk of recent crossflow work has focused on the HIFiRE-5 elliptic cone. Borg conducted an extensive experimental campaign on this geometry, including the first surface-pressure measurements of the traveling crossflow waves [5,43], in which he found that traveling crossflow did not reach significant amplitudes and was not the dominant mechanism for transition regardless of the freestream noise levels. Borg also made the first infrared measurements of the stationary crossflow waves in a quiet tunnel [44–46], and measurements of the effect of freestream noise [9].

Computational researchers at Texas A&M are now mostly focused on stationary crossflow growth and their secondary instabilities on the elliptic cone, but their recent work also includes in-depth computations of secondary instability growth on a cone at angle of attack [19]. Of particular relevance the the present work is the study by Moyes et al. [47] on the effect of different initial amplitudes for stationary crossflow vortices on an elliptic cone. Using nonlinear PSE, they show that as the initial amplitudes increase, the stationary vortices reach the nonlinear saturation stage earlier on the model, but at a lower amplitude. They then performed linear spatial biglobal analysis of the secondary instabilities for two initial amplitudes of the stationary vortex. This analysis revealed that the secondary instability behaved essentially the same in both cases, with

the exception being the higher initial amplitude support higher frequency secondary instabilities (a result of their formation in a thinner boundary layer, farther upstream).

Experiments are also being conducted at Texas A&M in the Mach-6 Quiet Tunnel [48]; again these are mostly focused on the elliptic cone geometry. The aim of these on-going experiments is to determine the effect of environmental noise levels on the development of the crossflow instability, extending earlier work by Borg [9].

Recent computations at Minnesota also supported the HIFiRE-5 elliptic cone geometry. Dinzl and Candler [49] introduced a new technique for simulating roughness in their DNS study of HIFiRE-5. Each grid point on the surface of the geometry is moved by a random amount, effectively modeling the inherent surface roughness of the vehicle. In this way the naturally most amplified stationary mode can be excited at larger initial amplitudes and in a more natural way than was previously possible. The resulting stationary vortices interacted with the wall to yield merging of the hot streaks downstream. This merging appears to be intimately related to the transition process in the present experiments (see Chapter 5), so the work of Dinzl and Candler might also be applied to a circular cone at angle of attack.

Research groups at Notre Dame are conducting experiments on the effect of different disturbance geometries on downstream crossflow development. These experiments are conducted on 7° half-angle cones at 6° angle of attack. Both of these groups are focusing on the flow-control properties of perturbations on the cone. One group is studying the use of plasma actuators to generate stationary waves at different wavenumbers [50, 51]. In collaboration with the US Air Force Academy, another group at Notre Dame is conducting experiments investigating the effect of different roughness wavenumbers on stationary and traveling crossflow wave development [52, 53]. The results of these studies are at present inconclusive, though early data appear to indicate that the plasma-actuators could delay transition in some cases. Using patterned discrete roughness at Mach 3.5, Schuele et al. [39] demonstrated that subcritical forcing can delay transition in a manner analogous to low speeds.

Corke et al. [53] recently repeated this experiment in a conventional Mach 6 facility. Unfortunately, it is not clear from the new data whether the subcritical forcing has the same effect. Due to the low bandwidth of the pitot probe, it was impossible to measure the second mode. Oil flow images are used to demonstrate the effectiveness of the roughness elements in generating the

stationary crossflow vortices, but the images are very low resolution and include a number of artifacts that look similar to the stationary vortices but throughout the image, making it difficult to determine if the oil flow is in fact revealing stationary vortices at all. However, several prior experiments by others have used infrared imaging or temperature-sensitive paint to show the existence of stationary crossflow vortices in conventional wind tunnels dating to at least 1969 [10, 54–58], so they are known to exist in high-disturbance flows. This work on subcritical forcing is promising, but more detailed studies must be conducted before the process is well understood at high speeds.

Researchers at NASA Langley are continuing to perform PSE and DNS computations to understand the fundamental physics of the stationary crossflow breakdown on a 7° circular cone at Mach 6 and 6° angle of attack [20, 59]. Their most recent results are very similar to the present experiments, even though the forcing of the stationary vortices in the computation used $5\ \mu\text{m}$ elements—25 times smaller than the present work. The perturbation heat flux of the stationary vortices in the computation is about half as large as in the experiment, but the parameters of the forcing could account for this. The NASA studies also reveal that the genuine secondary instability (as opposed to the modulated second mode) occupies a very wide frequency bandwidth, from 200–600 kHz.

1.3.2 Previous Work by the Author and Current Objectives

The author began in 2015 [10, 60] to measure secondary instabilities of stationary crossflow waves on a 7° half-angle cone at 6° angle of attack. Crossflow-dominated transition can be studied on a number of geometries, including elliptic cones [4, 5, 9, 44–46, 48, 49, 61–64] and more complex three-dimensional geometries. But the 7° half-angle cone is a simple, canonical geometry for studying hypersonic crossflow, agreed upon by the hypersonic transition community. A circular cone is also easy and inexpensive to manufacture.

At least two frequency bands of the secondary instabilities were measured in Reference [60]. The amplitudes of the instabilities increased with the addition of roughness elements near the nose. Two of the measured instabilities were captured over a range of axial Reynolds numbers of about 1–2 million, with amplitudes ranging from low to turbulent breakdown. For these instabilities, the wave speed and amplitude growth were calculated. The wave speeds were all near the edge

velocity. Measured growth before breakdown for the two instabilities was between e^3 and e^4 from background sensor noise levels. Simultaneous measurement of two frequency bands of the secondary instabilities was made during a single run. It was found that each mode was spatially confined within a small azimuthal region, and that the regions of peak amplitude for one mode correspond to regions of minimal amplitude for the other.

The present work extends these earlier experiments. Several experiments have been conducted on a 7° half-angle cone at 6° angle of attack. These experiments aim to study the growth of the stationary and traveling vortices, the growth of the secondary instabilities, and the ultimate nonlinear breakdown of the boundary layer.

1.4 Approaches to Generating Stationary Vortices

To study the growth and breakdown of stationary crossflow vortices, it is important to have a well-defined, small, controlled perturbation by which to seed the flow with streamwise vorticity. The perturbation must be large enough to dominate the generation of the stationary crossflow vortices—as opposed to other perturbations on the geometry—but not so large as to trip the boundary layer. Typically, roughness is used for this purpose, though other approaches are possible (for example by using plasma glow-perturbers, as explored by Yates et al. [50,51]).

In this context, “well-controlled” means that the roughness parameters can be precisely varied and measured. For discrete roughness elements the important parameters are the physical size of the element (e.g., diameter and height), the geometry of the element (e.g., cylinder or diamond), and the spacing between adjacent elements. For distributed roughness, the distribution of the random roughness should at least be measureable, if not pre-determined. For the present experiments, it is also important that the added roughness can be shown to dominate the generation of the stationary crossflow vortices, as opposed to roughness that is inherent to the model (e.g., the steps at interfaces).

In previous studies of crossflow, the roughness has typically not been well-controlled. The experiments of Swanson [56] used $2.5\ \mu\text{m}$ rub-on transfer dots. Though this technique is cost-effective, there is no way to precisely or repeatably apply the dots. Schuele et al. [39], and later Ward and Henderson [36], used a roughness made from Torlon plastic. The plastic was dimpled

with a small pin to create indentations of a known diameter, which can be controlled to a much higher degree of precision than the nail polish. Though the dimpling method is a significant advance in creating controlled roughness, it is not without problems. For instance, it is not clear that the indentations were stable; the Torlon may have relaxed after dimpling, leading to uncontrollable and non-uniform dimpling depths. Dimpled Torlon is still being used by Corke et al. [53] to investigate possible mechanisms for control of crossflow transition.

Chynoweth [65, 66] developed the Rod Insertion Method (RIM) to create well-controlled roughness arrays for the study of second-mode transition on a flared cone. The RIM inserts are fabricated from an aluminum base in which several small brass rods are press-fit. The rods are machined by hand to a specified height. The method is quite adaptable, enabling the use of different base materials (for instance IR emissive materials like PEEK) and a wide range of sizes of the roughness rods. Though they are time consuming to fabricate, the RIM roughnesses are a well-defined input to the crossflow boundary layer. A more detailed discussion of the RIM roughness used in the present work is provided in Section 2.2.2 and Chapter 4.

1.5 Heat Flux from Infrared Images

The heat flux to the model is of great practical interest, and is more useful for comparing to computations than is the temperature change of the model. A method for inferring heat flux from TSP images has been in use for several years at Purdue. It is desirable to have a similar method for use with infrared (IR) images. Borg [44] led the effort to begin infrared measurements in the BAM6QT, including providing the calcium fluoride window. Following his work, two students from the University of Naples, Salvatore Cerasuolo and Mirko Zaccara, came to Purdue University during the Fall of 2016 and 2017, respectively. First, they thermally calibrated the infrared camera (loaned from the University of Notre Dame) and the calcium fluoride window as one system, providing a mapping from the camera's digital values to a temperature. They also provided an optical calibration method to map the image coordinates to real-world points on a model. Then, over a series of runs on a sharp cone at zero-degrees angle of attack, they validated a one-dimensional finite-difference, inverse heat-transfer calculation method against theoretical and computational models [67, 68]. Zaccara also provided a two-dimensional inverse heat-transfer

code. Inverse methods find the convective heat transfer coefficient which minimizes the square error between the experimental surface temperature and a finite-difference computed surface temperature. Here one-dimensional means only wall-normal heat flux is considered (i.e., every pixel is treated as a separate system). A two-dimensional method includes transverse conduction in the spanwise direction.

The problem of estimating heat flux from surface temperature measurements has a long history. Walker and Scott [69] provide a nice review of the various methods used in the past. They divide the methods into three classes: class 1 are essentially analytical in nature; class 2 are finite difference methods; and class 3 are inverse methods, which attempt to minimize the difference between a calculated temperature based on an assumed heat flux and the experimentally measured temperature. All three classes have advantages and disadvantages, but the inverse methods are more robust, so they are more often studied in the current literature.

Fourier decomposition is often used in this context. Estorf [70] decomposed images using a spatial Fourier transform and then calculated the heat flux via time discretization. The author's method described in Section 3.6 appears to be essentially the same as that of Fernandes, et al. [71], who treated the 1D heat-transfer problem as a linear system (though the Fernandes, et al. method is for a finite slab as opposed to the author's semi-infinite model). By using Green's functions to calculate the analytical solution to the problem, they produced the system's "impulse response," which can then be used to calculate the heat flux from a generic experimental temperature. It is also worthwhile noting that a direct finite-difference routine heat-transfer calculation, called QCALC, has been in use for the same purpose by Borg [44] and translated to Matlab by Juliano [62]. The theory of the three methods in use at Purdue—direct, indirect, and Fourier-based—is provided in Appendix D.

Though the inverse method for heat flux calculation is validated and quite robust, it is computationally expensive and more complex than necessary for the cases of interest in the present experiment. Instead, a 1D and 2D code were created based on Fourier decomposition of the surface thermal boundary condition using the Fast Fourier Transform (FFT). The Fourier-based approach has several advantages over the inverse approach, or even a direct, finite-difference method. First, the Matlab FFT operation is extremely fast. This is especially beneficial for the 2D method which in a mesh-based approach requires a large number mesh points. There is no

computational mesh required in the FFT-based method. Furthermore, the surface heat flux can be computed analytically from the temperature response, as opposed to a finite-difference approach which is an approximation of the surface heat flux and depends heavily on the mesh step size. Of course, in both cases the calculated temperature history inside the model is an approximation of the true temperature history.

Despite its usefulness, there are many disadvantages to the Fourier approach as well. The semi-infinite method only works well for large PEEK thicknesses (around 5 mm or greater). Depending on the model under test, this assumption may not hold. In addition, the method requires a large number of time points for high accuracy, and the 2D problem additionally requires high pixel density in the transverse direction. Future researchers should consider their specific problem parameters before selecting the appropriate method. The author has written a single wrapping function, `qcalc`, that will perform the heat-transfer calculation using one of several different algorithms. This code and accompanying documentation are provided in Appendix E.

2. FACILITY AND MODELS

2.1 Boeing/AFOSR Mach-6 Quiet Tunnel

The present experiments were performed in the Boeing/AFOSR Mach-6 Quiet Tunnel (BAM6QT). The BAM6QT is a Ludwieg tube, with a long driver tube connected to a converging-diverging nozzle to accelerate flow to Mach 6. A schematic of the BAM6QT is provided in Figure 2.1.

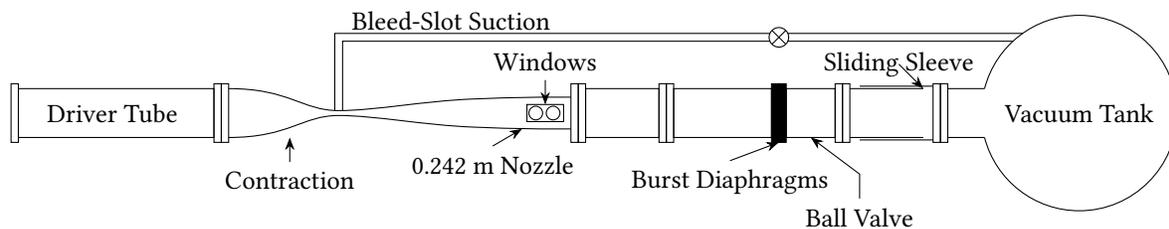


Figure 2.1. Schematic of the BAM6QT.

The BAM6QT is operated by first inserting the double burst diaphragms and pressurizing the driver tube to the desired pressure. Everything downstream of the diaphragms is at vacuum. The burst diaphragm system consists of thin aluminum sheets separated by an air gap. During pressurization the air gap is kept at the average pressure across the diaphragm system (roughly one half of the driver-tube pressure). When the tunnel has reached the correct stagnation pressure, the gap between the diaphragms is evacuated, bursting the upstream diaphragm and the downstream diaphragm in quick succession. After bursting, a shock travels downstream into the vacuum tank, and an expansion wave travels upstream through the converging-diverging nozzle and into the driver tube. Mach 6 flow starts after the expansion wave passes through the throat of the nozzle. The expansion wave reflects between the upstream and downstream ends of the driver tube. It takes approximately 200 ms for the wave to make one cycle of this reflection. Every time the wave hits the downstream end of the driver tube, the stagnation pressure drops slightly. The tunnel remains started for approximately 3–4 seconds.

Conventional wind tunnels have turbulent boundary layers on the tunnel walls. At supersonic speeds, these boundary layers radiate noise onto the model, affecting the transition mechanisms under investigation. A number of features of the BAM6QT keep the tunnel boundary layers laminar to enable operation at low freestream noise levels ($<0.02\%$). For details on the design and development of hypersonic quiet tunnels, including the BAM6QT, see Reference [72]. A bleed slot at the throat removes the boundary layer from the contraction section through a fast-opening butterfly valve into the vacuum tank, allowing a fresh laminar boundary layer to develop along the nozzle wall. The nozzle is highly polished to prevent roughness-induced transition on the walls. In addition, the nozzle is very long with a large radius of curvature to reduce the growth of the Görtler instability along the nozzle walls. Even with these elements, a turbulent boundary layer will develop in the nozzle at high enough Reynolds numbers. From 2010 to the end of 2016, the maximum stagnation pressure for quiet operation was 170 psia ($Re_\infty \approx 12 \times 10^6 \text{ m}^{-1}$). In December 2016, the upstream part of the tunnel was opened to repair a leak, and the maximum quiet stagnation pressure subsequently fell to 135 psia ($Re_\infty \approx 9.5 \times 10^6 \text{ m}^{-1}$). After the nozzle was repolished in early 2018, the maximum quiet pressure rose to 155 psia ($Re_\infty \approx 11 \times 10^6 \text{ m}^{-1}$).

The BAM6QT can also be operated without opening the bleed-slot fast-valve. This causes a turbulent boundary layer to grow on the nozzle walls, which allows the BAM6QT to operate as a conventional tunnel with higher freestream noise ($\approx 2\%$). The turbulent boundary layer is thicker than in the quiet laminar case, so the test Mach number for noisy flow is about 5.8. As of this writing, the BAM6QT is the larger of two operational quiet hypersonic tunnels in the United States.

2.1.1 Tunnel Pressure Measurement

To operate the tunnel and calculate the run conditions, one must know the stagnation pressure initially and at each point during the run. To collect this data, two pressure measurement devices are used. A Kulite XTEL-190-500A pressure transducer is flush-mounted to the wall at the beginning of the contraction section. This sensor measures the stagnation pressure during the run. The Kulite is calibrated with a 300 psia Paroscientific Inc. Model 740 Digiquartz Portable Standard pressure gauge attached just upstream of the diaphragm section. At some point during an entry,

the filling process is stopped at around five different pressures up to and including the highest stagnation pressure required for that entry. At each of these points, the Kulite voltage is recorded as well as a simultaneous reading of the Paroscientific gauge. Thus a separate Kulite calibration can be created for each entry. However, the calibrations do not drift very much, even over several years. Table 2.1 provides the calibrations for several entries to illustrate this fact. The calibration slope is very consistent, with variations on the order of $\pm 2\%$. The variation in the calibration offset is $\pm 50\%$, but this only ± 1 psia, which is less than 1% of the typical stagnation pressures. Note that the calibration process was slightly different for Entries 1 through 5, as described further in Reference [73].

Table 2.1. Summary of Contraction Kulite calibrations over several years.

Entry (Date)	Calibration Slope, psia V ⁻¹	Calibration Offset, psia
1 (06/2015)	29.31	-2.49
2 (08/2015)	30.75	-2.37
3 (09/2015)	29.59	-2.27
5 (01/2016)	29.47	-0.16
8 (02/2017)	29.35	-1.83
9 (09/2017)	29.71	-3.10
10 (01/2018)	29.53	-3.67
14 (08/2018)	29.23	-1.84
15 (10/2018)	29.41	-2.51
16 (12/2018)	29.20	-0.37
Mean	29.55	-2.06
Std. Dev.	0.45	1.1

2.1.2 Run Characteristics

The expansion-wave reflection cycle in the driver tube is illustrated in the wall pressure trace in Figure 2.2, measured with the aforementioned Kulite in the contraction section. Every 200 ms, the pressure drops by about 1%. Over the course of the typical region of analysis, from 0.5 to 2 seconds, the pressure drops by approximately 10%. The contraction pressure data is used to calculate the Reynolds number of the flow at times of interest.

There is a Dantec 55R45 hot film mounted 1.85 m (73 inches) downstream from the throat (or roughly 0.25–0.75 m upstream of the model). The uncalibrated hot-film data is used to detect the turbulence level of the nozzle wall. When the hot-film data shows a turbulent burst (a sharp spike), the data from that time is not used. The hot-film data in Figure 2.2 illustrates the expected trace from a quiet run. The startup transient lasts about 0.2–0.5 s, with fully quiet flow until about two seconds. After two seconds there is usually a small increase in noise shown on the hot film, whose origin is uncertain (for more discussion see Steen [74] and Gray [75]). Although the flow remains “quiet” to 3.7 seconds, to avoid the effects of this noise, pressure data from the model was generally not considered past two seconds.

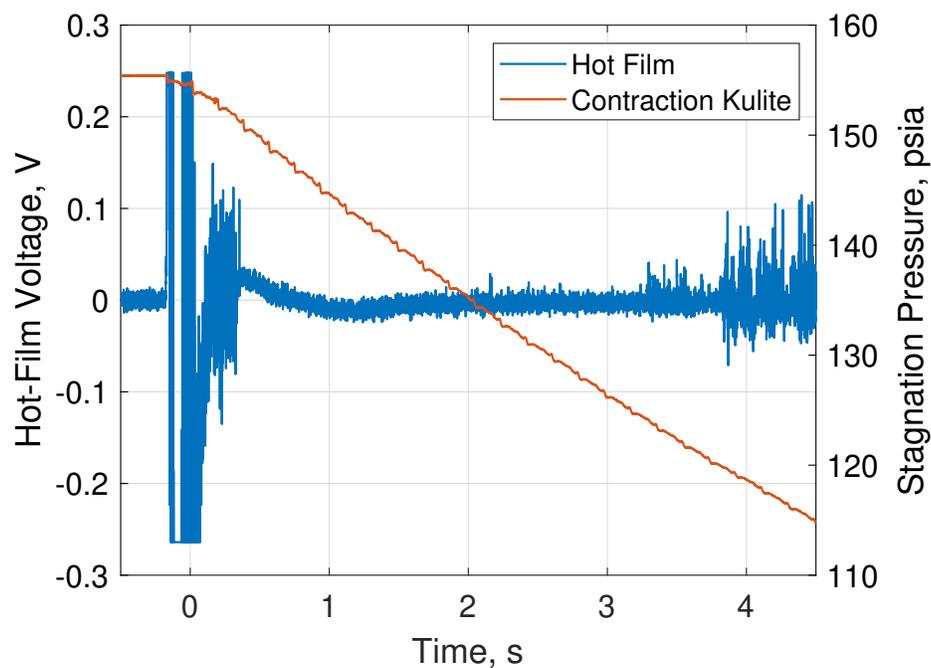


Figure 2.2. Examples of the measured stagnation pressure from the contraction Kulite and the uncalibrated hot-film voltage, from Run 1680.

2.2 Modular Cone

Muñoz et al. [76] developed a 7° half-angle circular sharp cone with independently rotatable segments to study the orientation of second mode and traveling crossflow wave-packets on a cone at angle of attack. This model inspired the author’s development of a similar cone with a

single, independently rotatable sensor section, which could be used to measure surface pressure fluctuations at any arbitrary azimuthal angle on the cone. The resulting system is called the Modular Cone, and was first used in February 2016. The initial design was mostly aluminum 6061, with a 17-4 PH stainless steel central shaft, angle of attack adapter, and nosetip. Since that first iteration, several additional components have been fabricated in both aluminum and Poly-Ether Ether Ketone (PEEK), a high-emissivity plastic used for infrared measurements. A family photo of the Modular Cone System is shown in Figure 2.3. A schematic of the Modular Cone is provided in Figure 2.4, and engineering drawings of the components of the Modular Cone are included in Appendix H.



Figure 2.3. The Modular Cone System.

The Modular Cone provides the ability to measure surface pressure fluctuations over a large portion of a cone—even the entire azimuthal extent if necessary—at any angle of attack over the course of several runs. In the latest iterations, a single ray of sensors spans the entire length of the Sensor Frustum. Before each run, the Sensor Frustum is rotated to a different azimuthal angle and locked into place. Over the course of dozens of runs, a map of the surface pressure

fluctuations over the measured region is slowly developed. Such a technique is not limited to studies of crossflow; any geometry which is mostly axisymmetric but has spanwise variations in the flow can be measured this way (for instance a flared cone or a cone with a fin).

Though this technique allows the measurement of the location and growth of secondary instabilities over a large portion of the cone, it is not without its drawbacks. The size and resolution of the measurement region is directly proportional to the number of runs. In addition, this technique can only be performed in a tunnel which can repeat the same run over and over with minimal variation in conditions. The BAM6QT satisfies this condition; some blowdown tunnels may not be able to.

One of the largest points of uncertainty is the azimuthal location of the sensors. Previous models had azimuthal scales etched on their bases, but these scales were difficult to use and had low precision and accuracy. The initial azimuthal scale on the Modular Cone consisted of Sharpie dot marks every 5° in addition to a long pointer mark on the cone along the sensor array. Both sets of marks were added by the author using a vertical mill, but they still had fairly low accuracy due to the author's inexperience in setting up the equipment. A more permanent solution was then added to the base of the Modular Cone by inscribing marks with a center drill, but this method suffered the same issues as the Sharpie marks.

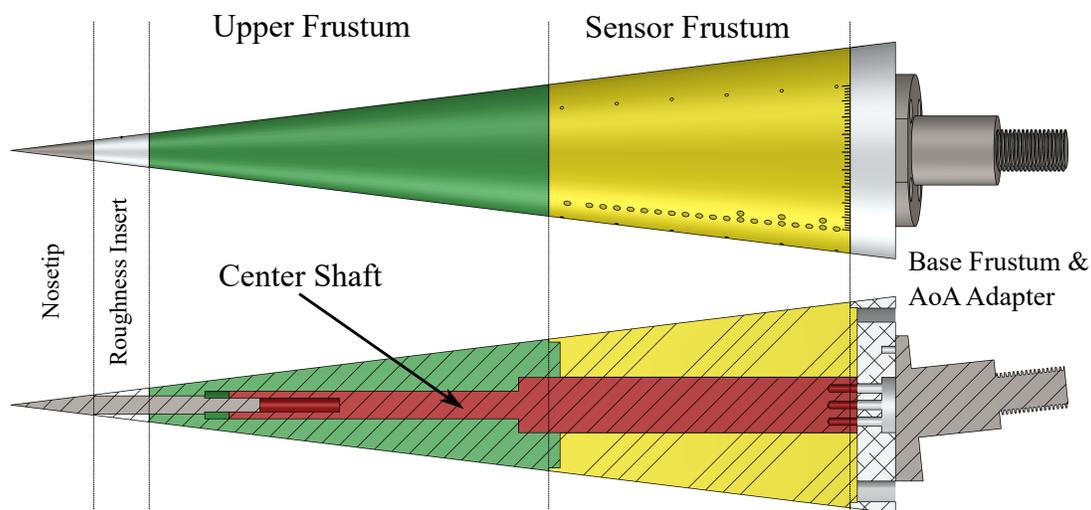


Figure 2.4. A schematic of the Modular Cone.

The solution was to have a Purdue machinist, Jim Younts, inscribe a vernier azimuthal angle scale on the cone. Two new parts were fabricated; a new sensor frustum with the main scale (2° increments) and a base frustum with the vernier scale (1.5° increments). The scale has a design precision of 0.5° , but smaller increments can be inferred fairly easily if necessary. The accuracy of the scale depends on the roll angle of the cone with respect to the angle of attack adapter. This can be estimated using IR imaging by examining where the lee ray appears in the IR images.

The Sensor Frustum used for PCB measurements has 23 sensor ports arranged along two azimuthal rays, as shown in Figure 2.4. The offset ray has three sensor stations for making repeatability measurements as the Sensor Frustum is rotated over the course of an entry. Table 2.2 provides the axial and azimuthal locations of the PCB sensor ports for this model. In addition, an array of Kulite sensors was used in a different Sensor Frustum to measure the traveling crossflow phase velocity. The positions of those four Kulites are listed in Table 2.3.

A 17-4 PH stainless steel nosetip threads into the forward end of the model. Between the nosetip and model is space for the roughness insert. The nosetip is common to all the models. As illustrated in the magnified view of Figure 2.5 the nosetip radius is approximately $30\ \mu\text{m}$. This image was taken with a Zygo Zegage white-light interferometer.

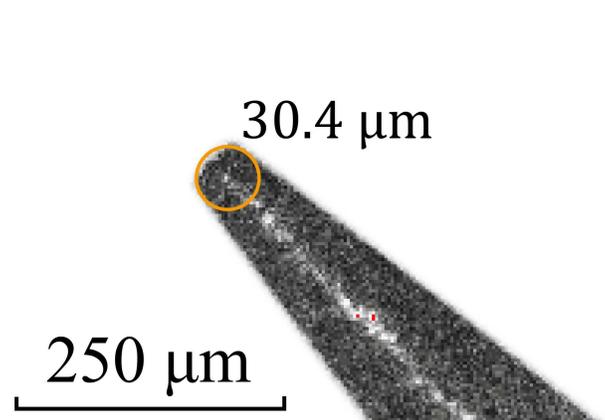
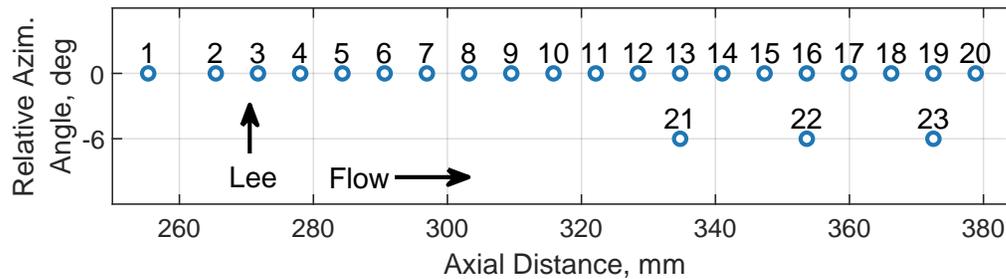


Figure 2.5. Magnified view of the sharp nosetip, showing the radius $r = 30.4\ \mu\text{m}$. Taken with the Zygo Zegage white-light interferometer.

Table 2.2. PCB sensor positions for the Modular Cone Sensor Frustum.

Sensor	Axial Position, mm	Azim. Offset, deg
1	255.3	0
2	265.4	0
3	271.7	0
4	278.0	0
5	284.3	0
6	290.6	0
7	296.9	0
8	303.2	0
9	309.5	0
10	315.8	0
11	322.1	0
12	328.4	0
13, 21	334.7	0, -6
14	341.0	0
15	347.3	0
16, 22	353.6	0, -6
17	359.9	0
18	366.2	0
19, 23	372.5	0, -6
20	378.8	0



2.2.1 Model Steps, Gaps, and Flaws

As with any engineering assembly, the interfaces between mating components of the Modular Cone are not perfectly flush. This can be especially problematic for the interfaces near the nosetip, where the boundary layer is thin. Unfortunately, it is often not practical to minimize the step between two components. For instance, the roughness insert is designed to be placed in several

Table 2.3. Positions of the Kulites in the Modular Cone.

Sensor	Axial Position, mm	Azim. Offset, deg
1	327.7	0
2	329.7	0
4	328.7	-2.75
5	331.5	-2.75

different azimuthal orientations, and to be interchangeable with other inserts. Ideally, the nosetip, roughness insert, and Upper Frustum would be machined together, to minimize the step at the interface, but that would—as one example—require as many nosetips as roughness inserts, which is not cost-effective. Furthermore, the step size is also dependent on the location tolerance of the alignment pin in the roughness insert and the corresponding hole in the Upper Frustum. Thus the step size for the baseline configuration of the RIM insert in Chapters 4 and 5 is not the same as the step size when the roughness is rotated to a different azimuthal orientation. In addition, the use of PEEK complicates the process, because the plastic material is difficult to machine to precise tolerances. Also, its thermal expansion coefficient is double that of aluminum, which can result in different step heights as the model heats up over the course of many runs.

Figure 2.6 provides a simplified schematic of the steps which are formed at the interfaces. Note that ranges are given for the step heights; the actual step height depends on the configuration of the components involved and the azimuthal location of the measurement. The steps were measured by taking a negative mold of the interfaces with a Struers RepliSet replication system, and measuring the mold in the Zygo Zegage white-light interferometer (see the following section for details on the Zegage).

In sum, though the step sizes at the component interfaces were minimized to the extent that it was practical, there are still modest steps at each interface. For the present work, an experiment was undertaken to ensure that the roughness elements—and not the steps at the interfaces—were responsible for inducing the stationary vortices. Those measurements are presented in the first part of Chapter 4. In addition, Table 2.4 lists the roughness Reynolds number, Re_{kk} , of the steps between the components at the wind ray, calculated using the Navier-Stokes solution described in Section 3.5. The boundary layer is thinnest at the wind ray, so these values represent a “worst-case”

level. Also listed is the Re_{kk} at the wind ray for a height of $k = 152 \mu\text{m}$, which represents the worst-case condition for the roughness insert used in Chapter 5. The Re_{kk} increases extremely quickly with increasing k , so even though the steps are on the same order as the roughness size, the Re_{kk} for the steps are an order of magnitude smaller. However, Re_{kk} is just one of many important parameters characterizing the effect of the roughness, so these values do not form the complete picture. A further discussion of the Re_{kk} of the roughness is provided in Chapter 4.

Unfortunately, it is not possible to be certain that the steps had no effect on the measurements presented in this work. Future experiments, as well as computations, should explicitly study the effect of various step sizes on this geometry to better understand their role in the transition process. It should also be noted that there was a small scratch on the nosetip at around $x = 32 \text{ mm}$. The scratch was about $5 \mu\text{m}$ deep, with no large protrusions. It extended roughly 20° on the sensor side of the model.

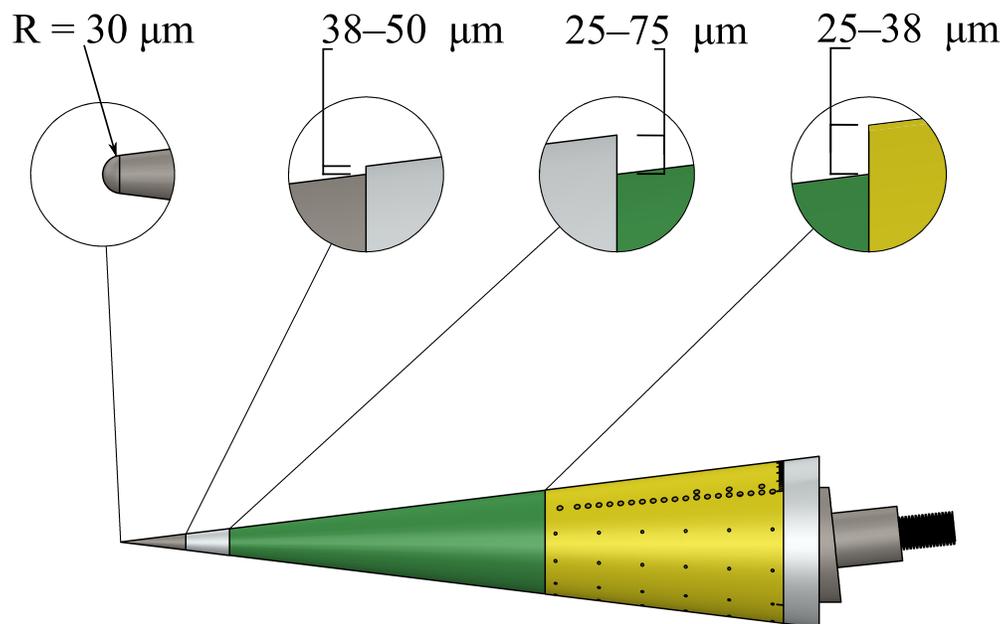


Figure 2.6. Illustration of the steps between the Modular Cone components, as well as the nosetip radius. Not to scale.

Table 2.4. Wind ray Re_{kk} of the steps at the component interfaces, compared with a $152\ \mu\text{m}$ reference (the maximum element height on the RIM insert shown in Figure 2.8). From the CFD computation discussed in Section 3.5, $Re_{\infty} = 11.6 \times 10^6\ \text{m}^{-1}$.

Interface (step height)	Re_{kk} , step	Re_{kk} , $152\ \mu\text{m}$
Nosetip/Roughness Insert ($50\ \mu\text{m}$)	14	280
Roughness Insert/Upper Frustum ($75\ \mu\text{m}$) ¹	25	156
Upper Frustum/Sensor Frustum ($38\ \mu\text{m}$)	3	61

¹ Backward facing step. Re_{kk} is not well defined for this case.

2.2.2 Roughness Elements

The Rod Insertion Method (RIM) for creating controlled, discrete roughness was developed by Chynoweth [65]. It comprises small, $560\ \mu\text{m}$ (0.022 inch) diameter brass rods press-fit into a small aluminum or plastic frustum. Table 2.5 provides a list of the roughness inserts used in the present work and their properties. Figure 2.7 is a photograph of the two roughness inserts examined in depth: a RIM insert with a nominal element height of $127\ \mu\text{m}$ and the sandblasted insert.

Though difficult and time-consuming to fabricate, the parameters of the roughness elements on the RIM inserts can be controlled with higher precision than is possible with earlier techniques. In addition to the RIM roughness, an insert was fabricated without any rods and sandblasted to a high amplitude of randomly distributed roughness. Such a roughness will tend to force the wavenumber that is locally most amplified (which can vary azimuthally) instead of a specific, constant input wavenumber.

Measurements of the roughness inserts were taken with a Zygo Zegage white-light optical profilometer; the measurements are of the height of the roughness above the surface of the roughness insert. The vertical resolution of the Zegage is quoted as 3 nm. For these measurements, a $2.75\times$ lens was used, which provides a $3\ \text{mm} \times 3\ \text{mm}$ measurement area on the roughness insert, with a $5\ \mu\text{m}$ horizontal resolution.

Because the elements are on a 7° half-angle incline to match the body of the cone, the roughness insert was placed on a rotating mount, inclined by 7° . Drawings of the mount are included in Reference [73]. A Newmark RT-2 motorized rotary stage was used to rotate the roughness insert. The RT-2 has a resolution of 0.32 arc-seconds, and an accuracy of 70 arc-sec. For the RIM insert, a

series of images of the roughness elements was taken by the Zygo and stitched together in Matlab. An image was taken with a roughness element centered in the field of view. Then, the RT-2 stage rotated the roughness to center the next element. This was repeated until the desired number of elements were measured. The stepper motor in the RT-2 stage conveniently has a resolution of 9° per 100,000 steps.

Table 2.5. Properties of the roughness inserts used in the present experiments.

Roughness	Num. Elements	Height, μm	Diameter, μm	Center-to-Center Spacing, deg.
RIM12x-5-22-9 ²	12	127 ¹	560 ¹	9 ¹
Sandblasted ³	–	3.8 (RMS)	–	–
Other RIM ⁴	12	50.8 ¹	560 ¹	9 ¹
		152 ¹		
		203 ¹		
		254 ¹		
		305 ¹		
		356 ¹		
		406 ¹		
635 ¹				

¹ denotes nominal quantity.

² used for measurements in Chapters 4 and 5.

³ used for measurements in Chapter 6.

⁴ used for measurements in Chapter 4.



Figure 2.7. A photograph of the two roughness inserts used in the present experiments: Sandblasted (left) and RIM (right).

2.2.3 RIM Roughness Properties

As described above, a series of measurements were taken of the individual elements on the RIM insert and stitched together. The RIM inserts used in this work have rods spaced 9° apart, yielding an azimuthal wavenumber of 40. Note however that there are only 12 elements on each insert, so the discrete roughness only covers between -20° and 90° on the sensor side of the model. Figure 2.8 provides a two-dimensional height map of the elements and their heights. Note that the height map is unwrapped from the cone using a constant radius for simplicity, and because the axial range of the measurements is fairly small. A more accurate map would have a taper moving downstream, with the front of an element occupying a larger azimuthal extent than the rear.

The height map shows that the element diameters and spacing are fairly uniform. There is a slight wiggle in the axial location of the elements, on the order of 10% of the roughness diameter. The center slice shows that the element heights are all close to nominal, but there is still considerable variation of up to 20% of the nominal height. Note that the offset of the azimuthal scale in both charts is only approximate. The relative azimuthal orientation of the RIM insert

with respect to the rest of the cone could vary by about 0.5° between installations (though once installed, its orientation was fixed).

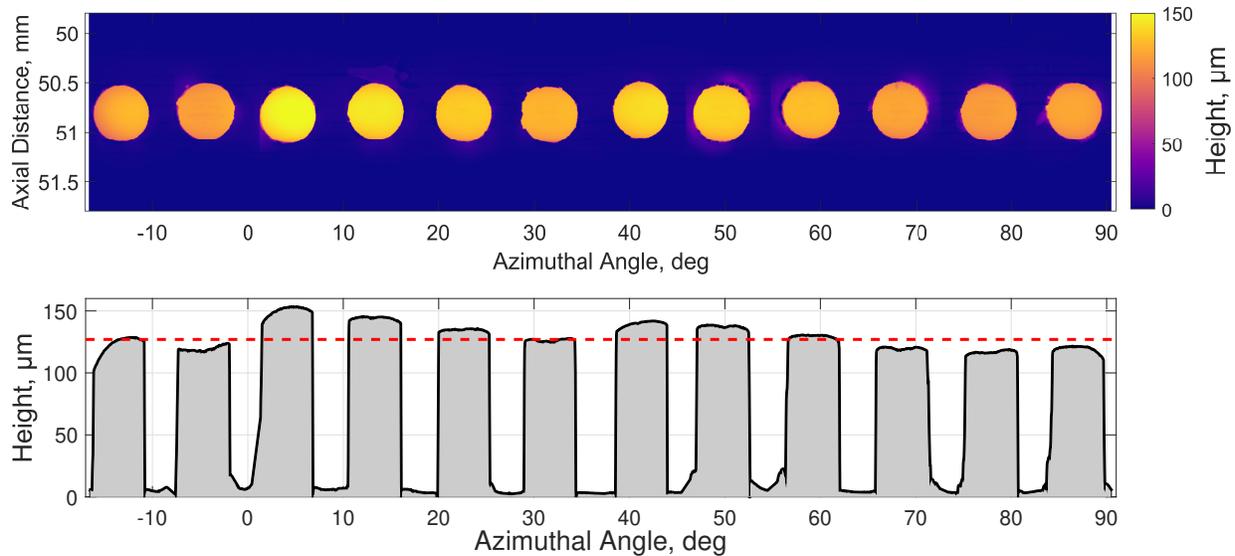


Figure 2.8. Stitched Zygo measurements of all 12 elements on the RIM roughness. A contour map of the elements, with the cone axis pointed down (top), and a slice along the $x = 50.8$ mm plane (bottom). The red dashed line is the nominal roughness height, $127 \mu\text{m}$ (0.005 inch).

There is some question as to the accuracy of the Zygo measurements, considering the measurements are unrolled from a conical surface, among other factors. To verify the measurements, an element was imaged in a confocal microscope at the University of Notre Dame. The mounting under the microscope was less precise than under the Zygo, so the raw data had an inclination with respect to the camera. The resulting height profile was rigidly rotated flat, and smoothed with a 20 point moving median filter to reduce spurious peaks in the data. Figure 2.9 shows the resulting comparison between the Zygo and confocal measurements. The profile is along an axial ray moving downstream through the element at 22.5° .

It is important to note that though the optical measurement techniques show non-uniform element heights, a third independent measurement disagreed. Jim Younts, a tool and die machinist at Purdue with several decades of experience, precisely measured the element heights with a 0.0001-inch dial indicator. He found that all of the elements were within $2.5 \mu\text{m}$ (2%) of the nominal value. The reason for this discrepancy is unknown. It should be noted that all three measurement

techniques are highly sensitive to the skill of the operator, and the optical measurements were performed by students with limited experience.

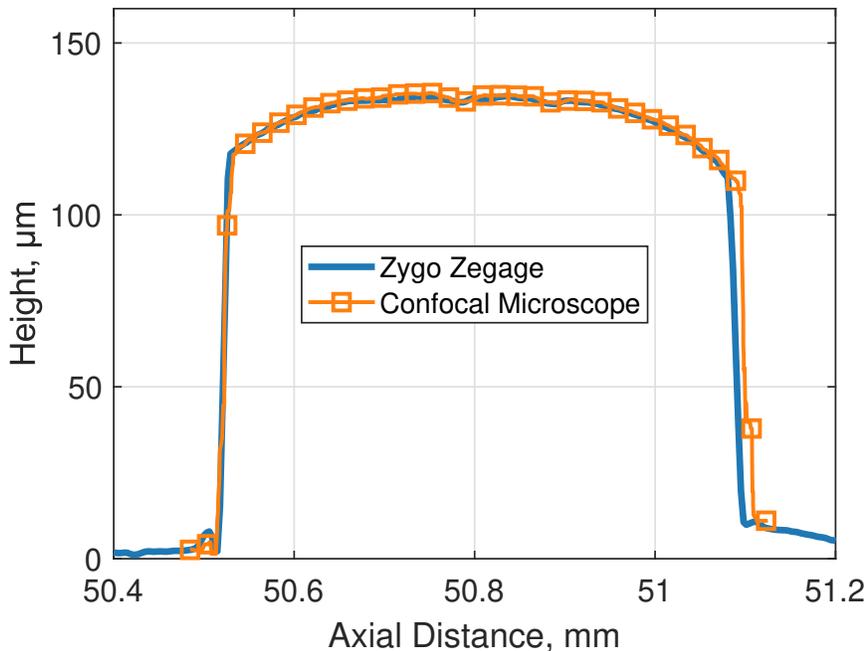


Figure 2.9. Comparison of measured height across a single roughness element using the Zygo optical profilometer and a confocal microscope.

A final property to examine is the forced wavenumber. Here the wavenumber is defined as the number of waves over the entire circumference, or

$$m \equiv \frac{360^\circ}{\Delta\theta}, \quad (2.1)$$

with $\Delta\theta$ being the azimuthal spacing of the roughness elements (nominally 9° center to center). Thus the expected wavenumber for the RIM insert is $m = 40$, though because the elements have a rectangular profile, there will be additional harmonics. One way to determine the magnitude of the forcing at each wavenumber is to analyze the Fourier coefficients of the roughness profile $z(\theta)$,

$$z(\theta) = \sum_{m=-N}^N Z(m)e^{jm\theta(\pi/180^\circ)}. \quad (2.2)$$

Figure 2.10 shows the one-sided discrete Fourier amplitudes for a wide range of wavenumbers. As expected, the dominant forcing is $m = 40$ at a height of $126 \mu\text{m}$, very close to the nominal value of $127 \mu\text{m}$. There is also clearly forcing at the higher harmonics $m = 80$ and $m = 160$, but the amplitudes are only about 30 % of the peak.

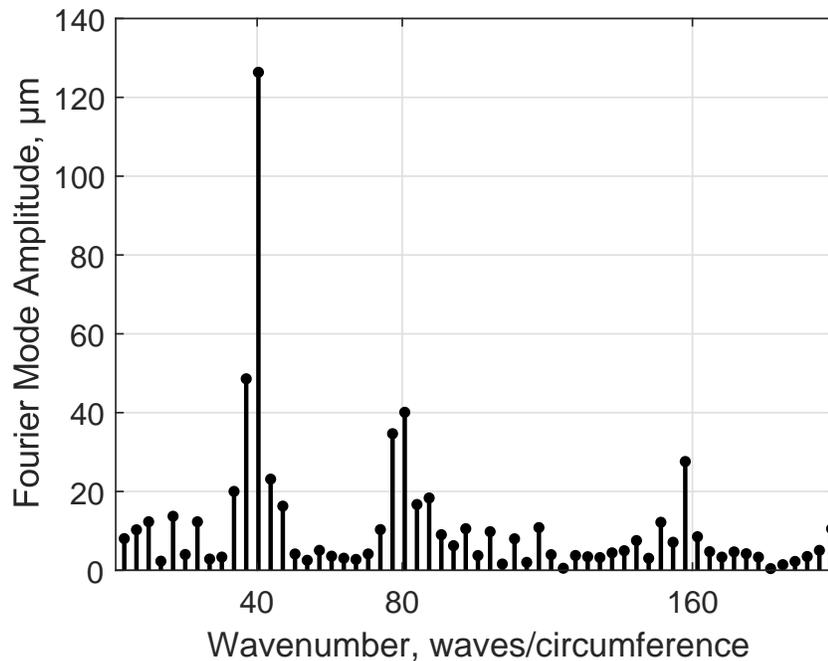


Figure 2.10. The discrete Fourier amplitudes of the roughness at wavenumbers up to 200 per circumference. The nominal roughness height is $127 \mu\text{m}$

2.2.4 Sandblasted Roughness Properties

For the Sandblasted roughness insert and a smooth insert (used for comparison), a single image was taken with the Zygo at $2.75\times$. Figure 2.11 shows these two images. There are distinct spanwise machining grooves on the nominally smooth insert (Figure 2.11(a)). The roughness on the Sandblasted insert is apparently more randomly distributed.

The relative distributions of the roughness is more evident in the histograms plotted Figure 2.12. The Sandblasted roughness obviously has a larger Root Mean Square (RMS) roughness height, and the nominally smooth roughness has a fat left tail associated with the machining

grooves. Figure 2.13 shows the roughness height histogram and estimated probability density function (PDF) for the Sandblasted case. The roughness is clearly well described by a normal distribution with RMS roughness height of $\sigma = 3.87 \mu\text{m}$. The correlation length of the Sandblasted roughness is approximately $50\text{--}60 \mu\text{m}$ in both the axial and spanwise directions. The nominally smooth roughness is approximately $50\text{--}60 \mu\text{m}$ in both the axial and spanwise directions. The nominally smooth roughness has an RMS level of about $1.3 \mu\text{m}$ (though the roughness is clearly not normally distributed, so the RMS value is misleading). The well-distributed spatial randomness of the Sandblasted roughness makes it more useful for studying “natural” transition scenarios than the relatively more ordered smooth roughness insert. Of course, for a real flight geometry the surface may have some inherent periodicity even if it is not explicitly included.

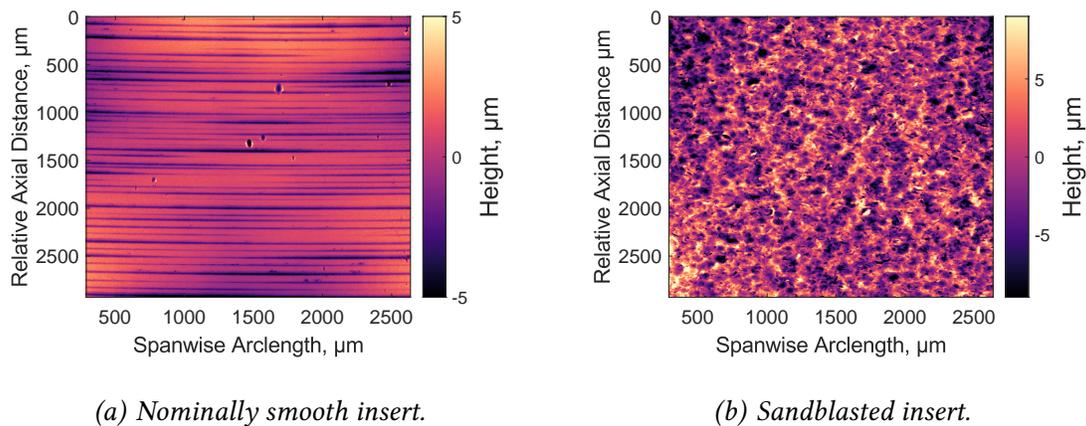


Figure 2.11. Roughness height maps for the nominally smooth and sandblasted inserts. Note the color scales are different.

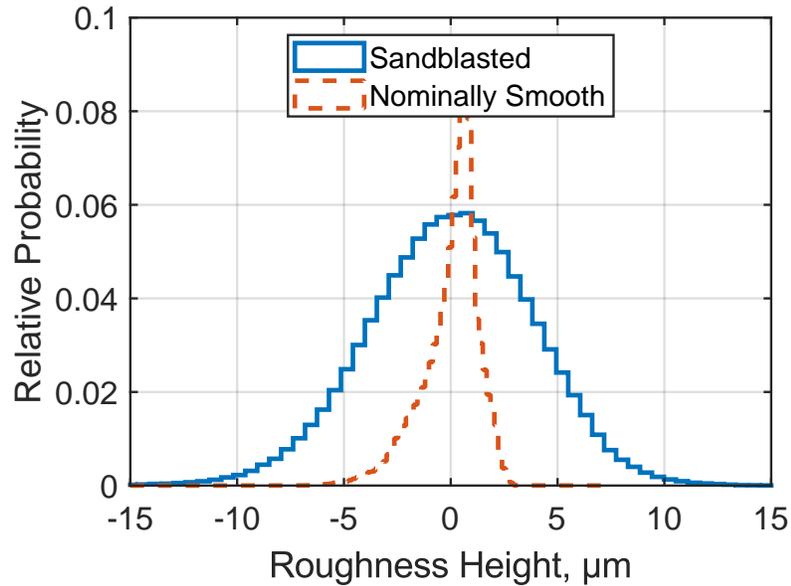


Figure 2.12. Histograms for the nominally smooth and Sandblasted roughness heights. Note that the smooth insert has a lower RMS height, as expected, but it has a fat left tail due to the machining grooves.

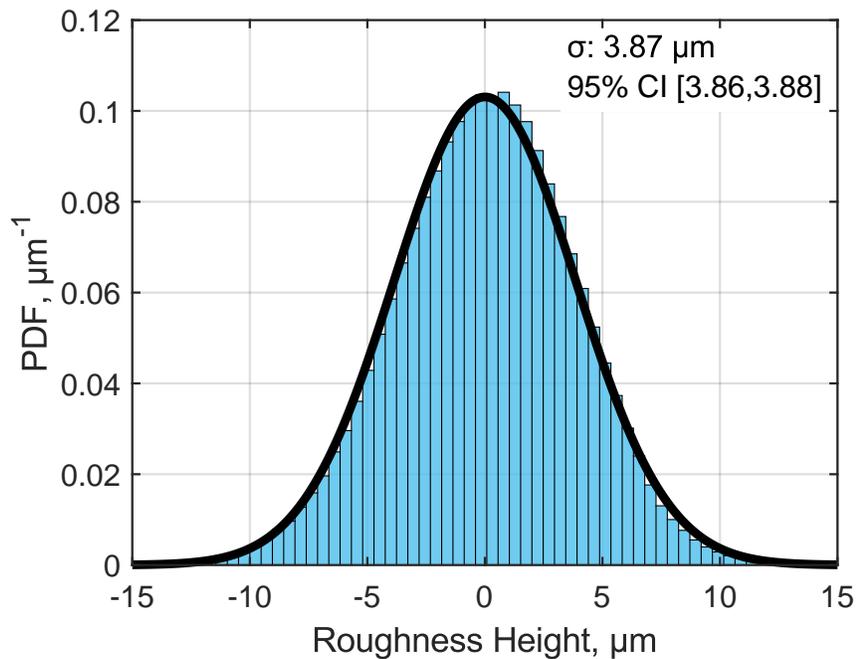


Figure 2.13. Probability density function of the Sandblasted roughness heights, shifted to have a mean of zero. The PDF is well modeled by a normal distribution with $\sigma = 3.87 \mu\text{m}$.

3. INSTRUMENTATION AND PROCESSING

3.1 PCB Piezotronics Pressure Sensors

The PCB Piezotronics 132B38—hereafter referred to as PCB132 or PCB sensor—is a piezoelectric pressure transducer useful for measuring high-frequency pressure fluctuations. The sensors are high-pass filtered above 11 kHz and have a resonant frequency greater than 1 MHz [77]. The response seems to be flat between about 20 and 300 kHz though its behavior outside this range is not well known [78]. The manufacturer states that the resolution is 0.001 psi, with single-point factory calibrations of around 100–200 mV psia⁻¹. Multi-point dynamic calibrations have been pursued in the Purdue 3-Inch Shock Tube by Berridge [79] and Wason [80], among others. Berridge shows that the single-point calibrations can be erroneous by nearly 20%. To solve this problem, new model X132B38 sensors are in development. The manufacturer states that these new sensors can measure at much lower frequencies than before and therefore can be calibrated in a more accurate manner [81]. The X-series sensors were not yet available as of this writing.

PCB132 sensors have been successfully used to measure high-frequency instabilities in hypersonic boundary layers in many previous experiments. The sensor is 0.125 inches in diameter and 0.3 inches long. The sensor's small size allows closely packed arrays to fit in small models like those used in the present experiments. Note that the PCBs are about 4° wide at their axial locations (and decreasing with increasing axial distance) on the models, and so a measurement at a particular azimuthal angle is really an average over ±2° around that angle. The sensing element is a 0.889 mm diameter circle placed in the center of the sensor. The manufacturer has determined that the “effective” sensing diameter is 0.97 mm [81]. This means the sensing surface is only about 1°–2° wide at the sensors' axial locations.

3.2 Kulite Pressure Sensors

Kulite XCQ-062-15A pressure transducers use a piezoresistive silicon diaphragm to measure the applied pressure [82]. The sensors are 1.7 mm (0.066 inch) in diameter and 9.5 mm (0.375 inch) long. The Kulites have a mechanical stop at 15 psia to prevent damage to the diaphragm at higher pressures. There is a temperature compensation module built into the sensor leads. The sensors are powered using custom-built signal conditioners that have two outputs: one channel is AC coupled and the other is DC coupled. The AC-coupled data has a gain of 10,000 \times and the DC-coupled data is amplified by 100 \times . For the data in this work, the Kulites were only AC coupled. The XCQ sensors have a very large internal resonance at around 200–300 kHz, so data are not examined above this range.

3.3 Oscilloscopes

The sensor measurements were recorded with Tektronix DPO7054, DPO7104, MDO3014 and DPO5034B digital oscilloscopes. The DPO7054 has a bandwidth of 500 MHz and the DPO5034B has a bandwidth of 350 MHz. Both were set to take 5 seconds of data at 2–5 MHz for PCB data. The DPO7104 has a 1 GHz bandwidth, but a lower internal memory than the others and so the sampling rate was limited to 2 MHz. The MDO3014 has a bandwidth of 100 MHz, and no internal hard drive. It was set to a sampling rate of 2.5 MHz. All the scopes are capable of AC and DC input coupling; PCBs were AC coupled while the other measurements were DC coupled. The input impedance was set to 1 M Ω . The scopes have a native vertical resolution of 8 bits, but using Hi-Res sampling mode the effective resolution is increased to 11–12 bits by sampling at the maximum rate for the scope and digitally averaging the result on the fly.

Close to the yaw ray the boundary layer is quite thin, and so the second mode frequencies are very large. Some measurements showed peak frequencies over 500 kHz. Harmonics of these high-frequency modes alias when sampled at 2 MHz, which is unfortunately necessary given the limited capabilities of some of the oscilloscopes. The Hi-Res sampling mode also acts as an anti-aliasing filter, so harmonics above the Nyquist frequency should not corrupt the data. However, as is clear by examining the spectra shown in Chapter 5, any such harmonics are at a very low amplitude, even without filtering.

3.4 Calculation of Flow Conditions

After filling the driver tube, the tunnel is allowed to equilibrate for 10 minutes. Just prior to running, the driver-tube pressure is recorded as the initial stagnation pressure. In addition, the temperature at the upstream end of the driver tube is measured and used as an estimate for the initial stagnation temperature. Throughout the run the stagnation pressure drops, as noted in Section 2.1.2. Using isentropic relations, the stagnation temperature at any point in the run is then

$$T_0(t) = T_{0,i} \left(\frac{p_0(t)}{p_{0,i}} \right)^{(\gamma-1)/\gamma}. \quad (3.1)$$

The viscosity during the run is calculated using Sutherland's Law without the low-temperature correction. The freestream unit Reynolds number at a given time t during the run can then be calculated as

$$\text{Re}_\infty(t) = \frac{p(t) M}{\mu(t)} \sqrt{\frac{\gamma}{RT(t)}}. \quad (3.2)$$

The Stanton number, St , is defined in this work using the freestream conditions,

$$\text{St} = \frac{Q}{\text{Re}_\infty \mu_\infty c_p (T_0 - T_w)}. \quad (3.3)$$

Note that the Stanton number is usually defined with the adiabatic wall temperature instead of the stagnation temperature, but T_{aw} is more difficult to obtain.

3.5 Computational Fluid Dynamics Model

A Navier-Stokes Computational Fluid Dynamics (CFD) simulation, calculated using US3D, was provided by Dr. Neal Bitter at Sandia National Laboratories. The calculation conditions for the CFD model are provided in Table 3.1. Figure 3.1 shows a contour of the crossplane density gradient, which highlights important features of the flow around the cone. Especially noticeable in this view is the large upwelling at the lee ray. Transition of the boundary-layer near this upwelling may be due to non-crossflow mechanisms.

The CFD data were used to normalize the pressure data, and to provide estimates for the second-mode frequency and laminar heating rate. To apply the pressure normalization, the freestream pressure was calculated from the stagnation pressure by isentropic relations. The experimental edge pressure was then determined as

$$p_{e,\text{exp}}(x, \theta) = p_{\infty,\text{exp}} \frac{p_{e,\text{CFD}}(\theta)}{p_{\infty,\text{CFD}}}. \quad (3.4)$$

Note that only the θ -dependence of the edge pressure is accounted for, because the pressure decreases only a small amount moving downstream at the PCB locations. To adjust the laminar computed heat transfer to experimental conditions, a simple Reynolds number scaling was used,

$$\text{St} \sim \sqrt{\text{Re}_{\infty}^{-1}}, \quad (3.5)$$

which should be acceptable for small deviations in Reynolds number between the experiment and the computation. The estimated second-mode frequency was calculated from the US3D boundary-layer thickness δ and edge velocity U_e as

$$f_{\text{est}} \approx \frac{U_{e,\text{CFD}}}{2\delta_{\text{CFD}}}. \quad (3.6)$$

The boundary-layer thickness was defined as the location at which the total enthalpy reaches 99.5 % of the freestream value (the default method in US3D). To adjust the estimate to experimental conditions requires the scalings

$$U_e \sim \sqrt{T_0}, \quad (3.7a)$$

$$\delta \sim \sqrt{\text{Re}_{\infty}^{-1}}. \quad (3.7b)$$

Table 3.1. US3D CFD computation conditions.

Mach	α , deg	T_0 , K	p_0 , MPa (psia)	T_w , K	Test Gas
6	6	428	1.069 (155)	300	air

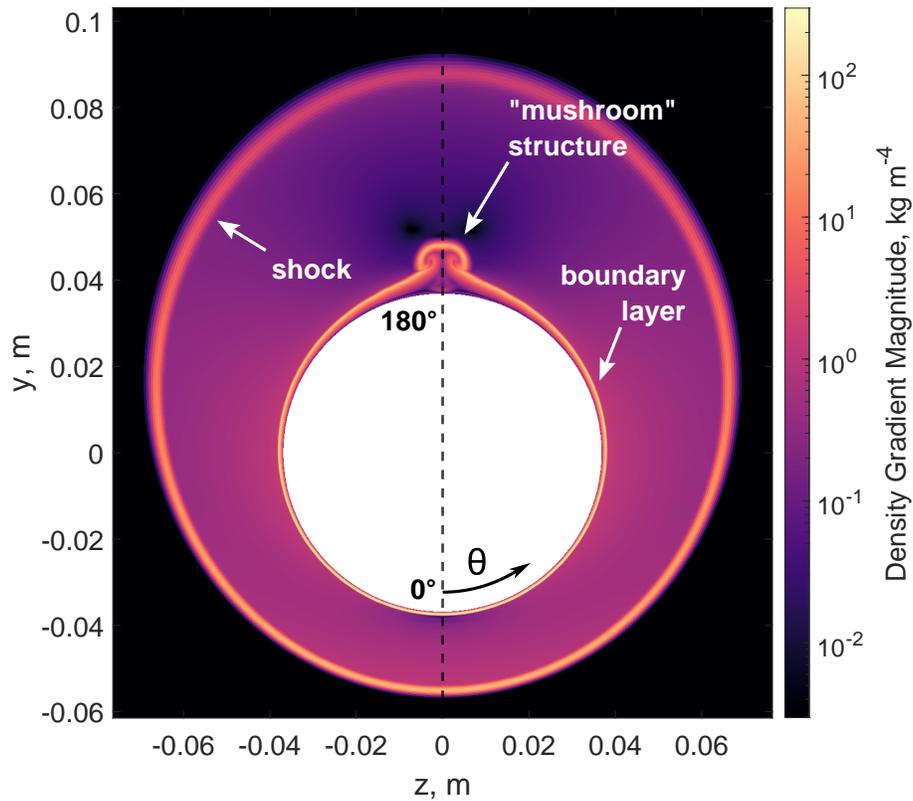


Figure 3.1. Contour of the crossplane density gradient at $x = 300$ mm from the Navier-Stokes solution. This view highlights the rapid azimuthal change in boundary-layer thickness and the mushroom-like upwelling at the lee ray.

3.6 Infrared Thermography

An Infratec ImageIR 8300 hp camera was used to record temperature images of the PEEK model. During the development of the IR system, this camera was borrowed from the University of Notre Dame. At the end of 2018, Purdue University acquired the same model of camera. According to the manufacturer the camera has a temperature resolution of 0.02 K and an accuracy of ± 1 K. The sensor has a 640×512 pixel resolution. The ImageIR 8300 hp can record full frames at up to 355 Hz, but a higher frame rate leads to a significant increasing in data size; a frame rate of 300 Hz was chosen for the data presented in this work. The spectral range of the camera is 2.0–5.7 μm .

Three f/3.0 lenses were used over the course of this research. A wide-angle 12 mm lens was used for most general-purpose imaging. A 25 mm lens was used when more spatial resolution was required (e.g., far forward on the cone). A 50 mm lens was used when the situation required very high spatial resolution (e.g., direct imaging of the RIM elements). Infratec provides a different calibration for each lens. The calibration range used for the present experiments was $-10-60$ °C, though the manufacturer has communicated that the calibration is valid for a few degrees outside of this range [83].

An infrared-transparent, 81 mm (3.2 inch) diameter calcium fluoride (CaF_2) window was developed by Borg [44]. The Notre Dame camera and the CaF_2 window were calibrated as one system by Cerasuolo [67] and Zaccara [68], however measurements with the newer Purdue camera use the factory calibration. The directional emissivity of the PEEK ϵ_θ and the window transmission loss were accounted for by assuming a transmissivity τ of 95 % and using a Stefan-Boltzmann scaling,

$$T_{\text{actual}} = \left(\frac{T_{\text{IR}}^4 - (1 - \tau \epsilon_\theta) T_{\text{amb}}^4}{\tau \epsilon_\theta} \right)^{1/4}, \quad (3.8)$$

where the ambient temperature T_{amb} was assumed to be 300 K. Because ϵ_θ and τ are very near to 1 in this case, the result should not be very sensitive to the ambient temperature selection. Note that this analysis is approximate, as τ depends on the wavelength of the transmitted light and the temperature of the window. In the future, a more thorough analysis should be undertaken of the IR reduction methods and the resulting uncertainties (as well as the sensitivity of the results to the parameters of the calculation method).

Details about the PEEK material properties, including measurements of the emissivity, can be found in References [67] and [68], and are additionally provided in the Purdue IR System Handbook, attached in full as Appendix I. These references also provide extensive information on the setup and use of the camera. In addition, Cerasuolo and Zaccara developed an optical calibration technique to map pixels in the IR image to physical points on the cone. This technique uses several images of a calibration plate which has markings at known locations. A nonlinear pin-hole camera model is least-squares fit to the images to provide the final mapping. More information on this process is left for References [67, 68] and Appendix I.

3.6.1 Algorithms for Inferring Heat-Transfer from Temperature Images

As discussed in Section 1.5, Cerasuolo, Zaccara, and the author have developed three basic algorithms for inferring the heat-transfer from a sequence of IR images: direct, indirect, and Fourier. A derivation of the theory behind each of these methods is provided in Appendix D, and the code is attached in Appendix E.

Unfortunately the inverse code provided by Cerasuolo and Zaccara is quite slow, making it impractical for day-to-day use. The author developed a reduction method based on the Fast Fourier Transform (FFT), which is computationally very efficient. However, the FFT-based code has a longer list of restrictive assumptions, so it may not be appropriate for all models. The FFT algorithm makes the following assumptions:

1. The heat transfer is either 1D (into the model) or 2D (into the model and transverse).
2. The heat transfer is semi-infinite. This is satisfied if the thermal penetration depth is much less than the thickness of the PEEK; for the BAM6QT run conditions the assumption is valid for PEEK thicknesses greater than about 5 mm.
3. For the 2D algorithm, the radius of the PEEK section of interest is much larger than the thermal penetration depth.
4. The thermal boundary condition can be accurately decomposed into a finite number of Fourier terms (i.e. no step discontinuities).
5. The camera frame rate is constant.

The penetration depth, given by

$$\eta(t) = 4\sqrt{\alpha t}, \quad (3.9)$$

is around 3 mm for PEEK in the BAM6QT (a run time of about 3 s). The thickness of the PEEK in the imaged region, however, is generally more than an order of magnitude larger.

A convenient case with which to validate the heat transfer codes is a sharp cone at zero degrees angle of attack. There exists a similarity solution for this geometry as reported by Sullivan [84] and used previously to validate temperature-sensitive paint techniques by Ward [38]. Cerasuolo and

Zaccara [67, 68] validated their 1D inverse code using this approach. Figure 3.2 shows the inferred heat transfer from the same set of temperature images, but using each of the three different algorithms. Table 3.2 provides the experimental conditions for the data in Figure 3.2. Note that the results using the direct and indirect methods are nearly identical. Also listed are the computation times in seconds for each method. All three algorithms result in errors of less than 5% from the theoretical solution. The direct and indirect methods are clearly more accurate than the Fourier method, but they take 3.5 and almost 20 times as long to compute, respectively. For a typical set of images used in Chapters 4–5 the FFT method takes roughly 80 seconds. Note that Cerasuolo and Zaccara’s data were collected without the precision angle-of-attack adapter developed by Chynoweth [66], so the angle of attack is not exactly 0.00° . Note that this comparison uses the original temperature data from Zaccara, so it has been calibrated for the presence of the window (i.e., no transmissivity correction was applied to the images).

Some measurements were also taken very far forward on the cone, where the model may no longer meet the semi-infinite assumption. In addition, these data were collected using the 50 mm lens, which was not calibrated with the window by Zaccara, so the factory calibration was used. Figure 3.3 shows IR heat-transfer data from the RIM insert and Upper Frustum at nominally 0° angle of attack. The accuracy is notably worse than farther downstream, with error on the order of 25%. The experimental conditions for these data are provided in Table 3.3.

It is unclear whether the error is due to the presence of the window or the breakdown of the assumptions in this region of the model. The IR camera will need to be calibrated by using a blackbody behind the window, as performed by Cerasuolo and Zaccara. This calibration must be repeated for every camera lens and window combination, so it will be a time consuming process. A rigorous analysis of the sources of uncertainty in the whole of the IR method should be conducted in the future.

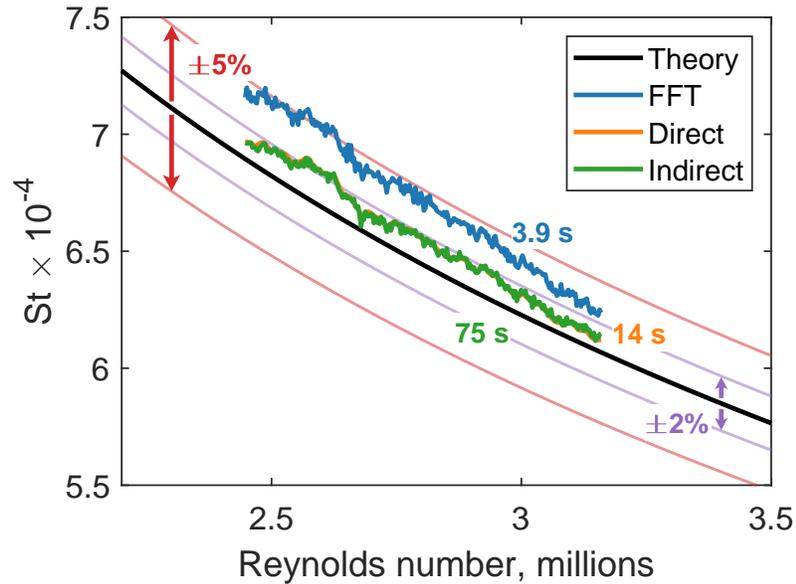


Figure 3.2. Comparison of the three methods against the theoretical solution for heat transfer on a cone at 0° angle of attack. Note that the direct solution (orange) is obscured by the indirect solution (green); they are nearly identical. The computation time for each method is listed in seconds for the author's PC. $Re_\infty = 8.9 \times 10^6 \text{ m}^{-1}$.

Table 3.2. Conditions and mesh points for the 0° angle of attack validation case presented in Figure 3.2.

p_0 , kPa	T_0 , K	Re_∞ , m^{-1}	Num. Points		
			Axial	Spanwise	Time
770	413	8.9×10^6	250	70	1350

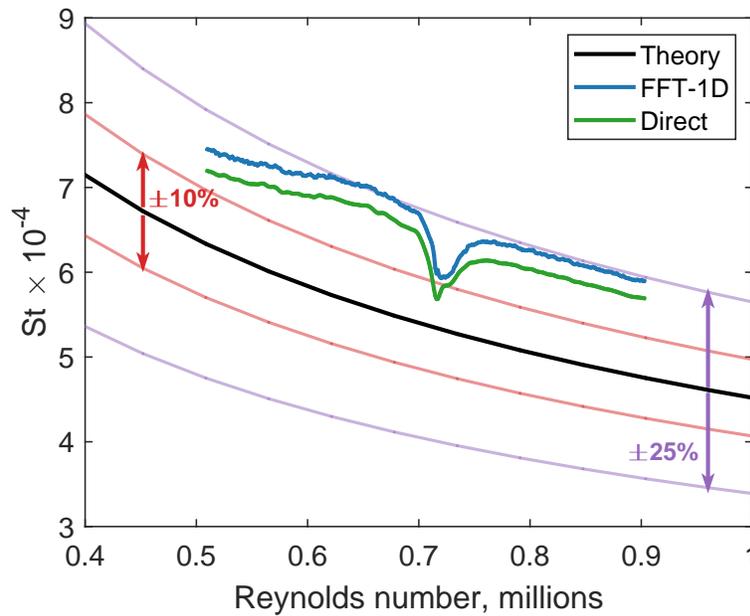


Figure 3.3. Accuracy of the heat-transfer reduction process near the nosetip. The dip in heat transfer at $Re \approx 0.7 \times 10^6$ is coincident with the interface between the roughness insert and the Upper Frustum, and is likely the result of inaccuracies in image registration. $Re_\infty = 11.3 \times 10^6 \text{ m}^{-1}$.

Table 3.3. Conditions and mesh points for the 0° angle of attack validation case presented in Figure 3.3.

p_0 , kPa	T_0 , K	Re_∞ , m^{-1}	Num. Points		
			Axial	Spanwise	Time
988	416	11.3×10^6	350	150	1200

The Fourier-based approach created by the author was also validated against the 1D inverse method provided by Mirko Zaccara on a cone at 6° angle of attack. As shown in Figure 3.4, there is a maximum difference of about 9% from the inverse method solution, and a median error of about 4%. The experimental conditions for this comparison are provided in Table 3.4. The Fourier method is two orders of magnitude faster to run than the inverse method. Note from Figure 3.4 that the 2D results agree quite well with the 1D results except at the peaks and valleys, which is to be expected given the larger spanwise gradients there. At locations farther forward on the cone, the 1D and 2D results will diverge more. The Fourier method agrees well with the old

QCALC results when run on elliptic cone data [85], and also with TSP results on a cone with a highly-swept fin [86].

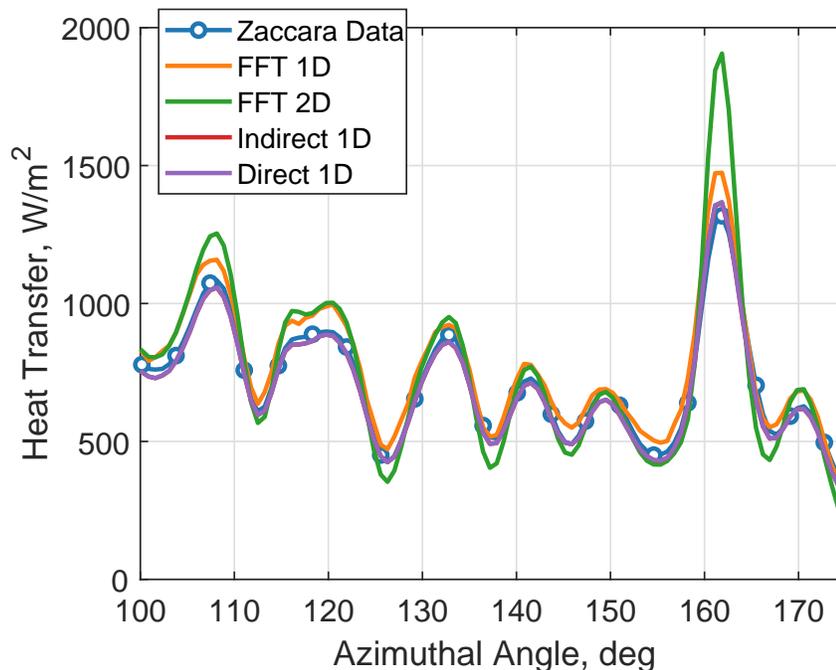


Figure 3.4. Comparison between the different 1D heat-transfer reduction methods and the Zaccara results [68] on a cone at angle of attack. Note that the direct and indirect results are nearly identical. $Re_\infty = 7.96 \times 10^6 \text{ m}^{-1}$.

Table 3.4. Conditions and mesh points for the 0° angle of attack validation case presented in Figure 3.4.

p_0 , kPa	T_0 , K	Re_∞ , m^{-1}	Num. Points		
			Axial	Spanwise	Time
708	419	7.96×10^6	–	125	700

3.6.2 Heat-Transfer Reduction Process

The following is a summary of the IR heat-transfer reduction process:

1. **Camera Optical Calibration** Several images are taken of a plate with markers at known locations. These images are then used to create a mapping from camera pixels to physical points in the tunnel, using a nonlinear pinhole camera model. See Section 4.1 in Zaccara [68] for details of the model and the calibration process.
2. **Image Acquisition** Images are taken during the run.
3. **Mesh Creation** A computational mesh is created of the surface geometry under test. For a cone, this mesh can be simply parameterized by the axial distance x and the azimuthal angle θ . Using the pinhole camera model, the physical coordinates of the cone surface in 3D space (x, y, z) are projected into their pixel locations (u, v) in the image. The mesh is offset to align it with the image of the model. Figure 3.5 shows an example mesh aligned with an image of the cone. Note that the mesh in the figure is intentionally sparse for clarity.

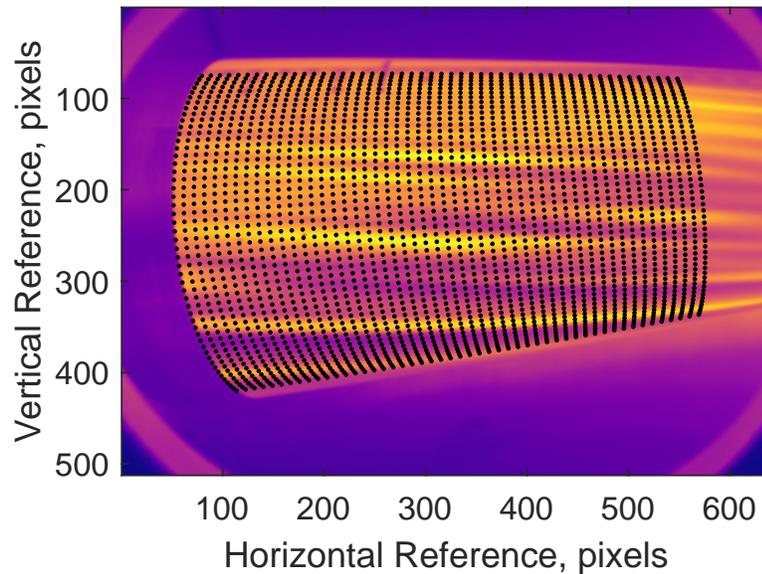


Figure 3.5. Example computational mesh for IR processing.

4. **Image Registration** The images are shifted to account for the movement of the tunnel relative to the camera during the run. The code uses a two-dimensional cross-correlation to find the displacement between two images in the horizontal and vertical directions. The registration

process proceeds backwards from the last frame and each frame is compared to the one immediately following it, which was shifted in the previous step. Figure 3.6 shows an example of the measured displacements for one run; the bottom x-axis shows the image frame, and the top x-axis shows the corresponding time during the run. The image displacement in the horizontal direction is significant, peaking at -34 pixels (roughly 10 mm). The inset plot shows that the small-amplitude oscillations of the image displacement continue well into the run. In addition, there is a constant offset of about 3 pixels between the pre-run frames and the end of the run. Note that $t = 0$ is the time at which the trigger signal is received by the camera; this time lags behind the diaphragm burst by almost 200 ms.

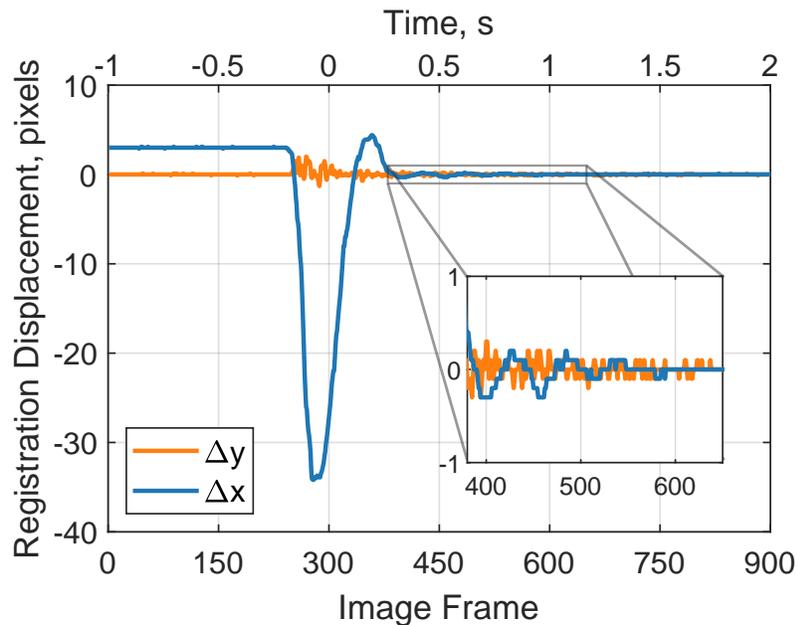


Figure 3.6. Example of the results of the image registration process.

- 5. Temperature Interpolation** The temperature at each pixel measured by the camera is interpolated onto the computational mesh points, and any emissivity and transmissivity corrections are applied (see Equation 3.8).
- 6. Heat-Transfer Calculation** The heat transfer is inferred for each mesh point from the temperature history at that point, using one of the algorithms discussed in Appendix D.

3.7 Pressure Data Reduction

Once the pressure fluctuations measured by the PCBs have been recorded on the oscilloscopes, they are digitally processed in a number of ways. Before any spectral quantities are calculated, a 0.1 s segment of the raw voltage data is taken at a desired time, which is usually determined to satisfy a desired Reynolds number. The sensor calibration is applied to the data segment to yield a pressure signal, which is then normalized by the local edge pressure as described in Section 3.5. This data segment is then used for all of the calculations described in the following sections.

3.7.1 Spectral Quantities and Fluctuation Amplitude

Power spectral densities are computed from the pressure data using Welch's method with 50% overlap between segments. The segment length is chosen to yield the desired frequency resolution, which is 5 kHz unless otherwise specified. The mean is removed from the signal, and it is then windowed with a Hann window. To compute the RMS pressure fluctuation amplitude within a given frequency band, a square root is taken of the PSD integral in the band.

The magnitude-squared coherence of signals from two sensors is a measure of the degree of linearity of a system (considering the two sensors to be an input and an output of the system). Here, the "system" is the boundary-layer instability in question. The coherence is a spectral quantity, so it is defined over a range of frequencies; in fact it is the normalized Fourier transform of the cross-correlation of the two signals. A coherence near 1.0 means that the sensors are measuring a process that has a high degree of correspondence at a given frequency, which is often an indication of linearity (i.e. the change in magnitude over the process is not amplitude dependent). Signals that have a high coherence can be used to infer other quantities, like the phase velocity of the instabilities, with confidence. Having a low coherence, however, does not have a clear implication. A low coherence could be the result of nonlinearity in the system or an indication that the signal-to-noise ratio is poor, among other factors. Thus a low coherence must be used in combination with other quantities (like the PSD and bicoherence) to properly understand the behavior of the measured instability.

3.7.2 Calculation of the Phase Velocity

The phase velocity is the propagation speed and direction of the instability wave crests. These quantities can be enlightening as to the underlying physical mechanisms, as demonstrated in Chapter 6, but they are also often useful for comparison with computations. The phase velocity was determined with a closely-packed array of four Kulite pressure sensors, using the method of Poggie et al. [4], Borg [5], and Ward [38].

First, the cross-spectrum is calculated of two pairs of signals from three or four unique measurement points. The phase delay of the signal between the two sensors is then given by

$$\tau(f) = \frac{\phi(f)}{2\pi f}, \quad (3.10)$$

where $\phi(f)$ is the phase of the cross-spectrum as a function of frequency f . To find the wave speed and angle on the cone, it is assumed that the sensors are close enough together that they lie approximately on a plane. The (x, θ) coordinates of the relevant sensors are then transformed to (ξ, η) coordinates by

$$\xi = \frac{x}{\cos(\varepsilon)}, \quad \text{distance along cone surface} \quad (3.11a)$$

$$\eta = r(x)(\theta - \theta_0), \quad \text{arclength azimuthally from reference} \quad (3.11b)$$

where θ_0 is the azimuthal angle of the reference sensor, ε is the cone half-angle, and $r(x)$ is the cone radius at axial location x .

To calculate the wave angle Ψ and wave phase speed c , two unique sensor pairs are required. Using four sensors, the wave angle can be calculated as

$$\Psi = \arctan \left(\frac{\tau_{34}(\xi_2 - \xi_1) - \tau_{12}(\xi_3 - \xi_4)}{\tau_{12}(\eta_3 - \eta_4) - \tau_{34}(\eta_2 - \eta_1)} \right), \quad (3.12)$$

where τ_{ij} is the phase delay from sensor i to sensor j . Note that for the present work a positive angle is oriented downstream and toward the lee ray as illustrated in Figure 3.7. The wave phase speed is then calculated by

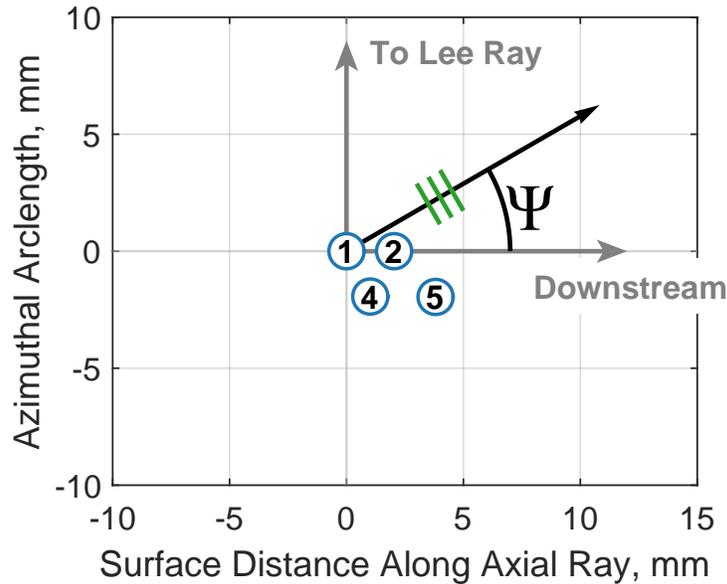


Figure 3.7. The orientation of the wave angle coordinate system with respect to the Kulite array used in this work. The outer circles at each sensor location indicate the approximate diameter of the Kulite sensing element. The green lines indicate the propagation direction of the wave crests. Note that typically for crossflow $\Psi < 0$, as the traveling waves move downstream from lee to wind.

$$c = \frac{(\eta_1 - \eta_2) \sin \Psi + (\xi_1 - \xi_2) \cos \Psi}{\tau_{12}}. \quad (3.13)$$

The finite sampling rate $T_s = 1/f_s$ means that a time delay of less than T_s is not resolvable, or equivalently the resolution of the calculated time delay τ is T_s . This uncertainty can be propagated through the calculations to give uncertainties in Ψ and c .

3.7.3 Quantifying Nonlinear Interaction

The advective term of the Navier-Stokes equations, $(\mathbf{u} \cdot \nabla) \mathbf{u}$, is inherently nonlinear (in fact this is a quadratic nonlinearity as it contains a product of \mathbf{u} and its own derivatives). The growth of disturbances in the boundary layer is thus also nonlinear, though for small disturbances one can

approximate the behavior as linear (the basis of Linear Stability Theory (LST)). Near breakdown, the amplitudes of the disturbances are large enough that the nonlinear effects cannot be ignored.

There are various methods by which to analyze the degree of nonlinearity of the measured disturbances. Traditional spectral methods like the power spectral density and coherence are founded on linear theory and so provide incomplete or misleading information. However, these methods can be extended to higher orders, which can quantify the nonlinearities in the data. The lowest order nonlinearities in the boundary layer are quadratic. Quadratic nonlinearities can be examined using the bispectrum and bicoherence, which are second-order analogs of the power spectrum and coherence.

The bispectrum of a time series $x(t)$ is defined as

$$B(f_1, f_2) = X(f_1)X(f_2)X^*(f_1 + f_2), \quad (3.14)$$

where f_1 and f_2 are given frequencies, $X(f)$ is the Fourier transform of $x(t)$ at frequency f , and the asterisk denotes a complex conjugate. In practice, the bispectrum is calculated using Welch's method of averaging windowed data segments,

$$B(f_1, f_2) = E \{X(f_1)X(f_2)X^*(f_1 + f_2)\}. \quad (3.15)$$

Traditionally it is demonstrated that if the three frequencies are phase-locked,

$$f_3 = f_1 + f_2 \quad (3.16a)$$

$$\phi_3 = \phi_1 + \phi_2 \quad (3.16b)$$

the expectation of the triple product in Equation 3.15 will be non-zero. However, any constant phase difference—not just $\phi_3 - \phi_1 - \phi_2 = 0$ —between the three frequencies will result in a non-zero bicoherence. If the phases are aligned randomly, however, the expectation will approach zero as the number of averages increases. Phase-locking is indicative of a quadratic nonlinearity.

As a way to quantify the relative magnitudes of nonlinear interactions between frequency triads, Kim and Powers [87] define the bicoherence as

$$b^2(f_1, f_2) = \frac{|B(f_1, f_2)|^2}{E \{|X(f_1)X(f_2)|^2\} E \{|X(f_1 + f_2)|^2\}}, \quad (3.17)$$

which represents the proportion of power at $f_3 = f_1 + f_2$ that is due to the nonlinear interaction between f_1 and f_2 . The bicoherence is usually bounded between 0 and 1, though Hinich and Wolinsky [88] note that this may not always be the case. The code used to calculate the bicoherence, as well as a validation case, are provided in Appendix F. Due to the inherent symmetries in the bicoherence, it is sufficient to compute the values for only positive frequencies [87].

Haubrich [89] shows that the 95% significance level of the bicoherence estimate is $b_{95\%}^2 = 3/N$, where N is the number of averages used in the calculation (see also Elgar and Guza [90]). If the value of b^2 is less than this threshold, it is statistically indistinguishable from zero. However, the actual magnitude of the bicoherence above this level may be misleading as to the amount or importance of nonlinearity in the underlying physical mechanism. As shown by Hinich and Wolinsky [88], the magnitude of the bicoherence using the standard normalization is affected by the length of the window used in its calculation. In addition, in certain cases the bicoherence can accurately indicate a nonlinearity in the signal when the power spectrum does not (see Appendix F). Thus the bicoherence is only one piece in the larger puzzle of quantifying the nonlinear interaction.

Figure 3.8 shows an example PSD and bicoherence calculated from PCB pressure fluctuation data. The PSD shows a large peak at around 170 kHz and another at around 330 kHz. The bicoherence shows significant nonlinear coupling ($b^2 = 0.41$) within the 170 kHz instability and between 170 kHz and 330 kHz. In addition, there are low levels of bicoherence at (330 kHz, 330 kHz), indicating harmonic generation at that frequency as well (which explains the small peak in the PSD at around 700 kHz). The location of maximum bicoherence is indicated on the plot by the blue dashed lines.

Before using the bicoherence to understand nonlinearities in the boundary layer, it is worthwhile to understand the nonlinear PCB response to an impulse input, so “true” measurements of nonlinear growth can be distinguished from sensor response. Such an impulse was provided to

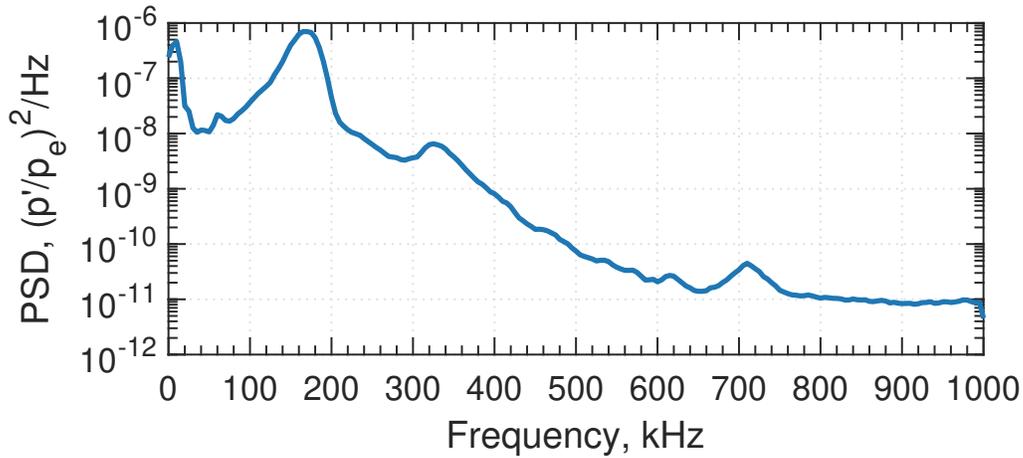
the PCB in the form of a mechanical impulse to the sting and separately as an incident shock wave in the Purdue 3-inch Shock Tube. The nonlinear response of the sensor seems to be confined to frequencies much lower and higher than the range of interest in the present work. See Appendix G for details.

3.7.4 Turbulence Metric

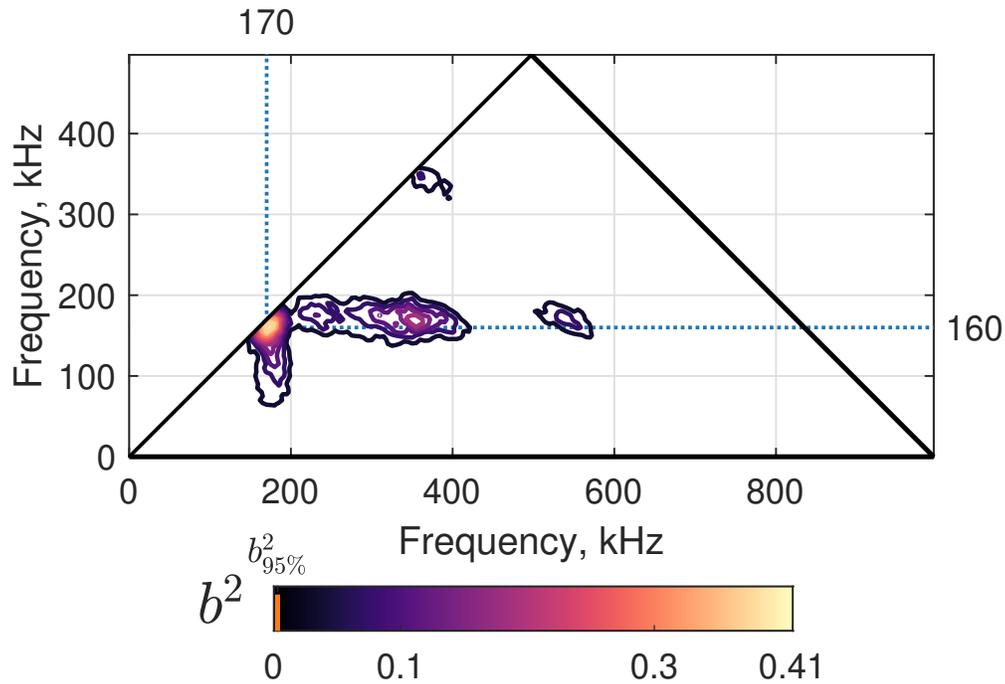
One important aspect of any boundary-layer transition experiment is the ability to determine when the flow becomes turbulent. As the transition process develops, turbulent spots begin to form intermittently and travel downstream. A measure of the degree to which the boundary layer has become turbulent is this turbulent-spot intermittency, γ , which represents the fraction of time that the flow is turbulent. Thus if $\gamma = 1$, the flow is entirely turbulent, and if $\gamma = 0$ the flow is entirely laminar. The transition process occupies the region between these two end points (though the initial stages of instability growth will also have $\gamma = 0$, as no turbulent spots will have yet formed). Unfortunately γ is quite difficult to estimate from experimental data, as it requires the ability to differentiate turbulent spots from instability waves and noise. Past approaches to the problem rely on detector functions [91], but these methods have parameters which must be carefully tuned to the data and which have little theoretical basis. The work of Casper et al. [92] to use wavelets for intermittency calculation is a step forward, and it has been used with some success by others [66].

The present author has previously used Casper's wavelet method for crossflow data [21]. However, the method is not practical for the new data in this work—the method requires the measured instabilities to be sufficiently band-limited that turbulent bursts can be resolved in regions of the spectrum where there is otherwise no measured activity. For the old crossflow data in Reference [21], when using the dimpled Torlon insert the measured secondary instabilities satisfy this condition (see also the data and accompanying discussion in Appendix C). As will be demonstrated in Chapter 5, when using the RIM insert the secondary instabilities occupy such a wide frequency band that the wavelet method cannot be applied.

Instead, the problem of intermittency calculation was recast. Rather than calculate the fraction of time that the flow is turbulent, one can examine the frequency content of the signal to determine



(a) Example PSD of secondary instability pressure fluctuations.



(b) Bicoherence of the data in the PSD, above.

Figure 3.8. An example of a bicoherence calculation to illustrate its features and to orient the reader. The 95% significance level, $b_{95\%}^2$, is provided as the orange line on the colorbar. The maximum bicoherence is given to two decimal places at the right of the colorbar. In every bicoherence plot, the maximum will be located with dashed lines to the top and right sides, which will label the frequency location of the maximum (170 kHz and 160 kHz in this example).

its similarity to a turbulent spectrum. This leads to the creation of a *turbulence metric*, rather than an experimental estimate of the intermittency. It can be used in a similar way—to define a location at which transition is nearly complete—but its basis is spectral rather than temporal.

The turbulence metric used in this work is the Shannon entropy, also known as the spectral or information entropy. The entropy in this context is not related to thermodynamic entropy, though they share a mathematical framework. The Shannon entropy H^* is defined as

$$H^* = - \sum_{i=1}^N p_i \log_2 p_i, \quad (3.18)$$

where p_i is a given probability distribution. Note that the maximum entropy can be shown to be $H_{\max} = \log_2 N$ when $p_i = 1/N$. The entropy is often normalized by this maximum to yield a metric bounded between zero and one,

$$H = \frac{H^*}{\log_2 N} \in [0, 1]. \quad (3.19)$$

The Shannon entropy can represent the information content in a probability distribution. Thus the entropy is maximized when the distribution is uniform, as no point is any more likely than any other point, and so there is minimal information contained within the distribution. The spectral entropy is widely used in speech processing to determine when a signal contains speech (i.e., information). See for example References [93–95]. Other measures of spectral information content have also been used in this field, for example the Spectral Flatness Measure in Johnston [96].

The application to speech processing hints at its use as a turbulence metric. Turbulence is characterized by a broadband spectrum with pressure fluctuations over a wide range of frequencies. At low frequencies, the turbulence spectrum will be fairly flat, though it will roll off at higher frequency. The metric is formed by applying the Shannon entropy to a section of the normalized power spectrum of the measured pressure signal. If the power spectrum is flat, or nearly so (as it is in turbulent flow), the entropy will be close to one. If there are peaks in the spectrum, the entropy will be lower according to their relative prominence. Note that aside from the trivial case of a zero-valued signal, the minimum entropy will occur when only a single element $p_i = 1$ and the rest are zero.

The entropy metric is derived as follows, using the notation of Reference [93]. First, a low-frequency band is chosen in the power spectrum corresponding to a range where the sensor response should be nearly flat. The spectrum is then normalized as a discrete probability distribution so that its sum is one,

$$P_i = \frac{S_i}{\sum_{i=1}^N S_i}, \quad (3.20)$$

where the S_i are the individual points in the power spectrum over the frequency band of interest. The normalized Shannon entropy of this distribution is then calculated,

$$H = -\frac{1}{\log_2 N} \sum_{i=1}^N P_i \log_2 P_i. \quad (3.21)$$

The value of H is the turbulence metric. When H reaches within some predefined threshold of one (which should be independent of the particulars of the environment), the flow is considered turbulent.

Though this method is simpler than a true intermittency estimate, it is not without flaws. If the flow is at such a low Reynolds number that no instabilities appear in the spectra, it will still appear flat and thus incorrectly identified as turbulent. This could be accounted for by combining H with a measure of the power within the chosen frequency band. In addition, this metric has not eliminated the need for an arbitrary threshold.

Figure 3.9 shows three spectra from different axial stations along the cone, as well as the entropies corresponding to each spectrum. The frequency band used in the calculation is 25–200 kHz, as indicated with the vertical dashed lines. In this case the entropy metric seems to work as designed. However, note that the frequency band includes the large rise in power at low frequencies (which is of unknown origin). This portion was included to “artificially” decrease the entropy for signals with low power in the secondary instability band, which otherwise would have an entropy of close to one.

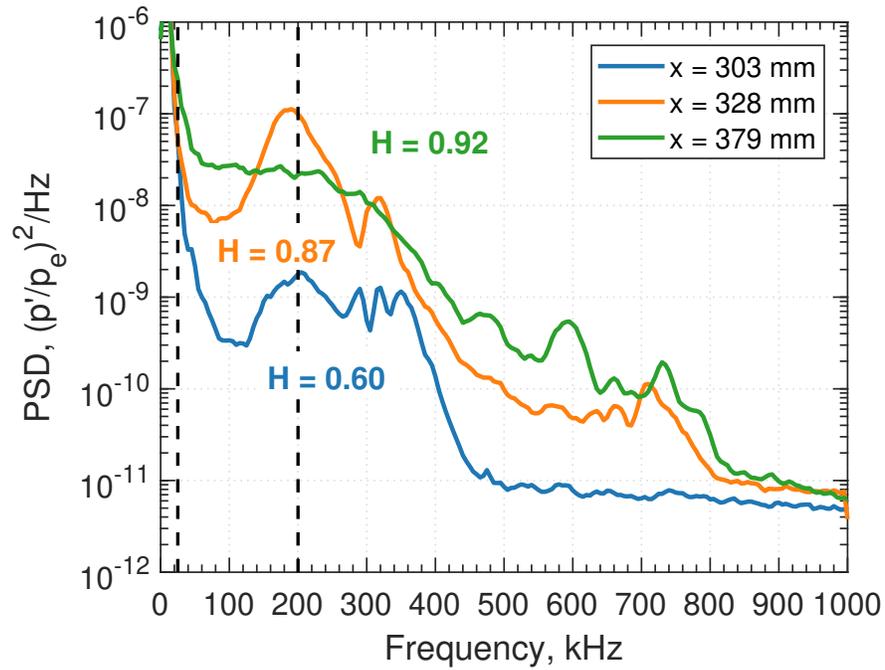


Figure 3.9. An example entropy calculation for three spectra at different axial stations. The spectra clearly show the growth of an instability at 200 kHz and a nearly turbulent flow at 379 mm. The entropy H correspondingly increases from 0.6 at 303 mm to 0.92 at 379 mm. $Re_\infty = 10.4 \times 10^6 \text{ m}^{-1}$.

4. DEVELOPING A CONTROLLED ROUGHNESS FOR THE STUDY OF CROSSFLOW-DOMINATED TRANSITION

The pursuit of a mechanism-based prediction method for crossflow-dominated transition requires close coordination between experimental and computational efforts. In particular, it is necessary to have a controlled method by which to induce the generation of stationary vortices, so that the input disturbance can be accurately modeled. In addition, as Schuele et al. [39] showed at Mach 3.5, it may be possible to use roughness to control crossflow-dominated transition, but doing so requires a highly repeatable method to create the roughness.

Chynoweth's RIM roughness technique [65,66] was a promising new approach to creating a controlled roughness array, using small brass rods press-fit into an aluminum body. These rods can be precisely machined to the desired specifications, which is an improvement over older techniques like the dimpled Torlon roughness used by Ward, Henderson, and Edelman [21, 37, 38].

However, it was not immediately obvious that the technique could be adapted to the study of crossflow on a cone at angle of attack. One problem is the sizing of the required components. The roughness location was determined by the neutral point of the stationary vortices [38, 50, 97], which is very near the nose tip ($x \approx 50.8$ mm). The radius of the roughness insert at this location is only about 6 mm, which makes it extremely difficult to insert closely-spaced roughness rods of a workable diameter. To further complicate matters, the interfaces between the roughness insert and the rest of the model must have as small a step as possible. However, there were several different roughness arrays of interest, so it was not practical to machine the roughness insert together with the nosetip and Upper Frustum (this would require as many nose tips as roughness inserts). It is important to determine if such a step overpowers the discrete roughness elements. Finally, it is also important to verify that the final roughness configuration is tall enough to dominate the production of the stationary vortices, but not so tall as to trip the flow. This chapter describes the process of verifying that the the RIM roughness is suitable for use on a cone at angle of attack.

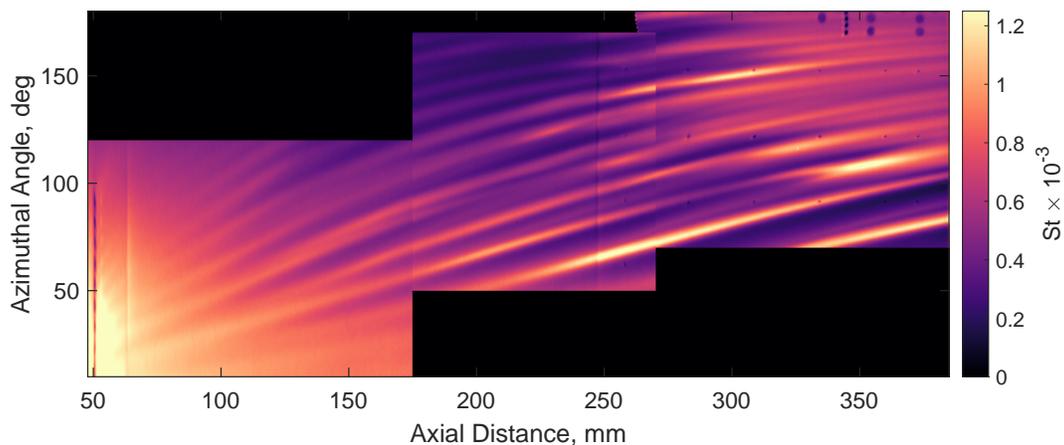
4.1 Demonstrating Control

To verify that the RIM elements were indeed dominating the generation of the stationary vortices, two cases were examined: a baseline case and a case with the roughness insert rotated by about 5° leeward. Because the roughness elements are visible in the forward part of an IR image, a more exact rotation angle can be determined. A profile from the roughness location was extracted from each image and a cross-correlation determined the actual shift. If the streaks in the heat transfer can be shown to follow the rotation of the roughness elements, with minor variation, it can be concluded that the RIM elements are dominating the production of the hot streaks, and therefore the stationary vortices. Keep in mind, however, that when the roughness elements are rotated, they are in a different flow field and thus the streaks they generate will have different growth rates along their paths. It is therefore reasonable to assume that this experiment will not produce an exact match between the amplitudes of the two cases.

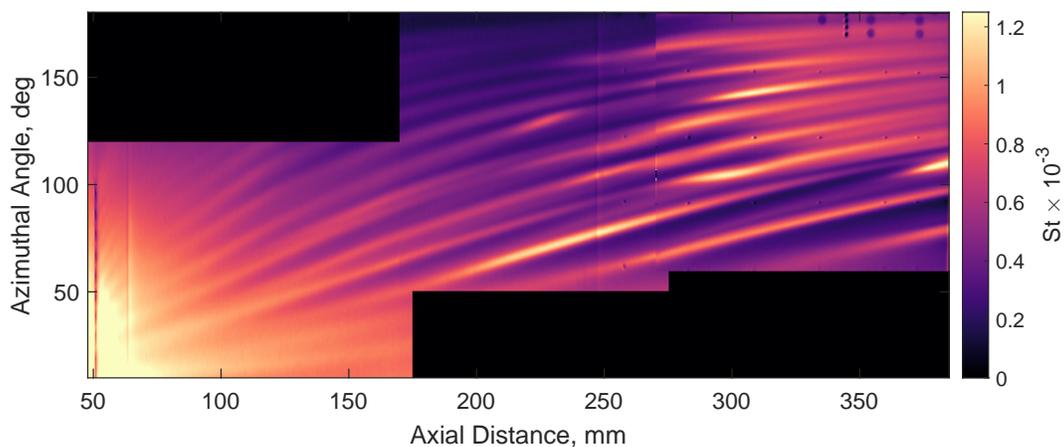
This experiment was conducted twice, once for elements 200 μm (0.008 inch) tall and then again for ones that are 127 μm (0.005 inch) tall. The taller elements are useful for demonstrating the effectiveness of the RIM elements because the stationary vortices have large initial amplitudes and so the signal-to-noise ratio is higher. However, the sensor measurements were made with 127 μm elements, so the experiment was repeated to ensure that the smaller elements were not too small.

4.1.1 200 μm Elements

Figure 4.1 shows the heat transfer images from the baseline and rotated cases. Note that downstream the pattern of vortices looks completely different. Using only information from the aft part of the cone (260 mm to 380 mm) it is in fact quite difficult, if not impossible, to determine if the streaks rotated with the elements. However, because the streaks can be easily followed upstream, it is sufficient to show that the streaks rotate with the roughness elements at an upstream location, where the differences in amplitude are not as dramatic.



(a) RIM elements in the baseline configuration. Images from 3 runs, left to right: 1403, 1402, 1401.



(b) RIM elements rotated approximately 5° leeward. Images from 3 runs, left to right: 1404, 1405, 1406.

Figure 4.1. Full heat transfer maps for the baseline and rotated element cases. $Re_\infty = 11 \times 10^6 \text{ m}^{-1}$. Run conditions are provided in Appendix A.

Before examining the shift in the streaks at an upstream location, it is important to discuss the spreading of the streamlines in this fully three-dimensional flow. Figure 4.2 shows the computational vortex paths superimposed on the paths of the streaks from Figure 4.1(a), traced using the algorithm detailed in Appendix E.2. The computations were provided by Alex Moyes from the EPIC code at Texas A&M University (TAMU); more information about their computation can be found in References [40, 98]. At higher azimuthal angles and farther aft on the cone, the

computations and the experiment match fairly well. At lower azimuthal angles, however, the agreement is poor. It is unknown if this is a computational discrepancy or if there are issues with the unwrapping of the cone at these angles. For the present experiments there were no fiducial marks on the upstream portion of the cone.

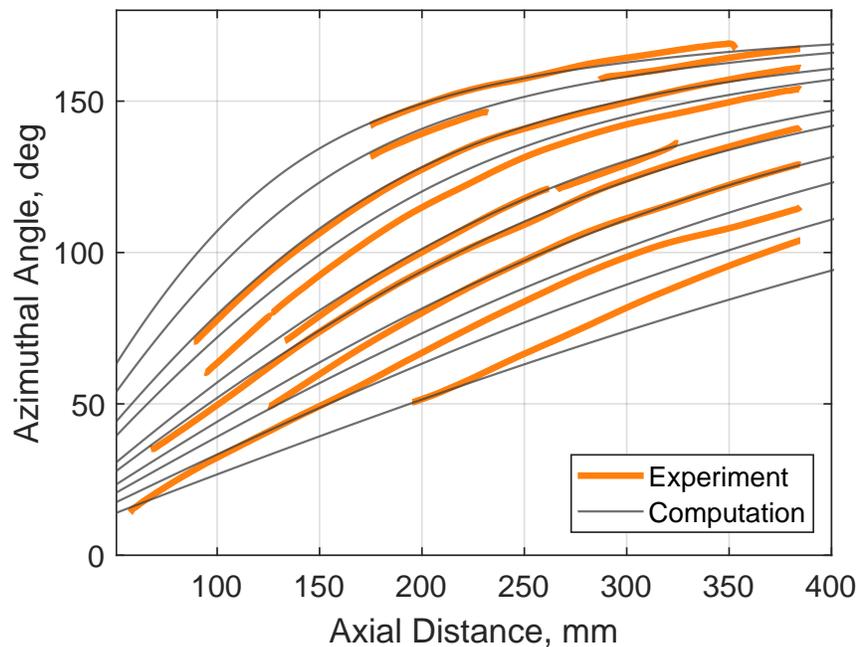


Figure 4.2. Computational vortex paths from Moyes et al. [40] superimposed on the experimental streaks in Figure 4.1(a).

However, the comparison is still instructive. It is immediately evident that near the roughness and at lower azimuthal angles the vortex paths spread rapidly, whereas at higher azimuthal angles they stay fairly parallel. Because of this large spreading, a shift in the location of the roughness elements would be expected to have a correspondingly larger shift downstream; i.e. unlike in a two-dimensional flow, the downstream shift of the vortices will not be exactly the same as the upstream shift of the elements. The amount of expected shift can be calculated from the

computational paths. If we assume that the azimuthal location θ_1 of the vortices at some x_1 can be determined as a smooth function of their upstream locations θ_0 at x_0 , then we can say

$$\theta_1 = f(\theta_0), \quad (4.1)$$

$$\hat{\theta}_1 = f(\hat{\theta}_0), \quad (4.2)$$

where the hat denotes the condition where the roughness elements have been shifted, i.e. $\hat{\theta}_0 - \theta_0 = \Delta$, the leeward shift of the roughness element in degrees. For small rotations, the hat condition can be Taylor expanded,

$$\hat{\theta}_1 \approx f(\theta_0) + f'(\theta_0)\Delta. \quad (4.3)$$

The expected shift at the downstream x_1 station is then

$$\hat{\theta}_1 - \theta_1 \approx f'(\theta_0)\Delta. \quad (4.4)$$

The function $f(\theta_0)$ was interpolated from the TAMU vortex paths and fit with a fifth-order polynomial. The derivative can then be calculated analytically. Figure 4.3 shows f' . The dashed line indicates that for a 2D flow, the downstream and upstream rotations would be identical. As was estimated from the vortex paths, at high azimuthal angles, near 140° , f' is near 1, so the upstream and downstream rotations will be about the same. For lower angles, however, the TAMU data estimates that the downstream rotation of the streaks could be as much as 2-3 times the upstream rotation. Such a large shift multiplier for the downstream location of the streaks will almost surely affect their development, which explains the obvious differences in amplitude in Figure 4.1.

Figure 4.4 shows a cross-section of the heat transfer at 150 mm from the nosetip for the baseline and rotated cases. The actual rotation of the roughness, about 3.6° , was determined by cross-correlating slices of the IR images at the roughness location for each case. Figure 4.4(a) also provides the heat transfer from the same location for an unshifted roughness element from a run which was conducted 2 months earlier (the dashed line). The two unshifted cases match quite well, which speaks to the repeatability of the experiments and the heat-transfer reduction method.

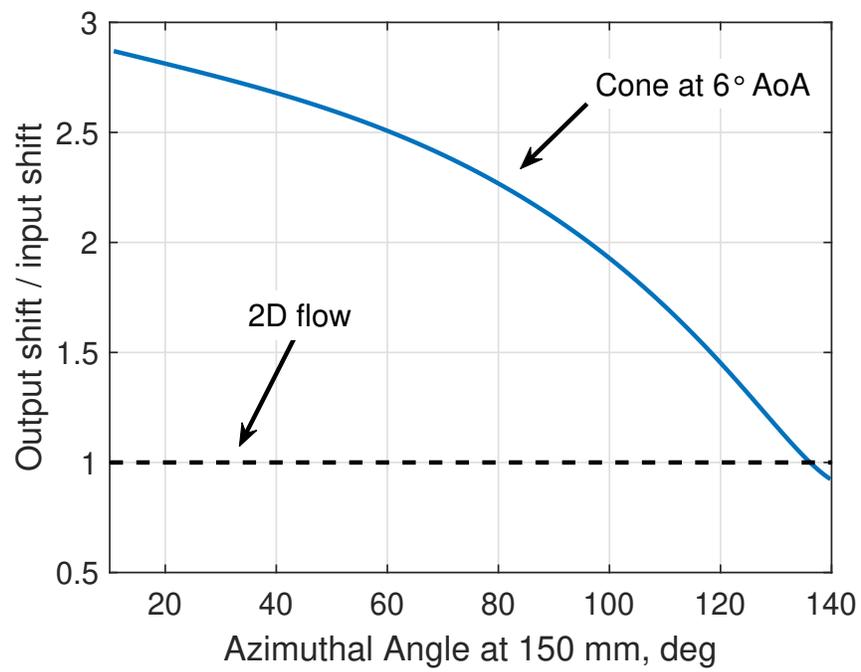


Figure 4.3. The relative downstream shift of the streaks for a unit upstream rotation of the roughness elements.

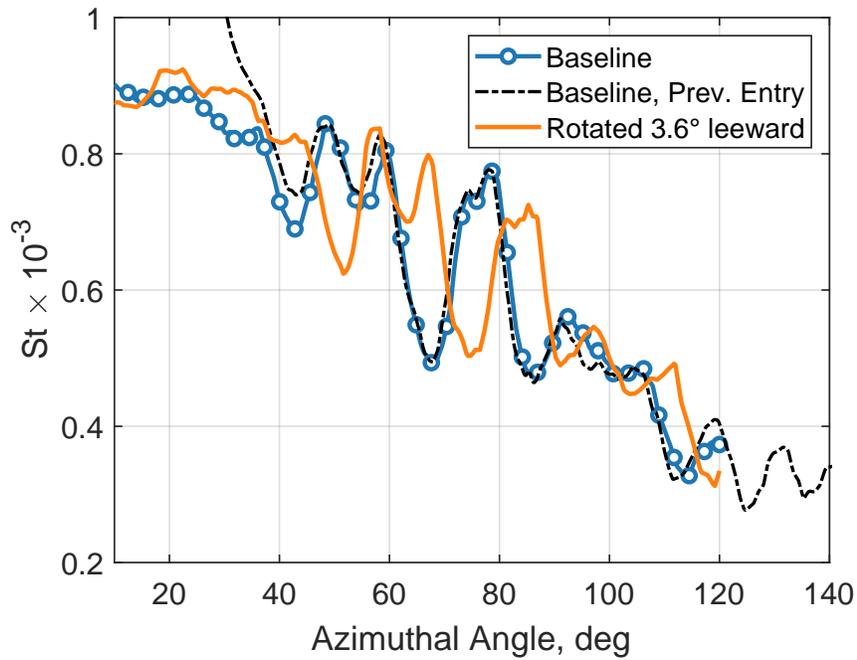
In Figure 4.4(b), the rotated data has been artificially shifted windward by the rotation angle of 3.6° . Note that at higher azimuthal angles, where $f'(\theta_1)$ is closer to 1, the two cases lie on top of one another. However, for lower azimuthal angles the actual shift in the streaks is evidently larger than the input of 3.6° , as expected from Figure 4.2. Despite the lack of agreement in peak location, however, the peak amplitudes of the two cases are fairly similar. The differences can likely be accounted for by the different amplitude histories of the two cases because of the change in the roughness locations.

Using the computational vortex paths, the azimuthal locations of the heat-transfer slices at 150 mm can be interpolated to their predicted positions at the axial location of the roughness using computational vortex paths from the EPIC code provided by Moyes et al. [19, 98]. This is shown in Figure 4.5. When the rotated case is artificially shifted windward by 3.6° , the locations of all the peaks almost exactly match the baseline case. This further reinforces the conclusion that the differences in peak location shown in Figure 4.4 are due to the streamline spreading.

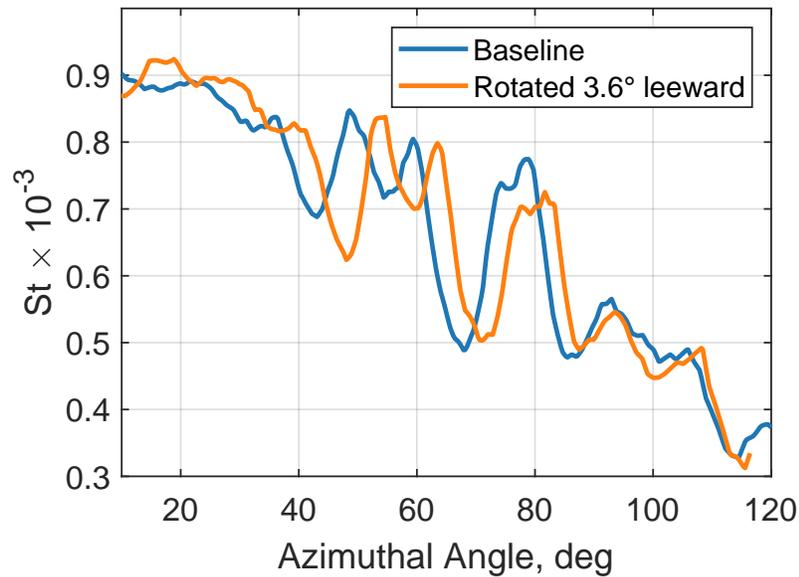
4.1.2 127 μm Elements

The previous experiment was conducted again using the 127 μm elements at the Reynolds number used for the sensor measurements, $\text{Re}_\infty \approx 10.5 \times 10^6 \text{ m}^{-1}$. Figure 4.6(a) shows azimuthal slices at 120 mm for two different rotations: two runs at the baseline rotation and one with the roughness rotated by 2.2° leeward. The rotation magnitude was determined by cross-correlating slices of the heat transfer image at the roughness location. Figure 4.6(b) shows the same slices of the data, but with two modifications to the azimuthal coordinates. First, the azimuthal coordinates at 120 mm were interpolated to their predicted positions at the roughness location ($x = 50.8 \text{ mm}$). Then the coordinates of the slices were shifted by the opposite of their known rotation (i.e., 2.2° windward). All of the peaks line up fairly well, indicating that even for smaller elements at a lower Reynolds number, the RIM elements are still dominating the generation of the stationary vortices.

It is also worthwhile to examine the locations of the downstream streaks with respect to the upstream roughness elements. Figure 4.7 shows the heat transfer perturbation at $x = 120 \text{ mm}$ for the baseline case, compared to the measured height profile of the elements at a $x = 50.8 \text{ mm}$. The perturbation is defined here as the heat transfer band-pass filtered between 9 and 90 waves per



(a) A direct comparison of the heat transfer at 150 mm for the two rotation cases. The dashed line is from a run with an unshifted roughness insert from the previous entry, 2 months beforehand.



(b) The same comparison, with the rotated case artificially shifted windward by 3.6°.

Figure 4.4. Comparison of heat transfer slices at 150 mm for the baseline and rotated cases. $Re_\infty = 11 \times 10^6 \text{ m}^{-1}$. The run conditions are provided in Appendix A. Baseline: Run 1403; Baseline, Prev. Entry: Run 1321; Rotated: Run 1404.

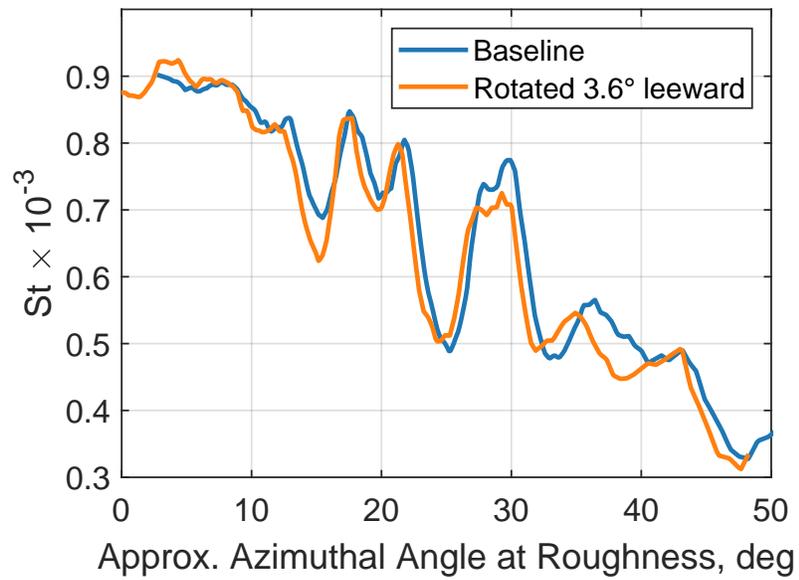
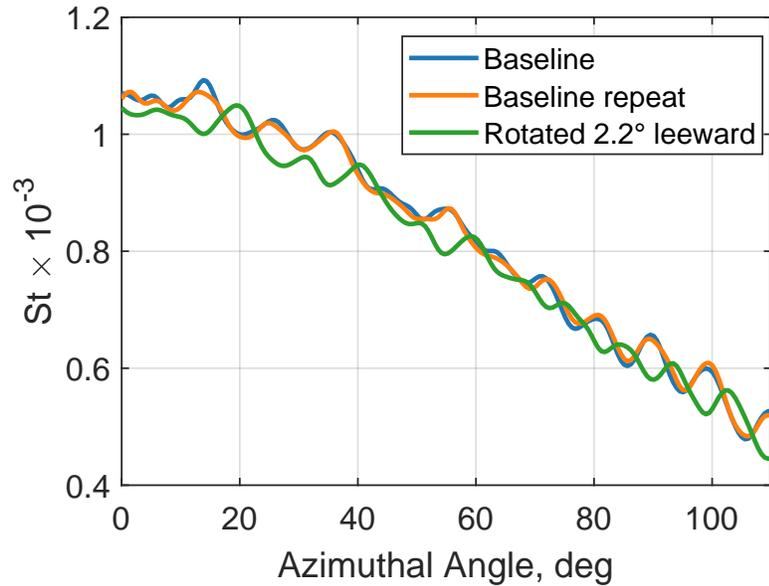


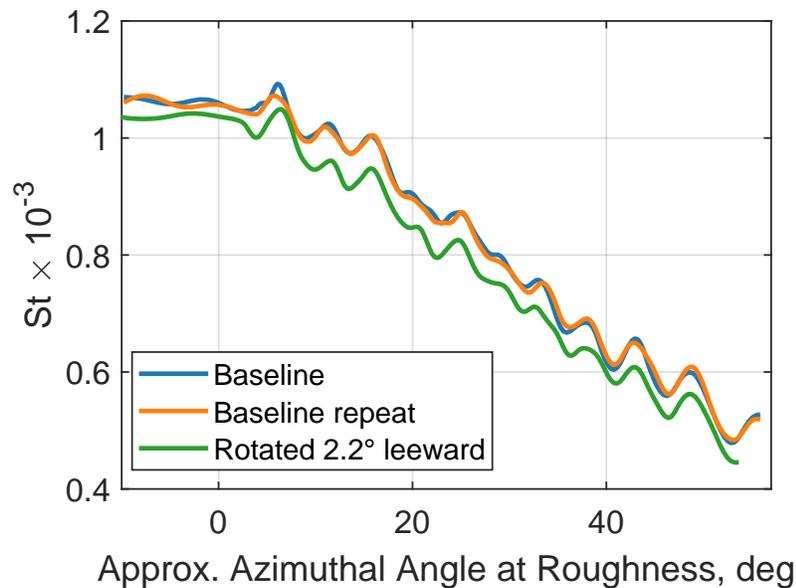
Figure 4.5. A comparison between the baseline and rotated cases at 150 mm. The x-axis has been interpolated to be the estimated azimuthal angle at the roughness axial location, given the mapping provided by TAMU. The rotated case has been artificially shifted windward by 3.6°.

circumference. The heat transfer azimuthal positions have been interpolated to the roughness location as before.

It is apparent from Figure 4.7 that each roughness element generates two streamwise vortices, with the leeward vortex slightly stronger than the windward one. This agrees qualitatively with low-speed DNS of roughness in a 3D flowfield [99], where the vortex is stronger on the lee side of an element. It is also interesting to note that the relative amplitudes of the vortices appear to be correlated with the roughness height; see for instance the low heat-transfer perturbation behind the element at 30° and the larger perturbation behind the next most leeward element.



(a) Slices at 120 mm.



(b) Slices at 120 mm interpolated to the azimuthal location of the streaks at the roughness. The rotated slice has been shifted by the opposite measured rotation of the roughness.

Figure 4.6. Comparison of slices at $x = 120$ mm for the baseline case and a case with the roughness rotated 2.2° leeward. Initial baseline case Run 1701 at $Re_\infty = 11 \times 10^6 \text{ m}^{-1}$, other two cases at $Re_\infty = 10.5 \times 10^6 \text{ m}^{-1}$ (Baseline repeat: Run 1703; Rotated: Run 1704).

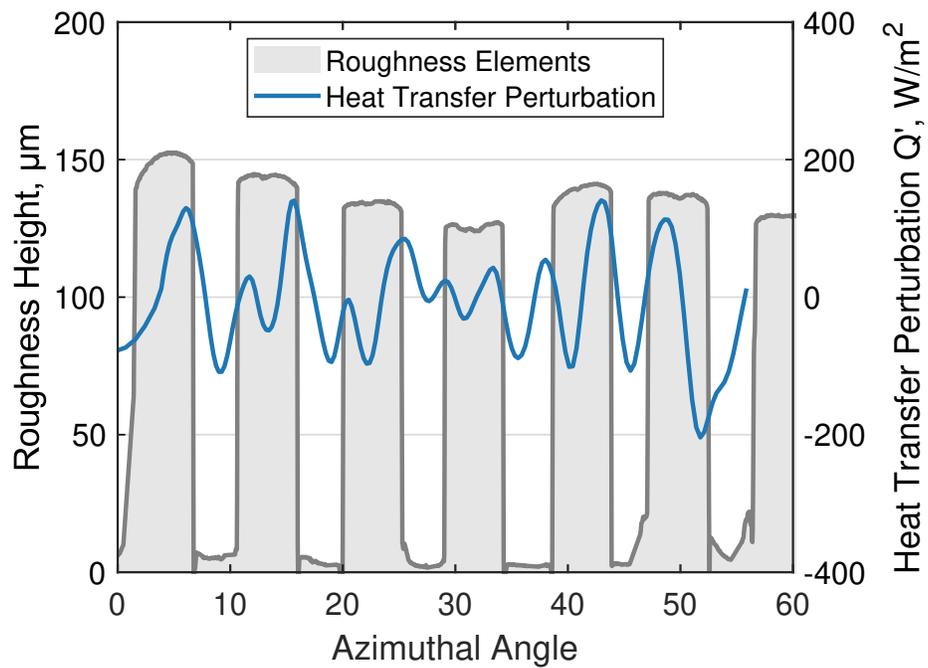


Figure 4.7. The heat transfer perturbation at 120 mm interpolated to the roughness position (50.8 mm) plotted over the measured roughness heights. Each element produces a pair of streaks, with the leeward streak at a higher amplitude. The relative amplitudes of the streaks seem to be correlated with the individual element heights. $Re_\infty \approx 10.5 \times 10^6 \text{ m}^{-1}$.

4.2 Roughness Reynolds Number

A common metric for the relative magnitude of surface roughness is the roughness Reynolds number, $Re_{kk} = U_k k / \nu_k$, where the subscript k indicates the velocity and viscosity are evaluated at the height of the roughness in an undisturbed flow. Schneider [100] provides a good overview of historical correlations of transition location with various roughness parameters, including Re_{kk} .

The Re_{kk} can be used to define a “critical” height, at which the roughness first accelerates transition compared to the same geometry without roughness. However, for the present work the roughness is *intended* to affect transition, that is to induce the growth of stationary crossflow vortices so their breakdown can be studied. Depending on the wavenumber of the input roughness, transition with roughness will almost always occur earlier than the smooth cone, by design. Instead, one must look for a small roughness height at which the qualitative behavior of the flow changes. This approach was used successfully by Chynoweth [66] to determine the appropriate roughness height on a flared cone model.

To that end, several roughness inserts were fabricated from PEEK with nominal element heights from 50.8 μm to 635 μm , and otherwise identical specifications to the RIM insert listed in Table 2.5. It was hoped that there would be a clear difference in breakdown behavior between two of these heights. Unfortunately, that was not the case.

Figure 4.8 shows heat-transfer images of the aft portion of the cone using four different nominal roughness heights: 50.8 μm , 127 μm , 203 μm , and 254 μm . A qualitative transition front is indicated by the white line in each image. This front is the contour at which the low-pass-filtered heat-transfer data reaches twice the laminar CFD prediction; this line is purely qualitative for comparison between the different heights.

Two things are immediately evident from the series of images in Figure 4.8. First, increasing the roughness height moves transition upstream, which is expected. In particular, individual vortex structures can be identified as moving upstream; see for example the hot streak going through (370 mm, 110°) for the 127 μm data (Figure 4.8(b)). Second, the qualitative behavior of the flow is the same for every height. In all cases there appears to be streak merging (which is examined further in Chapter 5) and an uneven transition front. In addition, the paths of the streaks

are the same for each case. This evidence suggests that the same mechanism is driving breakdown for roughness up to at least $k = 254 \mu\text{m}$.

For roughness heights above $254 \mu\text{m}$, transition occurs forward of the PEEK Sensor Frustum. To determine if even taller roughness elements could change the character of transition, an Upper Frustum and roughness inserts were fabricated from PEEK with roughness elements up to $k = 635 \mu\text{m}$. The forward portion of the cone, including the roughness elements themselves, was imaged using a 50 mm lens. Figure 4.9 shows a series of these images for roughness heights from $152 \mu\text{m}$ to $635 \mu\text{m}$. Note that the images in Figure 4.9 are very far forward on the model, so the local radius of curvature is small and therefore spanwise conduction effects may be important—especially for the strongly modulated heat-transfer caused by the roughness elements. Therefore the heat flux in these images was calculated using the 2D Fourier method described in Section 3.6 and derived in Appendix D.

Again, there are no sudden changes in the stationary vortex behavior as the height of the roughness elements is increased. The $635 \mu\text{m}$ roughness exhibit strong splitting and merging close to the elements, but it is unclear whether this is indicative of a change in the transition mechanism. For the $152 \mu\text{m}$ case, the streaks are difficult to see at all.

The only evidence of transition caused by non-crossflow mechanisms are a series of images taken using the $406 \mu\text{m}$ elements (the tallest elements seem to trip the flow almost immediately after the roughness). Figure 4.10 shows three runs at increasing unit Reynolds numbers, and therefore increasing Re_{kk} . At the lowest unit Reynolds number, Figure 4.10(a), the hot streaks in the imaged region look no different than usual, except for some broadening and streamwise waviness in the lowest streak. However, upon increasing Re_{∞} , a large turbulent wedge is visible. At the highest Re_{∞} , the wedge starts at around 120 mm, or only about 215 roughness diameters downstream from the elements. Farther downstream, the wedge has broadened to roughly 40° of azimuth, and appears to be affecting the neighboring streaks. This large turbulent wedge does not appear in the breakdown of the streaks for smaller roughness elements (cf. Figure 4.8). In addition, the wedge is reminiscent of features that appear in roughness-dominated breakdown in other situations [101, 102].

The Re_{kk} of the roughness that caused the turbulent wedge is approximately $Re_{kk} \approx 6900$. Figure 4.11 shows the estimated Re_{kk} of the $127 \mu\text{m}$ elements used for the measurements in Chapter

5, at $Re_\infty = 11.6 \text{ m}^{-1}$. The maximum Re_{kk} occurs near the wind ray, as expected, and is about $Re_{k,\text{max}} = 200$ for $Re_\infty = 11.6 \times 10^6 \text{ m}^{-1}$. This level is more than an order of magnitude lower than the roughness that caused the turbulent wedge in Figure 4.10, so it is believed that the $127 \mu\text{m}$ roughness induces stationary crossflow vortices but does not lead to transition via non-crossflow mechanisms. Note that the pressure fluctuation measurements reported in Chapter 5 were taken at $Re_\infty = 10.5 \times 10^6 \text{ m}^{-1}$, so the Re_{kk} was less than the maximum shown in Figure 4.11.

Though they are not directly comparable to crossflow-dominated boundary layers, other studies have found that the critical roughness is often larger than $Re_{kk} = 200$. See, for example, Braslow [103] who showed $Re_{k,\text{crit}} > 10,000$ at Mach 6 on a flat plate, and Gronvall et al. [104], whose computations indicated a marked change in the flowfield between $Re_{kk} = 250$ and $Re_{kk} = 1450$ on a sharp cone at 0° angle of attack. Wheaton and Schneider [105] studied near-critical roughness in the nozzle-wall boundary layer of the BAM6QT, and showed that $Re_{kk} = 322$ did not cause transition within the measurement range, though they note that the smooth-wall case also did not transition within this range, so it is impossible to identify the “true” critical Re_{kk} . Casper et al. [101] found the critical roughness on a sharp cone to be around $Re_{kk} = 100$, which is the lowest critical roughness level for hypersonic flow known to the author. However, none of these studies investigated the effect of a 3D boundary layer. In addition, the Re_{kk} is just one of many roughness parameters that can affect transition (including, for example, the roughness width and spacing). However, while the picture is incomplete the available data indicate that the chosen roughness height, $127 \mu\text{m}$, is tall enough to induce large stationary crossflow vortices but is not so tall as to directly trip the boundary layer.

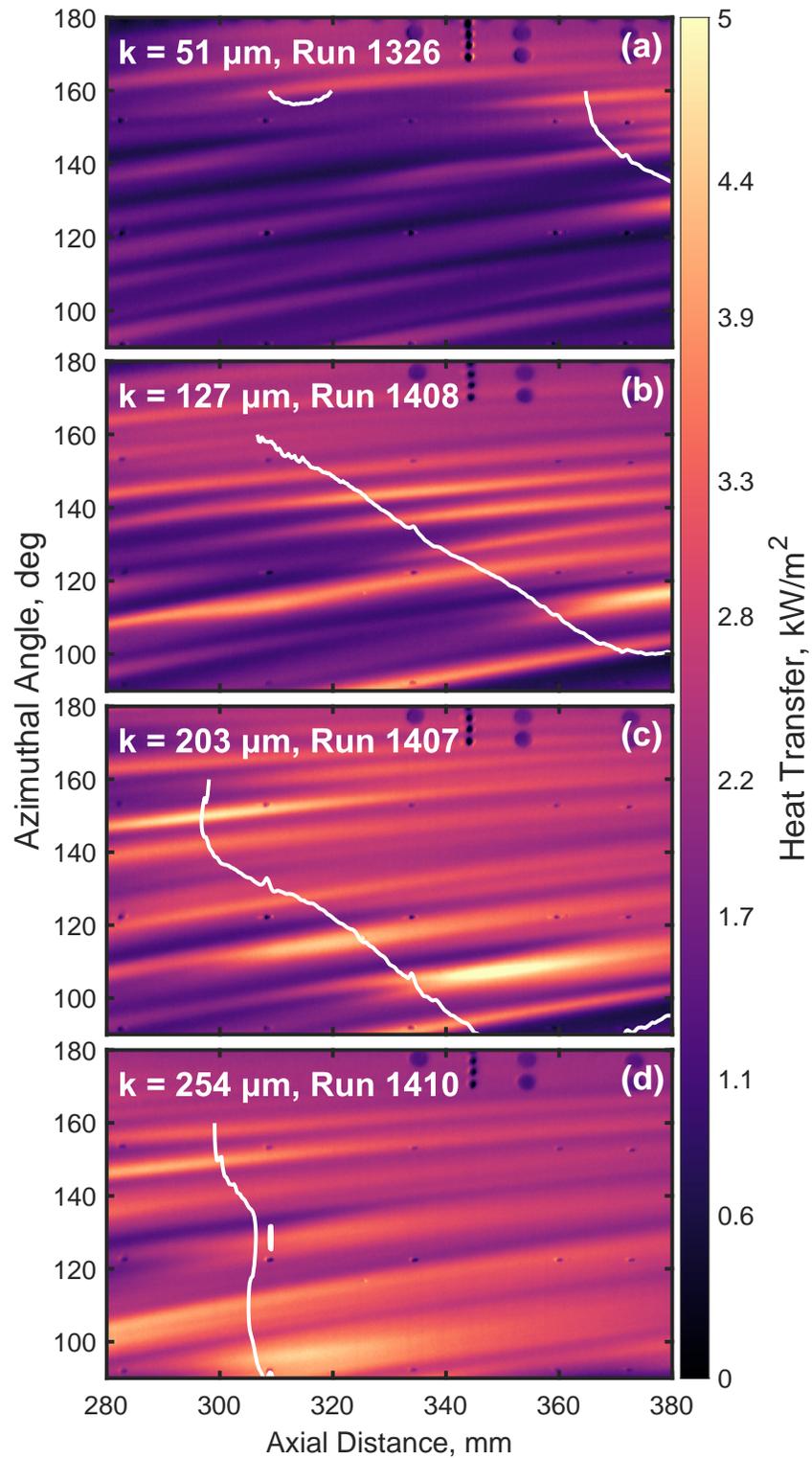


Figure 4.8. Heat-transfer images of the aft portion of the cone with different upstream roughness heights. The white line is an average, qualitative transition front, described in the text. $Re_\infty \approx 11.1 \times 10^6 \text{ m}^{-1}$.

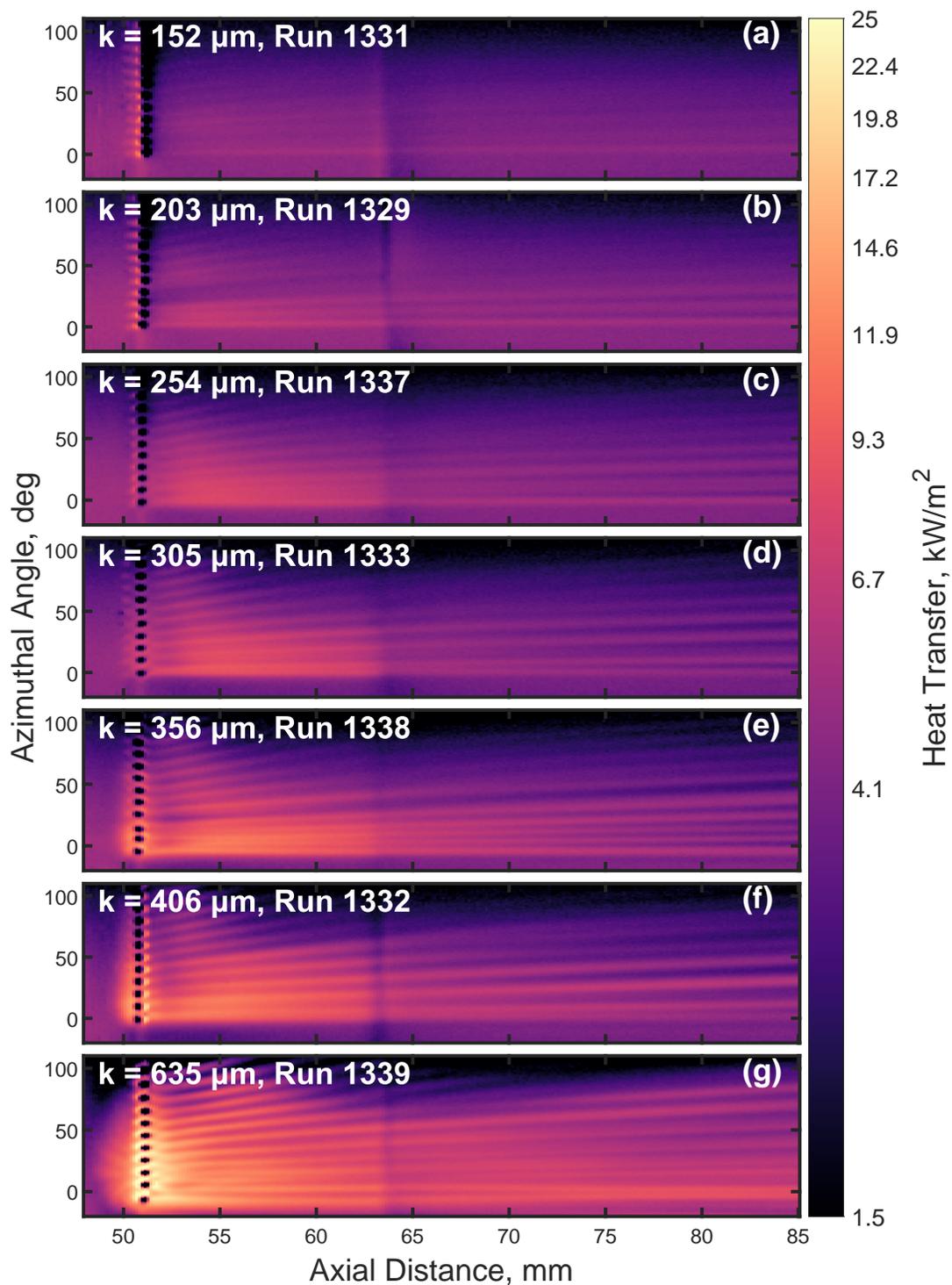


Figure 4.9. Images of the heat flux near the roughness elements (the dark circles at 50.8 mm) with increasing heights k . $Re_\infty = 11.1 \times 10^6 \text{ m}^{-1}$.

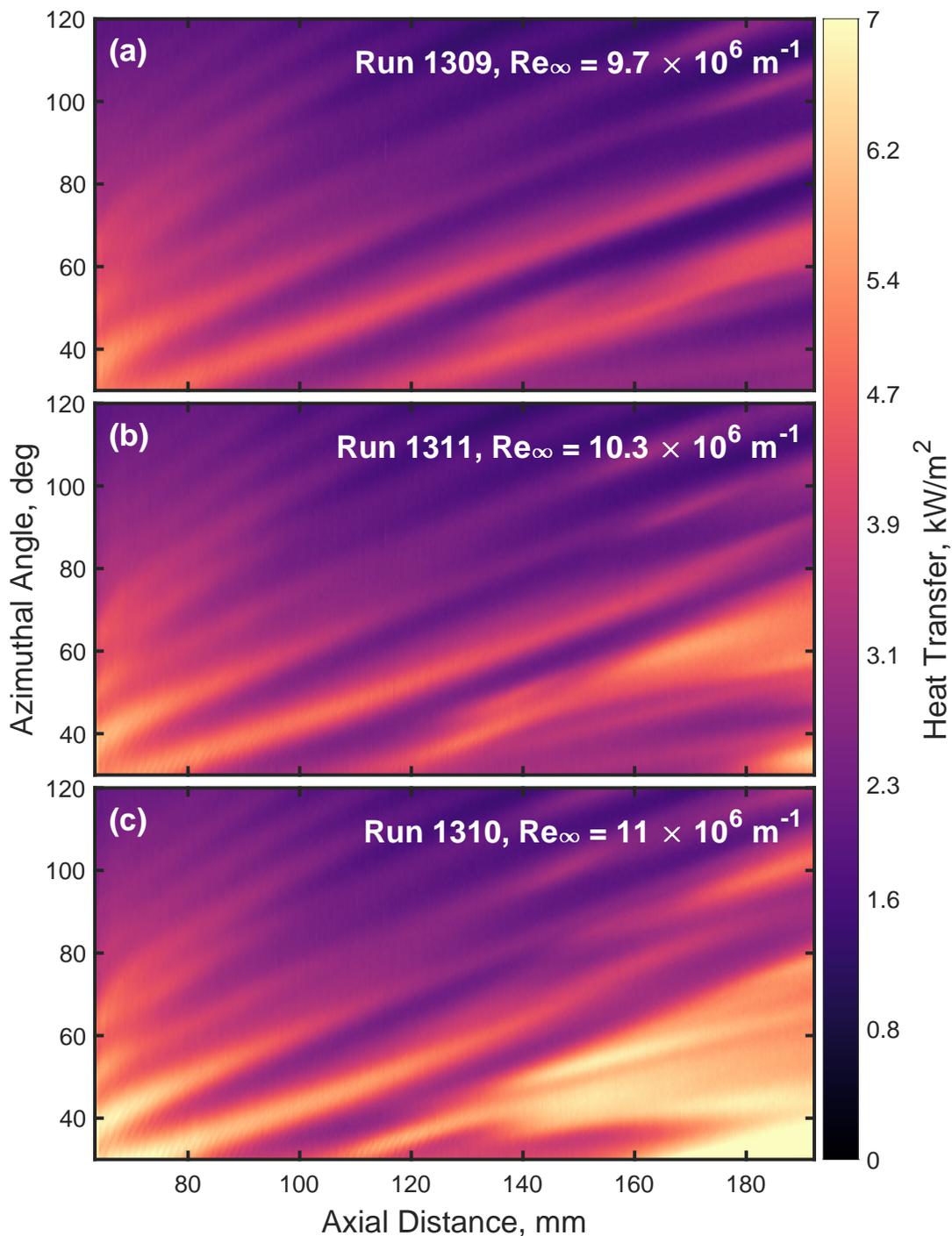


Figure 4.10. Images of the heat flux on the forward portion of the cone using roughness elements with heights nominally $k = 406 \mu\text{m}$. Images from three increasing unit Reynolds numbers are shown, illustrating how the turbulent wedge near the wind ray moves forward as the roughness Re_{kk} increases. At the maximum unit Reynolds number, the Re_{kk} of the roughness causing the wedge is $Re_{kk} \approx 6900$.

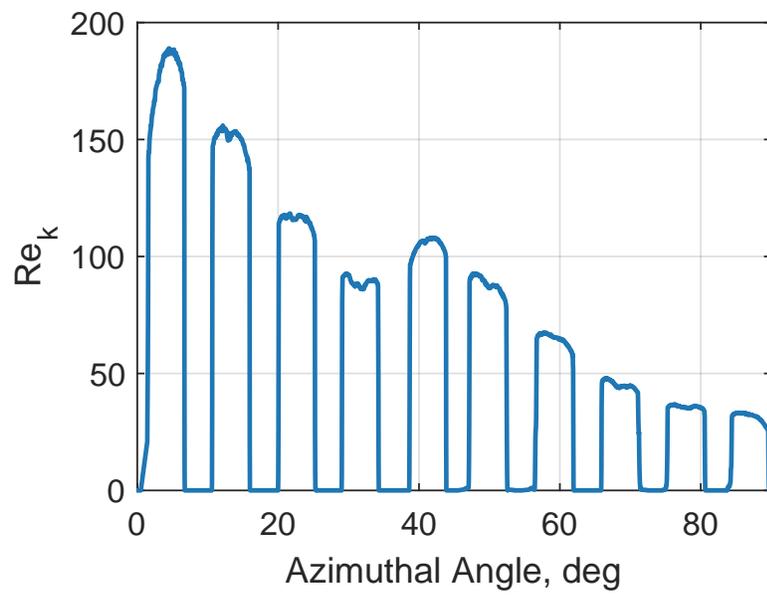


Figure 4.11. CFD-estimated Re_{kk} along the center of the nominally $127\ \mu\text{m}$ RIM roughness elements, $Re_\infty = 11.6 \times 10^6\ \text{m}^{-1}$.

4.3 Roughness Spacing

Whitehead [106] demonstrated that at low speeds, if discrete roughness elements are placed too closely together they no longer act as individual vortex generators. Given that the explicit purpose of the RIM roughness is to induce the stationary crossflow vortices, it is important to verify that the roughness is not too closely spaced. This is not obviously true, as even with a 9° spacing, the elements are only 1.8 diameters apart because they are so far forward on the model. Unfortunately there is no good guideline for what the appropriate roughness spacing should be. Saric estimates $d/\lambda \geq 0.5$ based on his experience at low speeds (personal communication cited in Schuele et al. [39]), where d is the element diameter and λ is the wavelength of the forced stationary mode. It is not clear whether this can be applied to hypersonic flows. Note that according to computations, the naturally most-amplified wavenumber is around 220 at $x = 50.8$ mm [107] (a spacing of 1.6°), but this is not currently feasible with the RIM technique.

Figure 4.12 shows heat-transfer images of the Sensor Frustum for four different roughness cases: a nominally smooth insert (a) with $k_{\text{RMS}} \approx 1.3 \mu\text{m}$, a two-dimensional roughness strip (b) with $k \approx 254 \mu\text{m}$, RIM roughness with elements 9° apart (c), and RIM roughness with elements 18° apart (d). Both RIM inserts had nominal element heights of $127 \mu\text{m}$. A two-dimensional roughness strip is the limiting case of very closely spaced rods. Thus if the RIM elements are too closely spaced it is expected that the downstream pattern will look similar to the 2D case. The results are quite interesting. The 2D roughness shows downstream behavior very similar to the smooth case, but at higher amplitudes: thin, uniform streaks at a high wavenumber. This indicates that the 2D roughness is exciting the naturally most-amplified wavenumber, which makes sense given the lack of a geometrically “preferred” wavenumber. However, the RIM roughness produces a much different pattern downstream. The streaks are not very uniform, showing splitting, merging, and a very jagged transition front. In addition, the streaks are generally fatter and at a slightly lower wavenumber. It is also clear that increasing the spacing between the elements to 18° (Figure 4.12(d)) does not change this general pattern. It therefore appears that the elements in the RIM roughness with 9° spacing are adequately far apart.

4.4 Direct Metal Laser Sintering Roughness Inserts

The RIM technique is extremely useful, but it is by no means perfect. This is especially true for the very small rods and spacing required on the cone at angle of attack, unlike its original usage on a flared cone [65]. It is desirable to have roughness elements at a very high wavenumber, both to force the naturally most-amplified waves and to study the possibility of boundary-layer control [39, 53].

A potential approach to this problem is to use Direct Metal Laser Sintering (DMLS), a form of 3D printing, to additively manufacture the roughness inserts from aluminum or steel. It was hoped that a computer-controlled process would be more uniform than a hand-machined insert, and would be capable of producing smaller element diameters and spacings. Unfortunately, the DMLS process still has a ways to go before it is useful in this capacity. The test pieces procured from the vendor were out of round by several millimeters, and the RMS surface roughness was quite high. Figure 4.13 shows a height map of the test piece taken with the Zygo Zegage white-light interferometer. The 3D-printing process leads to pronounced ridges over the entire surface, and the elements themselves are not as cylindrical as desired. The typical polishing process for DMLS parts involves tumbling them with an abrasive, but this is not practical for these inserts because that would also abrade away the roughness elements themselves. When the technology is more mature it may be worthwhile investigating DMLS again, but as it stands the process cannot match the capabilities of a skilled machinist for this application.

4.5 Summary of RIM Development

The experimental campaign described in this chapter was intended to demonstrate that the RIM roughness is suitable for the controlled study of crossflow transition. Measurements of the heat flux downstream of the roughness elements show that when the roughness is rotated by a small amount the heat-transfer footprint of the stationary vortices follow the rotation of the roughness elements. This indicates that the RIM elements are inducing the stationary vortices that are measured downstream.

In addition, it is important to determine that the selected roughness height, $127\ \mu\text{m}$, is not so tall as to trip the boundary layer directly. The experimental evidence suggests that such a

condition has been met. The only observable evidence of bypass transition was measured using elements nearly 3.5 times larger than the selected roughness height (and more than 30 times the Re_{kk} of the 127 μm elements). Thus though it is difficult to say for certain that the selected element size is not too tall, the balance of the evidence supports the conclusion that the RIM roughness with 127 μm elements are adequate for the controlled study of crossflow-dominated transition.

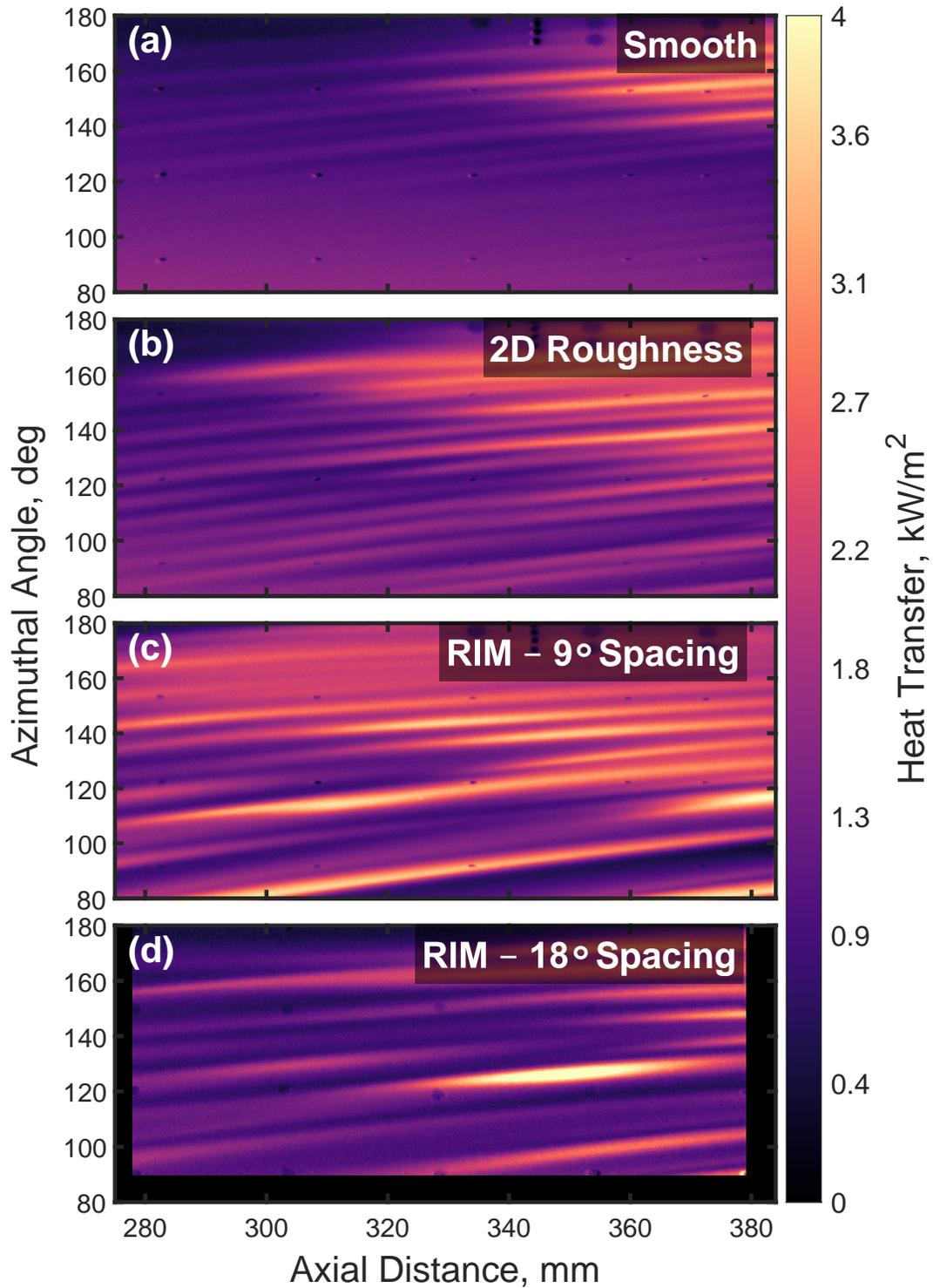


Figure 4.12. Comparison of different wavenumbers of the input roughness on the downstream vortex pattern. For (a)–(c), $Re_\infty = 10.8 \times 10^6 \text{ m}^{-1}$; for (d) $Re_\infty = 9.8 \times 10^6 \text{ m}^{-1}$. (a) Run 1415, $k_{\text{RMS}} \approx 1.3 \text{ } \mu\text{m}$; (b) Run 1417, $k \approx 254 \text{ } \mu\text{m}$; (c) Run 1408, $k \approx 127 \text{ } \mu\text{m}$; (d) Run 936, $k \approx 127 \text{ } \mu\text{m}$.

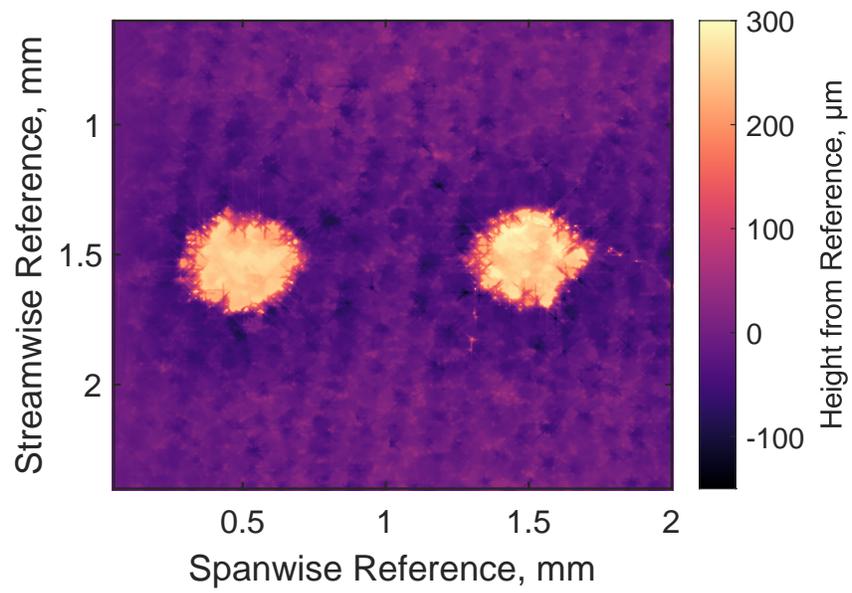


Figure 4.13. A height map of two of the test DMLS roughness elements.

5. MEASUREMENTS USING DISCRETE ROUGHNESS ELEMENTS

This chapter describes measurements made using a RIM roughness insert, with rods nominally $k = 127 \mu\text{m}$ (0.005 inch) in height, $D = 560 \mu\text{m}$ (0.022 inch) in diameter, and 9° apart. The previous chapter demonstrated that rods of this configuration are sufficiently tall to dominate the production of stationary crossflow vortices downstream; that they are not so tall as to lead to turbulence via bypass mechanisms; and that they are spaced far enough apart to act as individual vortex generators. In this chapter, before examining the nonlinear breakdown of the boundary layer, the repeatability of the primary and secondary instabilities is verified. Then, measurements of the nonlinear growth and breakdown of the instabilities are presented for a baseline case and a case with the roughness elements rotated slightly leeward. The RIM roughness parameters are well controlled and characterized, so it is hoped that the data in this chapter can be used as a validation case for future computations. The run conditions for the runs presented in this chapter are provided in Appendix A.

5.1 Repeatability

The measurements presented in this work are typically composites of data from several runs, and so it is vital to ensure that the data are repeatable from run to run. The stationary crossflow vortices and the secondary instability are measured via two independent methods, so the repeatability of both techniques must be analyzed. First, the infrared measurements of the stationary crossflow vortices are examined in Section 5.1.1. Section 5.1.2 demonstrates the repeatability of the secondary instability measurements using PCB pressure sensors.

5.1.1 Primary Instability Repeatability

For measurements with the Modular Cone, several runs are performed in a series with the Sensor Frustum rotated to different positions. However, during each of these measurements the

roughness insert is in the same location. Thus, if the stationary vortex behavior is repeatable, the infrared images of the Sensor Frustum should nominally be identical, save for the positions of the sensors. In this way the Modular Cone provides a built-in check of repeatability. A total of 50 runs were performed with the same RIM insert: 37 with the roughness in the baseline position and 13 with the roughness rotated leeward (see Section 5.4). Table 5.1 provides the mean and standard deviation of the initial stagnation pressure and temperature over all 50 runs.

Figure 5.1 shows slices of the IR-derived heat transfer at $x = 344$ mm for all 37 runs in Entry 16. These runs were performed over several days between December 21, 2018 and January 3, 2019. The azimuthal location of the sensors ranged from 112° to 144° , with seven positions repeated. The cross-section from each of these runs is displayed as a gray line on the plot, and the mean of all the runs is the thick blue line. The upper plot shows the standard deviation of the heat transfer at each point as a percentage of the mean at that point. It is clear that the heat transfer downstream is quite repeatable. The largest deviations, 10–20 %, occur on either side of the largest peaks in the heat transfer. This is likely due to small deviations in the azimuthal location of the peaks for each slice, which lead to larger amplitude deviations because the spanwise gradient in heat transfer is large on either side of the peak. It is unclear if the spanwise deviations in peak location are the result of true variation in the stationary vortices or just uncertainty in the mapping from pixels to points on the cone. However, regardless of its source, it is evident from Figure 5.1 that any such deviations are quite small.

Between Entry 16 and Entry 17, the model was uninstalled and disassembled, and the camera was relocated and recalibrated. In addition, IR measurements in Entry 17 used a completely different Sensor Frustum (the Mk. I PEEK frustum, see Reference [108]). Figure 5.2 compares the mean heat transfer from Entry 16 to the heat transfer from a run in Entry 17. The error bars are the standard deviation of the runs in Entry 16 at those points (see Figure 5.1). The agreement is excellent, with the most variation happening at the peaks in heat transfer.

Table 5.1. Variation in flow conditions for the 37 runs used in Figure 5.1.

	Stag. Pressure, psia	Stag. Temperature, K
Mean	135.4	419.6
Std. Dev.	1.6	1.5

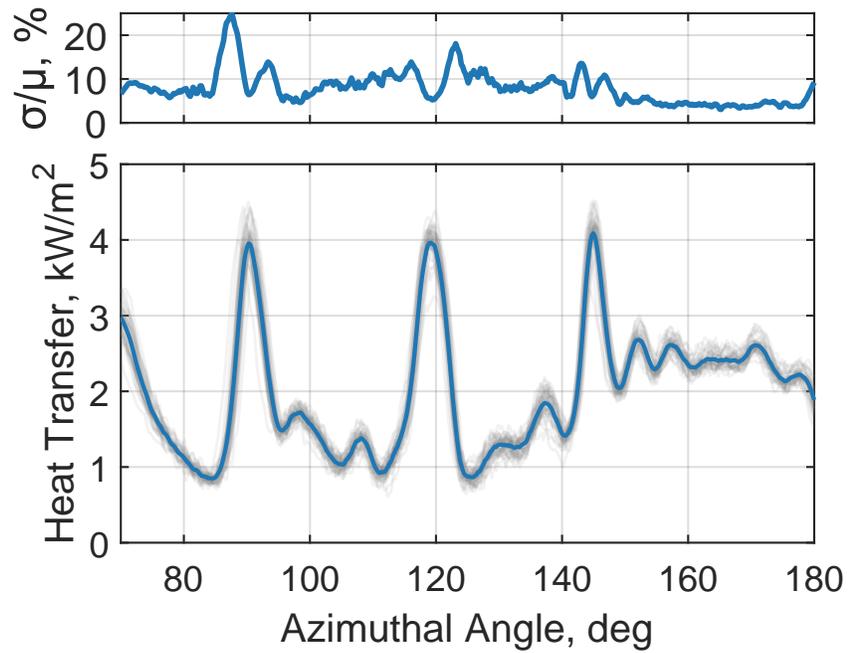


Figure 5.1. Slices of the IR-derived heat transfer at $x = 344$ mm for 37 runs. The azimuthal location of the sensors ranged from 112° to 144° . The gray lines are the individual slices and the blue line is the mean. The upper plot shows the standard deviation as a percentage of the local mean. $Re_\infty = 10.5 \times 10^6 \text{ m}^{-1}$.

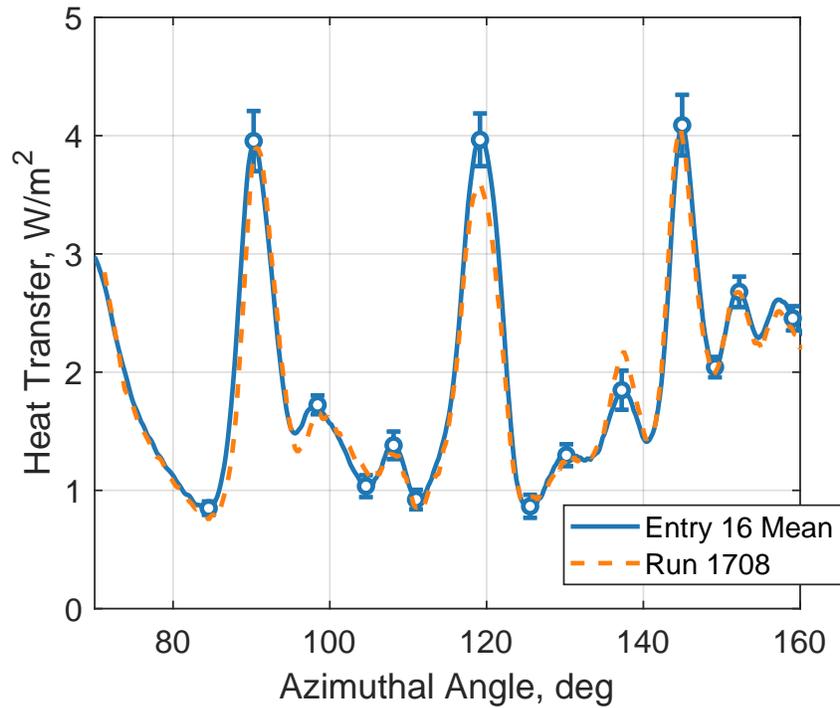


Figure 5.2. Comparison of the heat transfer at $x = 344$ mm between Entry 16 and Entry 17. The model was completely uninstalled and disassembled, and the camera was repositioned and recalibrated between these measurements. $Re_\infty = 10.5 \times 10^6 \text{ m}^{-1}$.

5.1.2 Secondary Instability Repeatability

Because the secondary instability measurements are local, unlike the global heat transfer measurement from infrared imaging, there are actually two repeatability studies that must be performed with the PCB sensors. The first is to examine measurement repeatability using the same sensor in the same location, but after several re-orientations of the Sensor Frustum. This information validates the run-to-run repeatability of the secondary instability measurements and can aid in understanding the effect of uncertainty in the azimuthal orientation of the Sensor Frustum. Figure 5.3 shows PSDs of three sensors from two different runs nominally at the same azimuthal angle. The measurements were separated by 40 runs and almost two weeks of testing. It is immediately evident that the secondary instability measurements are highly repeatable on the same sensor. This is further illustrated by the bicoherence for PCB 8, provided in Figure 5.4;

even the nonlinearities can be repeatably measured using the same sensor. Here, not only is the location of maximum bicoherence the same for both runs, the actual magnitude of the bicoherence is also nearly the same.

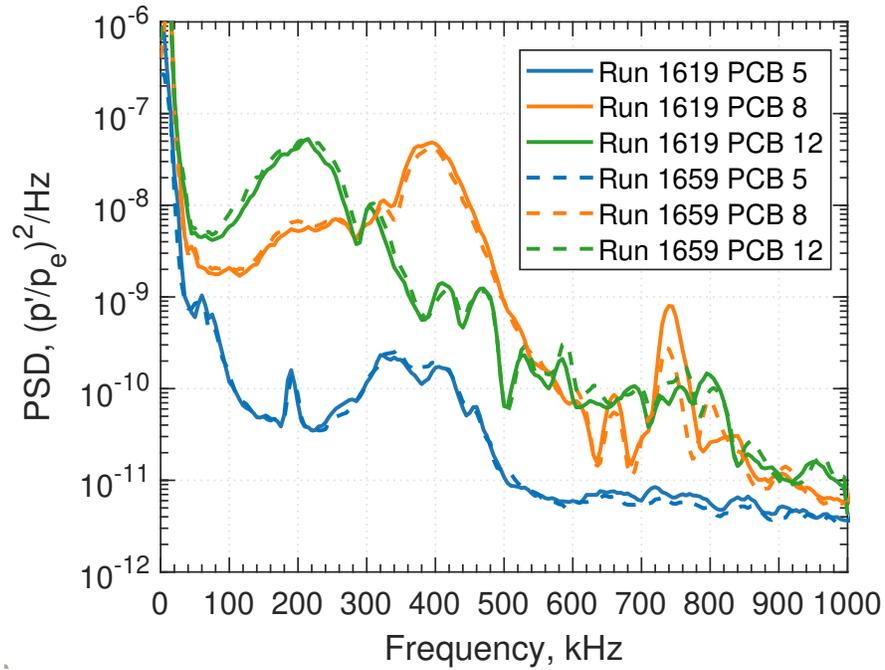


Figure 5.3. PSDs of measurements at 122° for three PCBs, several runs apart. $Re_\infty = 10.5 \times 10^6 \text{ m}^{-1}$.

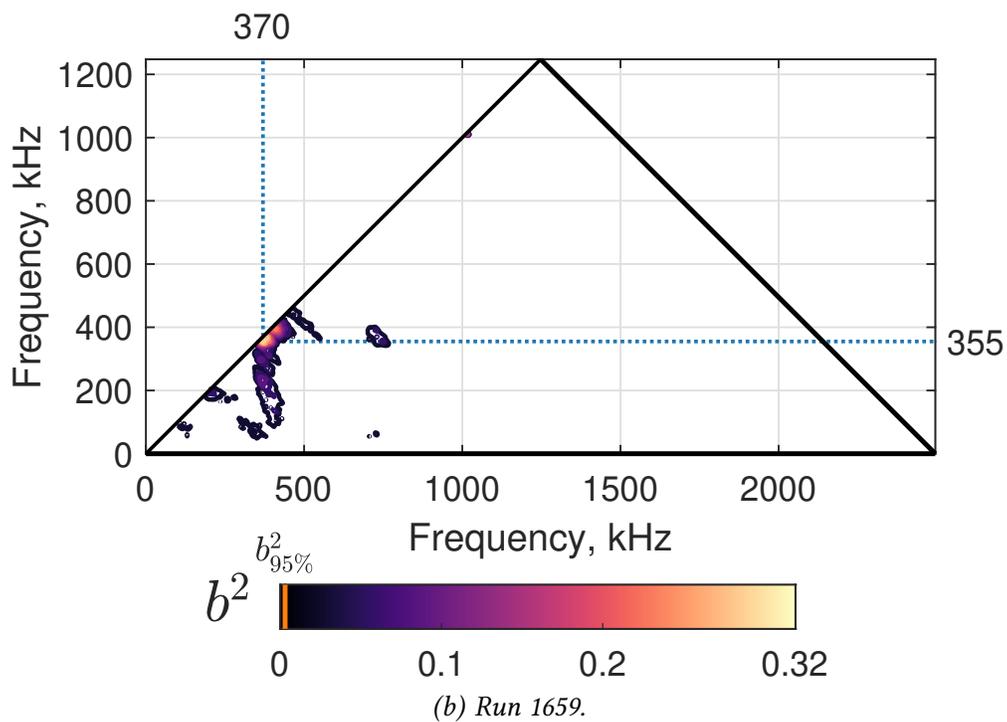
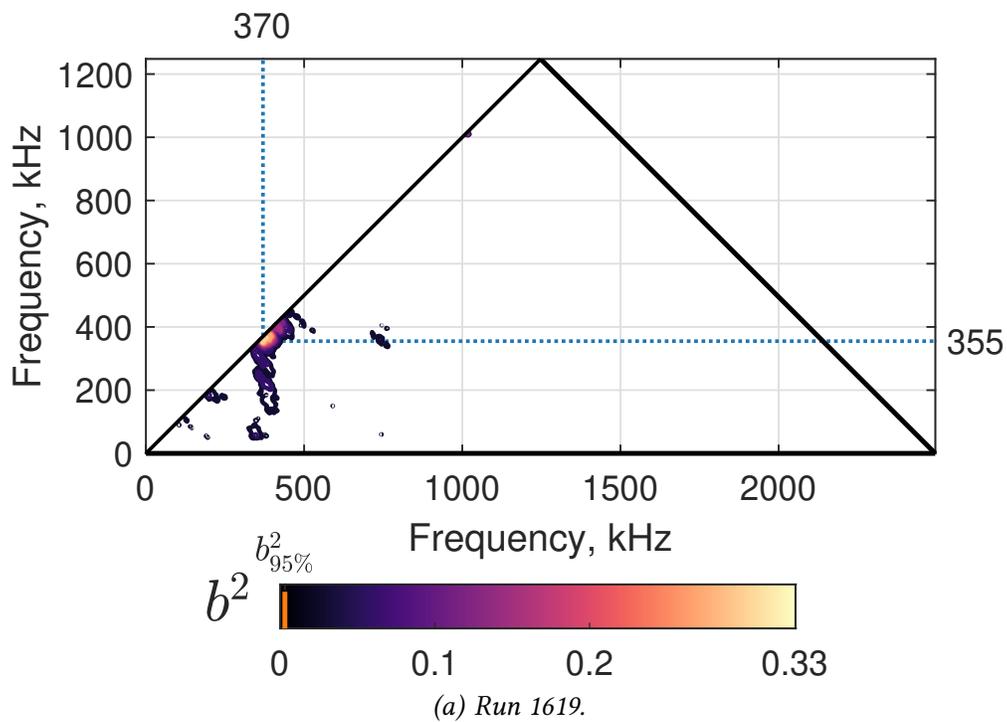


Figure 5.4. The bicoherence for PCB 8 ($x = 303$ mm) at 122° , 40 runs apart. These correspond to the two PSDs for PCB 8 in Figure 5.3. $Re_\infty = 10.5 \times 10^6 \text{ m}^{-1}$.

The second repeatability study makes use of the “repeatability pairs” of PCBs on the Modular Cone. As shown in Figure 2.4, there are three pairs of PCBs at the same axial stations but offset by 6° . Table 5.2 provides a list of the PCB repeatability pairs and their axial locations. To further demonstrate repeatability, measurements can be made with the three PCBs on the main sensor ray (13, 16, and 19), then the Sensor Frustum can be rotated by 6° and the same measurements can be repeated using the offset PCBs (21, 22, and 23). Figure 5.5 shows such a comparison at $\theta = 112^\circ$. At first glance, the results do not appear to match. However, on closer inspection, the peaks associated with the secondary instability do repeat quite well. For PCBs 13 and 21, the peaks at 275 kHz are very similar, as are the peaks at 150 kHz for PCBs 16 and 22. The differences within the repeatability pairs appear to be due to sensor resonances, as indicated in the figure, and to higher-frequency activity of an unknown origin.

Figure 5.6 shows the spectra from PCBs 13 and 21 for six azimuthal locations. A dashed gray line is drawn in each subfigure to show the second mode estimate f_{CFD} , derived from the CFD solution as described in Section 3.5. Over the 9° azimuthal range of the measurements in Figure 5.6, the second mode estimate changes by about 20 kHz. Note that the measured peak frequency varies around this estimate, as exemplified by the spectra from 132° to 138° . As will be examined later, the peak frequency seems to be correlated with the local boundary-layer thickness modulation caused by the stationary crossflow vortices, hinting that the measured instability is likely a modulated second mode as opposed to a true secondary instability.

It can be seen from Figure 5.6 that in general the repeatability is quite good at lower frequencies, even using different sensors. At higher frequencies the spectra have more deviation, but this could be due to differences in the sensor resonance characteristics. However, note that the ordinate scale is logarithmic, so the deviations at higher frequencies cause integrated amplitude differences on the order of 10 to 100 times smaller than the amplitude of the largest peak at the lower frequencies.

Given the wide azimuthal range of the measurements, examining all the spectra in detail is impractical. Instead, the pressure fluctuation amplitudes were calculated for each PCB in low and high-frequency bands. The low-frequency band is between 50 kHz and f_{CFD} , the estimated second mode frequency. The high-frequency band is between f_{CFD} and 1 MHz.

Figure 5.7 shows the resulting amplitudes for each PCB repeatability pair over the measured range of azimuthal angles. In general, the amplitude repeatability is excellent. For PCBs 13 and

21 (Figure 5.7(a)), the only point of disagreement is around 116° , at the location of the maximum amplitude in the high frequency band. The discrepancy at this location seems to be due to different levels of response near the first harmonic of the measured instability. Note that in Figures 5.7(b) and (c), the high-frequency amplitudes are much higher for the offset PCBs (22 and 23) at low azimuthal angles below about 120° , the result of the sensor resonances of the offset PCBs. All of the spectra from each repeatability pair have been provided in Appendix B, if the reader would like to examine detailed comparisons.

Table 5.2. Axial stations of the PCB repeatability pairs.

PCB Pair	Axial Loc. x , mm
13 & 21	335
16 & 22	354
19 & 23	373

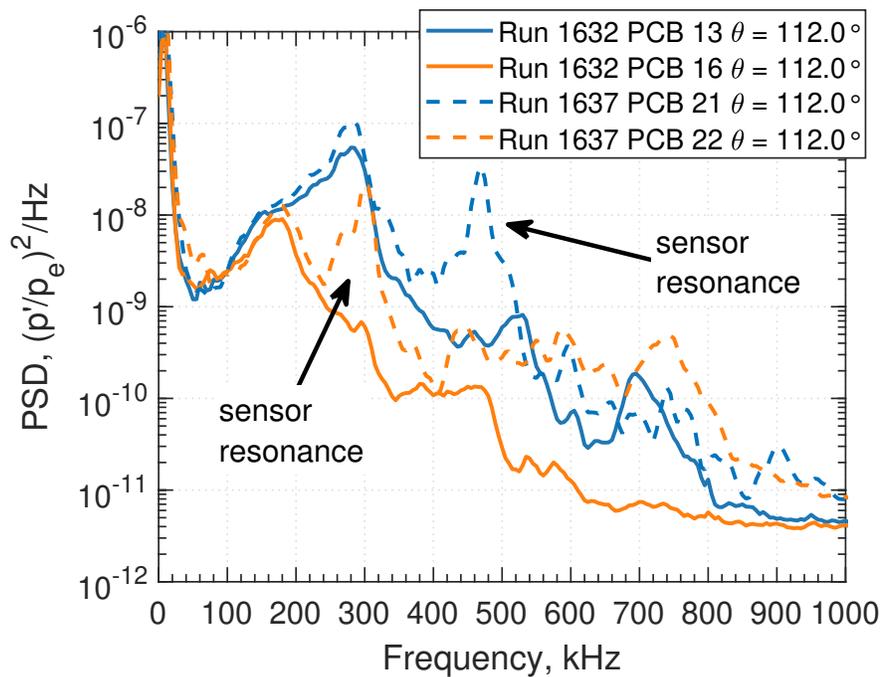


Figure 5.5. PSDs for measurements at the same azimuthal and axial locations, with different sensors. Note that PCB 21 and 22 exhibit large sensor resonances which obscure the repeatability at higher frequencies. $Re_\infty = 10.5 \times 10^6 \text{ m}^{-1}$.

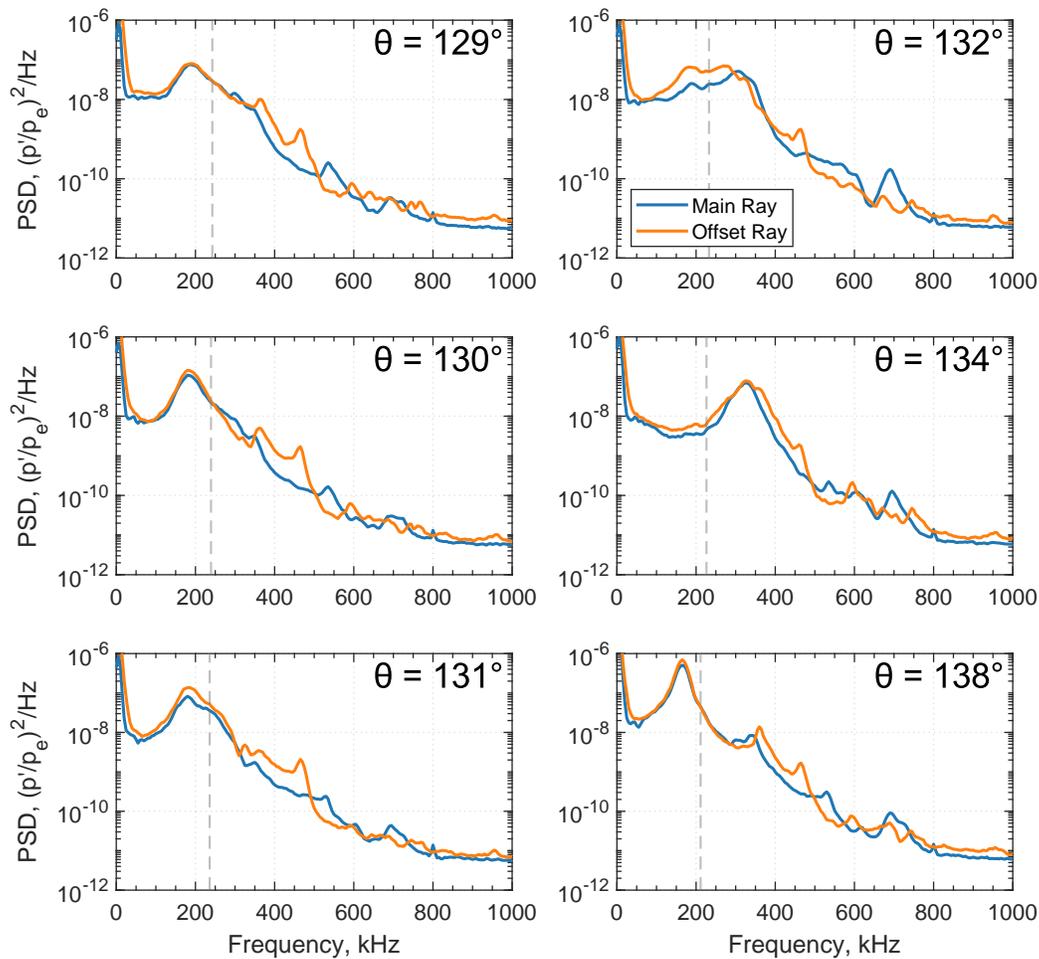


Figure 5.6. Several PSDs for PCBs 13 and 21 at the same axial and azimuthal stations. The dashed gray line is the estimated frequency of the second mode from the CFD solution. $Re_\infty = 10.5 \times 10^6 \text{ m}^{-1}$. See Appendix A for run numbers and conditions.

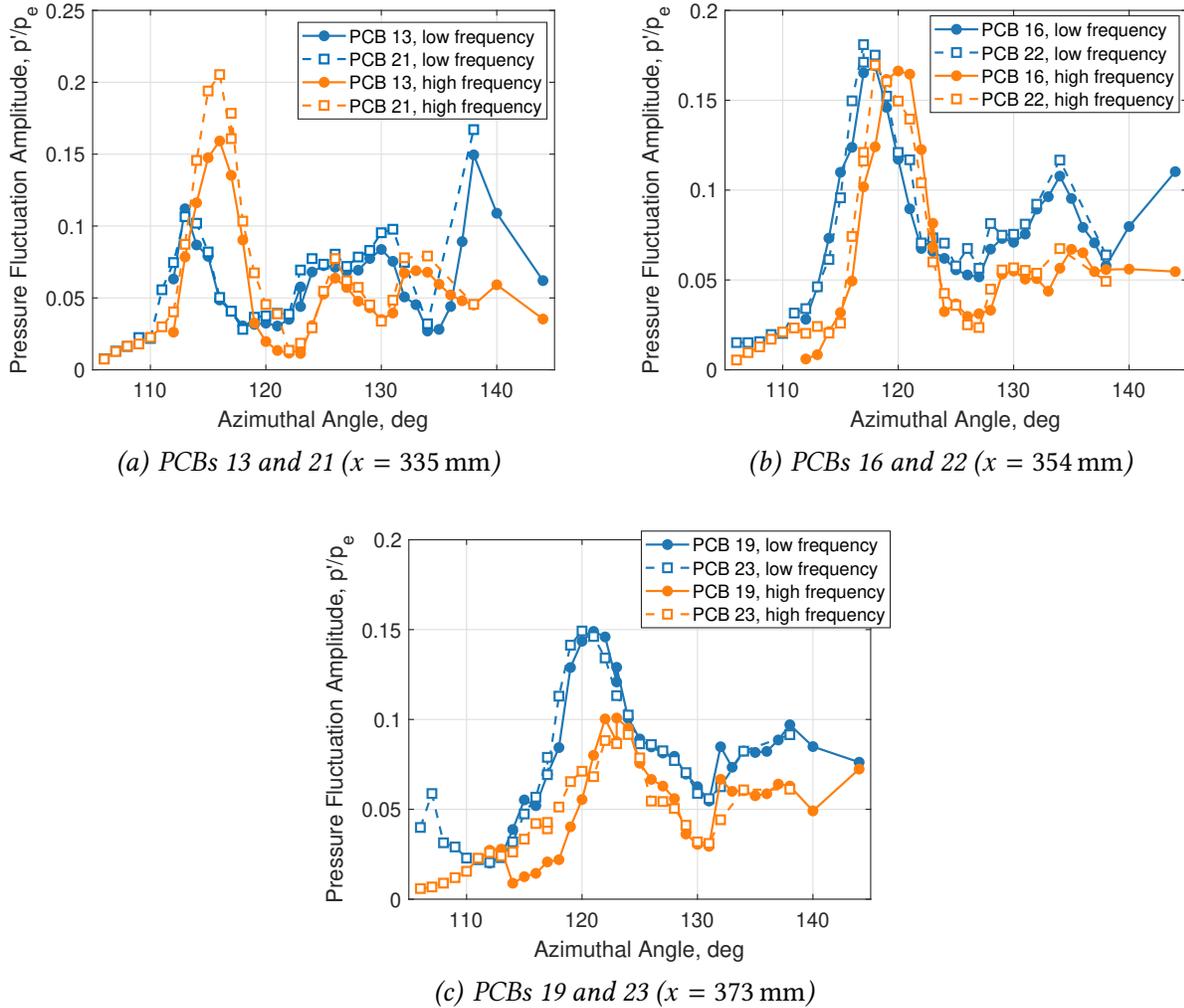


Figure 5.7. Pressure fluctuation amplitudes for the three repeatability pairs. The low and high frequency bands are $50-f_{\text{CFD}}$ kHz and $f_{\text{CFD}}-1000$ kHz, respectively, where f_{CFD} is the local second mode frequency estimated using the laminar CFD solution as described in Section 3.5. $Re_{\infty} = 10.5 \times 10^6 \text{ m}^{-1}$.

5.2 Primary Instability Measurements

The primary instability of the crossflow boundary layer manifests as either stationary or traveling vortices. Different techniques must be used to measure each form. The stationary crossflow vortices result in streaks of high heat flux on the cone's surface, so they are easily measured using a global imaging method like IR or TSP. The traveling vortices appear as low-frequency instabilities at stationary sensing locations, so they can be measured with surface pressure sensors. Unfortunately PCB sensors often do not have adequate response characteristics in the frequency range of the traveling waves, so Kulite sensors are more appropriate.

Though the topic of interest for the present experiments is the nonlinear breakdown of the secondary instabilities, it is important to have a detailed understanding of the primary instability behavior for two reasons. First, the secondary instabilities owe their existence (or their destabilized form) to the presence of the stationary vortices, so accurate computations of the secondary instabilities need to first match the behavior of the primary instabilities. Second, as will be discussed further in the next chapter, the role of the traveling vortices in transition and specifically the nature of their interaction with the secondary instabilities is poorly understood, so measurements of these waves will inform better physical models of the crossflow transition process.

5.2.1 Stationary Crossflow Vortices

Figure 5.8(a) shows three IR images, from three separate runs, stitched together to give a map of the surface heating over nearly the entire length of the model. The images have been shifted slightly to account for uncertainty in the optical calibration of the camera.

There are clear discontinuities in heat flux between the two upstream images. The heat flux is affected not only by the unit Reynolds number, but also the surface temperature, which increases through the run. In addition, the uncertainty in the heat flux calculation could vary with time during the run. Thus while the images are nominally taken at the same Reynolds number, $Re_\infty = 11.2 \times 10^6 \text{ m}^{-1}$, they are from different times during the run, and so the actual levels of heat flux do not match up exactly between each run. Figure 5.8(b) shows the Stanton number of the three images, which is a non-dimensional heat flux that takes the surface temperature variation

into account, among other factors. In this case the Stanton number normalization reduces the heat-flux discontinuities between two runs.

Figure 5.8(c) shows the heat flux perturbation Q' , which is the dimensional heat flux band-pass filtered between 18 and 180 waves per circumference. The perturbation map gives a better sense of the growth of the stationary vortices because it removes the effect of changes in the mean flow over the cone. In this view the streaks are quite prominent, as is their irregularity. Several of the streaks show hot-cold-hot heating patterns reminiscent of results on a flared cone [66]. The general pattern of the perturbation is qualitatively similar to computations (see, for instance, Figure 1(a) from Choudhari et al. [20]).

The full map of the heat flux reveals a number of interesting structures in the development of the stationary vortices that were not evident in smaller fields-of-view. Figure 5.9 highlights these wishbone-like structures, which appear to be vortices merging with each other. There are three such pairs of vortices in this case, and all three appear between about 70° and 110° . Interestingly, the heat flux peaks as the streaks merge, though it is unclear from the heat flux alone whether this occurs simultaneously with turbulent breakdown.

The underlying physics behind the wishbone structures is not clear, but low speed computations by Kurz and Kloker [99] might hold a clue. Figure 5.10 is a reproduction of Figure 12 from their paper. It shows vortices in a low speed DNS of roughness elements in a 3D boundary layer. In a 2D boundary layer, a roughness element creates two pairs of large counter-rotating vortices behind the element. However, the crossflow instability only supports co-rotating vortices. As shown by Kurz and Kloker, this leads to only one pair of co-rotating vortices behind the element, one strong and one weak. In Figure 5.10, the two vortices merge some distance behind the element in a wishbone pattern similar to the present data. Computations at high speeds are necessary to better understand the phenomenon in Figure 5.9, but considering the relationship between peak heating and streak merging, it seems worthwhile to investigate it further.

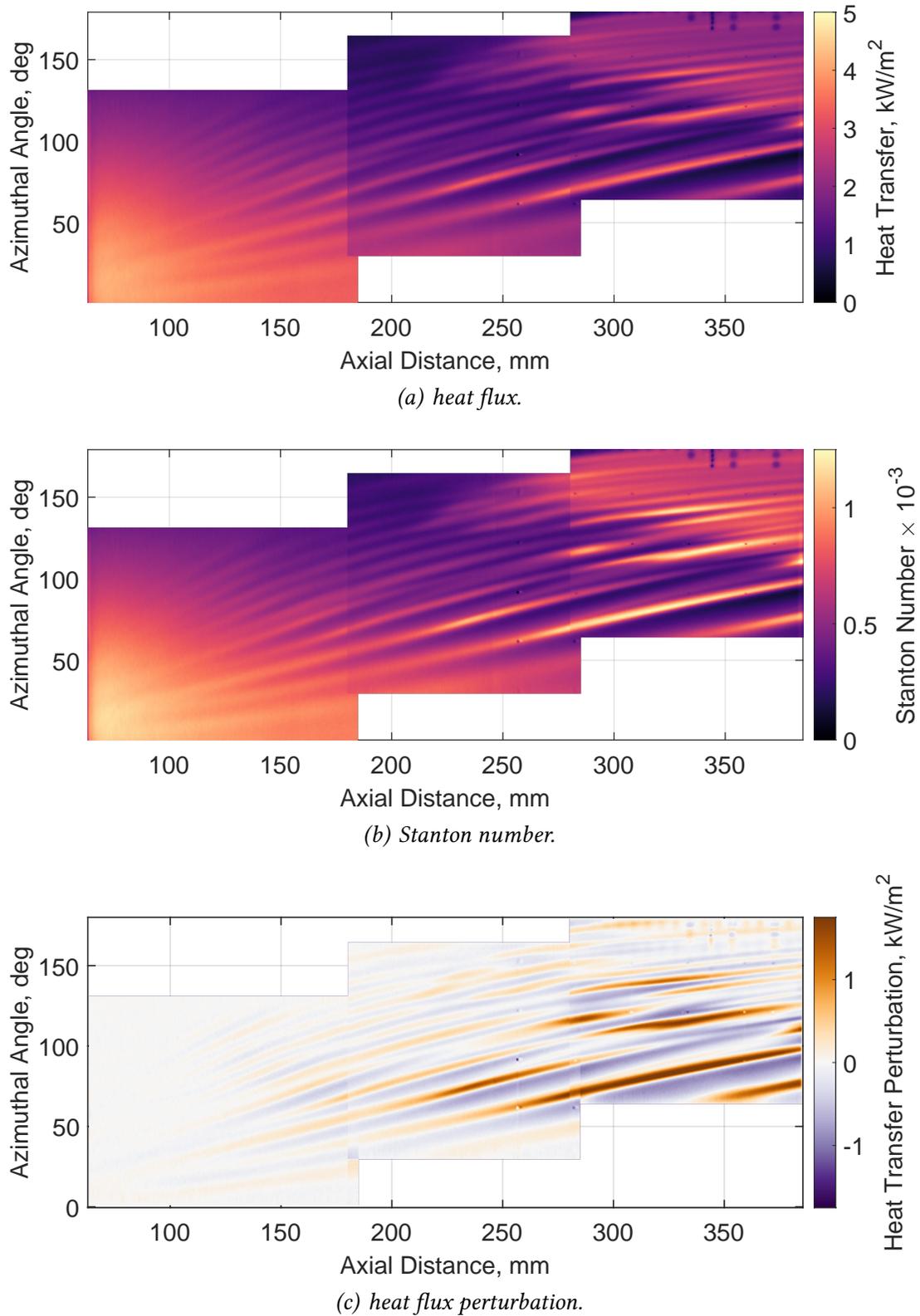
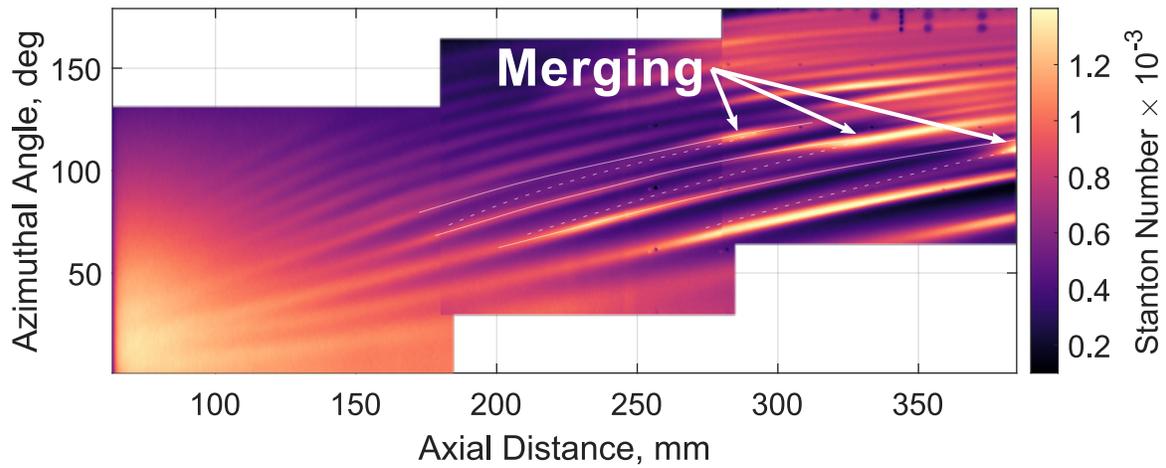
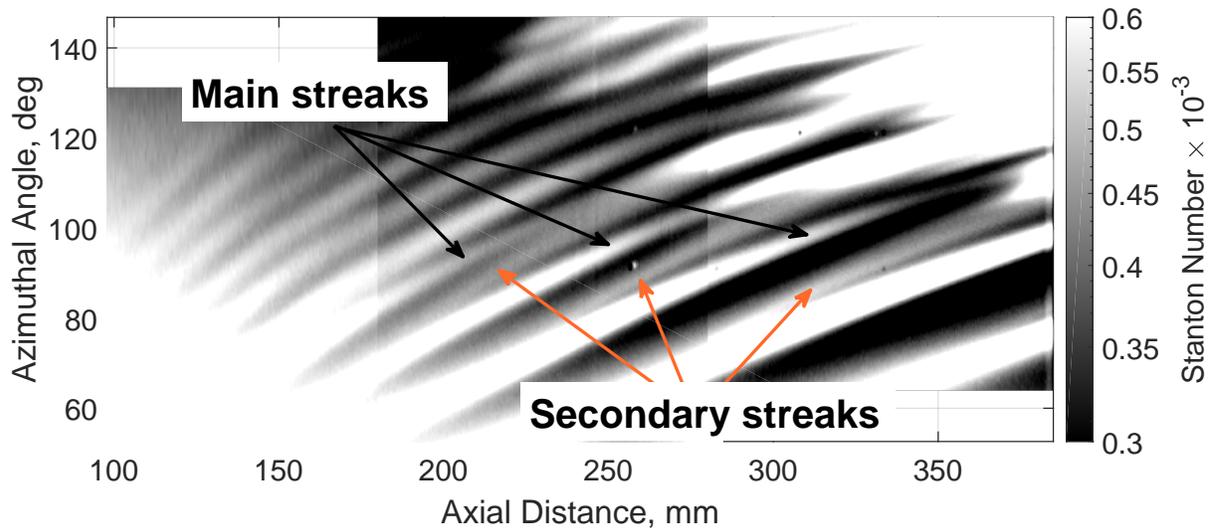


Figure 5.8. Surface heating caused by the stationary vortices. $Re_\infty = 11.2 \times 10^6 \text{ m}^{-1}$. Images from three runs, left to right: Run 1603, Run 1602, Run 1601.



(a) Full view indicating peak in heat flux at merge point.



(b) High-contrast image showing the main and secondary streaks.

Figure 5.9. Wishbone-type streak merging when using the RIM roughness.

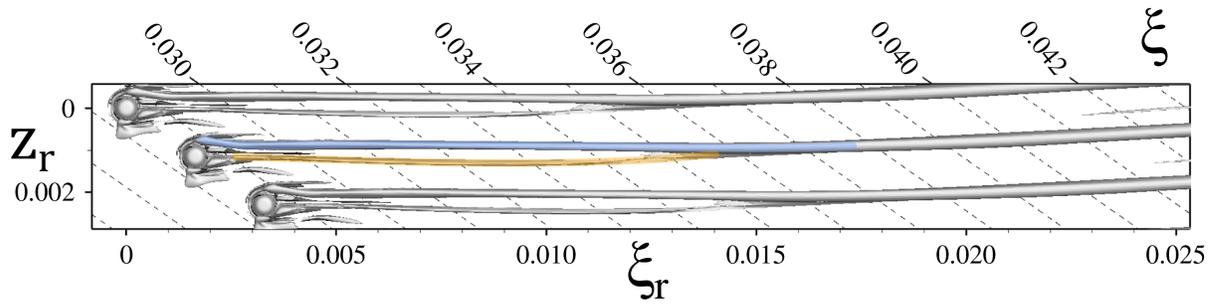


Figure 5.10. DNS of flow over isolated roughness in a 3D boundary layer. The main vortex (blue) and the secondary vortex (orange) merge in the similar way to the wishbone merging in Figure 5.9. Reproduced from Figure 12 in Kurz and Kloker [99], with permission. Color not in original.

For a spatial biglobal analysis of the secondary instabilities (see Reference [19]), it is imperative that computations are accurately modeling the spanwise wavenumber of the base flow (i.e. the stationary crossflow vortices). In previous experiments, the input wavenumber was chosen by measuring the naturally most-amplified wavenumber on a model without added roughness using images from the aft portion of the model. It is therefore important to determine how much the wavenumber changes upstream due to the effects of streamline spreading.

The computations of Moyes et al. [19] show that the most-amplified wavenumber actually decreases quite rapidly moving downstream. Their results indicate that for the experimental, natural-breakdown scenarios with a downstream wavenumber of around 60, the wavenumber at the roughness location would have to be around 220, which is at present impractical to fabricate. However, one could force a lower wavenumber upstream and compare the measured wavenumber downstream with a computation using the low input wavenumber, thereby validating the computational method's estimate of the wavenumber changes.

Most other studies using a known input wavenumber at hypersonic speeds have used small amplitude, dimpled Torlon [37,38,53], and it is not clear that these roughness elements were in fact dominating the generation of the stationary vortices downstream. The previous experiments also lack the spatial resolution to make a detailed comparison with computations. A notable exception is the work of Yates et al. [50], who have excellent, high-resolution images of the stationary vortices induced by a plasma perturber with a known wavenumber. However, if Yates et al. have examined the change in wavenumber over the length of the cone it has not yet been published.

Figure 5.11 shows a wavenumber spectrum from the heat flux data in Figure 5.8. For every axial station in the heat flux image, the power spectrum was computed using Welch's method. The circumferential wavenumber is the frequency in waves per degree multiplied by 360° . The resulting figure has been normalized by the maximum power in the entire imaged region, as the actual magnitudes are unimportant. Contours are also shown in Figure 5.11 to give a sense of the development. The green line is the computed wavenumber along a particular vortex for an upstream wavenumber of 50 (from Moyes et al. [19], Figure 23(b)). The wavenumber in the present experiments is 40, so this comparison is only qualitative.

Despite the differences in input wavenumbers, the results show similar trends. Note that the wavenumber varies in both the x and θ directions, so the experimental estimate will yield

an average wavenumber over the imaged azimuthal extent at any given x . The experimental wavenumber decreases downstream from an input of 40 to around 17 at the aft end. Interestingly, the power reaches a peak at around 330 mm, then decays, and then begins to increase again. This could be the result of the hot-cold-hot heating patterns on the individual streaks in this region, associated with the merging of the streak pairs (as shown in Figure 5.9).

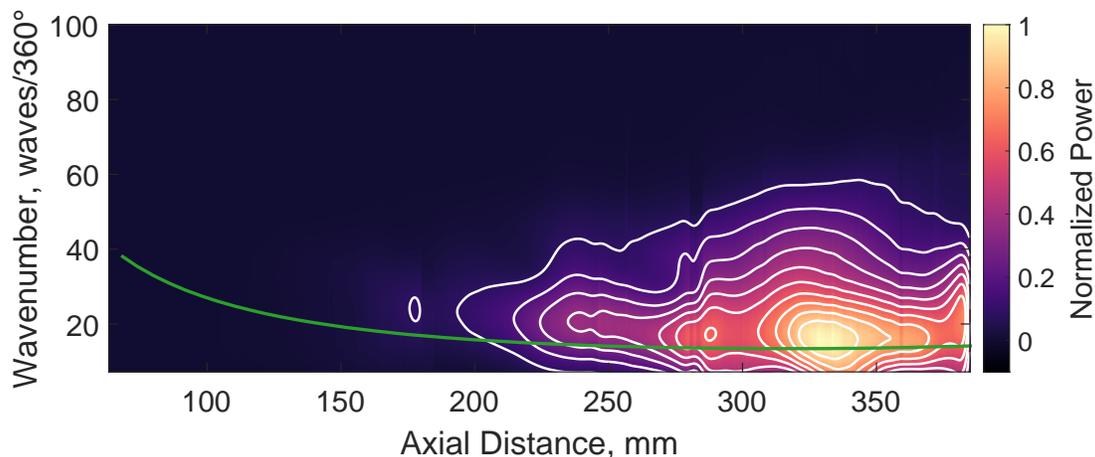


Figure 5.11. Wavenumber spectrum normalized by the maximum power. The wavenumber is the number of waves over the entire circumference of the cone, extrapolated from the imaged portion. The green line is the computed wavenumber from Moyes et al. [19] along a particular vortex for an upstream wavenumber of 50 (the present experiments used an upstream wavenumber of 40).

The peak heat flux along the streaks can be extracted semi-automatically using an algorithm provided in Appendix E. Figure 5.12(a) shows eight such paths overlaid on the Stanton number image. The vortices are numbered following the convention of Choudhari et al. [20], with increasing number toward the lee ray. The heat flux can also be interpolated along the extracted paths to show the growth of the peak heat flux. Figure 5.12(b) provides the ratio of the experimental Stanton number to the laminar computation along each streak. Initially, the ratio is roughly 1 for all the streaks, which indicates the boundary layer is mostly laminar at this point. Moving downstream, the heating on each streak begins to exceed the laminar level as the stationary vortices grow in amplitude. The peak St reaches over 5 times the laminar level for some streaks. However, given the uncertainties in the IR method (see Section 3.6), this level is only approximate.

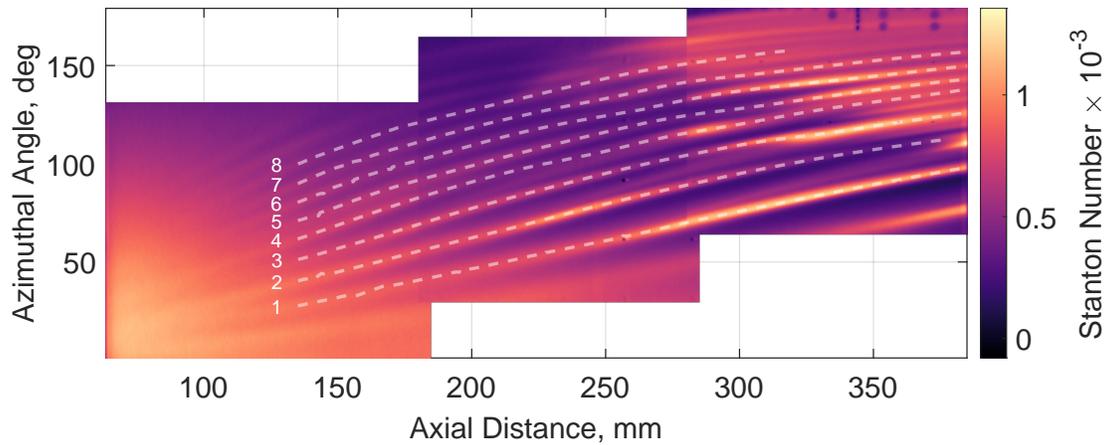
Interestingly, there appear to be two general classes of the vortex development: a monotonic increase in amplitude to turbulent breakdown, or a hot-cold-hot heating pattern. Figure 5.13 compares an example from each class. The first peak in the hot-cold-hot pattern seems to originate from the streak merging, as discussed previously. In general, the hot-cold-hot pattern appears at lower azimuthal angles, so it may be the result of the increasing Re_k at low azimuthal angles. However, note that for Vortex 6 and 7 there are noticeable kinks in the growth at around $Re = 3 \times 10^6$, even though there is no true double peak. The origin of these different classes can be further investigated by rotating the roughness insert by a small amount, which is discussed in Section 5.4.

Because the unit Reynolds number drops in steps throughout the run, measurements during a single run can provide information on the effect of small changes in the flow conditions. Figure 5.14(a) shows the heat flux along Vortex 4 at several different unit Reynolds numbers. The amplitude is $St_{\text{exp}}/St_{\text{CFD}}$, where St_{CFD} has been adjusted to the experimental unit Reynolds number using the simple scaling described in Section 3.5. The experimental heat flux is interpolated from a composite of images from two different runs (see Figure 5.12(a)), so the time during each run was calculated so that the two images had the same unit Reynolds number.

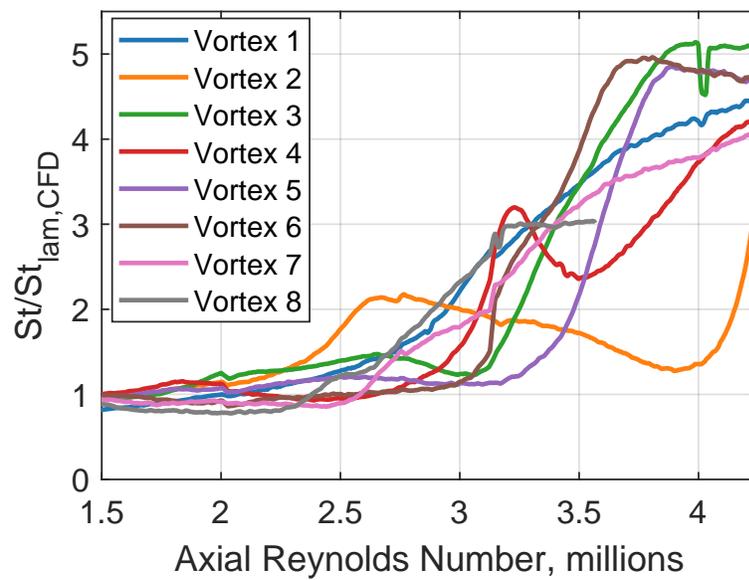
There are several things to note in Figure 5.14(a). The first peak in St moves steadily aft on the cone as the unit Reynolds number drops (lighter colors in the figure). This is a reasonable behavior, given that at lower Re_∞ , the Re_k of the roughness is lower, and therefore so are the initial amplitudes of the stationary vortices. Interestingly, the inset plot shows that the first peak reaches roughly the same amplitude over the entire range of Re_∞ , which varies by about 5%. The second peak, however, increases by about 10% at lower unit Reynolds numbers. The boundary layer in this region is likely turbulent, so the Stanton number probably varies with a different power of Re_∞ than used in the Section 3.5 scaling, which may explain the apparent increase in heat flux.

Figure 5.14(b) shows the same data, but plotted against Re , the Reynolds number based on axial distance. Both peaks in this coordinate system actually move forward by about 2% (to lower Re) at lower unit Reynolds numbers, which means the peak locations scale with some power of Re slightly less than one. Of course, the unit Reynolds number is one of several parameters that are changing over the course of the run. The stagnation temperature drops in the same manner

as Re , and the wall temperature increases. Both of these quantities surely have an effect on the crossflow instability, so the trends in Figure 5.14 are informative but far from comprehensive.



(a) Paths of the peak heat flux along streaks.



(b) Stanton number ratio along the streaks in Figure 5.12(a).

Figure 5.12. Amplitude growth of the streaks.

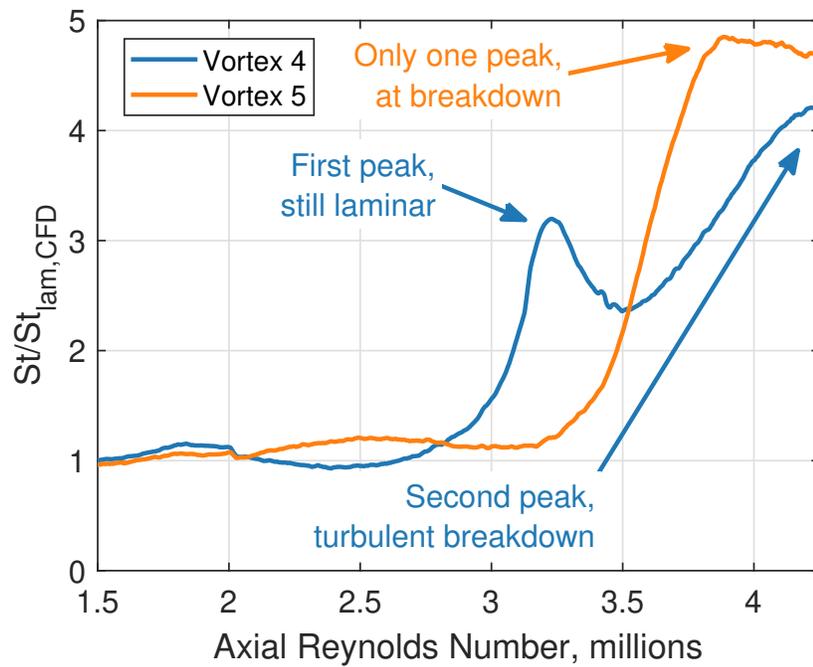
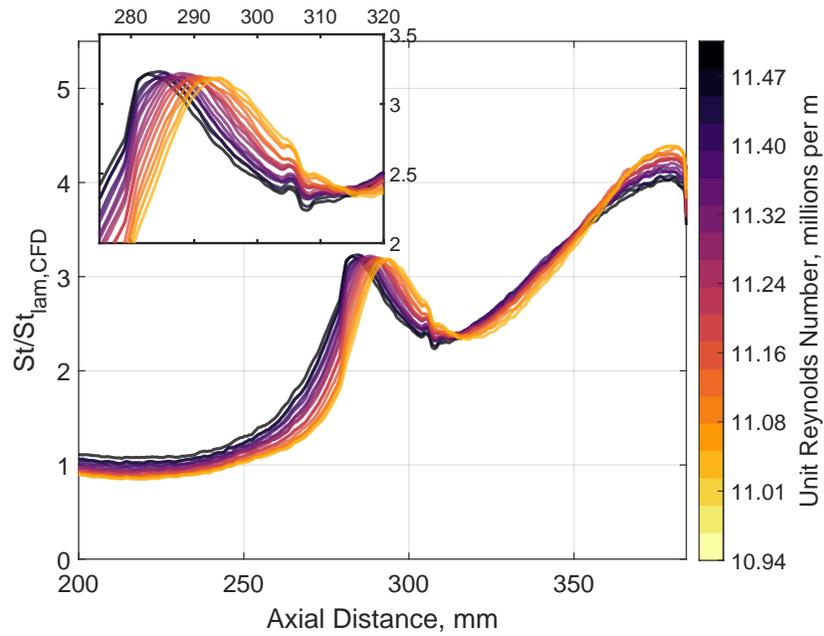
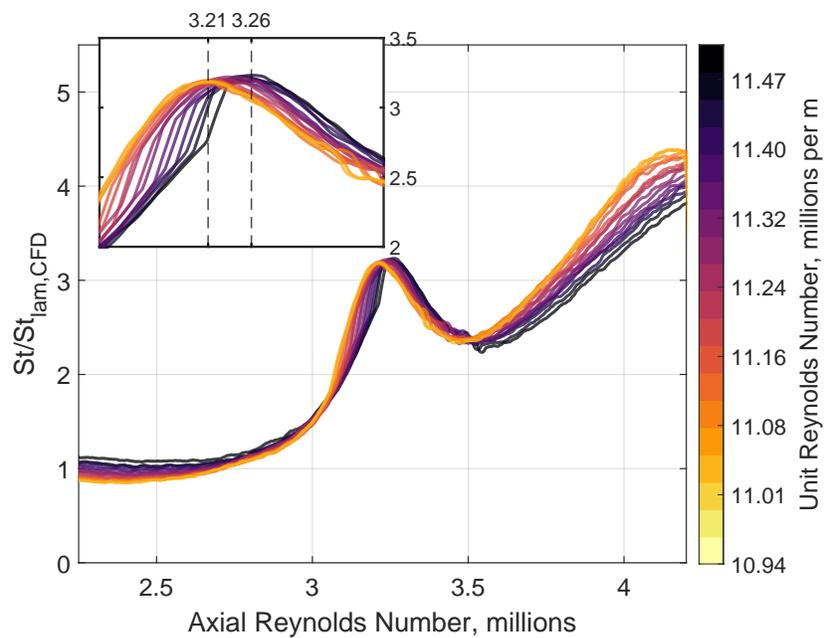


Figure 5.13. Comparison of two different classes of streak growth.



(a) St against axial distance.



(b) St against axial Reynolds number Re .

Figure 5.14. heat flux along Vortex 4 at several different unit Reynolds numbers (during the same run). The insets show more detail in the region of the first peak.

5.2.2 Traveling Crossflow Vortices

Though the traveling vortices may not be the dominant mechanism for transition, it is still important to understand their behavior, especially with regard to any interactions with the stationary mode. To that end, two runs were performed with Kulite sensors in a small cluster, as shown in Figure 5.15. Refer to Table 2.3 in Section 2.2 for the axial and azimuthal positions of the sensors. The arrangement is designed to enable the calculation of the traveling waves' phase velocity (see similar arrangements used by Borg [43] and Ward [38]). Unfortunately the Sensor Frustum with this array is made of aluminum, and time constraints did not allow the application of temperature-sensitive paint so there are no heat-transfer measurements with the Kulites. However, in light of the repeatability of this flow field demonstrated in Section 5.1.1, it is reasonable to assume the flow was roughly the same.

It is important to note, however, that the Kulite Sensor Frustum uses a different azimuthal angle scale than the PEEK one. The Kulite Sensor Frustum and accompanying Base uses center-drilled angle marks applied by the author (see Reference [73] for details including photographs of these marks), whereas the newer PEEK Sensor Frustum and Base uses a Vernier scale, precisely machined by an experienced machinist. Thus there is some uncertainty in the locations of measurements taken with the Kulites with respect to the same locations in the IR images. The offset uncertainty in azimuthal position of the PEEK Frustum can be estimated using the IR images, and is roughly $\pm 1^\circ$ with a resolution set by the Vernier scale at 0.5° . The uncertainty in the azimuthal angle offset using the Kulite Frustum cannot be as well quantified but is also around $\pm 1^\circ$ or larger, and with a 1° resolution. Thus due to uncertainty accumulation, the sensor measurements and the IR measurements could be nearly 3° apart. In the future, a second PEEK Frustum with both Kulite clusters and the Vernier scale should be fabricated to mitigate this issue.

Figure 5.16 shows the spectral quantities for the Kulite measurements in the position shown in Figure 5.15. The PSDs (Figure 5.16(a)) have a very large sensor resonance at 300 kHz for Kulites 1 and 2, and 225 kHz for Kulites 4 and 5. In addition, Kulite 1 shows a small peak at around 150 kHz, which could be a secondary instability, discussed further in Section 5.3. A notable absence from these spectra is any distinct instability in the traveling crossflow band around 45 kHz. At 129° there is a small peak in the spectrum at 50 kHz that could be traveling crossflow, but there is a

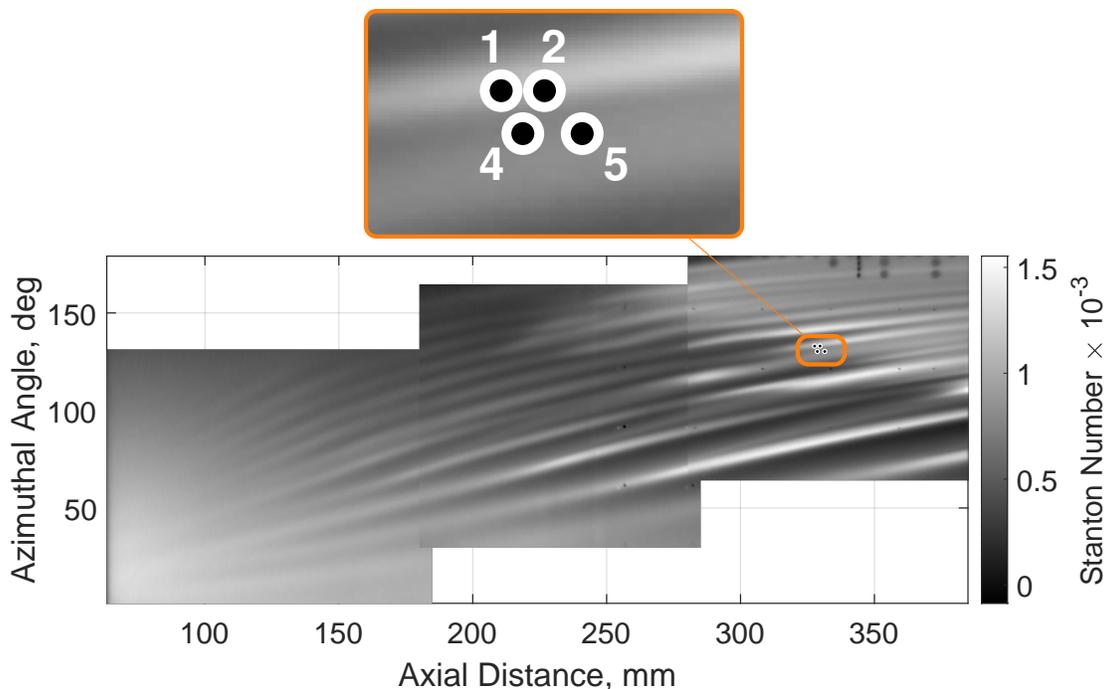
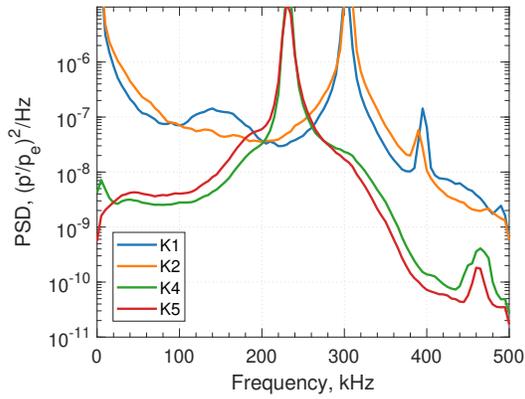


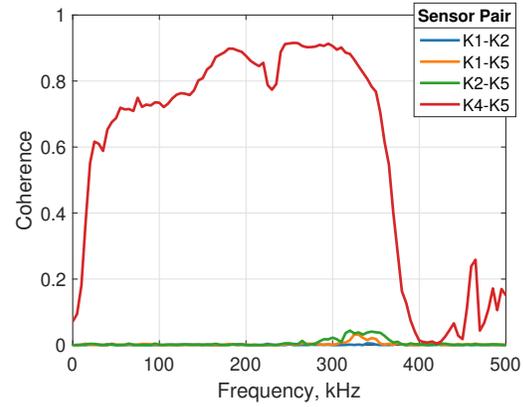
Figure 5.15. Locations of the four Kulite sensors. Their placement relative to the stationary vortices is visible in the zoomed inset.

large amount of background noise in the same frequency band. However, the lack of a distinct traveling crossflow peak is markedly different from measurements using random roughness, which is discussed in detail in Chapter 6.

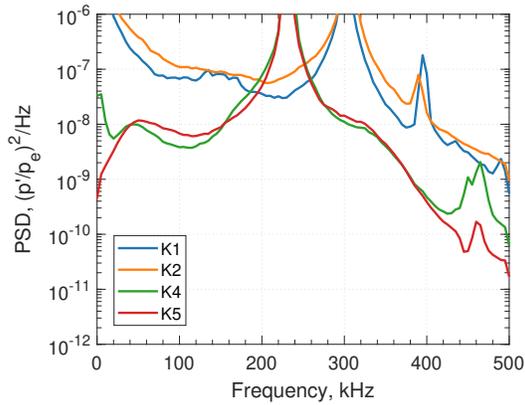
The coherence between sensor pairs is also interesting, though it leads to more questions than answers. The only non-zero coherence is between Kulites 4 and 5 (only 4 of the 6 possible sensor pairings are shown, for clarity). It is not obvious why this should be the case, but the effect was observed in both runs with the Kulites (Figure 5.16(b) and (d)). It is possible that the lack of coherence is indicative of equipment malfunction. But the same setup was used a few runs earlier to take measurements using the Sandblasted roughness insert, with every sensor pair having high coherence. In addition, the sensors were calibrated after these runs and seemed to be functioning properly. Another hypothesis is that the strong stationary vortices are modulating the traveling waves so significantly that there are distinct modes in this region, and only Kulites 4 and 5 are measuring the same one. Further experiments with both IR and Kulites are needed to solve this puzzle.



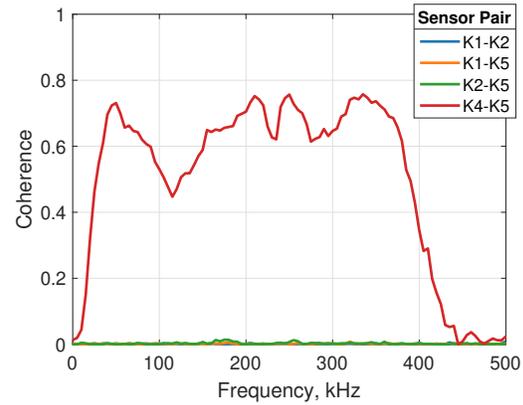
(a) Power spectral densities, $\theta = 133^\circ$, Run 1719. The narrow peaks at 225 kHz and 300 kHz are sensor resonances.



(b) Coherence, $\theta = 133^\circ$.



(c) Power spectral densities, $\theta = 129^\circ$, Run 1720. Note the possible traveling crossflow at 50 kHz on Kulites 4 and 5.



(d) Coherence, $\theta = 129^\circ$.

Figure 5.16. Spectral properties of the Kulite measurements. $Re_\infty = 10.5 \times 10^6 \text{ m}^{-1}$.

5.3 Secondary Instability Measurements

In Section 5.3.1 it will be demonstrated that two modes of secondary instability coexist in this case: a low-frequency and a high-frequency mode. These new measurements refine earlier data from experiments by the author using a dimpled Torlon roughness [21, 73]. Then in Section 5.3.2, the streamwise growth of the secondary modes will be examined and compared to the amplitude growth of the stationary crossflow vortices in the same region. Finally, in Section 5.3.3, the nonlinear behavior of the secondary modes near breakdown will be discussed.

The data in this section are synthesized from 20 individual PCB pressure sensors over the course of 29 runs with the Sensor Frustum at different azimuthal positions. Figure 5.17 shows the sensor locations for these data overlaid on an IR image; each white dot represents one sensor during one run. Note that the dots are smaller than the actual sensors.

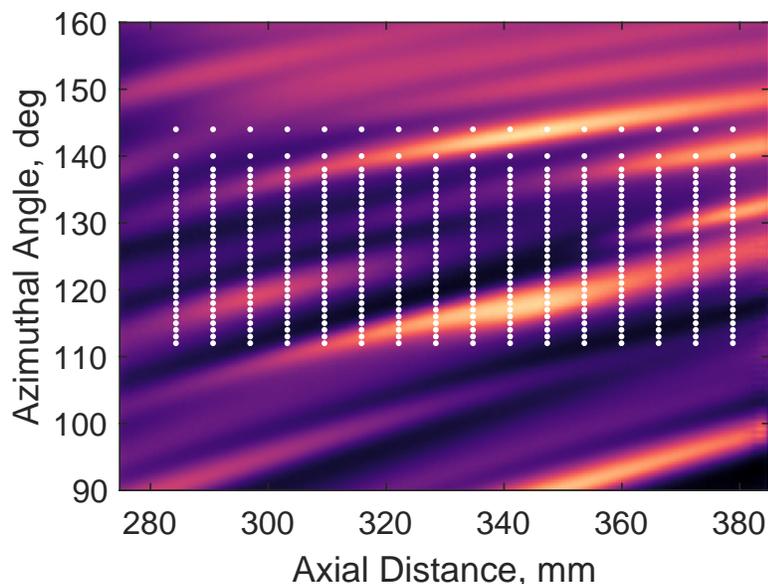


Figure 5.17. The measurement points for the data in this section overlaid on an IR heat-transfer image. The dots are not the same size as the PCB sensors.

5.3.1 Peak Frequencies

Prior measurements by the author showed that there exist two separate modes of the secondary instability, with frequencies apparently dependent on the local, modulated boundary layer thickness [21]. However, these measurements were made using a dimpled Torlon roughness, which as discussed previously is not well controlled. In addition, the measurements were low density and fairly limited in scope, with an axial span of only about 40 mm, and an azimuthal range of 18° . The measurements in the present experiment are at a much higher spatial resolution and have an expanded scope, so the behavior of the two modes can be examined more clearly.

Figure 5.18 shows PSDs of the secondary-instability pressure fluctuations at a constant axial position, for a range of azimuthal angles. The experimental peak frequency is marked with an open circle for each spectrum; the corresponding second-mode estimate from the laminar computation is marked as a filled circle. As is evident in the figure, the peak frequency can be tricky to determine; this is especially true for the measurements at 132° and 124° , where the fluctuations are very broad-band. To approximate the measured peak frequency, the spectra were smoothed with a 75 kHz moving average filter, and the location of the highest peak is extracted from the smoothed PSD in the range from 80 kHz to 700 kHz. This is not a perfect method for determining the “true” peak frequency—as is obvious from Figure 5.18—but it is an automatic and consistent metric, so it should still provide insight into the nature of the instabilities.

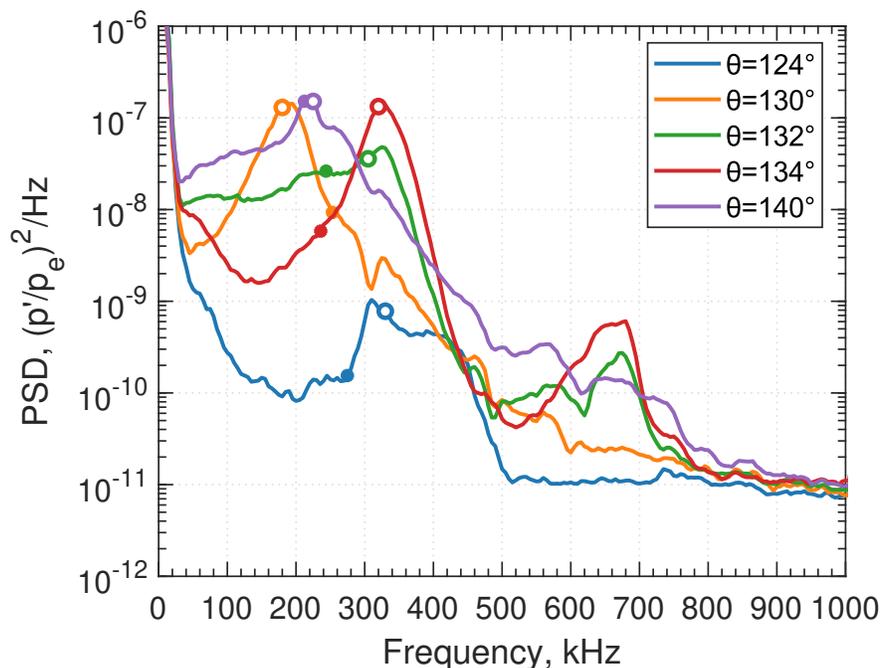


Figure 5.18. PSDs of the secondary instability for several different azimuthal angles at $x = 347$ mm. The open circles mark the approximate peak frequency, while the closed circles are the estimated second-mode frequency. $Re_\infty = 10.5 \times 10^6 \text{ m}^{-1}$.

It is clear from Figure 5.18 that the peak frequency changes quite a bit more than the estimated second-mode frequency (closed circles). In fact, the instability frequency seems to oscillate around the estimate, which is consistent with the modulated boundary-layer thickness caused by the stationary vortices. The peak frequency as a function of azimuthal angle is provided in Figure 5.19

for two axial stations. In addition, the plot shows the estimated second-mode frequency from the computation, $f_{\text{est}} = U_e/(2\delta)$ as the dashed line. The peak frequency oscillates around the estimate as the boundary layer thickens and thins due to the presence of the stationary crossflow vortices. The estimated second-mode frequency is very much an approximation; early measurements of the second mode show variation in the peak frequency with wall temperature and Reynolds number through their effects on the boundary-layer thickness [109, 110]. The measurements of Demetriades (See Fig. 22 in Ref. [109]) indicate that the frequency estimate is better described by $f_{\text{est}} \approx U_e/(2.2\delta)$ at high Reynolds numbers. This approximation is plotted in Figure 5.19 as the solid black line, and appears to be a better fit for the mean variations in instability frequency. The approximation $f_{\text{CFD}} = U_e/(2.2\delta)$ is used as the estimated second-mode frequency for the rest of this work.

The peak frequencies for every measurement point can be extracted and interpolated to form a peak-frequency map, shown in Figure 5.20. There are several lobes of high-frequency activity, with low-frequency regions in between. Note that because the boundary layer thickens downstream and toward the lee ray, the peak frequency drops in these directions.

To account for the axial and azimuthal changes in the mean flow, the experimental peak frequency can be normalized by the local second-mode estimate to yield a map of the peak frequency ratio, $f_{\text{exp}}/f_{\text{CFD}}$. A map of this ratio is shown in Figure 5.21. Normalizing the peak frequencies makes the pattern more distinct: four lobes of high-frequency modes interleaved with low-frequency ones. In addition, the lobes appear to have roughly constant frequency ratios for both the high and low-frequency cases. The frequency variation is evidently quite large, ranging from 60–160 % of the local second-mode estimate, though of course given the method for extracting the peak frequency this magnitude should only be taken qualitatively.

It is insightful to understand the location of these lobes of activity with respect to the stationary vortices. Figure 5.22 shows the contours of the peak frequency superimposed on an IR image of the flow at the same Reynolds number. The lobes of high-frequency modes correspond almost exactly to hot streaks in the IR, and vice versa for the low-frequency modes and cold streaks. The second-mode frequency is highly dependent on the local boundary-layer thickness, so one would expect a higher frequency instability under the thin troughs between vortices, which would also lead to higher heat transfer.

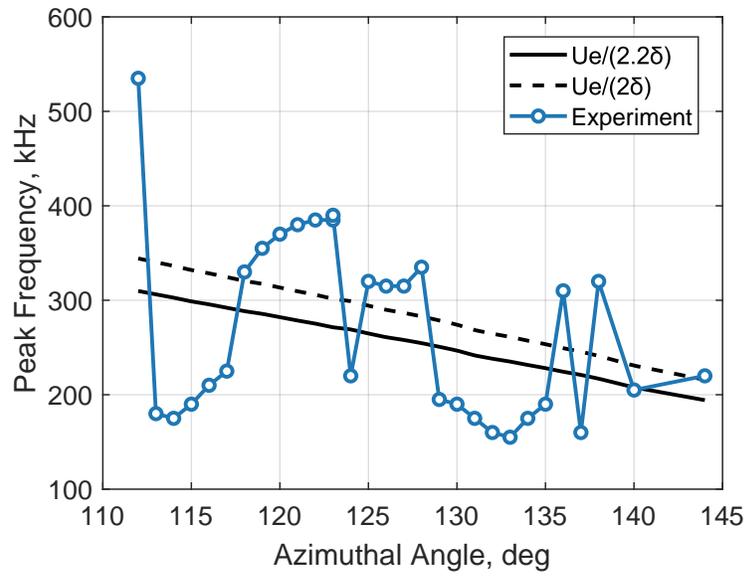
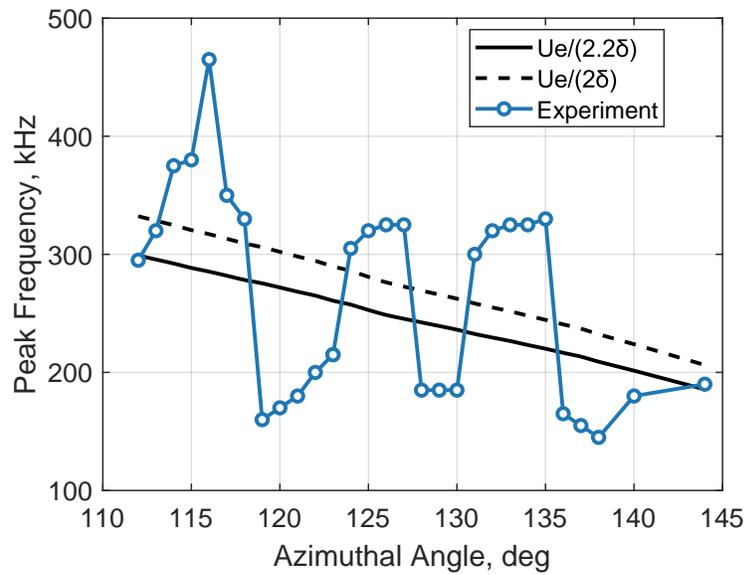
(a) $x = 303$ mm.(b) $x = 328$ mm.

Figure 5.19. Variation in peak frequency of the measured instabilities at two axial stations. Both U_e and δ were extracted from the Navier-Stokes solution as described in Section 3.5.

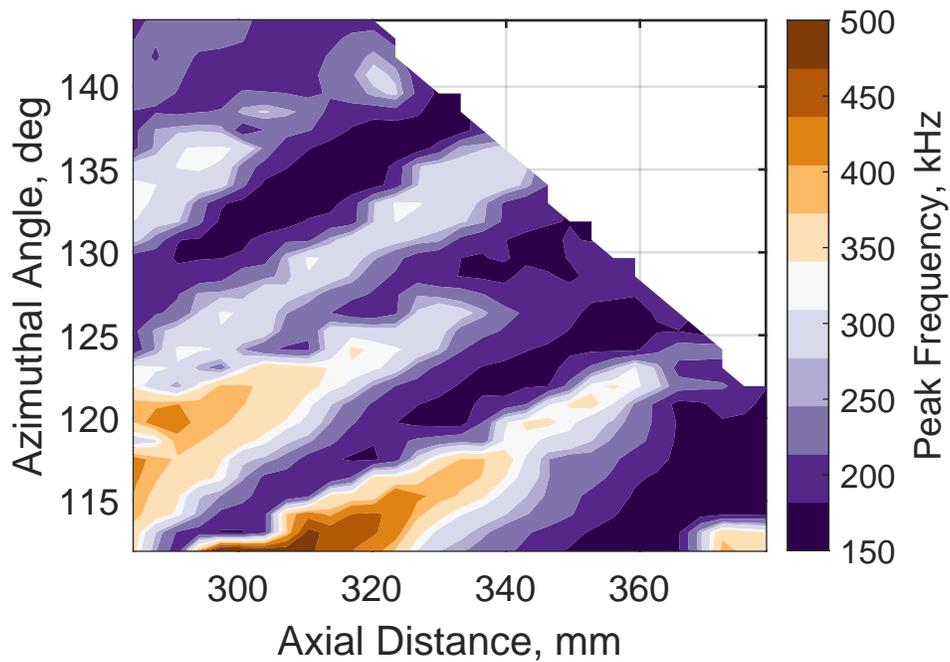


Figure 5.20. Contour plot of the experimental peak frequency over the entire measurement range. The peak frequency decreases for all instabilities in the streamwise direction, as the boundary layer thickens both axially and azimuthally. The flow in the upper-right corner is beginning to break down, so the peak identification is not as reliable. The data in this region have been omitted. $Re_\infty = 10.5 \times 10^6 \text{ m}^{-1}$.

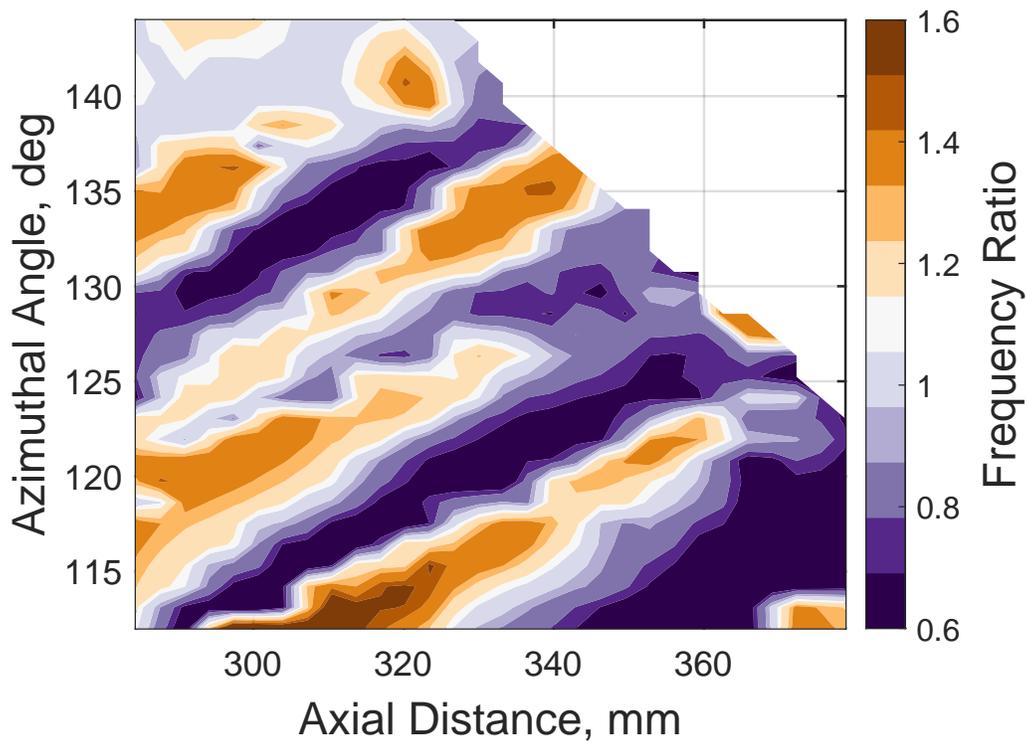


Figure 5.21. The ratio of the measured peak frequency to the estimated second-mode frequency, from the data in Figure 5.20. This ratio accounts for the changing boundary layer thickness, so the spanwise modulation in frequency is more distinct.

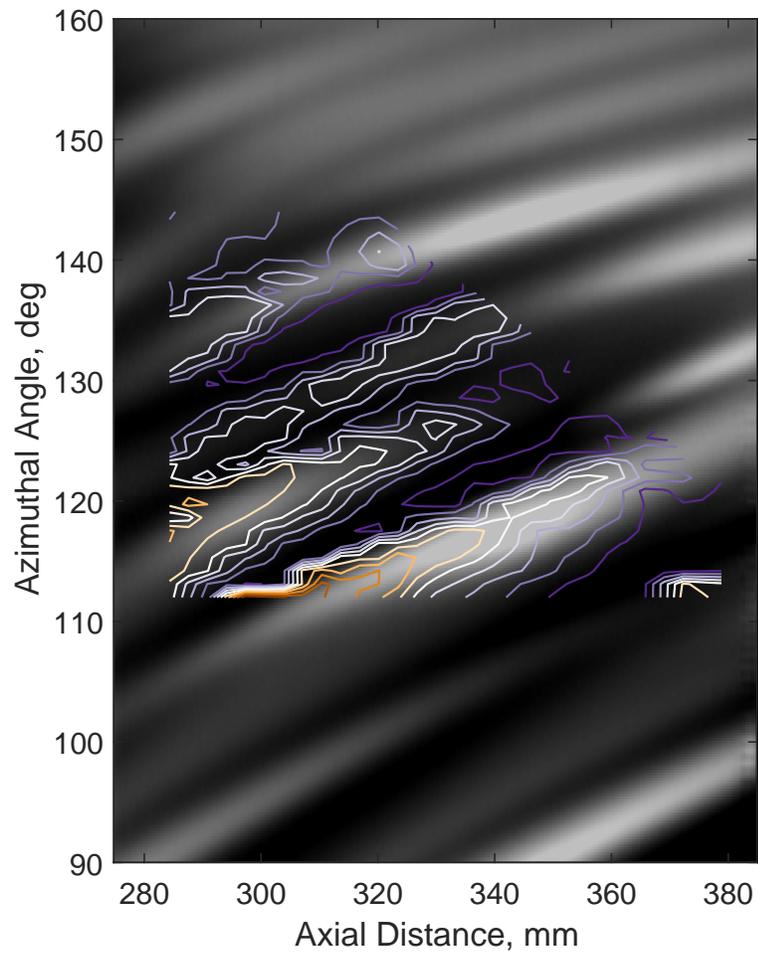


Figure 5.22. Contour plot of the peak frequency from Figure 5.20 overlaid on an IR image. The high-frequency modes are concentrated under the hot streaks, while the low-frequency modes grow in the cold spaces in between.

5.3.2 Amplitude Development

The previous section showed that the pressure fluctuation spectra generally fall into one of two classes: a Low-Frequency (LF) mode in the range $50-f_{\text{CFD}}$ kHz or a High-Frequency (HF) one from $f_{\text{CFD}}-1000$ kHz, where f_{CFD} is the estimated second mode frequency from the laminar computation. The wide HF band captures both the secondary instability and its first harmonic. To examine the growth of the secondary instability amplitudes, the amplitude was calculated in each band for every measurement point. Figure 5.23 shows contour plots of the amplitudes in each of these bands, giving a global sense of the development of the secondary instability.

In each of the amplitude maps there are several lobes of growth aligned along the stationary vortices. Figure 5.24 shows contours of the HF amplitude superimposed on an IR heat-transfer image. The lobes of HF growth are almost exactly aligned with the hot streaks in the IR. The amplitude of the HF instability appears to peak coincidentally with the peak heat transfer caused by the wishbone-type merging of the stationary vortices (see the merging at 295 mm and 120° in the IR, and in Figure 5.9). It is unknown if there is a physical connection between these processes.

The lobes of LF and HF growth are slightly offset, as illustrated in Figure 5.25. This makes sense given the connection between instability frequency and the local boundary-layer thickness. However, it appears in the Figure that the LF mode is far less localized than the HF one. It is possible that this apparent delocalization is an artifact of the higher levels of background noise at low frequencies (for instance, see the PSD at $\theta = 134^\circ$ in Figure 5.18), and so may not be indicative of any characteristic difference between the modes.

A more detailed sense of the breakdown behavior of each mode of instability can be gleaned from following the growth of the secondary instabilities along a stationary vortex. Figure 5.26 shows some of the discrete sensor locations used to measure the growth and breakdown of the LF and HF instability along a single stationary vortex (Vortex 4 in Figure 5.12(a)). Due to the slight offset in the lobes of instability growth, the LF measurement points are about a degree higher in azimuth than those of the HF growth, which follow the path of the hot streak.

The pressure fluctuation spectra from the measurement points in Figure 5.26 are shown in Figure 5.27. For the LF mode, in Figure 5.27(a), there is a monotonic increase in power in the LF band, and though the spectra still show the presence of an instability at $\text{Re} = 3.91 \times 10^6$, it

appears the flow is nearly turbulent by this point. The HF mode, on the other hand, is likely fully turbulent by $Re = 3.65 \times 10^6$, judging from the shape of the spectrum. In addition, there are clear harmonics of the HF mode at around 750 kHz. The increase in amplitude at low frequencies for the HF instability could indicate the presence of a separate LF mode. However, given that there is no distinct peak in this range, the increase in power is most likely due to spectral filling, a result of turbulent breakdown.

The amplitudes of the LF and HF instability along this vortex are compared to the stationary vortex amplitude in Figure 5.28. Three axial stations are indicated at the top of the figure, at which the Shannon entropy H was calculated to provide an approximate turbulence metric. At station (1), $H_{LF} \approx 70\%$ and $H_{HF} \approx 80\%$, indicating that both instabilities are fairly near breakdown at the beginning of the measurement range (though it should be noted that this conclusion not corroborated by the LF spectrum which shows very little activity at this point). At station (2), $H_{HF} = 99\%$, so it seems that the boundary layer along the path of HF growth has become mostly turbulent at this location. This is in general agreement with the analysis of the spectra, above. At station (3), $H_{LF} = 99\%$, so the boundary layer along the path of the LF mode has also become mostly turbulent at this point. Again, this corroborates the results from the spectra. It is interesting to note that H appears to reach 99% slightly downstream of the peak pressure fluctuation amplitude, when the amplitude has leveled off. This behavior is similar to intermittency measurements of the second mode on a flared cone [66].

The amplitudes of two vortices are shown in the figure: Vortex 4, which is under the HF path, and Vortex 5, which is just slightly leeward of the LF path. The amplitudes of the vortices seem to correspond to the behavior of the secondary instabilities. The first peak in the Vortex 4 amplitude occurs simultaneously with the peak in the HF instability. Both Vortex 5 and the LF instability begin to grow in amplitude around this point as well, and seem to reach breakdown at roughly the same location. Interestingly, the “transition location,” as loosely defined as $H_{LF} = 99\%$, occurs *after* the first peak in the amplitude of Vortex 4, almost to the second minimum of the amplitude. Traditional estimates of the transition location use the point at which the heat transfer begins to grow from the laminar value. In this case, using such an estimate would be too small by nearly $Re = 500 \times 10^3$, or roughly 50% of the measured length.

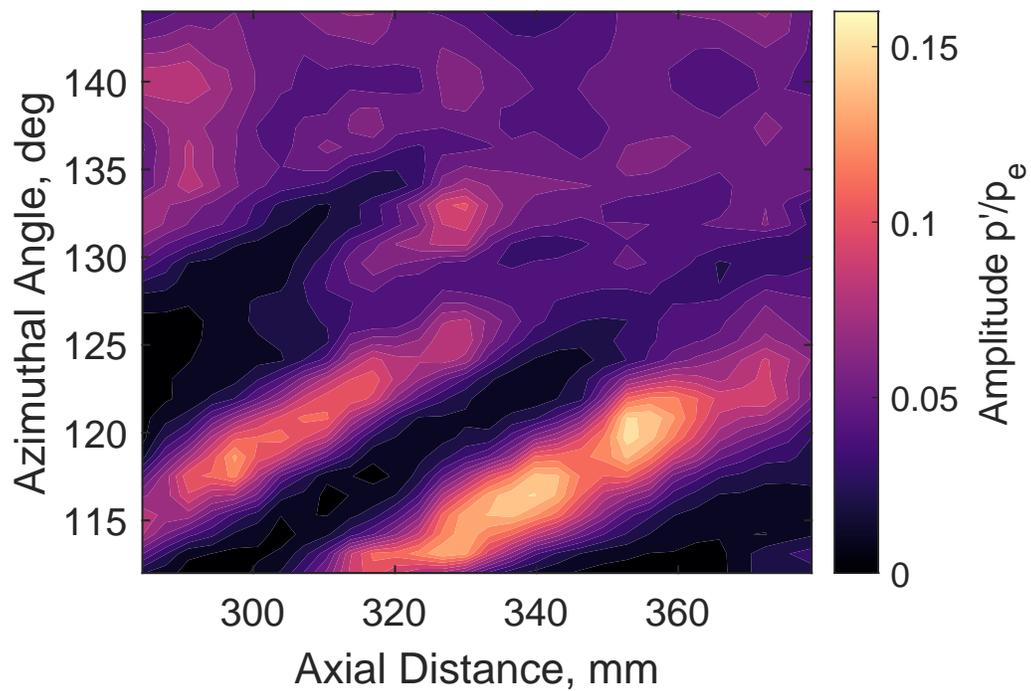
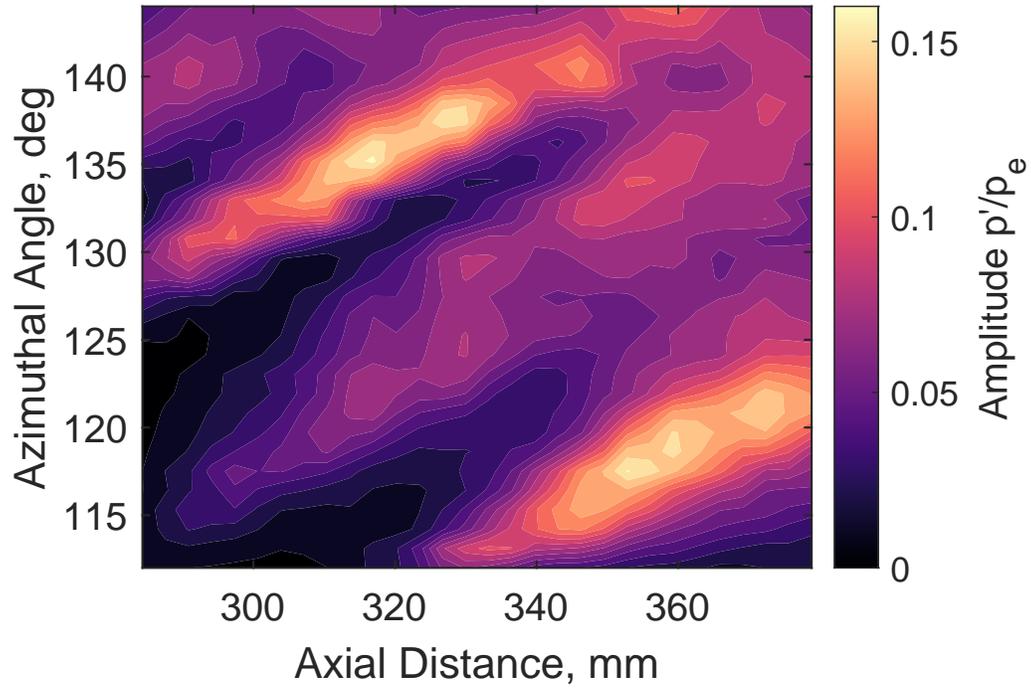


Figure 5.23. Pressure fluctuation amplitudes in the low and high frequency bands.
 $Re_{\infty} = 10.5 \times 10^6 \text{ m}^{-1}$.

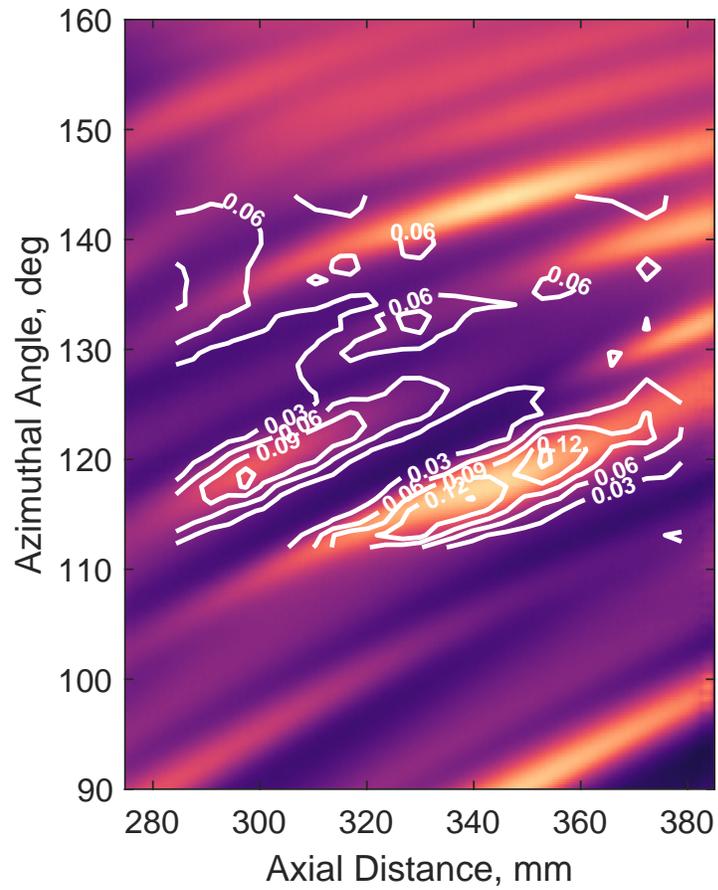


Figure 5.24. Contours of amplitude in the HF band overlaid on an IR image. The lobes of HF growth are almost exactly aligned with the hot streaks in the IR.

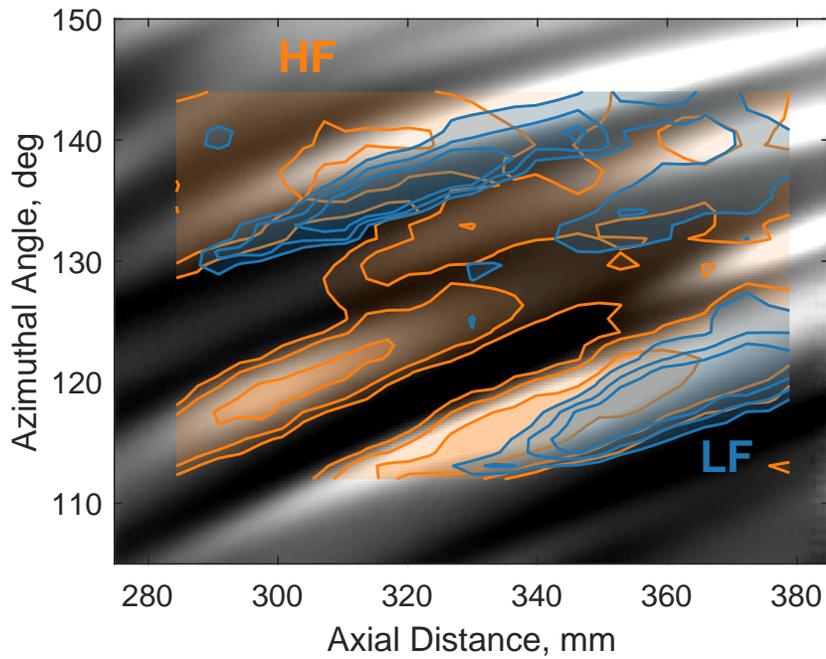


Figure 5.25. Contours of both secondary instability modes superimposed on an IR image of the same flow. LF is the low-frequency mode (Fig. 5.23(a)) and HF is the high-frequency mode (Fig. 5.23(b)).

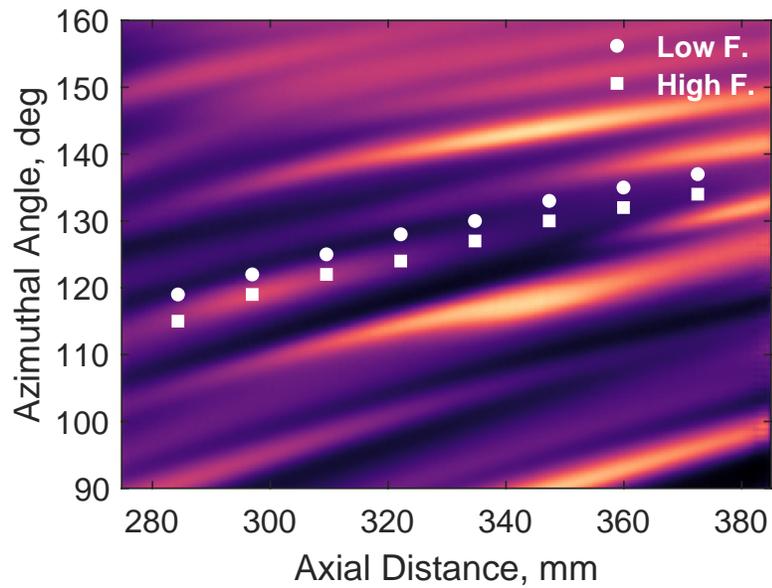
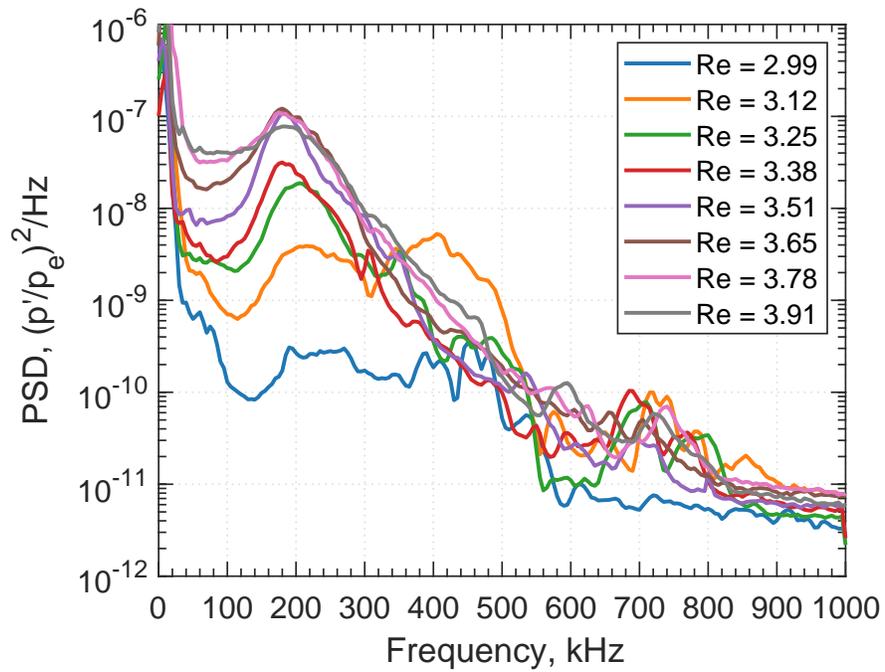
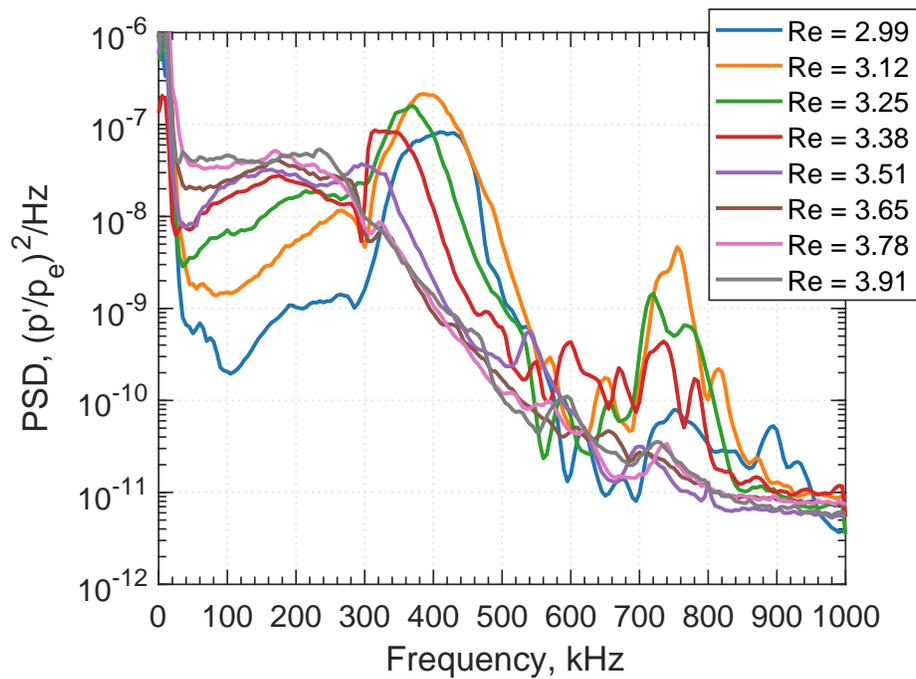


Figure 5.26. Individual measurement points following a stationary vortex, corresponding to the spectra in Figure 5.27.



(a) LF instability.



(b) HF instability.

Figure 5.27. Spectra of the measurements along a single vortex, for both instability modes. The increase in power at low frequencies for the HF mode may be spectral filling, a consequence of turbulent breakdown, or it could represent growth of a separate low frequency mode. The Reynolds numbers are in millions based on the axial distance of the measurement.

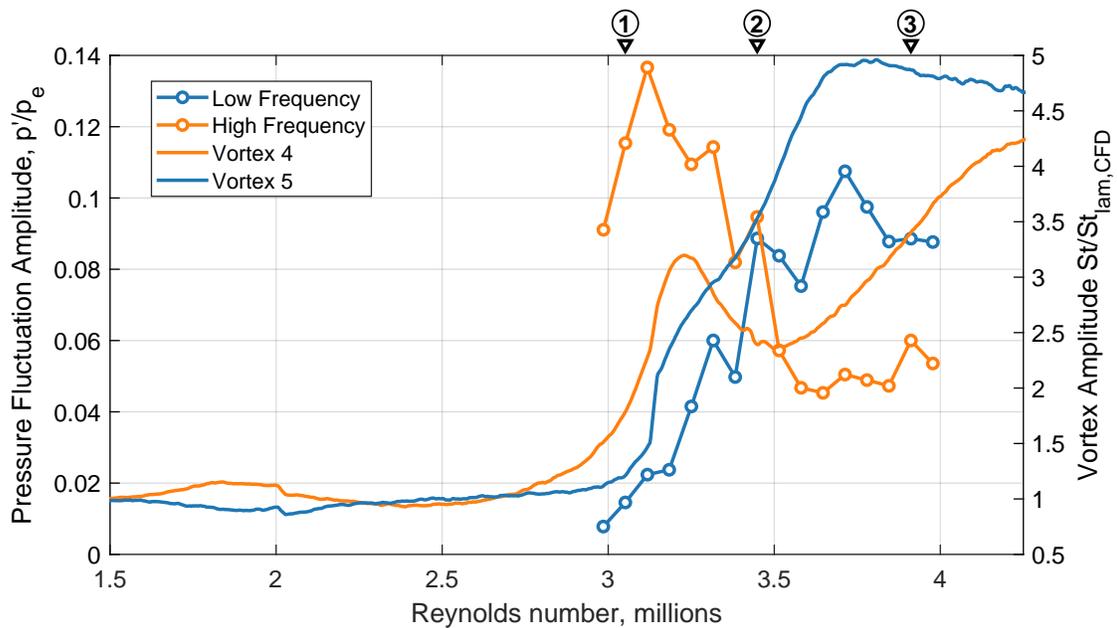


Figure 5.28. Comparison of the amplitude growth in the primary and secondary instabilities. Vortex 4 is physically closest to the high-frequency lobe while Vortex 5 is closest to the low-frequency one. The Shannon entropy H was calculated at three axial stations of interest, marked on the plot: (1) $H_{LF} \approx 70\%$, $H_{HF} \approx 80\%$; (2) $H_{HF} = 99\%$; (3) $H_{LF} = 99\%$. Note that the heat-flux measurements were taken at $Re_\infty = 11.2 \times 10^6 \text{ m}^{-1}$ whereas the pressure fluctuation measurements are from $Re_\infty = 10.5 \times 10^6 \text{ m}^{-1}$. See Figure 5.14 for the effect of small changes in unit Reynolds number.

5.3.3 Nonlinear Interactions

The nonlinear behavior of the secondary instabilities can be classified as one of two types: harmonic generation and nonlinear interaction between modes. Harmonic generation will occur when a mode of the instability becomes large enough in amplitude. Figures 5.29 and 5.30 provide example bicoherence measurements displaying harmonic generation. Figure 5.29 illustrates self-interaction in the low frequency mode. A peak in the bicoherence is observed at around (180 kHz, 180 kHz), coincident with the LF peak in the spectrum. The harmonic in this case would be at around 360 kHz, but it is not clear from the spectrum that such a peak exists. However, the bandwidth of the LF peak is fairly broad, and the magnitude of the nonlinear interaction is small ($b^2 = 0.15$), so it is possible that the harmonic is partially obscured by the long roll-off of the fundamental peak (see Appendix F). Figure 5.30 provides a similar example for the HF mode. In this case, there is a very strong ($b^2 = 0.41$) peak in the bicoherence corresponding to the large HF peak at around 360 kHz. Due to the strength of this self-interaction, a harmonic is easily identifiable in the spectrum at around 750 kHz.

In addition to harmonic generation, there is nonlinear interaction between the LF and HF modes in the regions where they overlap. Figure 5.31 provides one example of this interaction. There are two peaks clearly evident in the spectrum: one at 160 kHz and another at 320 kHz. It is possible that the higher frequency is in fact a harmonic of the LF peak (the frequency is almost exactly double), but the two instabilities appear to have roughly the same amplitude, which would be unusual for a harmonic (though not impossible given the non-flat PCB frequency response). The bicoherence shows high levels of harmonic generation within the HF peak causing the small but noticeable activity at 700 kHz. In addition, there is a small region of significant bicoherence at (320 kHz, 160 kHz), indicating the two modes are interacting with each other to transfer energy to an instability at 480 kHz. There does appear to be a small peak in this region in the PSD, though the bicoherence is fairly low so a distinct peak is not necessarily expected.

A more obvious case of LF-HF mutual interaction is shown in Figure 5.32. Again, there are two large-amplitude peaks very clearly visible in the spectrum at 235 kHz and 325 kHz. In this case the two frequencies are not integer multiples, so it is probably not a harmonic but rather a separate instability mode. The bicoherence shows three distinct regions of interaction. There are

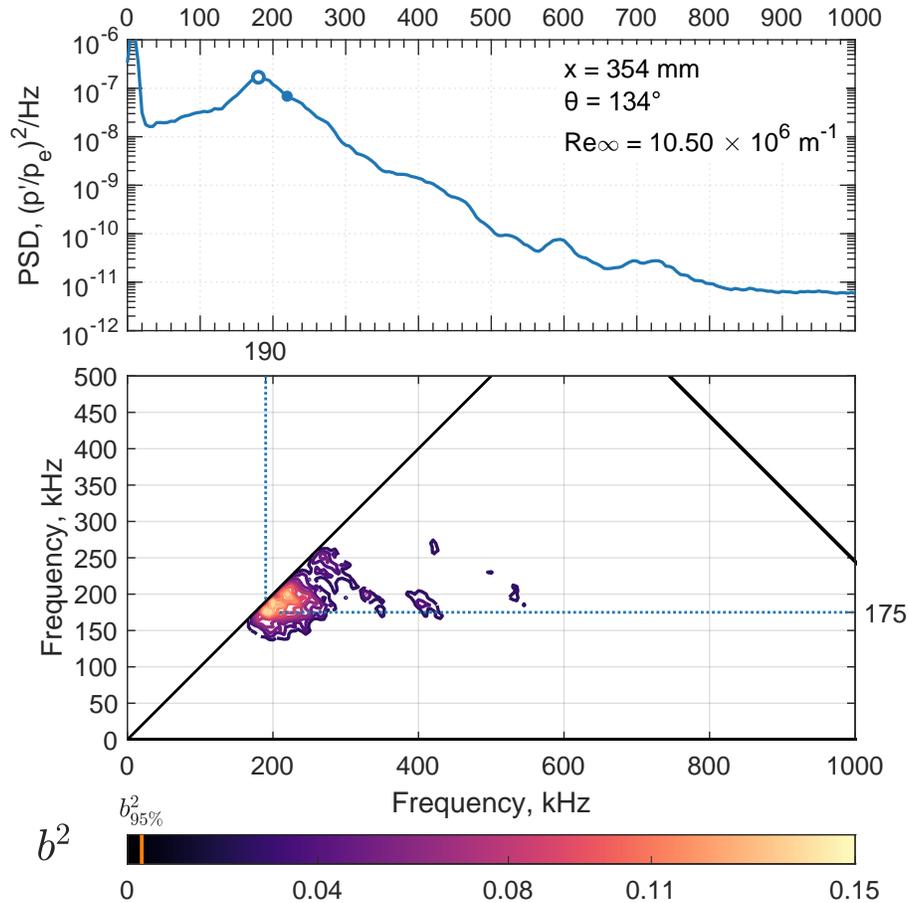


Figure 5.29. Harmonic generation in the LF mode. The filled dot on the spectrum is the CFD-estimated second-mode frequency. Run 1613.

high levels of harmonic generation within both the LF and the HF instability, but there is also bicoherence between the two modes, at (325 kHz, 225 kHz). Thus frequency peaks are expected at 450 kHz and 650 kHz (the harmonics), and also at 550 kHz (the mutual interaction). Indeed, looking at the PSD there are small but distinct peaks near the three expected frequencies.

Unfortunately, the results of the bicoherence calculation are not always easily interpretable. Figure 5.33 shows one example. The maximum bicoherence in this case is very large, $b^2 = 0.54$. In the spectrum, there is a broad, high-frequency peak at 500 kHz, as well as several smaller peaks at higher and lower frequencies. The measurement location is fairly far forward ($x = 310$ mm) and away from the lee ray ($\theta = 112^\circ$), so the boundary layer is quite thin, leading to the very high frequency of the instability. Unfortunately, these data were sampled at 2 MHz, so any harmonic of

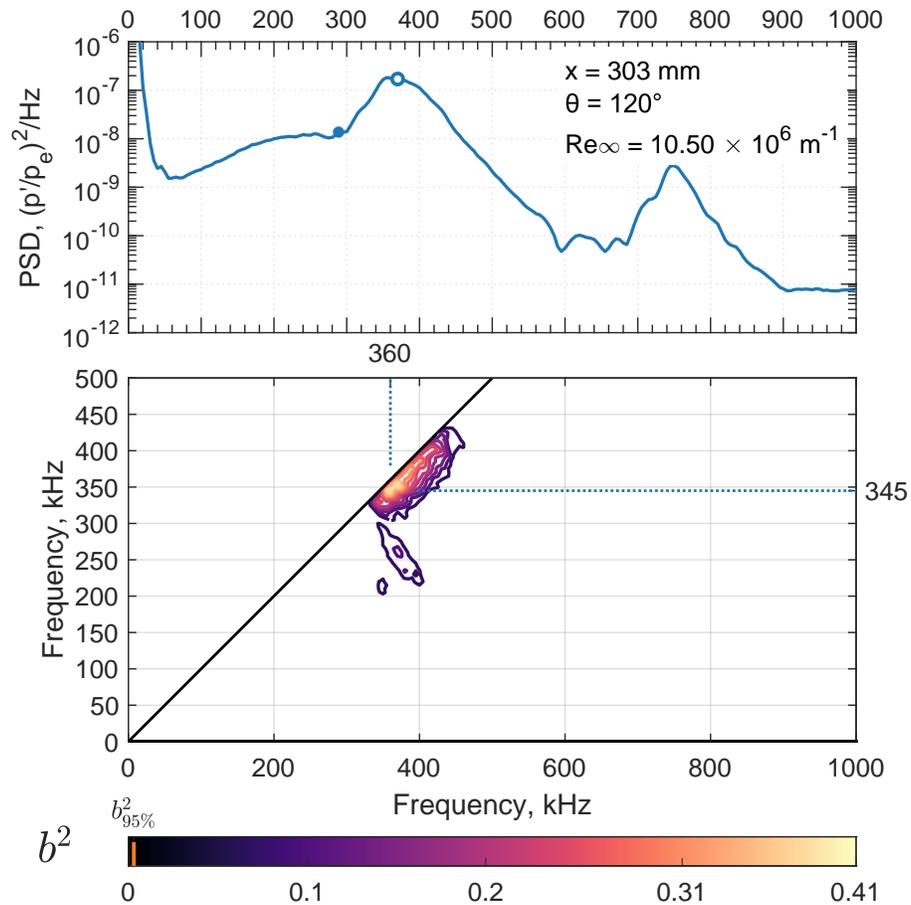


Figure 5.30. Harmonic generation in the HF mode. Run 1618.

the largest peak would be at least partially aliased. However, there is very little power at 1 MHz, so either there is no harmonic or it has been sufficiently suppressed by the scope's low-pass filter.

Strangely, the bicoherence does not show any nonlinear activity at the largest peak in the spectrum. Rather, the significant bicoherence is concentrated along the line $f_1 + f_2 = f_3 \approx 270$ kHz. The maximum bicoherence seems to show coupling between 45 kHz, 225 kHz, and 270 kHz (note the small peak in the spectrum at 270 kHz). The traveling crossflow wave, if it is present, would be in the region around 45 kHz, so it is possible that the bicoherence here is showing nonlinear interaction between traveling crossflow and the second mode. However, neither 45 kHz nor 225 kHz are apparent in the spectrum, so the true origin of this large bicoherence remains a mystery.

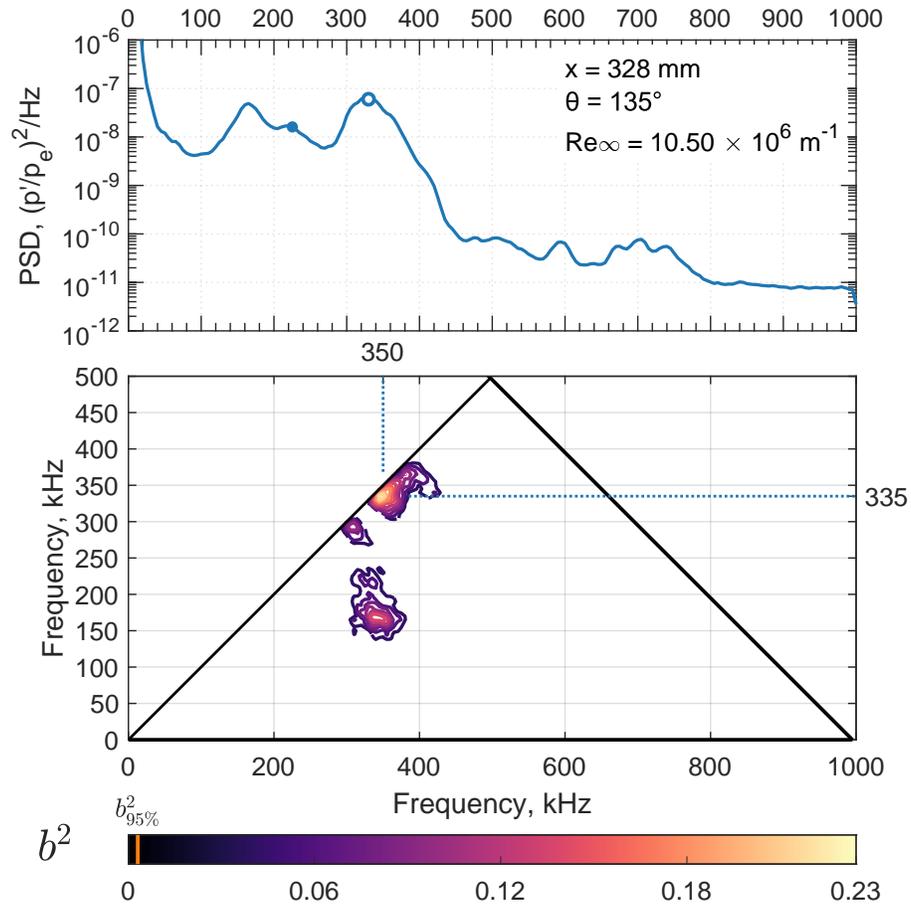


Figure 5.31. Possible nonlinear interaction between the two modes. Run 1640.

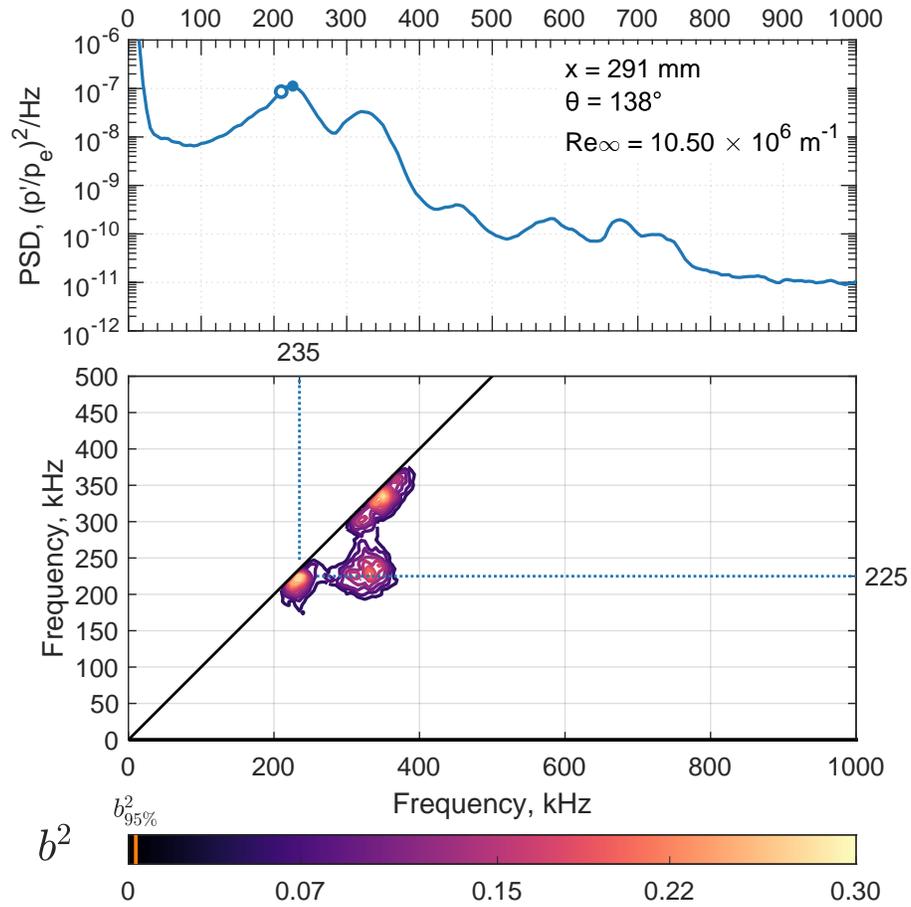


Figure 5.32. Nonlinear interaction between the LF and HF mode. Run 1623.

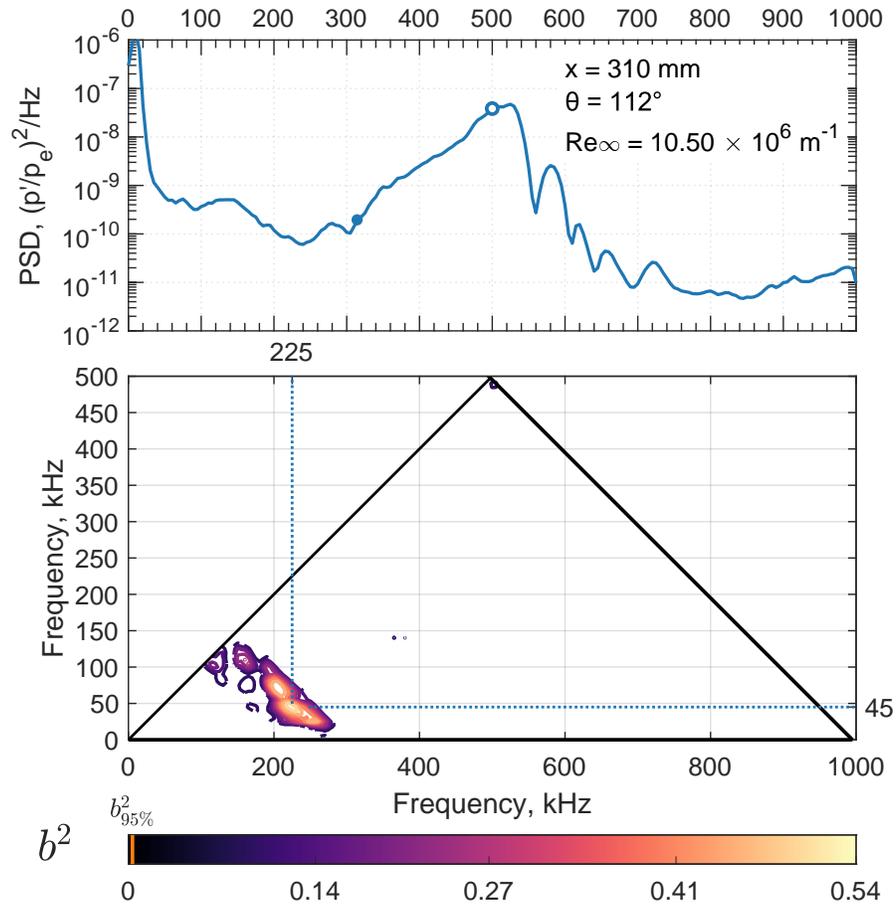


Figure 5.33. Nonlinear interaction between unidentified modes. Run 1632.

5.4 Measurements After a Small Leeward Shift of the Roughness

As previously discussed, the RIM roughness was rotated leeward by about 2° and the measurements were repeated. Chapter 4 studied the changes in the stationary vortices very far upstream to verify that the RIM elements were responsible for the observed pattern of streaks. This section explores the changes in the flow downstream, near breakdown.

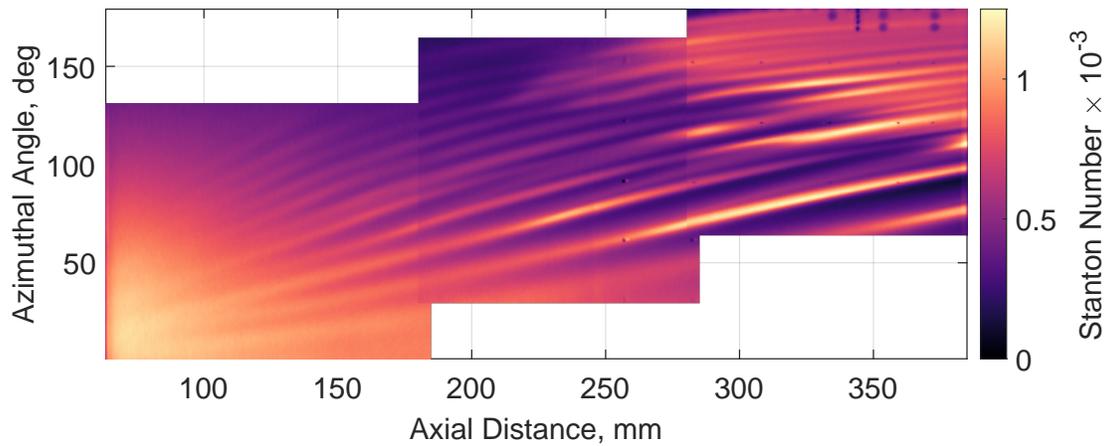
Figure 5.34 shows Stanton number images along the cone for the baseline configuration ((a), also Figure 5.8) and the shifted configuration (b). The range of Stanton numbers is the same between the two cases, and the streak patterns look broadly similar. On closer inspection, several common structures can be observed in the heat flux. The patterns of the vortices stay essentially the same when the roughness is rotated, except that they shift leeward—following the roughness—and forward on the cone.

The forward shift is more evident in Figure 5.35. Interestingly, when the roughness is rotated, the streak-merging wishbone structures stay within the same azimuthal band (70° – 110°). Because the streak necessarily shifts leeward to follow the rotation of the roughness, the wishbone pattern actually moves forward as well. It is not obvious why this might be the case, especially given that the roughness elements have a lower Re_k when they are rotated leeward, so the initial amplitudes of the streaks should be smaller. It is possible that the reduction in A_0 is balanced by increased growth rates due to stronger crossflow in this region, as well as the change in the neutral point with azimuthal angle.

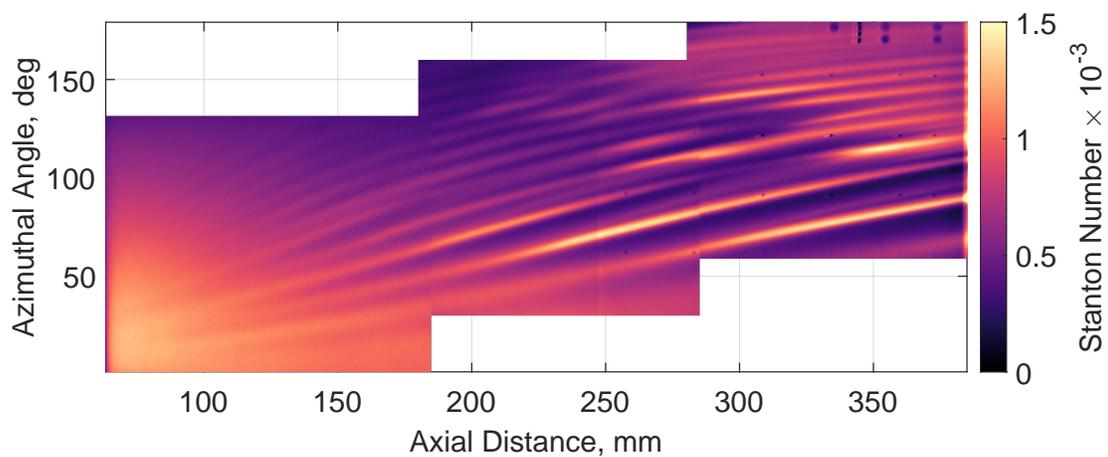
The breakdown of the secondary instabilities was also measured for the shifted case. Figure 5.36 shows contours of the secondary instability amplitude for the baseline (orange) and shifted (blue) cases; the contour levels are the same for each case. It seems that the lobes of secondary instability growth follow the same trend as the stationary vortices: the pattern remains fundamentally similar, but is shifted forward on the cone.

5.5 Summary of RIM Measurements

The measurements described in this chapter successfully serve two purposes. The first is to provide a detailed set of measurements of the nonlinear breakdown of a hypersonic crossflow boundary layer with well-defined and well-controlled inputs. These measurements can be used



(a) Baseline configuration. Images from three runs, left to right: Run 1603, Run 1602, Run 1601.

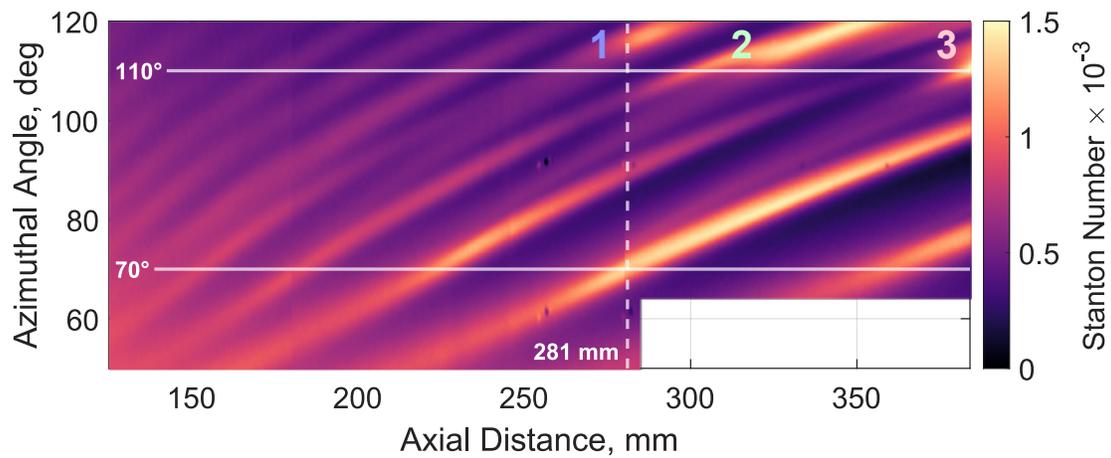


(b) Shifted configuration. Images from three runs, left to right: Run 1604, Run 1605, Run 1606.

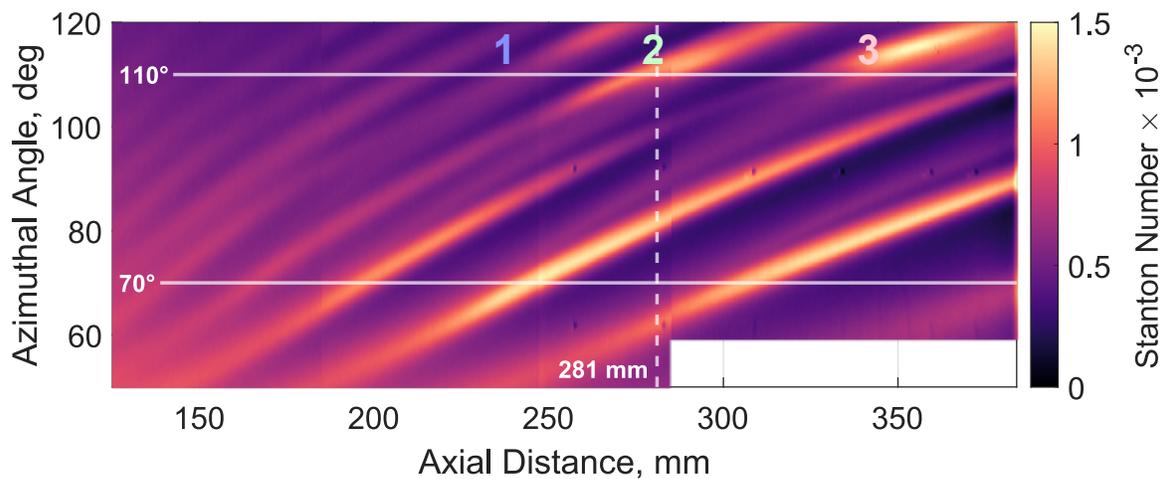
Figure 5.34. Unwrapped images of the Stanton number for both roughness configurations. $Re_\infty = 11.2 \times 10^6 \text{ m}^{-1}$.

as comparison cases for validating computational simulations, and as a starting point for semi-empirical transition-prediction methods based on the amplitudes of the secondary instabilities.

The second purpose is to begin to understand some of the fundamental physics behind crossflow transition in the hypersonic regime. These measurements are the most detailed yet available of the nonlinear stages of crossflow transition in quiet flow. They have demonstrated that when using the discrete RIM elements the dominant nonlinear mechanism seems to be the modulated



(a) Baseline configuration.



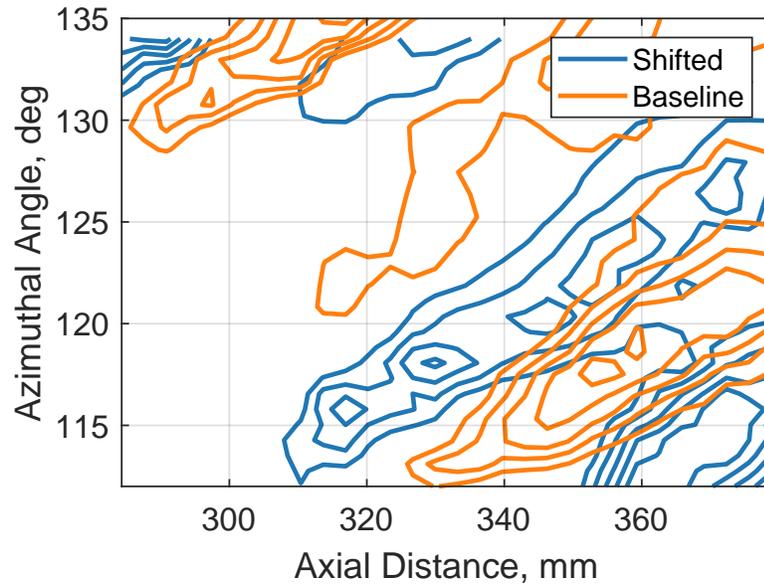
(b) Shifted configuration.

Figure 5.35. Detailed view of Figure 5.34, showing the forward movement of the wishbone structures when the roughness is rotated. The numbers indicate the same structure as it moves.

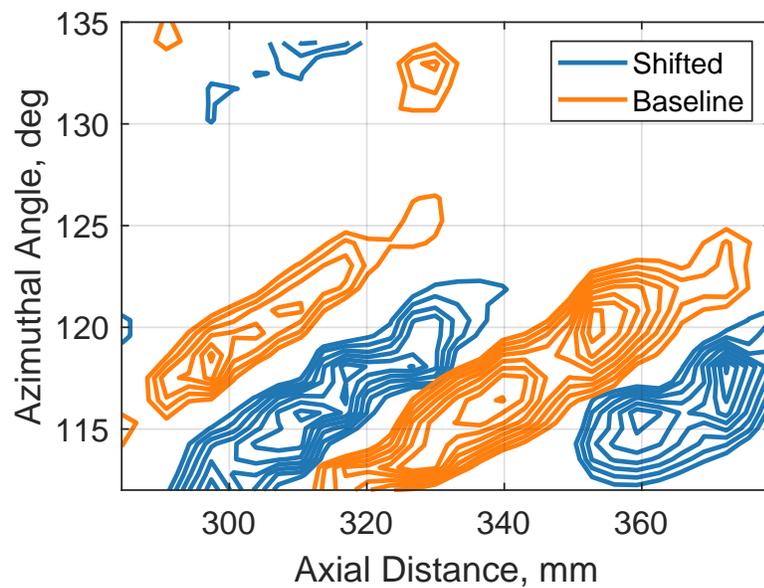
second mode, which has low-frequency and high-frequency components. The two components are spatially localized under different portions of the stationary vortices, but interact nonlinearly in the overlapping regions.

These data have also corroborated computational studies showing a decreasing wavenumber of the stationary vortices downstream. An input of $m = 40$ reduces to around $m = 20$ only 200 mm from the nose.

High-resolution measurements of the stationary vortex amplitudes also reveal interesting behavior that has thus far not been reported. There are two classes of hot streaks (under the troughs of the vortices): streaks that grow monotonically to breakdown, and streaks that peak twice. Furthermore, the first peak is sometimes coincident with the merging of two different vortices, and the location of this merging is azimuthally confined, even when the roughness is rotated. The cause of this behavior is not yet known.



(a) Low-frequency band.



(b) High-frequency band.

Figure 5.36. Comparison of the secondary instability amplitudes for the baseline and shifted cases. $Re_\infty = 10.2 \times 10^6 \text{ m}^{-1}$.

6. MEASUREMENTS USING SMALL-AMPLITUDE DISTRIBUTED ROUGHNESS

This chapter describes measurements made using the Sandblasted roughness insert, with an RMS roughness level of $k_{\text{RMS}} \approx 4 \mu\text{m}$. First, the repeatability of the primary and secondary instabilities is examined. Then, following the template of the previous chapter, measurements are presented of the stationary and traveling crossflow vortices, and finally the secondary instabilities. This chapter discusses only the general characteristics of the breakdown for this type of roughness. In Chapter 7 comparisons are drawn between these data and those of Chapter 5, which yield insight into some of the physical mechanisms involved in hypersonic crossflow-dominated transition.

6.1 Repeatability

As in the previous case using the RIM roughness, it is important to establish that the instability measurements are repeatable before data from multiple runs are synthesized. In addition to the stationary crossflow and secondary instability measurements, the repeatability of measurements of the traveling crossflow waves will also be examined, as they appear to be important to breakdown in this scenario. Due to time constraints, far fewer runs were completed with the Sandblasted insert than with the RIM insert. Consideration of the repeatability is therefore necessarily limited in scope.

As with the data in Chapter 5, several runs were conducted with the Sensor Frustum in different azimuthal orientations, but with the rest of the model fixed. Figure 6.1 shows azimuthal cross-sections of the heat flux from all 15 runs with the Sandblasted roughness insert. Figure 6.1(a) shows data from $x = 299 \text{ mm}$. In general, this far upstream the vortex pattern is quite repeatable. The standard deviation of the 15 runs is below 10% of the mean until close to the lee ray, where there is significant deviation. However, there is a large mushroom-shaped upwelling at the lee ray, the transition of which may not be due to crossflow mechanisms. Thus a lack of repeatability in this area may not be important to the present study.

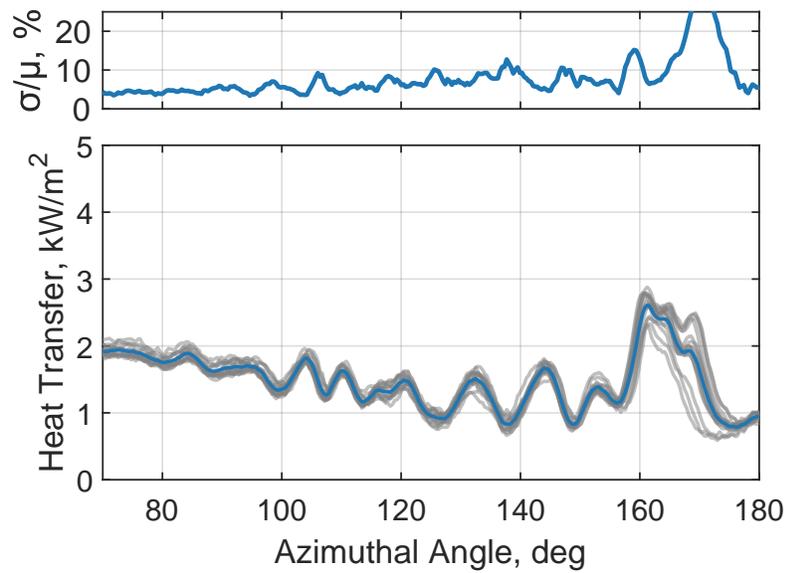
Farther downstream, the heating pattern becomes less repeatable. There is significant variation at around 100° and 140° , among other locations. The locations of the peaks and troughs in the heat flux seem to be fairly consistent, but their amplitudes in any given run can vary by 20 % or more of the mean. The region of highest variability, $120\text{--}140^\circ$ is also the region measured with the PCBs, so a lack of repeatability in this area may be connected to the presence of the sensors. Why the sensors would cause such variability in this case but not when using the RIM insert is an open question. It is very odd that the flow is so repeatable upstream, yet only 50 mm downstream there is this much variation. The cause of this extreme sensitivity is unknown.

Figure 6.2 shows the mean heat flux at $x = 343$ mm from the 15 runs in Entry 16, compared to an azimuthal cross-section at the same location and Reynolds number, but after a model change and a repositioning of the camera. This comparison is analogous to that shown in Figure 5.2. Again, the peaks and troughs generally seem to line up, but the amplitudes are quite different between the two setups. The fact that the locations of the streaks are roughly the same is encouraging: it indicates some consistency in the origin of the stationary vortices between runs. The lack of amplitude repeatability, however, is problematic.

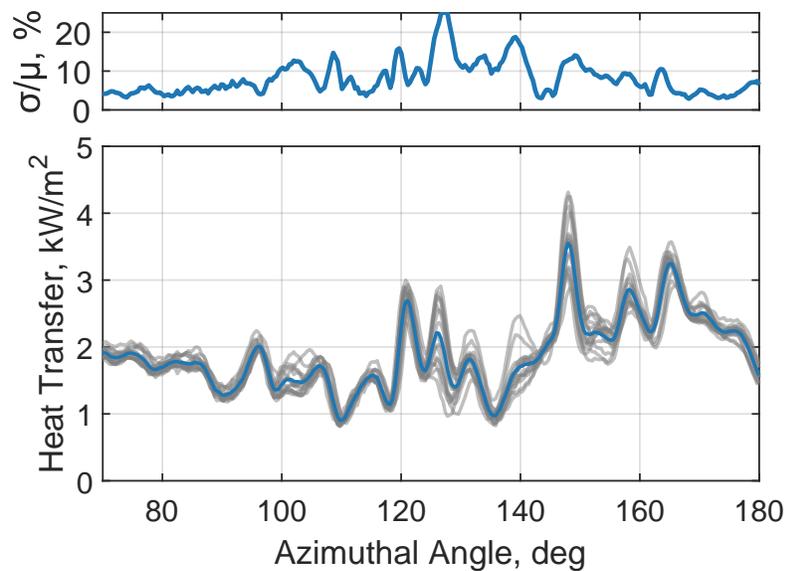
Though these results are mixed, the repeatability of the underlying flow can also be examined using surface pressure sensors. As will be shown in the next few figures, the secondary instability and traveling wave repeatability seems to be almost as strong as when using the RIM roughness. It is ultimately unclear, then, to what extent these heat-transfer results actually indicate some issue with the experiment. Nevertheless, until future experiments and computations can provide some more insight, most of the conclusions that are drawn from the data in this chapter are necessarily qualitative.

The traveling crossflow waves can be measured at around 45 kHz using surface pressure sensors. Kulites are much more suited to measurements at this low frequency than PCBs, so most of the analysis of traveling crossflow will be from Kulite data. Figure 6.3 shows spectra from Kulites 1 and 4 (both at around $x = 330$ mm) for two sets of repeat runs. Figure 6.3(a) shows data from a low azimuthal angle, $\theta = 90^\circ$, for two consecutive runs. In this case the spectra are almost on top of one another, indicating a high degree of repeatability when the model is not reconfigured. The short, narrow peaks at 225 kHz and 300 kHz are the resonances of each Kulite. Figure 6.3(b) shows spectra from the same Kulites at a higher azimuthal angle ($\theta = 135^\circ$) for non-consecutive

runs. The Sensor Frustum was rotated to several different azimuthal orientations between these two runs. Here the repeatability is not as clear. For Kulite 4 (orange line), the traveling wave peaks at 45 kHz and the shape of the spectra are similar, but the spectrum from Run 1718 is uniformly lower amplitude than Run 1711. For Kulite 1 (blue line), there is an additional instability, visible as a small bump in the spectra, at 100 kHz in Run 1718 which is not present in Run 1711. The reason for these discrepancies is unclear, but could be tied to the whatever is driving variation in the stationary vortex measurements. An alternative explanation is that the azimuthal angle was not exactly the same during these two runs, and due to the presence of large stationary vortices the traveling crossflow modes are highly localized in nature. This would also explain the excellent repeatability at lower azimuthal angles: the crossflow vortices are much weaker around 90° than 135° . Unfortunately, these Kulite measurements were made without IR or TSP, as mentioned in Chapter 5, so this explanation can not be confirmed. Note also that at this higher azimuthal angle the sensor resonances appear much larger in the spectra due to the higher overall pressure fluctuation levels.



(a) Slices at 299 mm.



(b) Slices at 343 mm.

Figure 6.1. Azimuthal cross-sections of the heat flux from 15 runs at two different axial stations. Each gray line is a slice from one run; the dark blue line is the mean for all the runs. The upper sub-plot shows the standard deviation at each point as a percentage of the local mean. $Re_\infty = 11.1 \times 10^6 \text{ m}^{-1}$.

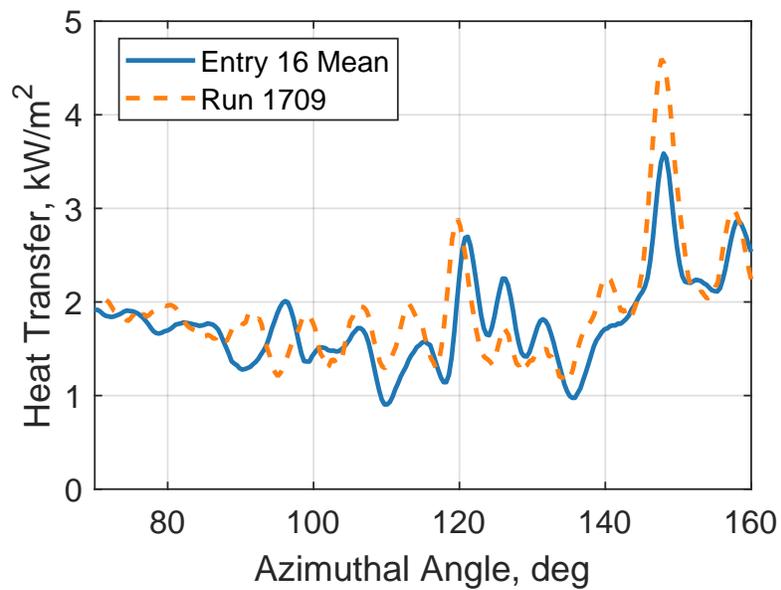
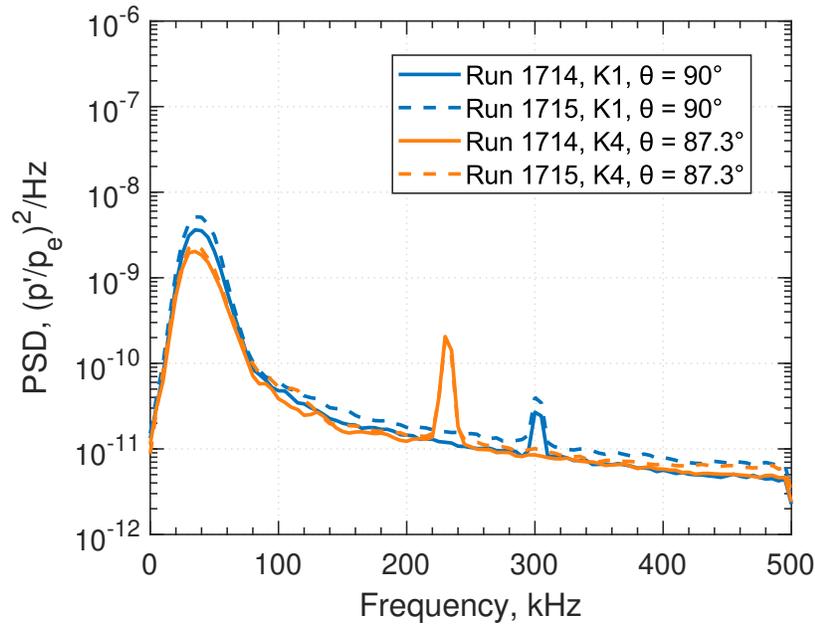
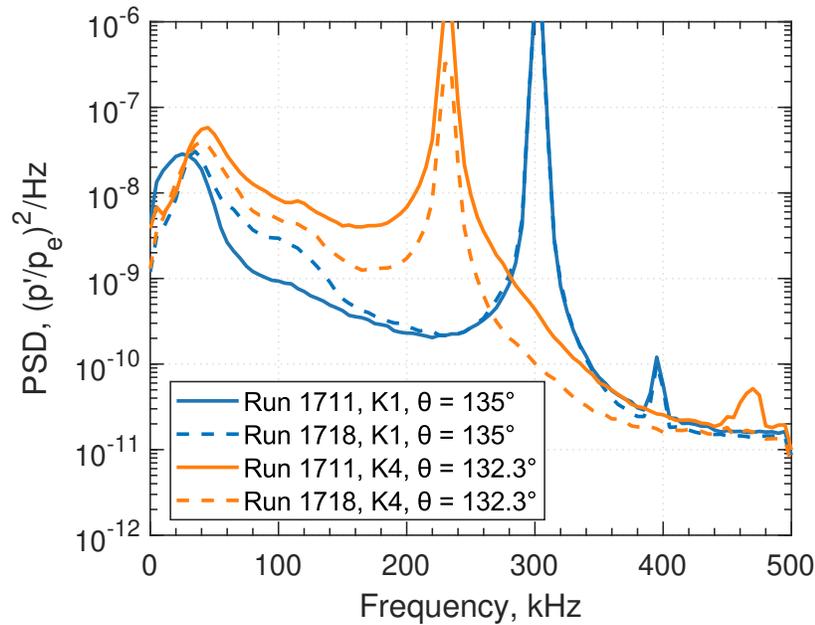


Figure 6.2. Comparison between the mean data from Entry 16 (Figure 6.1(b)) and data from after a change in the model and camera position (refer to Figure 5.2 for the same comparison with the RIM insert). The stationary vortices generated using the Sandblasted insert are evidently far more sensitive to upstream conditions than those from the RIM insert.



(a) Spectra from two consecutive runs, with no rotation of the Sensor Frustum.



(b) Spectra from two non-consecutive runs, but at nominally the same azimuthal angle.

Figure 6.3. Repeatability of the traveling crossflow wave measurements from the Kulite sensors. The origin of the discrepancies between runs 1711 and 1718 is unclear. The narrow peaks at 225 kHz and 300 kHz are the resonances of the sensors. $Re_\infty = 11 \times 10^6 \text{ m}^{-1}$.

PCBs were also used to measure the traveling crossflow waves and a higher-frequency secondary instability. Figure 6.4 shows spectra from three PCBs for two non-consecutive runs at nominally the same angle. The repeatability for the PCB measurements is excellent. The PCBs are twice the diameter of the Kulites, so they are less sensitive to small uncertainties in azimuthal location. This is further evidence that the Kulite repeatability issues in Figure 6.3 are rooted in the uncertainty in azimuthal position, rather than some issue with the repeatability of the flow itself.

As before, it is important to understand the repeatability of the nonlinear mechanisms as well. Figure 6.5 shows two plots of the bicoherence which correspond to the two PCB 18 spectra in Figure 6.4. The repeatability of the bicoherence is undeniably weaker than when using the RIM insert, but it is still quite good. Though the maximum bicoherence differs by about 30 %, the regions of high bicoherence are very similar. Interestingly, the bicoherence seems to show harmonic generation from both the traveling crossflow wave (45 kHz) and the secondary instability (150 kHz), but also interaction between these two frequencies as well as broad-spectrum interaction up to 400 kHz. Interactions of such a wide variety were not observed when using the RIM insert, and will be discussed in detail in Section 6.3.1.

The repeatability of the PCB measurements of the secondary instability was also examined using the three repeatability pairs of PCBs, as discussed at length in Section 5.1.2. Figure 6.6 shows four spectra from PCBs 13 and 21 ($x = 335$ mm). Recall that PCB 21 is offset from PCB 13 by 6° , so for each azimuthal angle in the figure, the spectra are from two different runs. The spectra seems to repeat quite well, except for resonance of PCB 21 at around 450 kHz, which is present at all azimuthal angles. However, there were only 8 overlapping runs with the Sandblasted insert as opposed to the 24 such runs with the RIM insert, so this analysis is not as definitive.

Figure 6.7 provides the pressure fluctuation amplitudes for the three repeatability pairs of PCBs as a function of azimuthal angle. In this case the low-frequency (LF) band is the traveling wave band, 25–75 kHz. The high-frequency (HF) band contains the secondary instability, 80–250 kHz. Here a constant band of frequencies was used to calculate the amplitude, unlike the axially and azimuthally dependent band used for the RIM case. This is because the frequency of the secondary instability in this case varies much less than when using the RIM insert, which will be illustrated in Section 6.3. For the admittedly small region of overlap between the main and offset rays, 129° – 136° , the amplitudes in both bands show very good repeatability.

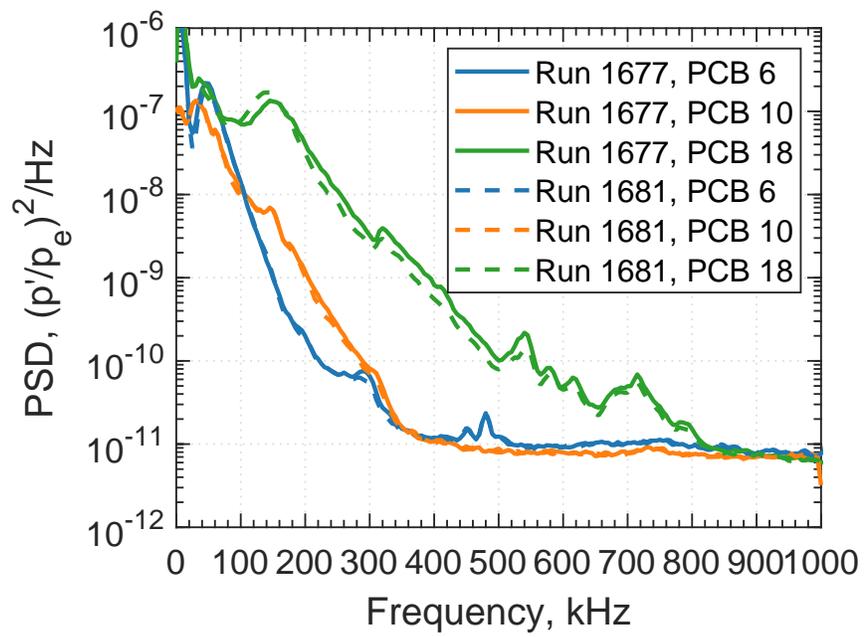
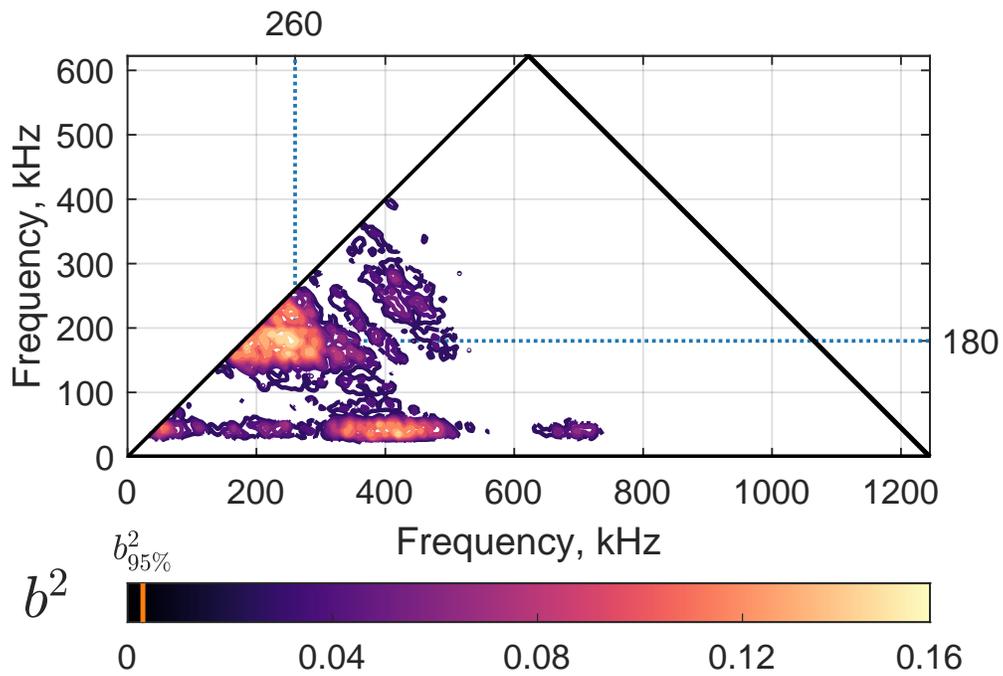
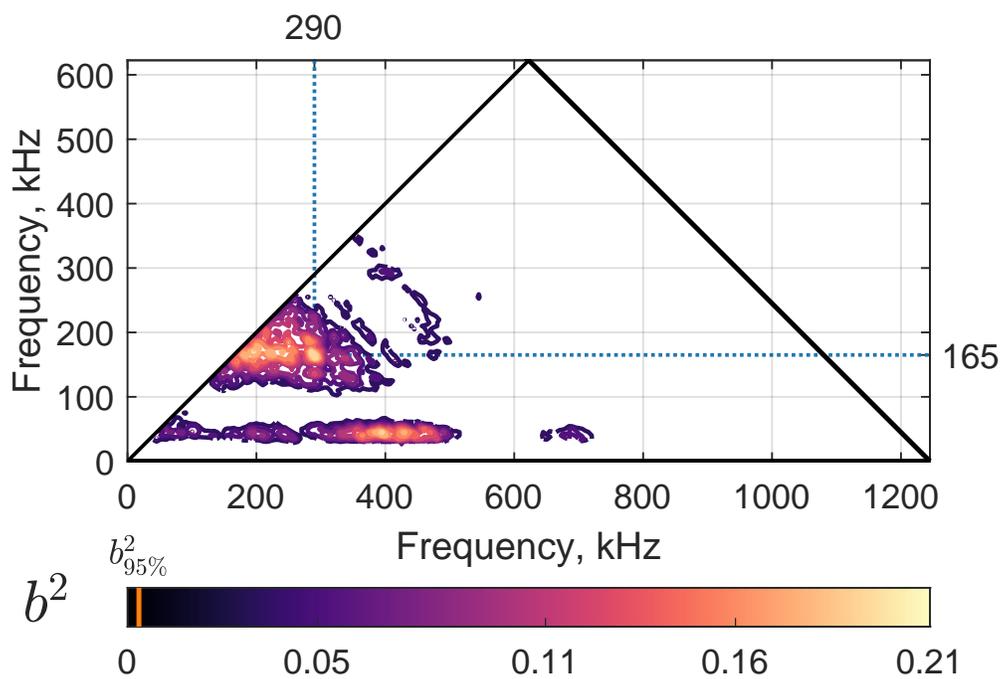


Figure 6.4. Repeatability of the traveling crossflow and secondary instability measurements for two runs at the same azimuthal angle, recorded by PCB sensors. $\theta = 137^\circ$ and $Re_\infty = 11.1 \times 10^6 \text{ m}^{-1}$.



(a) Run 1677.



(b) Run 1681.

Figure 6.5. Repeatability of the bicoherence for PCB 18, from the same data as Figure 6.4.

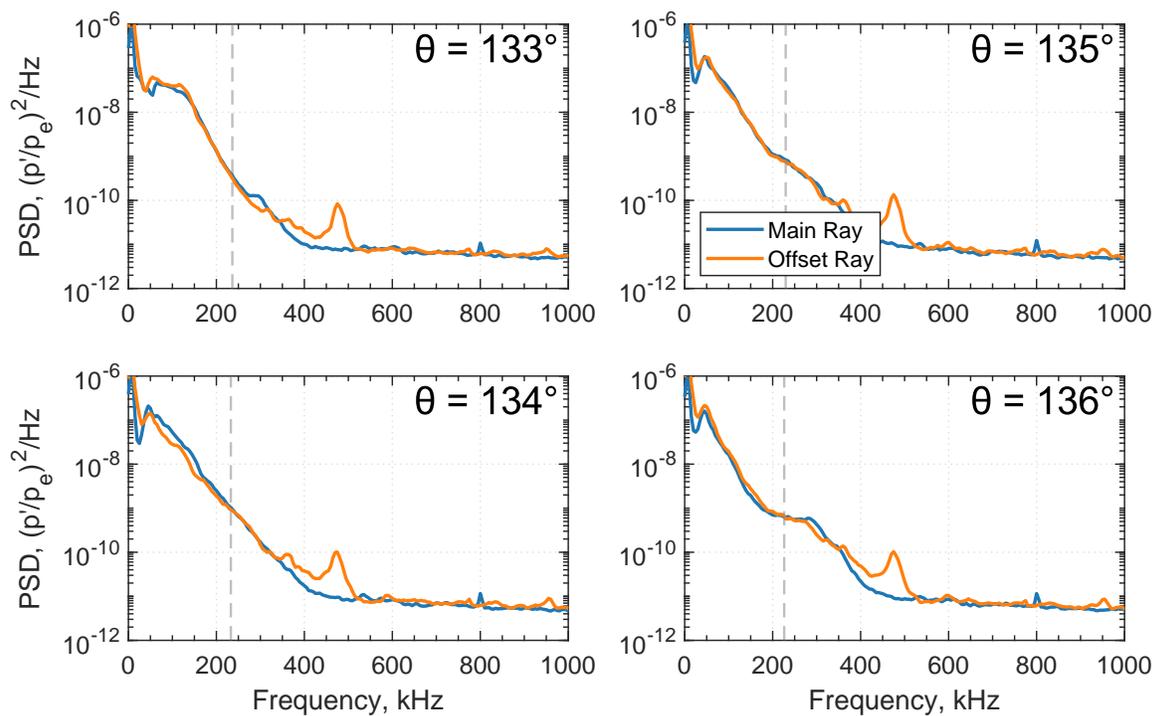


Figure 6.6. Repeat spectra from PCBs 13 and 21, from the same azimuthal angles. Recall PCB 21 is offset 6° windward from PCB 13, so the measurements at each azimuthal angle are from different runs. The small peak in the offset ray (orange line) at around 425 kHz is likely a sensor resonance.

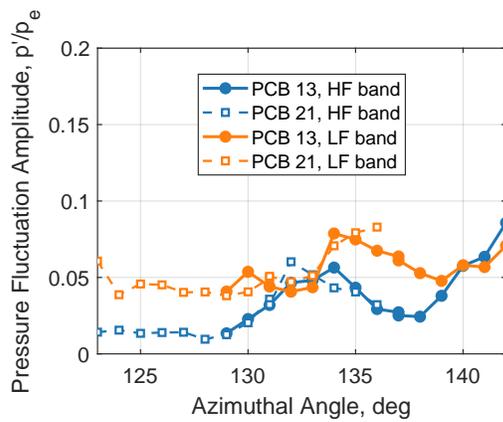
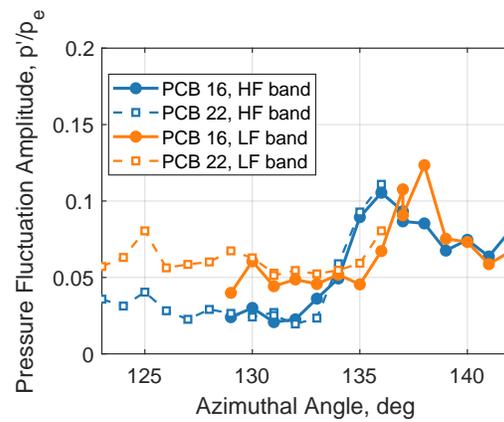
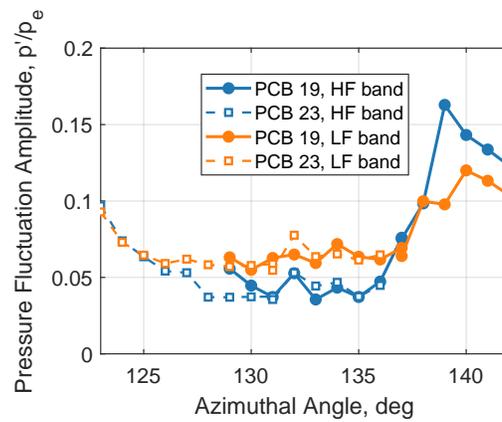
(a) PCBs 13 and 21 ($x = 335$ mm)(b) PCBs 16 and 22 ($x = 354$ mm)(c) PCBs 19 and 23 ($x = 373$ mm)

Figure 6.7. Pressure fluctuation amplitudes for the three repeatability pairs. The low and high frequency bands are 25–75 kHz and 80–250 kHz, respectively. $Re_\infty = 11.1 \times 10^6 \text{ m}^{-1}$.

6.2 Primary Instability Measurements

6.2.1 Stationary Vortices

Before using the Sandblasted insert, it was verified that the results looked similar to the “natural” transition using the nominally smooth insert. Figure 6.8 shows a comparison between heat flux images of the cone when using each insert. Recall that the nominally smooth insert actually has some inherent waviness from the machining process (see Figure 2.11). The two flows look qualitatively similar, except the Sandblasted case leads to higher heat flux—which is expected given the higher level of roughness. This is in contrast to the surface heat flux when using the RIM insert (Figure 5.8), which is much less uniform and has streaks at a lower wavenumber.

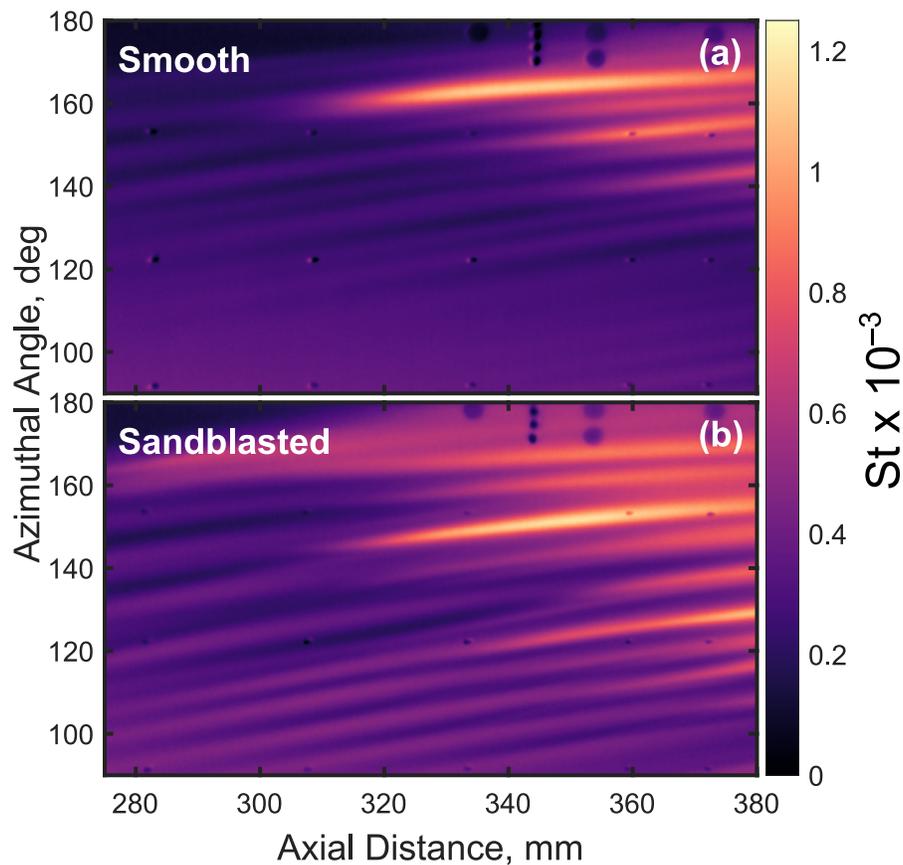


Figure 6.8. Comparison between heat flux images using the nominally smooth insert and the Sandblasted insert. $Re_\infty = 11 \times 10^6 \text{ m}^{-1}$. Smooth: Run 1416, Sandblast: Run 1709.

Given the results of the previous section, it is not entirely clear how repeatable the streak pattern is between runs—at least at higher azimuthal angles—so measurements of the streaks from different runs must be analyzed with care. Figure 6.9(a) shows three heat-transfer images from three separate runs stitched together. Very far upstream there is no apparent modulation of the heat flux, indicating that the stationary vortices are too weak in this region to measure via IR thermography. This is not entirely surprising given the very low RMS roughness, and is in contrast to the RIM results. Farther downstream, streaks start to appear between 200 mm and 250 mm. These streaks grow quite rapidly, and breakdown is visible for the most leeward streaks by the end of the model. The wavenumber of the streaks is quite high: at 90° there are 10 visible streaks, which gives an extrapolated wavenumber of 40 per circumference. Recall that the downstream wavenumber using the RIM insert is roughly 20. At the edge between the middle and aft images, there is a noticeable discontinuity in the streaks at high azimuthal angles. This discontinuity may be the result of the aforementioned repeatability issues. Figure 6.9(b) shows the Stanton number images. The maximum St is roughly the same as in the RIM case, about 1.25×10^{-3} . Unlike in that case, however, the Stanton normalization does not seem to reduce the discontinuity between images.

The heat flux perturbation is provided in Figure 6.9(c). The perturbation is the heat flux image from Figure 6.9(a) band-pass filtered between 18 and 180 waves per circumference, to eliminate the slow variations in the heat flux of the mean flow. The perturbation appears much more regular than the RIM case, with most of the splitting, merging, and other irregularity confined downstream near the lee ray. There do not appear to be any of the double peaks observed with the RIM insert.

The wavenumber spectrum of the heat flux images is provided in Figure 6.10. Recall that this spectrum is created by taking the power spectral density of each azimuthal cross-section, and converting the frequency from waves per degree to waves per circumference. The spectrum has been normalized by its maximum. In agreement with the simple analysis above, the wavenumber at the aft end of the cone is around 40, though there is also a peak at around 20. The lower wavenumber indicates that there is some modulation of the amplitudes of the vortices, affecting every other vortex; it could also indicate the presence of a subharmonic wavenumber. A downstream wavenumber of around 40 is close to earlier measurements by Ward [38] and Craig [3] on a nominally smooth cone.

The computed wavenumber along a particular streak for an upstream wavenumber of 220 is provided in Figure 6.10 as the green line (from Moyes et al. [19], Figure 4(b)). Having an understanding of the change in the wavenumber is especially critical for controlling crossflow, which may require subcritical wavenumbers (i.e. higher than the naturally most-amplified) at the upstream location.

The peak heat flux was extracted along several of the streaks, which are shown in Figure 6.11(a), numbered from wind to lee. The Stanton number for a few of the streaks compared to the computation is plotted in Figure 6.11(b). In this case, the growth of the vortices appears to be very steady until breakdown.

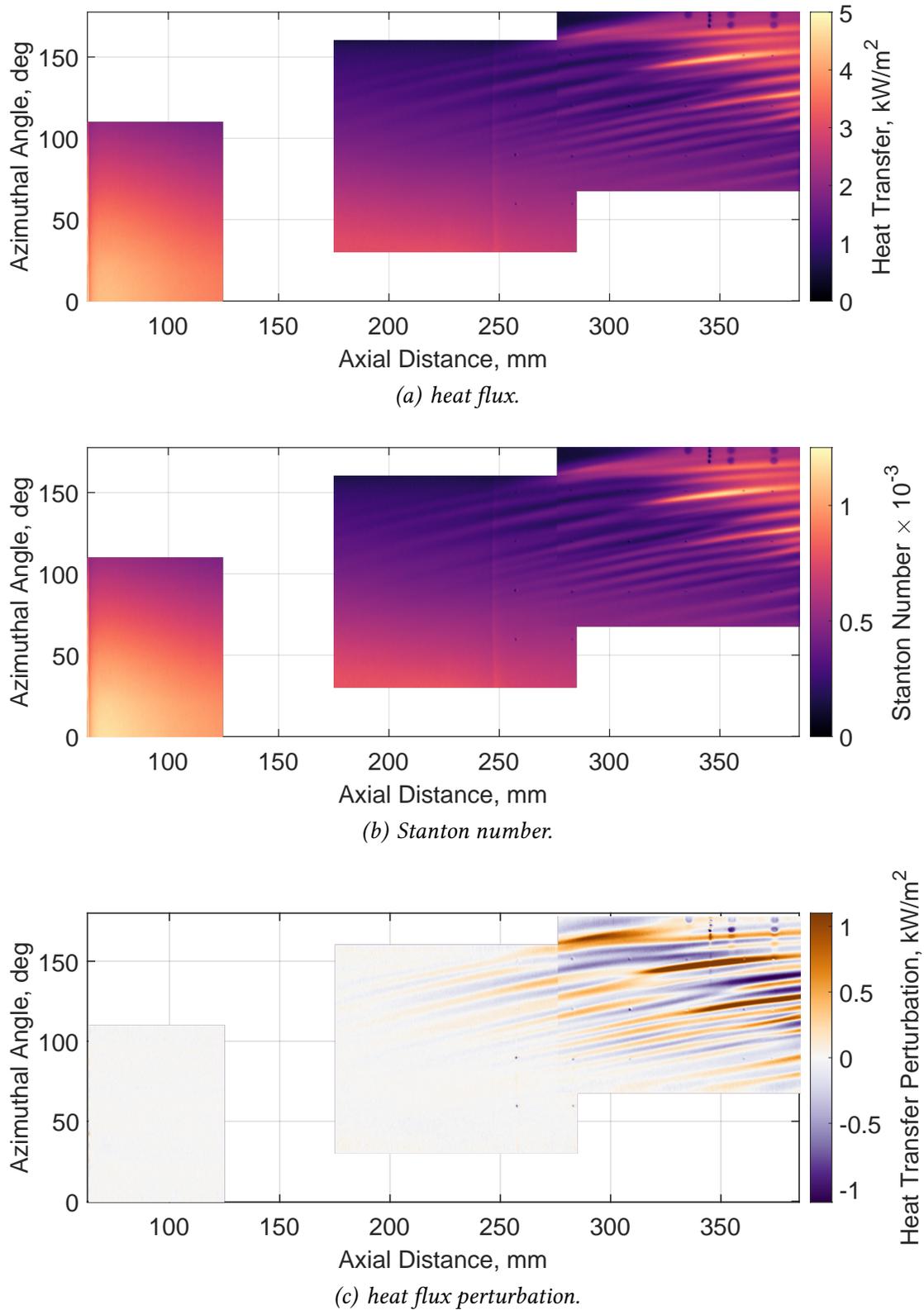


Figure 6.9. Surface heating caused by the stationary vortices. $Re_\infty = 11 \times 10^6 \text{ m}^{-1}$. Images from three runs, left to right: Run 1705, Run 1710, Run 1709.

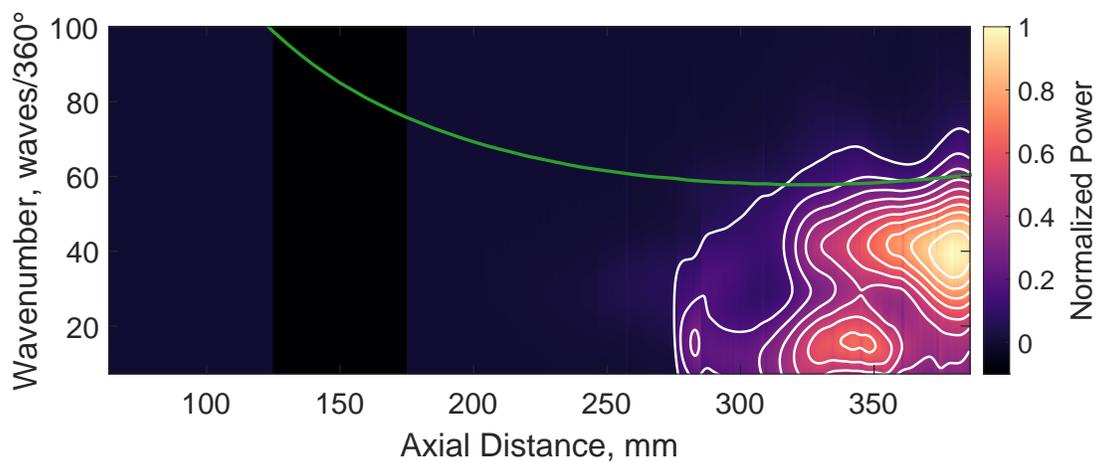
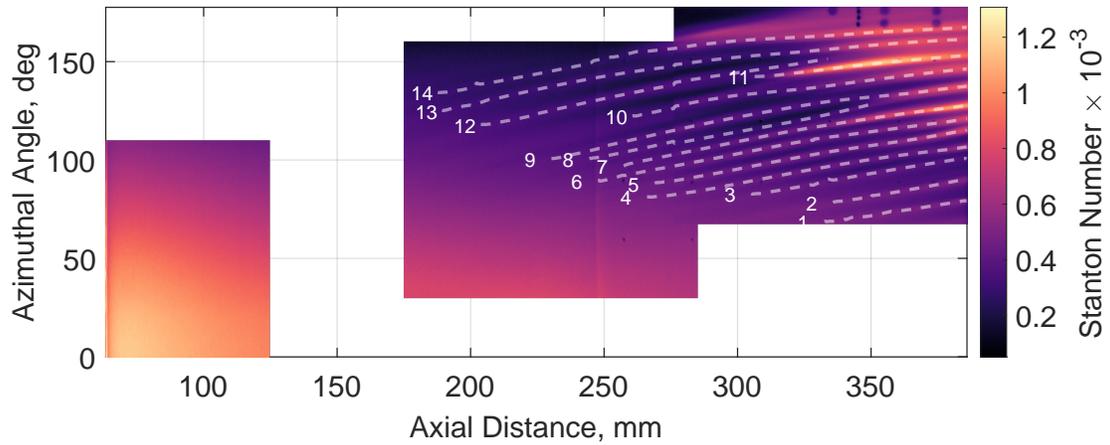
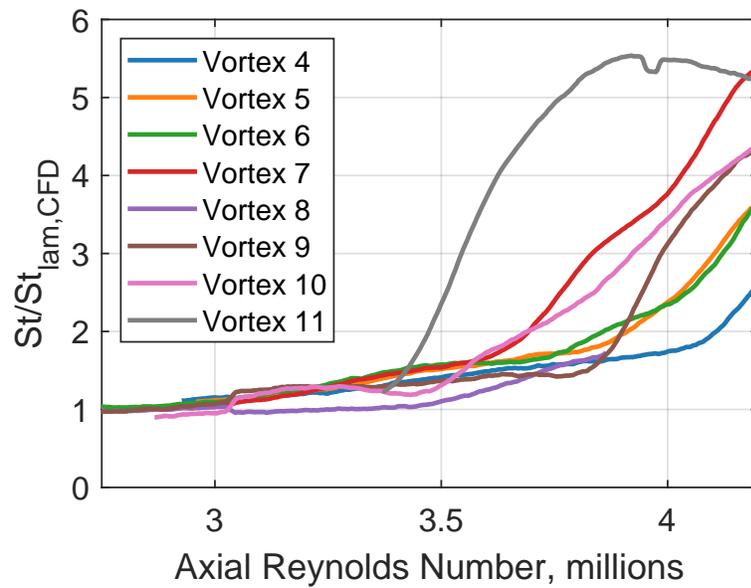


Figure 6.10. Wavenumber spectrum normalized by the maximum power. The wavenumber is the number of waves over the entire circumference of the cone, extrapolated from the imaged portion. The green line is the computed wavenumber from Moyes et al. [19] along a particular vortex for an upstream wavenumber of 220 (the present experiments have an unknown upstream wavenumber).



(a) Paths of peak heat flux along streaks.



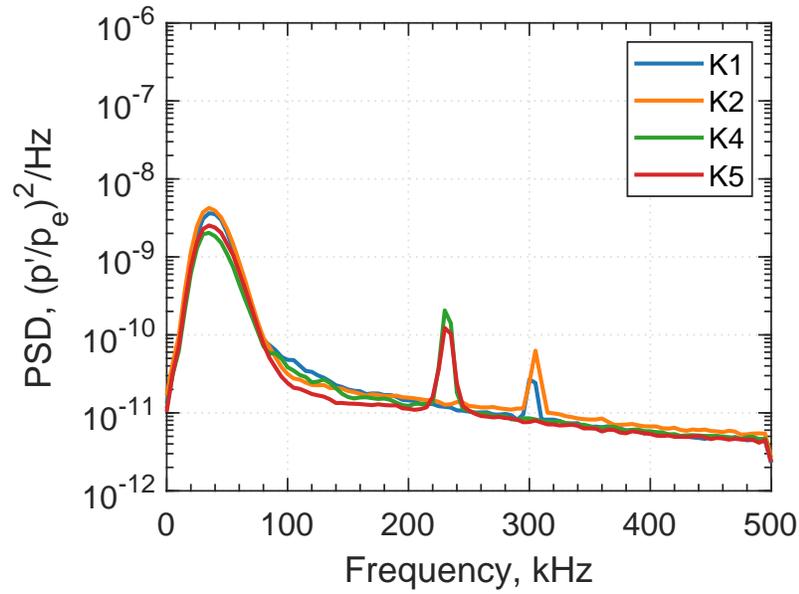
(b) Stanton number ratio along the streaks in (a).

Figure 6.11. Amplitude growth of the streaks.

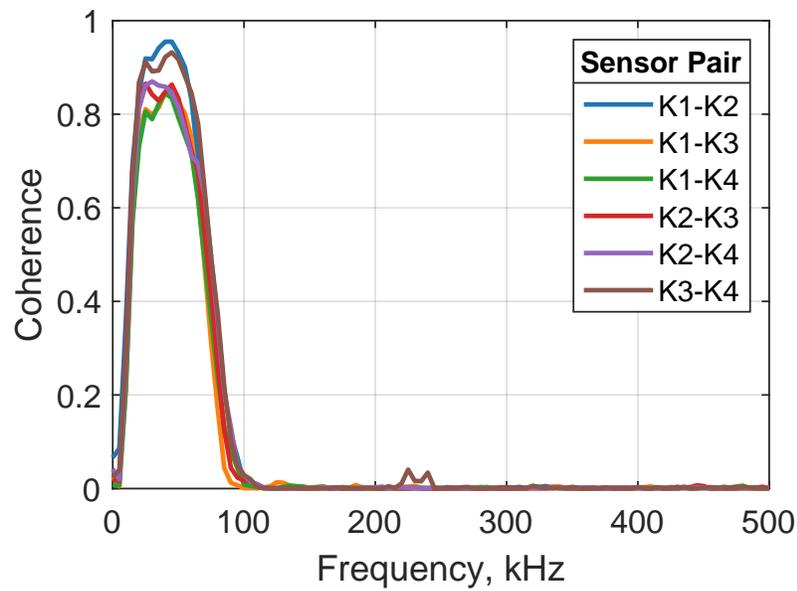
6.2.2 Traveling Vortices

The traveling crossflow vortices are readily apparent in the surface pressure fluctuations when using the Sandblasted roughness. Therefore it is important to understand the behavior of the traveling vortices on their own, to provide context for the following discussion of their nonlinear interactions. To that end, the Kulites were placed at 90° from the wind ray to measure the wave properties of traveling crossflow in a region minimally affected by the presence of the large stationary vortices. Figure 6.12 shows the power spectra of the four Kulite sensors at this location, as well as the coherence between all possible sensor pairings. The PSDs show very clear peaks at around 45 kHz, consistent with past measurements of traveling crossflow on this geometry. In addition, there is high coherence in this band between all the sensors. This is in contrast to the results when using the RIM insert (Figure 5.16(b)), where there was only non-zero coherence between Kulites 1 and 2.

The propagation angle and phase speed of the traveling vortices can be calculated using the method described in Section 3.7.2. In this case, the angle is measured with respect to the coordinate system shown in Figure 3.7, with a positive, acute angle indicating propagation downstream and toward the lee ray. Figure 6.13 shows the calculated wave propagation angle and phase speed as a function of frequency within the traveling crossflow band. Similar data from Ward [38] was extracted from his Figure 6.5 and is plotted as the open circles. The Ward data and the properties from the present experiment agree quite well. The phase speed of the traveling vortices are between roughly 100 m s^{-1} and 300 m s^{-1} , or about 10–30 % of the edge velocity. The propagation angles are negative and acute, indicating the constant-phase fronts of the traveling vortices propagate downstream from lee to wind, consistent with previous experiments and computations.



(a) Power Spectra.



(b) Coherence.

Figure 6.12. Spectral quantities of the Kulite measurements at a low azimuthal angle ($\theta = 90^\circ$, $x = 330 \text{ mm}$, $\text{Re}_\infty = 11 \times 10^6 \text{ m}^{-1}$). Run 1714.

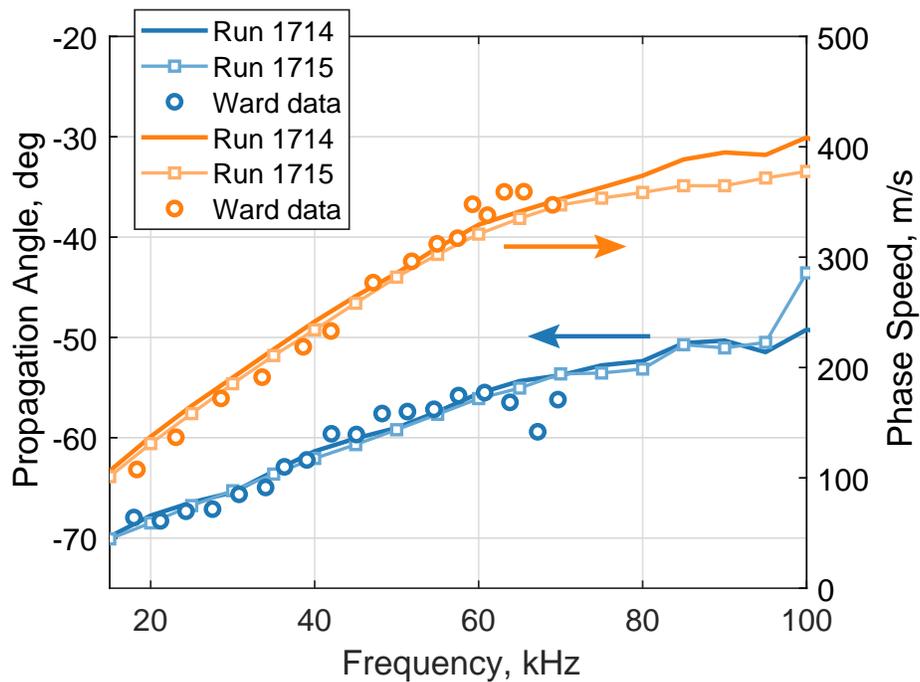


Figure 6.13. Wave properties for traveling crossflow at 90° and $Re_\infty = 11 \times 10^6 \text{ m}^{-1}$. Runs 1714 and 1715 were performed consecutively, with no rotation of the Sensor Frustum. The Ward data is from Reference [38], and is at $\theta = 90^\circ$ and $Re_\infty = 11.5 \times 10^6 \text{ m}^{-1}$.

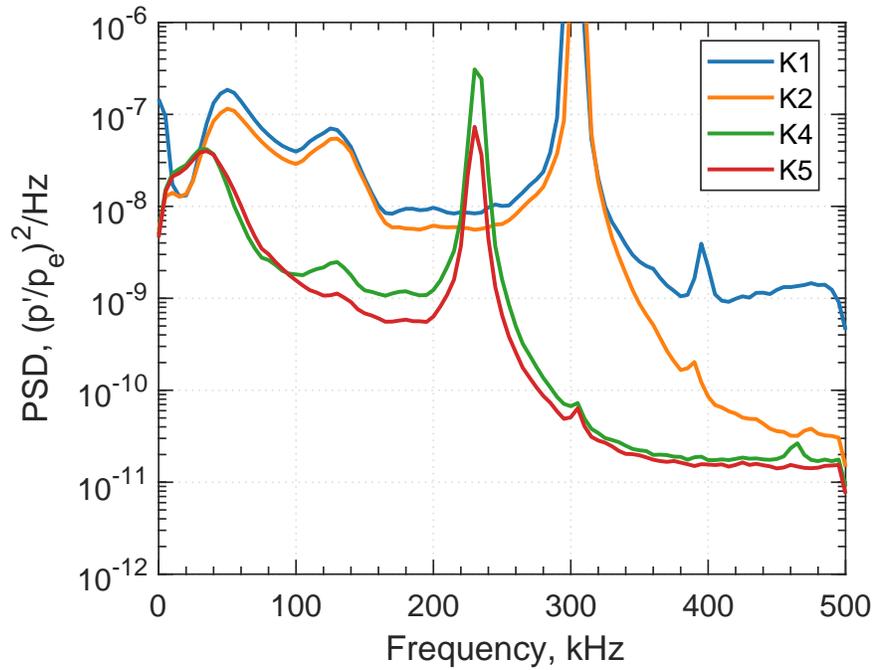
The same test was repeated at $\theta = 142^\circ$. At this higher azimuthal angle there are strong stationary crossflow vortices, so the Kulite measurements might lend some insight into any interactions between the stationary and traveling crossflow modes. Figure 6.14 shows the resulting power spectra and coherences. The traveling crossflow wave is still obvious in the spectrum at around 50 kHz, though the frequency at K1 and K2 is slightly higher than the frequency measured by K4 and K5 (which are 2.75° offset windward). This divergence in frequency is likely due to the presence of the stationary waves.

In addition, a second high-frequency instability has appeared at 125 kHz. Note that the estimated frequency of the second mode in this case is $f_{II} \approx 200$ kHz, so the peak at 125 kHz could be the second mode, but such a frequency ratio is on the low end of the range measured with the RIM insert (see Figure 5.21). It is also possible that this is a type-I secondary instability. The coherence between each sensor pairing shows the modulation of the traveling crossflow by the stationary vortices. Axial pairs K1/K2 and K3/K4 have broadband coherence, whereas other pairs only have limited coherence at the peak frequencies of the two modes.

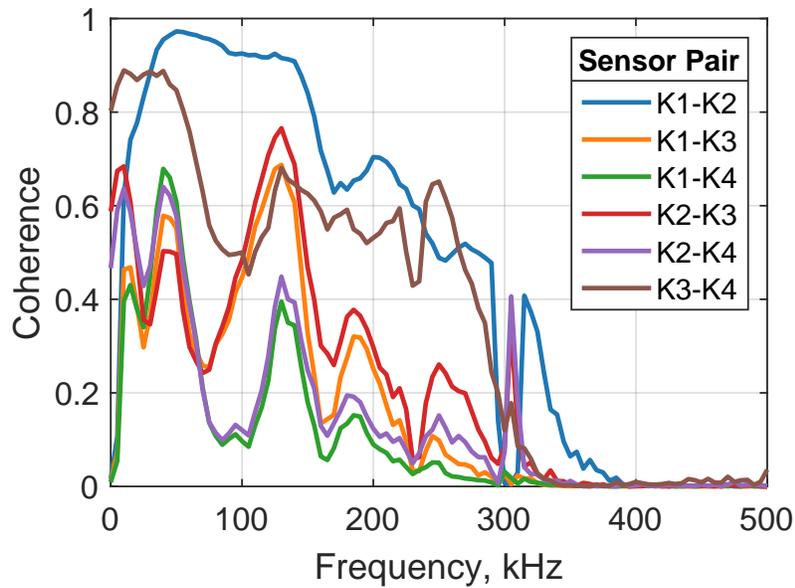
The wave properties of the two instabilities were calculated as before. Figure 6.15 shows the propagation angle and phase speed of the two modes in regions where the coherence is greater than 0.4. The dashed lines are the angle and speed from the data at $\theta = 90^\circ$, in Figure 6.13, above. The traveling wave propagation angle at the higher azimuthal angle is oriented slightly more windward (more negative) than at lower azimuthal angles, but they are very close. The phase speed of traveling crossflow at the two azimuthal locations is almost identical. The high-frequency instability has a much shallower propagation angle, around -30° , and is much faster, traveling at around 80–90 % of the edge velocity. Figure 6.16 illustrates the propagation paths of the two high-frequency instabilities relative to the path of the stationary vortices. The St image of the cone has been unrolled so that the image is in the same arclength reference frame used for the calculation of the traveling wave properties. Traveling crossflow moves almost orthogonally to the stationary vortex, whereas the higher frequency mode is slightly less inclined.

It is interesting to note that at low speeds the type-I secondary instability manifests as small vortices inclined with respect to the stationary vortex. See for example the PIV data from Serpieri and Kotsonis [111] and the DNS from Wassermann and Kloker [112]. It is possible, however, that there is an oblique second mode in this region, and so this is not alone definitive evidence that the

peak at 125 kHz is in fact a shear-layer secondary instability as opposed to a modulated second mode.



(a) Power Spectra.



(b) Coherence.

Figure 6.14. Spectral quantities of the Kulite measurements at a high azimuthal angle ($\theta = 142^\circ$, $x = 330$ mm, $\text{Re}_\infty = 11 \times 10^6 \text{ m}^{-1}$). There are two apparent modes of instability. The activity at around 50 kHz is traveling crossflow. The peak at 125 kHz may be a type-I secondary instability. Run 1713.

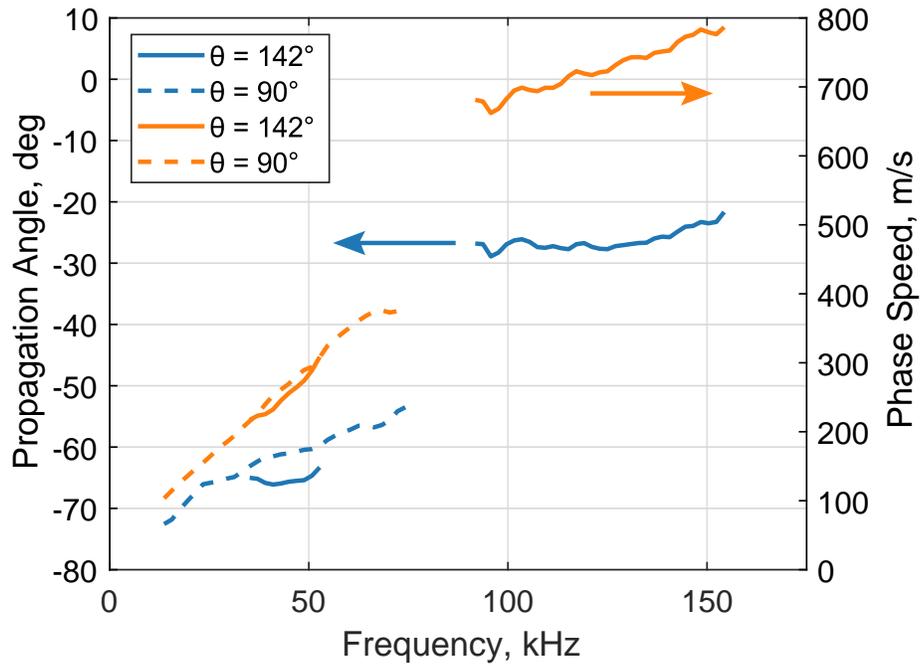


Figure 6.15. Wave properties for the measured instabilities at $\theta = 142^\circ$ and $\text{Re}_\infty = 11 \times 10^6 \text{ m}^{-1}$.

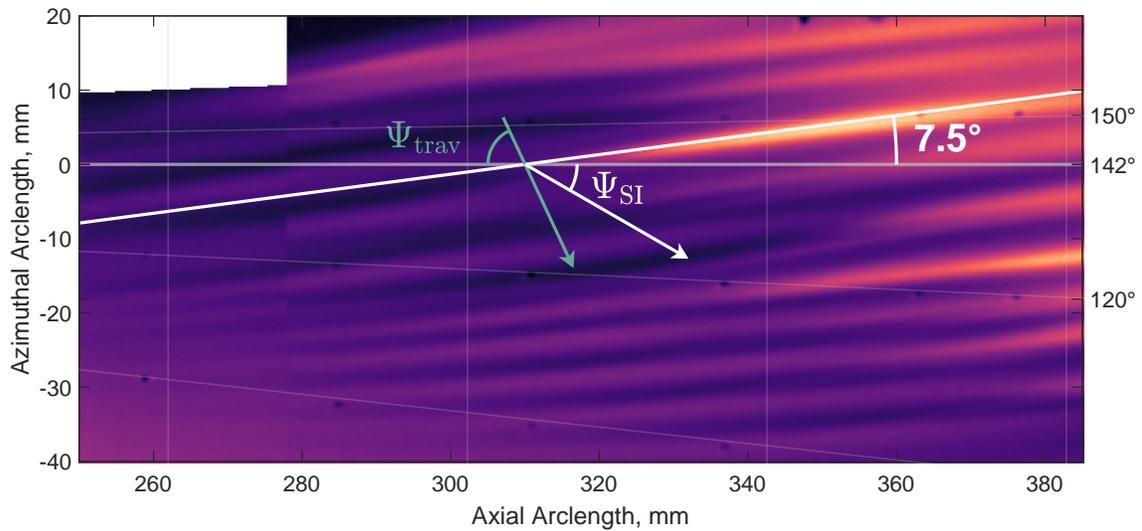


Figure 6.16. Propagation angles of the measured high-frequency instabilities compared to the stationary vortex paths. The line $y = 0$ is centered at $\theta = 142^\circ$. $\Psi_{\text{trav}} = -65^\circ$ and $\Psi_{\text{SI}} = -30^\circ$.

6.3 Secondary Instability Measurements

The secondary instabilities can be analyzed in the same manner as before, with one important difference. When using the Sandblasted insert the traveling crossflow waves are a distinct component of the surface pressure fluctuations, and there appears to be only one mode of secondary instability (as opposed to the low and high-frequency modes discussed in the previous chapter). Thus while there are still two frequency bands of interest, they represent a fundamentally different behavior, which will be further discussed in Chapter 7.

When using the Sandblasted insert the two modes of instability appear at roughly the same frequency regardless of the position of the sensor. This simplifies the analysis, allowing the use of a single, fixed frequency band in which to calculate the amplitude of the various modes. Traveling crossflow, as illustrated in the previous section, occupies the band between 25 kHz and 75 kHz. The secondary instability was usually within the range 80–250 kHz. Note that Craig and Saric [18] measured a type-I secondary instability on the same geometry in the 80–130 kHz band (their hotwire had a bandwidth of approximately 180 kHz). Figure 6.17 shows contour plots of the pressure fluctuation amplitudes within each of these bands.

The traveling crossflow amplitudes, in Figure 6.17(a), appear to be weakly modulated by the stationary vortices. Note that the PCBs often perform poorly at such a low frequency (see for instance the discussion in Ward [38]). To further complicate matters, as the flow transitions the amplitude at low frequencies rises due to factors other than the traveling crossflow instability. Thus while the apparent organization of the traveling crossflow is interesting, it is not providing a clear picture.

Measurements of the secondary instability, however, are much more reliable. Figure 6.17(b) shows two lobes of growth in this band, reminiscent of the low and high-frequency instabilities measured in Chapter 5. As with those measurements, the branches of traveling crossflow and secondary instability grow along trajectories roughly aligned with the stationary vortices. Figure 6.18 shows contours of the pressure fluctuation amplitudes for each instability superimposed on an IR image. Given the issues with repeatability after a model change as evidenced in Figure 6.2, the IR data in Figure 6.18 was taken concurrently with the PCB data (the heat-flux data in Section 6.2.1 is from after the model change). The IR images from all 15 runs with the PCBs were

averaged to produce the heat-transfer image in the figure. Both the traveling crossflow and the secondary instability grow generally along the same paths as the streaks, though the lobes of traveling wave growth seem to be inclined slightly with respect to the stationary waves. The secondary instability seems to be confined within the cold streaks (i.e. the upwelling and shoulder of the stationary vortex), and there is no high-frequency mode under the hot streak.

To understand the growth of the two instabilities in more detail, the power spectra can be examined along the lobes of growth of each mode. Figure 6.19 shows the individual sensor locations from which the power spectra were extracted. Every azimuthal angle is a different run, so these measurements are the composite of six separate runs.

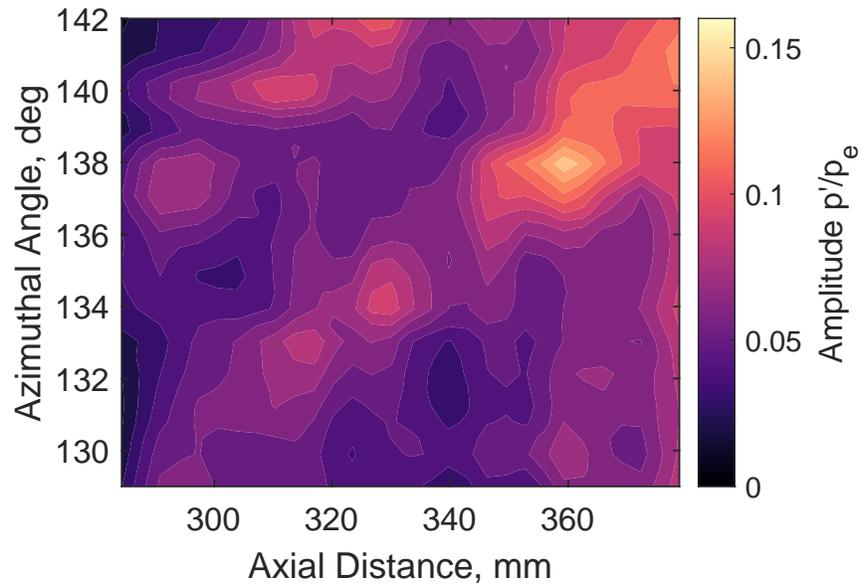
The spectra from the plotted measurement points are provided in Figure 6.20. The traveling crossflow increases in power slightly along the streamline, as shown in Figure 6.20(a). However, the growth at higher frequencies is much more rapid. The barest hint of a peak can be observed at around 275 kHz for the lowest Reynolds numbers, but at larger Re the growth becomes quite broadband. The origin of this high-frequency activity is uncertain, though the 275 kHz peak at $Re = 3.58 \times 10^6$ coincides with the estimated second-mode frequency.

The growth along the secondary instability path is more pronounced. Figure 6.20(b) shows a distinct peak forming at around 150 kHz and growing downstream. In addition, the traveling crossflow also grows, and there is broadband growth similar to Figure 6.20(a). It is particularly interesting that the secondary instability peak seems to *increase* in frequency by at least 20 kHz as the mode grows downstream. This is further illustrated in Fig. 6.21 by plotting the spectra against the Strouhal number f/f_{CFD} , where f_{CFD} is the CFD estimate of the second-mode frequency (Eq. 3.6); the estimate takes into account the growth of the boundary layer around the cone. With the RIM insert, the peak frequency of the secondary instabilities is inversely proportional to the boundary-layer thickness as the instabilities grow downstream (Fig. 6.21(a)), consistent with the behavior of the second mode. When using the Sandblasted insert, however, the peak frequency increases from about $0.5f_{CFD}$ to almost $0.8f_{CFD}$. This is atypical for a second-mode type instability, but may be consistent with computations of the type-I secondary instability [42], which exhibit a higher type-I frequency at higher azimuthal angles (where the boundary layer is thicker).

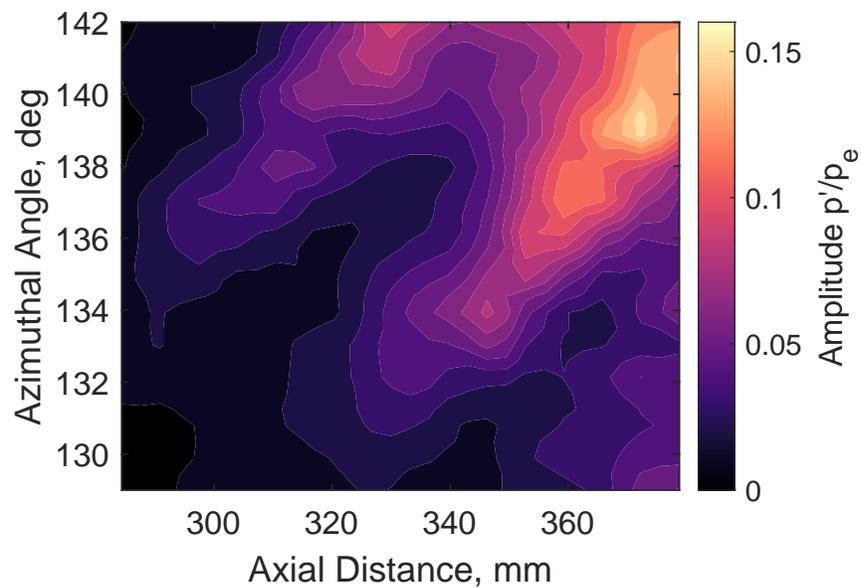
The pressure fluctuation amplitudes were also calculated at discrete measurement locations following the streaks. Figure 6.22 shows these amplitudes compared to the peak streak heating.

As numbered in Figure 6.11(a), Vortex 10 follows the hot streak directly leeward of the traveling wave measurement path. Vortex 9 follows the streak immediately windward of the secondary instability path. Note that the PCB data is from Entry 16 while vortex amplitude was taken in Entry 17, after the model change, so the comparison should be considered qualitative.

In general, the growth of both the primary and secondary instabilities appears to be monotonic, in contrast to the peaky behavior in Chapter 5. The spectra appear to be nearly turbulent at the highest Re and the Shannon entropy H at the last measurement station is around 95 %. It is interesting to note that the secondary instability grows quite rapidly, increasing in amplitude by a factor of 10 over the measurement region. In contrast, the traveling crossflow vortices grow only by a factor of 2 in the same region. In both cases, the stationary crossflow grows by about the same amount, a factor of 4. This “explosive” growth of the secondary instability suggests that an e^N -type correlation based on the secondary-instability amplitude is feasible, following the low-speed approach of Malik et al. [22].

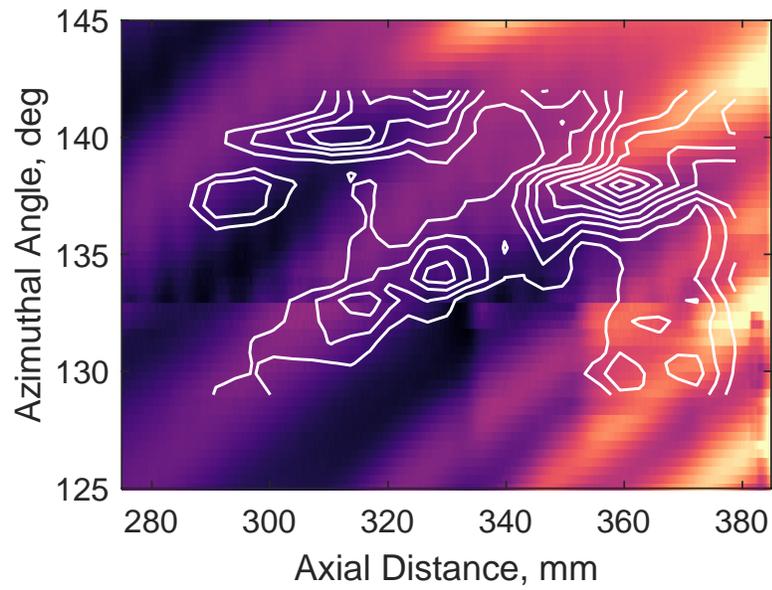


(a) *Traveling crossflow band, 25–75 kHz.*

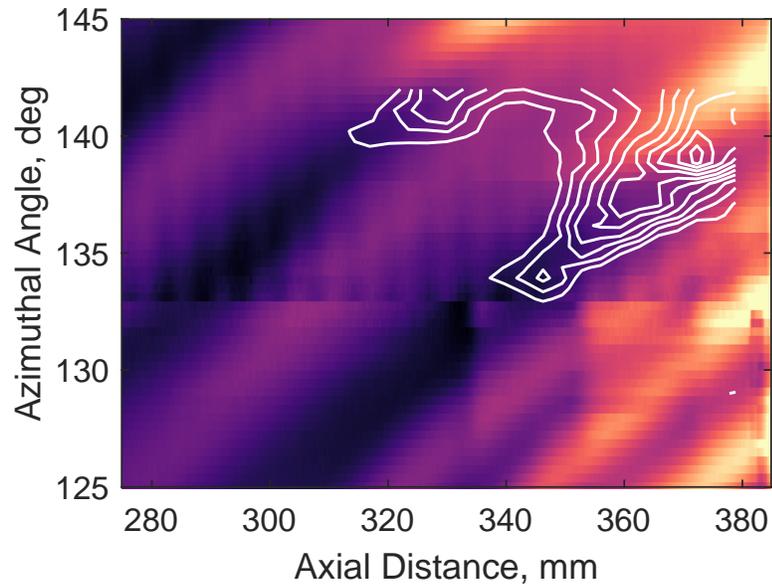


(b) *Secondary instability band, 80–250 kHz.*

Figure 6.17. Pressure fluctuation amplitudes of the two high-frequency instabilities. $Re_\infty = 11.1 \times 10^6 \text{ m}^{-1}$.



(a) *Traveling crossflow band.*



(b) *Secondary instability band.*

Figure 6.18. Contours of the instability pressure fluctuation amplitudes superimposed on a heat-transfer image. The image is the average of all 15 runs with the Sandblasted insert in Entry 16.

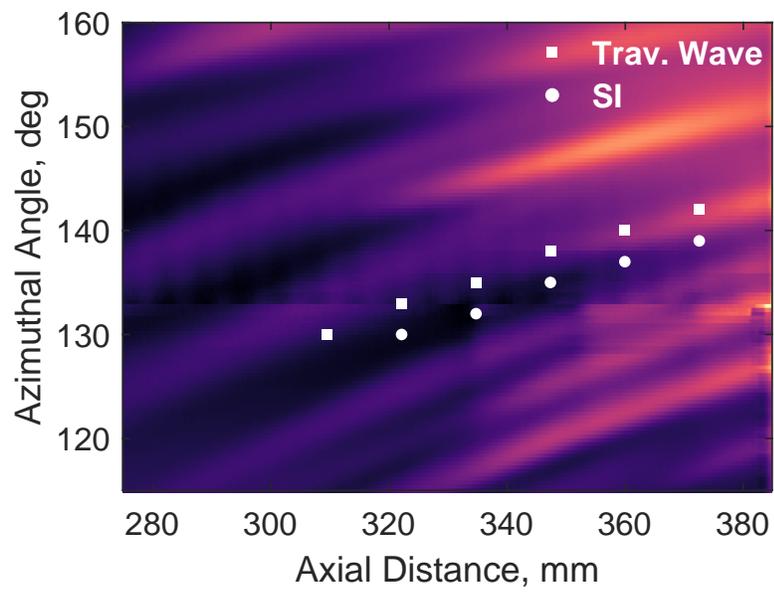
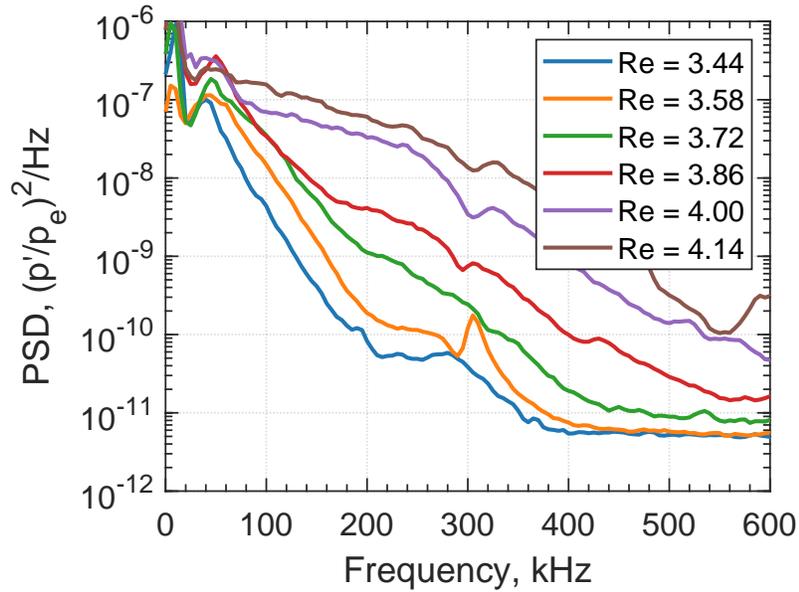
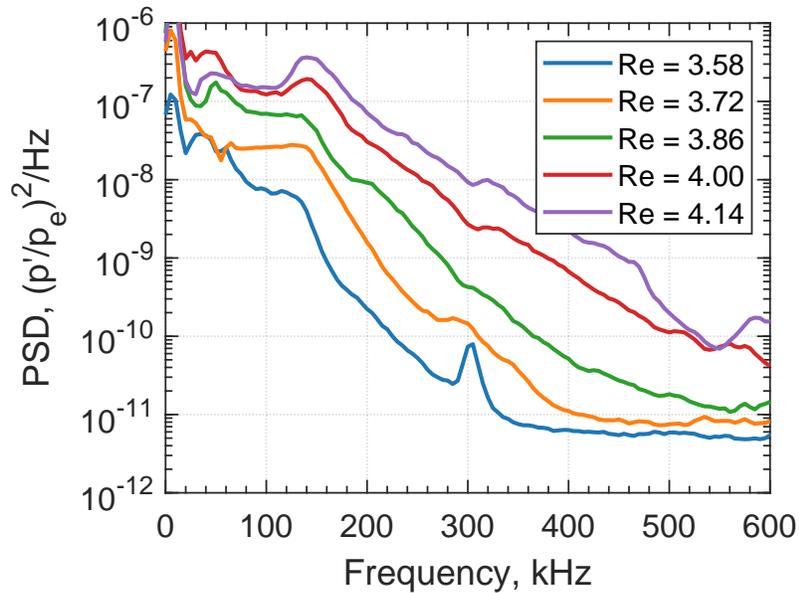


Figure 6.19. Discrete sensor locations used to measure the growth of the traveling crossflow and secondary instabilities. The plotted points correspond to the spectra in Figure 6.20.

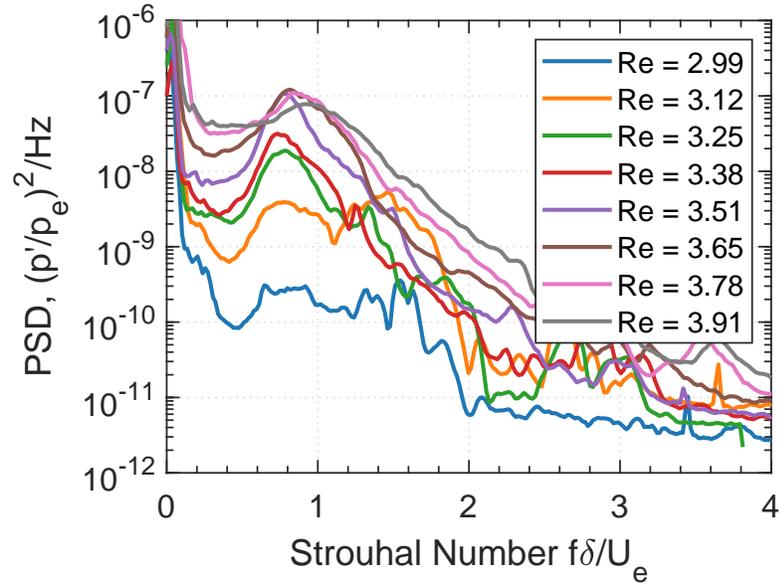


(a) Spectra along the lobe of growth of the traveling wave.

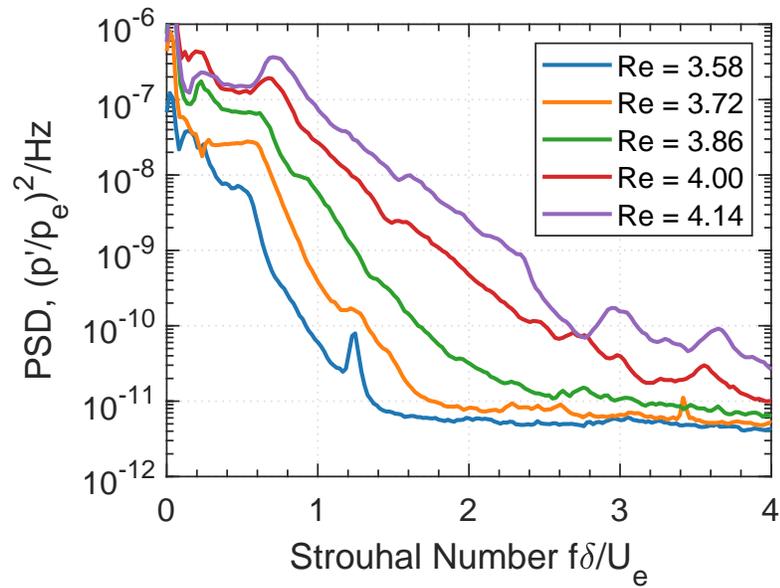


(b) Spectra along the lobe of growth of the secondary instability.

Figure 6.20. Spectra along the paths indicated in Figure 6.19. The legend indicates axial Reynolds numbers in millions. Note the increasing peak frequency as the secondary instability grows downstream.

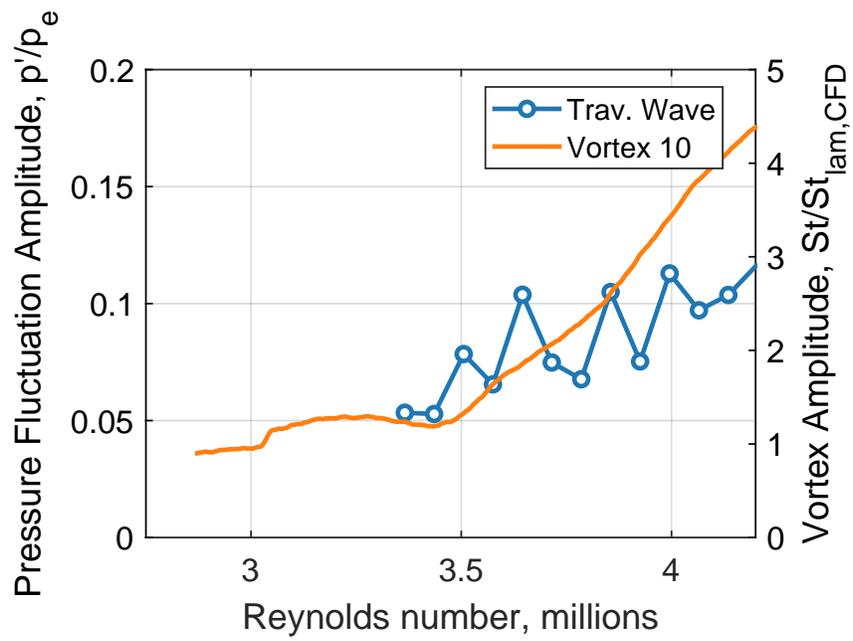


(a) RIM insert, low-frequency secondary instability.

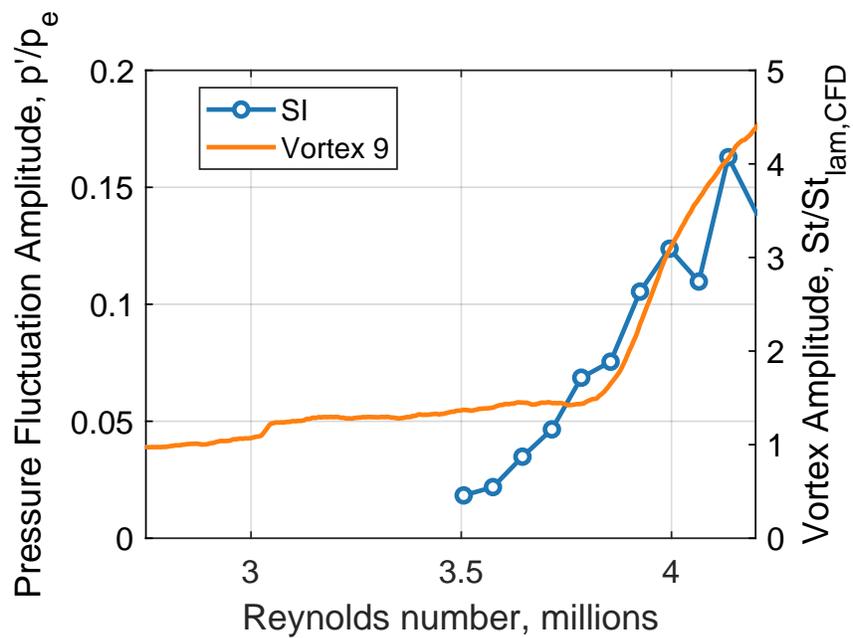


(b) Sandblasted insert, secondary instability.

Figure 6.21. Growth of the secondary instabilities against the Strouhal number f/f_{CFD} , illustrating the difference in behavior between the two roughness cases.



(a) Traveling crossflow amplitudes.



(b) Secondary instability amplitudes.

Figure 6.22. Pressure fluctuation amplitudes compared to neighboring stationary streaks.

6.3.1 Nonlinear Interactions with Traveling Crossflow

The bicoherence reveals several nonlinear interactions between the measured instabilities as the flow breaks down. Figure 6.23 illustrates nonlinear harmonic generation within the traveling crossflow vortices. The magnitude of the bicoherence is only 0.26, which is not large, but it is statistically significant and so implies that the traveling crossflow has grown to large enough amplitudes to become nonlinear. The first harmonic of the traveling vortices would be at 100 kHz, but any activity at this frequency is obscured by the broadband roll-off of the traveling vortices at 50 kHz.

The traveling vortices also interact with the high-frequency instabilities. Figure 6.24 shows broadband harmonic generation within the traveling crossflow band, as well as low levels of interaction between the traveling crossflow vortices and a high frequency instability at 300 kHz. It is unclear whether this peak is the second mode or a type-I secondary instability. The filled circle at 200 kHz in the PSD is the CFD-estimate of the second-mode frequency.

The secondary instabilities also reach nonlinear stages of growth. Figure 6.25 shows low levels of harmonic generation at the secondary instability around 150 kHz. As with the traveling crossflow, no harmonic at 300 kHz is obvious. But computations have shown that the type-I secondary instability has a very broad spectrum (nearly 600 kHz) [42], so it is not obvious that a distinct harmonic would ever be visible. The peak at 300 kHz in Figure 6.26 may be a harmonic of the secondary instability at 150 kHz, but it is difficult to say for certain as there is a known PCB sub-resonance near this frequency. The bicoherence for that case shows mostly harmonic generation at the secondary instability frequency, but also low levels of interaction between the traveling crossflow at 50 kHz and the two other instabilities.

As previously discussed, the PCBs do not always perform well at the low frequencies where the traveling crossflow grows. The Kulite measurements can help to corroborate the previous analysis. Figure 6.27 shows power spectra from Kulites 1 and 2 compared to similarly situated PCBs. Recall that the Kulite and PCB data were taken with different models, so the repeatability of the stationary vortices is tenuous. However, the spectra show very similar patterns. There are two distinct peaks: traveling crossflow at 50 kHz and the secondary instability at 125 kHz.

The bicoherence of the Kulite data corroborates that of the PCB data. There are four distinct regions of interaction, signified by moderate levels of bicoherence. There is harmonic generation within both instabilities, interaction between the two, and what appears to be interaction between the secondary instability and a higher-frequency mode at 200 kHz.

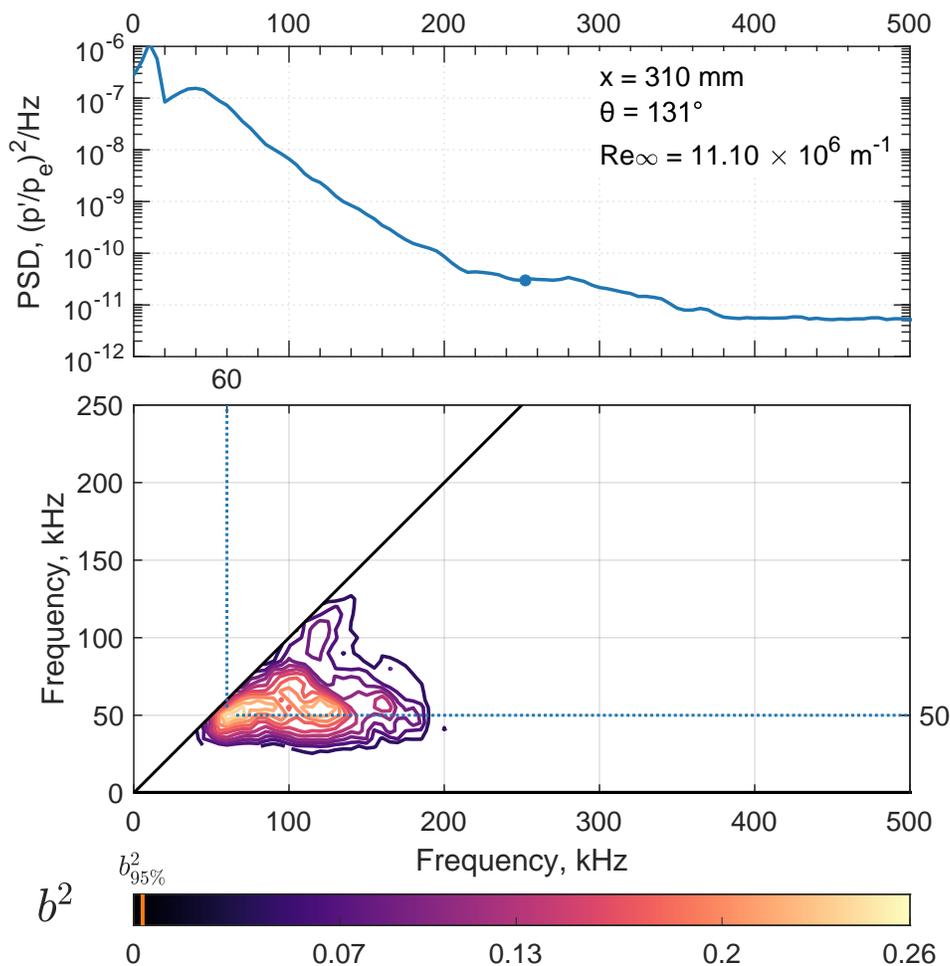


Figure 6.23. Bicoherence showing traveling crossflow wave harmonic generation. Run 1675.

6.4 Summary of Measurements with Distributed Roughness

The success and repeatability of the Sandblasted insert demonstrates the effectiveness of sandblasting in creating normally-distributed roughness at an RMS level sufficient to cause transition at the downstream end of the cone and to not be overpowered by other excrescences on the model. The normally-distributed roughness enables the growth of the stationary vortices at

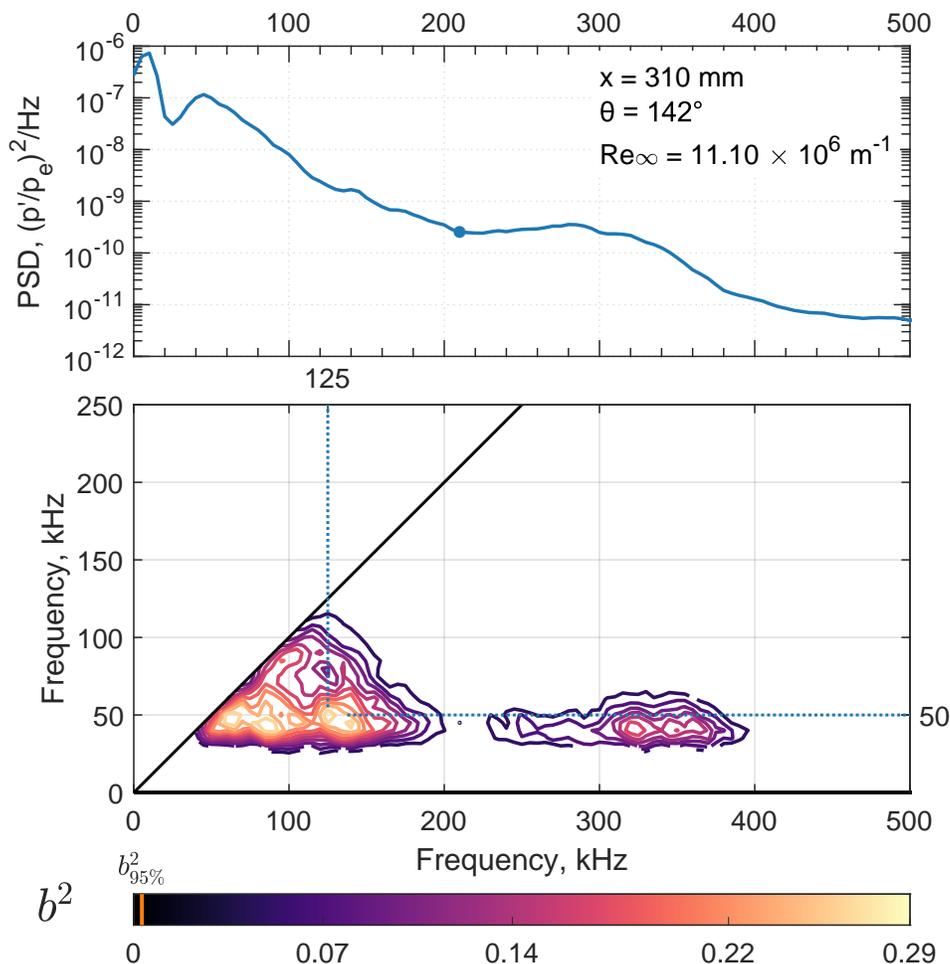


Figure 6.24. Bicoherence showing traveling crossflow harmonic generation and interaction with a high-frequency instability. Run 1680.

the naturally most-amplified wavenumber. This is in contrast to the RIM insert which forces a specific, “non-optimal” wavenumber.

The data in this chapter show that the difference in forcing leads to different nonlinear breakdown mechanisms. When using the Sandblasted insert, the stationary vortices grow in a more uniform way and traveling crossflow is distinct in the pressure fluctuation spectra. In further contrast to the RIM data—and to conventional wisdom—even in quiet flow the traveling crossflow vortices reach large enough amplitudes (about 10% of the edge pressure) to begin growing nonlinearly and to interact with the secondary instabilities of the stationary vortices.

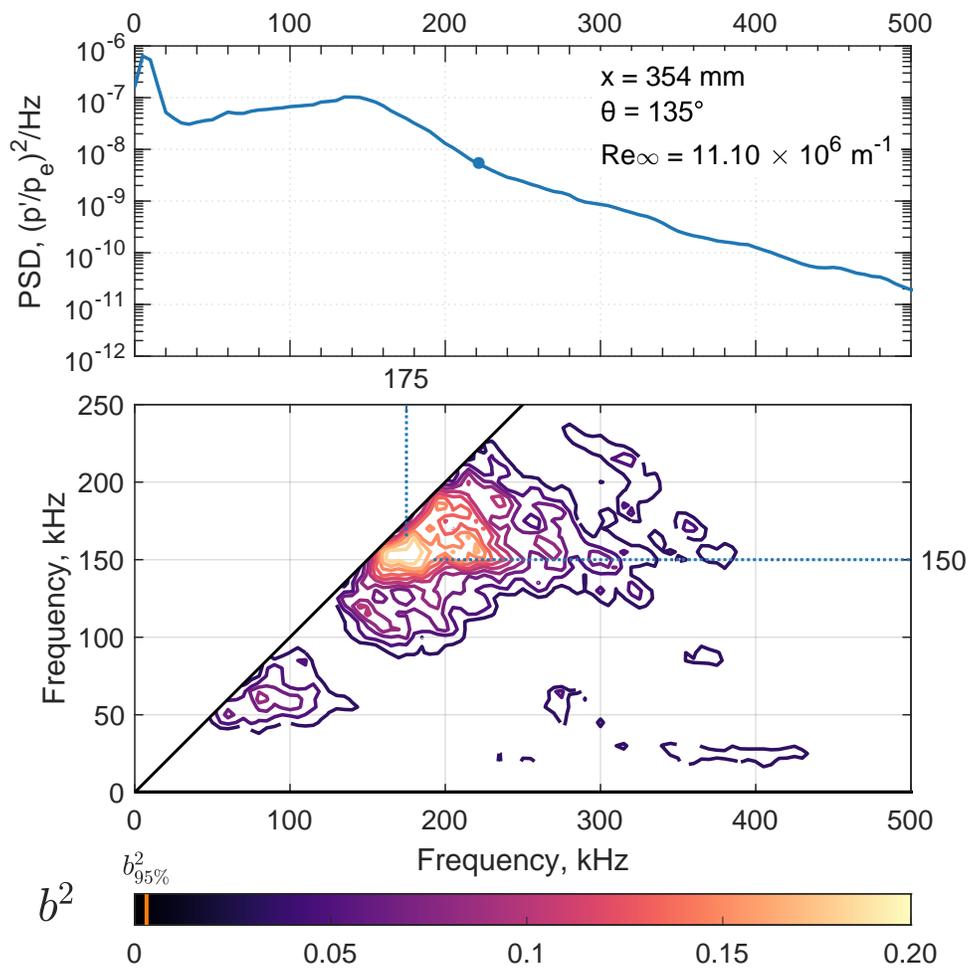


Figure 6.25. Bicoherence showing harmonic generation within the secondary instability band. Run 1673.

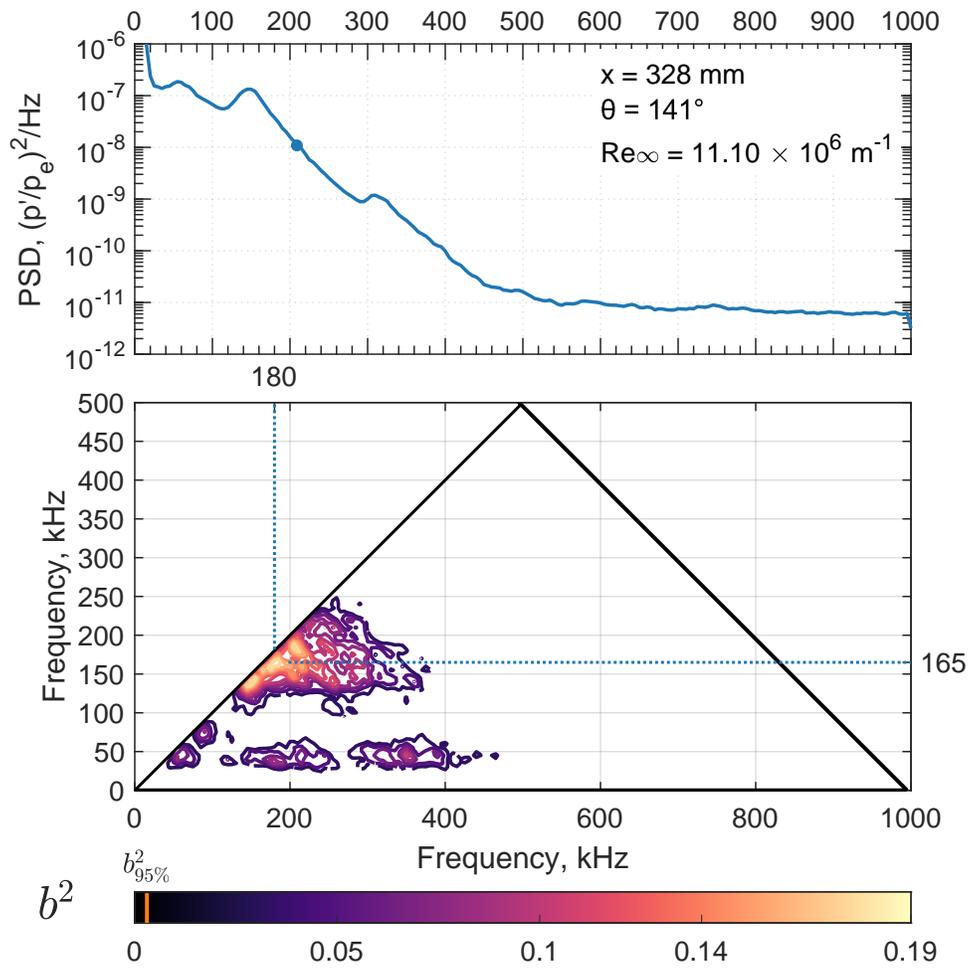


Figure 6.26. Bicoherence showing low levels of nonlinear interaction between traveling crossflow, the secondary instability, and a third peak of unknown origin. Run 1679.

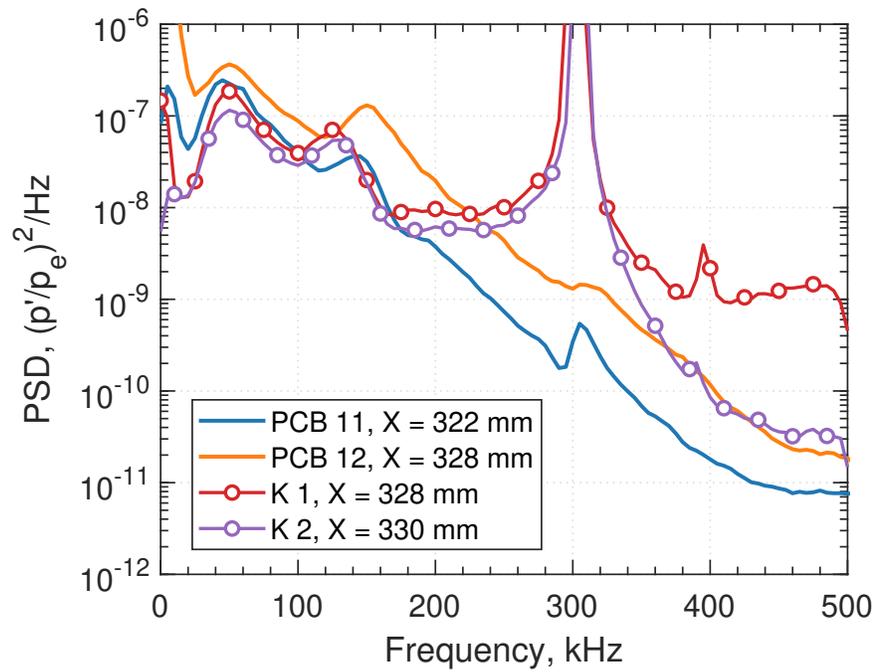


Figure 6.27. Comparison of power spectra from the Kulites and similarly situated PCB sensors. The Kulite data is from Entry 17, after the model change, so the repeatability of the stationary vortices is uncertain. $\theta = 142^\circ$, $Re_\infty = 11 \times 10^6 \text{ m}^{-1}$. PCB data, Run 1680; Kulite data, Run 1713.

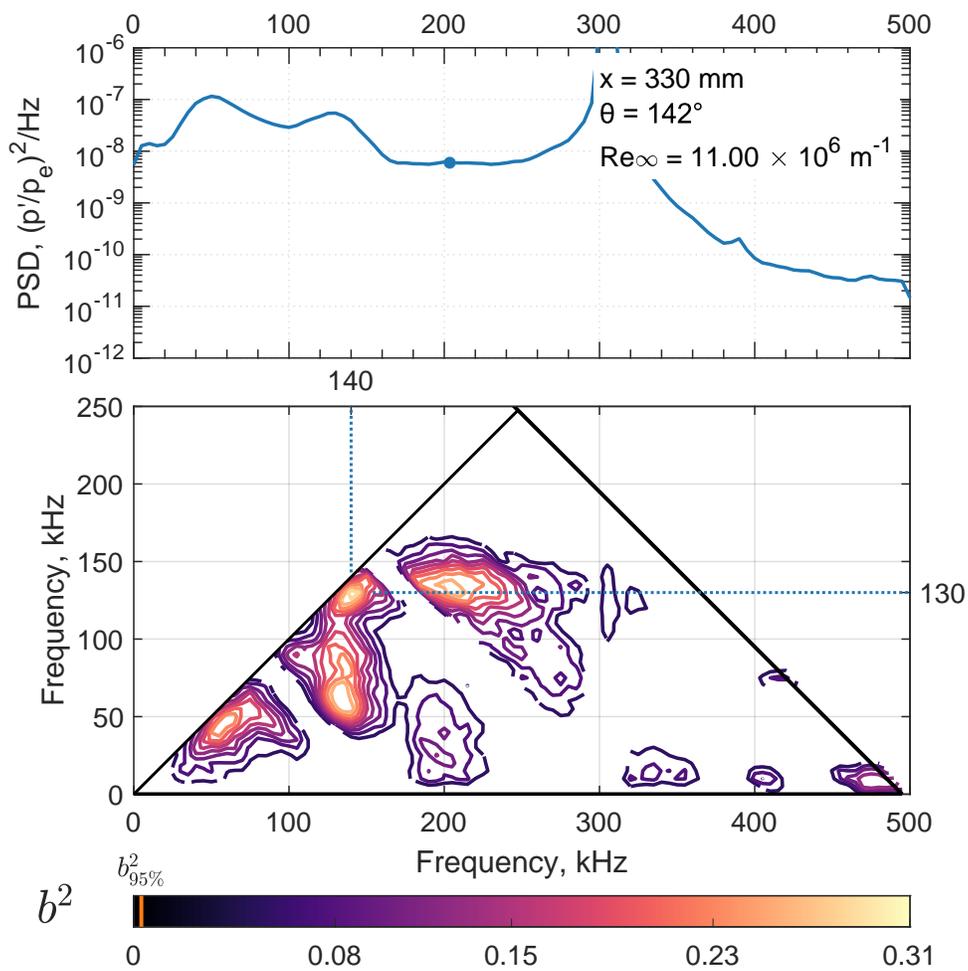


Figure 6.28. Bicoherence from a Kulite at a high azimuthal angle. Several regions of nonlinear interaction are evident, including harmonic generation within the traveling crossflow and secondary instability, and interaction between these two instabilities. Run 1713.

7. SUMMARY AND CONCLUSIONS

This thesis describes an experimental campaign on a 7° half-angle cone at 6° angle of attack in pursuit of a mechanism-based prediction method for crossflow-dominated transition. At low speeds, a semi-empirical e^N method based on the amplitude of crossflow secondary instabilities was more successful in predicting transition than a typical method based on the primary instability amplitudes, at least for one well-studied case. The goal of this work is to determine if such a method is feasible at hypersonic speeds as well.

The semi-empirical nature of the e^N method requires experiments with well-controlled and well-understood forcing of the stationary crossflow vortices. The Rod Insertion Method (RIM) roughness was adapted for this purpose from its original use on the Purdue flared cone. The discrete RIM roughness elements must be tall enough to dominate the generation of the stationary crossflow waves, but not so tall as to directly trip the flow. By making small rotations of the roughness elements, it was demonstrated that the elements are primarily responsible for the downstream pattern of heating associated with the stationary vortices. Thus the RIM roughness meets the first criterion, i.e. that the elements must be tall enough. A second experiment was performed using a range of different roughness heights to determine at which height the roughness directly trips the boundary layer. This critical height was not conclusively determined, but the selected roughness height was almost 3.5 times smaller than the first height at which tripping was observed.

Highly-detailed surface measurements were collected of the nonlinear breakdown when using this controlled roughness. These data provide information for validation of computations. Furthermore, use of the RIM insert revealed new insights into the nature of crossflow transition when using discrete roughness. Two modes of large surface pressure fluctuations were observed to grow nonlinearly before transition. The frequency of these modes appeared highly correlated with the modulation in the surface heat flux due to the large stationary crossflow vortices. In addition, there was no evidence of the presence of traveling crossflow waves in the surface pressure

fluctuations when using the RIM insert. Another interesting feature of the flow is the appearance of streak merging, qualitatively similar to low-speed computations of roughness-induced vortices in a 3D boundary layer. This streak merging seems to be coupled with the growth of the secondary instability, and so opens the door for further investigation.

The data collected when using the Sandblasted insert illustrates a fundamentally different nonlinear breakdown mechanism. The pressure fluctuation spectra show very clear traveling crossflow activity at low frequencies, and they reach large amplitudes on the order of 10 % of the edge pressure. The traveling crossflow appears to be weakly nonlinear near breakdown. In addition, the spectra look quite different from those in Chapter 5: there is usually no distinct high-frequency peak, and the peak frequency of the instabilities does not seem to be coupled with the local boundary-layer thickness. It is believed that the two instabilities present in the spectra of Chapter 6 are traveling crossflow and the type-I secondary instability of the stationary crossflow vortices. The low-frequency peak at around 45 kHz matches previous measurements and computations of traveling crossflow's frequency and phase velocity.

The fundamental goal of this research effort was to develop a mechanism-based transition prediction method using the secondary instability amplitudes, as was shown to work well by Malik, et al. [22] for one case at subsonic speeds. The first step towards this goal is to develop a controlled roughness which can repeatably induce strong stationary crossflow vortices without tripping the flow. The RIM roughness was shown to serve this purpose well. Unfortunately, the path to a prediction method is not as straight-forward as was outlined by Malik, et al. The nonlinear breakdown mechanisms at hypersonic speeds are highly sensitive to the forcing roughness. Thus even if a useful prediction method were created from the data with the controlled RIM insert, a different method may be required for cases more like the Sandblasted insert with low levels of distributed roughness. Furthermore, the nonlinear breakdown mechanisms may also be sensitive to the geometry; the second mode, for example, does not grow significantly in the crossflow-dominated region of an elliptic cone.

Thus the main conclusion from the present work is that the goal needs to change. Before attempting to create a transition prediction method, the underlying physics of the nonlinear breakdown must be better understood. What causes the change in breakdown mechanisms? How important are these differences? How do flight-like environments affect which nonlinear

breakdown mechanism is most important? Ultimately the purpose of this research is to provide a useful engineering tool to vehicle designers. But for such a tool to be useful, it is important to establish when and how it can be applied.

7.1 Open Questions and Future Work

The experiments described in this work are far from complete. There are several avenues of exploration that are left for future researchers. The following is a list of important open questions that merit further study.

1. *Why does the transition mechanism change between the two roughness cases? Is it the effect of roughness type (discrete or distributed) or roughness amplitude?*

Ideally one would also test a RIM insert that forces the naturally most-amplified wavenumber (about 220) and vary the height to determine if this is height or wavenumber dependent (or potentially both). At present this seems infeasible, though it would be good to develop new types of roughness for which such a high wavenumber might be possible. A computational study would help to shed light on the matter.

2. *What is happening away from the surface during breakdown?*

Measurements with the FLDI or a micro-pitot rig could help to understand the nature of the different nonlinear mechanisms presented in this work, though their spatial resolution would be limited.

3. *How do the steps at the model component interfaces affect the growth and breakdown of the measured instabilities?*

The effect of a small step is poorly understood. It would be worthwhile to produce several roughness inserts with slightly different aft diameters, so as to parametrically study the effect of changing the step on the nonlinear breakdown.

4. *What is really driving the process of streak merging when using the RIM insert, and can it be used to define a transition-prediction criterion?*

The preliminary results in this work suggest that there is a connection between the streak

merging and the nonlinear breakdown via second-mode waves. Different shapes of roughness (cylinder, half-cylinder, diamond, square, etc.) should be investigated to better understand the nature of the roughness-induced stationary vortices.

5. *Why do the traveling crossflow waves appear only rarely in the spectra when using the RIM insert? Are they present but obscured, or are they suppressed completely by the large stationary vortices?*

The role of traveling crossflow in the nonlinear breakdown when using the RIM insert is not clear. Off-surface measurements could help to better understand their behavior, especially with respect to interactions with secondary instabilities which are localized away from the wall.

6. *How does the freestream noise level affect transition when using RIM roughness?*

Some results from the Sandia Hypersonic Wind Tunnel [10] suggest that it is possible to induce large stationary crossflow vortices in a noisy environment by using the RIM roughness. If the resulting nonlinear breakdown mechanisms are similar, it could open up the study of crossflow-dominated transition in conventional wind tunnels.

7. *Is the traveling crossflow instability significant under noisy flow when using a distributed roughness?*

It is unclear whether traveling crossflow is present in noisy flow (see References [5, 10, 53]). To use data from conventional wind tunnels in the search for a mechanism-based prediction method for crossflow, it is imperative that the behavior of traveling crossflow in these tunnels is better understood.

REFERENCES

REFERENCES

- [1] Schneider, S. P., “Developing Mechanism-based Methods for Estimating Hypersonic Boundary-layer Transition in Flight: The Role of Quiet Tunnels,” *Progress in Aerospace Sciences*, Vol. 72, Jan. 2015, pp. 17–29.
- [2] Schneider, S., “Effects of High-Speed Tunnel Noise on Laminar-Turbulent Transition,” *Journal of Spacecraft and Rockets*, Vol. 38, No. 3, May 2001.
- [3] Craig, S. A. and Saric, W. S., “Experimental Study of Crossflow Instability on a Mach 6 Yawed Cone,” *AIAA Paper 2015-2774*, June 2015.
- [4] Poggie, J., Kimmel, R. L., and Schwoerke, S. N., “Traveling Instability Waves in a Mach 8 Flow over an Elliptic Cone,” *AIAA Journal*, Vol. 38, No. 2, Feb. 2000, pp. 251–258.
- [5] Borg, M. P., Kimmel, R. L., and Stanfield, S., “Traveling Crossflow Instability for the HIFiRE-5 Elliptic Cone,” *Journal of Spacecraft and Rockets*, Vol. 52, No. 3, May 2015, pp. 664–673.
- [6] Bippes, H., “Basic Experiments on Transition in Three-Dimensional Boundary Layers Dominated by Crossflow Instability,” *Progress in Aerospace Sciences*, Vol. 35, No. 4, May 1999, pp. 363–412.
- [7] Saric, W. S., Reed, H. L., and White, E. B., “Stability and Transition of Three-Dimensional Boundary Layers,” *Annual Review of Fluid Mechanics*, Vol. 35, Jan. 2003, pp. 413–440.
- [8] Deyhle, H. and Bippes, H., “Disturbance Growth in an Unstable Three-Dimensional Boundary Layer and Its Dependence on Environmental Conditions,” *Journal of Fluid Mechanics*, Vol. 316, June 1996, pp. 73–113.
- [9] Borg, M. P., Kimmel, R. L., Hofferth, J. W., Bowersox, R. D., and Mai, C. L., “Freestream Effects on Boundary Layer Disturbances for HIFiRE-5,” *AIAA Paper 2015-0278*, Jan. 2015.
- [10] Edelman, J., Casper, K., Henfling, J., Spillers, R., and Schneider, S., “Crossflow Transition on a Pitched Cone at Mach 8,” *AIAA Paper 2017-4299*, June 2017.
- [11] Radeztsky, Jr., R., Reibert, M., Saric, W., and Takagi, S., “Effect of Micron-Sized Roughness on Transition in Swept-Wing Flows,” *AIAA Paper 1993-0076*, Jan. 1993.
- [12] Reibert, M., Saric, W., Carrillo Jr., R., and Chapman, K., “Experiments on Nonlinear Saturation of Stationary Crossflow Vortices on a Swept-Wing Boundary Layer,” *AIAA Paper 1996-0184*, Jan. 1996.
- [13] Carrillo, Jr., R. B., Reibert, M. S., and Saric, W. S., “Distributed-Roughness Effects on Stability and Transition in Swept-Wing Boundary Layers,” Tech. rep., Final Report for NASA LaRC Cooperative Agreement NCC-1-194, NASA-CR-203580, 1996.
- [14] Malik, M. R., Li, F., and I. Chang, C., “Crossflow Disturbances in Three-Dimensional Boundary Layers: Nonlinear Development, Wave Interaction and Secondary Instability,” *Journal of Fluid Mechanics*, Vol. 268, June 1994, pp. 1–36.

- [15] Kohama, Y., Saric, W., and Hoos, J., "A High-Frequency, Secondary Instability of Crossflow Vortices that Leads to Transition," *Proceedings of the Royal Aeronautical Society Conference on Boundary Layer Transition and Control*, Feb. 1991, pp. 4.1–4.13, Cambridge, U.K.
- [16] White, E. and Saric, W., "Secondary Instability of Crossflow Vortices," *Journal of Fluid Mechanics*, Vol. 525, Feb. 2005, pp. 275–308.
- [17] Malik, M., Li, F., and I. Chang, C., "Nonlinear Crossflow Disturbances and Secondary Instabilities in Swept-Wing Boundary Layers," *IUTAM Symposium on Nonlinear Instability and Transition in Three-Dimensional Boundary Layers*, edited by P. W. Duck and P. Hall, Kluwer Academic Publishers, 1996, pp. 257–266, Manchester, U.K.
- [18] Craig, S. A. and Saric, W. S., "Crossflow Instability in a Hypersonic Boundary Layer," *Journal of Fluid Mechanics*, Vol. 808, Oct. 2016, pp. 224–244.
- [19] Moyes, A., Paredes, P., Kocian, T., and Reed, H., "Secondary Instability Analysis of Crossflow on a Hypersonic Yawed Straight Circular Cone," *Journal of Fluid Mechanics*, Vol. 812, Feb. 2017, pp. 370–397.
- [20] Choudhari, M. M., Li, F., Paredes, P., and Duan, L., "Computations of Crossflow Instability in Hypersonic Boundary Layers," *AIAA Paper 2017-4300*, June 2017.
- [21] Edelman, J. and Schneider, S., "Secondary Instabilities of Hypersonic Stationary Crossflow Waves," *AIAA Journal*, Vol. 56, No. 1, Jan. 2018, pp. 182–192.
- [22] Malik, M. R., Li, F., Choudhari, M. M., and Iyan Chang, C., "Secondary Instability of Crossflow Vortices and Swept-Wing Boundary-Layer Transition," *Journal of Fluid Mechanics*, Vol. 399, Nov. 1999, pp. 85–115.
- [23] Michel, R., Arnal, D., Coustols, E., and Juillen, J. C., "Experimental and Theoretical Studies of Boundary Layer Transition on a Swept Infinite Wing," *Laminar-Turbulent Transition*, Springer Berlin Heidelberg, 1985, pp. 553–561.
- [24] Poll, D. I. A., "Some Observations of the Transition Process on the Windward Face of a Long Yawed Cylinder," *Journal of Fluid Mechanics*, Vol. 150, Jan. 1985, pp. 329–356.
- [25] Gregory, N., Stuart, J., and Walker, W., "On the Stability of Three-Dimensional Boundary Layers with Application to the Flow Due to a Rotating Disk," *Philosophical Transactions of the Royal Society of London. Series A, Mathematical and Physical Sciences*, Vol. 248, No. 943, July 1955, pp. 155–199.
- [26] T. Lerche, "Experimental Investigation of Nonlinear Wave Interactions and Secondary Instability in Three-Dimensional Boundary-Layer Flow," *Proceedings of the 6th European Turbulence Conference, Advances in Turbulence VI*, July 1996, pp. 357–360, Lausanne, Switzerland.
- [27] Li, F. and Malik, M., "Fundamental and Subharmonic Secondary Instabilities of Görtler Vortices," *Journal of Fluid Mechanics*, Vol. 297, Aug. 1995, pp. 77–100.
- [28] Swearingen, J. and Blackwelder, R., "The Growth and Breakdown of Streamwise Vortices in the Presence of a Wall," *Journal of Fluid Mechanics*, Vol. 182, Sept. 1987, pp. 255–290.
- [29] Koch, W., Bertolotti, F., Stolte, A., and Hein, S., "Nonlinear Equilibrium Solutions in a Three-Dimensional Boundary Layer and Their Secondary Instability," *Journal of Fluid Mechanics*, Vol. 406, March 2000, pp. 131–174.

- [30] Bonfigli, G. and Kloker, M., "Secondary Instability of Crossflow Vortices: Validation of the Stability Theory by Direct Numerical Simulation," *Journal of Fluid Mechanics*, Vol. 583, July 2007, pp. 229–272.
- [31] Li, F. and Choudhari, M., "Spatially Developing Secondary Instabilities and Attachment Line Instability in Supersonic Boundary Layers," *AIAA Paper 2008-0590*, Jan. 2008.
- [32] Fujii, K., "Experiment of the Two-Dimensional Roughness Effect on Hypersonic Boundary-Layer Transition," *Journal of Spacecraft and Rockets*, Vol. 43, No. 4, July 2006.
- [33] Schneider, S. P. and Juliano, T. J., "Laminar-Turbulent Transition Measurements in the Boeing/AFOSR Mach-6 Quiet Tunnel," *AIAA Paper 2007-4489*, June 2007.
- [34] Hofferth, J. W., Bowersox, R. D. W., and Saric, W. S., "The Mach 6 Quiet Tunnel at Texas A&M: Quiet Flow Performance," *AIAA Paper 2010-4794*, July 2010.
- [35] Chynoweth, B. C., Ward, C. A. C., Greenwood, R. T., Mckiernan, G. R., Fisher, R. A., and Schneider, S. P., "Measuring Transition and Instabilities in a Mach 6 Hypersonic Quiet Wind Tunnel," *AIAA Paper 2014-2643*, June 2014.
- [36] Ward, C., Henderson, R., and Schneider, S., "Possible Secondary Instability of Stationary Crossflow Vortices on an Inclined Cone at Mach 6," *AIAA Paper 2015-2773*, June 2015.
- [37] Henderson, R., *Crossflow Transition at Mach 6 on a Cone at Low Angles of Attack*, Master's thesis, Purdue University, Dec. 2014.
- [38] Ward, C., *Crossflow Instability and Transition on a Circular Cone at Angle of Attack in a Mach-6 Quiet Tunnel*, Ph.D. thesis, Purdue University, Dec. 2014.
- [39] Schuele, C. Y., Corke, T. C., and Matlis, E., "Control of Stationary Cross-flow Modes in a Mach 3.5 Boundary Layer Using Patterned Passive and Active Roughness," *Journal of Fluid Mechanics*, Vol. 718, Feb. 2013, pp. 5–38.
- [40] Moyes, A., Paredes, P., Kocian, T., and Reed, H., "Secondary Instability Analysis of Crossflow on a Hypersonic Yawed Straight Circular Cone," *AIAA Paper 2016-0848*, Jan. 2016.
- [41] Balakumar, P. and Owens, L., "Stability of Hypersonic Boundary Layers on a Cone at Angle of Attack," *AIAA Paper 2010-4718*, June 2010.
- [42] Li, F., Choudhari, M. M., Paredes-Gonzalez, P., and Duan, L., "Secondary Instability of Stationary Crossflow Vortices in Mach 6 Boundary Layer over a Circular Cone," Tech. rep., NASA/TM-2015-218997, Dec. 2015.
- [43] Borg, M. P., Kimmel, R. L., and Stanfield, S., "Traveling Crossflow Instability for HIFiRE-5 in a Quiet Hypersonic Wind Tunnel," *AIAA Paper 2013-2737*, June 2013.
- [44] Borg, M. P. and Kimmel, R. L., "Simultaneous Infrared and Pressure Measurements of Crossflow Instability Modes for HIFiRE-5," *AIAA Paper 2016-0354*, Jan. 2016.
- [45] Juliano, T. J., Paquin, L. A., and Borg, M. P., "HIFiRE-5 Boundary-layer Transition Measured in a Mach-6 Quiet Tunnel with Infrared Thermography," *AIAA Journal*, Jan. 2019, pp. 1–10.
- [46] Borg, M. P. and Kimmel, R. L., "Measurements of Crossflow Instability Modes for HIFiRE-5 at Angle of Attack," *AIAA Paper 2017-1681*, Jan. 2017.
- [47] Moyes, A., Kocian, T. S., Mullen, D., and Reed, H. L., "Effects of Initial Disturbance Amplitude on Hypersonic Crossflow Instability," *AIAA Paper 2018-1820*, Jan. 2018.

- [48] Neel, I. T., Leidy, A., Tichenor, N. R., and Bowersox, R. D., "Influence of Environmental Disturbances on Hypersonic Crossflow Instability on the HIFiRE-5 Elliptic Cone," *AIAA Paper 2018-1821*, Jan. 2018.
- [49] Dinzl, D. J. and Candler, G. V., "Direct Simulation of Hypersonic Crossflow Instability on an Elliptic Cone," *AIAA Journal*, Vol. 55, No. 6, June 2017, pp. 1769–1782.
- [50] Yates, H., Juliano, T. J., Matlis, E. H., and Tufts, M. W., "Plasma-actuated Flow Control of Hypersonic Crossflow-induced Boundary-layer Transition in a Mach-6 Quiet Tunnel," *AIAA Paper 2018-1076*, Jan. 2018.
- [51] Yates, H., Juliano, T. J., Matlis, E. H., and Tufts, M. W., "Crossflow Transition Acceleration with Plasma Actuators in Hypersonic Quiet Flow," *AIAA Paper 2019-1909*, Jan. 2019.
- [52] Matlis, E. H., Arndt, A. J., Corke, T. C., and Semper, M., "Effect of Discrete Roughness on Cross-flow Transition at Mach 6," *AIAA Paper 2018-1077*, Jan. 2018.
- [53] Corke, T., Arndt, A., Matlis, E., and Semper, M., "Control of Stationary Cross-flow Modes in a Mach 6 Boundary Layer Using Patterned Roughness," *Journal of Fluid Mechanics*, Vol. 856, Oct. 2018, pp. 822–849.
- [54] McDevitt, J. B. and Mellenthin, J. A., "Upwash Patterns on Ablating And Nonablating Cones at Hypersonic Speeds," Tech. Rep. NASA TN D-5346, NASA, July 1969.
- [55] Adams, Jr., J. C., "Three-Dimensional Laminar Boundary-Layer Analysis of Upwash Patterns and Entrained Vortex Formation on Sharp Cones at Angle of Attack," Tech. Rep. AEDC-TR-71-215, Arnold Engineering Development Center, Dec. 1971.
- [56] Swanson, E. and Schneider, S., "Boundary-Layer Transition on Cones at Angle of Attack in a Mach-6 Quiet Tunnel," *AIAA 2010-1062*, Jan. 2010.
- [57] van den Kroonenberg, A., Radespiel, R., Candler, G., and Estorf, M., "Infrared Measurements of Boundary-Layer Transition on an Inclined Cone at Mach 6," *AIAA 2010-1063*, Jan. 2010.
- [58] Muñoz, F., Heitmann, D., and Radespiel, R., "Instability Modes in Boundary Layers of an Inclined Cone at Mach 6," *Journal of Spacecraft and Rockets*, Vol. 51, No. 2, March 2014, pp. 442–454.
- [59] Choudhari, M. M., Li, F., Paredes, P., and Duan, L., "Nonlinear Evolution and Breakdown of Azimuthally Compact Crossflow Vortex Pattern Over a Yawed Cone," *AIAA Paper 2018-1823*, Jan. 2018.
- [60] Edelman, J. B., Chynoweth, B. C., Mckiernan, G. R., Sweeney, C. J., and Schneider, S. P., "Instability Measurements in the Boeing/AFOSR Mach-6 Quiet Tunnel," *AIAA Paper 2016-3343*, June 2016.
- [61] Lakebrink, M. T. and Borg, M. P., "Traveling Crossflow Wave Predictions on the HIFiRE-5 at Mach 6: Stability Analysis Vs. Quiet Tunnel Data," *AIAA Paper 2016-0356*, Jan. 2016.
- [62] Juliano, T., Poggie, J., Porter, K., Kimmel, R., Jewell, J., and Adamczak, D., "HIFiRE-5b Heat Flux and Boundary-Layer Transition," *AIAA Paper 2017-3134*, June 2017.
- [63] Juliano, T. J., Poggie, J., Porter, K. M., Kimmel, R. L., Jewell, J. S., and Adamczak, D. W., "HIFiRE-5b Heat Flux and Boundary-Layer Transition," *Journal of Spacecraft and Rockets*, Vol. 55, No. 6, Nov. 2018, pp. 1315–1328.

- [64] Tufts, M. W., Gosse, R. C., and Kimmel, R. L., "Parabolized Stability Equation Analysis of Crossflow Instability on HIFiRE-5b Flight Test," *Journal of Spacecraft and Rockets*, Vol. 55, No. 6, Nov. 2018, pp. 1369–1378.
- [65] Chynoweth, B., *A New Roughness Array for Controlling the Nonlinear Breakdown of Second-Mode Waves at Mach 6*, Master's thesis, Purdue University, Aug. 2015.
- [66] Chynoweth, B., *Measurements of Transition Dominated by the Second Mode Instability at Mach 6*, Ph.D. thesis, Purdue University, May 2018.
- [67] Cerasuolo, S., *Heat Flux Measurements by Infrared Thermography in the Boeing/AFOSR Mach-6 Quiet Tunnel*, Master's thesis, Università degli Studi di Napoli Federico II, Jan. 2017.
- [68] Zaccara, M., *Infrared Thermography Data Reduction for Heat Transfer Measurements in the Boeing/AFOSR Mach-6 Quiet Tunnel*, Master's thesis, Università degli Studi di Napoli Federico II, Jan. 2018.
- [69] Walker, D. and Scott, E., "Evaluation of Estimation Methods of High Unsteady Heat Fluxes from Surface Measurements," *Journal of Thermophysics and Heat Transfer*, Vol. 12, No. 4, Oct. 1998, pp. 543–551.
- [70] Estorf, M., "Image Based Heating Rate Calculation From Thermographic Data Considering Lateral Heat Conduction," *International Journal of Heat and Mass Transfer*, Vol. 49, July 2006, pp. 2545–2556.
- [71] Fernandes, A. P., Santos, M. B. D., and Guimarães, G., "An Analytical Transfer Function Method to Solve Inverse Heat Conduction Problems," *Applied Mathematical Modelling*, Vol. 39, No. 22, Nov. 2015, pp. 6897–6914.
- [72] Schneider, S. P., "Development of Hypersonic Quiet Tunnels," *Journal of Spacecraft and Rockets*, Vol. 45, No. 4, July 2008.
- [73] Edelman, J., *Secondary Instabilities of Hypersonic Stationary Crossflow Waves*, Master's thesis, Purdue University, West Lafayette, IN, Dec. 2016.
- [74] Steen, L. E., *Characterization and Development of Nozzles for a Hypersonic Quiet Wind Tunnel*, Master's thesis, Purdue University, Dec. 2010.
- [75] Gray, K. A., *Flowfield Characterization of the Boeing/AFOSR Mach-6 Quiet Tunnel*, Master's thesis, Purdue University, Dec. 2018.
- [76] Muñoz, F., Heitmann, D., and Radespiel, R., "Instability Modes in Boundary Layers of an Inclined Cone at Mach 6," *AIAA Paper 2012-2823*, June 2012.
- [77] PCB Piezotronics, *Micro ICP Pressure Sensor Model 132B38 datasheet*, April 2019.
- [78] Beresh, S., Henfling, J., Spillers, R., and Pruett, B., "Measurement of Fluctuating Wall Pressures Beneath a Supersonic Turbulent Boundary Layer," *AIAA Paper 2010-0305*, Jan. 2010.
- [79] Berridge, D. C., *Generating Low-Pressure Shock Waves for Calibrating High-Frequency Pressure Sensors*, Ph.D. thesis, Purdue University, Dec. 2015.
- [80] Wason, M. M., *Calibration of High-Frequency Pressure Sensors Using Low-Pressure Shock Waves*, Master's thesis, Purdue University, May 2019.

- [81] Ort, D. J. and Dosch, J. J., "Influence of Mounting on the Accuracy of Piezoelectric Pressure Measurements for Hypersonic Boundary Layer Transition," *AIAA Paper 2019-2292*, Jan. 2019.
- [82] Kulite Semiconductor Products, Inc., *Kulite XCQ-062 Series Datasheet*, Feb. 2019.
- [83] G. Schivley, InfraTec GmbH, Private communication, Oct. 2018.
- [84] Sullivan, J., Schneider, S., Liu, T., Rubal, J., Ward, C., Dussling, J., Rice, C., Foley, R., Cai, Z., Wang, B., and Woodiga, S., "Quantitative Global Heat Transfer in a Mach-6 Quiet Tunnel," Tech. rep., NASA/CR-2012-217331, Feb. 2012.
- [85] M.P. Borg, AFRL, Private communication, Dec. 2017.
- [86] Turbeville, F. D. and Schneider, S. P., "Transition on a Cone with a Highly-Swept Fin at Mach 6," *AIAA Paper 2019-3217*, June 2019.
- [87] Kim, Y. C. and Powers, E. J., "Digital Bispectral Analysis and Its Applications to Nonlinear Wave Interactions," *IEEE Transactions on Plasma Science*, Vol. 7, No. 2, June 1979, pp. 120–131.
- [88] Hinich, M. and Wolinsky, M., "Normalizing Bispectra," *Journal of Statistical Planning and Inference*, Vol. 130, March 2005, pp. 405–411.
- [89] Haubrich, R. A., "Earth Noise, 5 to 500 Millicycles Per Second: 1. Spectral Stationarity, Normality, and Nonlinearity," *Journal of Geophysical Research*, Vol. 70, No. 6, March 1965, pp. 1415–1427.
- [90] Elgar, S. and Guza, R. T., "Observations of Bispectra of Shoaling Surface Gravity Waves," *Journal of Fluid Mechanics*, Vol. 161, No. -1, Dec. 1985, pp. 425.
- [91] Ching, C. Y. and Lagraff, J. E., "Measurement of Turbulent Spot Convection Rates in a Transitional Boundary Layer," *Experimental Thermal and Fluid Science*, Vol. 11, No. 1, July 1995, pp. 52–60.
- [92] Casper, K. M., Beresh, S. J., Henfling, J. F., Spillers, R. W., and Pruett, B. O. M., "Toward Transition Statistics Measured on a 7-Degree Hypersonic Cone for Turbulent Spot Modeling," *AIAA Paper 2014-0427*, Jan. 2014.
- [93] Shen, J. L., Hung, J. W., and Lee, L. S., "Robust Entropy-based Endpoint Detection for Speech Recognition in Noisy Environments," *ICSLP*, Dec. 1998.
- [94] Baraniuk, R., Flandrin, P., Janssen, A., and Michel, O., "Measuring Time-frequency Information Content Using the Renyi Entropies," *IEEE Transactions on Information Theory*, Vol. 47, No. 4, May 2001, pp. 1391–1409.
- [95] Obin, N. and Liuni, M., "On the Generalization of Shannon Entropy for Speech Recognition," *2012 IEEE Spoken Language Technology Workshop (SLT)*, IEEE, Dec. 2012.
- [96] Johnston, J., "Transform Coding of Audio Signals Using Perceptual Noise Criteria," *IEEE Journal on Selected Areas in Communications*, Vol. 6, No. 2, Feb. 1988, pp. 314–323.
- [97] Li, F., Choudhari, M., Chang, C. L., and White, J., "Analysis of Instabilities in Non-axisymmetric Hypersonic Boundary Layers Over Cones," *AIAA Paper 2010-4643*, June 2010.
- [98] Oliviero, N. B., Kocian, T. S., Moyes, A., and Reed, H. L., "EPIC: NPSE Analysis of Hypersonic Crossflow Instability on Yawed Straight Circular Cone," *AIAA Paper 2015-2772*, June 2015.

- [99] Kurz, H. B. E. and Kloker, M. J., "Mechanisms of Flow Tripping by Discrete Roughness Elements in a Swept-Wing Boundary Layer," *Journal of Fluid Mechanics*, Vol. 796, April 2016, pp. 158–194.
- [100] Schneider, S., "Effects of Roughness on Hypersonic Boundary-Layer Transition," *Journal of Spacecraft and Rockets*, Vol. 45, No. 2, March 2008.
- [101] Casper, K. M., Johnson, H. B., and Schneider, S. P., "Effect of Freestream Noise on Roughness-Induced Transition for a Slender Cone," *Journal of Spacecraft and Rockets*, Vol. 48, No. 3, May 2011, pp. 406–413.
- [102] Kimmel, R., "Roughness Considerations for the HIFiRE-1 Vehicle," *AIAA Paper 2008-4293*, June 2008.
- [103] Braslow, A. L., "A Review of Factors Affecting Boundary-Layer Transition," Tech. Rep. TN D-3384, NASA, Aug. 1966.
- [104] Gronvall, J. E., Bisek, N. J., and Poggie, J., "High Fidelity Computational Study of the HIFiRE-1 Boundary Layer Trip," *AIAA Paper 2014-0433*, Jan. 2014.
- [105] Wheaton, B. M. and Schneider, S. P., "Hypersonic Boundary-Layer Instabilities due to Near-Critical Roughness," *Journal of Spacecraft and Rockets*, Vol. 51, No. 1, Jan. 2014, pp. 327–342.
- [106] Whitehead, Jr., A. H., "Flow-Field and Drag Characteristics of Several Boundary-Layer Tripping Elements in Hypersonic Flow," Tech. Rep. TN-D-5454, NASA, Oct. 1969.
- [107] Moyes, A.J., Texas A&M University, Private communication, Sept. 2018.
- [108] Chynoweth, B., Edelman, J., Gray, K., Mckiernan, G., and Schneider, S., "Measurements in the Boeing/AFOSR Mach-6 Quiet Tunnel on Hypersonic Boundary-Layer Transition," *AIAA Paper 2017-3632*, June 2017.
- [109] Demetriades, A., "Laminar Boundary Layer Stability Measurements at Mach 7 Including Wall Temperature Effects," Tech. Rep. DTIC ADA049535, U.S. Air Force Office of Scientific Research, Nov. 1977.
- [110] Stetson, K. F., Thompson, E. R., Donaldson, J. C., and Siler, L. G., "Laminar Boundary Layer Stability Experiments on a Cone at Mach 8. Part 1: Sharp Cone," *AIAA Paper 1983-1761*, July 1983.
- [111] Serpieri, J. and Kotsonis, M., "Three-dimensional Organisation of Primary and Secondary Crossflow Instability," *Journal of Fluid Mechanics*, Vol. 799, June 2016, pp. 200–245.
- [112] Wassermann, P. and Kloker, M., "Mechanisms and Passive Control of Crossflow-vortex-induced Transition in a Three-dimensional Boundary Layer," *Journal of Fluid Mechanics*, Vol. 456, April 2002.
- [113] VanSant, J. H., "Conduction Heat Transfer Solutions," Technical Report UCRL-52863, Lawrence Livermore National Laboratory, March 1980.

APPENDICES

A. SELECTED RUN CONDITIONS

Table A.1. Entry 9.

Run	p_0 , psia	T_0 , °C	Re_∞ , $\times 10^6 \text{ m}^{-1}$	Roughness	Rough. Offset, deg
936	126.2	419.4	9.8	RIM7x-5-22-18	0

Table A.2. Entry 13.

Run	p_0 , psia	T_0 , °C	Re_∞ , $\times 10^6 \text{ m}^{-1}$	Roughness	Rough. Offset, deg
1309	122.5	414.5	9.7	RIM12x-16-22-9P	15
1310	140.3	417.1	11	RIM12x-16-22-9P	15
1311	132	418.1	10.3	RIM12x-16-22-9P	15
1321	139.8	416.1	11	RIM12x-8-22-9P	0
1326	140.4	415	11.1	RIM12x-2-22-9	15
1329	141.3	416.4	11.1	RIM12x-8-22-9P	15
1331	139	412.6	11.1	RIM12x-6-22-9P	15
1332	140.2	414.7	11.1	RIM12x-16-22-9P	15
1333	138.5	411.6	11.1	RIM12x-12-22-9P	15
1337	139.4	413.2	11.1	RIM12x-10-22-9P	15
1338	139.8	413.9	11.1	RIM12x-14-22-9P	15
1339	140.3	414.7	11.1	RIM12x-25-22-9P	15

Table A.3. Entry 14.

Run	p_0 , psia	T_0 , °C	Re_∞ , $\times 10^6 \text{ m}^{-1}$	Roughness	Rough. Offset, deg
1401	144.1	422.6	11	RIM12x-8-22-9P	0
1402	141.6	419.2	11	RIM12x-8-22-9P	0
1403	140.5	417.4	11	RIM12x-8-22-9P	0
1404	142.2	420.3	11	RIM12x-8-22-9P	3
1405	142.2	420.4	11	RIM12x-8-22-9P	3
1406	141.2	418.6	11	RIM12x-8-22-9P	3
1407	142.6	418.7	11.1	RIM12x-8-22-9P	0
1408	141.8	417.4	11.1	RIM12x-5-22-9	0
1408	134.6	411.2	10.8	RIM12x-5-22-9	0
1410	140.9	415.8	11.1	RIM12x-10-22-9P	0
1415	135.1	412.2	10.8	smooth	0
1416	140.9	418.1	11	smooth	0

Table A.4.: Entry 16.

Run	p_0 , psia	T_0 , °C	Re_∞ , $\times 10^6 \text{ m}^{-1}$	Azim. Angle, deg	Roughness	Rough. Offset, deg
1601	129.2	397.1	11	180	RIM12x-5-22-9	0
1602	134.7	407	11	180	RIM12x-5-22-9	0
1603	137.5	412	11	180	RIM12x-5-22-9	0
1604	138.4	413.6	11	180	RIM12x-5-22-9	3
1605	138.2	413.3	11	180	RIM12x-5-22-9	3
1606	139	414.6	11	180	RIM12x-5-22-9	3
1612	136.7	422.2	10.5	124	RIM12x-5-22-9	0
1613	135.3	419.6	10.5	134	RIM12x-5-22-9	0
1614	135.7	420.3	10.5	130	RIM12x-5-22-9	0
1615	134.9	418.7	10.5	140	RIM12x-5-22-9	0
1616	135.2	419.3	10.5	132	RIM12x-5-22-9	0
1617	135.3	419.6	10.5	128	RIM12x-5-22-9	0
1618	135.6	420.1	10.5	120	RIM12x-5-22-9	0
1619	135.6	420	10.5	122	RIM12x-5-22-9	0
1620	134.3	417.6	10.5	126	RIM12x-5-22-9	0
1621	134.7	418.4	10.5	144	RIM12x-5-22-9	0
1622	134.8	418.6	10.5	136	RIM12x-5-22-9	0
1623	134.5	418.1	10.5	138	RIM12x-5-22-9	0
1624	135.2	419.3	10.5	116	RIM12x-5-22-9	0
1625	134.7	418.5	10.5	118	RIM12x-5-22-9	0
1626	135.2	419.4	10.5	114	RIM12x-5-22-9	0
1627	135.2	419.3	10.5	125	RIM12x-5-22-9	0
1628	136.5	420.8	10.5	129	RIM12x-5-22-9	0
1629	135.6	420	10.5	129	RIM12x-5-22-9	0
1630	134.8	418.5	10.5	131	RIM12x-5-22-9	0
1631	135.4	419.6	10.5	121	RIM12x-5-22-9	0

Table A.4 continued from previous page

Run	p_0 , psia	T_0 , °C	Re_∞ , $\times 10^6 \text{ m}^{-1}$	Azim. Angle, deg	Roughness	Rough. Offset, deg
1632	135.2	419.3	10.5	112	RIM12x-5-22-9	0
1633	135.4	419.7	10.5	112	RIM12x-5-22-9	0
1634	135.2	419.4	10.5	127	RIM12x-5-22-9	0
1635	135.3	419.5	10.5	137	RIM12x-5-22-9	0
1636	135	419.1	10.5	117	RIM12x-5-22-9	0
1637	136.2	421.1	10.5	118	RIM12x-5-22-9	0
1638	136.2	421.2	10.5	118	RIM12x-5-22-9	0
1639	135.8	420.5	10.5	119	RIM12x-5-22-9	0
1640	135.4	419.8	10.5	135	RIM12x-5-22-9	0
1641	134.7	418.5	10.5	123	RIM12x-5-22-9	0
1642	134.6	418.2	10.5	115	RIM12x-5-22-9	0
1643	135.2	419.3	10.5	113	RIM12x-5-22-9	0
1644	135.8	420.4	10.5	133	RIM12x-5-22-9	0
1645	135.6	420	10.5	123	RIM12x-5-22-9	0
1646	131.4	412.3	10.5	140	RIM12x-5-22-9	3
1647	134	417.1	10.5	120	RIM12x-5-22-9	3
1648	136.5	421.8	10.5	112	RIM12x-5-22-9	3
1649	136.6	421.9	10.5	114	RIM12x-5-22-9	3
1650	135.7	420.3	10.5	116	RIM12x-5-22-9	3
1651	135.4	419.7	10.5	130	RIM12x-5-22-9	3
1652	135.4	419.7	10.5	134	RIM12x-5-22-9	3
1653	135.2	419.3	10.5	118	RIM12x-5-22-9	3
1654	134	417.1	10.5	128	RIM12x-5-22-9	3
1655	134.3	417.7	10.5	126	RIM12x-5-22-9	3
1656	134.8	418.5	10.5	132	RIM12x-5-22-9	3
1657	133.9	416.9	10.5	124	RIM12x-5-22-9	3
1658	135	419	10.5	122	RIM12x-5-22-9	3

Table A.4 continued from previous page

Run	p_0 , psia	T_0 , °C	Re_∞ , $\times 10^6$ m ⁻¹	Azim. Angle, deg	Roughness	Rough. Offset, deg
1659	136.7	421.1	10.5	122	RIM12x-5-22-9	0
1660	135.5	419.9	10.5	114	RIM12x-5-22-9	0
1661	135.1	419.1	10.5	132	RIM12x-5-22-9	0
1666	143.8	420.8	11.1	134	RIM-Sandblast-MkII	15
1667	143.5	420.3	11.1	138	RIM-Sandblast-MkII	15
1668	143.4	420.1	11.1	136	RIM-Sandblast-MkII	15
1669	142.4	418.4	11.1	130	RIM-Sandblast-MkII	15
1670	142.9	419.3	11.1	140	RIM-Sandblast-MkII	15
1671	143.1	419.6	11.1	132	RIM-Sandblast-MkII	15
1672	143.4	420.2	11.1	132	RIM-Sandblast-MkII	15
1673	143.5	420.4	11.1	135	RIM-Sandblast-MkII	15
1674	143.8	420.8	11.1	139	RIM-Sandblast-MkII	15
1675	143.8	420.9	11.1	131	RIM-Sandblast-MkII	15
1676	143.2	419.9	11.1	133	RIM-Sandblast-MkII	15
1677	143.5	420.3	11.1	137	RIM-Sandblast-MkII	15
1678	143.3	420	11.1	129	RIM-Sandblast-MkII	15
1679	143.5	420.3	11.1	141	RIM-Sandblast-MkII	15
1680	144	421.1	11.1	142	RIM-Sandblast-MkII	15
1681	143.6	420.6	11.1	137	RIM-Sandblast-MkII	15

Table A.5.: Entry 17.

Run	p_0 , psia	T_0 , °C	Re_∞ , $\times 10^6$ m ⁻¹	Azim. Angle, deg	Roughness	Rough. Offset, deg
1701	125.8	401	10.5	–	RIM12x-5-22-9	0
1703	136	420.8	10.5	–	RIM12x-5-22-9	0
1704	136	420.9	10.5	–	RIM12x-5-22-9	3.6
1705	141.4	419	11	–	RIM-Sandblast-MkII	0

Table A.5 continued from previous page

Run	p_0 , psia	T_0 , °C	Re_∞ , $\times 10^6 \text{ m}^{-1}$	Azim. Angle, deg	Roughness	Rough. Offset, deg
1709	141.6	419.2	11	–	RIM-Sandblast-MkII	0
1710	142.2	420.3	11	–	RIM-Sandblast-MkII	0
1713	141.5	419	11	142	RIM-Sandblast-MkII	0
1714	141.7	419.5	11	90	RIM-Sandblast-MkII	0
1715	142.7	421.1	11	90	RIM-Sandblast-MkII	0
1719	135.4	418.6	10.5	133	RIM12x-5-22-9	0
1720	135	419	10.5	129	RIM12x-5-22-9	0

B. PCB REPEATABILITY SPECTRA

The following figures show the pressure fluctuation PSDs from the PCB repeatability pairs for every angle in the Chapter 5 data set, at $Re_\infty = 10.5 \times 10^6 \text{ m}^{-1}$. See Section 5.1.2 for more details.

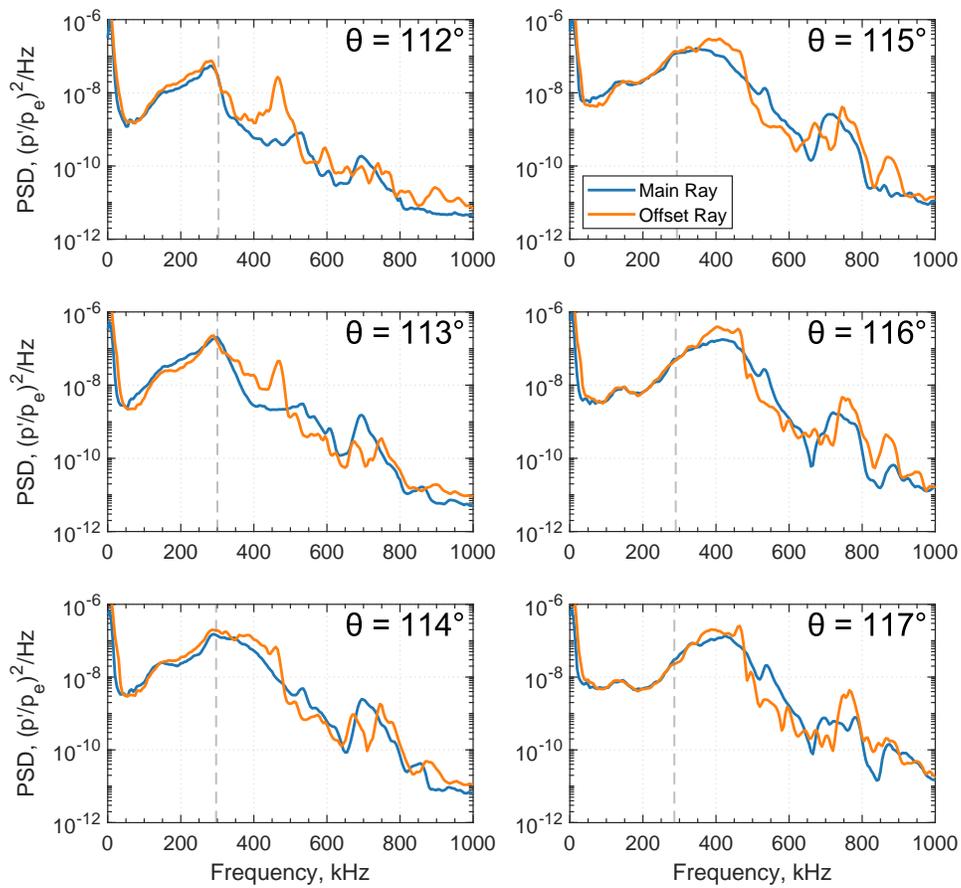


Figure B.1. PCBs 13 & 21, set 1.

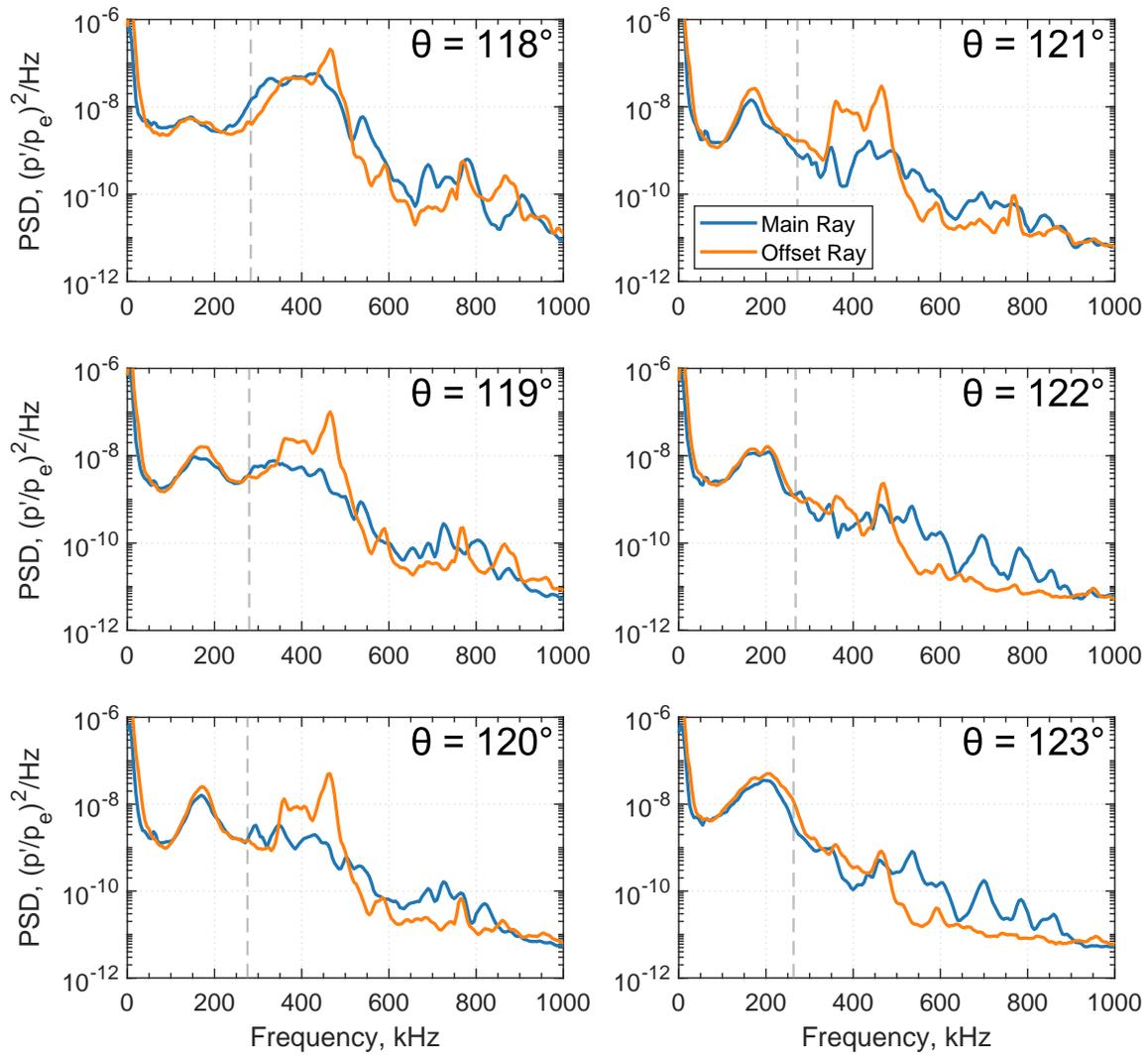


Figure B.2. PCBs 13 & 21, set 2.

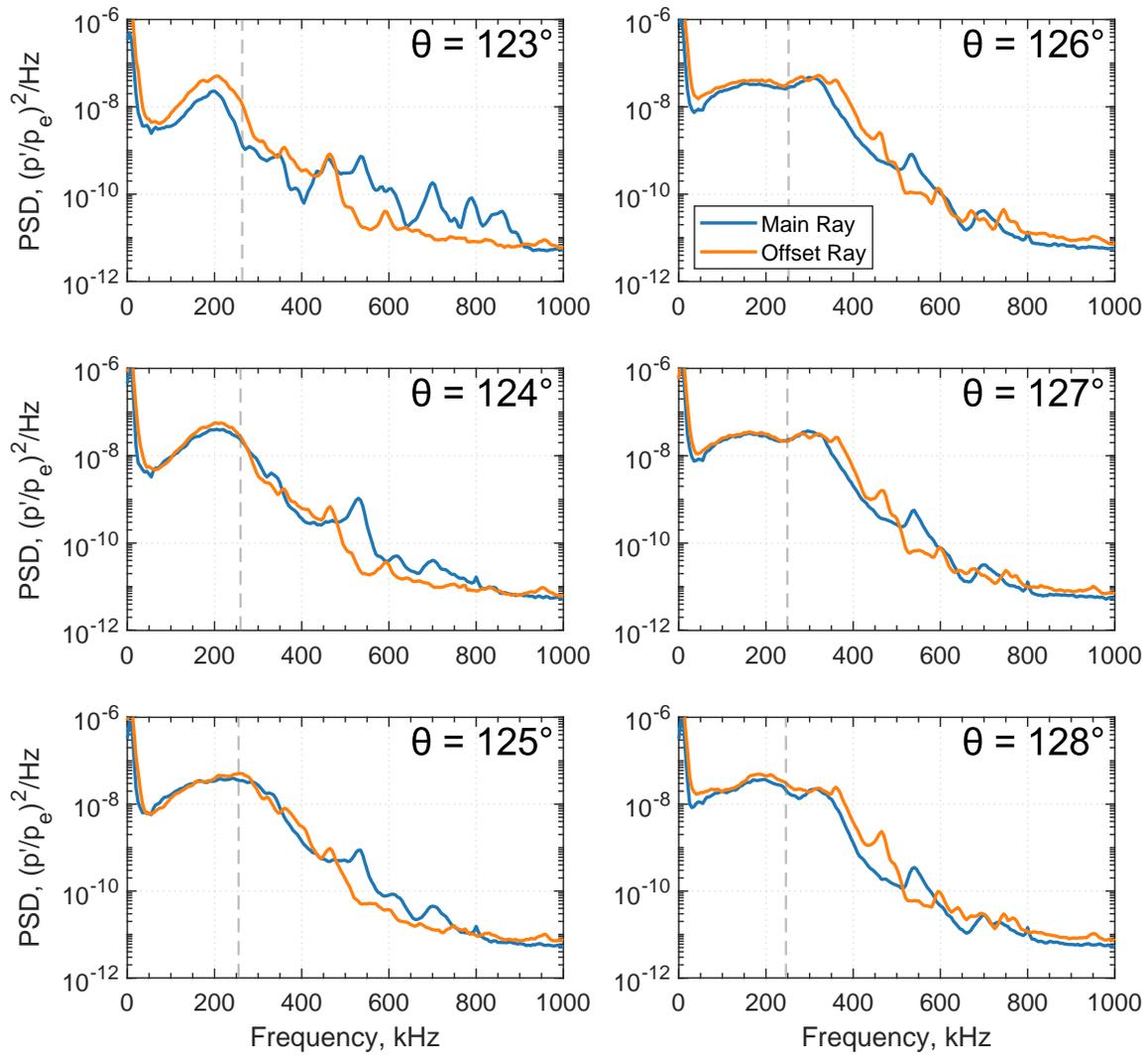


Figure B.3. PCBs 13 & 21, set 3.

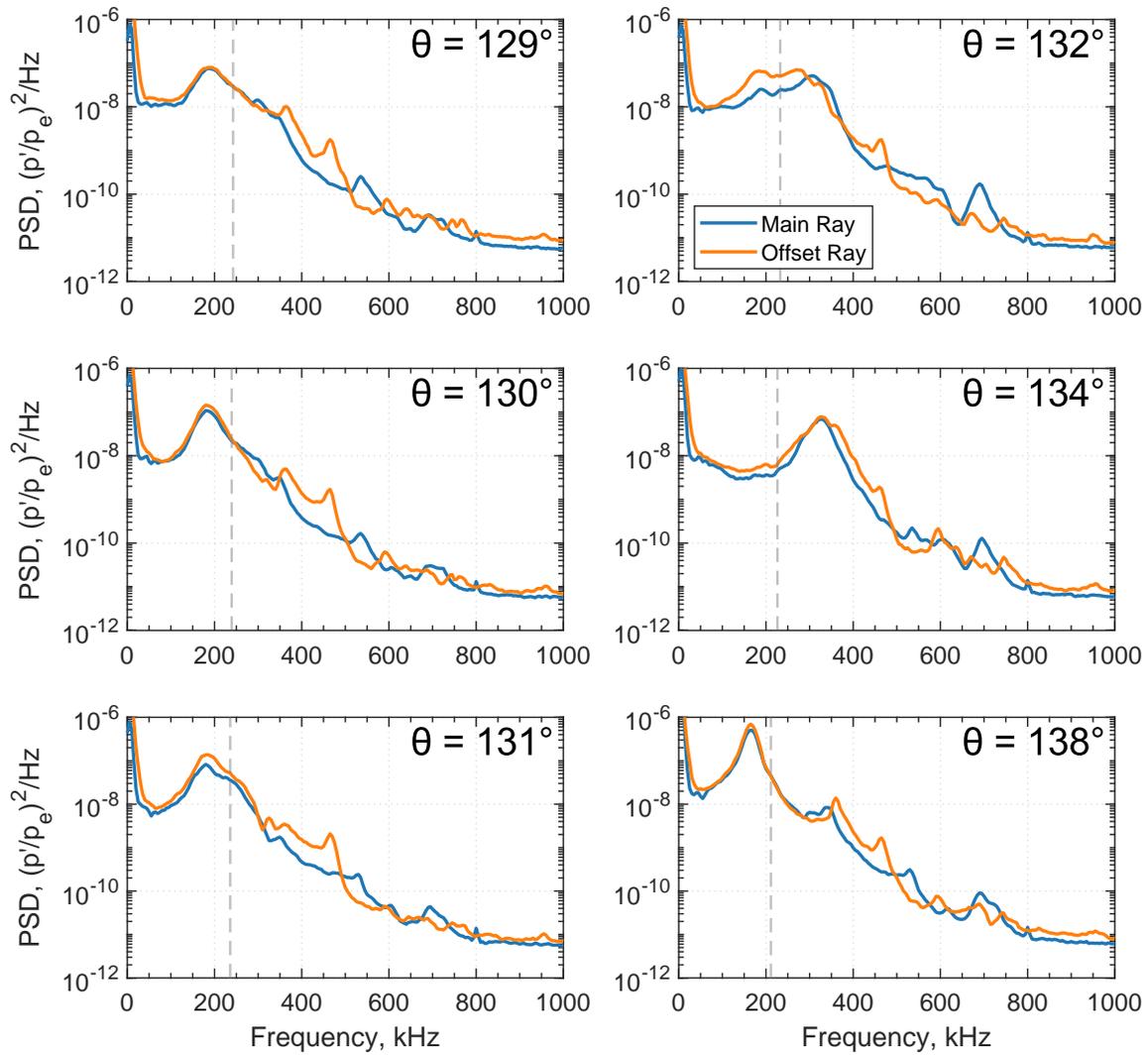


Figure B.4. PCBs 13 & 21, set 4.

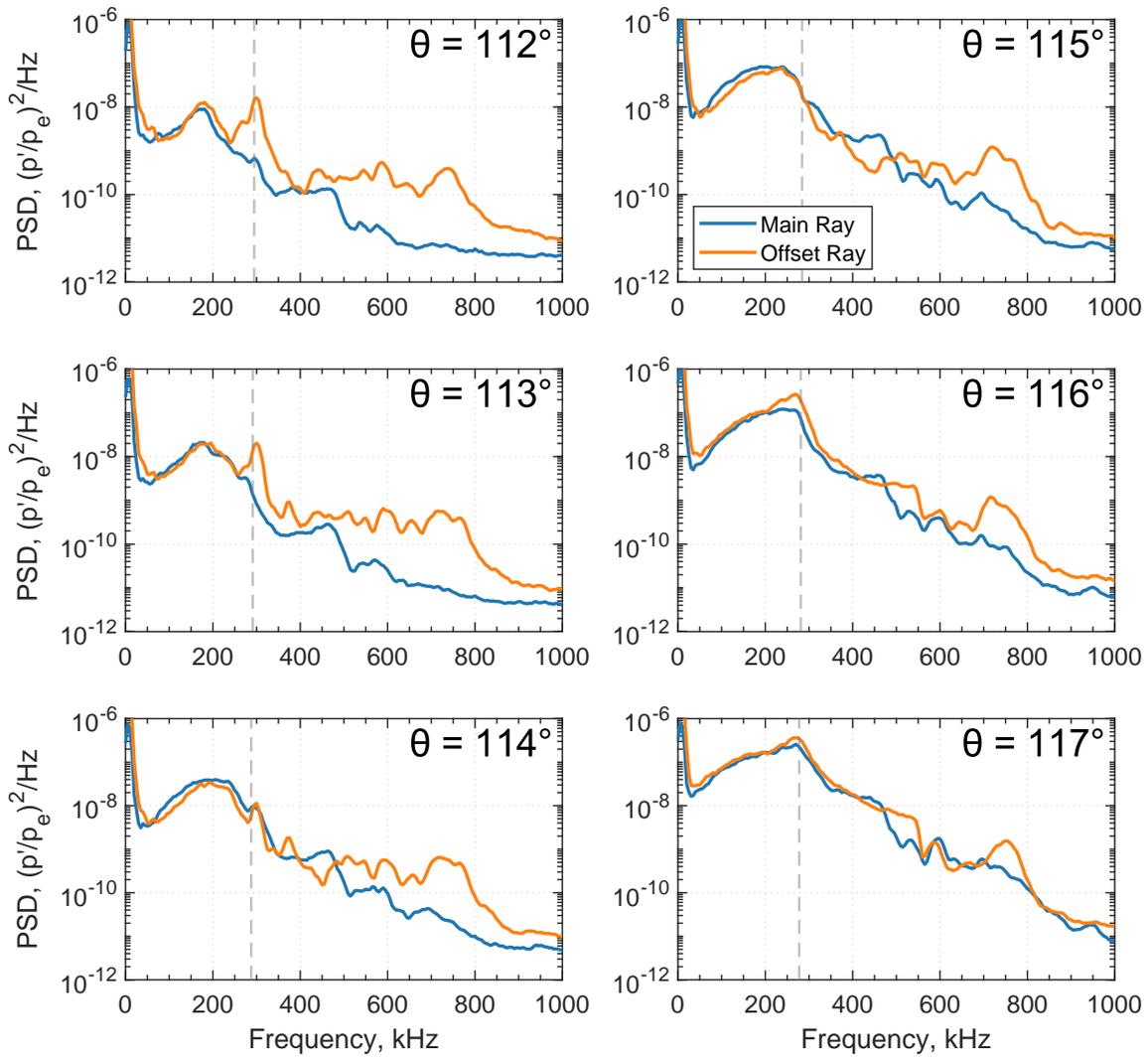


Figure B.5. PCBs 16 & 22, set 1.

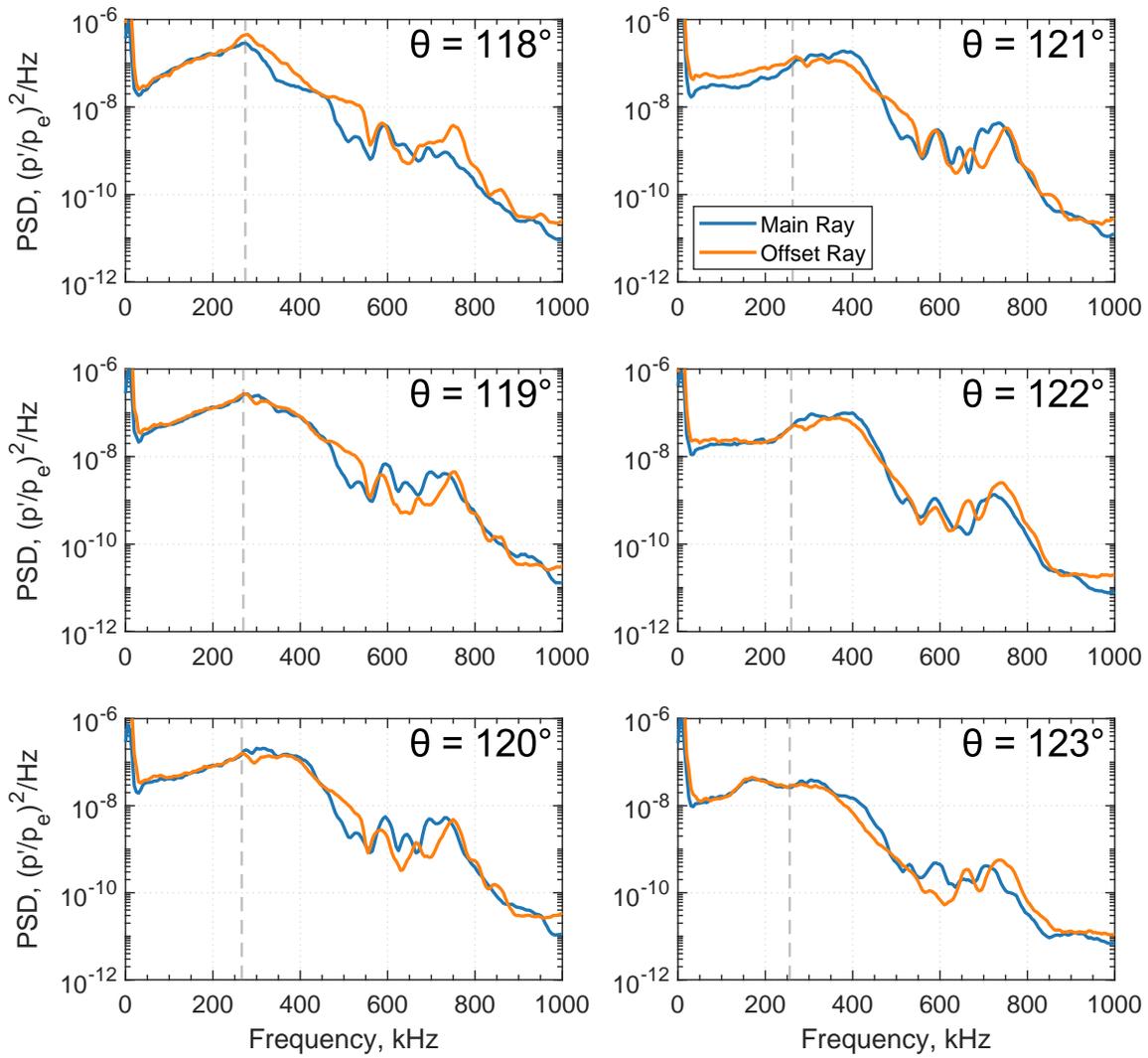


Figure B.6. PCBs 16 & 22, set 2.

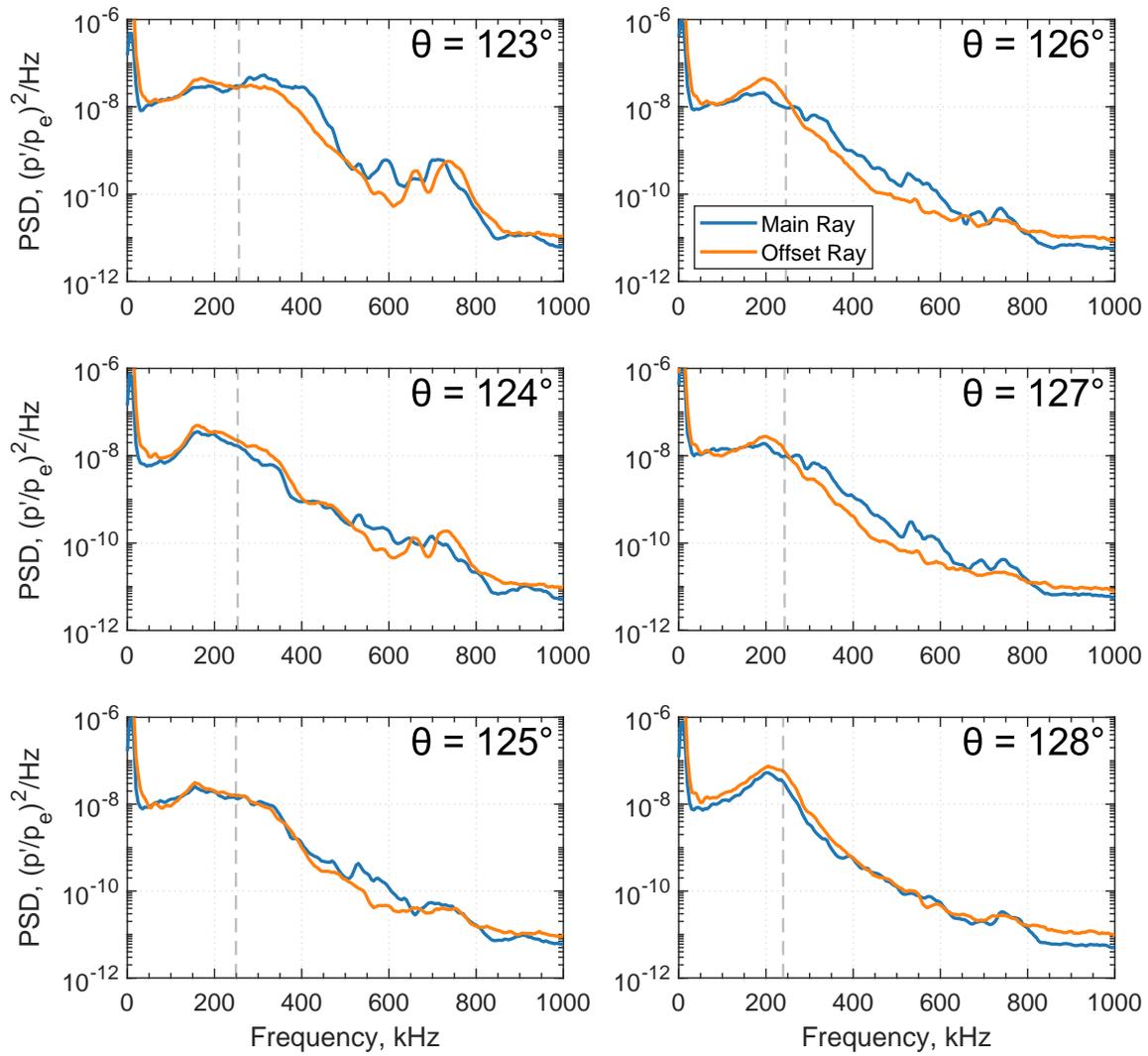


Figure B.7. PCBs 16 & 22, set 3.

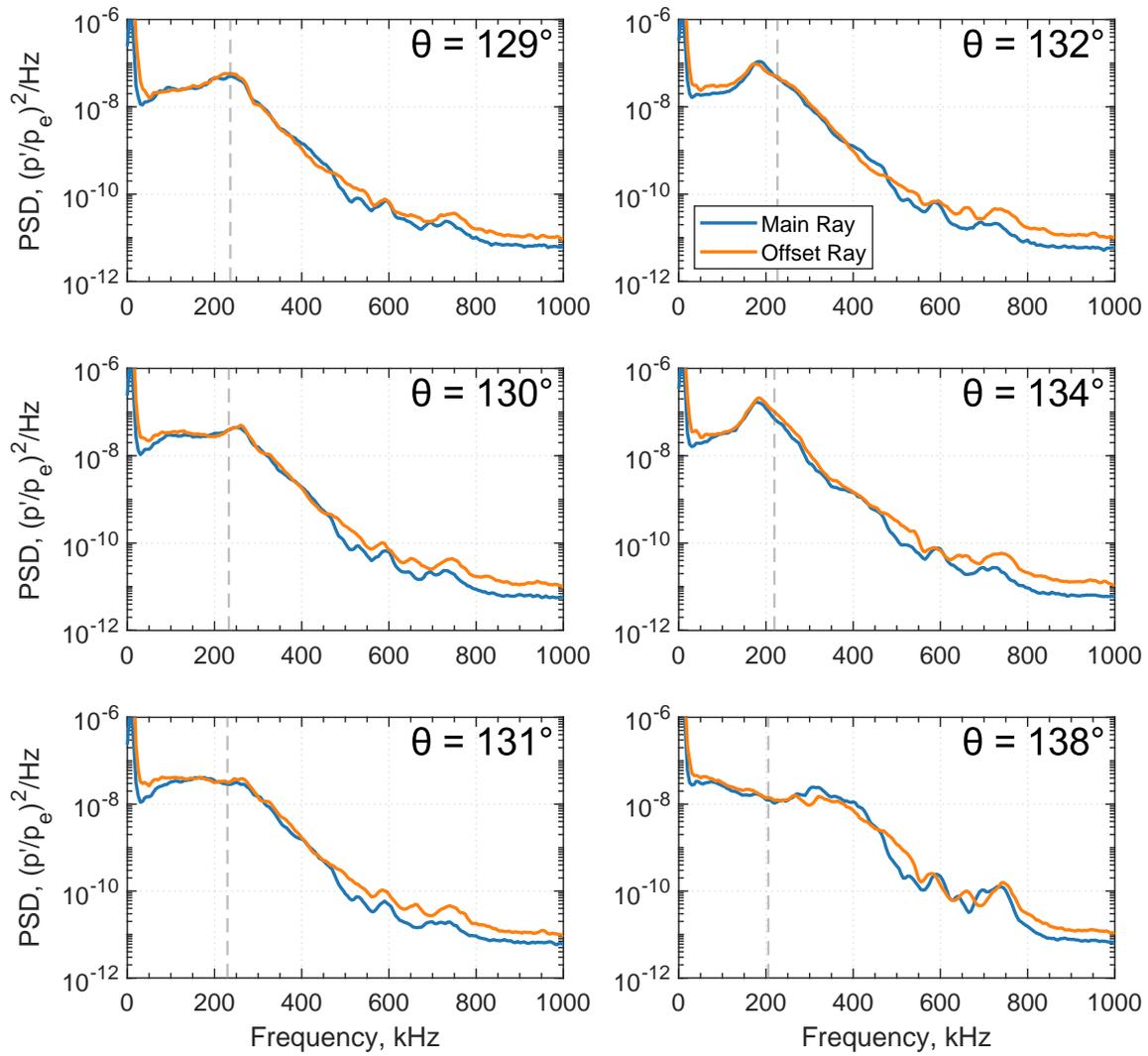


Figure B.8. PCBs 16 & 22, set 4.

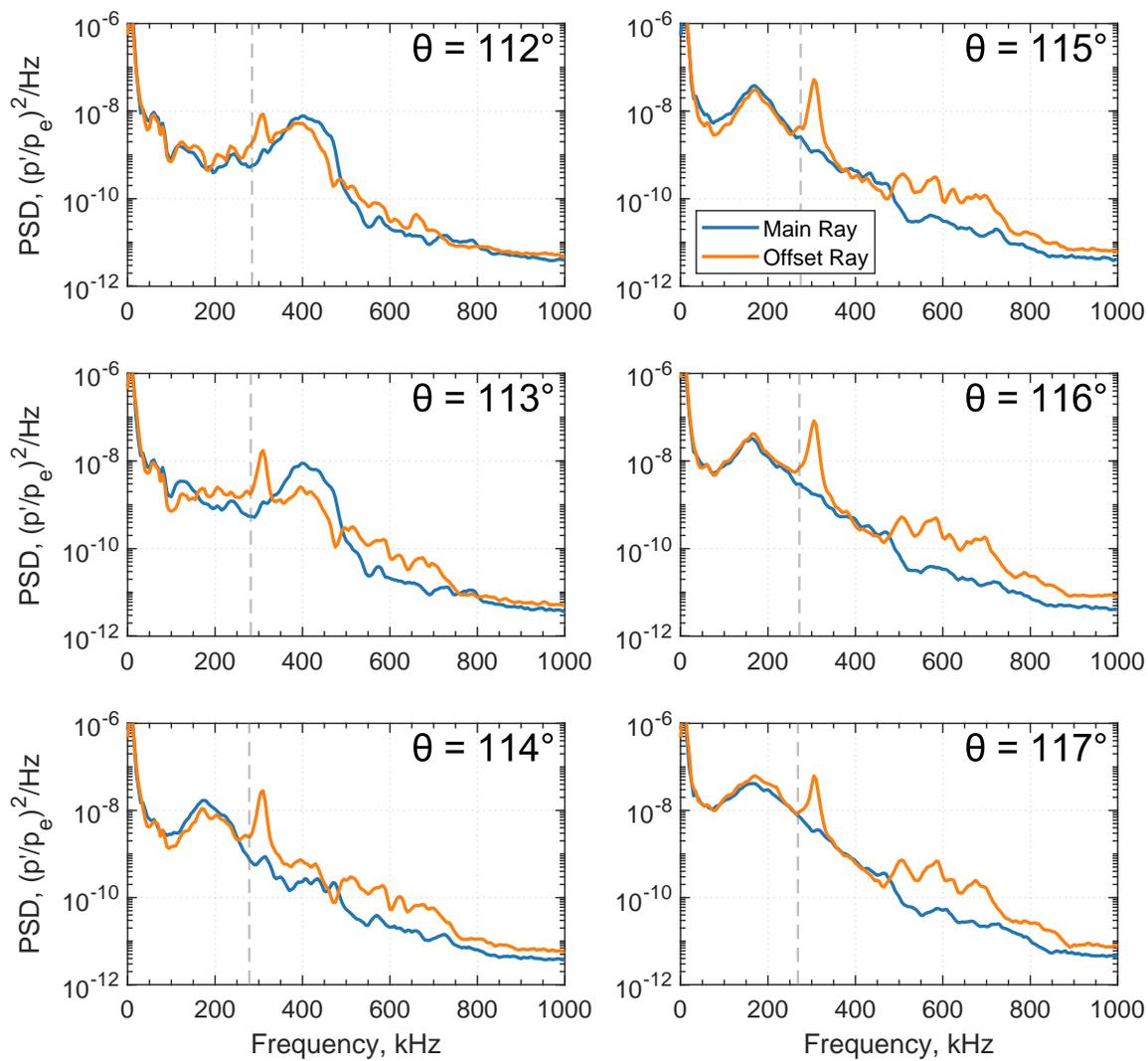


Figure B.9. PCBs 19 & 23, set 1.

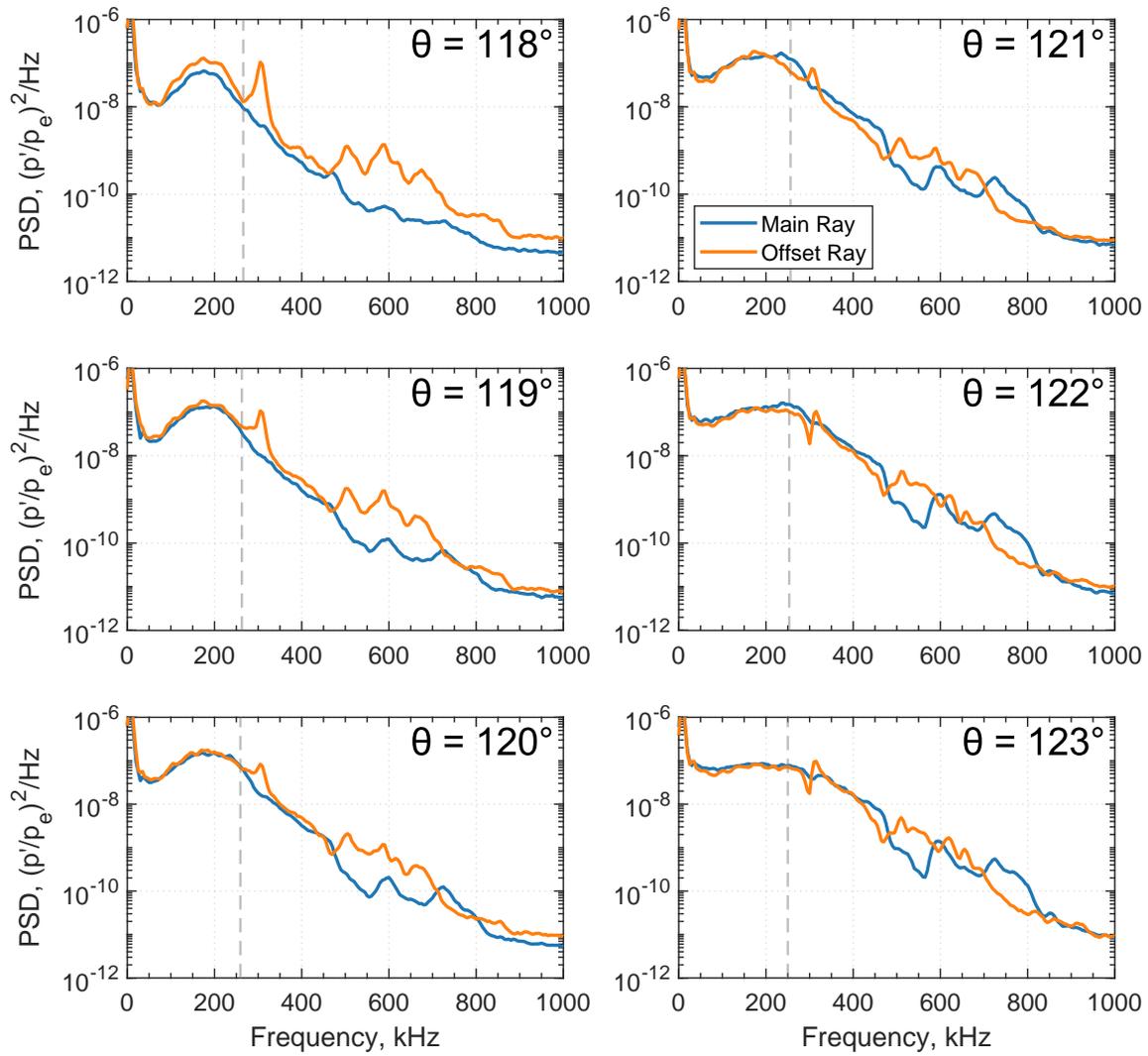


Figure B.10. PCBs 19 & 23, set 2.

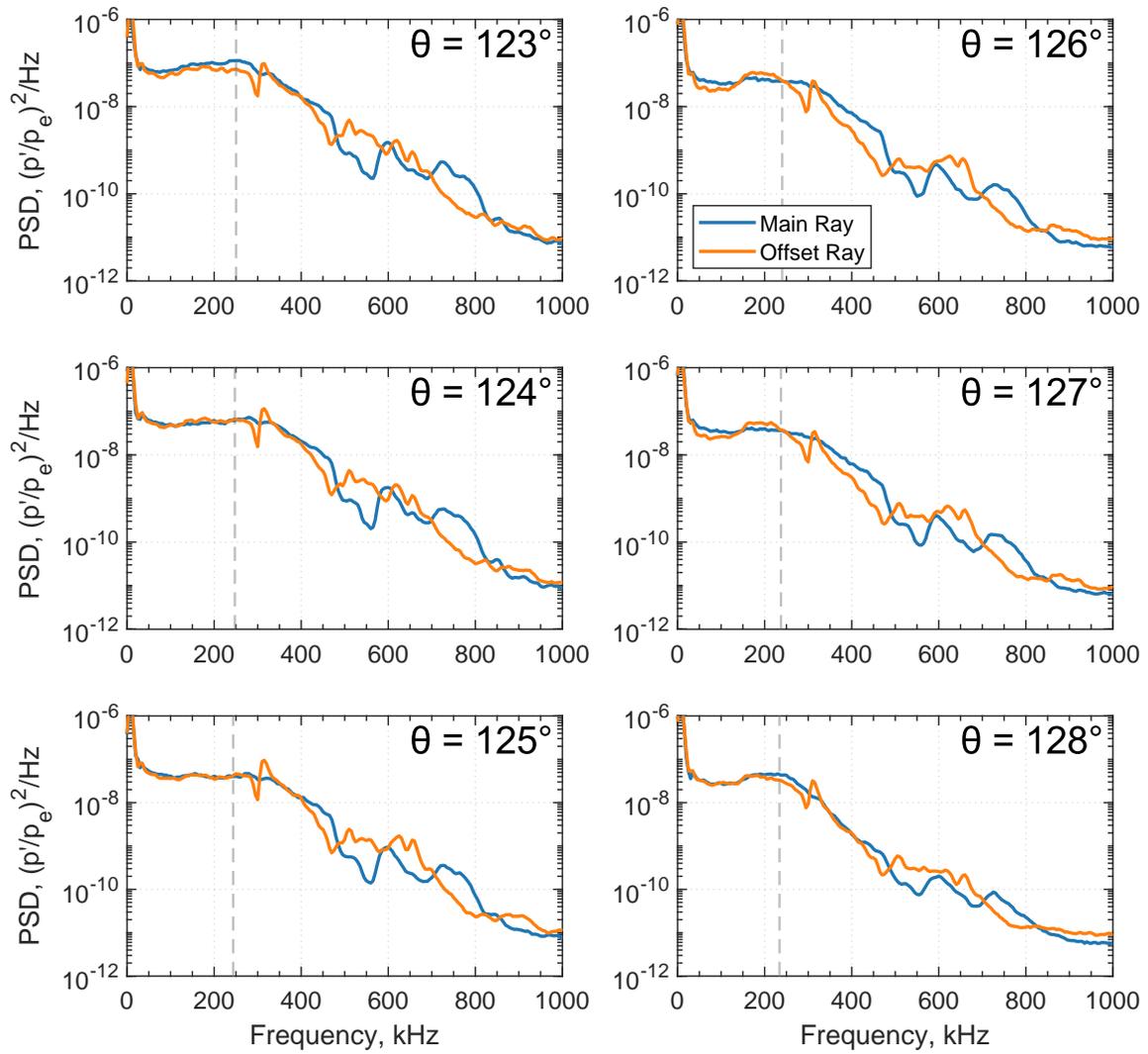


Figure B.11. PCBs 19 & 23, set 3.

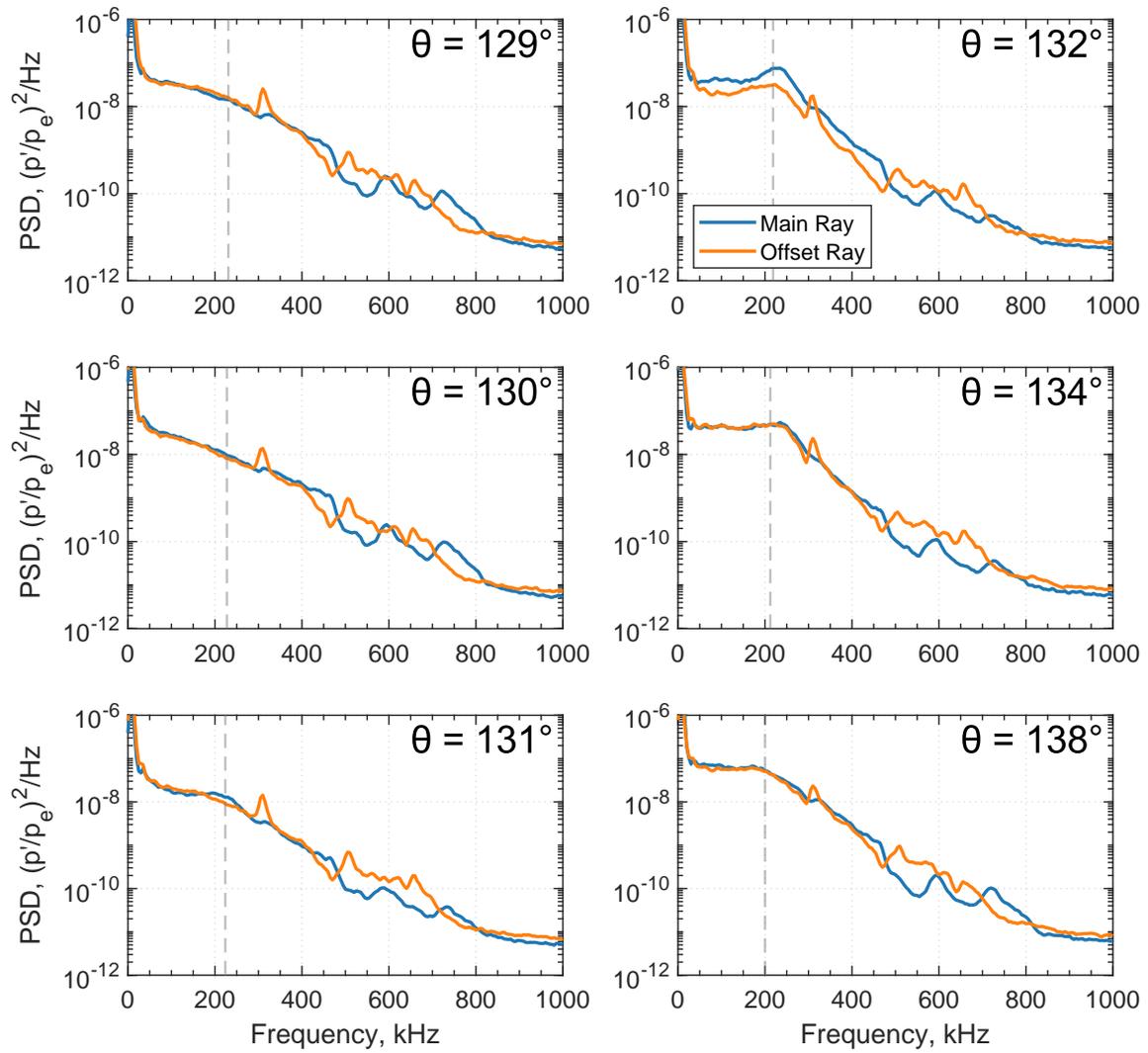


Figure B.12. PCBs 19 & 23, set 4.

C. SUPPLEMENTARY MEASUREMENTS USING SMALL DISCRETE ROUGHNESS

Earlier measurements by the author suggest that there may be a continuum of dominant transition mechanisms, from stationary and traveling crossflow to the modulated second mode. As described more fully in References [21, 73], when using a dimpled Torlon roughness with a height of $38\ \mu\text{m}$ both the traveling crossflow waves and the modulated second mode are prominent in the pressure fluctuation spectra. Figure C.1 shows this data (reproduced from Figures 5(a) and 7(a) in Edelman and Schneider [21]). There is an obvious secondary instability centered around 290 kHz, as well as traveling crossflow at 45 kHz. Interestingly, the bicoherence shows harmonic generation within both of these instabilities, as well as some limited interaction between them.

Figure C.2 shows similar data from a run using a RIM insert with elements nominally $51\ \mu\text{m}$ tall. In this case there is a noticeable traveling crossflow instability at 45 kHz, but there is no significant bicoherence at this peak. Instead, the second mode peak at 295 kHz appears to be dominating the nonlinear breakdown.

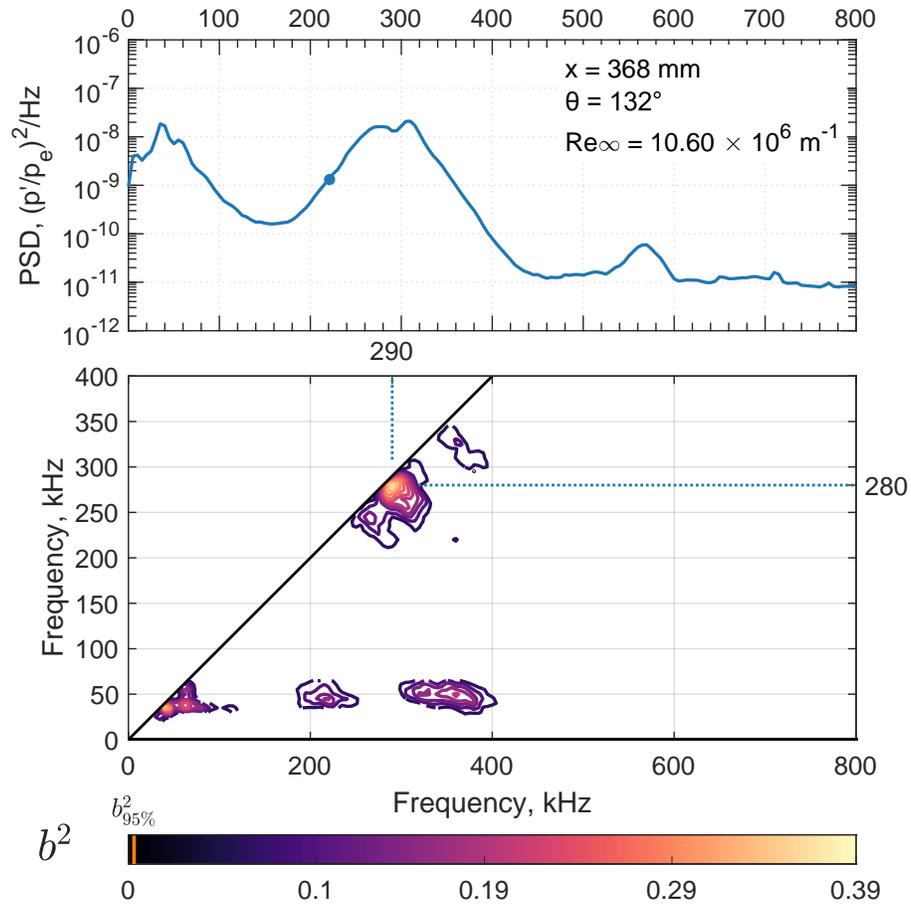


Figure C.1. PSD and bicoherence of data taken with a dimpled Torlon insert, $k \approx 38 \mu\text{m}$. Reproduced from the original data in Reference [21].

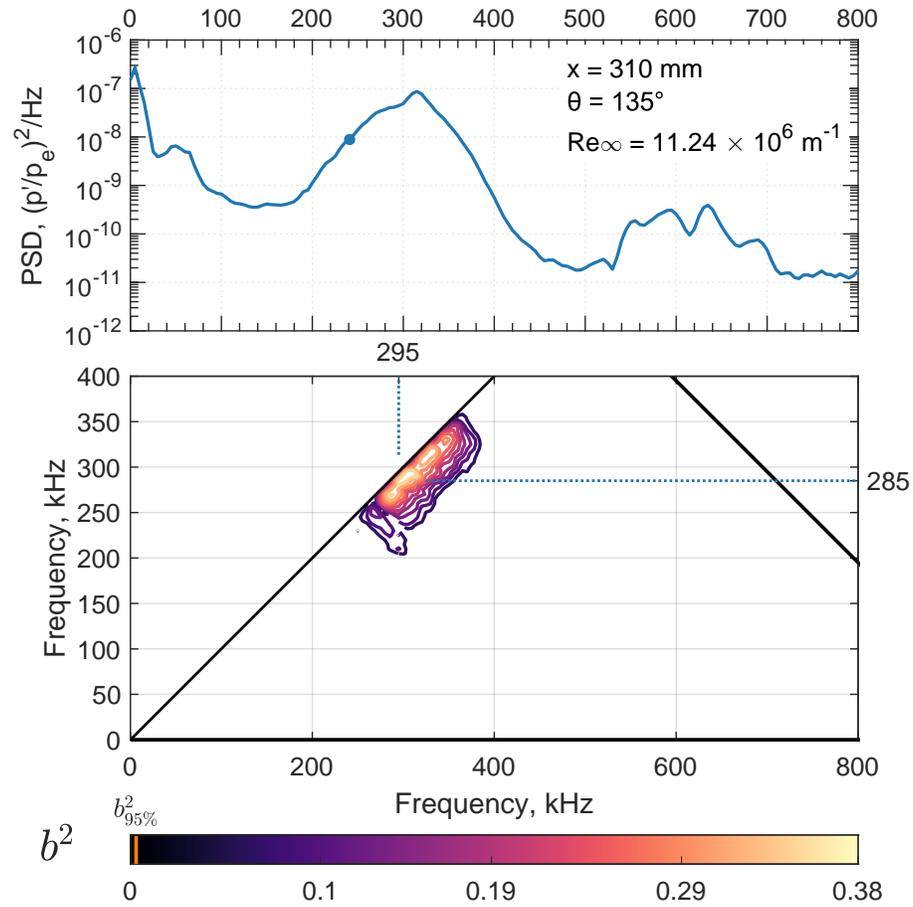


Figure C.2. PSD and bicoherence of data taken using a RIM insert with nominal height $k = 51 \mu\text{m}$.

D. THEORY OF THE IR PROCESSING TECHNIQUES

D.1 Direct Method

The direct method is—as its name implies—straight-forward and easy to implement. Both the direct and inverse methods are based heavily on Zaccara [68] and Cerasuolo [67], with minor differences in implementation to optimize the code for a faster run time. We start by defining the heat transfer problem as

$$T_t = \alpha T_{xx}, \quad (\text{D.1a})$$

$$T(x, 0) = T_0, \quad (\text{D.1b})$$

$$T(0, t) = T_s(t), \quad (\text{D.1c})$$

$$T(\infty, t) = T_0. \quad (\text{D.1d})$$

Note that this problem implicitly uses the semi-infinite assumption in the last boundary condition. However, numerically this is implemented as $T(L, t) = T_0$, where L is the last mesh point, so really the boundary condition is isothermal. However, for large L , the isothermal and semi-infinite solutions should converge to the same result.

The Purdue 1D direct method uses a finite difference scheme, discretizing the problem as

$$\frac{\partial T}{\partial t} \approx \frac{T_i^{(n+1)} - T_i^{(n)}}{\Delta t}, \quad (\text{D.2a})$$

$$\frac{\partial^2 T}{\partial x^2} \approx \frac{\theta \delta^2 T_i^{(n+1)} + (1 - \theta) \delta^2 T_i^{(n)}}{\Delta x^2}, \quad (\text{D.2b})$$

$$\delta^2 T_i \equiv T_{i-1} - 2T_i + T_{i+1}, \quad (\text{D.2c})$$

where n is the time index, i is the mesh index, and $\theta \in [0, 1]$ is a parameter which controls the degree to which the method is explicit or implicit. For most purposes, a purely implicit method $\theta = 1$ is a good choice. However, changing θ changes the numerical order and dissipation of the

method, so its 'optimal' value may be something other than 1. Determining the best choice of θ is left to the end user.

To implement the method, the finite differences are turned into a matrix equation, using

$$\beta = \frac{\alpha \Delta t}{\Delta x^2}, \quad (\text{D.3a})$$

$$\mathbf{D} = \begin{bmatrix} -2 & 1 & 0 & \cdots \\ 1 & -2 & 1 & 0 \\ 0 & 1 & -2 & 1 \\ \vdots & 0 & 1 & \ddots \end{bmatrix}. \quad (\text{D.3b})$$

The heat transfer problem is thus reduced to,

$$\vec{T}^{(n+1)} - \beta \theta \mathbf{D} \vec{T}^{(n+1)} = \vec{T}^{(n)} + \beta(1 - \theta) \mathbf{D} \vec{T}^{(n)} + \vec{b}. \quad (\text{D.4})$$

Here the vector \vec{b} contains the boundary conditions. The method marches from the initial condition forward in time by steps of Δt , solving for $\vec{T}^{(n+1)}$ as

$$\vec{T}^{(n+1)} = \mathbf{A}^{-1} \mathbf{B} \vec{T}^{(n)} + \mathbf{A}^{-1} \vec{b}, \quad (\text{D.5a})$$

$$\mathbf{A} = \mathbf{I} - \beta \theta \mathbf{D}, \quad (\text{D.5b})$$

$$\mathbf{B} = \mathbf{I} + \beta(1 - \theta) \mathbf{D}, \quad (\text{D.5c})$$

where \mathbf{I} is the identity matrix.

The boundary conditions in the finite difference form are

$$T_1 = T_s(t), \quad (\text{D.6a})$$

$$T_L = T_0. \quad (\text{D.6b})$$

The boundary conditions can be enforced by letting

$$\mathbf{D}_{1,i} = 0, \quad (\text{D.7a})$$

$$\mathbf{D}_{L,i} = 0, \quad (\text{D.7b})$$

$$\mathbf{B}_{1,i} = 0, \quad (\text{D.7c})$$

$$\mathbf{B}_{L,i} = 0, \quad (\text{D.7d})$$

$$\vec{b}_1 = T_s(t + \Delta t), \quad (\text{D.7e})$$

$$\vec{b}_L = T_0. \quad (\text{D.7f})$$

By doing so, Equation D.5a becomes

$$T_1^{(n+1)} = T_s(t + \Delta t), \quad (\text{D.8a})$$

$$T_L^{(n+1)} = T_0. \quad (\text{D.8b})$$

The direct method can be implemented easily in Matlab directly from Equation D.5a. However, the method can be optimized by a few tricks. First, the temperature vector \vec{T} can be recast as the temperature difference vector $\vec{\phi} = \vec{T} - T_0$. This simplifies the boundary and initial conditions, $\vec{\phi}(t = 0) = 0$ and $\vec{\phi}(x = L) = 0$. Second, the transition matrix $\mathbf{T} = \mathbf{A}^{-1}\mathbf{B}$ can be pre-calculated before marching through time, as none of the parameters in \mathbf{T} are dependent on time. One can also reduce computational effort by pre-calculating $\mathbf{A}^{-1}\vec{b}$, even though the boundary condition vector *does* change with each iteration. The way to do this is to note that

$$\mathbf{A}^{-1} \begin{bmatrix} \kappa \\ 0 \\ 0 \\ 0 \\ \vdots \end{bmatrix} = \kappa \mathbf{A}^{-1} \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ \vdots \end{bmatrix}, \quad (\text{D.9})$$

for some scalar κ . Thus the vector

$$\vec{A}' = \mathbf{A}^{-1} \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ \vdots \end{bmatrix} \quad (\text{D.10})$$

can be pre-computed, and multiplied by the scalar $\phi_s = T_s(t + \Delta t) - T_0$ for each step. This leads to the numerical implementation,

$$\vec{\phi}^{(n+1)} = \mathbf{T}\vec{\phi}^{(n)} + \vec{A}'\phi_s(t + \Delta t). \quad (\text{D.11})$$

Isothermal Inner Boundary, $T(L) \neq T_0(0)$:

Suppose that the initial condition $T_0(x) \neq \text{const}$. The code will force $T_0(0) = 0$, but this leaves $\vec{\phi}_0(L) = T_0(L) - T_0(0)$. It is fairly simple to add this condition to the finite difference equation, by letting

$$\vec{c} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ \vec{\phi}_0(L) \end{bmatrix}, \quad (\text{D.12})$$

and adding $\mathbf{A}^{-1}\vec{c}$ to the right hand side of Equation D.11,

$$\vec{\phi}^{(n+1)} = \mathbf{T}\vec{\phi}^{(n)} + \vec{A}'\phi_s(t + \Delta t) + \vec{A}'_{inner}. \quad (\text{D.13})$$

Because this case is isothermal, $\phi_0(L)$ is constant, and thus $\vec{A}'_{inner} = \mathbf{A}^{-1}\vec{c}$ can be computed before the time-marching.

Adiabatic Inner Boundary:

If an adiabatic back-face condition is more realistic than an isothermal one, this can also be implemented. The discretization for the heat flux at the last mesh point, $i = L$, becomes

$$-k \left(\frac{-3T_L + 4T_{L-1} - T_{L-2}}{2\Delta x} \right) = q_{inner} = 0, \quad O(\Delta x^2) \quad (\text{D.14})$$

which leads to a modification of the \mathbf{A} matrix,

$$A_{L,L} = -3, \quad (\text{D.15a})$$

$$A_{L,L-1} = 4, \quad (\text{D.15b})$$

$$A_{L,L-2} = -1. \quad (\text{D.15c})$$

The calculation can then proceed as before using Equation D.11. Note that with the new form of \mathbf{A} , the inner boundary condition term \vec{A}'_{inner} sets the level of the heat flux at this boundary. For an adiabatic boundary, $\vec{A}'_{inner} \equiv 0$, but it is possible to set a constant non-zero heat flux at this boundary as well.

D.2 Indirect Method

The direct method described above solves the so-called ‘direct’ heat transfer problem, i.e. going from a measured surface temperature to a calculated surface heat flux. One of the problems with the direct method is that a noisy temperature input becomes a noisy heat flux output. The indirect method can help to alleviate this issue without directly filtering the data. An indirect method uses the results from the ‘inverse’ problem: given a surface heat flux, what is the surface temperature?

The basic concept of the indirect method is to find the heat flux which minimizes the squared error between the measured surface temperature and the calculated surface temperature from the guessed heat flux. Because this nonlinear optimization is computationally expensive, the Purdue approach (from Zaccara and Cerasuolo, [67, 68]) uses the direct method to calculate heat flux up to a time of interest, then uses the indirect method to proceed. The direct method is the same as that described in the previous section, so it will not be discussed further here.

The method of Zaccara and Cerasuolo optimizes for the the convective heat transfer coefficient h , from the relation

$$q_s = h(T_w - T_{aw}) + \sigma\epsilon(T_w^4 - T_{surr}^4). \quad (\text{D.16})$$

However, the convective coefficient is not of immediate importance to most researchers in the BAM6QT and the equations become much trickier, so optimizing for the heat flux q_s itself is what will be examined first. Note that the optimized q_s will include a radiation component which has been separately considered in the Italian students' formulation. However, convective heat transfer dominates in the conditions of the BAM6QT, so the radiative component can be considered negligible. The original method using h will be discussed after.

The indirect method starts by guessing a surface heat flux, \hat{q}_s . The optimization method is nonlinear, so it is critical that the initial guess is close to the correct solution, or the optimization could find a local minimum instead of a global one. A two-step method, which calculates a guess for \hat{q}_s , is discussed in Section D.4. For many purposes, however, the guess can be the average heat flux that one expects in these conditions, provided by a Schmidt-Boelter gauge or experience. However, it is vital that the results be checked appropriately when using this method, to ensure that it has indeed found the global minimum.

Using the initial guess (which is really a time series, $\hat{q}_s[n]$), the method calculates the surface temperature via the same matrix equation provided in Equation D.11. However, the form of A is slightly different to accommodate the new boundary condition:

$$A_{1,1} = -\frac{3k}{2\Delta x}, \quad (\text{D.17a})$$

$$A_{1,2} = \frac{2k}{\Delta x}, \quad (\text{D.17b})$$

$$A_{1,3} = -\frac{k}{2\Delta x}, \quad (\text{D.17c})$$

$$\vec{b}_1 = -\hat{q}_s. \quad (\text{D.17d})$$

This new discretization is derived in the same manner as for the adiabatic inner boundary, shown in Equation D.14. Note that the vector \hat{A}' can be pre-calculated and used as in the direct method, except this time multiplying by $\hat{q}_s(t + \Delta t)$ for each iteration instead of the experimental surface temperature.

Once the surface temperatures from the inverse solution have been calculated for the time region of interest, they are compared to the experimental temperatures. The error metric is the sum of the squared error, $\sum(T_{inverse} - T_{experiment})^2$. The nonlinear optimization routine `fmincon` in Matlab is used to minimize this squared error, subject to a minimum and maximum \hat{q}_s . The output from the optimization routine is the ‘best’ guess for the surface heat flux which yields the measured surface temperatures.

Optimization for Convective Coefficient h :

If you would like to maintain the separation between convection and radiation, or if you are more interested in the convective coefficient than the heat flux itself, this can be implemented in as follows (generally from Refs. [67, 68], with optimization improvements).

The full boundary condition at the surface is

$$k \frac{\partial T}{\partial x} \Big|_{x=0} = h(T_s - T_{aw}) + \sigma \epsilon (T_s^4 - T_{surr}^4). \quad (D.18)$$

Unfortunately this equation is highly-nonlinear (the radiative component has T_s^4), so it must be linearized to proceed via an efficient matrix solver. This can be accomplished by using the experimental surface temperature in the radiative term,

$$k \frac{-3T_1 + 4T_2 - T_3}{2\Delta x} \approx h(T_1 - T_{aw}) + \sigma \epsilon (T_{exp}^4 - T_{surr}^4). \quad (D.19)$$

This leads to the modifications to the \mathbf{A} and \vec{b} terms,

$$A_{1,1} = -\frac{3k}{2\Delta x} - h, \quad (D.20a)$$

$$A_{1,2} = \frac{2k}{\Delta x}, \quad (D.20b)$$

$$A_{1,3} = -\frac{k}{2\Delta x}, \quad (D.20c)$$

$$\vec{b}_1 = -hT_{aw} + \sigma \epsilon (T_{exp}^4 - T_{surr}^4). \quad (D.20d)$$

The calculation then proceeds as before.

Note that the matrix $\mathbf{A}(h)$ is now dependent on the optimization variable h in the (1,1) index, so it must be recalculated at each time step and for each successive run of the optimization routine, which is quite computationally expensive. However, much of the inverse of \mathbf{A} can be pre-computed, which should reduce the calculation time by a non-negligible fraction.

We optimize the calculation by noting that

$$(\mathbf{A} + h\mathbf{G})^{-1} = \mathbf{A}^{-1} - \frac{1}{1 + \text{tr}(h\mathbf{G}\mathbf{A}^{-1})} \mathbf{A}^{-1}(h\mathbf{G})\mathbf{A}^{-1}, \quad (\text{D.21})$$

assuming that $\text{rank}(\mathbf{G}) = 1$ and both \mathbf{A} and $\mathbf{A} + h\mathbf{G}$ are invertible. Thus if we let

$$\mathbf{G} = \begin{bmatrix} -1 & 0 & \cdots \\ 0 & 0 & \\ \vdots & & \ddots \end{bmatrix}, \quad (\text{D.22})$$

and \mathbf{A} does *not* contain the $-h$ term, then the transition matrix $\mathbf{T}(h)$ can have several of its components pre-computed as

$$\mathbf{T}(h) = \mathbf{A}^{-1} - \frac{h}{1 + h \cdot g} \mathbf{B}', \quad (\text{D.23a})$$

$$g = \text{tr}(\mathbf{G}\mathbf{A}^{-1}), \quad (\text{D.23b})$$

$$\mathbf{B}' = \mathbf{A}^{-1}\mathbf{G}\mathbf{A}^{-1}. \quad (\text{D.23c})$$

In this way the transition matrix computation only has to subtract two matrices for every different $h(t)$ instead of the relatively costly solution of the $n \times n$ linear system. However, this requires the explicit computation of \mathbf{A}^{-1} which may be less accurate and more time-consuming than the Matlab linear-solution algorithm.

D.2.1 A More Efficient Algorithm

The indirect algorithm just described is simple to understand and execute, but it is quite inefficient. For every point, the solver must compute the surface temperature for the entire measurement range, and because the initial guess is never perfect, this operation will take place

hundreds of times per point. The result is a very large number of operations even with small mesh sizes. A more efficient algorithm can be derived by recognizing that the optimization function only needs to calculate the surface temperature; the value of any of the interior temperatures is irrelevant, except at $t = 0$. Starting from Equation D.13 and using q_s instead of ϕ_s , it can easily be shown that the surface temperature at any time t depends only on the initial temperature configuration and the surface heat flux,

$$\vec{\phi}(\Delta t) = \mathbf{T}\vec{\phi}(0) - \vec{A}'q_s(0), \quad (\text{D.24})$$

$$\begin{aligned} \vec{\phi}(2\Delta t) &= \mathbf{T}\vec{\phi}(\Delta t) - \vec{A}'q_s(\Delta t) \\ &= \mathbf{T}^2\vec{\phi}(0) - \mathbf{T}\vec{A}'q_s(0) - \vec{A}'q_s(\Delta t), \end{aligned} \quad (\text{D.25})$$

$$\vdots$$

$$\vec{\phi}(N\Delta t) = \mathbf{T}^N\vec{\phi}(0) - \mathbf{T}^{N-1}\vec{A}'q_s(0) - \dots - \mathbf{T}\vec{A}'q_s(N-2) - \vec{A}'q_s(N-1). \quad (\text{D.26})$$

Fortunately, the optimizer only cares about the surface temperature, $\vec{\phi}_1$, so these operations can be vectorized even further by selecting only the first element of $\vec{\phi}$ and forming them into a vector, as follows,

$$\vec{\Phi} = \begin{bmatrix} \vec{\phi}_1(0) \\ \vec{\phi}_1(\Delta t) \\ \vdots \\ \vec{\phi}_1(N\Delta t) \end{bmatrix} = \begin{bmatrix} \overbrace{\begin{bmatrix} 1 & 0 & \cdots & 0 & 0 \\ - & T_{1,j} & - \\ - & T_{1,j}^2 & - \\ & \vdots & \\ - & T_{1,j}^N & - \end{bmatrix}}^{\mathbf{D}} & \vec{\phi}(0) \\ - \underbrace{\begin{bmatrix} 0 & 0 & \cdots & 0 & 0 \\ \vec{A}'_1 & 0 & \cdots & 0 & 0 \\ T_{1,j}\vec{A}' & \vec{A}'_1 & \cdots & 0 & 0 \\ \vdots & & & & \\ T_{1,j}^N\vec{A}' & T_{1,j}^{N-1}\vec{A}' & \cdots & \vec{A}'_1 & 0 \end{bmatrix}}_{\mathbf{F}} & \begin{bmatrix} q_s(0) \\ q_s(\Delta t) \\ q_s(2\Delta t) \\ \vdots \\ q_s(N\Delta t) \end{bmatrix} \end{bmatrix}, \quad (\text{D.27})$$

where $T_{1,j}^k$ is the first row of \mathbf{T}^k , such that $T_{1,j}^k \vec{A}'$ is a scalar. We immediately note that the matrix \mathbf{F} is the lower triangular portion of a Toeplitz matrix, which can be formed from a vector very efficiently in Matlab using the `toeplitz` and `tril` functions. In addition, $T_{1,j}$ and all of its powers are known before the calculation begins. Thus the function to optimize becomes

$$\min E(\vec{q}_s) = \sum \left(\vec{\Phi}_{\text{exp}} - \vec{\Phi} \right)^2 = \sum \left(\vec{\Phi}_{\text{exp}} - \mathbf{D}\vec{\phi}(0) + \mathbf{F}\vec{q}_s \right)^2, \quad (\text{D.28})$$

where the only large computation that must be made in the optimization routine is the product $\mathbf{F}\vec{q}_s$, as \mathbf{D} and \mathbf{F} can be formed before any computation begins, and $\mathbf{D}\vec{\phi}(0)$ can be computed before the optimization begins for each mesh point.

For moderate numbers of interior points, this alternative algorithm leads to a substantial increase in performance (around a 30 % decrease in computation time). However, for very large numbers of interior points the performance will certainly degrade, as it becomes more computationally intensive to generate \mathbf{D} and \mathbf{F} , though it may still be faster than the naïve implementation for the same number of points. However, interior mesh sizes of around 25 points are more than adequate to get good heat transfer data, and \mathbf{D} and \mathbf{F} can be formed at this size in negligible time.

The same algorithm can be applied to the direct method, but the performance increase is not nearly as significant. Note also that this algorithm is numerically unstable for large numbers of time points. As k increases, the estimate T^k will become corrupted by accumulating numerical errors.

D.3 Fourier-Based Method

The analytical solution to the semi-infinite problem with a sinusoidally varying surface temperature can be derived as follows. Suppose the heat transfer problem is given as,

$$T_t = \alpha T_{xx}, \quad (\text{D.29a})$$

$$T(x, 0) = 0, \quad (\text{D.29b})$$

$$T(0, t) = \hat{T}_s e^{j\omega t}, \quad t > 0. \quad (\text{D.29c})$$

The solution $T(x, t)$ can then be assumed to vary sinusoidally as well,

$$T(x, t) = g(x)e^{j\omega t}. \quad (\text{D.30})$$

Substituting this into the problem yields

$$\begin{aligned} j\omega g(x)e^{j\omega t} &= \alpha g''(x)e^{j\omega t}, \\ g''(x) - \frac{j\omega}{\alpha} g(x) &= 0, \end{aligned} \quad (\text{D.31})$$

which can be readily solved to give

$$g(x) = C \exp\left(- (1 + j) \sqrt{\frac{\omega}{2\alpha}} x\right), \quad (\text{D.32a})$$

$$T(x, t) = \hat{T}_s \exp\left(- (1 + j) \sqrt{\frac{\omega}{2\alpha}} x\right) e^{j\omega t}. \quad (\text{D.32b})$$

Note that the negative root of the characteristic equation is chosen to satisfy the infinity boundary condition for $g(\infty) \rightarrow 0$. If $\omega < 0$, the positive root must be chosen for the same reason. In this case, however, $g(x)_{\omega < 0} = g(x)_{\omega > 0}^*$.¹

For an arbitrary surface temperature function $T_s(t) = f(t)$, the function can be decomposed into its Fourier modes,

$$f(t) = \sum c_n e^{j\omega_n t}. \quad (\text{D.33})$$

The problem is linear, and thus the solution for the arbitrary function becomes

$$T(x, t) = \sum c_n \exp\left(- (1 + j) \sqrt{\frac{\omega_n}{2\alpha}} x\right) e^{j\omega_n t}. \quad (\text{D.34})$$

For the case when $n = 0$, i.e. if there is a non-zero average of $f(t)$, we must use the semi-infinite solution to the unit step input:

$$T_0(x, t) = c_0 \operatorname{erfc}\left(\frac{x}{\sqrt{4\alpha t}}\right), \quad (\text{D.35})$$

where erfc is the complementary error function, $1 - \operatorname{erf}$.

The surface heat flux is defined as $q_s = -kT_x|_{x=0}$, which can be evaluated analytically for the series as

$$a_{n,n>0} = \sqrt{\frac{|\omega_n|}{2\alpha}} (1 + j), \quad (\text{D.36a})$$

$$a_{n,n<0} = a_{n,n>0}^*, \quad (\text{D.36b})$$

$$q_s(t) = k \left(\sum_{n \neq 0} c_n a_n e^{j\omega_n t} + \frac{c_0}{\sqrt{\pi\alpha t}} \right). \quad (\text{D.36c})$$

Numerical Implementation

As mentioned, a numerical solution of the PDE on a 1D mesh is time consuming due to the matrix operations involved and the necessity of keeping around the entire temperature solution

¹Note that $\sqrt{-j\omega_n} = j(j+1)\omega_n/\sqrt{2} = (j-1)\omega_n/\sqrt{2} = -(1-j)\omega_n/\sqrt{2} = \overline{-(1+j)\omega_n/\sqrt{2}}$.

along the mesh. A fast fourier transform, or `fft`, operation is much faster. The numerical implementation of the above theory takes advantage of this speed.

The same operation loops over every pixel in the analysis region of the IR thermograph. The surface temperature trace is turned into a $\phi(t) = T_s(t) - T_s(0)$ for each pixel. First, an `fft` is taken of the surface temperature time trace giving $[c_n]$. A frequency vector $[\omega]$ is also created. Then the $[a_n]$ vector is formed from $[\omega]$. The $[c_n]$ and $[a_n]$ vectors are combined and the `ifft` is taken. Finally, the c_0 case is added. This algorithm runs 2 orders of magnitude faster than the inverse solution method.

A complication arises for this method due to the finite length of the data. The discrete Fourier transform implicitly assumes that the data are periodic in time, regardless of the actual nature of the measured signal. Thus if the experimental temperature is transformed as-is, there will be significant ringing in the solution due to the step discontinuity between the last measurement point and the first one. In addition, a finite data record leads to a small but non-zero frequency resolution, so in general the discrete Fourier transform should be extended (often via zero padding).

For the purposes of calculating the heat transfer, however, zero-padding the temperature data is impractical. Adding a large amount of zeros to the end of the record will reduce the average temperature over the entire record, which in turn will affect the calculated response through the c_0 term. A better method is to extend the signal with a continuous curve from the last point to the first point, thereby creating a periodic signal that is continuous by design. This is accomplished in the code using a cubic Bézier curve, which matches the value and slope at the last point and the same properties at the first point. The result is a continuous, periodic signal with continuous first derivatives.

2D Method

A 2D theory can also be developed following similar logic. In this case, we examine a slice of the cone at a constant axial position and consider both lateral and radial conduction. Again,

because the penetration depth is so small compared to the thickness of the PEEK, a semi-infinite model can be considered. The heat conduction problem in this case is then,

$$T_t = \alpha(T_{rr} + r^{-1}T_r + r^{-2}T_{\theta\theta}), \quad (\text{D.37a})$$

$$T(r, \theta, 0) = 0, \quad (\text{D.37b})$$

$$T(R, \theta, t) = T_s(\theta, t). \quad (\text{D.37c})$$

The semi-infinite model on a polar domain implies that $r \gg 1 \implies r^{-1} \ll 1$. In addition, define the arclength coordinate z as

$$z = r\theta \implies r\partial\theta = \partial z. \quad (\text{D.38})$$

Furthermore, to make the boundary condition easier, let $x = R - r$, such that

$$T_{rr} = T_{xx}, \quad (\text{D.39a})$$

$$T(0, z, t) = T_s(z, t). \quad (\text{D.39b})$$

Thus the problem can be redefined as

$$T_t = \alpha (T_{xx} + T_{zz}). \quad (\text{D.40})$$

Suppose the boundary condition and solution can be written as the double sum of Fourier modes in both z and t ,

$$T_s(z, t) = \sum_m \sum_n c_{mn} \exp(j\omega_n t) \exp(j\omega_m z), \quad (\text{D.41a})$$

$$T(x, z, t) = \sum_m \sum_n c_{mn} g(x) \exp(j\omega_n t) \exp(j\omega_m z), \quad (\text{D.41b})$$

where $g(x)$ is a shape function which must satisfy

$$g(0) = 1, \quad (\text{D.42a})$$

$$g(\infty) \rightarrow 0, \quad (\text{D.42b})$$

to satisfy the boundary conditions of the problem. Using the orthogonality properties of the Fourier decomposition, we can say each mode T_{mn} must solve the PDE. Substituting T_{mn} into the equation yields the ODE,

$$j\omega_n g(x) = \alpha \left(g''(x) - \omega_m^2 g(x) \right). \quad (\text{D.43})$$

This can be readily rearranged to give the solution,

$$g''(x) - \left(\omega_m^2 + \frac{j\omega_n}{\alpha} \right) g(x) = 0, \quad (\text{D.44a})$$

$$g(x) = e^{\lambda x}, \quad (\text{D.44a})$$

$$\lambda = -\sqrt{\omega_m^2 + j\omega_n/\alpha}. \quad (\text{D.44b})$$

Here only the negative root has been chosen to satisfy $g(\infty) \rightarrow 0$. Equations (28) and (29) represent the solution $g(x)$ for $m \neq 0$ and $n \neq 0$. The zero-frequency cases must be treated separately.

Case where $m = 0$:

In the case where $m = 0$, there is no frequency content in the z direction, i.e. $T_s(z, t) = T_s(t)$. Since there is no variation in the z direction there can be no lateral conduction. This case can then be treated as the 1D case, yielding the solution

$$T_{0n} = c_{0n} \exp \left(-(1+j) \sqrt{\frac{\omega_n}{2\alpha}} x \right) e^{j\omega_n t}. \quad (\text{D.45})$$

Case where $m = n = 0$:

For the case where there is no frequency content in t or z , i.e. $T_s = \text{const.}$, again there can be no lateral conduction. The solution is then the 1D solution to a step input,

$$T_{00} = c_{00} \text{erfc} \left(\frac{x}{\sqrt{4\alpha t}} \right). \quad (\text{D.46})$$

Case where $n = 0$:

The final case is when there is some arbitrary z dependence but only a step in time, i.e $T_s = T_s(z)$. In this case, we substitute T_{m0} into the PDE to yield another PDE,

$$g_t = \alpha g_{xx} - \alpha \omega_m^2 g. \quad (\text{D.47})$$

Note that this is the heat equation again, but with an extra zeroth-order term. The solution is to multiply through by $e^{\alpha \omega_m^2 t} = e^{\beta t}$. This yields,

$$\begin{aligned} e^{\beta t} g_t + \beta e^{\beta t} g - \alpha e^{\beta t} g_{xx} &= 0, \\ \frac{\partial}{\partial t} (e^{\beta t} g) - \alpha e^{\beta t} g &= 0. \end{aligned} \quad (\text{D.48})$$

Make the change of variables $v(x, t) = e^{\beta t} g$ to arrive at

$$v_t - \alpha v_{xx} = 0, \quad (\text{D.49})$$

subject to the boundary conditions

$$v(0, t) = e^{\beta t}, \quad (\text{D.50a})$$

$$v(\infty, t) \rightarrow 0. \quad (\text{D.50b})$$

This problem has a known solution, tabulated in [113],

$$v(x, t) = \frac{1}{2} e^{\beta t} \left(e^{-x\sqrt{\beta/\alpha}} \operatorname{erfc} \left(\frac{x}{\sqrt{4\alpha t}} - \sqrt{\beta t} \right) + e^{x\sqrt{\beta/\alpha}} \operatorname{erfc} \left(\frac{x}{\sqrt{4\alpha t}} + \sqrt{\beta t} \right) \right), \quad (\text{D.51a})$$

$$g(x, t) = e^{-\beta t} v(x, t). \quad (\text{D.51b})$$

Again, the surface heat transfer can be calculated as

$$q_s(z, t) = -k T_x \Big|_{x=0}. \quad (\text{D.52})$$

The derivative can be applied individually to each term in the Fourier decomposition, leading to the following modes:

$$q_{mn} = k c_{mn} \sqrt{\omega_m^2 + \frac{j\omega_n}{\alpha}} e^{j\omega_n t} e^{j\omega_m z}, \quad (\text{D.53a})$$

$$q_{m0} = k c_{m0} \left(\sqrt{\frac{\beta}{\alpha}} \operatorname{erf}(\sqrt{\beta t}) + \frac{e^{-\beta t}}{\sqrt{\pi\alpha t}} - 1 \right) e^{j\omega_m z}, \quad (\text{D.53b})$$

$$q_{0n} = k c_{0n} \sqrt{\frac{|\omega_n|}{2\alpha}} (1 + j) e^{j\omega_n t}, \quad (\text{D.53c})$$

$$q_{00} = \frac{k c_{00}}{\sqrt{\pi\alpha t}}, \quad (\text{D.53d})$$

with the appropriate conjugate symmetry enforced to arrive at a real result.

D.4 Two-Step Method

An extremely robust method can be created by combining the Fourier and inverse methods. First, the fft method is performed using a large smoothing factor. The results of this computation are very smooth but likely inaccurate. However, if the fft results are given as the initial guess to the inverse method, the end result is both smooth and quite accurate. This method is mostly useful for when the signal-to-noise ratio is low and neither the fft method nor the direct method produce acceptable results on their own. Because the fft method is extremely fast, the two-step method does not take much more time to compute than the inverse method on its own.

E. HEAT TRANSFER CODES

E.1 Heat Transfer Calculation

qcalc

Calculate heat transfer from an sequence of temperature images

Syntax

```
Q = qcalc(Temp,fs)
Q = qcalc(Temp,fs,params)

Q = qcalc(__,'algorithm',algorithmname)
Q = qcalc(__,'material',materialstruct)
Q = qcalc(__,'material',materialname)

Q = qcalc(__,Name,Value)

[Q,compTime] = qcalc(__)
```

Description

`Q = qcalc(Temp,fs)` calculates the heat transfer inferred from the sequence of temperature images `Temp` and the known frame rate `fs`.

`Q = qcalc(Temp,fs,params)` defines the computation parameters in the `params` structure, which is an optional argument. If it is used, it must be placed in the third argument position.

`Q = qcalc(__,'algorithm',algorithmname)` uses the specified algorithm to compute the heat transfer. The available algorithms are listed below.

`Q = qcalc(__,'material',materialstruct)` uses the structure `materialstruct` to specify the material properties for the calculation. The structure must have `k`, `rho`, and `cp` fields defined.

`Q = qcalc(__,'material',materialname)` uses a named material for the computation. The default is 'peek'.

`Q = qcalc(__,Name,Value)` uses the name, value pairs to set the computation parameters, and can be set in any order.

`[Q,compTime] = qcalc(__)` also returns the computation time, in seconds.

Input Arguments

[collapse all](#)

▼ **Temp — Temperature array**
3-D Array

The temperature array must be $N \times M \times P$, where P (the number of frames) is strictly greater than 1. The units of `Temp` can be degrees Celsius or Kelvin. `Temp` may have any number of pixels from 1 to Inf ($N \times M \geq 1$).

[collapse all](#)

▼ **fs — Camera frame rate**
scalar value

The frame rate is a positive scalar specifying the number of frames per second taken by the camera. Typical values are 200-300 fps (or Hz).

[collapse all](#)

▼ **params — (Optional) structure of parameters for the computation**
struct

Instead of (or in addition to) using the Name,Value parameter syntax, you can define a parameters structure with field names of the parameters you wish to set and values equal to the desired values. For example, if you want to set Nsmooth to 11, you can either write:

```
[__] = qcalc(Temp,fs,'Nsmooth',11)
```

or

```
params.Nsmooth = 11;  
[__] = qcalc(Temp,fs,params)
```

If you use the parameters structure it **must** be the third argument.

Name-Value Pair Arguments

Specify optional comma-separated pairs of Name,Value arguments. Name is the argument name and Value is the corresponding value. Name must appear inside single quotes (' '). You can specify several name and value pair arguments in any order as Name1,Value1,...,NameN,ValueN.

[collapse all](#)

▼ **algorithm — Computational algorithm**
'fft1d' (default) | 'direct1d' | 'ihtp1d' | ...

The available algorithms are 'fft1d','fft2d','direct1d', and 'ihtp1d', which are described in the More About section, below. The direct and ihtp algorithms can be modified to run more efficiently in certain circumstances using the 'fast-' flag, e.g. 'fast-ihtp1d'. In addition, the ihtp algorithm can run the fft algorithm first to provide a suitable initial guess for the optimization routine using the '-2step' flag, e.g. 'ihtp1d-2step'. To run a computation using parallel computing, use the '-par' flag (note that often this will not increase the speed of the computation). The flags can be specified in any order, i.e. both 'fast-direct1d-par' and 'direct1d-fast-par' use the same algorithm (the fast direct 1-D algorithm in parallel mode).

[collapse all](#)

▼ **material — Material name or structure**
'peek' (default) | struct

The default material is PEEK, which should be fine for most users of qcalc. If you would like to use different material properties you can either (a) use a material structure with k, rho, and cp fields, or you can edit the parseInputs subfunction to allow for different named materials.

[collapse all](#)

▼ **units — Output units**
'Wm2' (default) | 'kWm2'

The default output units are W/m^2 . You can select kW/m^2 if you prefer.

[collapse all](#)

▼ **time — Time from run start to plot image**
1 (default) | positive scalar

If 'plot' is 'on', qcalc will plot the frame at time 'time', in seconds.

[collapse all](#)

▼ **plot — Whether to plot the result or not**
'off' (default) | 'on'

Plotting is off by default. If qcalc is called with no output arguments or if 'plot' is turned 'on', then qcalc will plot the frame at the desired time.

[collapse all](#)

▼ **tstart — Length of pre-run data**
-1 (default) | negative scalar

tstart indicates how much pre-run data was taken. It **must** be a negative scalar, i.e. there must be some non-zero amount of pre-run data. The default is -1, i.e. 1 second of pre-run data.

[collapse all](#)

▼ **Nsmooth — Number of frames for moving average filter**
3 (default) | odd positive integer

A moving average filter is applied to the time history of each pixel for the 1-D methods. The number of samples used in the filter is set by Nsmooth. As per MATLAB's smooth function, if Nsmooth is not odd, it will be reduced by one. Nsmooth must be at least 3 (the default).

[collapse all](#)

▼ **fsz — Spanwise sampling rate**
empty (default) | positive scalar | 1-D vector

This is only required for the 2-D algorithms, indicating the spanwise sampling rate in points per meter (of the computational mesh, which is not necessarily the same as the the number of pixels per meter in the original image). If this is not set and the 'fft2d' algorithm is requested, qcalc will throw an error.

[collapse all](#)

▼ **spanindex — Index of the spanwise direction**
2 (default) | 1

For the 2-D methods qcalc needs to know which direction is spanwise. By default this is dimension 2 (columns), but if it is rows you can set it to 1. qcalc transposes the images in this case.

[collapse all](#)

▼ **L — Length of internal mesh**
0.005 (default) | positive scalar

The direct and ihtp algorithms use a finite difference scheme on a 1-D internal mesh of length L, in meters. This does not have to be the true thickness of the model, but it should be at least as thick as the thermal penetration depth (around 3 - 5 mm).

[collapse all](#)

▼ **n — Number of internal mesh points**
25 (default) | positive integer

The number of internal mesh points for the finite difference calculator is set by `n`. For most purposes the value of `n` does not need to be very large (25 is the default), but it may be worthwhile to do a grid convergence study for your specific application. If `L` is small and `n` is large, you may run into numerical precision issues.

[collapse all](#)

▼ **bc — Back-face boundary condition**
'isothermal' (default) | 'adiabatic'

The back face is the internal face of the model (so not the model surface). The boundary condition can either be isothermal, meaning it is the same temperature as the initial temperature for all time, or adiabatic, meaning the temperature slope at the back face is zero. Isothermal is usually an appropriate choice. If the model is very thin and if the internal surface of the model is bounded by stagnant air, setting `bc` to 'adiabatic' may be a better choice. Use of symmetries in the model can help, i.e. a thin flat surface with the same flow on both sides can be modelled as adiabatic with the back face at the centerplane of the two surfaces.

[collapse all](#)

▼ **theta — Crank-Nicolson parameter**
1 (default) | scalar in [0, 1]

The parameter `theta` controls the degree to which the finite difference method is implicit or explicit. 1 (the default) means the method is fully implicit in time (BTCS), whereas 0 is fully explicit (FTCS). A `theta` of 1/2 is also known as the Crank-Nicolson scheme.

[collapse all](#)

▼ **phi0 — Initial temperature profile**
0 (default) | scalar | 1-D vector

The 'direct1d' algorithm can have a non-zero initial condition (where zero here is the temperature difference from the initial surface temperature). It is not recommended to use this parameter unless you have a good model for the internal temperature profile from some other source.

[collapse all](#)

▼ **Rsmooth — Spatial smoothing parameter**
0.5 (default) | positive scalar

The finite difference and 2-D algorithms are spatially smoothed at each frame using a Gaussian kernel with a standard deviation of `Rsmooth` pixels.

[collapse all](#)

▼ **startframe — Starting frame for optimization**
`length(Temp) - 30` (default) | positive integer

The inverse methods are too computationally expensive to run over the entire data record, so the direct method is run first up to `startframe`, at which point the optimization begins. To use the inverse method for the entire measurement range (not recommended), set `startframe` to 1.

[collapse all](#)

▼ **qinit — Initial guess for Q**
2000 (default) | scalar | 2-D matrix

The inverse method is an optimization process, and so needs an initial guess for the solution. You can use the default (a constant 2000 W/m²), specify your own Qinit(t), or use the 'fft1d' method to calculate an initial guess by specifying the '-2step' flag in the algorithm statement.

[collapse all](#)

- ✓ **filter — Turns spatial filtering on or off**
'on' (default) | 'off'

It is usually a good idea to have a small amount of spatial filtering, especially for the finite difference methods.

Output Arguments

[collapse all](#)

- ✓ **Q — Heat transfer**
3-D Array, same size as Temp

The heat transfer output Q is the same size as the input, with units specified by the 'units' parameter (default W/m²)

[collapse all](#)

- ✓ **compTime — Computation time**
scalar value

The heat-transfer computation time in seconds.

More About qca1c

Algorithm	Method of Operation	Speed	When to Use
fft1d	The 1-D fft algorithm decomposes the temperature history for each mesh point into its Fourier modes using the Fast Fourier Transform, then computes the sum of the responses to each individual frequency component (using semi-infinite theory).	Fast	Because the fft1d algorithm is so fast, it is a good first choice. It may not be appropriate for situations where the material thickness is small compared to the thermal penetration depth or when the camera frame rate is low.
fft2d	The 2-D fft algorithm operates on the same principle as the 1-D algorithm, but it allows for conduction in the spanwise direction (as opposed to purely normal to the surface). It makes all of the same assumptions as fft1d, but in addition it assumes that the thermal penetration depth is much smaller than the local radius of curvature.		If you are concerned about tangential conduction (e.g., in streaky flows), then it may be worth trying the fft2d algorithm. This method is still quite fast, though it has more restrictive assumptions. It may yield inaccurate results in areas of high surface curvature.

Algorithm	Method of Operation	Speed	When to Use
direct1d	The direct method solves the heat equation on a 1-D mesh using a finite difference Crank-Nicolson method with variable parameter θ which controls the degree of implicitness of the solver. When $\theta = 1$, the method is fully implicit in time.	Medium-Slow	The direct method can be used whenever the 1-D assumption is approximately valid (i.e., whenever there is minimal tangential conduction). It has the least restrictive set of assumptions, and many of the solution parameters can be finely controlled. However, it is not as fast as the fft-based solvers, nor is it as robust in the presence of noise as the ihtp solvers.
fast-direct1d	The fast version of the 1-D algorithm leverages the fact that only the three points closest to the surface are necessary to compute the heat transfer to significantly reduce the required memory for computation. It works by expanding the surface temperature (and the next two mesh points) in powers of the transition matrix T to compute every point in time in one matrix operation for each mesh point (pixel).	Medium-Fast to Slow	When the number of 1-D internal mesh points is comparable to or larger than the number of time points, the fast-direct1d method can improve performance over the standard direct1d method. However, for long times the algorithm can be numerically unstable and could in fact be slower. Unlike the direct1d method, the fast version cannot start from a non-zero initial temperature profile on the internal mesh points.
ihtp1d	The <i>inverse heat transfer problem</i> or <i>ihtp</i> tries to find the heat transfer Q which minimizes the error between the measured temperature and the temperature calculated using Q as a boundary condition. It makes the same assumptions as the direct1d methods, but it is less sensitive to noise. Use of any of the ihtp algorithms requires a suitable initial guess for the heat transfer at each point in space and time (see the 2step-ihtp1d algorithm, below).	Slow	The ihtp1d method is quite robust, but it is very slow. It is not a good first choice unless there is some reason to believe the SNR is very low or the other methods are providing weird results.

Algorithm	Method of Operation	Speed	When to Use
fast-ihtp1d	The fast version of the ihtp1d algorithm makes the same changes as the fast-direct1d makes to direct1d. However, because the optimization routine must calculate the temperature hundreds of times to find the correct Q, the performance increase is substantial.	Medium-Slow	If you want to use an inverse method, the fast version is a good starting point. Keep in mind that it has the same stability issues as the fast-direct1d algorithm, meaning it could become numerically unstable if performed on very long datasets. However, usually the optimization is only performed over a small portion of the run so these issues should never be apparent.
(fast)-ihtp1d-2step	The 2step version of the inverse codes uses the fft1d algorithm to compute an initial guess for Q, for use in the optimization algorithm. Because the run time of the fft1d code is very small compared to the run time of the ihtp1d methods, this is usually a good option.	Medium-Slow to Slow	Using the 2step flag is a good place to start with the inverse methods, unless you already have a good estimate of the heat transfer from some other source.

See Also:

[qfft1d](#), [qfft2d](#), [qdirect1d](#), [qihtp1d](#)

```

1 function varargout = qcalc(Temp,fs,varargin)
2 % Calculate heat transfer from an sequence of temperature images
3
4 narginchk(2,40);
5 nargoutchk(0,2);
6
7 if isstruct(varargin{1})
8     paramStruct = varargin{1};
9     fields = fieldnames(paramStruct);
10
11     inargs = varargin(2:end);
12     for ff = 1:length(fields)
13         inargs = [inargs, {fields{ff}}, {paramStruct.(fields{ff})}];
14     end
15 else
16     inargs = varargin;
17 end
18
19 opts = parseInputs(Temp,fs,inargs{:});
20
21 if ~strcmp(opts.algorithm,'bench')
22     algopts = parseAlg(opts.algorithm,{'fft1d','fft2d','direct1d','ihtp1d','
23         bench'});
24 else
25     algopts.alg = 'bench';
26
27 switch algopts.alg
28     case {'fft1d','fft2d','direct1d','ihtp1d'}
29
30         algstr = ['q',algopts.alg,'_fast'*algopts.fast,'_par'*algopts.par];
31         algstr = replace(algstr,char(0),'');
32
33         if ~algopts.precalc
34             [Q,ct] = feval(algstr,Temp,fs,opts.parameters);
35         else

```

```

36     [Q0,ct1] = qfft1d(Temp,fs,opts.parameters);
37     lenihtp = length(opts.parameters.startframe:size(Q0,3));
38     opts.parameters.Qinit = reshape(Q0(:,:,opts.parameters.startframe:
39         end),...
40         size(Q0,1)*size(Q0,2),lenihtp);
41     [Q,ct2] = feval(algstr,Temp,fs,opts.parameters);
42
43     ct = ct1 + ct2;
44
45     end
46     case 'bench'
47         results = benchmark();
48         disp(results)
49         varargout{1} = results;
50         return
51     otherwise
52
53         error('Unknown algorithm selected.');
```

```

54
55     end
56
57     if strcmp(opts.parameters.units,'kWm2')
58         Q = Q/1000;
59     end
60
61     if nargin>0
62         varargout = cell(1,nargin);
63
64         if nargin == 1
65             varargout{1} = Q;
66         else
67             varargout{1} = Q;
68             varargout{2} = ct;
69         end
70     end
71
72     if nargin==0 || strcmp(opts.parameters.plot,'on')
```

```

71     frame = fs*(abs(opts.parameters.tstart)+opts.parameters.time);
72
73     figure
74
75     if opts.parameters.spanindex==2
76         imagesc(Q(:,:,frame));
77         ylabel('Spanwise Reference, px');
78         xlabel('Streamwise Reference, px');
79     else
80         imagesc(flipud(Q(:,:,frame)'));
81         xlabel('Spanwise Reference, px');
82         ylabel('Streamwise Reference, px');
83     end
84
85     ax = gca;
86     cb = colorbar;
87
88     switch opts.parameters.units
89         case 'Wm2'
90             unitstr = 'W/m^2';
91         case 'kWm2'
92             unitstr = 'kW/m^2';
93     end
94
95     ylabel(cb,['Heat Transfer, ', unitstr],'fontsize',14);
96
97     ax.FontSize = 14;
98     ax.LineWidth = 1;
99     ax.YDir = 'normal';
100
101 end
102
103 end
104
105 function opts = parseInputs(Temp,fs,varargin)
106 % Parses the input arguments to the qcalc function

```

```

107
108 % expectedAlgs = {'fft1d','fft2d','direct1d','ihtp1d',...
109 %     'fast-ihtp1d','fast-ihtp1d-2step','ihtp1d-2step','fast-direct1d','bench
    '};
110 defaultPlot = 'off';
111 expectedPlot = {'on','off'};
112 defaultBC = 'isothermal';
113 expectedBC = {'isothermal','adiabatic'};
114 defaultMatl = 'peek';
115 expectedMatl = {'peek'};
116
117 validScalarPosNum = @(x) isnumeric(x) && isscalar(x) && (x>0);
118 isint = @(x)(round(x)==x && isscalar(x));
119
120 lent = size(Temp,3);
121
122 % input parser for argument handling
123 % first do the required an optional arguments
124 p = inputParser;
125 p.FunctionName = 'qcalc';
126 p.StructExpand = false;
127 p.KeepUnmatched = true;
128 addRequired(p,'Temp',...
129     @(x)(isnumeric(x) && size(x,3)>1 && isreal(x) && ~any(isnan(x(:)))));
130 addRequired(p,'fs',@(x)( isnumeric(x) && isscalar(x) && x>0 ));
131 addParameter(p,'material',defaultMatl,@(x)( isstruct(x) ||...
132     (any(validatestring(x,expectedMatl))) ));
133 addParameter(p,'algorithm','fft1d');
134
135 % now the common parameters
136 addParameter(p,'units', 'Wm2',...
137     @(x) any(validatestring(x,{'Wm2','kWm2'})));
138 addParameter(p,'time',1,...
139     @(x)( isnumeric(x) && isscalar(x) && (x>0) && (fs*x <= size(Temp,3))));
140 addParameter(p,'plot',defaultPlot,...
141     @(x) any(validatestring(x,expectedPlot)));

```

```

142 addParameter(p, 'pbar', []);
143 addParameter(p, 'tstart', -1, @(x)(isnumeric(x) && isscalar(x) && (x<0)));
144
145 % now the algorithm specific parameters
146 addParameter(p, 'Nsmooth', 3, @(x)( isnumeric(x) && isscalar(x) && (round(x)==x)
    && (x>=3)));
147 addParameter(p, 'fsz', [], @(x) (isnumeric(x) && (isscalar(x) || isvector(x))));
148 addParameter(p, 'spanindex', 2, @(x)((x==1) || (x==2)));
149 addParameter(p, 'L', 0.005, validScalarPosNum);
150 addParameter(p, 'n', 25, @(x)(validScalarPosNum(x) && (round(x)==x)));
151 addParameter(p, 'bc', defaultBC, ...
152     @(x) any(validatestring(x, expectedBC)));
153 addParameter(p, 'theta', 1, @(x)(isscalar(x) && (x>=0) && (x<=1)));
154 addParameter(p, 'phi0', 0, @isnumeric);
155 addParameter(p, 'Rsmooth', 0.5, validScalarPosNum);
156 addParameter(p, 'startframe', (lent>30)*(lent-30) + (lent<=30)*(lent), @(x)(isint
    (x) && (x>0) && (x<=lent)));
157 addParameter(p, 'Qinit', 2000, @(x)(isscalar(x) || ismatrix(x)));
158 addParameter(p, 'filter', 'on', ...
159     @(x) any(validatestring(x, {'on', 'off'})));
160
161
162 % parse it and do some additional checks
163 parse(p, Temp, fs, varargin{:});
164
165 matl = p.Results.material;
166 alg = p.Results.algorithm;
167
168 if ischar(matl)
169     switch matl
170         case 'peek'
171             material.rho = 1300;
172             material.cp = 1026;
173             material.k = 0.29;
174         otherwise
175             error('It should not be possible for this error to be thrown...');

```

```
176     end
177 else
178     if ~all(isfield(p.Results.material,{'rho','cp','k'}))
179         error('Input material must have rho, cp, and k fields defined');
180     end
181
182     validMtlProp = @(x)(isnumeric(x) && isscalar(x) && (x>0));
183
184     if ~all(arrayfun(validMtlProp,[p.Results.material.rho,p.Results.material.
185         cp,p.Results.material.k]))
186         error('Input material properties are invalid. They must be positive,
187             numeric scalars.');
```

```
186     end
187
188     material = p.Results.material;
189 end
190
191 opts.parameters.material = material;
192 opts.algorithm = alg;
193
194 opts.parameters.units = p.Results.units;
195 opts.parameters.time = p.Results.time;
196 opts.parameters.plot = p.Results.plot;
197 opts.parameters.pbar = p.Results.pbar;
198 opts.parameters.tstart = p.Results.tstart;
199 opts.parameters.Nsmooth = p.Results.Nsmooth;
200 opts.parameters.fsz = p.Results.fsz;
201 opts.parameters.L = p.Results.L;
202 opts.parameters.n = p.Results.n;
203 opts.parameters.bc = p.Results.bc;
204 opts.parameters.theta = p.Results.theta;
205 opts.parameters.phi0 = p.Results.phi0;
206 opts.parameters.Rsmooth = p.Results.Rsmooth;
207 opts.parameters.startframe = p.Results.startframe;
208 opts.parameters.Qinit = p.Results.Qinit;
209 opts.parameters.spanindex = p.Results.spanindex;
```

```
210 opts.parameters.filter = p.Results.filter;
211
212
213 end
214
215 %% sequential codes
216 function [Q,ct] = qfft1d(Temp,fs,opts)
217 % Calculates the heat transfer using the fft-based approach.
218 %
219 % SYNTAX:
220 %1 [Q,ct] = qfft1d(Temp,fs,opts)
221 % DESCRIPTION:
222 % [Q,ct] = qfft1d(Temp,fs,opts) calculates the heat transfer Q from the
223 % images in 3-D array Temp, sampled at frame rate fs, using the parameters
224 % specified in the structure opts. It is only intended to be used through the
225 % <code>qcalc</code></a> function.
226 %
227 % INPUTS:
228 % Temp:      Temperature array {3-D array}
229 %            The temperature array Temp has size N x M x P where each image
230 %            is M x N, and there are P images. P must be greater than 1. The
231 %            temperature can be in degrees Celsius or Kelvin.
232 % fs:        Camera frame rate {scalar value}
233 %            The camera frame rate is specified in frames per second \(Hz\).
234 % opts.Nsmooth: Smoothing factor {scalar integer}
235 %            The Nsmooth field of the <code>opts</code> structure controls
236 %            the moving average filtering of the data pre- and post-
237 %            computation. Nsmooth represents the number of samples using the
238 %            moving average, and so should be odd \(the code will reduce its
239 %            value by 1 if Nsmooth is even\).
240 % opts.material: Material properties {structure}
241 %            This structure sets the material properties. It must have
242 %            <code>k, rho,</code> and <code>cp</code> fields.
243 % opts.pbar:    uiprogressdlg handle {graphics handle}
244 %            Handle to uiprogressdlg component for user feedback.
245 %
```

```

246 % OUTPUTS:
247 % Q:          Heat transfer array {3-D array}
248 %           The heat transfer is a 3-D array the same size as the input
249 %           <code>Temp</code> array. The output units are W/m<sup>2</sup>.
250 % ct:         Computation time {scalar value}
251 %           The computation time only includes the looping over every
252 %           pixel, not any of the setup.
253 %
254 % MORE ABOUT:
255 % <p>
256 % The fft algorithm works by decomposing the temperature signal from each
257 % pixel into sinusoidal components, and summing the response to each
258 % individual component.
259 % </p>
260 % <p>
261 % The fft algorithm extends the input signal using a Bezier curve so that
262 % it is approximately C1, which should reduce Gibb's phenomena (ringing).
263 % The length of the extension is determined to make the signal length a
264 % power of 2. The padding is removed from the returned heat transfer.
265 % </p>
266 %
267 % SEE ALSO:
268 % qcalc, qfft2d, qdirect1d, qihtp1d, makePeriodic
269 %#
270
271 fftw('dwisdom',[]);
272 fftw('planner','estimate');
273
274 pbar = opts.pbar;
275 Nsmooth = opts.Nsmooth;
276
277 % has to be an odd smoothing span (Matlab's smooth does this automatically)
278 if ~mod(Nsmooth,2)
279     Nsmooth = Nsmooth - 1;
280 end
281

```

```

282 smoothFilter = ones(Nsmooth,1)/Nsmooth;
283
284 if ~isempty(pbar)
285     pbar.Message = 'Collecting variables...';
286 end
287
288 % set up constants
289 dt = 1/fs;           % camera frame rate
290 alpha = opts.material.k/(opts.material.rho*opts.material.cp); % thermal
    diffusivity
291 k_cond = opts.material.k;
292 % form Delta T matrix
293 [xlen,thlen,tlen] = size(Temp);
294
295 Temp = reshape(Temp,xlen*thlen,tlen); % for parallelization we need a column
    vector
296 T0 = Temp(:,1);
297 Temp = Temp-T0;           % use Delta T (from run start)
298
299
300 % Run computation
301 if ~isempty(pbar)
302     pbar.Message = 'Calculating...';
303 end
304
305 % fft is most efficient for powers of 2 (sometimes)
306 M = length(Temp(1,:));
307 N = 2^(nextpow2(M));
308 Next = N-M;
309
310 SIZE = xlen*thlen;
311 Q = zeros(SIZE,M);
312
313 t = 0:dt:(N-1)*dt;
314 %t = t + opts.tstart;
315

```

```

316 % form the offset
317 stepIn = real(1./sqrt(pi*alpha*t'));
318 stepIn(1:round(-opts.tstart*fs)) = stepIn(round(-opts.tstart*fs)+1);
319
320 % form the frequency vector (we know the size and we know N is even)
321 kk = -N/2:N/2-1;
322 kk = ifftshift(kk);
323 w=2*pi*fs*kk/N;
324
325 % define the system frequency response coefficients
326 an = (sqrt(abs(w)/(2*alpha)))*(1j+1);
327 an = transpose(an);
328 an(w<0) = conj(an(w<0)); % enforce symmetry for real signal
329
330 tic
331
332 for jj = 1:SIZE
333
334
335     % Smooth the temperature
336     % Note this is inlined from Matlab's smooth/moving function,
337     % which has a TON of overhead. Inlining gives a 3x speedup!
338     DeltaT = Temp(jj,:);
339     c = filter(smoothFilter,1,DeltaT');
340     cbegin = cumsum(DeltaT(1:Nsmooth-2)');
341     cbegin = cbegin(1:2:end)./(1:2:(Nsmooth-2))';
342     cend = cumsum(DeltaT(M:-1:M-Nsmooth+3)');
343     cend = cend(end:-2:1)./(Nsmooth-2:-2:1)';
344     DeltaT = [cbegin;c(Nsmooth:end);cend];
345
346     % find the Fourier coefficients
347     %cn=fft([DeltaT; repmat(DeltaT(end),Next,1)],N)/N; % make up for the lack
           of 1/N in Matlab FFT
348     cn = fft(makePeriodic(DeltaT,Next,30),N)/N;
349
350     % calculate coefficients for q_s Fourier series

```

```

351     en = cn.*an;
352     en(1) = 0;
353
354     % undo FFT and add constant offset term
355     qs_Fo = N*real(ifft(en));
356     qs_Fo = k_cond*qs_Fo + k_cond*cn(1).*stepIn;
357
358     % Smooth the result
359 %     d = filter(smoothFilter,1,qs_Fo(1:M));
360 %     dbegin = cumsum(qs_Fo(1:Nsmooth-2));
361 %     dbegin = dbegin(1:2:end)./(1:2:(Nsmooth-2))';
362 %     dend = cumsum(qs_Fo(M:-1:M-Nsmooth+3));
363 %     dend = dend(end:-2:1)./(Nsmooth-2:-2:1)';
364 %     Q(jj,:) = [dbegin;d(Nsmooth:end);dend]';
365     Q(jj,:) = qs_Fo(1:M)';
366
367     if ~isempty(pbar)
368         pbar.Value = jj/SIZE;
369     end
370
371 end
372
373
374 ct = toc;
375
376 if ~isempty(pbar)
377     pbar.Message = 'Finishing up...';
378 end
379
380 % reshape Qs_Fo vector to a 3D array
381 Q = reshape(Q,xlen,thlen,M);
382
383
384 end
385
386 function [Q,ct] = qfft2d(Temp,fs,opts)

```

```
387 % Calculates the heat transfer using the fft-based approach.
388 %
389 % SYNTAX:
390 %1 [Q,ct] = qfft2d(Temp,fs,opts)
391 % DESCRIPTION:
392 % [Q,ct] = qfft2d(Temp,fs,opts) calculates the heat transfer Q from the
393 % images in 3-D array Temp, sampled at frame rate fs, using the parameters
394 % specified in the structure opts. The qfft2d algorithm allows
395 % spanwise heat transfer as well as wall-normal heat transfer. It is only
396 % intended to be used through the
397 % qcalc function.
398 %
399 % INPUTS:
400 % Temp:      Temperature array {3-D array}
401 %            The temperature array Temp has size N x M x P where each image
402 %            is M x N, and there are P images. P must be greater than 1. The
403 %            temperature can be in degrees Celsius or Kelvin.
404 % fs:        Camera frame rate {scalar value}
405 %            The camera frame rate is specified in frames per second (Hz).
406 % opts.fsz:  Spanwise sampling rate {scalar value}
407 %            The spanwise sampling rate for the data in samples per meter.
408 %            If this is not provided, an error will be thrown. You should
409 %            only call qfft2d from the qcalc
410 %            wrapper function to properly validate your inputs.
411 % opts.spanindex:  Index of spanwise direction {1 | 2}
412 %            The index of the spanwise direction in the images. If
413 %            spanindex == 1 the image arrays are transposed.
414 % opts.material:  Material properties {structure}
415 %            This structure sets the material properties. It must have
416 %            k, rho, and cp fields.
417 % opts.pbar:     uiprogressdlg handle {graphics handle}
418 %            Handle to uiprogressdlg component for user feedback
419 %
420 % OUTPUTS:
421 % Q:           Heat transfer array {3-D array}
422 %            The heat transfer is a 3-D array the same size as the input
```

```
423 %           <code>Temp</code> array. The output units are W/m<sup>2</sup>.
424 % ct:       Computation time {scalar value}
425 %           The computation time only includes the looping over every
426 %           pixel, not any of the setup.
427 %
428 % MORE ABOUT:
429 % The 2D fft algorithm decomposes the temperature signal into spanwise
430 % slices with different frequencies in time and space and sums the
431 % responses to these components.
432 %
433 % SEE ALSO:
434 % qfft1d, qdirect1d, qihtp1d
435 %#
436
437 pbar = opts.pbar;
438
439 if opts.spanindex == 1
440     Temp = permute(Temp,[2,1,3]);
441 end
442
443 if isscalar(opts.fsz)
444     fsz = opts.fsz*ones(size(Temp,1));
445 else
446     fsz = opts.fsz;
447 end
448
449 if ~isempty(pbar)
450     pbar.Message = 'Collecting variables...';
451 end
452
453 % set up constants
454 dt = 1/fsz;           % camera frame rate
455 alpha = opts.material.k/(opts.material.rho*opts.material.cp); % thermal
    diffusivity
456 k_cond = opts.material.k;
457
```

```
458 % form Delta T matrix
459
460 % permute temp so x index is last (more efficient looping)
461 Temp = permute(Temp,[2,3,1]);
462 [thlen,tlen,xlen] = size(Temp);
463
464 filtSize = opts.Rsmooth;
465 for kk = 1:xlen
466     Temp(:,:,kk) = imageFilt(Temp(:,:,kk),filtSize);
467 end
468
469 tlen0 = tlen;
470 thlen0 = thlen;
471
472 Q = zeros(thlen,tlen,xlen);
473
474 t = 0:dt:(tlen-1)*dt;
475
476 Temp = Temp - Temp(:,1,:);
477
478
479 N = tlen;
480
481 if mod(N,2)==0
482     kk=-N/2:N/2-1; % N even
483 else
484     kk=-(N-1)/2:(N-1)/2; % N odd
485 end
486
487 wn = 2*pi*fs*kk/N;
488
489 M=thlen;
490
491 if mod(M,2)==0
492     pp=-M/2:M/2-1; % N even
493 else
```

```

494     pp=-(M-1)/2:(M-1)/2; % N odd
495 end
496
497 [WN,PP] = meshgrid(wn,2*pi*pp/M);
498
499 WN = ifftshift(WN);
500 PP = ifftshift(PP);
501
502 %% start computation
503 if ~isempty(pbar)
504     pbar.Message = 'Calculating...';
505 end
506 tic
507
508 stepIn = k_cond./sqrt(pi*alpha*t);
509 stepIn(1:fs*abs(opts.tstart)) = stepIn(abs(opts.tstart)*fs+1);
510
511
512 for jj = 1:xlen
513
514     fszx = fsz(jj);
515     T_slice = Temp(:,:,jj);           % T_slice is the time history
516     % of all the pixels in a
517     % spanwise row @ const. x, it
518     % is size M x N
519
520 %     T_slice = imageFilt(T_slice,opts.Rsmooth);           % filter temperature
map
521
522     %% do Fourier decomposition
523
524     z = (0:1/fszx:(thlen-1)/fszx)';
525
526     WM = fszx*PP;
527

```

```

528   Cmn = fft2(T_slice)/(M*N); % remember to include M*N to get Fourier
      coefficients
529
530   % Case 1: m ~= 0 & n ~= 0
531   Amn = k_cond*sqrt(WM.^2 + 1j*WN/alpha);
532   Emn = Cmn.*Amn;
533   Emn(WN == 0 | WM == 0) = 0;
534
535   Qs_2d = real(ifft2(Emn))*M*N;
536
537   % Case 2: n == 0, m ~= 0 (i.e. no oscillations in time, but changes in
538   % z allowed)
539   Wm0 = WM(:,1);
540   B = alpha*Wm0.^2;
541   Cm0 = Cmn(:,1);
542   Cm0(1) = 0;
543
544   T = repmat(t,length(B),1);
545   BB = repmat(B,1,length(t));
546   Cm0_mat = repmat(Cm0,1,length(t));
547
548   exp_kernel = exp(1j*z*Wm0');
549   Am0_matrix = k_cond*Cm0_mat.*(sqrt(BB/alpha).*erf(sqrt(B*t)) + exp(-B*t)./
      sqrt(pi*alpha*T) - 1);
550
551   Qs_2d = Qs_2d + 2*real(exp_kernel*Am0_matrix);
552
553   % Case 3: m == 0, n ~= 0 (i.e. const. value spanwise, but oscillating
554   % in time)
555   W0n = WN(1,:);
556
557   C0n = Cmn(1,:);
558   C0n(1) = 0;
559
560   A0n = (1j+1)*sqrt(abs(W0n)/(2*alpha));
561   A0n(W0n<0) = conj(A0n(W0n<0));

```

```

562     E0n = C0n.*A0n;
563     Q0n_vec = k_cond*N*real(iff(E0n));
564
565     Qs_2d = Qs_2d + repmat(Q0n_vec,thlen,1);
566
567     % Case 4: m == n == 0, i.e. step function in space and time
568     C00 = Cmn(1,1);
569     Qs_2d = Qs_2d + repmat(stepIn*C00,thlen,1);
570
571     Q(:, :, jj) = Qs_2d(1:thlen0,1:tlen0);
572
573     if ~isempty(pbar)
574         pbar.Value = jj/xlen;
575     end
576
577 end
578
579 if ~isempty(pbar)
580     pbar.Message = 'Finishing up...';
581 end
582
583 Q = permute(Q,[3,1,2]); % permute back to the start order
584
585 if opts.spanindex == 1
586     Q = permute(Q,[2,1,3]);
587 end
588
589 ct = toc;
590 end
591
592 function [Q,ct] = qdirect1d(Temp,fs,opts)
593
594 pbar = opts.pbar;
595
596 tic;
597

```

```

598 if ~isempty(pbar)
599     pbar.Message = 'Collecting variables....';
600 end
601
602 phi0 = opts.phi0.*ones(opts.n,1);
603 L = opts.L;
604 n = opts.n;
605 bc = opts.bc;
606 theta = opts.theta;
607
608 % define space and time constants
609 dt = 1/fs;
610 x = linspace(0,L,n);
611 dx = x(2)-x(1);
612
613 % define ode constants
614 alpha = opts.material.k/(opts.material.rho*opts.material.cp); % thermal
        diffusivity
615 k_cond = opts.material.k;
616 beta = alpha*dt/(dx*dx);
617
618 % reshape temperature for parfor compatibility
619 [xlen,thlen,tlen] = size(Temp);
620
621 % filter if desired
622 if strcmp(opts.filter,'on')
623     if ~isempty(pbar)
624         pbar.Message = 'Filtering temperature images....';
625     end
626     filtSize = opts.Rsmooth;
627     for kk = 1:size(Temp,3)
628         Temp(:,:,kk) = imageFilt(Temp(:,:,kk),filtSize);
629     end
630 end
631
632 Temp = reshape(Temp,xlen*thlen,tlen);

```

```

633 T0 = Temp(:,1);
634 Temp = Temp-T0;
635
636 % preallocate array
637 Q = zeros(size(Temp));
638 [Npts,~] = size(Temp);
639
640 % generate derivative matrix
641 D2 = gallery('tridiag',n,1,-2,1); % 2nd derivative matrix
642 D2(1,:) = 0;
643 D2(n,:) = 0; % enforcing boundary conditions
644
645 % generate ode matrices
646 I = eye(n);
647 A = I - beta*theta*D2;
648 B = I + beta*(1-theta)*D2;
649
650 B(1,:) = 0; % more boundary condition handling
651 B(n,:) = 0;
652
653 pvec = zeros(n,1);
654
655 switch bc
656     case 'isothermal'
657         bc2 = pvec;
658         bc2(n) = phi0(end);
659         Ap2 = A\bc2;
660     case 'adiabatic'
661         A(n,:) = 0;
662         A(n,n) = -3;
663         A(n,n-1) = 4;
664         A(n,n-2) = -1;
665
666         Ap2 = zeros(n,1);
667 end
668

```

```

669 pvec(1) = 1;
670 Ap = A\pvec;
671
672 % "transition" matrix
673 Tmat = A\B;
674
675 Q = Q';
676
677 if ~isempty(pbar)
678     pbar.Message = 'Calculating....';
679 end
680 for pt = 1:Npts
681
682     % loop through every point
683     Tpt = Temp(pt,:);
684
685     % calculate entire time history for each point
686     phi = phi0;
687
688     for tt = 1:tlen
689         phi = Tmat*phi + Ap*Tpt(tt) + Ap2;
690         % calculate q at wall (2nd order finite difference)
691         Q(tt,pt) = -k_cond*(-3*phi(1) + 4*phi(2) - phi(3))/(2*dx);
692
693     end
694
695     if ~isempty(pbar)
696         pbar.Value = pt/Npts;
697     end
698 end
699
700 Q = reshape(Q',xlen,thlen,tlen);
701 ct = toc;
702 end
703
704 function [Q,ct] = qihtp1d(Temp,fs,opts)

```

```

705 pbar = opts.pbar;
706 startFrame = opts.startframe;
707 QQ0 = opts.Qinit.*ones(size(Temp,1)*size(Temp,2),length(startFrame:size(Temp
      ,3)));
708 n = opts.n;
709 L = opts.L;
710 bc = opts.bc;
711 theta = opts.theta;
712
713 tsmooth = opts.Nsmooth;
714 if ~mod(tsmooth,2)
715     tsmooth = round(tsmooth-1);
716 end
717 smoothFilter = ones(tsmooth,1)/tsmooth;
718
719 if ~isempty(pbar)
720     pbar.Message = 'Collecting variables....';
721 end
722
723 tic;
724
725 dt = 1/fs;
726 x = linspace(0,L,n);
727 dx = x(2)-x(1);
728
729 alpha = opts.material.k/(opts.material.rho*opts.material.cp);
730 k_cond = opts.material.k;
731 beta = alpha*dt/(dx*dx);
732
733 [xlen,thlen,tlen] = size(Temp);
734
735 % filter if desired
736 if strcmp(opts.filter,'on')
737     if ~isempty(pbar)
738         pbar.Message = 'Filtering temperature images....';
739     end

```

```

740     filtSize = opts.Rsmooth;
741     for kk = 1:size(Temp,3)
742         Temp(:,:,kk) = imageFilt(Temp(:,:,kk),filtSize);
743     end
744 end
745
746 Temp = reshape(Temp,xlen*thlen,tlen);
747 T0 = Temp(:,1);
748 Temp = Temp-T0;
749
750 Q = zeros(size(Temp));
751 Q_nonlin = zeros(size(Temp,1),length(startFrame:size(Temp,2)));
752 [Npts,~] = size(Temp);
753
754 D2 = gallery('tridiag',n,1,-2,1);    % 2nd derivative matrix
755 D2(1,:) = 0;
756 D2(n,:) = 0;                        % enforcing boundary conditions
757
758 I = eye(n);
759 A = I - beta*theta*D2;
760 B = I + beta*(1-theta)*D2;
761
762 B(1,:) = 0;
763 B(n,:) = 0;
764
765 switch bc
766     case 'isothermal'
767
768     case 'adiabatic'
769         A(n,:) = 0;
770         A(n,n) = -3;
771         A(n,n-1) = 4;
772         A(n,n-2) = -1;
773 end
774
775 Tmat = A\B;

```

```

776
777 pvec = zeros(n,1);
778 pvec(1) = 1;
779 Ap = A\pvec;
780
781 options = optimoptions(@fmincon,'Display','off','Algorithm','active-set','TolX
      ',1e-12,'TolFun',1e-12);
782 lb=-100000*ones(length(startFrame:tlen),1);      % lower boundary for the
      minimization process
783 ub=100000*ones(length(startFrame:tlen),1);      % upper boundary for the
      minimization process
784
785 Aiht = A;
786 Aiht(1,1) = -3*k_cond/(2*dx);
787 Aiht(1,2) = 2*k_cond/dx;
788 Aiht(1,3) = -k_cond/(2*dx);
789
790 Apiht = Aiht\pvec;
791 Tiht = Aiht\B;
792
793 Tsurf = Temp(:,startFrame:end);
794 M = size(Tsurf,2);
795
796 Q = Q';
797
798 if ~isempty(pbar)
799     pbar.Message = 'Beginning iterations...';
800 end
801 for pt = 1:Npts
802
803     % Note: Mirko's code divides the temperature into two pieces
804     % the first piece uses a direct solver, the second piece
805     % is shorter and uses the inverse method.
806     Tpt = Temp(pt,:);
807
808     phi = zeros(n,1);

```

```

809     for tt = 1:startFrame-1
810         phi = Tmat*phi + Ap*Tpt(tt);
811         Q(tt,pt) = -k_cond*(-3*phi(1) + 4*phi(2) - phi(3))/(2*dx);
812     end
813
814     qq0 = QQ0(pt,:)';
815
816     % inlining Matlab's smooth for speed
817     DeltaT = Tsurf(pt,:);
818     c = filter(smoothFilter,1,DeltaT');
819     cbegin = cumsum(DeltaT(1:tsmooth-2)');
820     cbegin = cbegin(1:2:end)./(1:2:(tsmooth-2)');
821     cend = cumsum(DeltaT(M:-1:M-tsmooth+3)');
822     cend = cend(end:-2:1)./(tsmooth-2:-2:1)';
823     DeltaT = [cbegin;c(tsmooth:end);cend];
824
825     qq = fmincon(@(qq)(ihtp_func(qq,DeltaT,phi,Tiht,Apiht)),...
826         qq0,[],[],[],[],lb,ub,[],options);
827
828
829     Q_nonlin(pt,:) = qq';
830
831
832     if ~isempty(pbar)
833         pbar.Value = pt/Npts;
834     end
835 end
836
837
838 if ~isempty(pbar)
839     pbar.Message = 'Finishing up...';
840 end
841
842 Q = Q';
843 Q(:,startFrame:end) = Q_nonlin;
844 Q = reshape(Q,xlen,thlen,tlen);

```

```

845 ct = toc;
846 end
847
848 %% helper functions
849 function err = ihtp_func(qq,Tsurf,phi,Tiht,Apiht)
850
851 err = 0;
852 for ii = 1:length(Tsurf)-1
853     phi = Tiht*phi - Apiht*qq(ii);
854     Twa_an = phi(1);
855     err = err + (Tsurf(ii+1)-Twa_an).^2;
856 end
857
858 end
859
860 function err = ihtp_func_fastMat(qq,Tsurf,d,B)
861 % minimization function for the ihtp solver
862 % note that this is LINEAR so we can use efficient matrix algebra
863 % the inputs are as follows:
864 %
865 % qq: the estimate for the surface heat transfer (what we're looking for
866 % in the optimization)
867 % Tsurf: the experimental temperature history
868 % d: Tihtp^k(1,:) dot phi0, see write up
869 % b: a constant vector that is composed of the various powers of the
870 % transition matrix (first row) dotted with the bc vector, see write up
871
872 Twa_an = d + B*(-qq); % note b and qq have to both be column vecs
873
874 err = sum((Twa_an(1:end-1)-Tsurf(2:end))).^2;
875
876
877 end
878
879 function y = makePeriodic(x,Next,Nmean)
880 % calculates an extension to make x into a smooth periodic function

```

```
881 %
882 % SYNTAX:
883 %1 y = makePeriodic(x,Next,Nmean)
884 %
885 % DESCRIPTION:
886 % y = makePeriodic(x,Next,Nmean) calculates the cubic Bezier curve of length
      Next
887 % that will
888 % smoothly connect the last points of x to the first points,
889 % thereby creating one period of a smooth periodic function. This is useful
890 % for preparing a non-periodic signal for frequency decomposition with the fft
      .
891 % The Bezier curve matches both the value of the first and last point and
892 % the approximate slopes there.
893 %
894 % INPUTS:
895 % x:          Input signal {1-D column vector}
896 %             The input signal must be a column vector. If it isn't,
897 %             makePeriodic will throw an error when it tries to concatenate x
898 %             and the extension. There is no input validation in makePeriodic
899 %             because it is only intended to be used within the qcalc
900 %             function, where the input signal will always be a column
901 %             vector.
902 % Next:       Extension length {positive integer}
903 %             Next is the number of samples to extend the signal. It should
904 %             be an integer larger than 1.
905 % Nmean:      Number of points used in derivative estimate {positive integer}
906 %             For a noisy signal, the difference between the first two points
907 %             will be a poor approximation for the actual derivative there,
908 %             and similarly for the last two points. A better method is to
909 %             take the average slope over some Nmean number of points to
910 %             provide an estimate for the true derivative.
911 %
912 % OUTPUTS:
913 % y:          Output signal {1-D vector}
914 %             The output signal is [x; Extension].
```

```

915 %
916 % MORE ABOUT:
917 % A cubic Bezier curve is a curve described by only four points: two
918 % end points and two control points. This compact definition makes it a
919 % memory efficient way to generate a smooth curve. In addition, it has the
920 % nice property that the slope at the first point is equal to the slope of
921 % the line from the first point to the second point, and similarly for the
922 % end point. This makes it easy to match both slopes and values with a
923 % smooth curve, which is the desired behavior in makePeriodic.
924 %#
925
926 xdiff1 = sum(diff(x(1:Nmean)))/Nmean;
927 xdiffend = sum(diff(x(end-Nmean:end)))/Nmean;
928
929 P0 = x(end);
930 P3 = x(1);
931
932 halfDist = round(Next/2);
933
934 P1 = P0 + halfDist*xdiffend;
935 P2 = P3 - halfDist*xdiff1;
936
937 ss = linspace(0,1,Next)';
938
939
940 Ext = ((1-ss).^3)*P0 + (3*ss.*(1-ss).^2)*P1 + (3*(1-ss).*ss.^2)*P2 + (ss.^3)*
      P3;
941
942 y = [x; Ext];
943
944
945 end
946
947 function [algopts, flags] = parseAlg(algstr, algs)
948
949 if ismember('-', algstr)

```

```
950     out = split(algstr, '-');
951
952     numopts = length(out);
953
954     flags = {};
955     for nn = 1:numopts
956         flags = [flags; out{nn}];
957     end
958 else
959     flags = {algstr};
960 end
961
962
963 algopts.fast = any(strcmp(flags, 'fast'));
964 algopts.par = any(strcmp(flags, 'par'));
965 algopts.precalc = any(strcmp(flags, '2step'));
966
967 algind = ismember(flags, algs);
968
969 algopts.alg = flags{algind};
970
971 end
```

E.2 Image Processing

To study the growth of the stationary crossflow waves, it is useful to automatically extract the peak heating along streaks in the IR image. The algorithm used in this work is semi-automated, and proceeds as outlined in Table E.1.

Table E.1. Method to trace peaks in heat transfer images.

```

1: procedure StreakTracker(Q,X,theta)
2:   Img ← zeros(size(Q))           ▶ Img is set to 1 wherever there is a peak
3:   for all x in X do
4:     s ← Q(x,theta)               ▶ Get the spanwise slice at x
5:     s ← filter(s)
6:     Peaks ← findpeaks(s)
7:     Img(x,Peaks) ← 1
8:   end for
9:   Img ← morphops(Img)           ▶ Use morphological operations to connect
                                   ▶ unconnected parts of streaks and remove noise.
10:  CC ← bwconncomp(Img)          ▶ Get the connected components of the binary image
11:  for all streak in CC do
12:    U ← unique(streak.x(:))      ▶ Get the unique x values for the component
13:    streak.x ← U
14:    for all x in U do
15:      V ← streak.y(x)           ▶ Get all the y values at this x in this component
                                   ▶ There may be more than one y value for a given x due to the morph. ops.
16:      streak.y(x) ← mean(V)      ▶ Reduce to a single y value
17:    end for
18:    sort(streak.y,streak.x) by streak.y
19:  end for
20: end procedure

```

F. BICOHERENCE CODE AND VALIDATION

F.1 Code Listing

```

1 function [ bic, F1, F2, BS, sigmaBS, LS, bicH, lambda ] = bicohere( y, fs, N,
    PO, plot_bool )
2 % BICOHERE calculates the bicoherence and bispectrum of signal y(t)
3 % INPUTS:
4 % y:          signal of interest
5 % fs:         sampling frequency (Hz)
6 % N:          number of time segments for bispectrum analysis
7 % PO:         percent overlap of segments in range [0,1]
8 % plot_bool:  plot or no?
9 % OUTPUTS:
10 % bic:        magnitude squared bicoherence matrix
11 % F1:         bicoherence x variable (Hz) 0:fs/2 (meshgridded)
12 % F2:         bicoherence y variable (Hz) 0:fs/4
13 % BS:         unnormalized bispectrum (complex)
14 % sigmaBS:    unused, kept around for backwards compatibility
15 % LS:         length of one window
16 % bicH:       unused, kept around for backwards compatibility
17 % lambda:     unused, kept around for backwards compatibility
18 %
19 % for more information see:
20 % Digital Bispectral Analysis and Its Applications to Nonlinear Wave
21 % Interactions, Kim & Powers, IEEE Transactions on Plasma Science, Vol. 7,
22 % No. 2, 1979
23 %
24 % Normalizing Bispectra, Hinich & Wolinsky, Journal of Statistical Planning
25 % and Inference, Vol. 130, 2005
26 %
27 % Diagnosis of Process Nonlinearities and Valve Stiction
28 % Choudhury, Shah, and Thornhill

```

```
29 % AIC, Springer, 2008
30 %
31 % Josh Edelman, Jan. 2015
32 %
33 %% Changelog
34 %
35 % v3.1 - 2/24/2019
36 % -Removed difference interactions because apparently it is
37 % redundant information (see Kim and Powers original paper)
38 % (even though apparently a lot of people don't realize this?)
39 % -Squeezed every ounce of speed I could by including Matlab
40 % functions, reducing the domain to a minimum, etc.
41 % --NOTE: bicohere returns ONE SIDED spectrum and freq vectors now
42 %
43 % v3 - 2/20/2019
44 % -Brand new algorithm for computing Y3k, using Hankel matrices! Blazing
45 % fast now
46 % -added computation of difference interactions
47 % --NOTE: frequency vectors now include negative frequencies!!
48 % --NOTE: removed one-sided spectrum compensation b/c including both sides
49 %         to account for difference interactions
50 % -added power compensation for window (was incorrect before, but only
51 %         affects the non-normalized bispectrum)
52 % -changed from Hann window to Blackman window
53 % -averaging of bispectrum is now correct
54 %
55 % v2.5 - 10/24/2018
56 % -corrected normalization for Kim and Powers
57 % -added normalization for Hinich
58 %
59 %
60 % v2.0 - 9/25/2016
61 % -changed to more efficient, matrix-based algorithm
62 %
63 % v1.1 - 7/7/2015
64 % -corrected definition of LS
```

```

65 % -corrected treatment of one-sided spectrum
66 %
67 %
68 % v1.0 - Jan 2015
69
70 %% define variables
71 tic
72 LS = floor(length(y)/(N+P0*(1-N))); % length of segment
73
74 % if LS isn't even, make it so
75 % this greatly simplifies things later on
76 % (could have made it odd, but it's too late now)
77 if mod(LS,2)
78     LS = LS-1;
79 end
80
81 SZ = LS/2;      % this is why it's nice to have LS even
82
83 RTP1 = zeros(SZ,round(SZ/2));
84 RTP2 = RTP1;
85 B = RTP2;
86
87 lambda = B;
88 bicH = 0;
89
90 % custom Blackman window
91 % to avoid all the overhead from
92 % the blackman function call
93 wSamps = (0:LS-1)/LS;
94 WIN = 0.42 - 0.5*cos(2*pi*wSamps) + 0.08*cos(4*pi*wSamps);
95 Wcomp = sqrt((WIN*WIN')/LS);      % window compensation
96
97 sigmaBS = 0;
98
99 if ~isrow(y)
100     y = y';

```

```

101 end
102
103
104 %% calculate bispectrum for each segment
105
106 % put the fdomain functionality right here to avoid function overhead
107 fks=0:LS/2-1; % LS is ALWAYS even (see above)
108 T=LS/fs;
109 w=2*pi*fks/T;
110 f = w/(2*pi);
111
112 SZ2 = round(SZ/2);
113
114 % same thing with Matlab's hankel function
115 NC = SZ;
116 NR = SZ2;
117 Rmat = zeros(NC-2,1);
118
119 % the tril only keeps the principle domain of the bicoherence,
120 % but it is NOT present in Matlab's hankel
121 IJhankel = tril((1:NC)' + (0:(NR-1)));
122 IJhankel(IJhankel==0) = LS-1; % everything else is set to 0 (Rmat(end))
123
124
125 for k = 0:N-1
126     startIndex = floor((1-P0)*k*LS)+1;
127     endIndex = startIndex + LS - 1;
128
129     yk = y(startIndex:endIndex).*WIN; % define segment for analysis
130     yk = yk-sum(yk)/LS;
131
132     % Normally need a 1/LS on fft output,
133     % but it cancels for bicoherence
134     Yk=fft(yk);
135
136     % Strictly speaking we don't need to account

```

```

137 % for the one-sidedness or window because those
138 % factors all cancel in bicoherence,
139 % but it really seems wrong to not include them...
140 Yk = 2*Yk(1:SZ)/LS;
141 Yk = Yk./Wcomp;
142
143 Ykt = transpose(Yk);
144 YYk = Ykt*Yk(1:SZ2); % build up the matrix of Yk*Yl for all (k,l)
145
146 % Hankel matrix is constant on anti-diagonals
147 % recall that f1+f2=f3 is constant on
148 % anti-diagonals, so Hankel is perfect! And
149 % super efficient!
150
151 X = [Ykt; Ykt(SZ); Rmat]; % do the hankel matrix creation
152 Y3k = X(IJhankel);
153
154 %NOTE: at least on my computer, real(x).^2 + imag(x).^2
155 % is actually noticeably faster than abs(x).^2
156 % (maybe 20 ms out of 200 ms run time)
157 RTP1 = RTP1 + (real(YYk).^2 + imag(YYk).^2);
158 RTP2 = RTP2 + (real(Y3k).^2 + imag(Y3k).^2);
159 B = B + YYk.*conj(Y3k);
160
161
162 end
163
164 % Note again we don't account for norm in average (1/N)
165 % because its 1/N^2 for top and 1/N * 1/N for bottom (they cancel)
166 bic = (abs(B).^2)./(RTP1.*RTP2);
167
168 F1 = f;
169 F2 = f(1:SZ2);
170 BS = B/(N*(LS^3)); % need to account for the lack of 1/LS in matlab
171 % because we aren't normalizing by RTPs here
172

```

```
173
174 %% return variables and plotting
175
176 if plot_bool
177     bicMax = max(max(bic));
178
179     figure
180     ah1 = subplot(4,1,2:4);
181     imagesc(f/1000,f(1:SZ2)/1000,transpose(bic));
182     set(gca,'ydir','normal');
183     xlabel('f_1, kHz');
184     ylabel('f_2, kHz');
185     %zlim([0,1]);
186     colormap(flipud(gray))
187     colorbar('Location','east');
188     %caxis([0,0.5]);
189     hold on
190     plot3([0,fs/2000],[0,fs/2000],[bicMax,bicMax],'k:','LineWidth',0.5);
191     axis equal
192     xlim([0,fs/2000]);
193     ylim([0,fs/4000]);
194
195     ah2 = subplot(4,1,1,'align');
196     [pxx,f2] = pwelch(y,WIN,round(P0*LS),LS,fs);
197     semilogy(f2/1000,pxx);
198     axis tight
199     xlim([0,fs/2000]);
200
201     title('Welch Periodogram and Squared Bicoherence');
202 end
203
204 %profile viewer
205 toc
206 end
```

F.2 Code Validation

The above code was validated using a nonlinear input signal y :

$$\begin{aligned} f_1 &= 100 \text{ kHz}, \\ f_2 &= 245 \text{ kHz}, \\ f_s &= 2 \text{ MHz}, \\ x &= \sin(2\pi f_1 t) + \sin(2\pi f_2 t), \\ y &= x + 0.05x^2 + n(t). \end{aligned}$$

The quadratic operation on x results in six phase-locked frequency components in y :

1. $(f_1, f_1) = (100, 100)$ kHz
2. $(f_2, f_2) = (245, 245)$ kHz
3. $(f_1, f_2) = (100, 245)$ kHz
4. $(f_1, f_2 - f_1) = (100, 145)$ kHz
5. $(f_2 - f_1, f_2 + f_1) = (145, 345)$ kHz
6. $(f_2 - f_1, 2f_1) = (145, 200)$ kHz

Note that because the bicoherence is symmetric about the line $f_1 = f_2$, the order of the frequencies doesn't matter. Figure F.1 shows the power spectrum and bicoherence of y . The frequency resolution was set to 2 kHz. There are 6 points of bicoherence nearly equal to 1, corresponding to the expected points, above.

F.3 Performance of Bicoherence on Hidden Signals

In the validation case presented in the previous section, all of the harmonics were obvious in the power spectrum and could easily be tied to the relevant peaks in the bispectrum. This is not always the case, but as the following examples demonstrate relying solely on the power spectrum

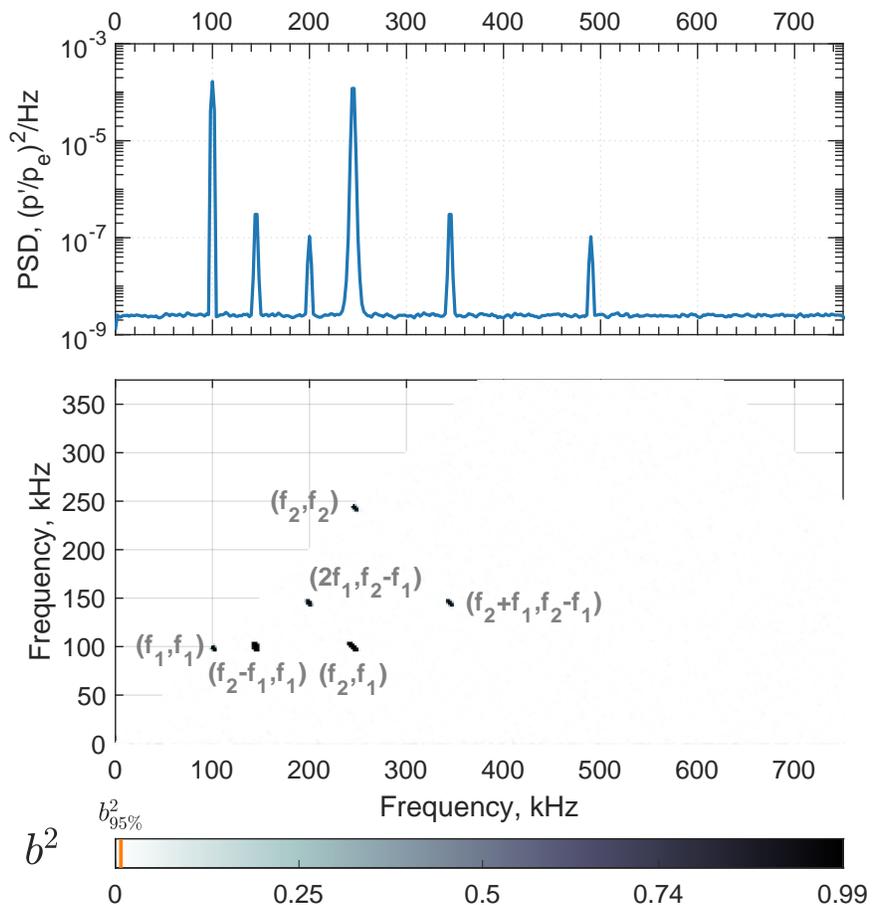


Figure F.1. Power spectrum and bicoherence for the validation case.

to determine the presence of nonlinear signals can sometimes overlook important features of the signal.

F.3.1 Single Harmonic in the Presence of Noise

The signal $x(t)$ is constructed as the sum of a sine wave and its phase-locked first harmonic,

$$x(t) = 10 \sin(\omega t + \phi) + 0.075 \sin(2\omega t + 2\phi) + n(t), \tag{F.1}$$

where $n(t)$ is the sum of zero-mean, Gaussian noise and a second component that is the integral of Gaussian noise to provide an increase in low-frequency noise power that can be observed in the experimental power spectra.

The power spectra for $\omega_1 = 2\pi 100$ kHz are shown in Figure F.2 both with and without the added noise term $n(t)$. Without noise, the 100 kHz peak and its harmonic are obvious. When noise of a sufficient level is added, the harmonic at 200 kHz is completely obscured. Using only the power spectrum, it would seem in the noisy case that the signal has no phase-locked components; indeed in that spectrum there is only a single frequency component (other than the wide-band noise).

The bicoherence, however, is able to accurately ascertain that there is phase locking between 100 kHz and 200 kHz, even when the latter is not apparent in the power spectrum. The magnitude of the bicoherence is substantially reduced—in theory it should be 1.0 at (100 kHz, 100 kHz)—but the presence of a nonlinearity is clearly indicated.

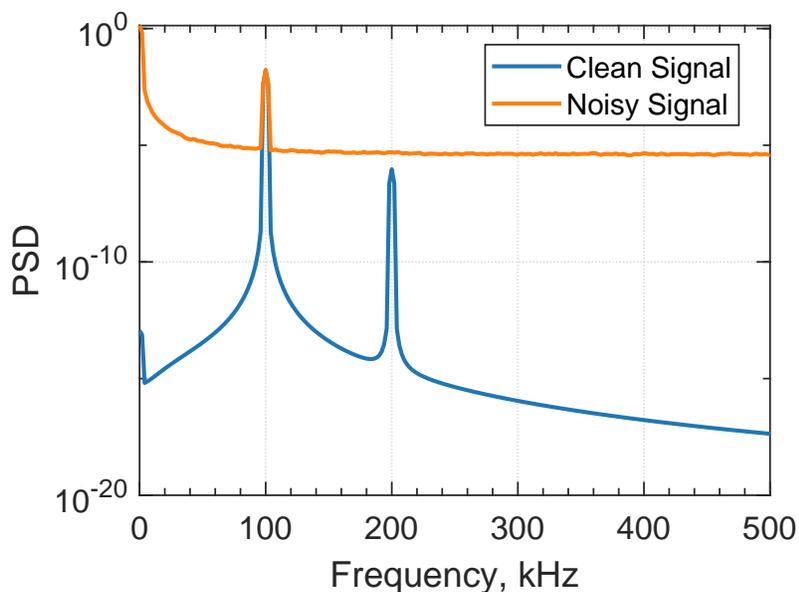


Figure F.2. Power spectra from the first test case: a single sinusoid and its phase-locked first harmonic, with and without additive noise.

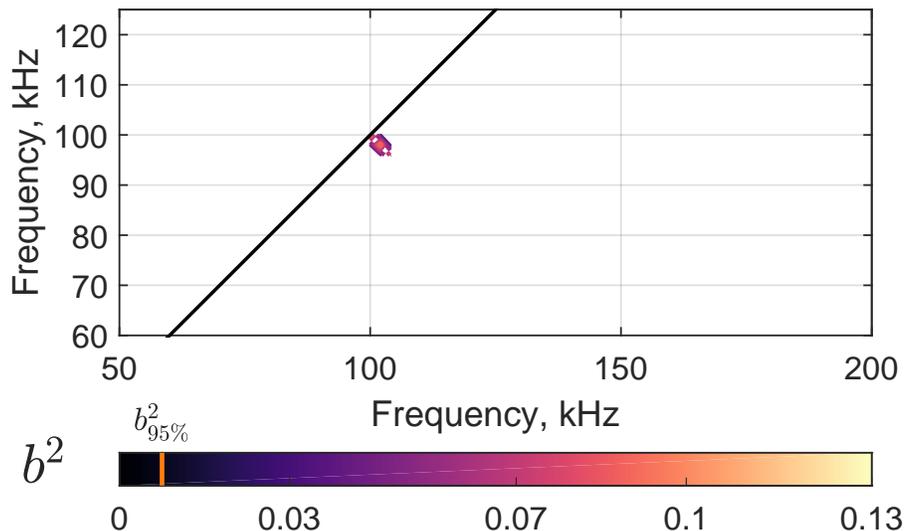


Figure F.3. The bicoherence of the first test case with additive noise. Note that the maximum theoretical bicoherence is 1.0 at (100 kHz, 100 kHz).

F.3.2 Wide-Band Quadratic Interaction With Noise

Perhaps a more relevant test case to the present work is a quadratically nonlinear, wide-band signal. Figure F.4 shows the power spectra of x , $0.25x^2$, and the combined signal $x + 0.25x^2 + n(t)$. Here $x(t)$ is the sum of sinusoids from 35 kHz to 150 kHz every 0.5 kHz with amplitudes peaking at 45 kHz. Note that in the spectrum of the combined signal, it is not apparent that there is a harmonic at all. The wide-band of the primary signal overlaps with much of the harmonic leading to what appears to be a single, wide-band signal with a long roll-off. The bicoherence for this case is shown in Figure F.5. The bicoherence accurately captures the quadratic nonlinearity, with a maximum of 0.99.

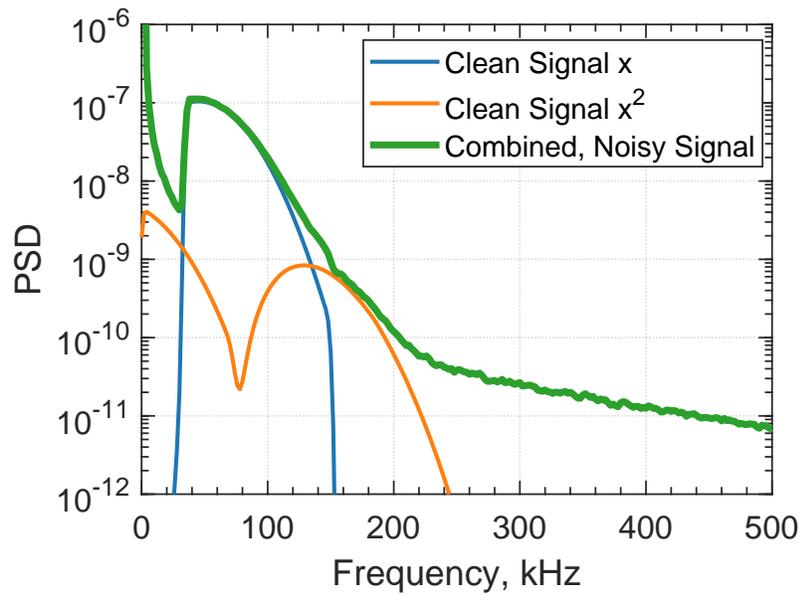


Figure F.4. Power spectra for individual components of a quadratically-nonlinear signal and their sum.

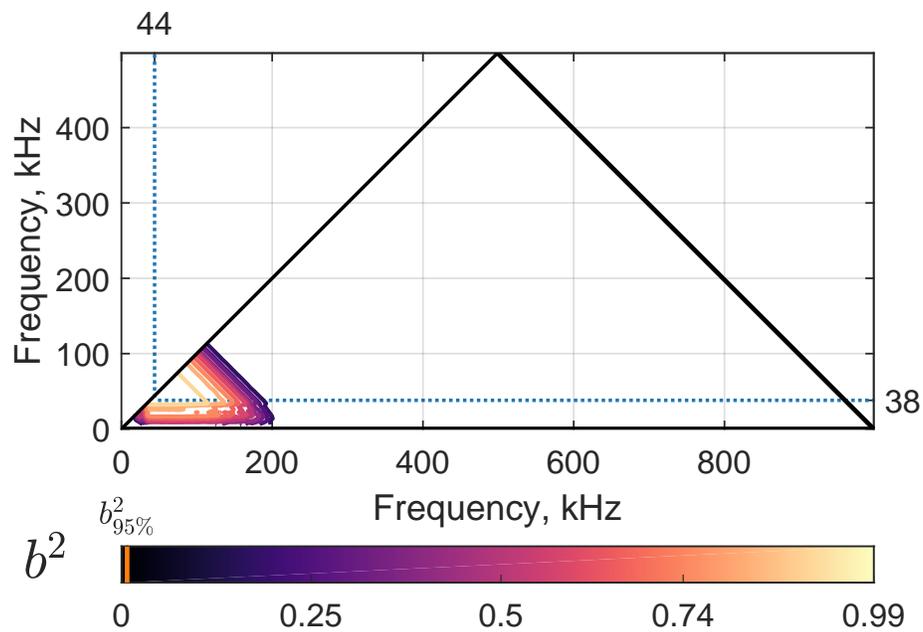


Figure F.5. The bicoherence of the wide-band test case accurately captures the nonlinearity in the signal.

G. NONLINEAR RESPONSE OF A PCB PRESSURE TRANSDUCER TO MECHANICAL AND ACOUSTIC IMPULSE

Because the bicoherence was used to understand the behavior of nonlinear breakdown, it is important to examine the nonlinear mechanical response of the PCB sensors to ensure that it is not confused for flow mechanisms. To do this, the sting was hit with a mallet and the sensor response was recorded. Figure G.1 shows the time-domain response of the PCB to the mallet impulse. Unfortunately the signal has clipped at ± 0.025 V, but it is still useful. The orange section shows the region used for spectral analysis. Figure G.2 shows the PSD of the response. Note the small peaks at 5 kHz and 8 kHz, followed by a rapid roll off. Figure G.3 shows the bicoherence of the same time data. There are only two regions of non-negligible bicoherence: a strong interaction region at about (30 kHz, 10 kHz) and a weaker interaction region around (600 kHz, 20 kHz). Neither of these regions are in the frequency band of the measured secondary instability.

Because the sensors are used to measure acoustic fluctuations and not mechanical loads, it is also important to understand any nonlinear behavior in the sensor due to these fluctuations. Data from the Purdue 3-inch Shock Tube was provided by Wason [80]. A PCB sensor was placed in pitot configuration, and an incident shock was measured as shown in Figure G.4. The bicoherence of the sensor response following this shock is provided in Figure G.5. As with the mechanical impulse, the sensor nonlinearities do not appear to be significant in the regions of interest. A closer view of the region below 1 MHz is shown in Figure G.6. There does seem to be weak nonlinearity associated with the sensor sub-resonance at 300 kHz and dispersed through the region of interest, though due to the very short time scale of the shock tube there is considerable noise in the bicoherence (not enough window averages can be taken). It is not clear how inherent sensor nonlinearities would interact with true nonlinearities in the measurement.

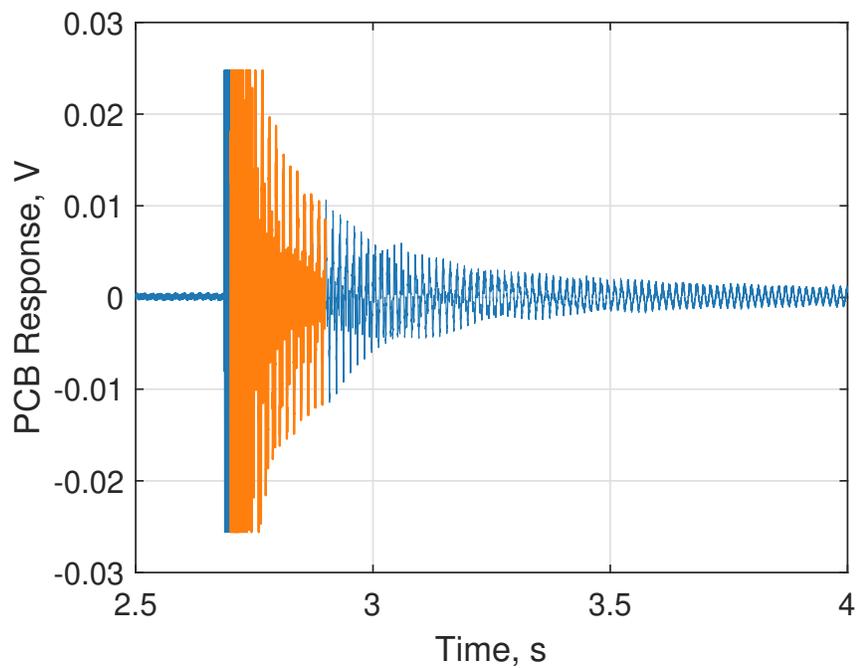


Figure G.1. The sensor voltage response to the mallet hit.

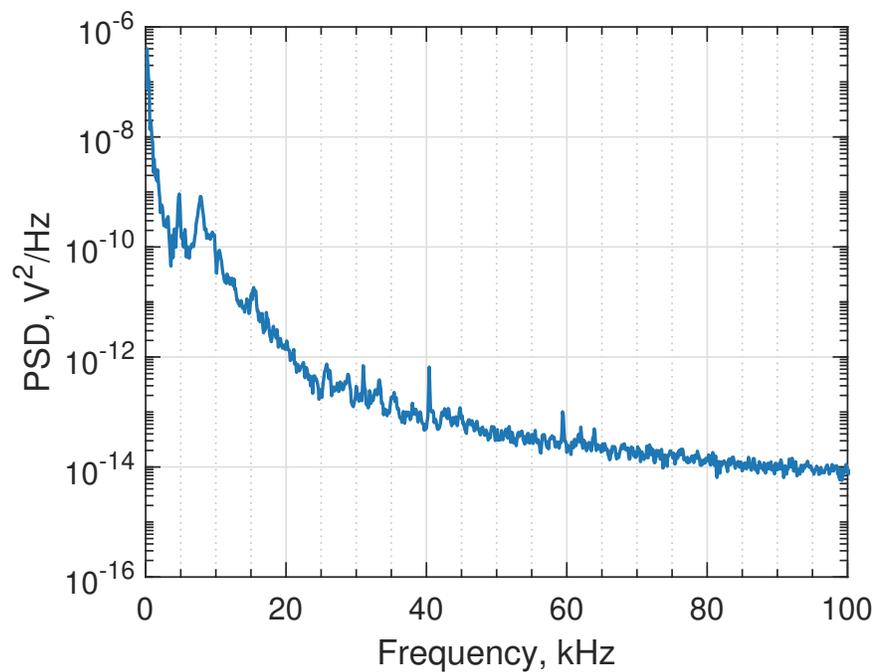


Figure G.2. Spectrum of the PCB's mechanical response.

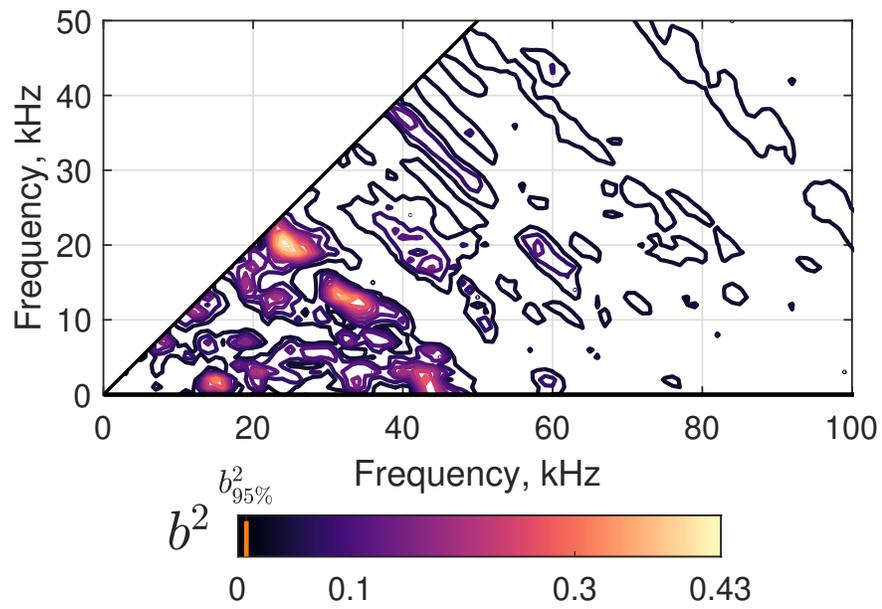


Figure G.3. Bicoherence of the PCB's mechanical response.

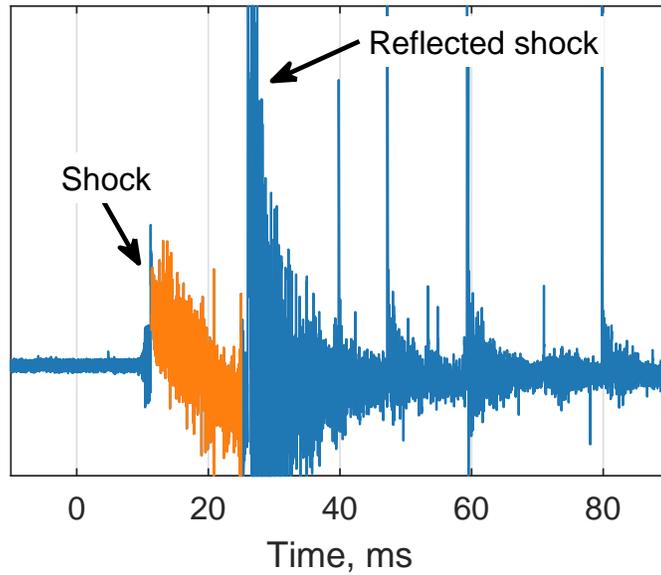


Figure G.4. Sensor response to an acoustic impulse (sensor in pitot mode in a shock tube, see Ref. [80]). The orange section was used for the following analysis. Amplitude units are arbitrary.

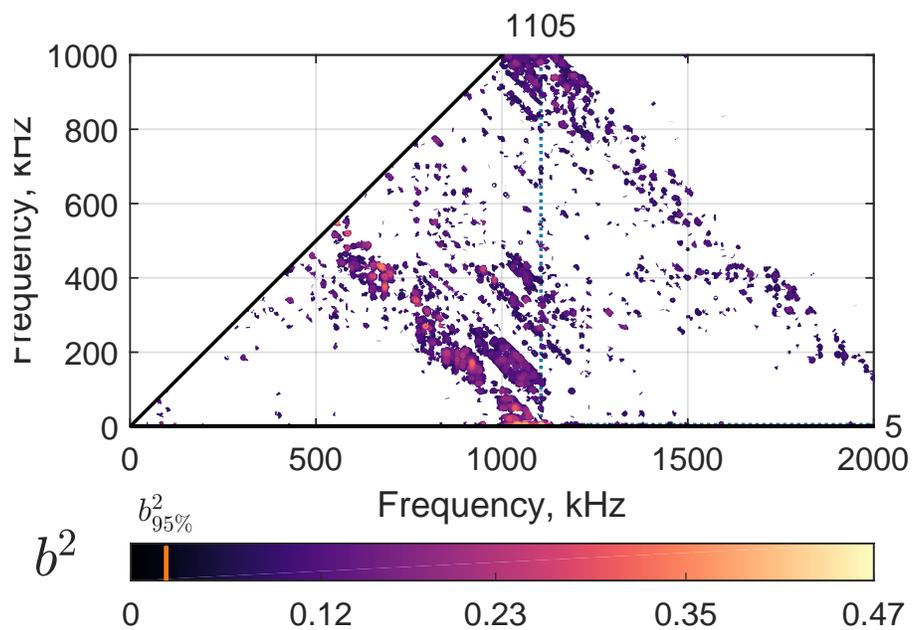


Figure G.5. Bicoherence of the PCB's response to the shock.

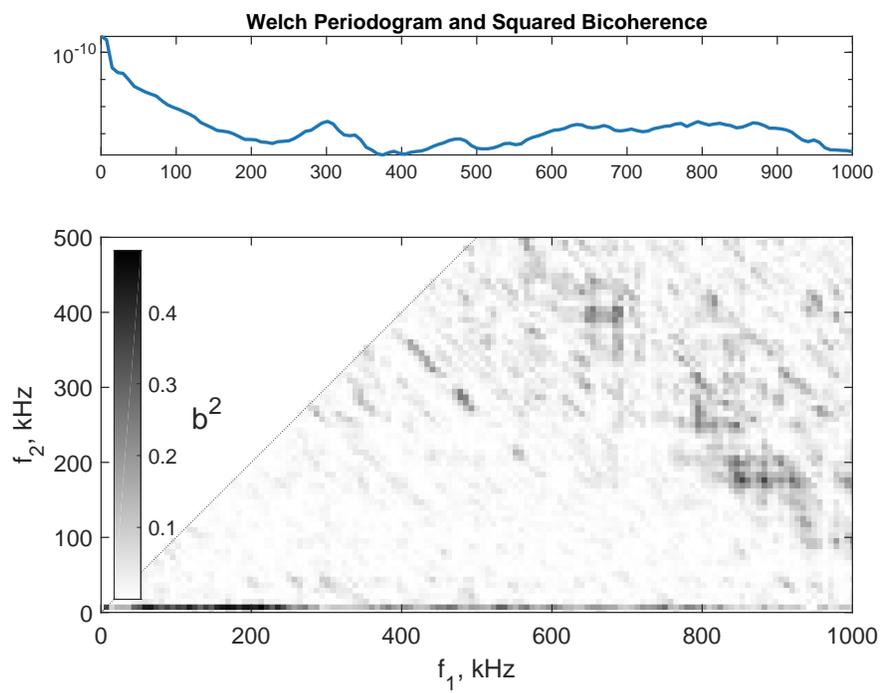
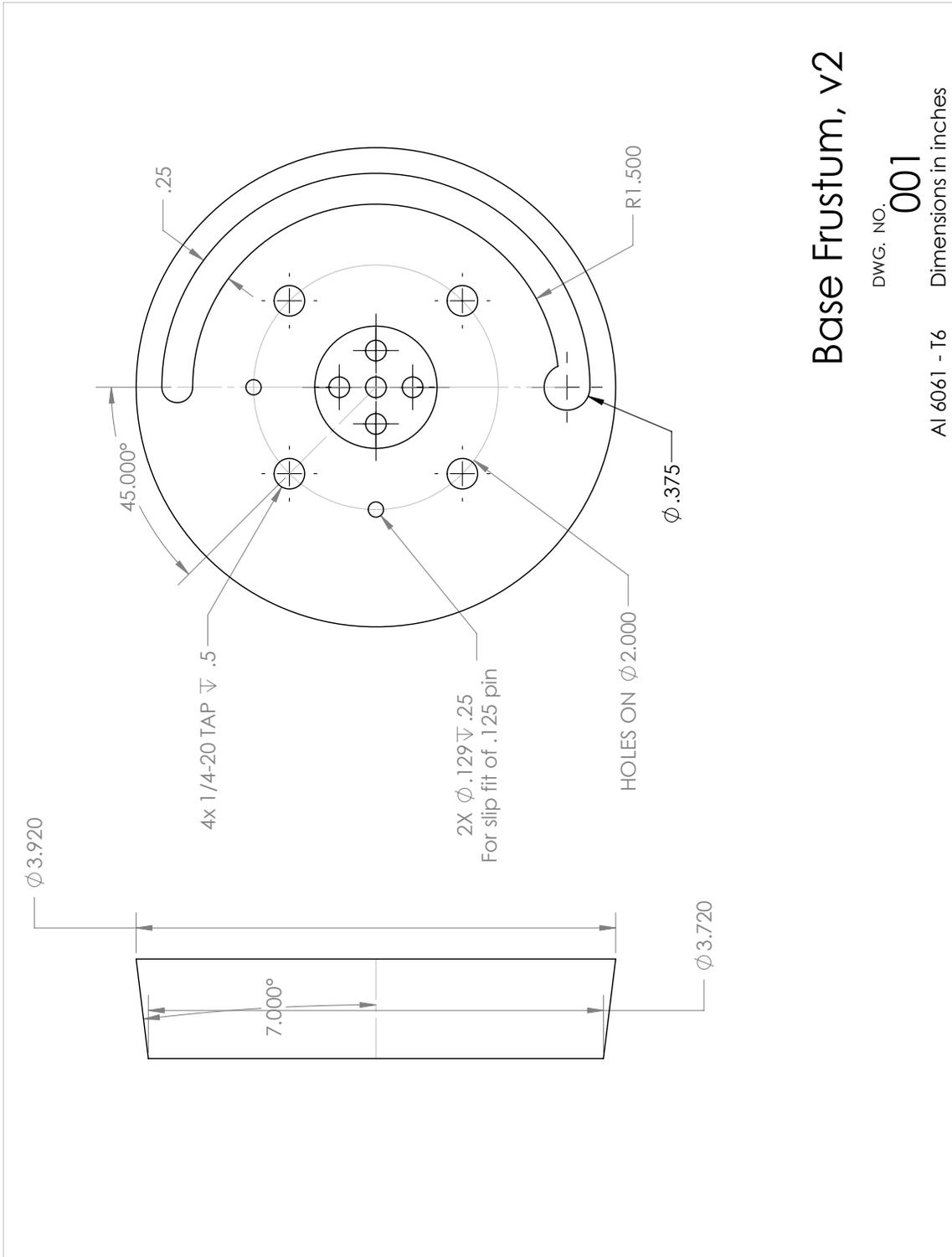


Figure G.6. Bicoherence of the PCB's response to the shock, refined view.

H. ENGINEERING DRAWINGS

Modular Cone Drawings

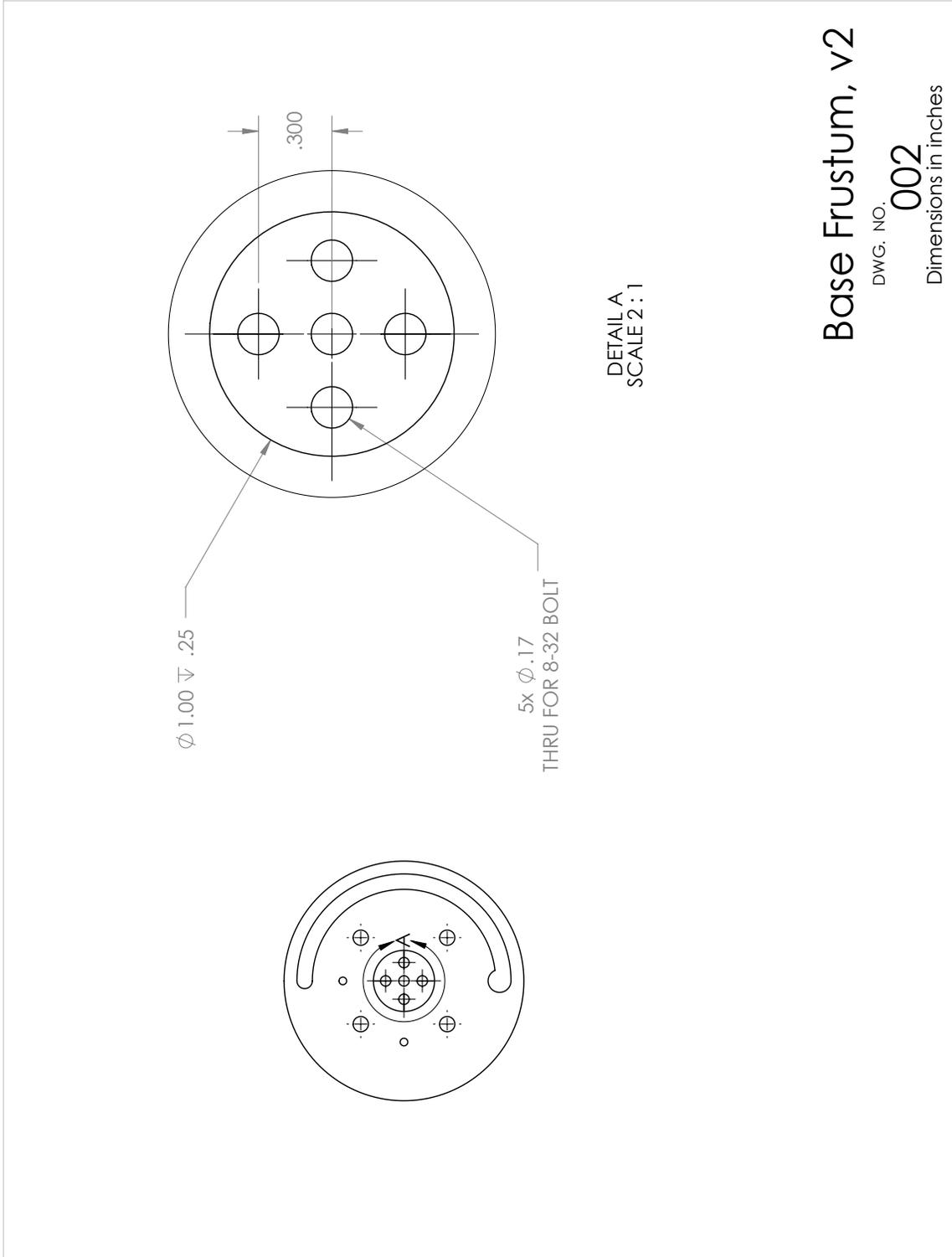


Base Frustum, v2

DWG. NO. 001

AI 6061 - T6 Dimensions in inches

Modular Cone Drawings



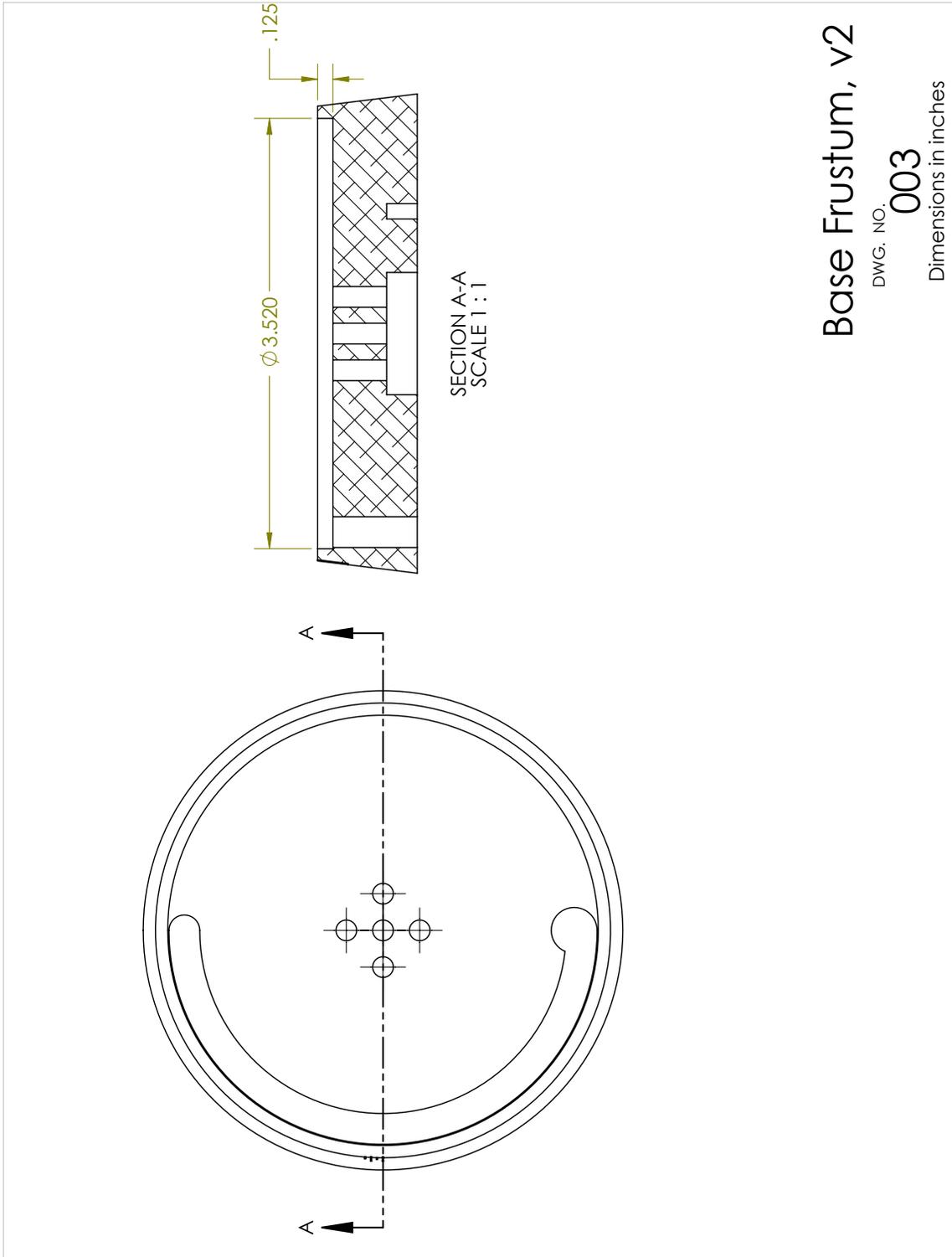
Base Frustum, v2

DWG. NO. 002

Dimensions in inches

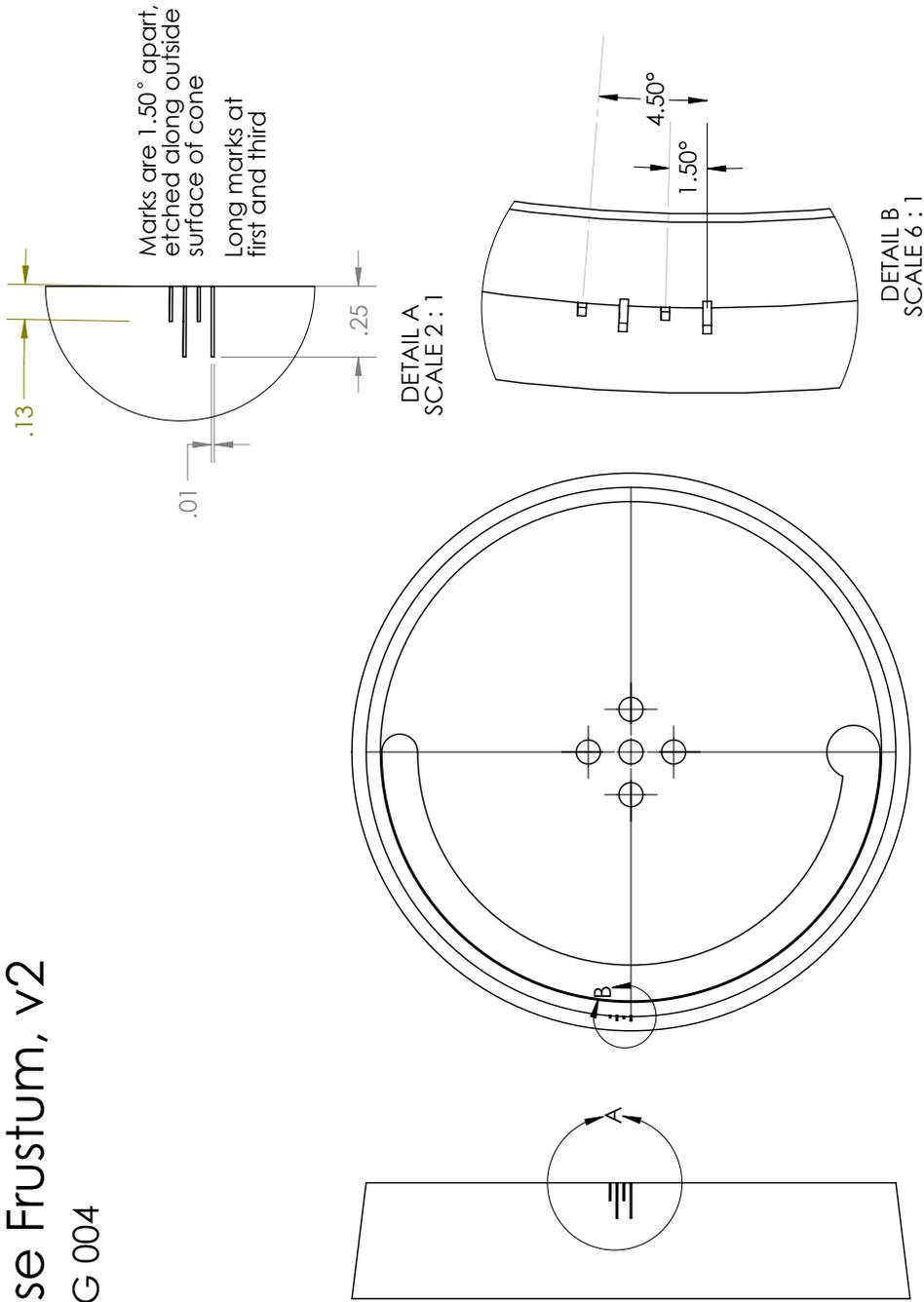
DETAIL A
SCALE 2:1

Modular Cone Drawings



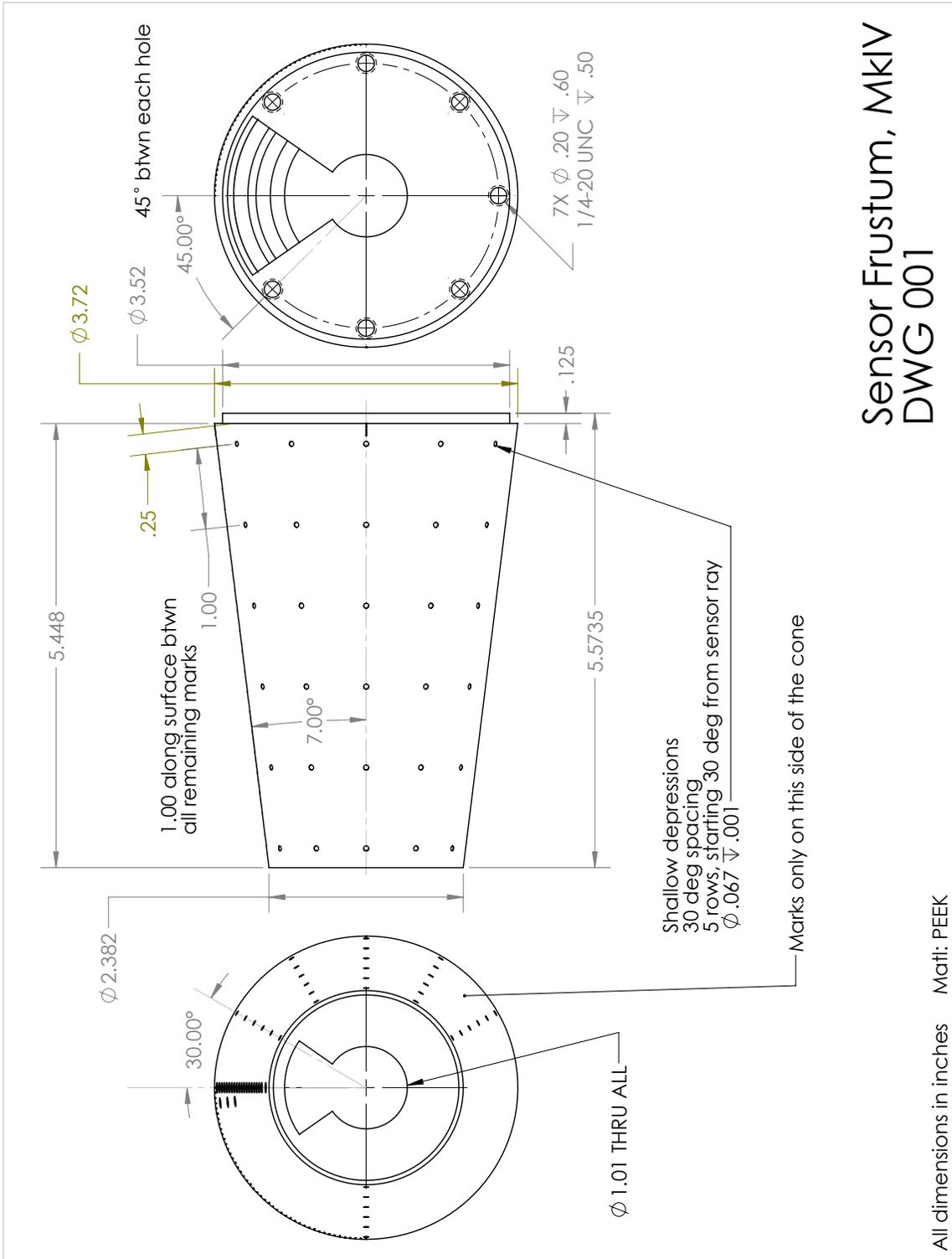
Modular Cone Drawings

Base Frustum, v2
DWG 004

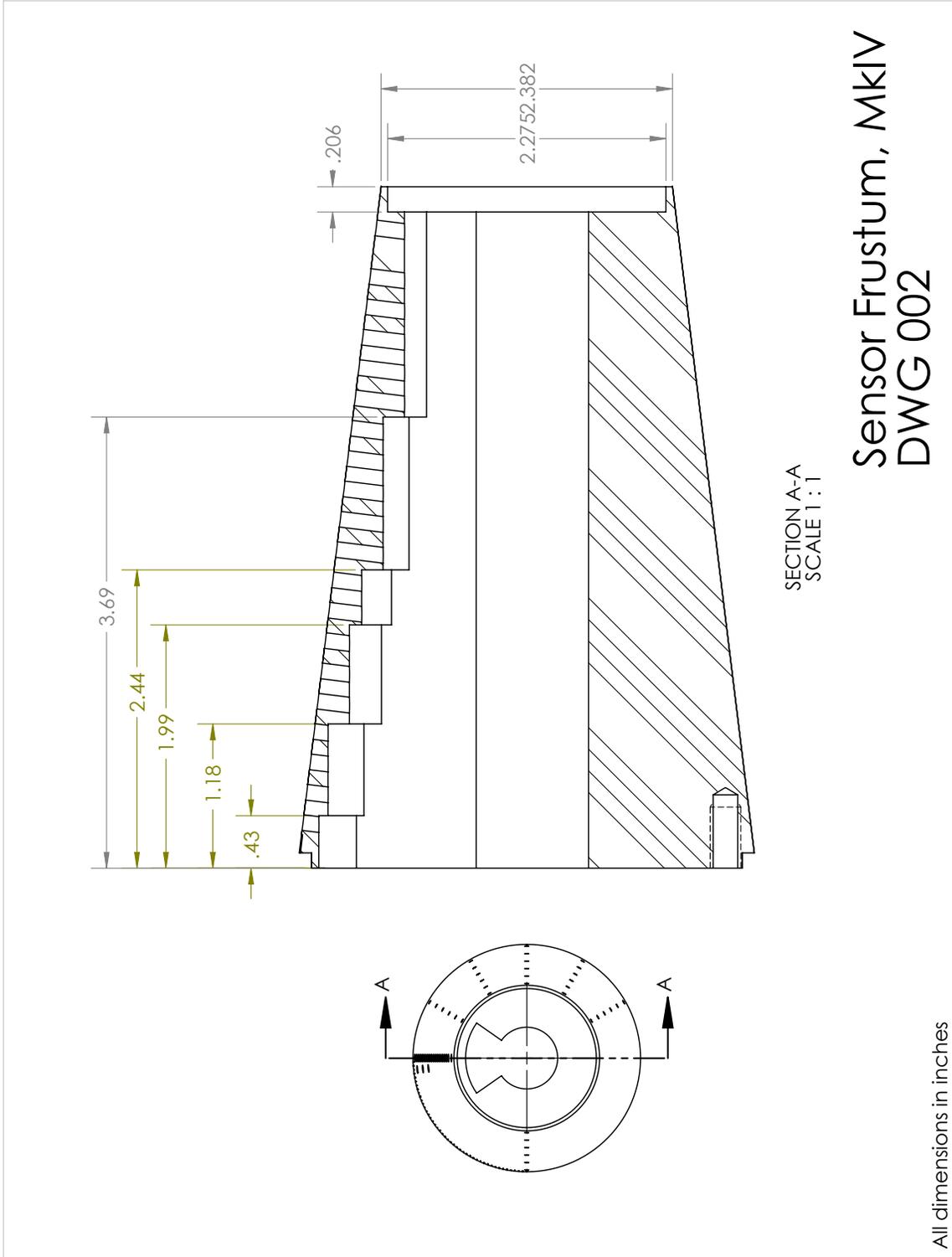


All dimensions in inches

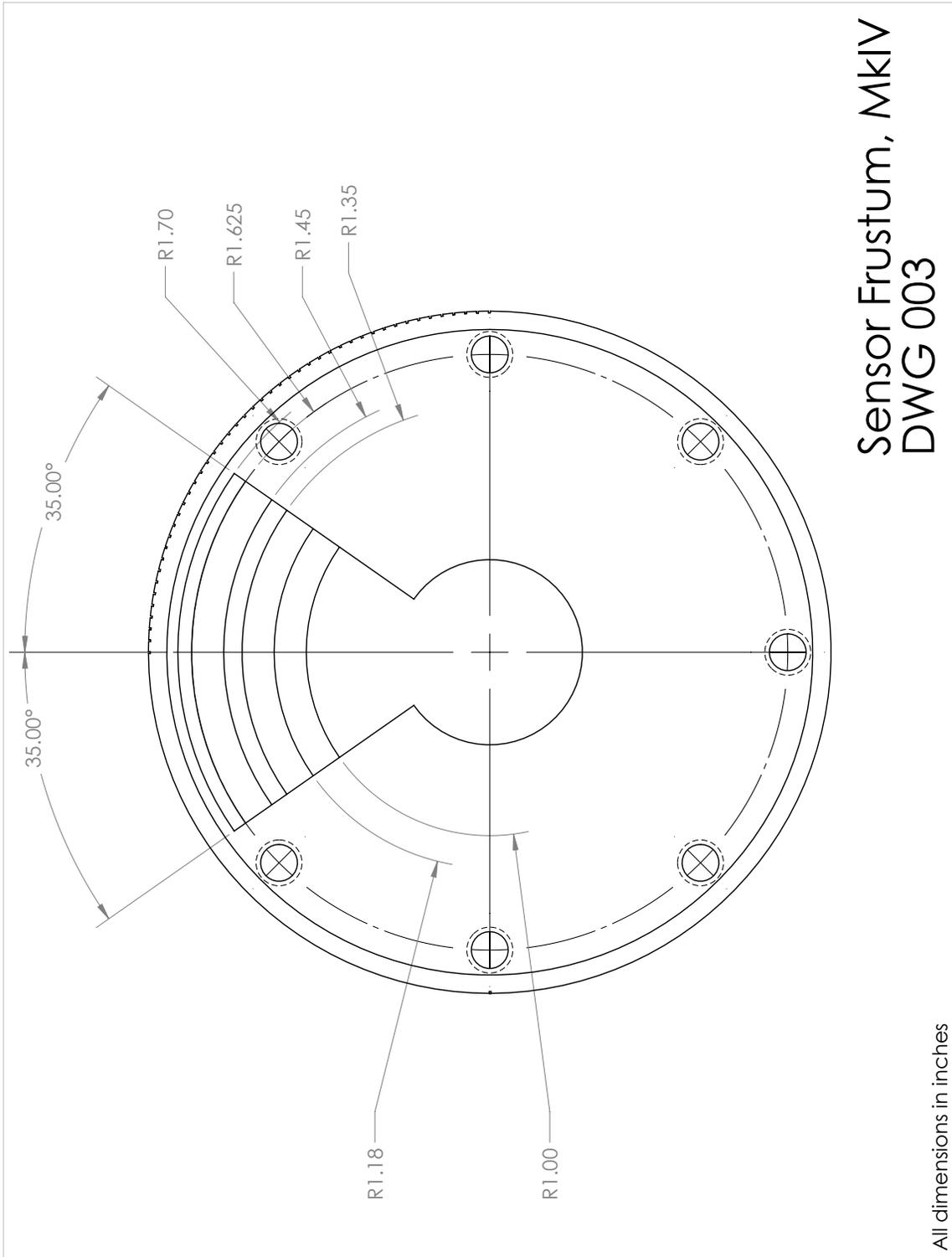
Modular Cone Drawings



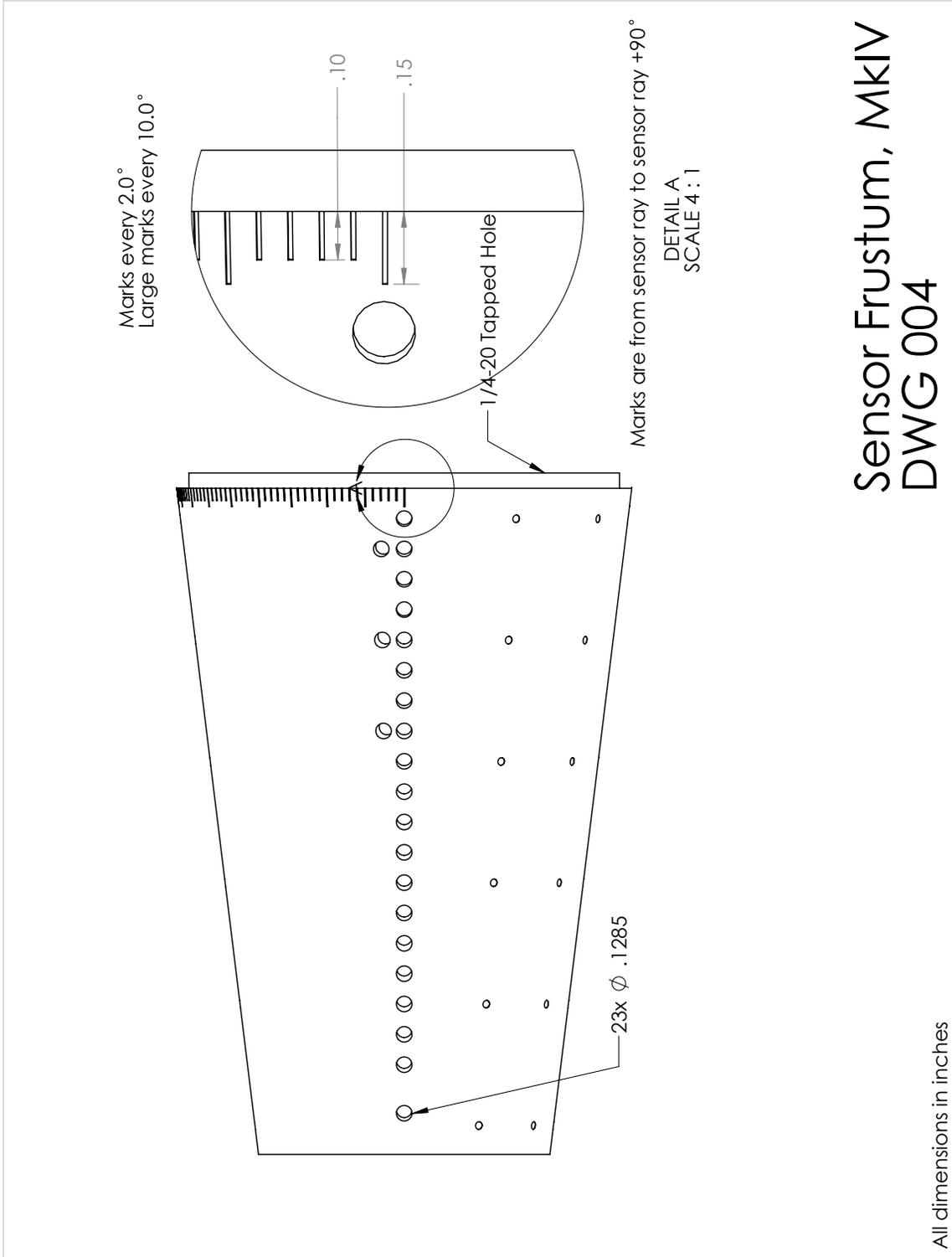
Modular Cone Drawings



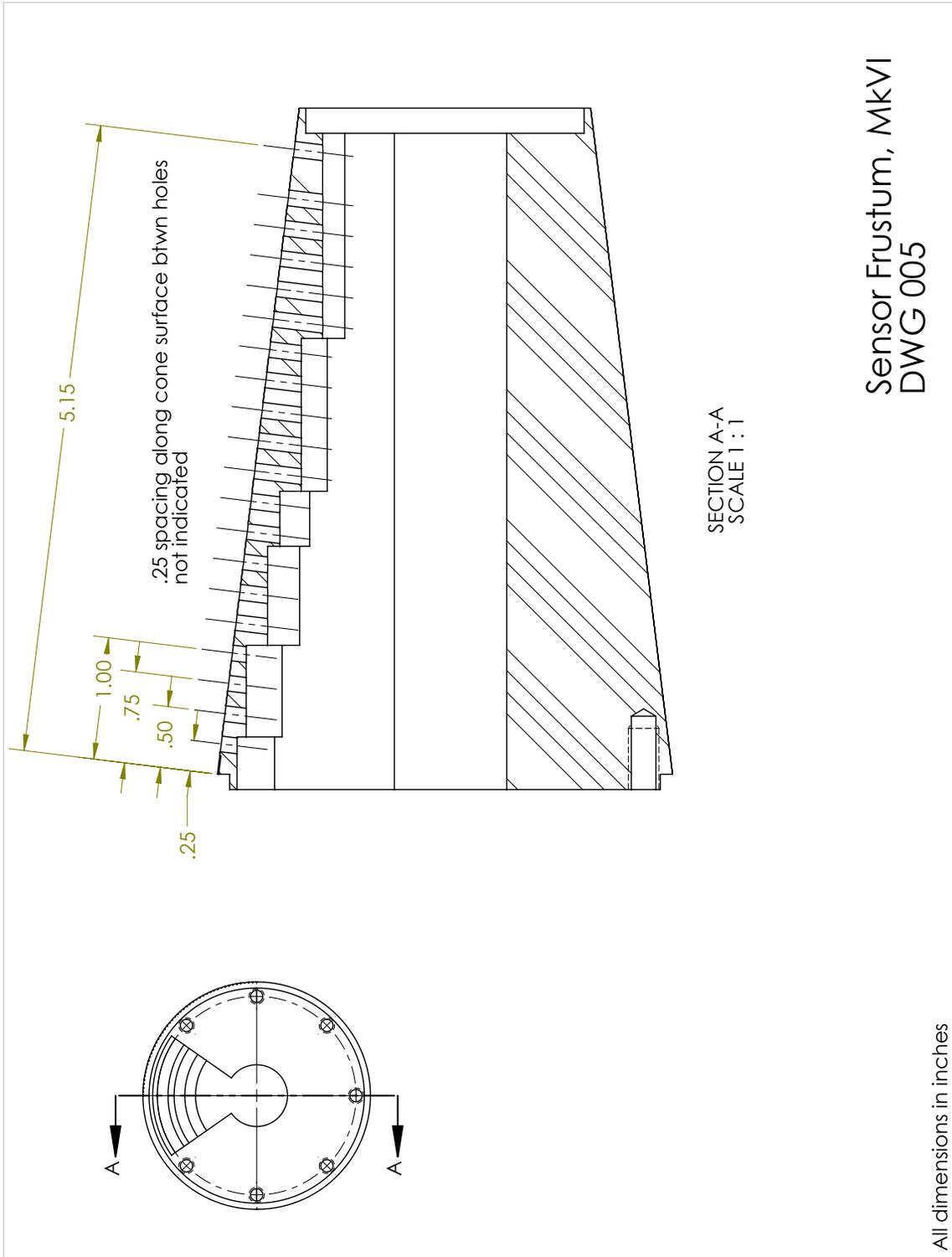
Modular Cone Drawings



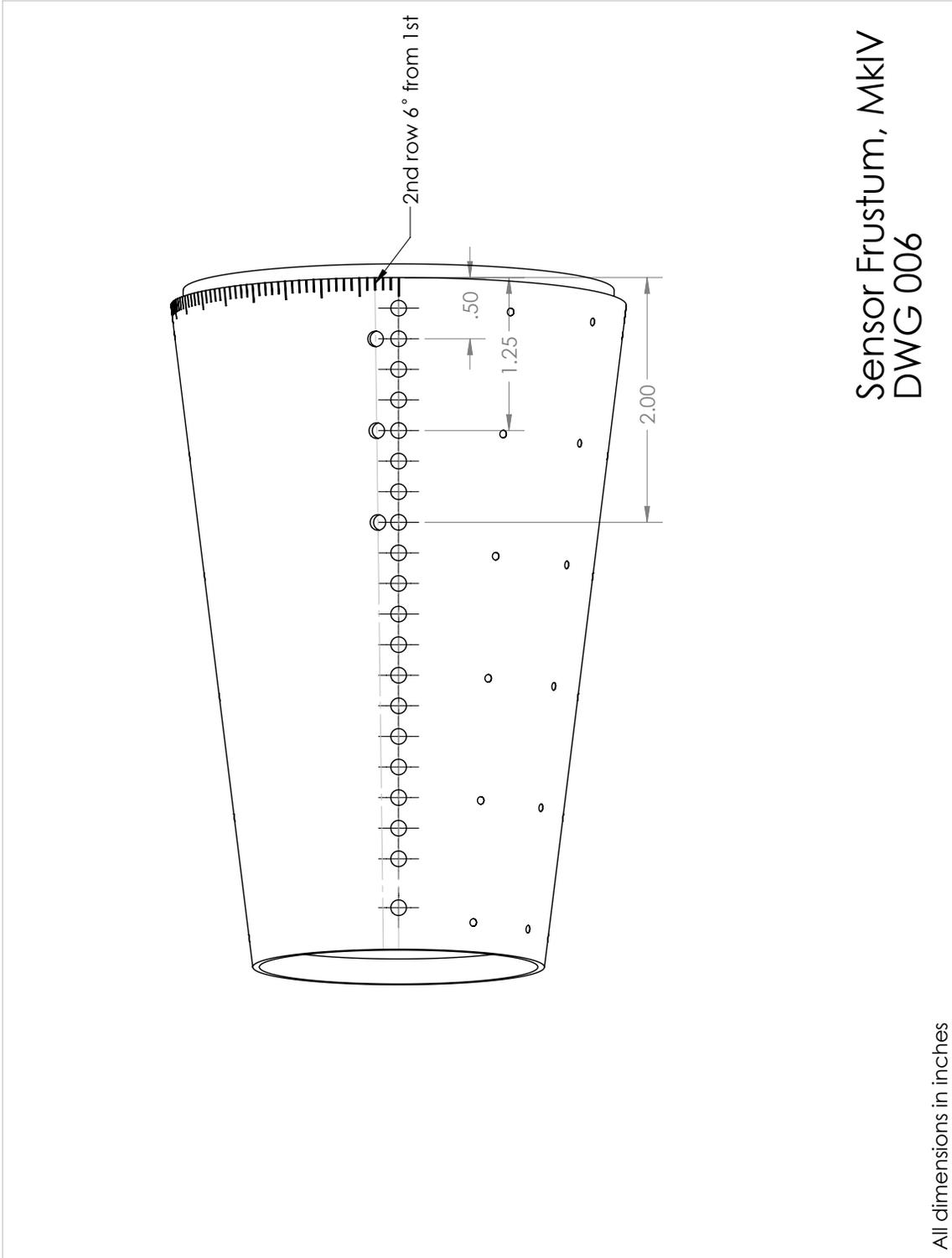
Modular Cone Drawings



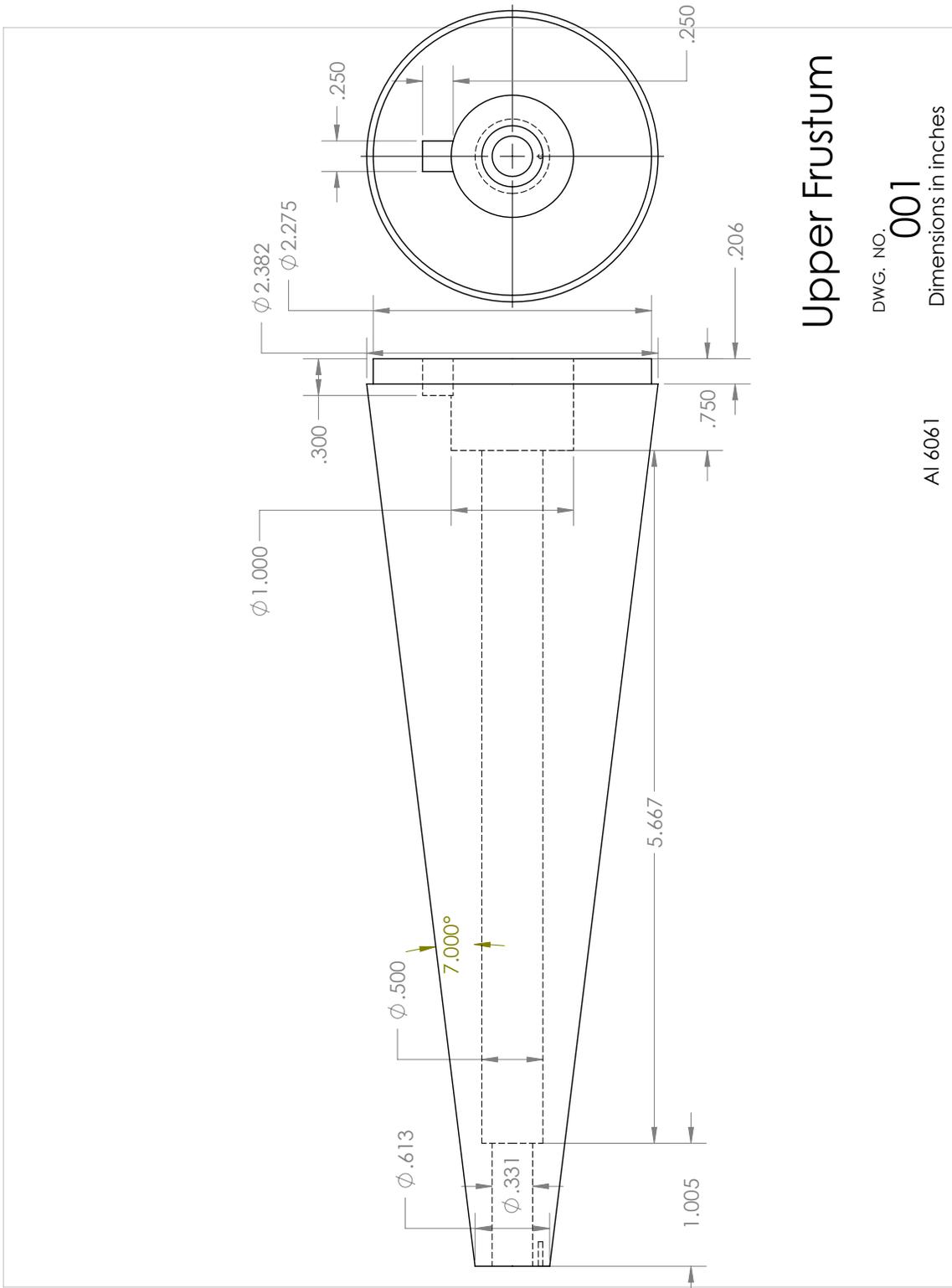
Modular Cone Drawings



Modular Cone Drawings



Modular Cone Drawings

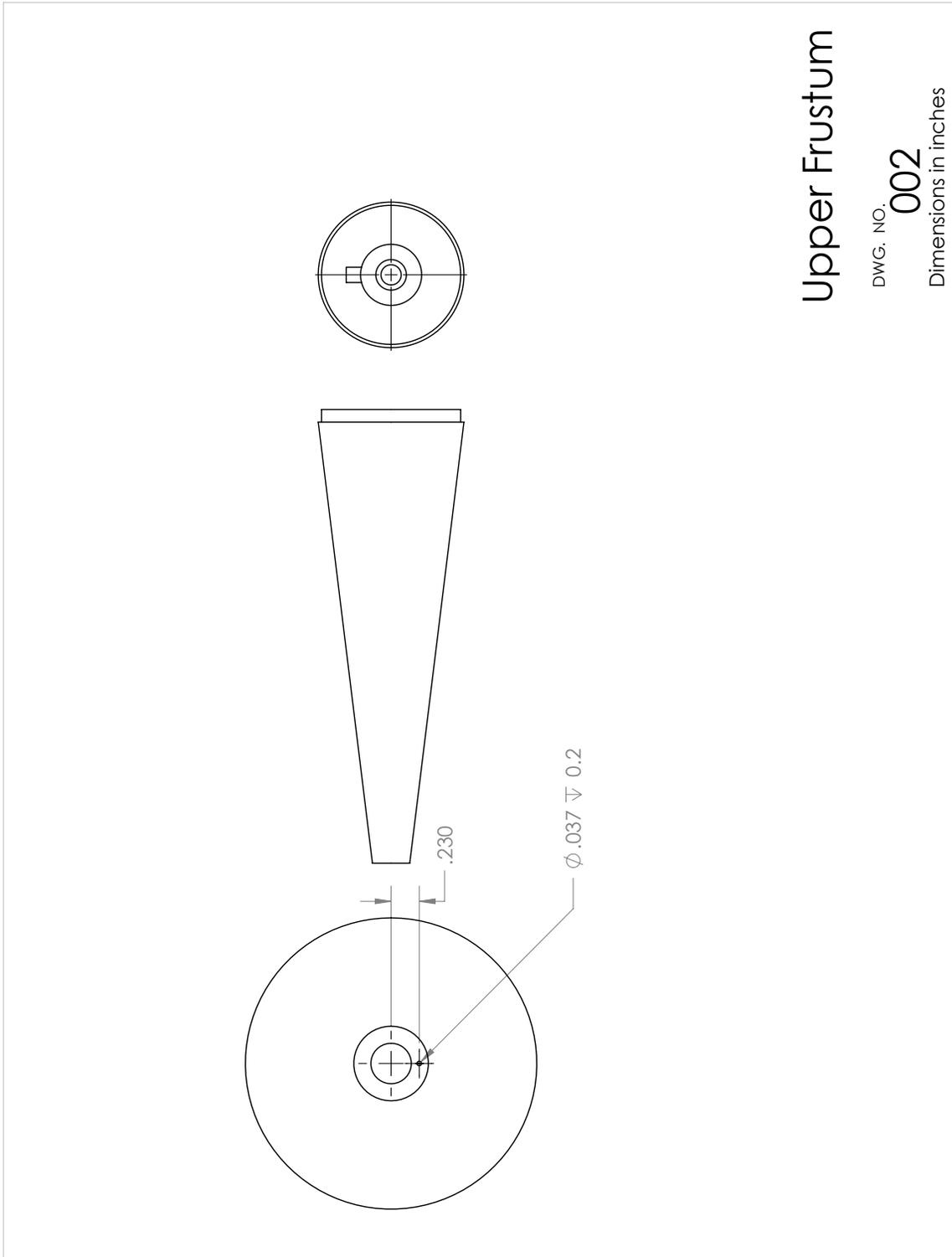


Upper Frustum

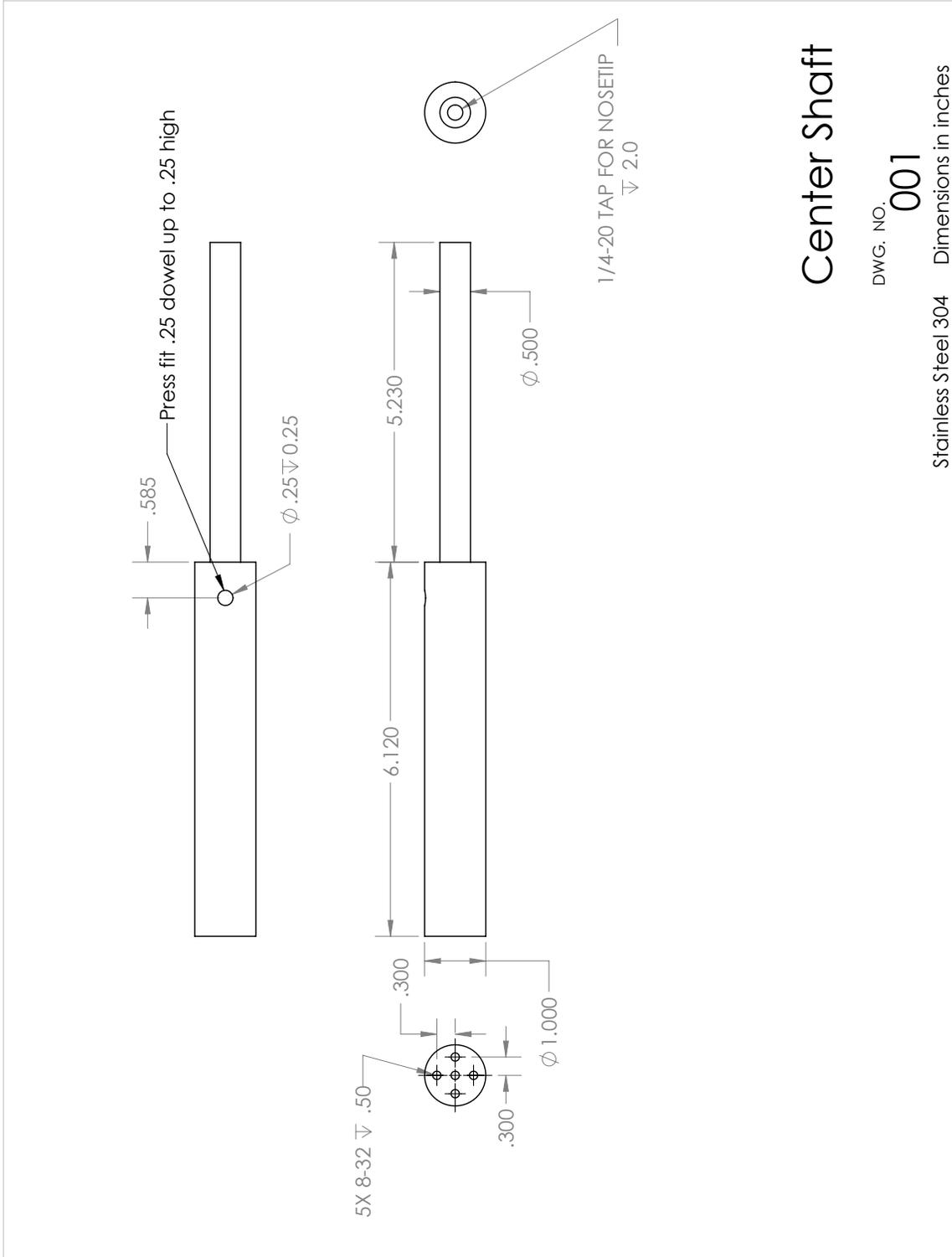
DWG. NO. 001
Dimensions in inches

AI 6061

Modular Cone Drawings



Modular Cone Drawings

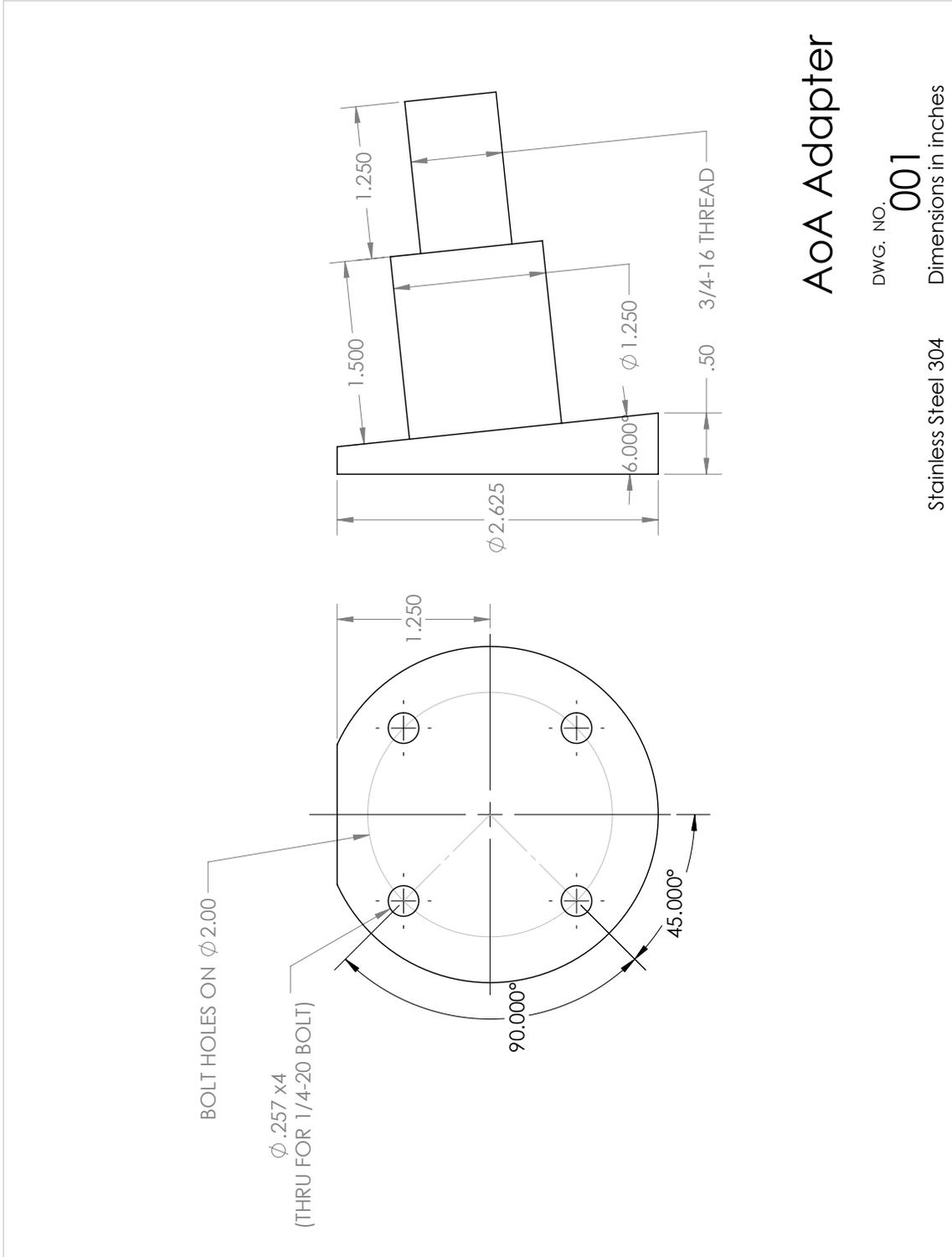


Center Shaft

DWG. NO. 001

Stainless Steel 304 Dimensions in inches

Modular Cone Drawings

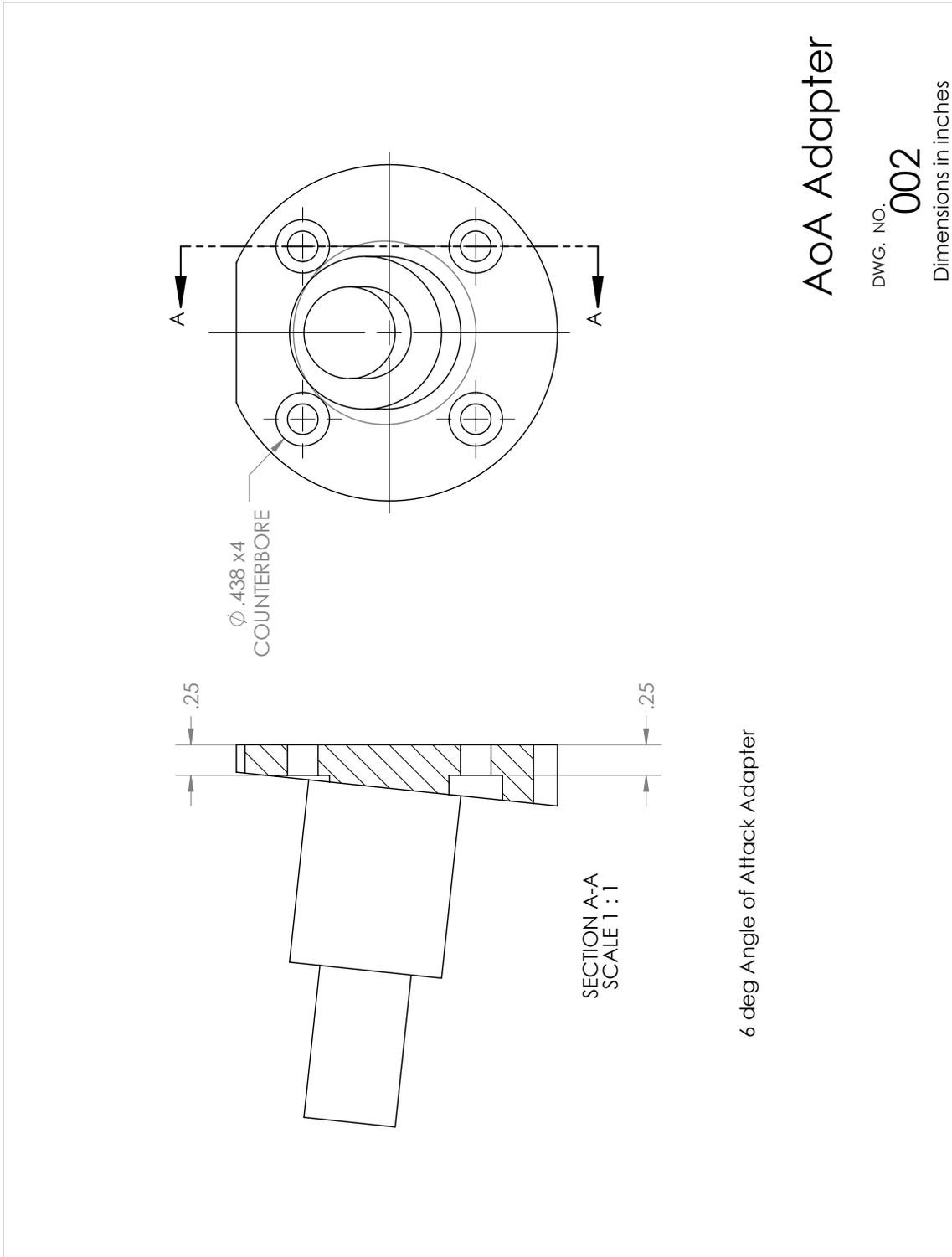


AOA Adapter

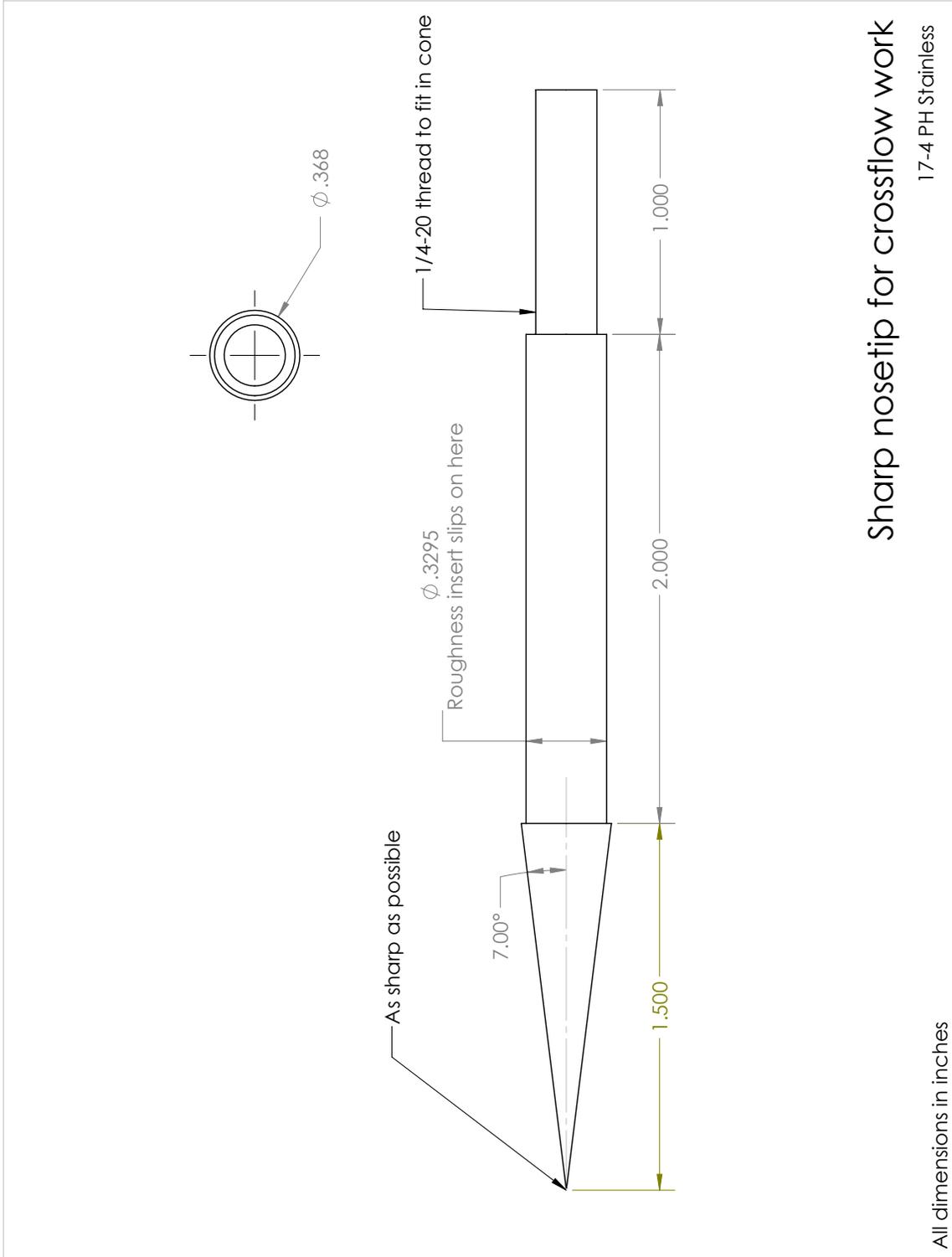
DWG. NO. **001**

Stainless Steel 304 Dimensions in inches

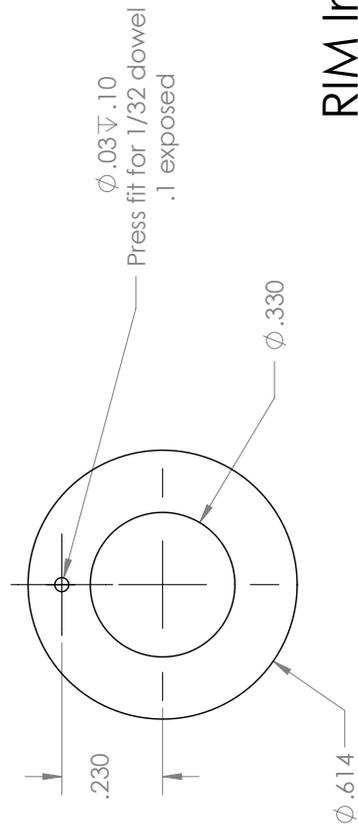
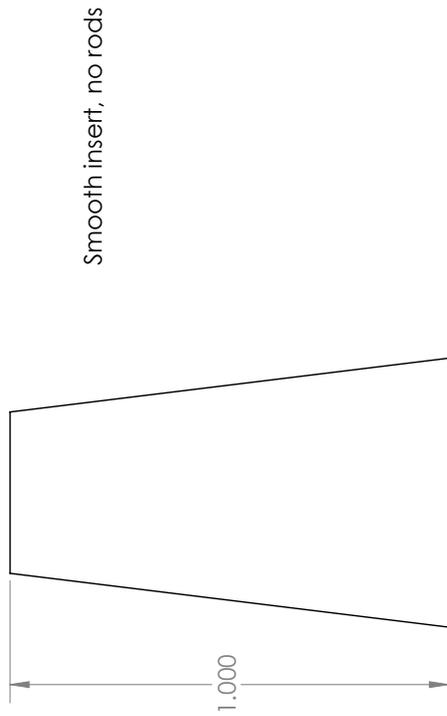
Modular Cone Drawings



Modular Cone Drawings



Modular Cone Drawings



RIM Insert

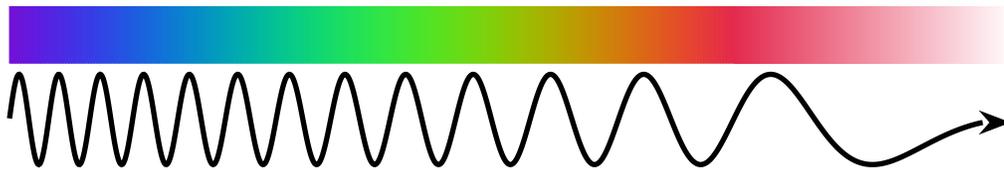
DWG. NO. **001**
AI 6061 Dimensions in inches

I. BAM6QT INFRARED MEASUREMENT SYSTEM HANDBOOK

Handbook on IR Heat Transfer Measurements in the BAM6QT

“Any product that needs a manual to work is broken.”

Elon Musk



Josh Edelman

Rev. 1B, 2019

Revision	Date	Author(s)	Description
1	4/25/2019	JBE	Initial writing.
1B	6/19/2019	JBE	Minor revisions, end of Edelman ownership.

Forward

This handbook will attempt to provide all the necessary information for an experienced tunnel operator to successfully acquire IR images and reduce them to heat transfer. The first chapter is a quick reference, which provides useful data on the properties of PEEK, the IR camera and windows, and an FAQ, among other things. This should be the first reference for any quick question on the particulars of the system. Chapter 2 provides an overview of the theory and history of IR thermography, and a description of the theory and numerical implementation of the various heat-transfer reduction algorithms included in the Appendix. Chapter 3 describes best practices for IR model design and how to prepare models for IR measurements. Chapter 4 explains how to set up the IR camera and prepare it for use. The .m files for the codes will be placed somewhere on the group website.

I have spent considerable time learning to use the apparatus and preparing this overview. However, my efforts would be in vain without help from a number of other researchers. Salvatore Cerasuolo, Mirko Zaccara, and their advisor, Dr. Gennaro Cardone, led the development of an IR technique for the BAM6QT. I borrow heavily on their expertise in this handbook. Dr. Matt Borg of AFRL provided the calcium fluoride window, and his significant experience in IR imaging was extremely helpful. Dr. Tom Juliano, Harrison Yates, and Carson Running at the University of Notre Dame graciously loaned their IR camera to us for several years before we acquired our own. The IR camera itself was purchased using funds from the UDRI contract. Finally, I am grateful for the support of the AFOSR via the NDSEG fellowship while preparing this document.

I expect that there will be several revisions as aspects of the system change, as people learn more tips and tricks, and as errors are discovered within. The version history should be kept up to date by whomever is responsible for its upkeep.

Josh Edelman, 2019

Contents

1

Quick Reference	
1.1 PEEK Properties	7
1.2 IR Camera Properties	7
1.3 IR Window Properties	8
1.3.1 Original CaF ₂ Window	8
1.3.2 Small Sapphire Window	8
1.3.3 New CaF ₂ Window	9
1.3.4 MgF ₂ Window	9
1.3.5 Big Sapphire Window	9
1.4 Typical Camera Settings	9
1.5 Typical Process	9
1.6 FAQ	11
1.7 Useful Contact Information	12

2

IR-capable Model Design	
2.1 Meeting the Theories' Assumptions	13
2.1.1 1D or 2D Heat Transfer	13
2.1.2 Semi-infinite Boundary Condition	14
2.1.3 Adiabatic or Isothermal Boundary Condition	14
2.1.4 Constant-Frequency Time Data	14
2.1.5 Physical Model Design	15
2.2 Preparing a Model For Imaging	15
2.3 Use of Sensors	15

3

Setting Up The Camera	
3.1 Installing the Window	16
3.1.1 IR Assembly Installation	16
3.1.2 IR Assembly Uninstallation	17
3.2 IR Camera Physical Setup	18
3.2.1 Unpacking the Camera	19
3.2.2 Changing the Lens	20
3.2.3 Mounting the Camera on the Optical Table	21
3.2.4 Connecting Cables to the Camera	27
3.2.5 Setting Up the Interface Hardware	30
3.2.6 Turning on the Camera	30
3.3 Optical Calibration	34
3.3.1 Performing the Calibration	34
3.3.2 Calibration Plate Spacer Color Codes	36

3.4	Uninstalling the Camera	36
3.4.1	Removing the GigE Adapter	37

	Using the Camera	
4.1	IRBIS Software	38
4.2	Acquiring Data	45
4.3	Exporting Data	45
4.4	Disconnecting the Camera	46



Safety Warning

THIS IS A SAFETY WARNING. IT PROVIDES SAFETY CRITICAL INFORMATION. FAILURE TO HEED SAFETY WARNINGS COULD RESULT IN SERIOUS INJURY OR DEATH.



Warning

This is a warning. It will alert you to particularly important information, common mistakes, and ways you can seriously damage equipment. Please heed all warnings.



Before Continuing

This is a stop sign. It will tell you what you need to have completed to proceed to the next step.



For Your Information

This is a tip. It provides and emphasizes useful information for your IR journey!

A Note on Citation

Whenever discussing the Purdue IR method, please cite the Masters theses of Cerasuolo and Zaccara (see citations in Edelman's dissertation), and cite Borg & Kimmel (AIAA 2016-0354) when discussing the calcium fluoride window. If you reference the original QCALC routine, also cite Borg & Kimmel. If you use PIRANHA or the `qcalc` codes, cite Edelman's PhD dissertation.

Chapter 1

Quick Reference

“To my sister’s eyes, there is nothing which cannot be explained if one has access to a proper reference library.”

Elizabeth Gilbert

1.1 PEEK Properties

Specific heat c_p , J kg ⁻¹ K ⁻¹	1026
Thermal Conductivity k , W m ⁻¹ K ⁻¹	0.29
Density ρ , kg m ⁻³	1300
Thermal Diffusivity α , m ² s ⁻¹	0.217×10^{-6}
Emissivity, ϵ , function of viewing angle θ	$[0.91\cos(\theta)]^{0.03/\cos(\theta)^{1.35}}$
Physical Specs	Victrex 450 G unfilled, natural (tan) color

1.2 IR Camera Properties

Model Number	Infra-tec ImageIR 8300 HP
Spectral Range	2.0 μm to 5.5 μm
Pixel Pitch	15 μm
Detector	InSb
Detector Resolution	512 \times 640
Temperature Measurement Range	-40 °C to 1500 °C
Measurement Accuracy	± 1 °C
Temperature Resolution	20 mK
Frame Rate, full frame	max 350 Hz
Dynamic Range	16 bit
Integration Time	0.6 μs to 20 000 μs
Lenses	12 mm, 25 mm, 50 mm, close-up lens for 50 mm
Dimensions	244 \times 120 \times 160 mm
Weight	3.3 kg

1.3 IR Window Properties



Warning

The IR window is extremely fragile and difficult to clean. Do not touch the surface of the window or any of the camera optics. Wear gloves when handling the window. Do not let the camera lens touch the window at any time.



Safety Warning

THE NEW CAF₂ AND MGF₂ IR WINDOWS HAVE NOT YET BEEN PRESSURE RATED, AS OF REV 1. DO NOT USE THESE WINDOWS UNTIL THEY HAVE BEEN RATED.

1.3.1 Original CaF₂ Window

The Calcium Fluoride (CaF₂) window is designed to fit inside the downstream porthole window (designed and paid for by Matt Borg, AFRL). The following properties are those of the glass window itself, not the steel frame.



Safety Warning

THE CAF₂ IR WINDOW IS ONLY RATED TO A MAWP 285 PSIG, AS OF REV 1. DO NOT USE THE IR WINDOW ABOVE THIS PRESSURE UNLESS IT HAS BEEN RE-RATED.

Diameter, mm	101.4
Thickness, mm	44.45
AR Coating	BBAR-CAF ₂ -3-5
Manufacturer	ISP Optics

The two flat surfaces and the curved outer surface are polished. There is a small bevel on the edges.

1.3.2 Small Sapphire Window

The small sapphire window was fabricated to the same physical specifications as the CaF₂ window. It fits in a similar frame, designed and manufactured by TriModels.



Safety Warning

THE SMALL SAPPHIRE IR WINDOW IS RATED TO A MAWP 285 PSIG, AS OF REV 1B. DO NOT USE THE IR WINDOW ABOVE THIS PRESSURE UNLESS IT HAS BEEN RE-RATED.

Diameter, mm	101.4
Thickness, mm	44.45
AR Coating	SAR-3,000-2S
Manufacturer	Guild Optical
Optical Coating	IMPhotonix

1.3.3 New CaF₂ Window

The new CaF₂ was purchased with AFOSR funds in the Spring of 2019.

Diameter, mm	101.4
Thickness, mm	44.45
AR Coating	AR/AR 2–6 μm
Manufacturer	Knight Optical

The two flat surfaces and the curved outer surface are polished. There is a small bevel on the edges.

1.3.4 MgF₂ Window

Magnesium Fluoride (MgF₂) is stronger than CaF₂. A test window was purchased with AFOSR funds in the Spring of 2019.

Diameter, mm	101.4
Thickness, mm	44.45
AR Coating	AR/AR 2–6 μm
Manufacturer	Knight Optical

The two flat surfaces and the curved outer surface are polished. There is a small bevel on the edges.

1.3.5 Big Sapphire Window

Money for a large sapphire window was awarded as part of the FY19 DURIP grant. When the window is fabricated, the specifications will go here.

1.4 Typical Camera Settings

Integration Time	1290 μs
Lens Focal Length	12 mm
Frame Rate	300 Hz

1.5 Typical Process

1. Install IR window in BAM6QT (Section [3.1](#))
2. Set up IR camera and associated hardware (Section [3.2](#))
3. Perform optical calibration, if needed (Section [3.3](#))
4. Install model
5. Take data (Chapter [4](#))
6. Export data as ASCII .asc file (Section [4.3](#))
7. Apply corrections to optical calibration to match image
8. Register images and convert to temperature
9. Perform heat transfer reduction

10. Save results
11. Perform post-entry optical calibration, if needed
12. Put camera and IR window away

1.6 FAQ

Safety of People and Equipment

Q: It looks like there's a chip in the window. Is it OK to use?

A: **NO.** You should never use a damaged window.

Q: Oh no! I accidentally smudged one of the IR optical surfaces! What do I do?

A: Don't panic, **but under no circumstances should you clean the IR optics by yourself unless you are specifically trained to do so. Never use acetone or ethanol near the optics.** Find the IR Lead and they will help you to clean the optics or find someone who can.

Lenses

Q: How do I know what lens to use?

A: Usually you only need the 12 mm lens. If you want to zoom in on some feature of the model, use the 25 or 50 mm lens.

Q: What's that funny circle in the middle of my image?

A: This is the reflection of the camera sensor off the window. You can reduce it by angling the camera with respect to the plane of the window.

Q: How do I focus the lenses?

A: With the lens already mounted on the camera, hold the forward-most ring of the lens and gently twist it. Use the right-hand rule to determine the direction: to focus closer, twist out, etc. **Note: do not force the focus ring to rotate. It has a mechanical stop at its limits.**

Data

Q: Where should I save the data?

A: You may save data during your entry to the DATA drive on the laptop (the 1TB drive), or to an external drive. However, if you save to the internal drive, **you must remove your data from the laptop after your entry is over.** IR images take a lot of space, so we need to keep the disk as empty as possible. The IR Lead has the authority to delete any data on the disk if it begins to get too full.

Q: The output from the camera shows that it is taking frames before it is triggered. What is going on?

A: The camera should be set up to record 1 second of pre-run data (see Section 4.1), so it must always be collecting and deleting data before it receives the trigger signal.

IRBIS Software

Q: What is the bright red box in the bottom left corner of the screen?

A: That is the current camera temperature. If it is red, the camera is far too hot. Make sure the room is cool, and open the optical table (only if the tunnel is unpressurized). Wait until the camera cools down to use it further.

Q: Why aren't my settings saving?

A: Remember to click the **green check mark** to close options windows and save the settings. Clicking the x mark reverts the settings.

Q: The software says I exceeded my calibration range. Is this bad?

A: You should always try to select a calibration range that completely encompasses your data. However, if you exceed it by a few degrees, the calibration should still be valid (according to InfraTec).

1.7 Useful Contact Information

Gary Schivley, InfraTec sales rep: G.Schivley@infratec.de

BAM6QT IR Lead: jedelma@purdue.edu

Jon Lavoie, Guild Optics (sapphire optics): jlavoie@guildoptics.com

Michael Maldari, IMPhotonix (optical coatings): mike.maldari@imphotonix.com

Boedeker Plastics (PEEK supplier): www.boedeker.com

Chapter 2

IR-capable Model Design

“You cannot defend your design without knowing what you’re designing for.”

I. M. Pei

Infrared-capable models have a number of advantages over those used with TSP. They are electrically insulating, so PCB ground-looping is less of an issue. The surface roughness and any steps in the model stay constant from entry to entry. And the model can be cleaned during an entry if it becomes dirty. However, for all its advantages, designing models for use with the IR system is substantially more involved than for other measurements. This chapter describes the necessary steps one must take for a successful model.

2.1 Meeting the Theories’ Assumptions

As described in Edelman’s dissertation, all of the heat transfer reduction codes available for use make some assumptions. To get accurate heat transfer data, it is essential that the model is designed to maintain the validity of the assumptions of any particular reduction routine. This section describes how to design a model to accommodate each assumption.

2.1.1 1D or 2D Heat Transfer

Whether the assumption of 1D or 2D heat transfer is valid depends almost entirely on the geometry in question. If there are no large gradients in heat transfer in any direction (such as a straight cone at zero degrees AoA), then 1D is valid. However, sometimes even streaky flows (like crossflow or Görtler) can make the 1D assumption and still yield largely accurate results.

The relevance of 2D conduction effects in streaky flows can be quantified via the Modulation Transfer Function (MTF) and the modified Fourier number [?]. This function is given as

$$F = \frac{\sqrt{\pi} \operatorname{erf}(\sqrt{\operatorname{Fo}_\omega})}{2 \sqrt{\operatorname{Fo}_\omega}}, \quad (2.1)$$

where Fo_ω is the modified Fourier number,

$$\operatorname{Fo}_\omega = \omega^2 \alpha t, \quad \omega = 2\pi/\lambda, \quad (2.2)$$

where λ is the wavelength of the relevant streaks. On a cone, the wavenumber of the stationary cross-flow waves is given by $m = 360^\circ f$, where f is the frequency of the waves in cycles/degree. If we let $\beta = \sqrt{\alpha t} m/r(x)$, then the MTF is given by

$$F = \frac{\sqrt{\pi} \operatorname{erf}(\beta)}{2 \beta}. \quad (2.3)$$

The MTF reaches 95% (i.e., 2D conduction will have about a 5% effect on the heat transfer) at $\beta_{95} = 0.39$. Thus for a cone, if

$$\beta_{95} = \sqrt{\alpha t} m/r(x) < 0.39, \quad (2.4)$$

$$\Rightarrow x > 2.6\sqrt{\alpha t} m/\tan \delta \quad \text{or}, \quad (2.5)$$

$$\Rightarrow m < 0.39x \tan \delta/\sqrt{\alpha t}. \quad (2.6)$$

then 2D conduction can be neglected. For PEEK in BAM6QT conditions on a 7° half angle cone, these conditions mean that 2D conduction can be neglected if

$$x > 0.0137m \quad (\text{for } t = 2s), \quad (2.7)$$

$$m < 73x. \quad (2.8)$$

In general, it is unlikely that 2D conduction effects will matter for most of the models used in the BAM6QT. If 3D effects are relevant (wall normal, transverse, and downstream conduction), then none of the currently available algorithms will be effective. However, 3D effects will only matter if there are significant temperature gradients in every direction, or if the material is non-isotropic. It is up to the experimenter to determine if such conditions are relevant to a specific geometry.

2.1.2 Semi-infinite Boundary Condition

Both the `fft` and direct/indirect codes can make the assumption that the farfield boundary condition (i.e., not the surface) is semi-infinite (technically the Italian code assumes an isothermal farfield condition, but for a large mesh size this is equivalent to a semi-infinite assumption). To meet this condition, the thermal penetration depth for a given material and run time must be smaller than the thickness of the PEEK at every imaged location. The thermal penetration depth is $\delta(t) = 4\sqrt{\alpha t}$ where α is the material thermal diffusivity and t is the time into the run. For PEEK at 3 seconds, δ is about 3 mm. PEEK thicknesses larger than 5 mm should generally be acceptable.



For Your Information

Note that the 2D algorithm also requires that the radius of the PEEK section is much larger than δ , the thermal penetration depth.

2.1.3 Adiabatic or Isothermal Boundary Condition

The direct and indirect codes allow either an adiabatic or isothermal backside boundary condition. The `fft` code would require significant changes, but it is possible to implement this condition if needed. To meet the specified back-face condition, either the internal surface of the PEEK must be surrounded by stagnant air (roughly adiabatic) or directly touching a large heat-sink of metal (roughly isothermal). Note that the semi-infinite boundary condition in the direct and indirect codes is implemented as an isothermal condition at $T = T_{initial}$.



Warning

It is possible to have other boundary conditions on the actual model than the three discussed here. The relevant boundary conditions should always be considered so they can either be accounted for or properly explained.

2.1.4 Constant-Frequency Time Data

The `fft` method assumes that the frame rate of the camera is constant (i.e., no dropped frames). If the camera has dropped frames, these time points will need to be interpolated before using the method.

However, preliminary testing seems to indicate that a few dropped frames, or a small variation in the frame rate, does not have a significant effect on the resulting heat transfer calculation. You may use the `fft` method in this case, but accurate results are not guaranteed.

2.1.5 Physical Model Design

It is necessary to work with the machinists to ensure that the PEEK components are the correct size. PEEK is technically dimensionally stable, but it expands quite a bit when hot. It may be useful to add a few thousandths to the longitudinal direction of a part to ensure that there are no gaps when it cools. In addition, steps can form at interfaces if the PEEK is too large in diameter and then expands in the tunnel. There is no good way to ensure the model interfaces will mesh perfectly a priori, so it is very important to communicate with the shop what the requirements are.



For Your Information

A note on part interfaces: where ever PEEK parts mate with aluminum or steel, there will be heat conduction out of the PEEK, and therefore the 1D assumption may not be strictly valid.

2.2 Preparing a Model For Imaging

Once the model has been fabricated there is not much left to do to prepare it for an entry. First, the model must be cleaned of any smudges (anti-seize is a common culprit). The PEEK can be readily cleaned with acetone. If the acetone is not strong enough, the shop also has a mineral-spirits wash station that should be more effective. To make registration or other marks on the model, you must use the Sakura silver-ink calligraphy pen, or some similar metallic pen. Regular Sharpie will not show up in IR (though the silver Sharpie does show up, but not well). However, note that any registration marks will not be visible during tunnel start-up because of the massive heating of the model. A possible way to have registration marks visible during the start-up is to use press-fit metal rods instead of a pen. Such a method should be more accurate in terms of mark placement, but could affect the nature of the flow if the press-fits aren't smooth. This technique has not been attempted so try it at your own risk.



For Your Information

It is recommended to use gloves when handling the PEEK. It is very easy to accidentally get anti-seize everywhere, and it doesn't always clean off well.

2.3 Use of Sensors

It is useful to have an array of sensors in the field of view of the camera, to facilitate image registration during processing. However, both PCBs and Kulites heat up the PEEK when they are turned on. It is usually necessary to make a run with the sensors turned off or rotated out of view to get useful IR data.

Chapter 3

Setting Up The Camera

“By failing to prepare, you are preparing to fail.”

Benjamin Franklin



Safety Warning

ALWAYS SECURE THE LEXAN SAFETY SHIELD OVER THE OPTICAL TABLE BEFORE PRESURIZING. THE CAF2 WINDOW IS QUITE BRITTLE AND IF IT FAILS, IT WILL FAIL EXPLOSIVELY.

3.1 Installing the Window

The following is a set of instructions for installing the IR window and its frame into the porthole window frame. They were provided by Matt Borg, who designed the IR window, and have been lightly edited for clarity. Note that it is best to install and uninstall the IR assembly while the porthole window frame is installed in the tunnel.



Safety Warning

ALWAYS VISUALLY CHECK THE IR WINDOW FOR DEFECTS (CHIPS, SCRATCHES, SCUFFS, ETC.) BEFORE INSTALLING IT IN THE TUNNEL. IF THERE ARE ANY DEFECTS IN THE GLASS, DO NOT USE THE WINDOW AND REPORT IT IMMEDIATELY.

3.1.1 IR Assembly Installation

1. The IR window is designed to fit **only in the downstream porthole location**.
2. Remove the 8x bolts from the downstream porthole (see Figure 3.1).
3. Remove the retaining ring and the soft-metal gasket.
4. Remove the Plexiglas window (use nitrile gloves to avoid smudging the Plexiglas). Make note of the correct orientation so it can be reinstalled later (the top surface is marked).

5. Remove the 2x dowel pins from the top and bottom of the porthole window frame.
6. Ensure the o-ring is properly seated in the frame.
7. Carefully install the IR window assembly in the porthole frame. **Do not touch the window itself. Use gloves when handling the assembly.** The dowel pin on the IR assembly should be oriented at the **top**.
8. Install the soft metal gasket and retaining ring.
9. Install the 8x bolts.
10. Torque the bolts to 100 in-lbs in the following steps: 30 in-lbs, 60 in-lbs, 100 in-lbs. Tighten them in a star pattern as indicated by the numbers in Figure 3.1. Ensure that all the bolts are tightened by re-tightening them in a circle after finishing the 100 in-lbs star pattern.
11. Check the inside of the tunnel to ensure the window is seated properly. There should be no large steps between the tunnel wall and the IR window frame.

3.1.2 IR Assembly Uninstallation

1. Remove the 8x bolts.
2. Remove the retaining ring and soft metal gasket.
3. Remove the IR assembly. This is generally not particularly easy. **Press only on the IR assembly frame and never press or tap on the IR window itself.** Do not tap or hit the IR assembly. Gently lift the IR assembly from inside the tunnel by pulling on the lip of the frame with a cleanly gloved hand. At the same time, push and pull outwards on the assembly. Once the dowel pin is free, the assembly should be easy to remove.
4. Reinstall the two dowel pins in the porthole window frame.
5. Reinstall the Plexiglas window, noting which face is marked as the top.
6. Reinstall the soft metal gasket and retaining ring.
7. Install the 8x bolts and torque as described in Step 10 of the installation instructions, above, and shown in Figure 3.1.
8. Check the inside of the tunnel to ensure the window is seated in the proper orientation. If there is a noticeable step between the tunnel wall and the Plexiglas, it may be upside down.

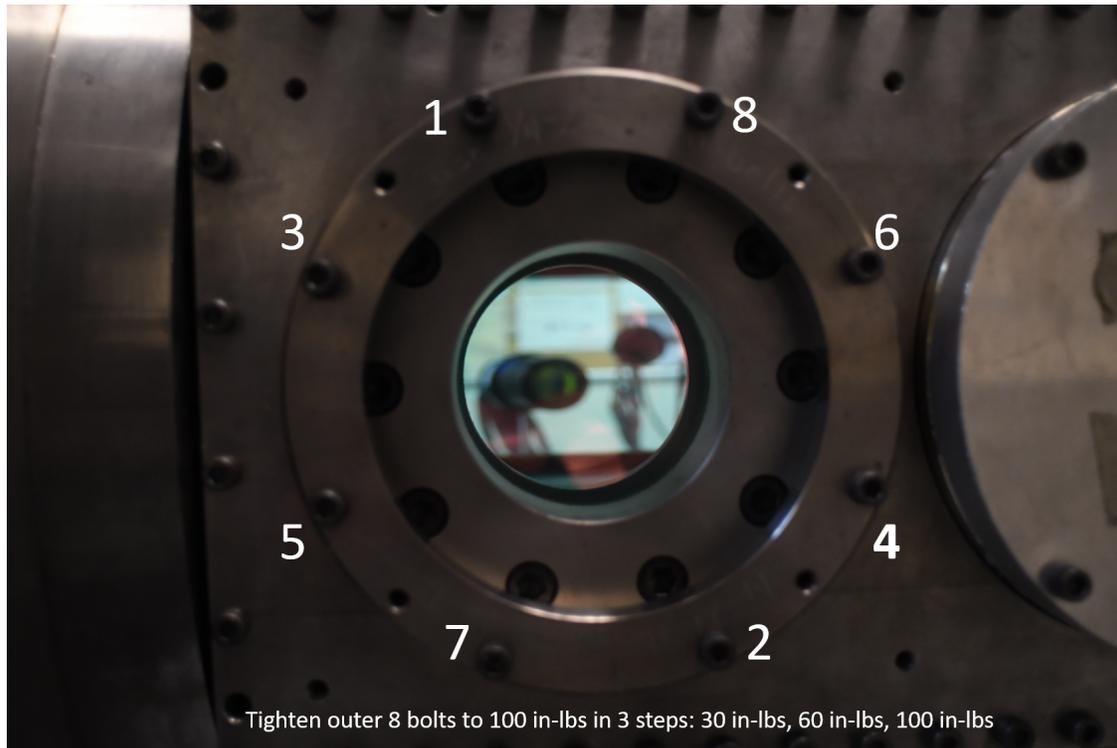


Figure 3.1: The bolt tightening pattern for the IR window.

3.2 IR Camera Physical Setup



Before Continuing

Remember to install the IR window before proceeding to camera setup!



Warning

Be extremely careful to ensure that the camera lens does not touch the IR window surface at any point! **This cannot be overstated.**

Once the IR window has been installed, you can start to set up the camera. You will need

1. The black Pelican case containing the camera and lenses
2. The IR laptop in the laptop bag
3. The green fiber-optic cable
4. The black GigE/USB-C converted box
5. The USB-C cable

3.2.1 Unpacking the Camera

Figure 3.2 shows the layout of the camera and lenses in the Pelican case. Remember this configuration so you can put it back correctly when you are uninstalling. Note that it does not matter what lens is attached to the camera when it is placed in the case, but one lens has to be. Keep track of the rear lens cap for whichever lens is on the camera.

Carefully remove the camera from the Pelican case, as well as the camera powerpack, the trigger hardware, and the fiber optic cable (the cable may be stored separately). Place these on the optical table as demonstrated in Figure 3.3 (the actual location of the items of course does not matter).

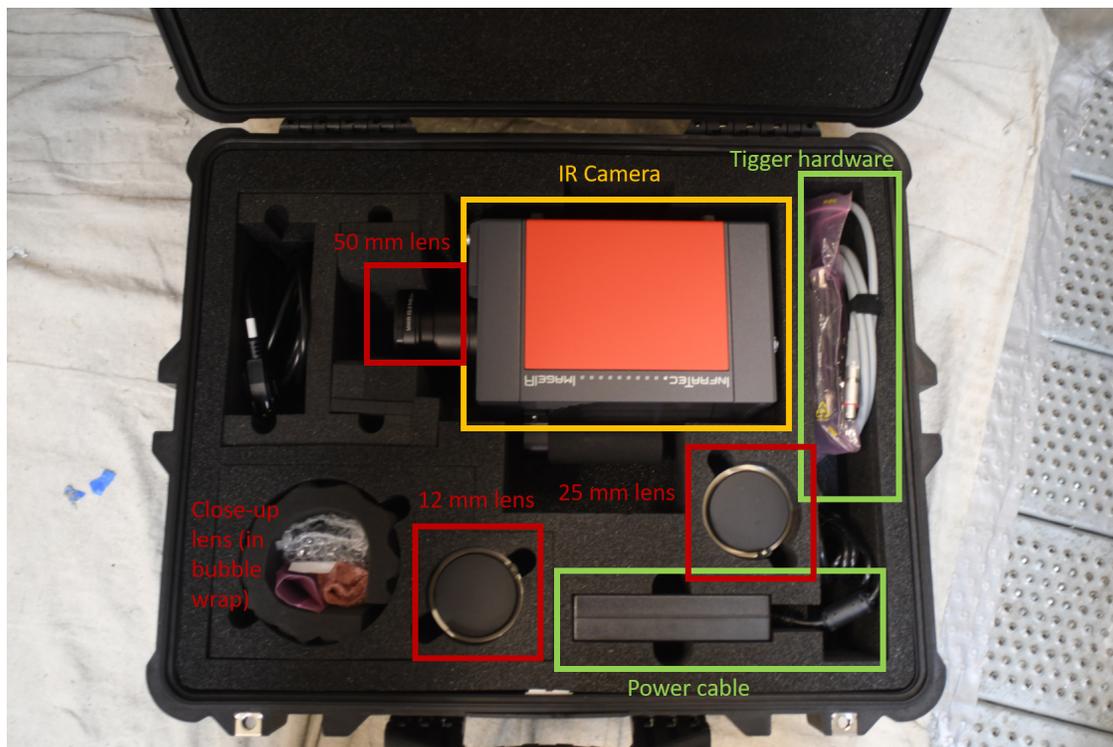


Figure 3.2: The camera Pelican case correctly packed. Note that you can put any lens on the camera when you store it, but one lens does need to be mounted. The close-up lens is **only** for use with the 50 mm lens.

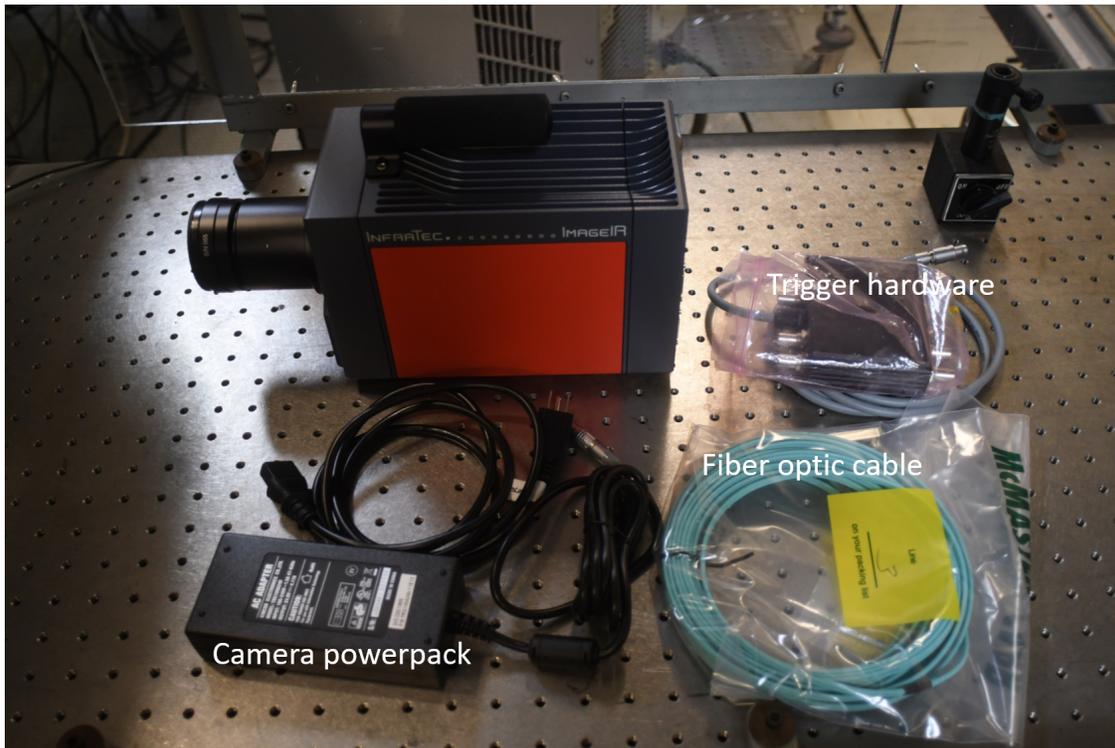


Figure 3.3: The camera and the necessary interface components removed from the Pelican case.

3.2.2 Changing the Lens



Warning

Always move carefully, being aware of where the lenses are at all times so that they are not scratched. It is better to be a little slow than a little careless.

If you need to change the lens, first find the lens you want in the case and take it out. Leave the lens caps on for now. When swapping the lenses, the highest priority is keeping the IR camera sensor free of dust and any other contaminants, so it should be exposed for as little time as possible (though as noted in the warning above, do not move so quickly as to damage the lenses). The priority when changing the lenses is to prevent damage to the camera; lenses are relatively inexpensive. Use the following procedure to swap the lenses:

1. Orient the camera so that you can easily unscrew the lens. It is acceptable to have it facing you and facing perpendicular to you. Do not place it on its rear (the face with the power button).
2. Remove the rear lens cap from the lens you wish to install and place it face down in a clean location, while holding the lens so the rear is pointed down. The rear of the lens is the part with the end with the bare metal threading. The idea is to minimize the possibility of dust falling on the rear element.
3. Carefully grip the rear-most (closest to the camera) barrel of the lens and unscrew it, but pause before fully removing it.
4. In a quick but careful movement, remove the lens, holding it downward so the rear element is covered, and place the new lens on the camera. Screw the new lens in hand tight. **Do not over-tighten the lens.** It should not be so tight that you need any equipment to loosen it.

5. Place the rear lens cap on the lens you have just removed, and put it away in the Pelican case.

3.2.2.1 Choosing the Lens You Need

In general, the longer the focal length, the more zoomed in the resulting image will be. The 12 mm lens is appropriate for most cases in the BAM6QT. If you want to focus on a smaller area to resolve finer features, the 25 mm is a fine choice. The 50 mm should be used for very small features (areas of around 2 inches square), and can be zoomed further using the screw-on close-up lens (areas of around 1 inch square).

3.2.3 Mounting the Camera on the Optical Table



Warning

The mounting plate needs to be tightly screwed into the camera, but be careful not to over-tighten it. You should be able to easily unscrew the mount by inserting a hex screwdriver into the hole in the pylon and twisting.

The camera is mounted by a dove-tail plate to the ThorLabs big mount (in the Lista cabinets where the Conaxes are kept). The mounting plate should generally be set up correctly. In this case, you can (carefully) screw the pylon to the threaded plate embedded in the bottom of the camera until it is tight. The plate should be oriented forward along the axis of the camera, as shown in Figure 3.4.

If the mount does not go to this orientation when it is tight, it needs to be adjusted. You will need to remove the two Phillips-head screws holding the pylon mount to the dove-tail plate. Loosen the hex cap-screw holding the pylon to the plate so the plate can rotate freely but is still attached. Screw the pylon onto the camera and tighten it. When the pylon is fixed, rotate the mounting plate so the slots orient along the axis of the camera and tighten it in place. Now, unscrew the mounting plate and pylon unit from the camera, and reattach to the dove-tail plate as shown in Figure 3.4.

Once the camera and mounting plate have been attached, place it aside. Get the big ThorLabs mount, shown in Figure 3.5, and place it at the marked location on the IR table (or a different location if your situation requires). Ensure that the two metric screws on the dovetail mount are loose so the mount can freely slide in. Also ensure that the lock-release handle is firmly locked before attaching the camera. If you need to adjust the height or rotation of the mount, try to get it as close as possible before the camera is attached. Secure the ThorLabs mount to the optic table before attaching the camera. If you need to move the mount with the camera attached, always keep one hand securely on the camera, and move the mount extremely slowly and carefully.

Before sliding the camera into place, put the black paper shield around the lens as shown in Figure 3.6. This shield prevents external light from reflecting off the window and into the lens.



Warning

Be very careful when the camera is mounted in the dove-tail bracket but not yet tightened. Whenever possible, keep one hand on the camera in this configuration. Be especially careful not to move the camera further forward than you intend, as this could cause the lens to hit the window. Be aware of the distance between the lens and the window at all times.

To fix the camera to the ThorLabs mount, carefully slide the camera dovetail plate into the ThorLabs mounting slot as shown in Figure 3.7. Keeping one hand on the camera at all times, tighten the two metric screws on the mounting bracket so that the camera can no longer slide. These screws should be tight, but again not so tight that you cannot loosen them.

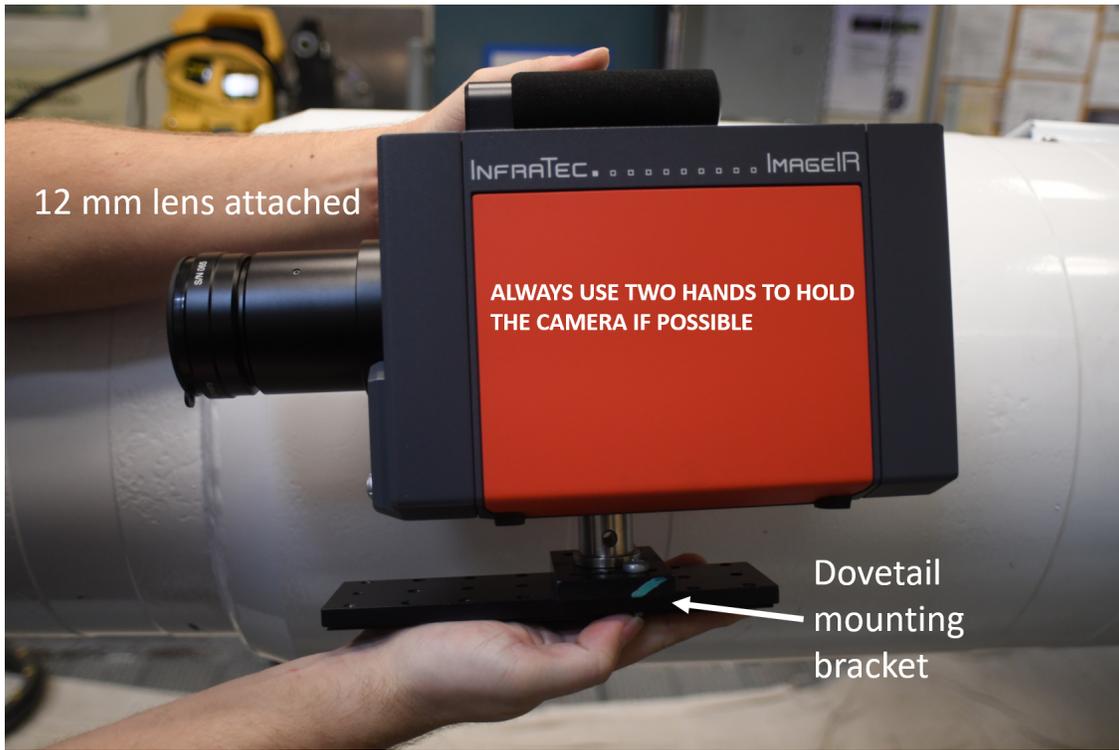


Figure 3.4: The correct mounting of the dove-tail plate.



Warning

To repeat, the screws on the mount are metric. **Do not use an imperial Allen key on these screws, you will strip them.** The proper Allen key is shown in Figure 3.8, though of course any 3 mm Allen key is acceptable.

Another view of the camera properly mounted is shown in Figure 3.9.

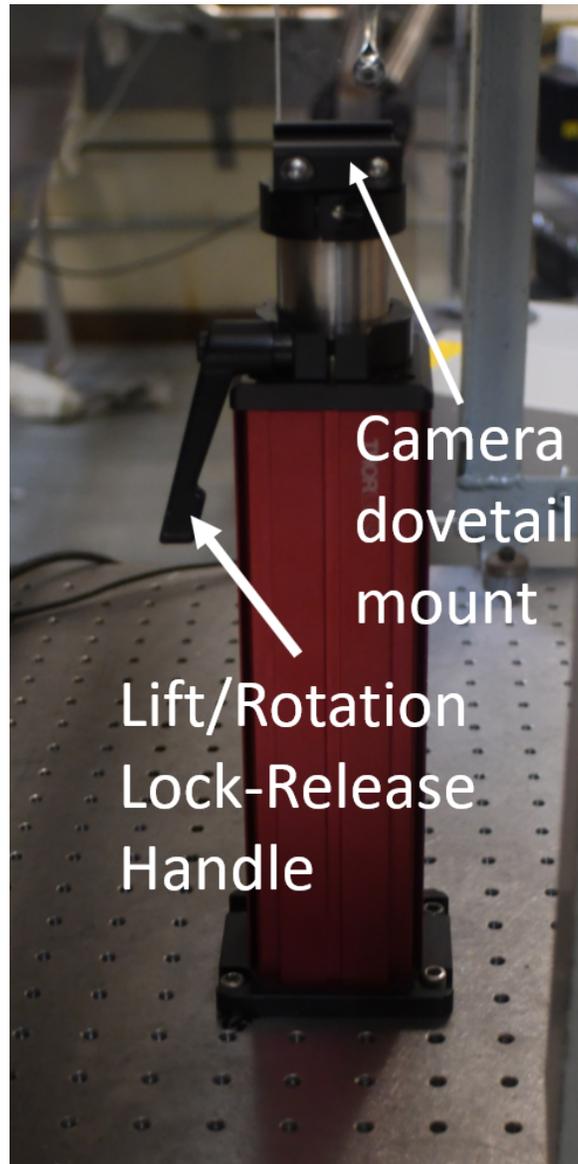


Figure 3.5: The ThorLabs mount. The dovetail mounting plate slides into the top piece, so the two screws should be fairly loose. Make sure the lock-release handle is firmly locked before attaching the camera.

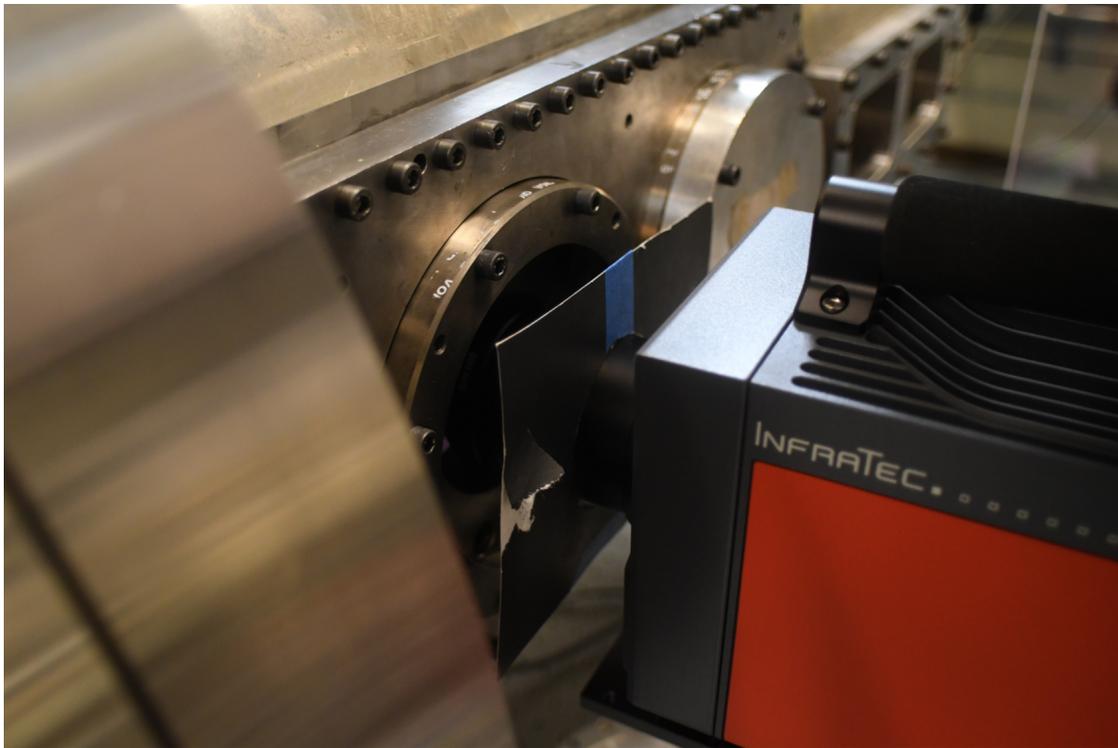


Figure 3.6: The paper shield installed on the IR camera lens.

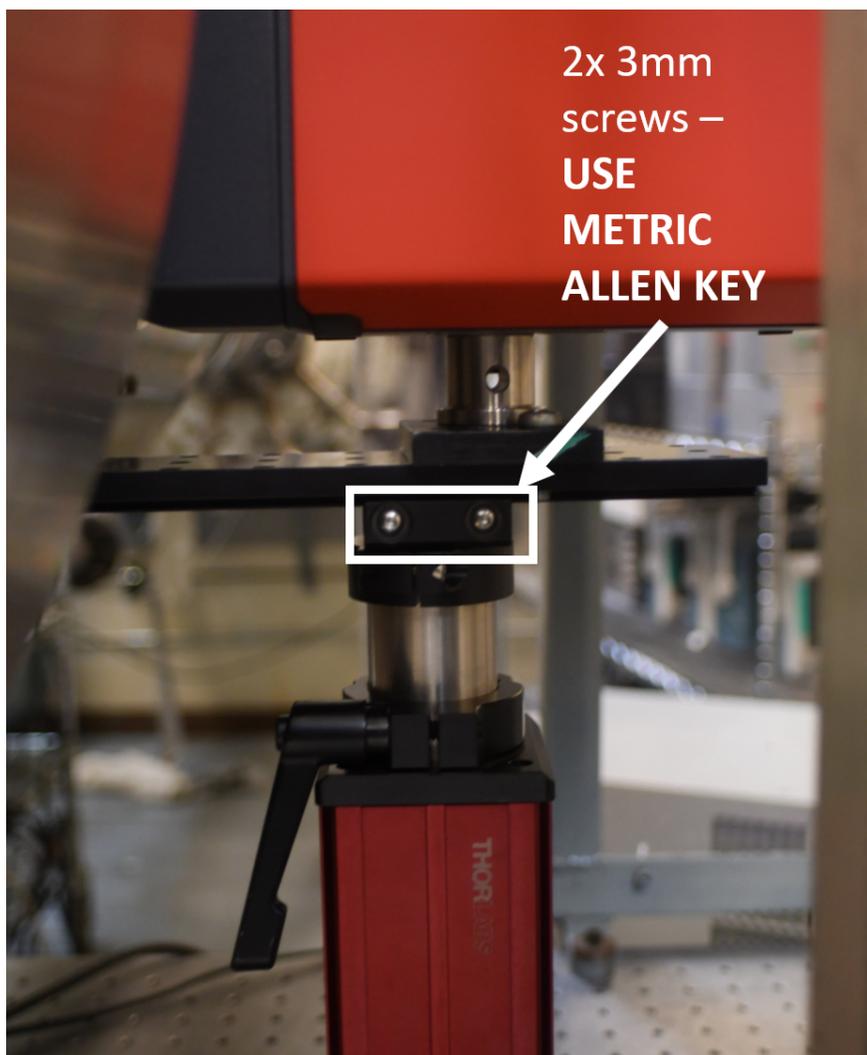


Figure 3.7: The dovetail plate properly inserted into the ThorLabs mount.



Figure 3.8: This is the metric Allen Key, for tightening the two screws on the Thor Labs dove-tail mounting plate. It is your friend. **Do not lose it.**

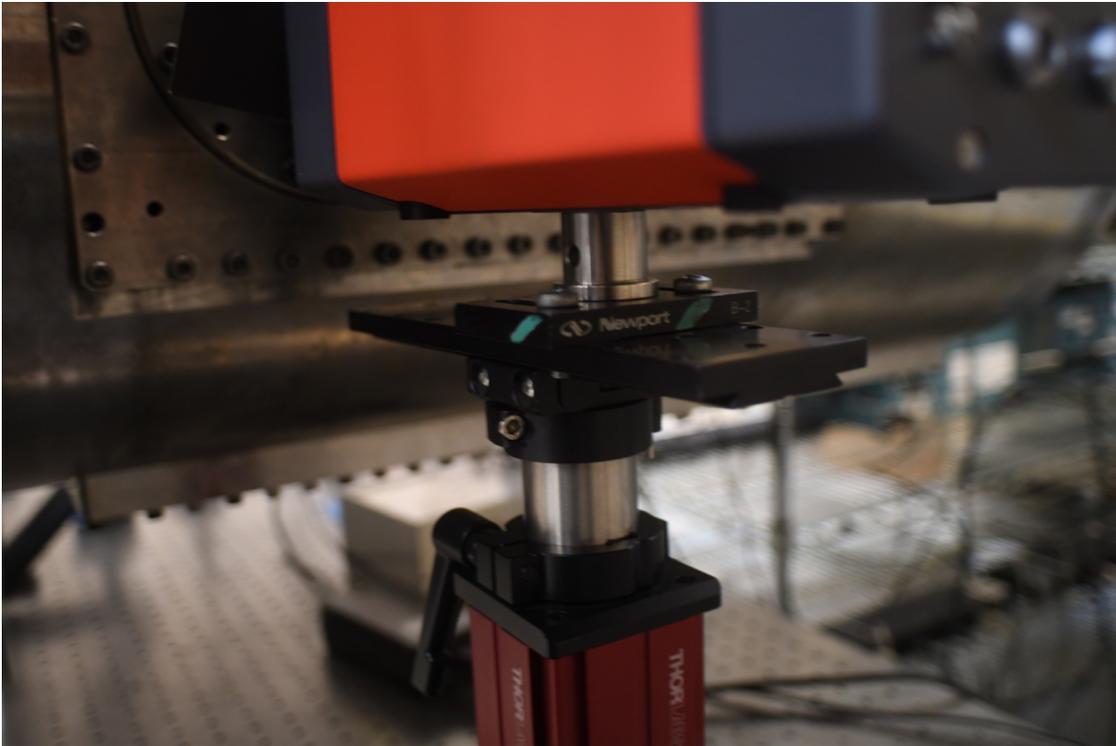


Figure 3.9: A view of the properly mounted camera from the back.

3.2.4 Connecting Cables to the Camera

There are three connections that need to be made to the back of the camera. Be sure to have those cables ready before proceeding:

1. The camera power pack and associated cables
2. The green fiber optic cable with the silver GigE interface attached (it should not be stored unattached)
3. The trigger box



Warning

Never force the cables into the camera. They only fit one way, so if it isn't going in you probably have it oriented incorrectly.

An image showing the relevant ports on the back of the camera is provided in Figure 3.10. It does not matter which order you plug the cables in. The power plug and the trigger cable each have red marks on the cable and the top of the port. For each cable, align the two red marks and push in until the connector clicks into place.

To install the fiber optic cable, first ensure that the cable is mated to the GigE adapter. If it is not, carefully insert the double fiber optic cable into the adapter. Be careful not to scratch or dirty the exposed fiber. The fiber should click into the adapter. Once the GigE adapter is attached, orient it so the white label is facing you, as shown in Figure 3.11. Then gently press the cable into the camera until it locks into place. It may take a small amount of force, so push until it clicks.

Figure 3.12 shows the trigger input box. The inputs are labeled. In general, the input from the oscilloscope (or other trigger) goes into IN1. If you would like to record a pulse when a frame is captured, take that from OUT1. You are certainly free to use additional in and out ports, but they will likely not of any benefit to you. You can configure what the input and output signals do using the IRBIS software (Section 4.1).

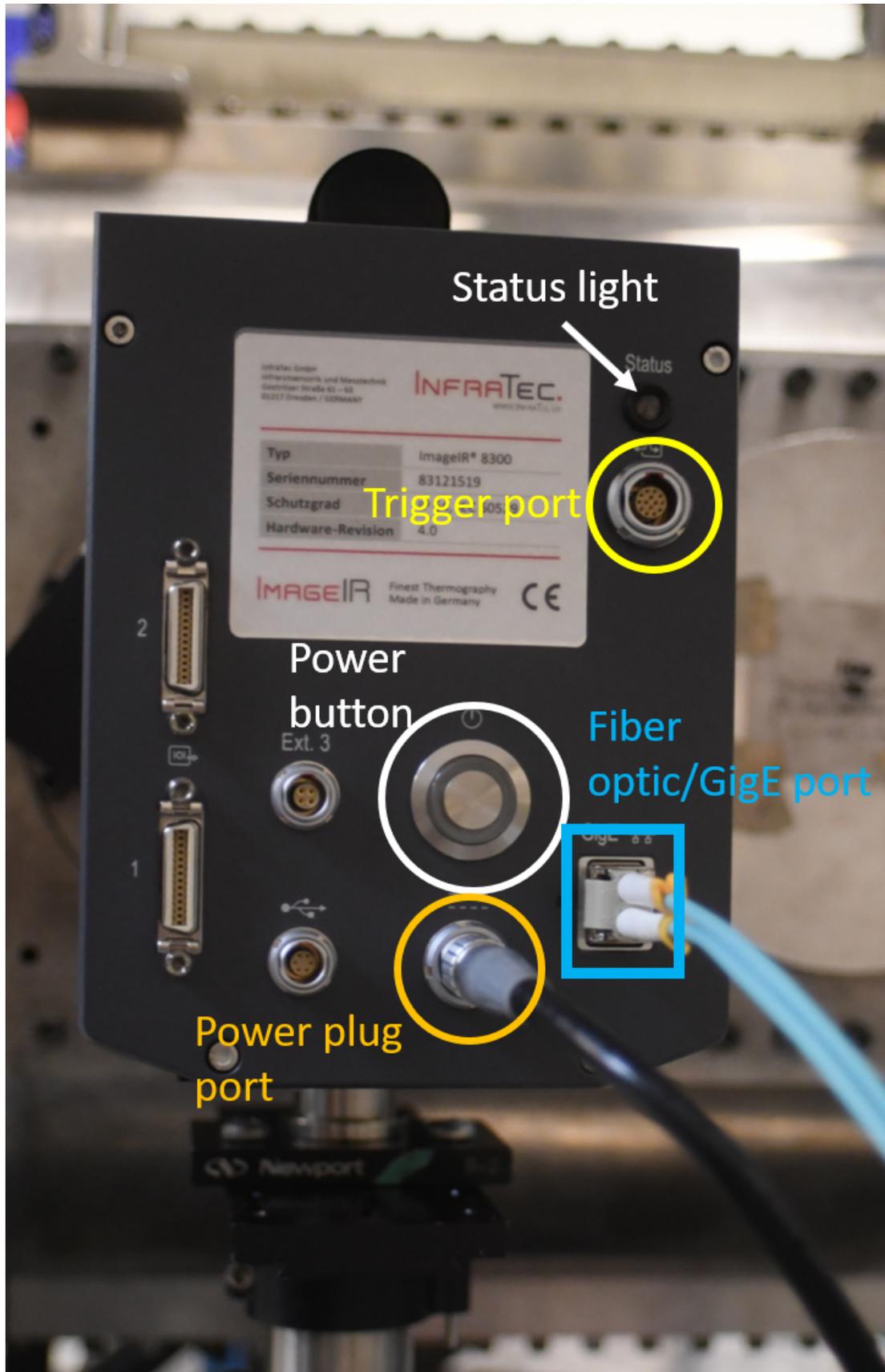


Figure 3.10: The locations of the important ports on the back of the IR camera.



Figure 3.11: The proper orientation for inserting the fiber optic cable into the camera. Note that the white label on the GigE adapter is oriented toward the power button (toward the user as well).



Figure 3.12: The trigger input box and cable.

3.2.5 Setting Up the Interface Hardware

For this procedure you will need the following:

1. The laptop and power pack
2. The converter box (shown in Figure 3.13) and power pack
3. The USB-C cable

Begin by placing the converter box as shown in Figure 3.13. It does not matter in which order you connect the cables to it. Figure 3.14 labels the ports on the back of the box. The two GigE ports on the top map to the two USB-C ports on the bottom, i.e. if you plug the fiber optic into the left port on the top, as shown in the figure, you must plug the USB-C cable into the left port on the bottom. The fiber optic cable and GigE adapter are inserted the same way as was done for the camera side of the connection. The white label on the GigE adapter faces **down** in this case. Be careful not to step on or yank the extra fiber optic as you make this connection.



For Your Information

It is helpful to run the fiber optic cable down the optical table and across the floor, bringing it up only at the desk. Use some duct tape to ensure the fiber optic cable stays attached to the floor and does not become a trip hazard.

Plug the power cable into the back of the box and then plug it in to the wall socket in the way that makes sense. Finally, plug the USB-C cable into the lower port directly under the fiber optic cable, as described above.

You may place the laptop wherever you like, as long as it can easily be reached from the converter box by the USB-C cord, and as long as it does not impede movement around the tunnel. It is helpful to use the yellow or green stools as a table for the laptop, as shown in Figure 3.15. As demonstrated in that figure, plug the USB-C cable from the converter box into the appropriate port on the left side of the laptop.

3.2.6 Turning on the Camera

Once everything is connected, turn on the camera by pressing the power button on the back. It will take a few minutes for the Stirling cooler to warm up. During this time, you will see a blinking blue status light as shown in Figure 3.16. You will also hear a terrifying grinding sound, this is entirely normal. Once the camera is done warming up, the status light will turn green and the grinding sound will become a quieter whir. At this point you can connect to the camera in IRBIS (Section 4.1). If the grinding sound does not go away, turn the camera off and find the IR Lead Student.



For Your Information

The camera will make a terrifying grinding sound when warming up. This is normal. If the camera continues to make the same sound after the status light has turned green, turn it off and find the IR Lead Student.

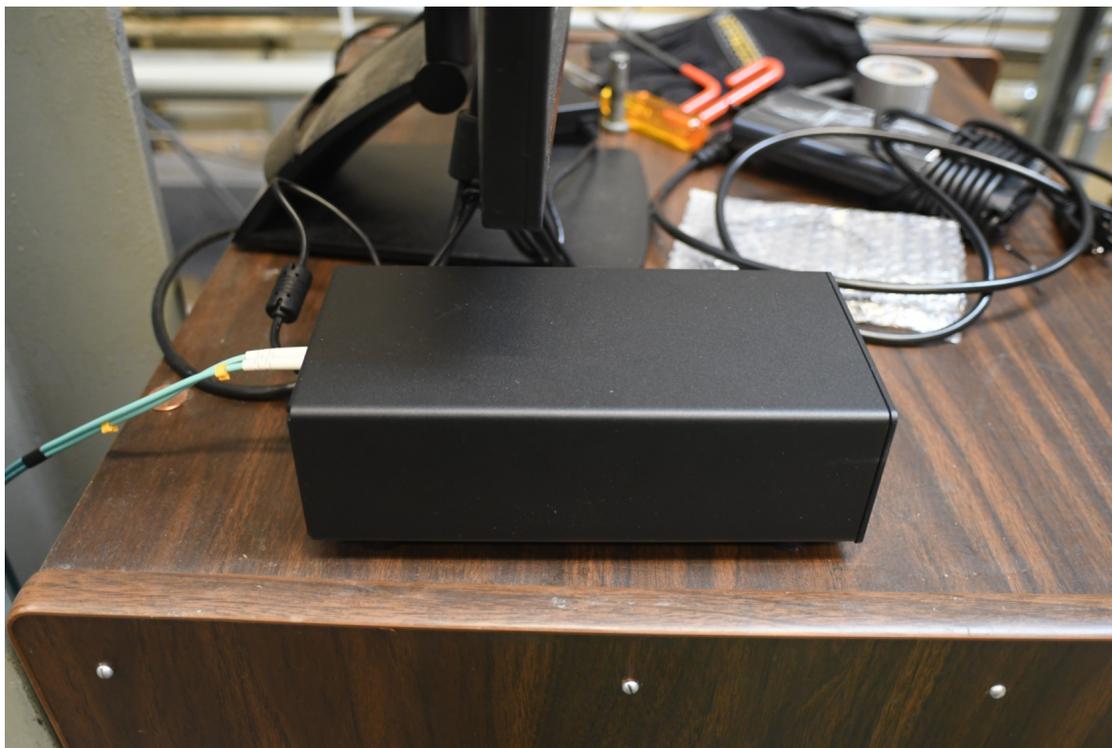


Figure 3.13: The GigE to USB-C converter box.



Before Continuing

Before fully installing your model, especially if you are not performing an optical calibration, put your model in the tunnel to verify that it is within the camera's viewing angle and that the camera is focused correctly. You may have to focus the lens. Hold the forward-most ring of the lens and gently twist it. Use the right-hand rule to determine the direction: to focus closer, twist out, etc. **Note: do not force the focus ring to rotate. It has a mechanical stop at its limits.** If you need to adjust the viewing angle of the camera, you can unlock the ThorLabs mount to raise or lower it (or adjust the lateral angle), or you can unscrew the ThorLabs mount from the table to adjust its lateral position. **When you are adjusting the position of the camera, on any axis, you must always securely hold on to the camera with one hand. If you need help, ask another student. And as always, be aware of the position of the lens with respect to the IR window.**

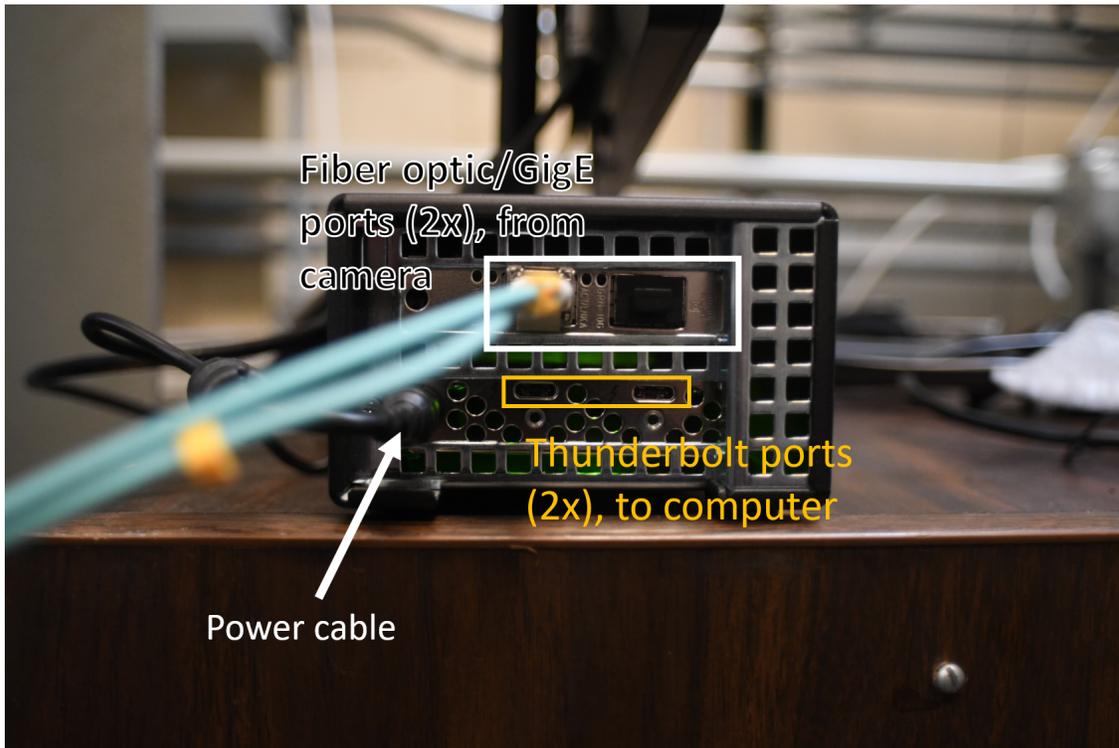


Figure 3.14: The rear ports on the converter box.



Figure 3.15: The laptop connected to the converter box.

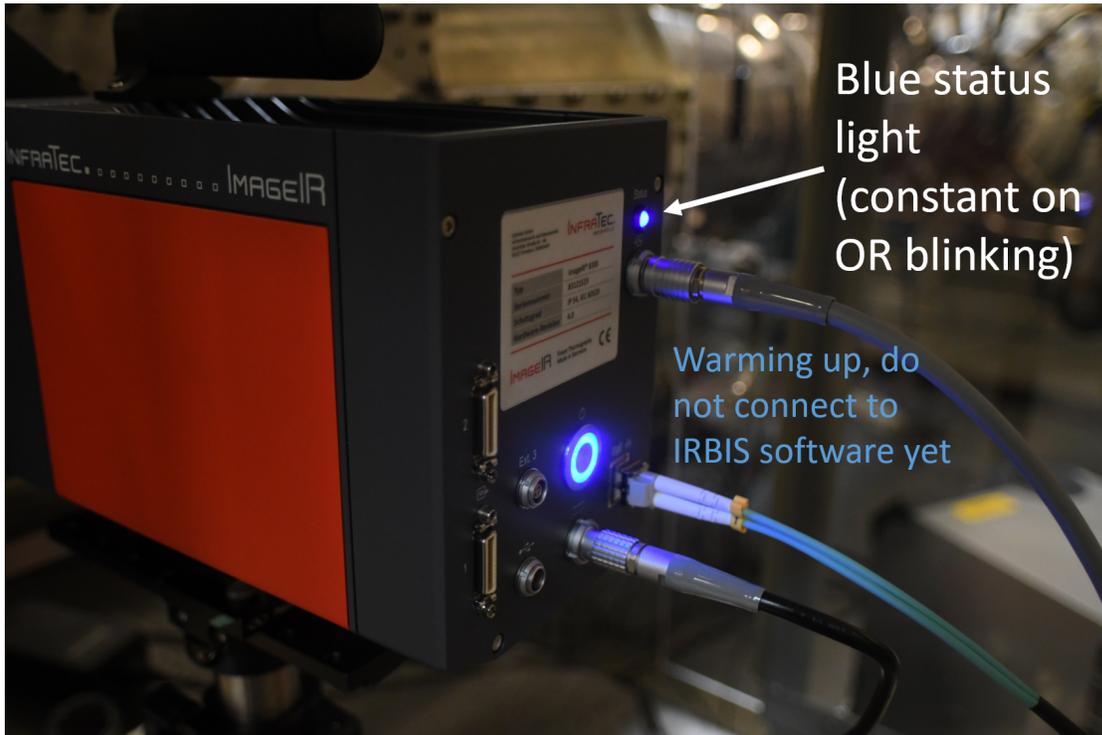


Figure 3.16: The camera shows a blue status light when warming up.

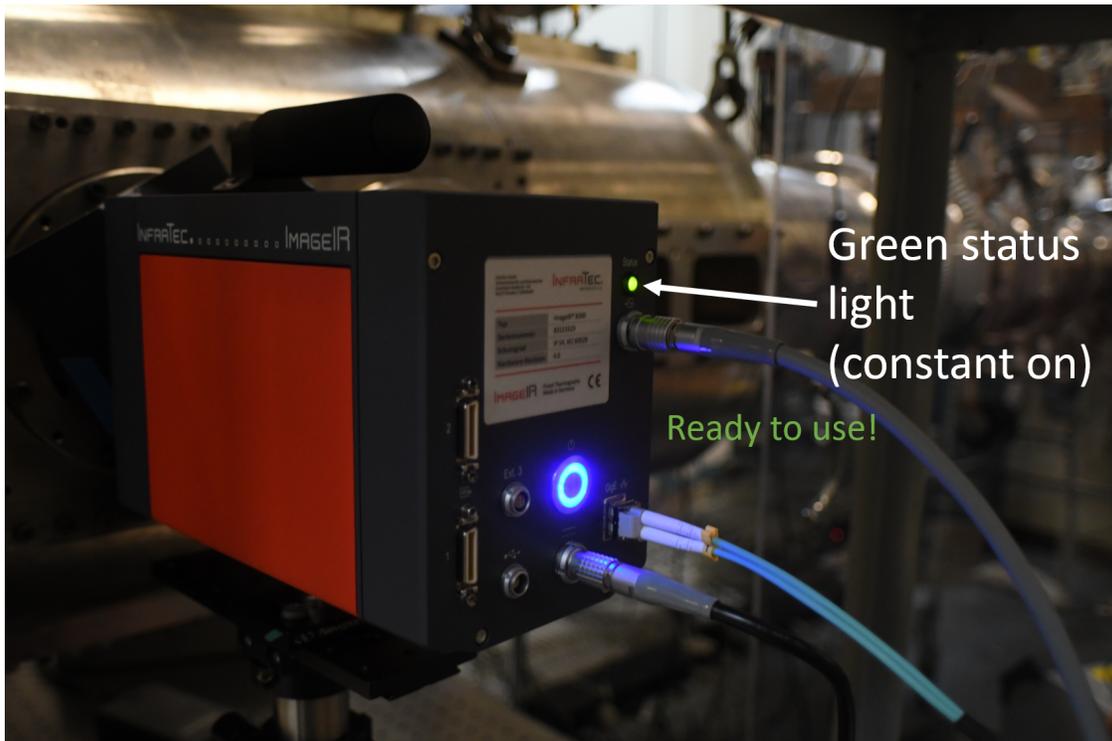


Figure 3.17: The camera shows a green status light when it is ready to use.

3.3 Optical Calibration

This section is optional. It is helpful if you can easily represent your model geometry using simple functions or a point cloud. If this is not the case, you can just process the IR images without associating the individual pixels with physical coordinates. If you do not need to perform the calibration, skip to the next chapter. Either way, be sure to tape the black paper shield around the IR window before starting to take images.

The optical calibration process creates a digital camera “model,” which can be used to convert image coordinates (in pixels) to real-world coordinates (in meters, or whatever unit you like). The process works by taking several images of a calibration plate at different planes in the tunnel (parallel to the window, at different distances from the window). A least-squares fitting algorithm finds the calibration marks on the plate and calculates the best-fitting camera model. For details, see References [?] and [?].

To begin this process you will need:

1. The forked sting and 2x shoulder bolts (in a pink bag next to the sting)
2. The calibration plate, spacers, shoulder bolts, nuts, and mount
3. Digital protractor
4. Black electrical tape



Before Continuing

You will need to know what calibration planes are necessary for your specific model. These are the distances from the window at which you place the calibration target; they should extend from slightly in front of the model to slightly beyond the visible portion of the model. In particular, you must know what your desired focus plane is so you can image that one first (once you have correctly focused on it, of course).



Before Continuing

Before taking images during the optical calibration, lock the optical table in place and ensure that the weight distribution on the table is roughly what it will be during a run. Remove the large test-section bolts from the table (put them on the floor), install the Lexan shields, and place any other equipment as you expect it to be distributed during a typical run.

3.3.1 Performing the Calibration

Before installing the calibration plate, you should apply the black electrical tape to the downstream side of the IR window frame to reduce reflections. The tape should extend fairly far around the tunnel, and it must be as flat as possible (no bubbles). However, **never let the tape touch the surface of the IR window**. If during the calibration you see hot reflections from spots that you did not cover, you can cover them at any time. To remove the tape, carefully peel it off the surface of the tunnel. Use a small amount of ethanol on a Kim-Tec wipe and gently wipe down the area where the tape was, being very careful not to let ethanol get on the IR window.

Use the following procedure to install the calibration plate:

1. Insert the forked sting into the sting mount all the way to the back (i.e. until you cannot push it back further).
2. Insert the calibration plate mount into the forked sting and tighten it in place using the two sting shoulder bolts.
3. Use the digital protractor on the flat mount surface to level the sting/mount.
4. As you tighten the sting, the mount will rotate slightly. Try to account for this when you rotate the sting before tightening. You are aiming to achieve level within a few tenths of a degree.

Once the sting is locked in place, you are ready to mount the calibration plate itself. Before doing this, you must figure out what your focus plane is going to be. For a cone at angle of attack, rolled 45° toward the camera, the focus plane is about 1.8 inches, to give you some idea. Choose the correct spacers and shoulder bolts for your focus plane, and install the calibration plate with the marking holes facing the camera. Subsection 3.3.2 describes the available spacers and their color codes. With the calibration plate installed and tightened, close the tunnel, but leave the bleed air on.



For Your Information

It is helpful to leave the calibration plate in the tunnel with the bleed air on for 20 minutes or so. As the plate heats up, the contrast between the markers and the surface of the plate increases, which makes the calibration algorithm more accurate.

After the plate is hot, you should see in the IRBIS software a grid of bright dots on a darker background. When most of the dots are easily distinguishable from the background, you are ready to take the calibration data. To take the data, use the following settings in IRBIS (described further in 4.1):



For Your Information

To use the PIRANHA calibration utility, your files need to be named in the following format: `plate_<plane>`, where `<plate>` should be the calibration plane distance times 10 (i.e. 1 inch is `plate_10`). For negative distances, add an `m` before the distance, and for distances less than 1, you need a leading zero (as in `plate_05` for 0.5 inches).



For Your Information

The plate calibration images must be output as a `.asc` file. IRBIS should append a `_1` before the file extension, but if it does not, you must add this yourself to work with PIRANHA.

After you have taken and saved all your images (3 – 4 planes should be sufficient, but more is better), you can open the calibration utility in PIRANHA and perform the calibration.



Before Continuing

When you are finished with your optical calibrations, put away all the equipment into their labeled Lista drawers downstairs in 29B. If you have used electrical tape, remove it and clean the tunnel surface as previously described.

Congratulations! You have successfully set up the IR camera and performed an optical calibration. After you have installed your model, proceed to the next chapter (make sure you place the model in the camera's field of view before you lock it in place).

3.3.2 Calibration Plate Spacer Color Codes

There are three spacers available: 0.2 inch, 0.4 inch, and 0.8 inch. Table 3.1 provides the color codes for a few calibration planes. The plates and shoulder bolts are marked with nail polish of the appropriate color. Figure 3.18 shows the possible orientations of the calibration mount, denoted sides 1 and 2. A few distances from the centerline are labeled. Note that the calibration plate thickness is 0.4 inch, and is implicitly included in every combination of plates in Table 3.1.

Table 3.1: Selected calibration plane configurations.

Distance from \mathcal{C} (inch)	Color Code	Orientation	Plates Needed (inch)
1.2	Red ●	Side 1	0.2, 0.8
1.4	Pink ●	Side 2	none
1.6	Blue ●	Side 2	0.2
1.8	Orange ●	Side 2	0.4
2	Green ●	Side 2	0.2, 0.4

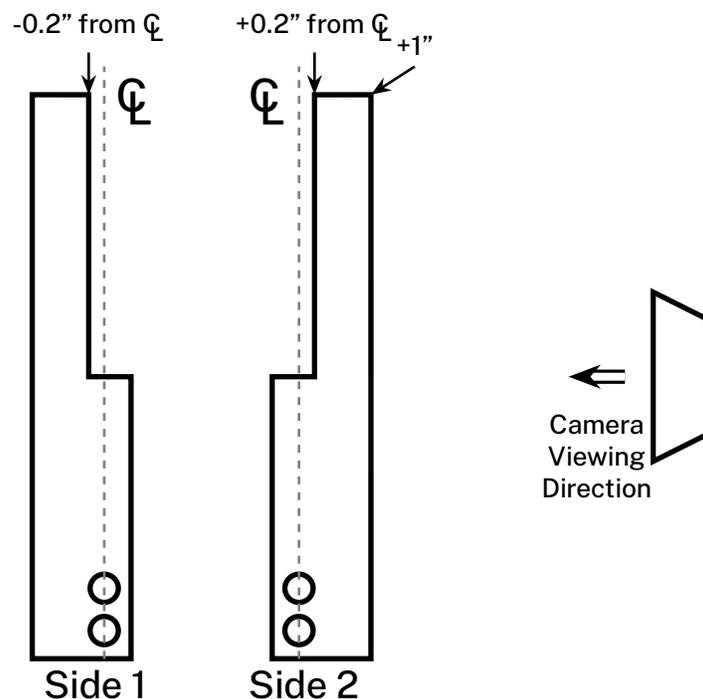


Figure 3.18: The two possible orientations for the calibration mount. The centerline refers to the tunnel centerline. The view is looking down towards the floor if the camera is on the north side of the tunnel.

3.4 Uninstalling the Camera

The disassembly of the camera is fairly straightforward and follows similar procedures and rules as the installation. Here are a few differences of which you should be aware:

1. Before turning off the camera, disconnect from it in the IRBIS software (see Section 4.1).
2. Remember to put the lens cap back on the lens as soon as possible after you remove the camera from the mount.
3. To remove the GigE adapter from the camera and converter box, follow the instructions in the next subsection.
4. Repack the Pelican case and put all the IR equipment back on its shelf in 29B.
5. When uninstalling the IR window, **always** check the window for damage before putting it away.

3.4.1 Removing the GigE Adapter



Warning

Please be careful with the fiber optic cable when removing the GigE adapter from the camera and converter box. Never yank on either the cable or the adapter; both should be able to be removed with little force.

Removing the GigE adapter is not as simple as pulling it out. First you have to press the khaki tab on the green fiber optic cable and pull the cable out of the adapter. Next, pull out the silver handle on the GigE adapter and gently pull the adapter out of its socket. When it is completely free, insert the fiber optic cable back into the adapter until the khaki tab clicks into place.

Chapter 4

Using the Camera

“For me, the camera is a sketch book, an instrument of intuition and spontaneity.”

Henri Cartier-Bresson



Before Continuing

Have you installed the Lexan shield over the optical table? If not, do that before proceeding!



Warning

Do not use the camera for a long time when it is very hot in the room (25 °C to 26 °C or hotter). The camera needs to cool its sensor to function properly and it cannot do that in a hot environment. The camera temperature can be checked in the IRBIS software, and should be about 40 °C or cooler. The box in IRBIS showing the camera temperature will turn red if the camera is getting too hot.

4.1 IRBIS Software

To run the camera and export the images you will need to use Infratec's IRBIS 3.1 plus software. This software should be installed on the dedicated laptop. Log in to the laptop using the “IR user” credentials.

When you open IRBIS, the screen shown in Figure 4.1 should appear. If it does not, the installation has become corrupted (this has happened once before), and you will need to contact Infratec. The main window of IRBIS has several components, annotated in the figure. Most users will only need to worry about the Toolstrip, the Live Image, and the Favorites List. The IRBIS manual, which should be stored with the IR camera (and is also placed in the IR code repository), is a useful guide for the other features.



Before Continuing

Before trying to connect to the camera in IRBIS, the camera must be on, warmed up (see Section 3.2.6), and physically connected to the laptop via the fiber-optic and USB-C cables.

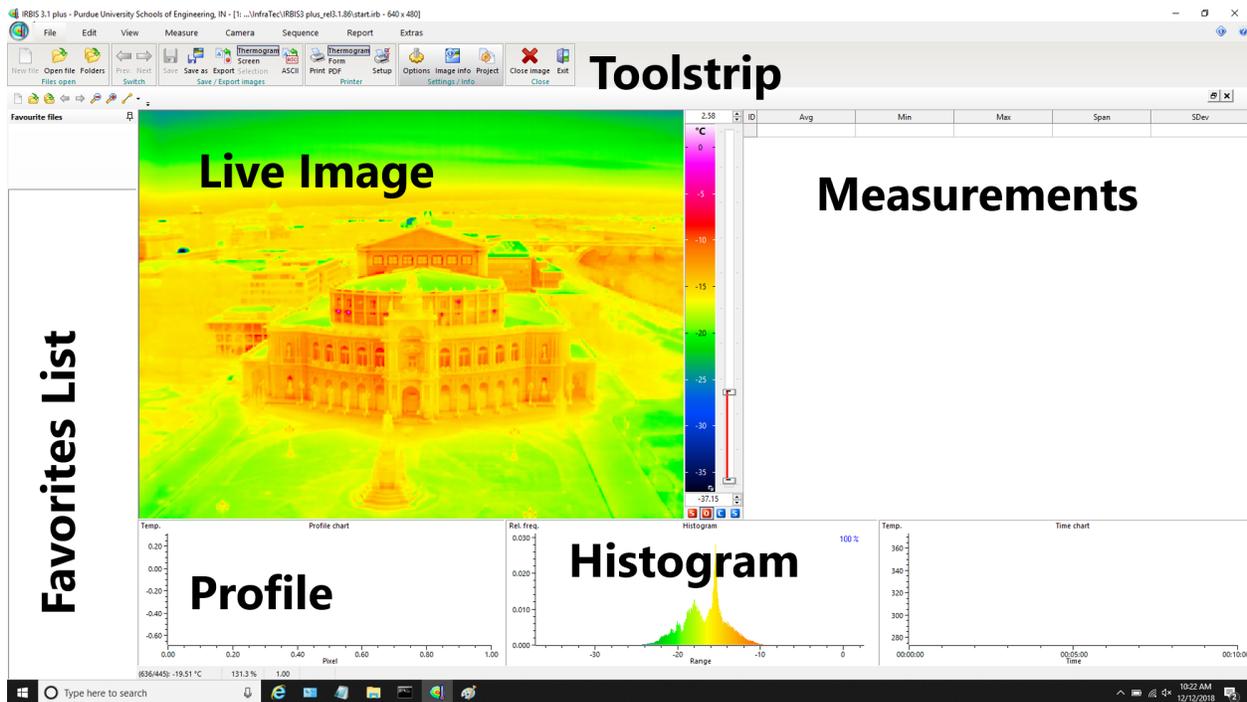


Figure 4.1: The opening screen of IRBIS.

To connect to the camera, go to the Camera tab the Toolstrip and click the **Connect** button as shown in Figure 4.2. A small window will pop up prompting you to select the camera. Use the drop-down menu to select **Image IR 4**. Click **Apply** and IRBIS should connect to the camera.



For Your Information

If IRBIS throws an error when you try to connect, it is often because the user has forgotten to plug the camera into the laptop (via the fiber-optic cable into the converter box, and the USB-C from the converter box into the laptop). Also make sure that the power block for the converter box is plugged into a working outlet. There should be a green light on in the inside of the box when it is powered.

When the camera is connected, a live image from the camera should appear in the Live Image window. If you don't see an image, click the **Live** button in the Camera tab. If you see an image but it's only white noise, ensure that you have removed the lens cap. Once the camera is successfully connect, you can configure it for acquiring your data. To change the colormap used in displaying the image, right click on the colorbar to the right of the Live Image and change the map. There are four buttons just below the colorbar that control the range of the colormap. From left to right they are:

orange S constant range

orange O range floats as the data changes (usually the best choice)

blue C range is the entire calibration range

blue S range is the min and max within the selection (You can select a region by using the Selection tool in the Measure tab.)

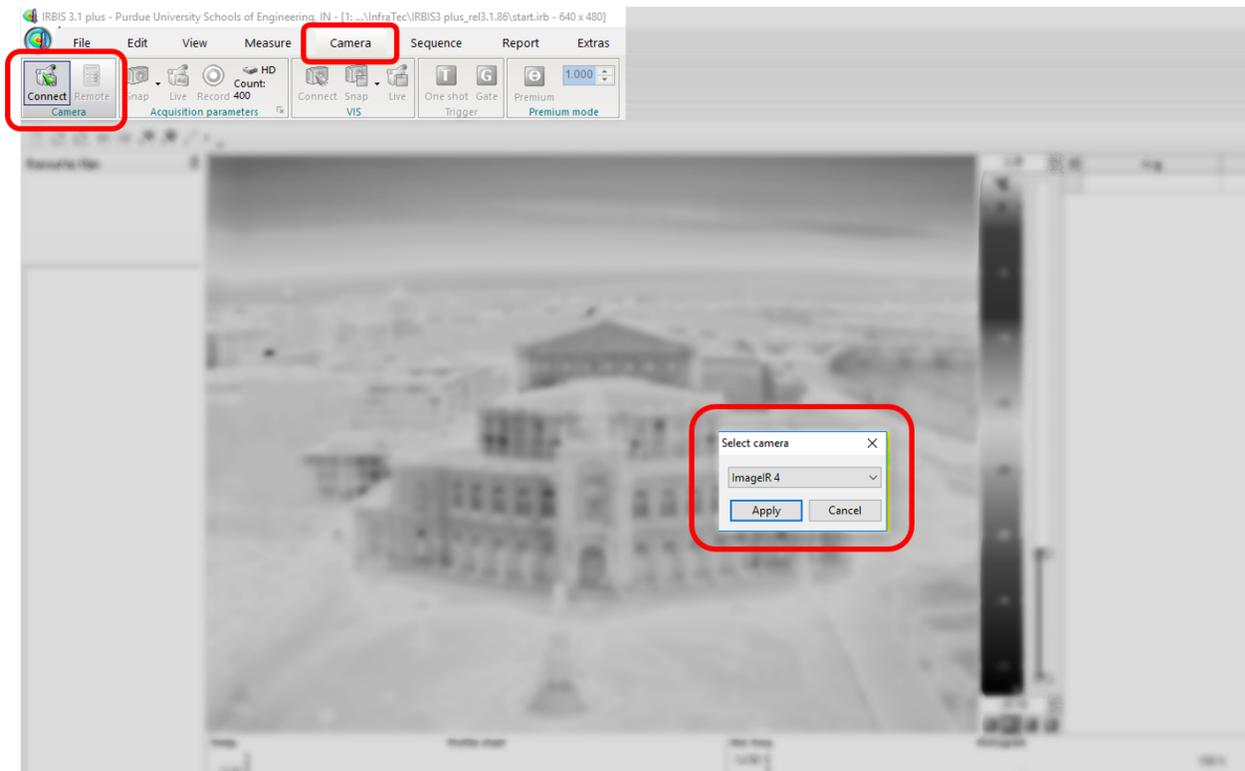


Figure 4.2: Connecting to the camera.

The camera acquisition settings can be manipulated using two windows, as shown in Figure 4.3. To open the Remote Control (right window in the figure) click the **Remote** button in the Camera tab, next to the **Connect** button. The Acquisition window (left window in the figure) can be opened by clicking the small arrow in the bottom right corner of the “Acquisition Properties” sub-menu within the Camera tab. This sub-menu is highlighted in Figure 4.4.

Within the Acquisition Properties window, there are several important settings. The Folder field points to the location you wish to save the files. This can be changed as often or as little as you like. The Name field should be changed for every run, though IRBIS should append numbers to the end of the filename in sequence if the name already exists. Above the Folder field is a checkbox titled “Frames per File.” If this value is less than the number of frames you plan to take, IRBIS will save the output file (extension .irb) in multiple pieces. You may change this to fit your personal preferences.

The Speed panel allows you to set the camera frame rate, and the Frames panel lets you set either the total number of frames to acquire or a length of time. The Start Trigger panel configures the camera triggering. The Trigger dropdown must be set to **Camera** and the “Behavior” must be **Burst** for the camera to trigger properly. It is usually advisable to acquire pre-run data. You can do so by checking the “Pretrigger” checkbox and entering the number of desired pre-trigger frames. The default behavior of the `qca1c` codes assumes one second of pre-run data. To save your settings, click the green checkmark at the top of the window, as indicated. Clicking the red x will cancel the settings.

To open the ImageIR Remote, click the **Remote** button in the Camera tab, highlighted in Figure 4.5. There are four tabs within the Remote window. The first is the Calib (Calibration) tab, shown in Figure 4.6. In this tab, select the integration time you wish to use from the list. The calibration must correspond to the lens that is installed on the camera (e.g. 12 mm in the figure). The 1274 μ s calibration is usually a good choice. The chosen calibration range should bound your data, so if you expect to have very high heating

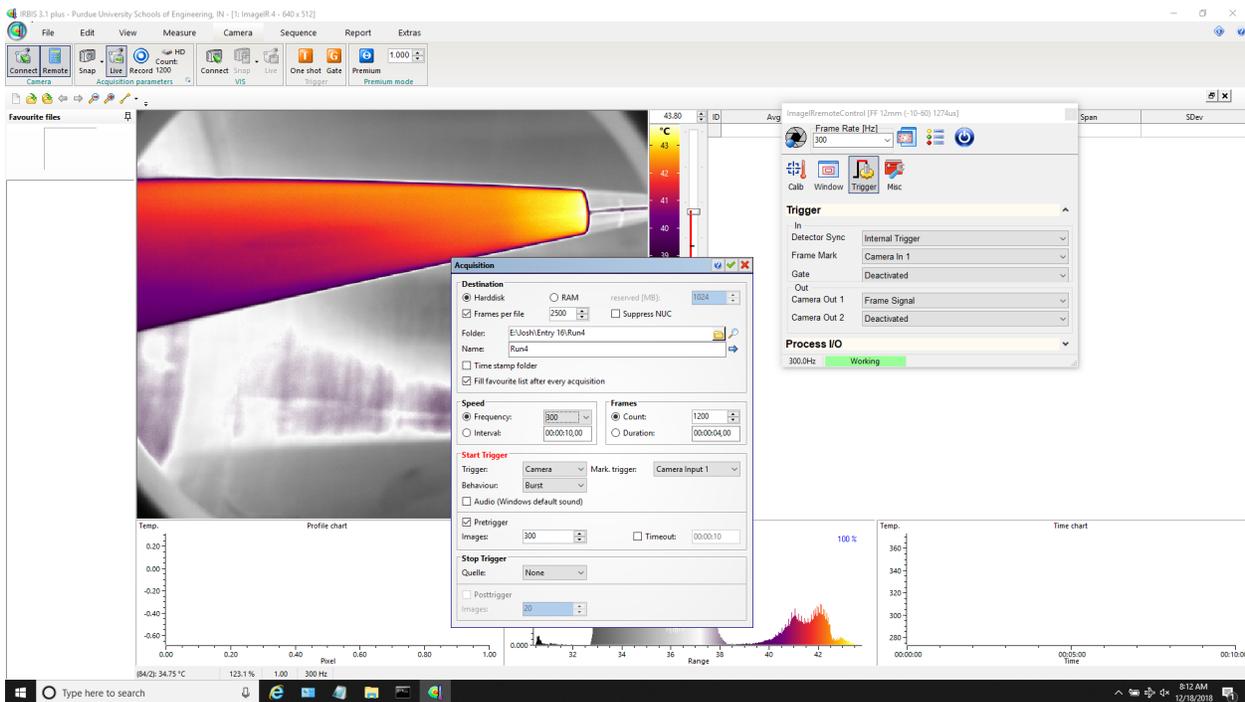


Figure 4.3: The Camera tab.

(temperatures above 60 °C), you may need to select a different calibration. Note that if you are using the 300 mm close-up lens with the 50 mm lens, that takes a separate calibration from the unmodified 50 mm calibration. The Frame Rate dropdown at the top of the Remote should be set to whatever you entered in the Acquisition window. If it is not, you will have to change it in the Remote as well.

The Window tab in the Remote lets you choose the field of view of the image. The default is Full Frame, which is best for most cases. If you need a faster frame rate than 355 Hz, you can select a smaller field of view. The Half Frame and Quad Frame options are centered on the center of the image; if you want a smaller frame centered on some other location, use the “Windowing - Random Frame” tab (note that this does not mean you select a random frame, just that it is not one of the pre-defined frames).

The final window is the “Trigger” tab in the Remote. This tab is opened in Figure 4.5. The Detector Sync should be Internal Trigger. The Frame Mark is whatever input channel you have used, in this case Camera In 1. If you want to record the when the camera has taken an image, set Camera Out 1 to Frame Signal and connect the Camera Out 1 channel on the trigger box to an oscilloscope.



For Your Information

It is good practice to check that the trigger is set up correctly throughout the day. Sometimes it can reset, leading to no trigger and no data.

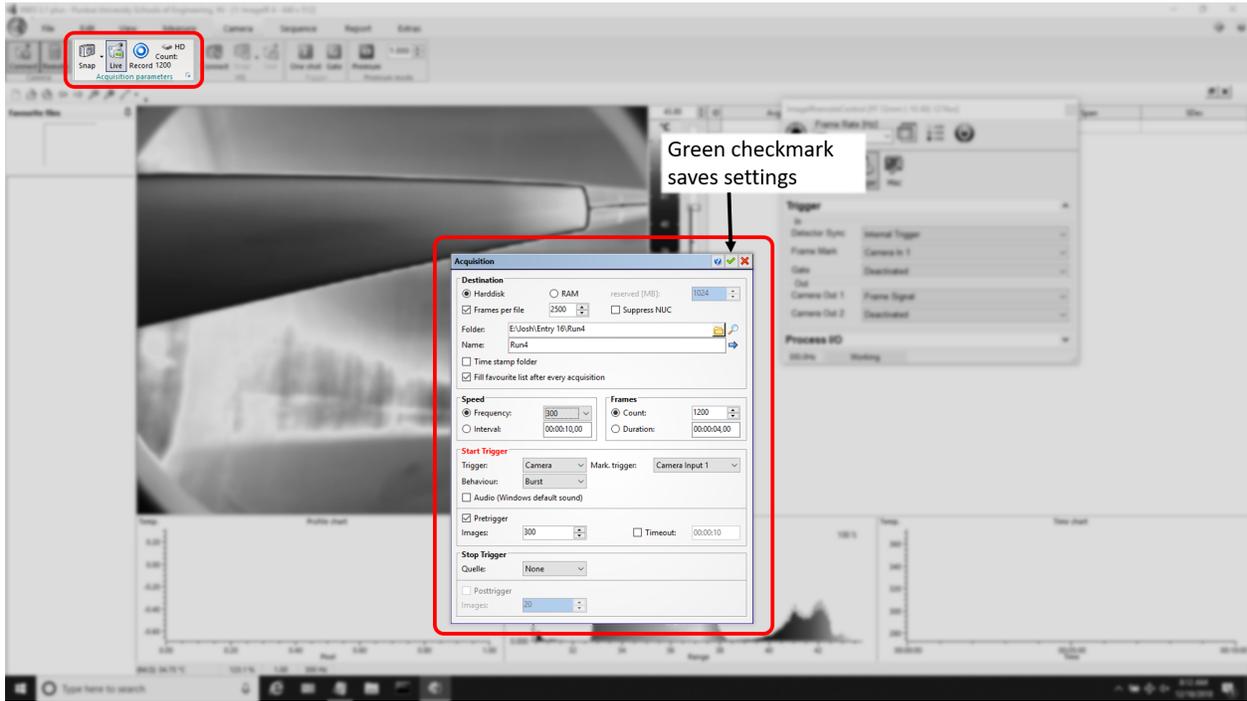


Figure 4.4: The Acquisition Properties window.

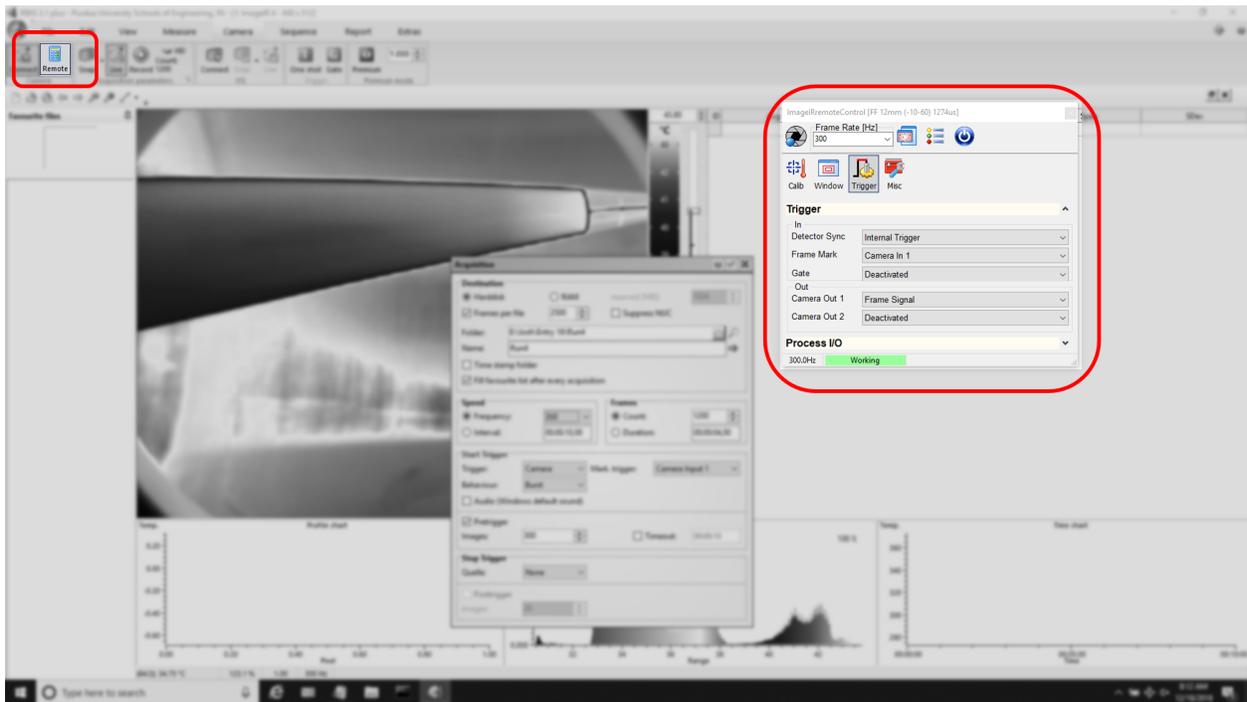


Figure 4.5: The Remote Control.

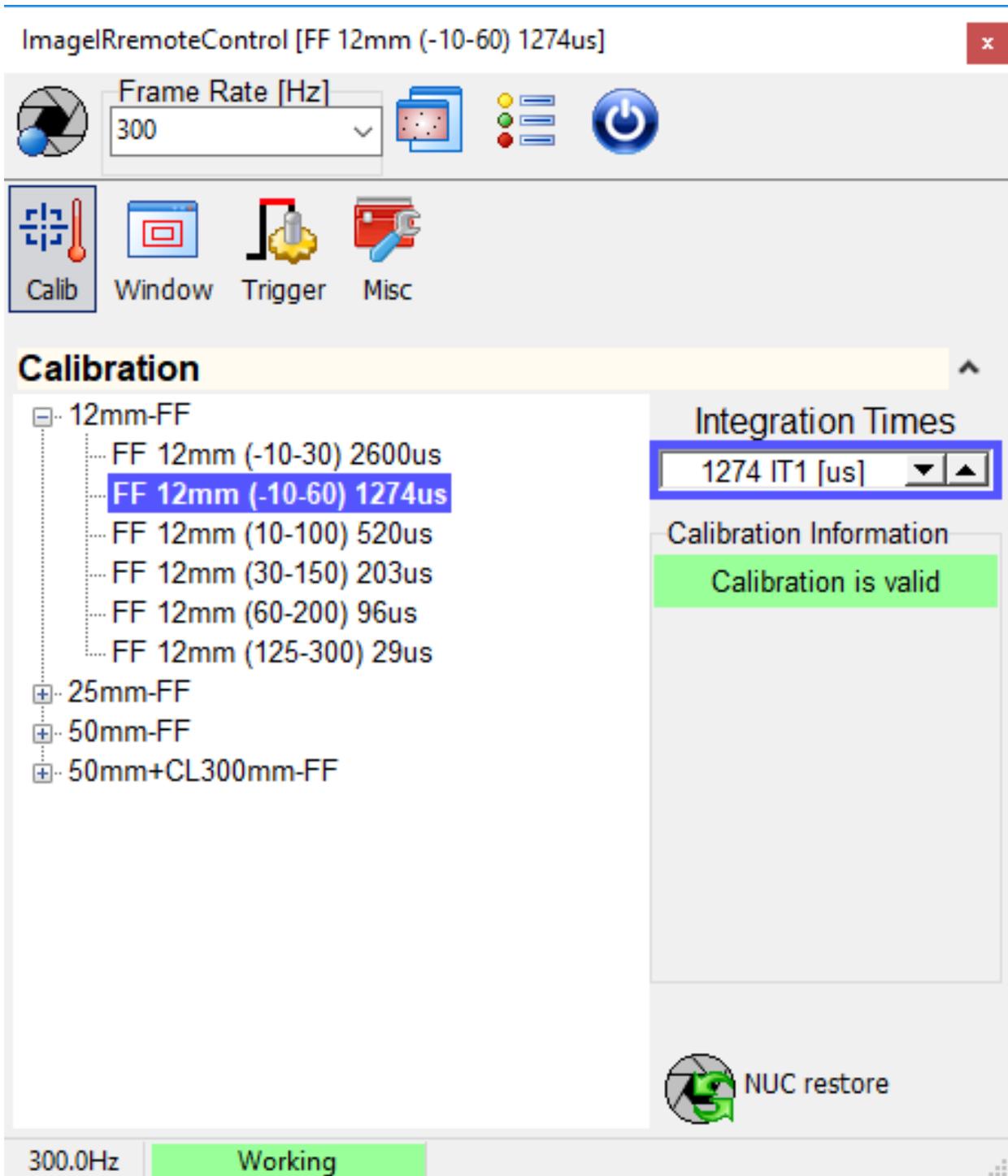


Figure 4.6: The Calibration tab.

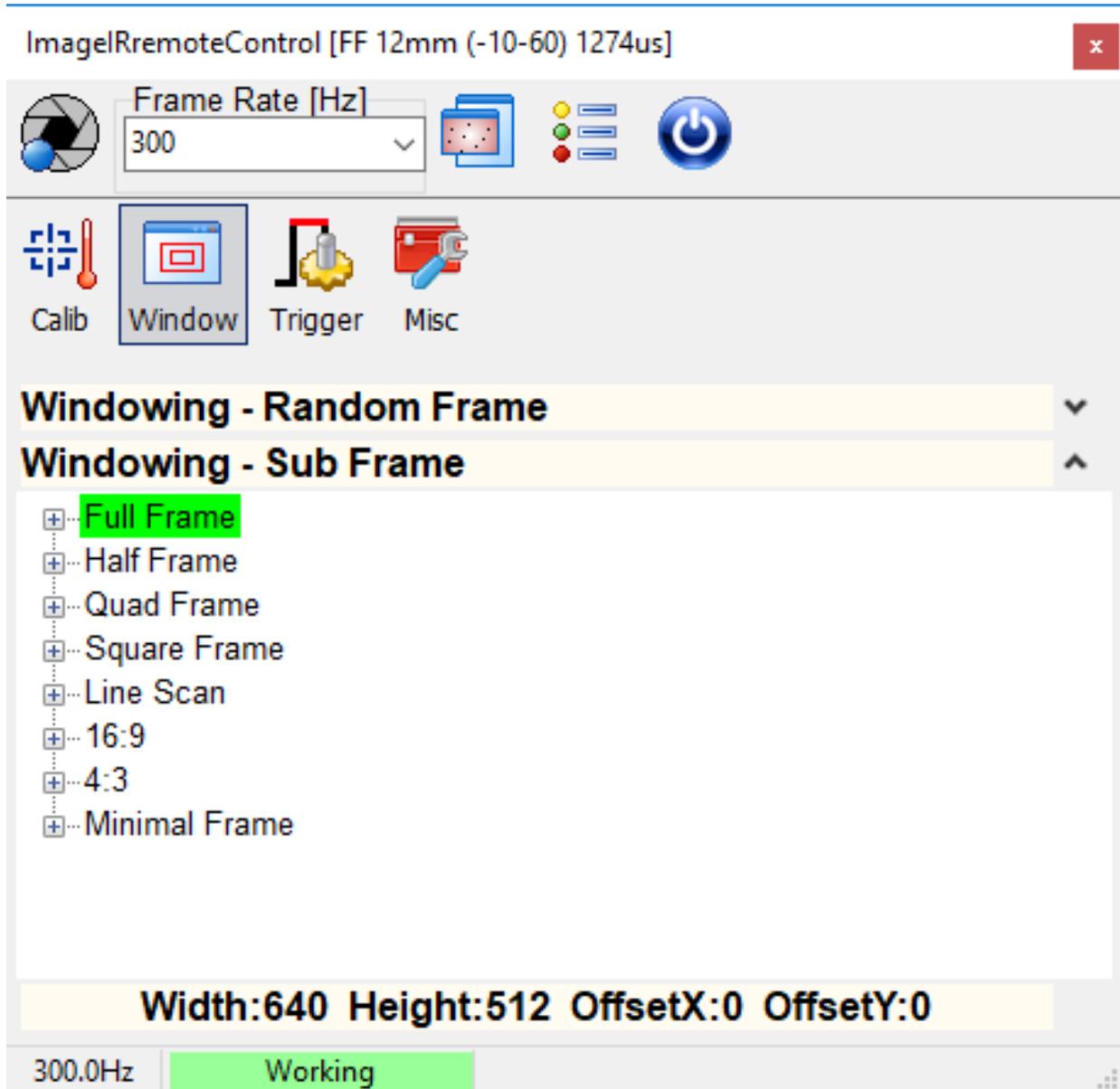


Figure 4.7: The Windowing tab.

4.2 Acquiring Data

When you are ready to acquire data, click the **Record** button in the Camera tab. A window will pop up showing that it is waiting for a trigger. The buffer will fill to your desired amount of pre-run data; don't trigger before the buffer is full. To take a single image without waiting for a trigger signal, go to the File tab and click the **ASCII** button in the Save/Export Images panel in the Toolstrip.

4.3 Exporting Data

To export the data, click on the Sequence tab. Ensure that the export more is set to IRB ASCII by clicking the dropdown arrow at the right of the Export button (highlighted in Figure 4.8). When you are ready to save the data, select the first frame of the .irb file in the Favorites List and then click the Export button.

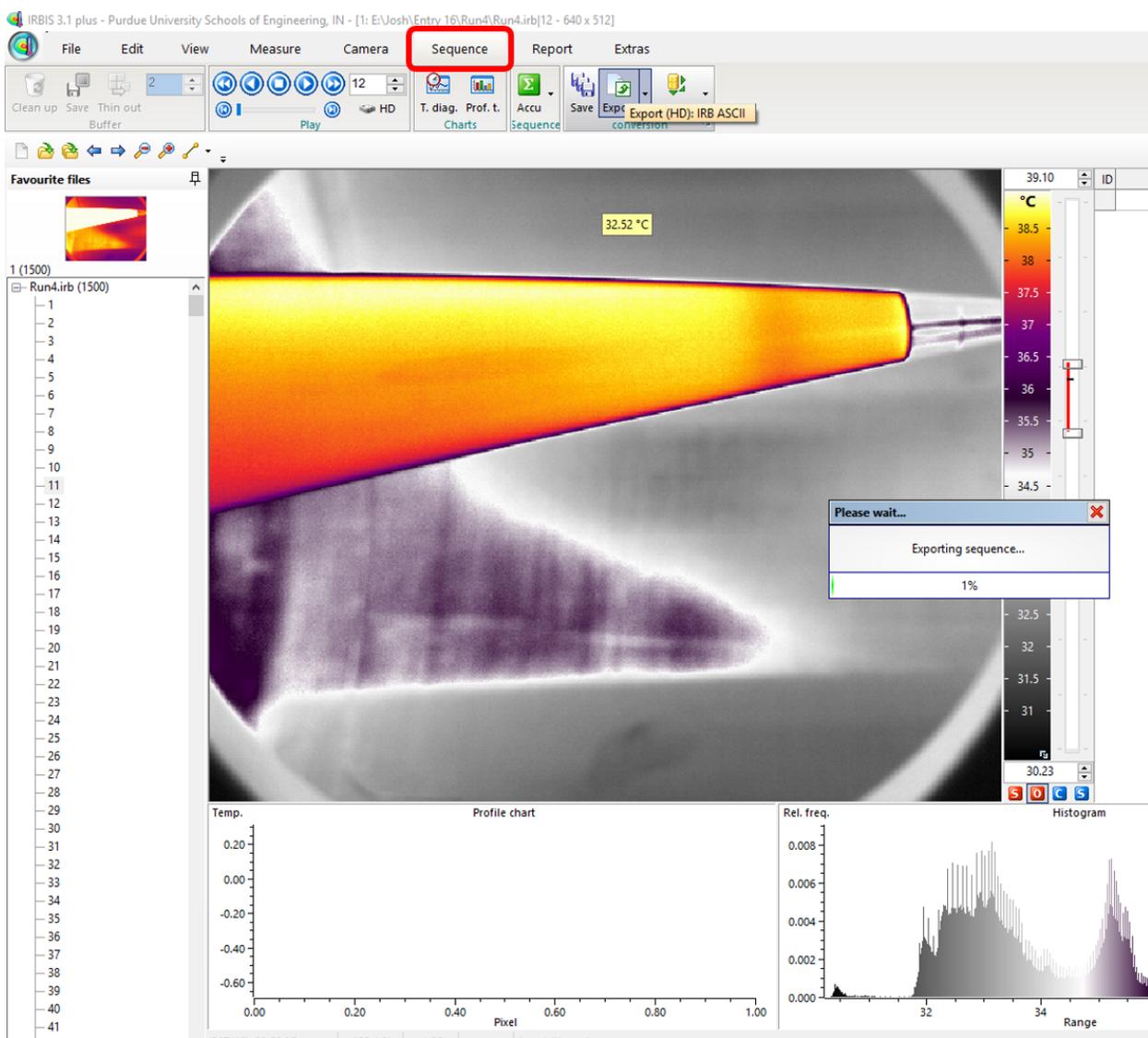


Figure 4.8: Exporting data.

4.4 Disconnecting the Camera

Before you start unplugging cables, click **Connect** in the Camera tab to disconnect the camera. Once the camera is disconnected, you can turn it off and begin putting things away. **You should disconnect and turn off the camera at the end of every day.**