

Uncertainty Quantification for Airfoil Icing

Anthony Matteo DeGennaro

A Dissertation

Presented to the Faculty
of Princeton University
in Candidacy for the Degree
of Doctor of Philosophy

Recommended for Acceptance
by the Department of
Mechanical and Aerospace Engineering
Adviser: Clarence W. Rowley III

November 2016

ProQuest Number: 10195629

All rights reserved

INFORMATION TO ALL USERS

The quality of this reproduction is dependent upon the quality of the copy submitted.

In the unlikely event that the author did not send a complete manuscript and there are missing pages, these will be noted. Also, if material had to be removed, a note will indicate the deletion.



ProQuest 10195629

Published by ProQuest LLC (2016). Copyright of the Dissertation is held by the Author.

All rights reserved.

This work is protected against unauthorized copying under Title 17, United States Code
Microform Edition © ProQuest LLC.

ProQuest LLC.
789 East Eisenhower Parkway
P.O. Box 1346
Ann Arbor, MI 48106 - 1346

© Copyright by Anthony Matteo DeGennaro, 2016.

All rights reserved.

Abstract

Ensuring the safety of airplane flight in icing conditions is an important and active arena of research in the aerospace community. Notwithstanding the research, development, and legislation aimed at certifying airplanes for safe operation, an analysis of the effects of icing uncertainties on certification quantities of interest is generally lacking. The central objective of this thesis is to examine and analyze problems in airfoil ice accretion from the standpoint of uncertainty quantification.

We focus on three distinct areas: user-informed, data-driven, and computational uncertainty quantification. In the user-informed approach to uncertainty quantification, we discuss important canonical icing classifications and show how these categories can be modeled using a few shape parameters. We then investigate the statistical effects of these parameters. In the data-driven approach, we build statistical models of airfoil ice shapes from databases of actual ice shapes, and quantify the effects of these parameters. Finally, in the computational approach, we investigate the effects of uncertainty in the physics of the ice accretion process, by perturbing the input to an in-house numerical ice accretion code that we develop in this thesis.

Acknowledgements

Mom and Dad – I have to start by thanking you two first. You guys are the most selfless, dedicated, self-sacrificing parents on earth, and I owe everything to you and the countless opportunities you have created for me. You have always been fantastic role-models for me and I would be lucky if I could become half the parent that you both were to me.

Kelsey Carroll – babe, you are the love of my life. You have been the most steadfast, loyal, and loving source of support that any guy could ask for. You have always been there for me throughout this doctoral process; you’ve laughed and danced with me when times were good and lifted me up when they were bad. You are my heart and my strength and I am immeasurably grateful that you are in my life. This entire dissertation is for you.

Gigi Martinelli – Gigi, I have a lot to thank you for. You have been a fantastic role-model and mentor to me over the past four years. There aren’t many people whose company I would enjoy for eight straight hours in a car from Princeton to Athens, Ohio, but you are definitely one of them. Thanks for helping me grow as an independent researcher, for always having just the right words of advice for me at just the right time, and for being my friend.

Clancy Rowley – my advisor. Clancy has provided me with technical advice and assistance and for that, I am grateful.

Matt Feldman – there aren’t many folks I’d willingly sit down with and have extensive political discussions with, but you are number one on that list. You are always challenging me to think, re-think, and defend my underlying core axioms and principles. Besides that, though, sometimes I just want to be heard, and you are always right there whenever I need someone to listen to me rant about anything at all. The Dallas Cowboys still suck, though; remember that.

Mark Lohry – my right-hand man! One of the only people here to whom I can air my non-PC, cynical, line-toeing side and expect a laugh instead of a prompt report to

the Princeton Sensitivity Committee. I'd roast you right here on the spot, but this is a page in my doctoral dissertation and hence not the most appropriate of venues for that particular idea. Thanks for teaching me how to smoke meats and helping me get better at coding/Linux.

Mike Hepler – you really made my fourth year here a lot of fun, man. And, of course, you taught me the three cardinal principles: break your enemy's heart, know and admire others' work, and never drink against the grain of the liquor. I will never forget those.

Steve Atkinson, Matt Plasek, and Shoham Bhadra – I had a blast playing guitar in our band with you all, and I still hope that we can find a way someday in the future to come together for one more show.

Scott Dawson – everybody's favorite Aussie! Thanks for being an all around good-guy, and honestly one of the most good-natured, even-tempered people I've ever met. I hope our paths cross again someday down the line; maybe then you might have better luck in trying to beat me in the 40-yard dash.

Manny Sosa – I'm pretty sure Manny will never read this, but he has got to be in these acknowledgements anyway. As my Krav Maga instructor, Manny introduced me to lots of really cool ways to hurt people (only for the lawful, legitimate purposes of self-defense, of course!). I am thankful for that and, hopefully, I will remain a lifelong student of self-defense.

Finally, I would like to thank the Federal Aviation Administration (FAA) for their gracious financial support under the Joint University Program (JUP).

This dissertation carries T-3331 in the records of the Department of Mechanical and Aerospace Engineering.

To my parents, Kelsey Carroll, and fellow researchers.

Contents

Abstract	iii
Acknowledgements	iv
List of Tables	xi
List of Figures	xii
1 Introduction	1
1.1 Motivation and Goals	3
1.2 Approach	3
1.3 Contributions	5
2 Polynomial Chaos	7
2.1 General Formulation	9
2.2 Stochastic Collocation	10
2.3 Statistics and Analysis of Variance	13
2.4 Overview of PCE Algorithms	14
3 User-Informed UQ	16
3.1 Introduction	16
3.2 Wing Icing Classifications	17
3.2.1 Rime Ice	17
3.2.2 Ice Roughness	18
3.2.3 Ridge Ice	19

3.2.4	Horn Ice	19
3.3	Application of PCE to Airfoil Icing	20
3.3.1	Selection of Flow Solver and Aerodynamic Conditions	21
3.3.2	Ridge Ice Case	23
3.3.3	Horn Ice Case	29
3.4	Summary	34
4	Data-Driven UQ	35
4.1	Introduction	35
4.2	Low-Dimensional Modeling using POD	37
4.3	Simulation Database	39
4.3.1	Modeling using POD	40
4.3.2	Airfoil Icing UQ: Five Parameter Scenario	45
4.4	Experimental Database	48
4.4.1	POD of the Ice Shape Data	53
4.4.2	Airfoil Icing UQ: 10 Parameter Scenario	60
4.4.3	A Data-Driven Ice Accretion Model	66
4.5	Database Partitioning	72
4.5.1	The XOR Similarity Metric	73
4.5.2	Graph Laplacian and Spectral Partitioning	75
4.5.3	Clustering in POD coordinates	78
4.5.4	Cluster Modeling	80
4.5.5	Cluster Uncertainty Quantification	80
4.6	Summary	85
5	Computational UQ	88
5.1	Overview of the Icing Process	89
5.2	Code Structure	92

5.3	Droplet Simulation Module	94
5.3.1	Droplet Advection	94
5.3.2	Tracking Particles	97
5.3.3	Impingement Modeling: Bouncing, Deposition, and Splashing . .	99
5.4	Thermodynamics Module	101
5.4.1	Mass/Energy Conservation	103
5.4.2	Closure Constraints	104
5.4.3	Convective Heat Transfer Computation	105
5.4.4	Solution Method	106
5.4.5	Ice Addition	108
5.5	Validations	109
5.5.1	Droplet Simulation	110
5.5.2	Icing Calculations	112
5.6	UQ Results	116
5.6.1	Two Parameters: T_∞ and LWC	117
5.6.2	Three Parameters: T_∞ , LWC, and ΔT	119
5.6.3	Three Parameters: T_∞ , LWC, and MVD	121
5.6.4	Three Parameters: T_∞ , LWC, and Roughness	123
5.7	Summary	124
6	Conclusion	129
A	UQ for Cargo Hold Fires	132
A.1	Introduction	132
A.2	Simulation methodology	134
A.2.1	Discontinuous Galerkin simulation tool	134
A.2.2	Cargo hold geometry and boundary conditions	136
A.3	2-D Cargo Hold Fire with Uncertain Location/Temperature	138

A.4 Summary	144
A.5 Acknowledgments	146
B Droplet Distributions	147
C Ice Test Matrix	149
Bibliography	151

List of Tables

3.1	Comparison of Statistical Moments for Monte Carlo and PCE: $(\sigma_R, \sigma_{S_r}) = (10\%, 1.25\%)$	25
3.2	Comparison of Statistical Moments for Monte Carlo and PCE: $(\sigma_R, \sigma_{S_r}) = (40\%, 5\%)$	26
3.3	Comparison of Statistical Moments for Monte Carlo and PCE: $h = \mathcal{N}_{1/2^+}(0, 0.5^2), s = \mathcal{N}(1, 0.45^2)$	31
3.4	Comparison of Statistical Moments for Monte Carlo and Multi-Element PCE: $h = \mathcal{N}_{1/2^+}(0, 0.5^2), s = \mathcal{N}(1, 0.45^2)$	33
4.1	Data correlations	45
4.2	Sobol Indices (Single Parameter)	46
A.1	Discrete simulation parameters for uncertainty quantification study. The parameter sweep is performed using a tensor product of these values. . . .	139
A.2	Statistical quantities of interest for $t_R(Z)$	142
B.1	27-Bin Droplet Distributions	148
C.1	Ice Accretion Test Matrix	149
C.2	Ice Accretion Test Matrix (continued)	150

List of Figures

2.1	Univariate Legendre polynomials ϕ_k , $k = 1 \dots 10$	11
2.2	Gauss-Legendre nodes for $d = 2, 3$	12
2.3	Smoylak sparse grid constructions, using Fejer nodes of level 4.	13
3.1	Ridge/horn parameterization.	21
3.2	FLO103 verifications.	23
3.3	Ridge ice shape variations.	23
3.4	Ridge UQ results (small uncertainty).	25
3.5	Ridge UQ results (large uncertainty).	26
3.6	Ridge ice flowfields.	29
3.7	Horn ice shape variations.	30
3.8	Horn UQ results.	30
3.9	Horn UQ results (multi-element PCE).	33
4.1	NASA CRM horn ice.	40
4.2	Horn profiles, unaligned and aligned.	42
4.3	POD eigenvalues and modes for the scaled/shifted data.	43
4.4	POD reconstructions of the ice.	43
4.5	Spanwise variation of the POD modes.	44
4.6	Variation of the POD modes on the mean.	44
4.7	5-dimensional parameter space ice shape variations.	46

4.8	Output probability density functions (5 parameter UQ study).	47
4.9	Good/bad horns (5 parameter UQ study).	49
4.10	Effect of horn position on performance (5 parameter study).	49
4.11	Effect of POD coefficients on performance (5 parameter study).	50
4.12	Scaling transformation example.	52
4.13	Experimental ice shape dataset.	53
4.14	Ice shape definition (experimental database).	54
4.15	POD eigenvalues (experimental database).	55
4.16	Mean and POD Modes (experimental database).	57
4.17	Projection of ice shapes onto POD modes (experimental database).	59
4.18	POD coefficient distributions (experimental database).	62
4.19	1,921 Latin Hypercube samples.	64
4.20	Statistics computed using 1,921 LHS samples.	65
4.21	Spatial statistics of C_L , computed from LHS sample shapes.	66
4.22	Statistics for the top/bottom deciles of C_L .	69
4.23	POD coordinates, colored according to physical parameters.	70
4.24	Purely data-driven model ice shapes.	71
4.25	Random ice shapes from data-driven model.	71
4.26	Example depiction of the XOR of two ice shapes.	73
4.27	Distribution of point-to-point XOR distances, normalized by $\sigma = 3000$.	74
4.28	Eigenvalues of the Laplacian (ordered by magnitude).	76
4.29	Laplacian ice shape clusters.	77
4.30	K-means clustering of ice shapes in POD coordinates.	79
4.31	Modeling of the first group of ice shapes.	81
4.32	Modal ice shape variations (small shape cluster).	81
4.33	Small cluster POD coefficient statistics.	82
4.34	Output statistics (small shape cluster UQ study).	83

4.35	Top/bottom deciles of C_L (small shape cluster UQ study).	84
4.36	Sensitivity of C_L (small shape cluster UQ study).	85
5.1	Rime vs. glaze ice flowfields.	92
5.2	Modular structure of CATFISH	93
5.3	Collection efficiency calculation via a Lagrangian method.	96
5.4	Quadtree illustration.	99
5.5	Illustration of heat transfer mechanisms in airfoil icing.	102
5.6	Illustration of the finite volume method.	107
5.7	Illustration of the area ablation effect.	109
5.8	Droplet trajectories and collection efficiencies.	113
5.9	Benchmark tests of CATFISH.	114
5.10	Illustration of driving the thermodynamics to steady-state.	115
5.11	2 parameter UQ study results.	119
5.12	Dangerous/benign ice shapes (2 parameter UQ study).	120
5.13	Effects of varying MVD.	122
5.14	Computational UQ results (first 3 parameter study).	126
5.15	Computational UQ results (second 3 parameter study).	127
5.16	Computational UQ results (third 3 parameter study).	128
A.1	Cargo hold fire example.	137
A.2	Effects of varying source location.	138
A.3	Effects of varying source temperature.	138
A.4	Temperature field time evolution.	139
A.5	PCE surrogate for $t_R(Z)$.	141
A.6	Temperature field snapshots with $Z = (0.48, 1.21)$.	141
A.7	Statistical quantities of interest for time-averaged ceiling temperature.	143
A.8	Distributions of maximum ceiling temperature value/location.	143

A.9 Ceiling temperature distributions at quadrature nodes.	144
A.10 Ceiling temperature distributions (interpolations).	145

Chapter 1

Introduction

In an aggregate sense, aviation is one of the safest means of transportation that exists today. According to statistics provided by the National Transportation Safety Board (NTSB), of the 34,678 transportation fatalities registered in 2013, only 443 were related to aviation, compared to 32,719 highway-related fatalities, 891 rail-related fatalities, and 615 marine-related fatalities [15]. Ensuring the continuation and improvement of this tradition of safe airplane flight is the responsibility of the Federal Aviation Administration (FAA), which has the authority to establish regulations governing certification rules and standards to assess the airworthiness of airplanes.

One longstanding phenomenon that poses issues for aircraft safety is the in-flight accumulation of ice on an aircraft in hazardous atmospheric conditions. Several high-profile accidents have occurred where *ex post-facto* analyses have implicated icing as the culprit. In October 1994, an Avions de Transport (ATR) ATR 72 series airplane was en route from Indianapolis to Chicago when it encountered severe icing conditions; tragically, this resulted in a catastrophic loss of control which crashed the airplane and killed all 68 passengers/crew. Subsequent investigation by NTSB, NASA, and others revealed that supercooled large drops (SLDs) had impinged on the airplane wing aft of the deicing boots and created a ridge of ice on the upper surface that resulted in the loss of control [13]. In

January 1997, an Embraer EMB 120 Brasilia model airplane had departed from Cincinnati and was headed for Detroit when it too experienced unrecoverable loss of control due to ice accretion on the wings. The resulting crash killed all 29 people on board [14]. In addition to these commercial aircraft accidents, icing has also caused a number of private airplane crashes – 26 total accidents involving the Cessna 208B Caravan were recorded between 1987 and 2003, with 47 total fatalities involved [99]. Between 1990 and 2000, structural ice accretion accounted for 153 crashes [49].

These accidents (and others like them) have inspired the FAA/NTSB to devote significant resources toward improving airplane safety with regards to icing. This is underscored by a recent addendum to Title 14 of the Code of Federal Regulations (14 CFR), part 25 (Airworthiness Standards: Transport Category Airplanes), which enacts stricter certification rules for flight in supercooled large droplet (SLD) icing conditions [6]. Aside from regulations, the FAA and others have taken an active interest in funding research aimed at improving icing safety. As we will see in this thesis, the literature encompasses work that addresses informational deficiencies across a wide range of topics, including computation, experimental observations, fundamental physical understanding, sensitivity analyses/optimizations, etc.

Notwithstanding this progress, ice accretion is a process which is, in practice, subject to a wide range of uncertainty. The sources of these uncertainties are manifold, but what is important is that the existing literature lacks a thorough treatment of this topic. Stated plainly, this is the central objective of this thesis – to address the topic of uncertainty in the ice accretion problem. The work involved in this area falls into one of a few categories: describing/modeling sources of uncertainty, developing and/or implementing methods for quantifying uncertainty and statistically exploring a parameter space, and analyzing the statistics of a particular output quantity of interest. For completeness, we will address all three of these categories in this thesis.

1.1 Motivation and Goals

This work is motivated by the observation that airfoil ice accretion is a fundamentally uncertain problem, which makes certification for safe flight challenging. Sources of uncertainty can be meteorological (e.g., free-stream temperature), aerodynamic (e.g., turbulence), or geometric (e.g., airfoil profile and ice shape). We have found that the literature does not provide a thorough treatment of how to assess the statistical effects of these uncertainties on important safety/performance metrics.

The goal of this work is to help close the informational deficiency gap regarding uncertainty quantification (UQ) for the icing problem. We wish to apply the methods and techniques developed by the uncertainty quantification community to problems in airfoil ice accretion. Doing this will involve first modeling the sources of uncertainty in the icing problem; this input-process modeling can be approached in several ways and constitutes a large part of this thesis. Once an input parameter space has been determined, we must select a set of tools for statistically exploring the input parameter space. The tools we employ in this thesis are chosen for their accuracy, efficiency, and other attractive properties that will become evident in later chapters. We will use these tools to investigate how aerodynamic performance metrics (e.g., lift) are affected statistically by uncertainty in the ice shape or in underlying physical conditions.

1.2 Approach

The approach we follow in this work is one which combines a number of techniques from the uncertainty quantification, low-dimensional modeling, and computational modeling communities. The breadth of techniques employed is needed to address the full scope of the icing problem.

The general procedure in approaching UQ for any application is to first identify a set of governing parameters whose variation controls the output quantity of interest, endow

these parameters with a probability distribution that reflects the uncertainty present in the problem, and then statistically explore the effects of this uncertainty by leveraging data obtained at select locations in that parameter space. The first of these objectives is arguably the most important, as the subsequent UQ results can only be meaningful if they are generated using reliable and comprehensive data and/or input models. In the icing problem, we will discuss three specific approaches for identifying a governing parameter space – user-informed, data-driven, and computational.

The “user-informed” approach is the most straightforward to understand and implement. Here, the user is assumed to have *a priori* knowledge of a set of parameters that controls all deviations in ice shape from some baseline mean shape. These parameters are not arbitrary, however; they correspond to scalings/translations that empirically describe a wide range of ice shapes found in the literature. The user then proceeds to study how aerodynamic performance is statistically affected by uncertainties in these parameters. This approach is advantageous when the dominant sources of uncertainty are easy to identify, as is the case when those sources are simple scalings (e.g., the “height” of an ice horn). Of course, care must be taken to ensure that the allowable parametric variations produce ice shapes that are reasonably reflective of those observed in nature, and not arbitrary or unrealistic.

The “data-driven” approach explicitly addresses the deficiencies of the user-informed approach by grounding the allowable shape variations in actual data. In this method, we first obtain a database of ice shapes (which may be either experimental or computational in nature) generated under different physical conditions. We then build a parametric model that captures the variations present in this dataset using Proper Orthogonal Decomposition (POD). We further show how to incorporate physical information into this parametric model, thereby producing a data-driven model of icing. As we will see, this method can suffer from too much “diversity” in the data – that is, having a database whose members are more dissimilar to one another requires a more complicated model (i.e., more POD

modes) than a database whose members exhibit less variation. One way to alleviate this situation is to use clustering techniques to partition the data into separate groups, which results in several databases which are easier to model. We also show how these clustering techniques can be employed as an empirical classification scheme, which allows us to learn groupings for similar ice shapes.

The computational approach complements the other approaches by enabling an investigation of uncertainties in the physics that give rise to icing. As a part of this thesis, we develop a computational ice accretion code, which calculates the ice shape on an airfoil given different physical conditions. This code first computes the distribution of impinging water droplets over the surface of the airfoil (using an aerodynamic flow solution provided by a separate in-house CFD solver), and next computes the resulting ice shape by solving mass and energy conservation equations. We apply the techniques of uncertainty quantification to investigate how statistical variations in the governing physical parameters (e.g., temperature, liquid water content, accretion time) of this code can produce different ice shapes with different aerodynamic characteristics.

Regardless of which of the three techniques just described is used to identify an uncertain parameter space, a method must be used to quantify the uncertainties that result. In this thesis, we make extensive use of polynomial chaos expansions (PCE) and related quadrature techniques in order to perform efficient and accurate UQ. We also, to a lesser extent, make use of Latin Hypercube sampling (LHS) and quasi Monte Carlo sampling.

1.3 Contributions

We make several contributions in this thesis, all related to the application of modeling and uncertainty quantification to the problem of airfoil ice accretion. Enumerated explicitly, these contributions are as follows:

1. Quantification of the statistical effects of ice accretion uncertainty by means of polynomial chaos expansions (PCE), quasi Monte Carlo sampling, and Latin Hypercube sampling (LHS).
2. Low-dimensional modeling of large databases of ice shapes using Proper Orthogonal Decomposition (POD).
3. Development of a purely data-driven, statistical model of ice accretion.
4. Empirical classification scheme for databases of ice shapes using spectral graph partitioning and clustering techniques.
5. Development, verification, and implementation of an in-house computational ice accretion code.
6. Three different approaches for identifying uncertain parameter spaces in ice accretion
 - user-informed, data-driven, and computational.

Chapter 2

Polynomial Chaos

The purpose of this chapter is to provide a brief introduction to the methods of polynomial chaos expansions (PCE), which are applied extensively in this thesis. For a more thorough treatment, the interested reader is referred to standard references in the field (e.g., [34, 104, 55]).

One of the main objectives of this thesis is the quantification of uncertainty in aerodynamic performance metrics, resulting from uncertainty in the parameters that govern the ice shape. We assume for the time being that we have perfect statistical knowledge of these parameters – that is, we know how many parameters there are, whether they are bounded or not, and what the joint probability density function for these parameters is (how we arrive at this knowledge is a separate issue). Our goal is to quantify the statistical relationship between these input parameters and an output metric. Traditionally, uncertainty quantification (UQ) problems such as this have been approached through Monte Carlo methods. In a Monte Carlo approach, one draws a population of N random samples from the event space in question, with assurance of statistical convergence guaranteed by the Central Limit Theorem at rate $N^{-1/2}$. A drawback of these methods is that they are sampling-based, and as such often require undesirably or unfeasibly large sample sizes. A

relatively new alternative to this approach which is gaining increasing popularity is to use polynomial chaos expansions (PCE).

Polynomial chaos is a spectral method, and as such is a fundamentally different approach to UQ than traditional sampling-based methods. In PC methods, one seeks to approximate the input-output mapping from the uncertain parameter space to the output quantities of interest. This is done by using polynomials that are orthogonal with respect to the probability measure on the underlying input parameter space.

There are several advantages underlying this approach. First, if the input-output mapping is smooth, and the dimensionality of the input parameter space is relatively modest (e.g., $d < 5$), then the methods used to construct a PCE (which will be discussed in this chapter) can be much more efficient than Monte Carlo methods. Additionally, the significance of obtaining an explicit polynomial surrogate cannot be understated – once constructed, the PCE can be Monte Carlo sampled at almost no computational cost, and local and global sensitivities can be evaluated analytically.

There are two main approaches to UQ using PCE: the stochastic Galerkin method, and the stochastic collocation method (see Xiu[104] for more information about both). The first method can be thought of as an extension of traditional Galerkin methods – it is used to quantify the propagation of uncertainty through a set of governing differential equations that are assumed to be known. The governing stochastic equations are orthogonally projected onto the span of a PCE basis, and all expansions are truncated at finite order. This results in a new system of coupled, deterministic equations which must be solved for the modes of the solution expansion. The stochastic collocation method, on the other hand, is essentially a discrete Galerkin approach, whereby the integrals in the projection equations are approximated by a quadrature rule of sufficient accuracy. This process results in the evaluation of the governing equation at a finite number of nodes, or “collocation points.”

It is important to note that the former method is intrusive, in the sense that the governing equations (and any code which solves them) are fundamentally changed by the

Galerkin projection. The stochastic Galerkin approach can be very difficult to implement when the governing equations are large or complicated, since the new, coupled equations must be first derived and then solved. The stochastic collocation method, on the other hand, does not require the derivation of new equations, and so any legacy codes for solving the original equations may still be used. This represents a key advantage of the stochastic collocation method: all that is required are solutions at select points in parameter space. As such, the method need not be limited to applications involving governing differential equations; in general, all that is required is a set of uncertain parameters along with an output metric controlled by those parameters. It is for this reason that we will be using the stochastic collocation method for PCE in this thesis.

2.1 General Formulation

Let $Z = (Z_1, \dots, Z_d)$ be a vector of d independent random variables, with probability density functions $(\rho(Z_1), \dots, \rho(Z_d))$, and let $y(Z) : \mathbb{R}^d \mapsto \mathbb{R}^m$ represent a mapping for m different quantities-of-interest that depend on Z . In the context of this thesis, Z will represent a set of random parameters that control the ice shape on an airfoil, and $y(Z)$ will be a vector of some aerodynamic performance metrics (e.g., lift, drag).

The goal of the method is to approximate each component of $y(Z)$ in terms of some basis functions Φ_k . Assuming – for ease of exposition – that $y(Z)$ is simply a scalar, we can write the N^{th} -order PC approximate of $y(Z)$ as:

$$y(Z) \approx y^N(Z) = \sum_{|i|=0}^N y_i \Phi_i(Z). \quad (2.1)$$

Here, $i = (i_1, \dots, i_d)$ is a multi-index, and $|i| = \sum_{j=1}^d i_j$.

The issue to address is how the basis Φ_k is chosen. Mutual independence of the random variables $Z_1 \dots Z_d$ implies that the probability space is simply the d -dimensional tensor

product of each of the individual variables. Because of this, we can define the multivariate basis functions $\Phi_k(Z)$ as simply the products of univariate polynomials:

$$\Phi_i(Z) = \prod_{k=1}^d \phi_{i_k}(Z_k), \quad (2.2)$$

where we choose these univariate polynomials to be orthonormal with respect to the weighted inner product:

$$\langle \phi_i(Z_k), \phi_j(Z_k) \rangle = \int_{\Gamma_k} \phi_i(Z_k) \phi_j(Z_k) \rho(Z_k) dZ_k = \delta_{ij}, \quad (2.3)$$

where δ_{ij} is the Kronecker delta function and Γ_k denotes the support of Z_k . It follows from this that the multivariate basis functions Φ_k are also mutually orthonormal with respect to the weighted inner product:

$$\langle f, g \rangle = \int_{\Gamma} f(Z) g(Z) \rho(Z) dZ, \quad (2.4)$$

where $\rho(Z) = \prod_{k=1}^d \rho(Z_k)$ denotes the joint probability density function of Z , and has support Γ .

Eq. (2.4) establishes a direct correspondence between the input parameter probability density function and the polynomial basis chosen. For example, if our input parameters have uniform distributions, then the appropriate PC basis is the Legendre polynomials. The univariate Legendre polynomials, defined on $Z \in [-1, 1]$, are shown in Fig. 2.1 for illustration.

2.2 Stochastic Collocation

We now turn our attention to a method for determining the coefficients y_i in the expansion (2.1). One approach is to calculate them by taking an inner product with Φ_j : because

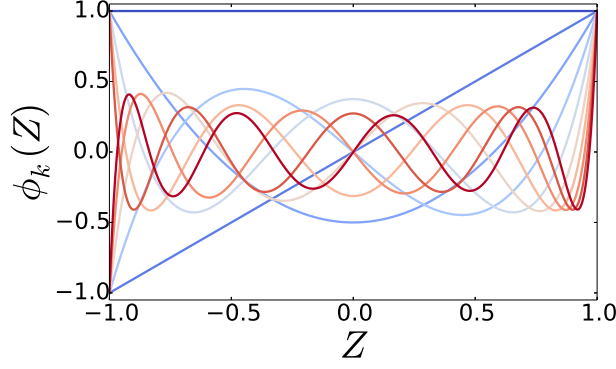


Figure 2.1: Univariate Legendre polynomials ϕ_k , $k = 1 \dots 10$.

the Φ_j are orthonormal, we have

$$y_j = \langle y, \Phi_j \rangle. \quad (2.5)$$

Note that one could also take $y(Z)$ to be a vector of several different aerodynamic quantities of interest: in this case, the coefficients y_i in the expansion (2.1) are vectors, and each component of y_i is determined by an equation such as (2.5), for the corresponding component of y .

The issue now is how we choose to approximate the projection integrals in (2.5). A commonly used approach is to use a quadrature method, in which the inner product is approximated discretely:

$$y_j = \langle y, \Phi_j \rangle \approx \sum_{k=1}^Q y(Z^{(k)}) \Phi_j(Z^{(k)}) w_k, \quad (2.6)$$

where $Z^{(k)}$ denote a set of Q quadrature nodes, with corresponding weights w_k . A possible choice is to use Gauss quadrature [35], in which the function $y(Z)$ is evaluated on a grid consisting of the tensor product of n separate 1-D quadrature point sets in parameter space (see Fig. 2.2 for an illustration). An attractive feature of Gauss quadrature is accuracy: by definition, a univariate Gauss quadrature rule using Q points is exact for any polynomial integrand of order $2Q - 1$. It can be shown [35] that the nodes which satisfy this property are given by the zeros of the Q^{th} basis polynomial (i.e., ϕ_Q). However, because the multi-

dimensional quadrature rule is a tensor product of univariate rules, this method does not scale well with large d , since the number of quadrature nodes grows exponentially with the dimension d .

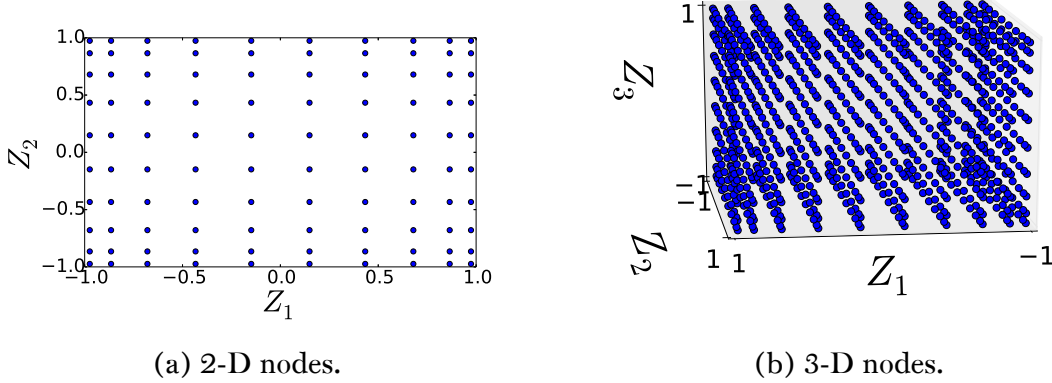


Figure 2.2: Gauss-Legendre nodes for $d = 2, 3$.

An alternative to Gauss quadrature is to use sparse grid methods[84, 55, 33], in which the total number of grid points used is lessened by using only a subset of the full tensor product (see Fig. 2.3 for an illustration). Any valid quadrature rule may in principle be used to construct a corresponding sparse grid rule; hence, the nodes need not be the zeros of the basis polynomials (as is the case in Gauss quadrature). In particular, nodal sets may be chosen which are nested, meaning that successive iterative grid refinements reuse previous coarse grid points. This is an attractive feature, since it is possible to refine a PC surrogate calculated on a coarse grid without “throwing away” those coarse grid points (which is generally not the case in Gauss quadrature).

In the studies contained in this thesis, we compute PC expansions by stochastic collocation using either Gauss quadrature or sparse grid construction (which will be noted appropriately). The sparse grid constructions are computed using DAKOTA, an open-source code for optimization and UQ developed by Sandia National Laboratory [1].

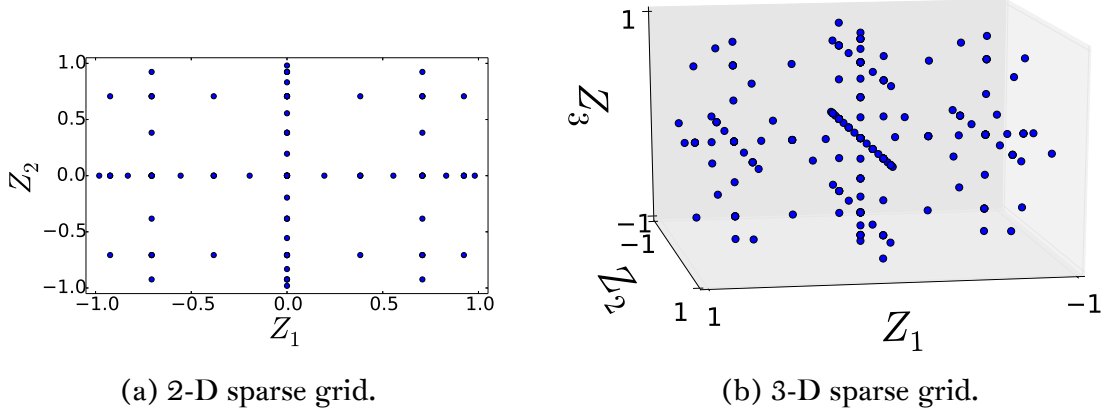


Figure 2.3: Smoylak sparse grid constructions, using Fejer nodes of level 4.

2.3 Statistics and Analysis of Variance

If we have a sufficiently accurate PC expansion for the observable, defined as in (2.1), then we may retrieve statistical moments through a few simple post-processing steps. For example, the mean is approximated by the expected value of the PC expansion: noting that $\Phi_0 = 1$, we have

$$\mu = \mathbb{E}[y] = \int_{\Gamma} y \rho(Z) dZ = \langle y, \Phi_0 \rangle = y_0. \quad (2.7)$$

Similarly, the variance can be approximated as

$$\sigma^2 = \mathbb{E}[(y - \mu)^2] = \langle y - y_0, y - y_0 \rangle = \sum_{|i|=1}^N y_i^2. \quad (2.8)$$

A nice feature of PC expansions is that the use of orthogonal polynomials makes it possible to analytically compute analysis of variance (ANOVA) metrics. For example, Sobol indices, which provide a metric of the relative “importance” of each of the uncertain parameters on the output, may be directly computed from the PC model coefficients [87]. Specifically, the “total” Sobol index T_i is defined as the fraction of the total variance con-

tributed by all those polynomials in the PC expansion which involve Z_i :

$$\begin{aligned} T_i &= \frac{\mathbb{E}[\text{Var}(y|Z_{-i})]}{\text{Var}(y)}, \quad i = 1, \dots, d \\ &= \frac{1}{\sigma^2} \sum_j y_j^2 \|\phi_j\|^2 \end{aligned} \tag{2.9}$$

where j in the above expression is understood to index only those terms in the PC expansion which involve parameter Z_i , Z_{-i} denotes all parameters except Z_i , $\mathbb{E}[\cdot]$ denotes the expected value, and $\text{Var}(\cdot)$ denotes the variance.

2.4 Overview of PCE Algorithms

Putting together the results of this section, we can outline a simple set of steps which describes how to implement a PCE-based uncertainty quantification, based on the stochastic collocation method:

Algorithm 1: PCE UQ: Full Tensor Grid

Input: Uncertain parameters $Z = (Z_1 \dots Z_d)$ with known PDF $\rho(Z)$

Output: PCE surrogate $y_N(Z) = \sum_{|i|=0}^N y_i \Phi_i(Z)$

(1) *Basis:* choose PC basis corresponding to $\rho(Z)$

(2) *Nodes/Weights:* compute the $(N + 1)$ point univariate Gauss-quadrature nodes/weights; form the d -dimensional tensor product

(3) *Simulation:* solve the deterministic simulation at each of the quadrature points

(4) *Reconstruction:* evaluate discrete projection integrals and solve for the PCE surrogate

(5) *Statistics:* calculate statistics/ANOVA from PCE surrogate

Algorithm 2: PCE UQ: Adaptive Sparse Grid

Input: Uncertain parameters $Z = (Z_1 \dots Z_d)$ with known PDF $\rho(Z)$

Output: PCE surrogate $y_N(Z) = \sum_{|i|=0}^N y_i \Phi_i(Z)$

(1) *Basis:* choose PC basis corresponding to $\rho(Z)$

(2) *Initialization:* compute initial coarse grid, set tolerance TOL

while $err < TOL$ **do**

 (3) *Simulation:* solve the deterministic simulation at each of the quadrature points

 (4) *Reconstruction:* evaluate discrete projection integrals and solve for the PCE surrogate

 (5) *Direction:* compute Sobol indices; next refinement direction is that which has the highest Sobol index

 (6) *Refinement:* compute refined grid points

 (7) *Convergence:* compute convergence metric err (e.g., \mathcal{L}_2 norm of the change in the response covariance matrix)

end

Chapter 3

User-Informed UQ

3.1 Introduction

The intent of this chapter is to provide an introduction to the ways in which the UQ techniques described in the previous chapter may be applied for the purposes of studying airfoil icing. As such, the focus is three-fold: first, we wish to introduce a few canonical classifications of airfoil/wing icing that occur in practice; second, we aim to parameterize shape variations within these classifications using a few geometric parameters; third, we will apply PCE techniques to quantify uncertainty in airfoil performance, and compare the results against Monte Carlo simulations. The goal of doing this is to give convincing evidence that PCE methods provide a computationally efficient and accurate framework for icing UQ.

A brief remark is in order regarding what this chapter is not. As we will see, the parameters that control the ice shape used in this chapter are devised on the basis of engineering intuition: we identify typical geometric scalings and translations that are commonly discussed in the icing literature, and use these as our parameter spaces. We do not attempt more sophisticated means for either identifying or modeling the input stochastic processes that control the ice shape. That subject is addressed in Chapter 4 (data-driven modeling

of ice shapes) and Chapter 5 (computational/physical governing parameters). Our focus here is on the PCE methods and how they apply to icing.

We structure this chapter in the following way. First, we provide a brief section regarding background information on the types of ice that we will be studying, particularly with respect to the ways in which these shapes may be parameterized. Next, we set up several UQ experiments and perform them with both Monte Carlo and PCE methods, and compare the results. We note that the bulk of this chapter has been published; see [28].

3.2 Wing Icing Classifications

The consensus in the literature is that wing icing may be conveniently divided into a few simple classifications. Each of these is briefly discussed in turn. The interested reader is referred to the literature for a more in-depth discussion[17].

3.2.1 Rime Ice

This type of ice accumulates under icing conditions characterized by a combination of colder temperatures and lower atmospheric liquid water content (LWC). Under such conditions, there is enough heat transfer to completely freeze all (or nearly all) of the water mass impinging on the wing from the freestream, and so the resulting shape closely follows the impingement distribution. Discussions and studies of the general effects of streamwise accretion on airfoil lift, drag, pressure distribution, stall characteristics, etc. can be found in [18, 5, 12, 19, 22].

Aerodynamically, the consensus is that this category of shapes presents less of an aerodynamic danger than the other categories, since the accretions smoothly follow the airfoil contour. For this reason, in this chapter, we do not study rime accretions. Instead, we focus on the types of accretion that represent more dangerous threats to aerodynamic performance.

3.2.2 Ice Roughness

This category refers to the development of ice elements that are small compared to other macroscopic ice accretions, but large enough to interfere with the boundary layer. These elements generally have heights at least equal to the height of the boundary layer (and in many cases much larger). Hence, they act as small-scale flow obstacles and induce separation on the order of the height of the roughness element. Ice roughness may occur as an early stage accretion, leading to the development of a large ice formation (e.g., horn or ridge), or it may occur on the surface of a large ice formation. The latter case has been empirically observed to occur in both glaze and rime accretions: areas with relatively low heat transfer (such as near the leading edge) tend to be smooth, while areas with higher rates of heat transfer tend to accumulate ice roughness.

Roughness elements can energize the boundary layer and promote heat transfer by increasing turbulent viscosity and mixing, and since convective heat transfer plays a major role in the ice shape, these elements can be crucial in determining the ice shape. It is for this reason that much research has been invested in understanding and predicting the development of ice with roughness. Hansman [37] first conjectured the connection between ice roughness and the role it plays in glaze ice shape development; Shin [82, 7] performed experiments to confirm this hypothesis and empirically characterize ice roughness distributions. Studies have been performed aimed at understanding the effects of ice roughness distributions on boundary layer transitioning [45, 44, 25, 26], describing the effects of roughness on airfoil lift [23, 54], and analyzing the regions of the airfoil surface most sensitive to roughness [16].

While ice roughness may lead to early trailing-edge separation, it typically does not cause the large scale separation bubbles characteristic of the more dangerous horn and ridge ice. Additionally, ice roughness exists on a length scale below that which can be modeled in our in-house meshing software; indeed, most computational icing codes treat

roughness using subgrid models (e.g., as extensions to a turbulence model). For these reasons, we will not discuss this situation any further in this chapter.

3.2.3 Ridge Ice

In order to combat ice development, de-icing mechanisms may be installed on lifting surfaces, the two most common of which are pneumatic boots (usually on smaller private airplanes) and thermal surface heating protection systems. However, these mechanisms are oftentimes limited to the tip of the leading edge and do not protect areas farther aft. If a significant amount of the impinging water mass is not removed completely, it can runback along the surface and refreeze in the unprotected areas, creating a sharp, discontinuous step, or “ridge” [63, 4] (see Fig. 3.1 for a visualization).

Because of its discontinuous nature, an ice ridge essentially acts as a flow obstacle, and has been investigated as such. In particular, it is often modeled with a quarter circle round geometry at some location aft of the stagnation point. The effect of such a geometry on the flowfield physics has been thoroughly studied [52, 51, 20, 50, 36, 53]. It is generally acknowledged that the major parameters at play in this scenario are the size and location of the quarter-round; spanwise uniformity is another factor in 3-D icing studies. Depending on the size and location, the ridge may cause either a large-scale separation bubble, or complete separation. It is therefore a particularly dangerous type of accretion, and hence is one of the categories we will study in this chapter.

3.2.4 Horn Ice

Horn ice forms in icing conditions which are relatively warmer with higher amounts of liquid water content in the free-stream. It can be understood in terms of a division between areas near the stagnation point of the airfoil, for which rates of ice accretion are lower, and areas aft of that, which experience higher rates. This differential in ice accretion rates

is fundamentally due to a similar differential in convective heat transfer on the surface, which can be explained using elemental boundary layer theory. The Reynolds Analogy implies that the local rate of convective heat transfer is proportional to the local skin friction [77, 80]. The local skin friction is influenced by the mean flow gradient inside the boundary layer, as well as Reynolds stresses from turbulent eddies. Local surface velocity on an airplane wing spikes at areas just aft of the stagnation point, which leads to higher flow gradients and provides a partial explanation of the differential heat transfer. The other mechanism is provided by ice roughness elements, which (as previously discussed) increase the eddy viscosity and tend to be more highly concentrated aft of the stagnation point [38, 37]. The combination of these two aerodynamic effects is responsible for the heat transfer distributions that give rise to horn ice.

Horn ice formations are large leading edge protrusions that can disrupt the flow near the leading edge upper surface, which is where most of the airfoil lift is generated. The consensus amongst the icing community is that the parameters that dominate aerodynamic performance are the horn height (normal to the airfoil surface), angle with respect to the chordline, and location. Various studies have been performed investigating the relative effects of these parameters on the leading edge separation bubble aft of the horn and on airfoil performance [70, 71, 72, 76, 74, 73, 46]. The ability of the horn to separate the flow near the leading edge makes this type of icing particularly dangerous, and it is for this reason that horn ice is the other category of icing that we will study in this chapter.

3.3 Application of PCE to Airfoil Icing

The PCE stochastic collocation method presented in the previous chapter is applied to the ridge and horn ice problems in this section, with the goal of quantifying uncertainty in airfoil aerodynamic performance metrics, such as max lift coefficient and stall angle of attack. The results are compared to Monte Carlo simulations.

Both the ridge and horn cases are modeled as 2-D phenomena in the following studies. In both cases, we parameterize the variation in the ice shape using two independent parameters. This parameterization is discussed in what follows.

We model ridge ice shapes as backward-facing quarter circle rounds, which are parameterized by the radius R of the round and the location S_r aft of the airfoil leading edge (see Fig. 3.1). The radius R governs uncertainty in the size of the ridge ice, while the position S_r governs uncertainty in the position at which the ridge forms aft of the deicing boot.

We select a profile for the mean horn geometry from Papadakis[73]. The two independent stochastic parameters that govern perturbations from the mean shape are the height H (normal to the airfoil surface) and separation distance S_h of the horn. The mean horn profile used is shown in Fig. 3.1.

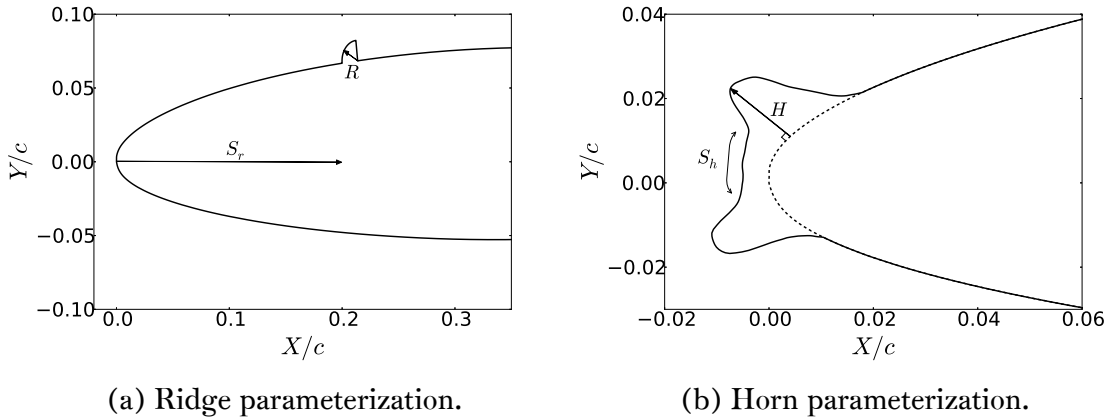


Figure 3.1: Parameterization convention for the ridge and horn ice scenarios. The figure at the right represents the mean horn ice shape used in this work, obtained from Papadakis[73].

3.3.1 Selection of Flow Solver and Aerodynamic Conditions

Practical implementation of uncertainty quantification analysis relies upon fast and accurate flow solvers. In order to evaluate the aerodynamic characteristics for the different ice shapes considered, we need a reliable, tested flow solver. In this thesis, we use

FLO103, a well-validated, in-house code for the solution of the two-dimensional Reynolds Averaged Navier-Stokes (RANS) equations developed over the course of many years by Martinelli and Jameson [59, 91, 90].

A one-equation Spalart-Allmaras turbulence model [85] provides closure for the RANS, which is capable of accurately modeling mildly separated flow near the stall regime. The discretization of the spatial operators is carried out by using a second order cell-centered finite-volume method in which the viscous stresses are computed using a discrete form of Gauss’ theorem. The key to the flow solver efficiency is a full approximation multigrid time-stepping scheme, which accelerates the rate of convergence to a steady state. All calculations are performed on a boundary-fitted, structured “C”-mesh, generated with an in-house hyperbolic grid generator. These meshes contain 513 cell-centered nodes in the direction along the airfoil, and 129 cell-centered nodes in the direction normal to the airfoil, with approximately 65 nodes inside the boundary layer.

In our problem setup, the airfoil profile used is the NACA 63A213 at a Reynolds number of $Re = 4.5 \times 10^5$ and Mach number of $M = 0.21$. These conditions are chosen for several reasons. First, we wish to verify our flow solver against published experimental data from the ice accretion literature, and the conditions cited were investigated by Papadakis [73]. As shown in Fig. 3.2, FLO103 is able to reproduce to high accuracy both the airfoil lift curve and pressure distribution over the range of angles of attack for which there are steady solutions.

Second, we wish to select a lower Reynolds number than that encountered in flight. This is because higher Reynolds numbers would require finer grid resolutions, which would require longer solution times. Because the foci of this paper are the icing parameterization and UQ methodologies, and because we verify all of our PCE solutions against quasi-Monte Carlo solutions (which are computationally laborious), we select a lower Reynolds number to ease the computational burden. Note, however, that the computa-

tional approaches used in this paper are, in principle, applicable to the higher Reynolds numbers that are representative of flight conditions.

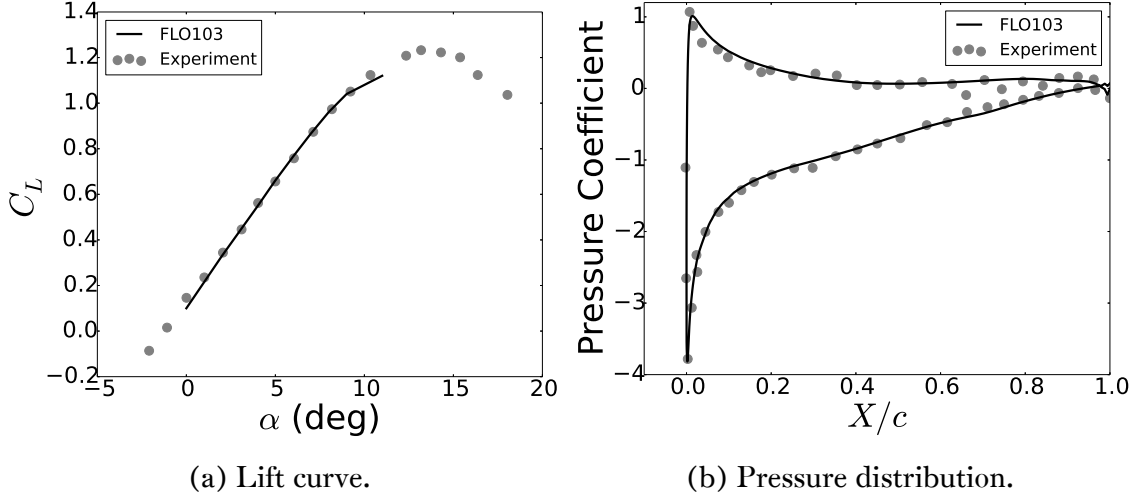


Figure 3.2: Comparisons of the lift curve (*left*) and pressure distribution at $\alpha = 8^\circ$ (*right*) for a NACA 63A213 airfoil at $Re = 4.5 \times 10^5$ and $M = 0.21$. The experimental data are provided in [73].

3.3.2 Ridge Ice Case

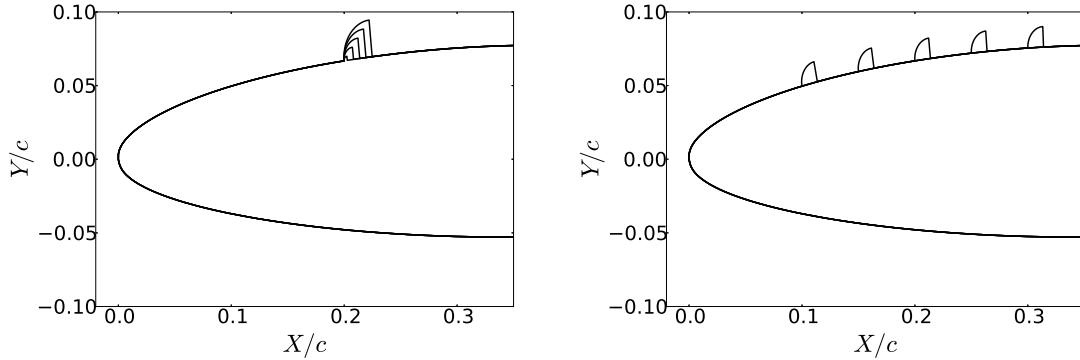


Figure 3.3: *Left*: Ridge shapes corresponding to $R = \mu_R + \{-2, -1, 0, 1, 2\}\sigma_R$ with $\sigma_R = 40\%$ (and with $S_r = \mu_{S_r}$). *Right*: Ridge shapes corresponding to $S_r = \mu_{S_r} + \{-2, -1, 0, 1, 2\}\sigma_{S_r}$ with $\sigma_{S_r} = 5\%$ (and with $R = \mu_R$).

We begin with the ridge ice case and quantify uncertainty in three separate aerodynamic performance metrics: $C_{L_{\max}}$, α_{\max} , and L/D_{\max} . In each of the cases, there are two

uncertain input parameters: ridge radius, R , and ridge position, S_r . Both of these parameters assume independent Gaussian distributions, where $\mu_R = 1.39\%$ of the chord, and $\mu_{S_r} = 20\%$ of the chord; these were selected in agreement with Bragg[17]. We specify σ_R as a percentage of μ_R and σ_{S_r} as a percentage of the chord length. We present two UQ investigations in which the standard deviation pairs (σ_R, σ_{S_r}) take the values $(10\%, 1.25\%)$ and $(40\%, 5\%)$. Fig 3.3 shows the independent effects of the ridge parameters on ridge size and position.

As mentioned, we compare our PCE methods to quasi-Monte Carlo simulations [62]. The basic algorithm used in this scheme involves inverse transform sampling (see the work of Devroye [30]) for selecting samples. The cumulative distribution space is sampled using an ergodic dynamical system with an invariant measure that matches the desired distribution. This algorithm has a notable advantage of efficiency over standard Monte Carlo algorithms. Numerical experiments suggest that samples obtained through this method converge in distribution at a rate proportional to $1/N$, where N is the number of samples. This is in contrast to standard Monte Carlo, in which the rate of convergence is $1/\sqrt{N}$.

Table 3.1: Comparison of Statistical Moments for Monte Carlo and PCE: $(\sigma_R, \sigma_{S_r}) = (10\%, 1.25\%)$

	$C_{L_{\max}}$		α_{\max} (deg)		L/D_{\max}	
	MC	PCE	MC	PCE	MC	PCE
Mean	0.87	0.86	9.8	9.6	16.6	16.5
Variance	0.0024	0.0025	0.16	0.13	4.0	4.4
Skewness	-0.33	-0.37	-0.50	-0.96	0.56	0.40
Kurtosis	2.3	2.2	2.9	2.7	3.0	2.6

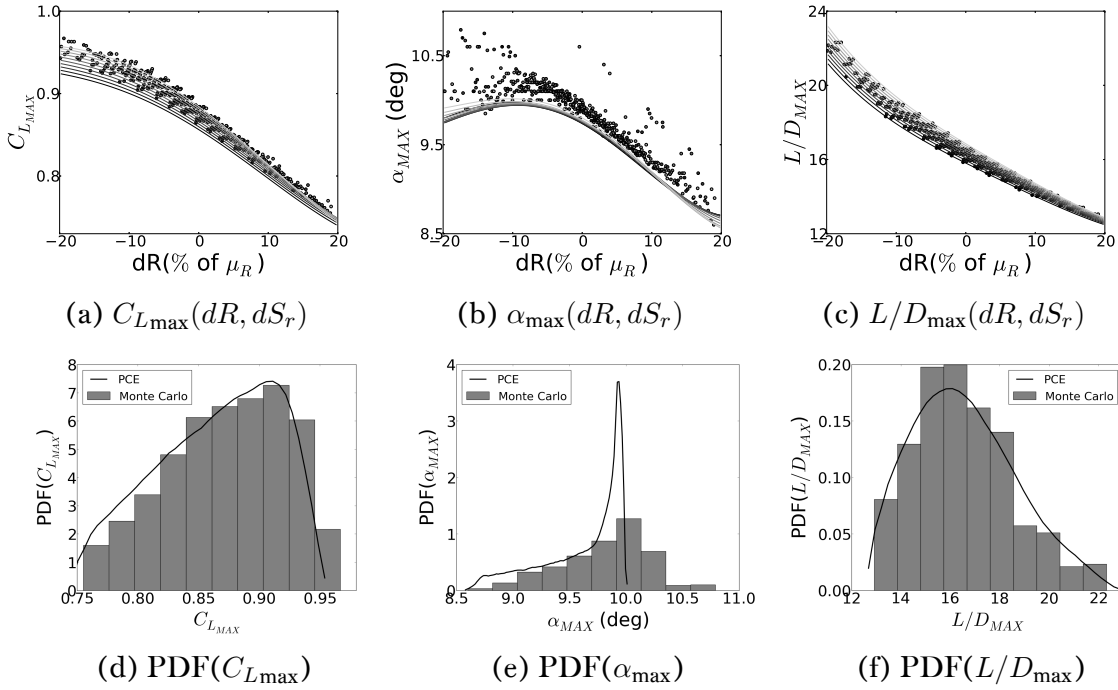


Figure 3.4: Comparisons of PCE surrogate and quasi Monte Carlo results, including output maps (*top*) and statistics (*bottom*) for "small" ridge uncertainty.

Table 3.2: Comparison of Statistical Moments for Monte Carlo and PCE: $(\sigma_R, \sigma_{S_r}) = (40\%, 5\%)$

	$C_{L_{\max}}$		α_{\max} (deg)		L/D_{\max}	
	MC	PCE	MC	PCE	MC	PCE
Mean	0.85	0.85	9.2	9.2	19.4	19.7
Variance	0.020	0.022	0.72	0.69	110	120
Skewness	-0.29	-0.38	-0.60	-1.4	1.3	1.2
Kurtosis	2.1	2.2	2.9	4.4	4.3	3.7

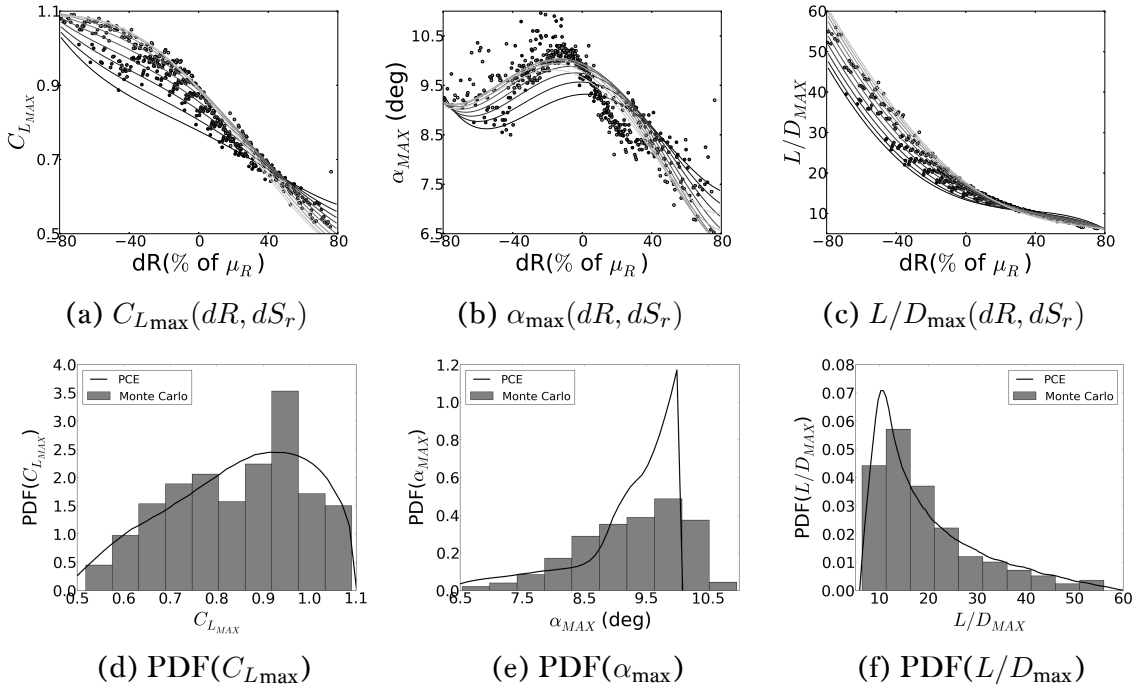


Figure 3.5: Comparisons of PCE surrogate and quasi Monte Carlo results, including output maps (*top*) and statistics (*bottom*) for “large” ridge uncertainty.

The top rows of Fig. 3.4–3.5 compare the surrogate maps created using the PCE stochastic collocation method to results obtained through quasi Monte Carlo sampling for $C_{L_{\max}}$, α_{\max} , and L/D_{\max} . In these figures, the gray scale is chosen to represent different values of dS_r (dark to light transition indicates increasing values of dS_r). The bottom rows present normalized histograms of the quasi Monte Carlo results and compare them to the results of propagating the input distribution through the PCE surrogate map. Ta-

bles 3.1 and 3.2 give quantitative comparisons of the statistics for the Monte Carlo and PCE method results for the ridge studies. The quasi Monte Carlo method used 500 samples per case, whereas the PCE method utilized 4th order polynomial expansions and hence required a 5×5 collocation mesh. It should be noted that the Jacobi polynomials—not the Hermite polynomials—were used as the basis in the PCE scheme. There are two reasons for this choice. First, the input distribution in all cases was a truncated Gaussian (i.e., both dR and dS_r were truncated at $\pm 2\sigma$). Second, use of the Hermite polynomials might have presented a practical problem, as the collocation nodes tend to lay far out in the tails of the distribution. Sampling at these extreme nodes (corresponding to extreme ridge radii and/or positions) might have presented a problem for the RANS solver. Hence, we used a Jacobi expansion which approximates a truncated Gaussian in distribution and does not require sampling at extreme positions. Specifically, denoting the univariate Jacobi polynomials as $\{J_i^{(\alpha,\beta)}\}$, we used the linear expansion $\sum_{i=0}^5 a_i J_i^{(2,2)}$ to approximate a truncated Gaussian. See Xiu[104] for an in-depth discussion and for the numerical values of the coefficients a_i .

Examining the data in Fig. 3.4 and 3.5, we observe that the dominant parameter in all cases is the ridge radius, R ; the ridge position S_r has a relatively smaller effect on the metrics. A larger ridge size (i.e., increasing dR) leads to a monotonic decrease in $C_{L_{\max}}$ and L/D_{\max} . A ridge which is closer to the leading edge (i.e., decreasing dS_r) also results in a monotonic decrease in $C_{L_{\max}}$ and L/D_{\max} . This is quite intuitive, since a large ridge radius tends to promote large scale flow separation at lower angles of attack, and a ridge closer to the leading edge disrupts more of the flow over the airfoil.

It is also clear that the agreement between the PCE and Monte Carlo schemes is best for the metrics $C_{L_{\max}}$ and L/D_{\max} ; agreement for α_{\max} is less satisfactory. This can be attributed to the smoothness of the maps. The maps $C_{L_{\max}}(dR, dS_r)$ and $L/D_{\max}(dR, dS_r)$ are both very smooth—in spectral terms, most of the energy of the PCE expansions which approximate them is contained in the low order modes.

In contrast, $\alpha_{\max}(dR, dS_r)$ is not as smooth, as can be seen in the Monte Carlo results, particularly at extreme values of dR . This reflects a nontrivial amount of energy in higher order spectral terms which are neglected in our 4th order PCE expansions. However, this should be considered in context with how we obtain the values of α_{\max} . Our algorithm for detecting $C_{L\max}$ and α_{\max} involves testing the curvature of discrete points on the lift curve—if the curvature exceeds some calibrated bounds, then stall is assumed to have occurred, and $C_{L\max}$ and α_{\max} are interpolated from the discrete points. As one might expect, this method works well for detecting $C_{L\max}$, since that quantity is relatively constant near stall. However, it does not perform as robustly for α_{\max} , since that quantity is much more sensitive to perturbations from the true value near stall. Thus, in a sense, we do not even wish to recover the higher order modes for α_{\max} , since these reflect the lack of robustness of our algorithm instead of the general trends in the parameter space.

It should also be noted that if greater refinement of the PCE results for α_{\max} is desired, this could be achieved either by retaining higher order terms in the PCE expansion of α_{\max} (p -refinement), or by dividing the stochastic space into smaller, discrete elements (h -refinement), or a combination of these tactics. In the next section on horn icing, we explicitly demonstrate how to apply stochastic h -refinement to improve PCE results.

The effect of an increasing ridge radius on the flowfield is shown in Fig. 3.6. This figure shows contours of x -momentum as well as streamlines of the flow for increasing ridge radius. The solution for the smallest ridge closely resembles that of the clean airfoil. A large-scale separation bubble forms aft of the ridge for the medium sized case before reattaching to the airfoil surface; this separation bubble reduces lift and increases drag. For the largest ridge, the separation bubble caused by the ridge is so large that the flow does not reattach, leading to early stall. This figure demonstrates how different our UQ study is from a sensitivity study, in which the effect of only small perturbations is investigated. In contrast, we are investigating a large amount of uncertainty in ridge size and position, and this leads to a large spectrum of possible flowfields.

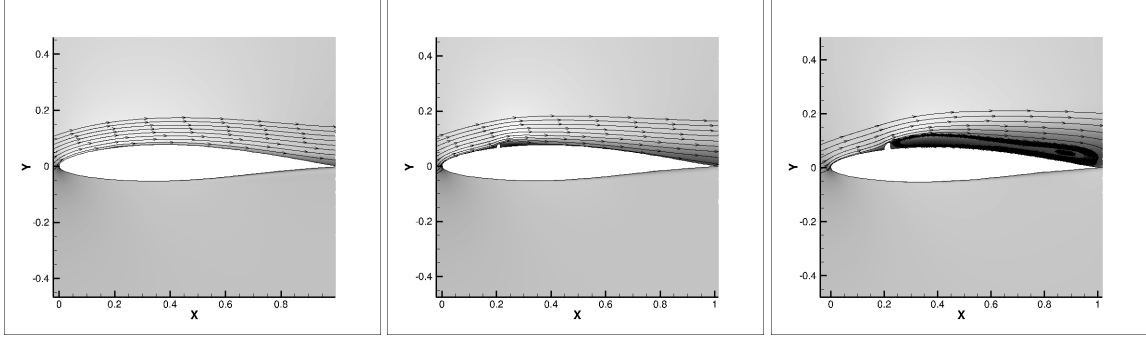


Figure 3.6: Flowfield for ridge ice cases at $\alpha = 6^\circ$ in which $dR = -80, 0$, and 80% of μ_R .

3.3.3 Horn Ice Case

We approach the horn ice UQ problem in the following way. First, we identify a canonical horn ice shape, which was originally produced by NASA’s LEWICE icing code for 22.5 minutes of ice accretion on a NACA 63A213 airfoil (see Papadakis[73] for more details). That shape is identified in Fig. 3.1. We allow a scaling of the horn height by a parameter which we denote as h , with $h \in [0,1]$ (0 corresponds to the clean airfoil, and 1 corresponds to the horn height given in the figure). We also allow a scaling of the inter-horn separation distance by a parameter which we denote as s , with $s \in [0.1,1.9]$ (that is, the inter-horn separation distance in the figure can vary by $\pm 90\%$ of its nominal value). We wish to investigate uncertainty in aerodynamic performance metrics given uncertainty in h and s , where h is the positive half of the Gaussian $\mathcal{N}(0, 0.5^2)$ (which we will denote as $\mathcal{N}_{1/2}^+(0, 0.5^2)$), and s is the Gaussian $\mathcal{N}(1, 0.45^2)$. It should be noted that both parameters are truncated at 2σ distance from the mean. The horn variations produced by these parameters are shown in Fig. 3.7.

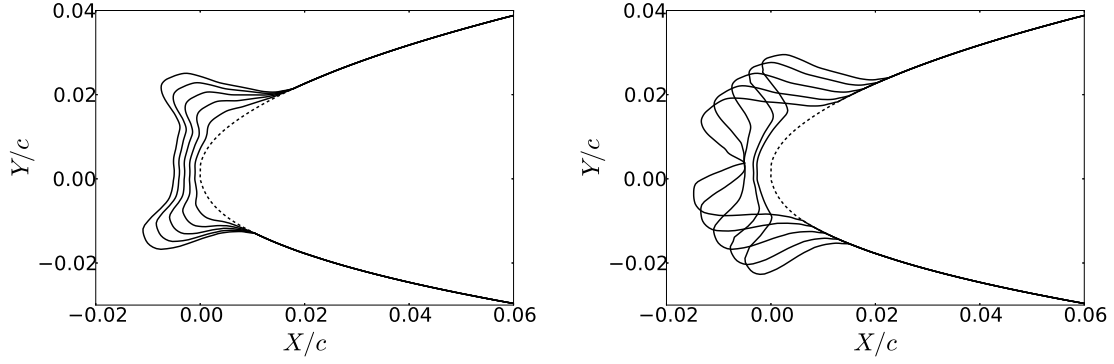


Figure 3.7: *Left:* Horn shapes produced by variation of the parameter $h \in \{0.2, 0.4, 0.6, 0.8, 1\}$ ($s = 1$). *Right:* Horn shapes produced by variation of the parameter $s \in \{0.1, 0.5, 1, 1.5, 1.9\}$ ($h = 1$).

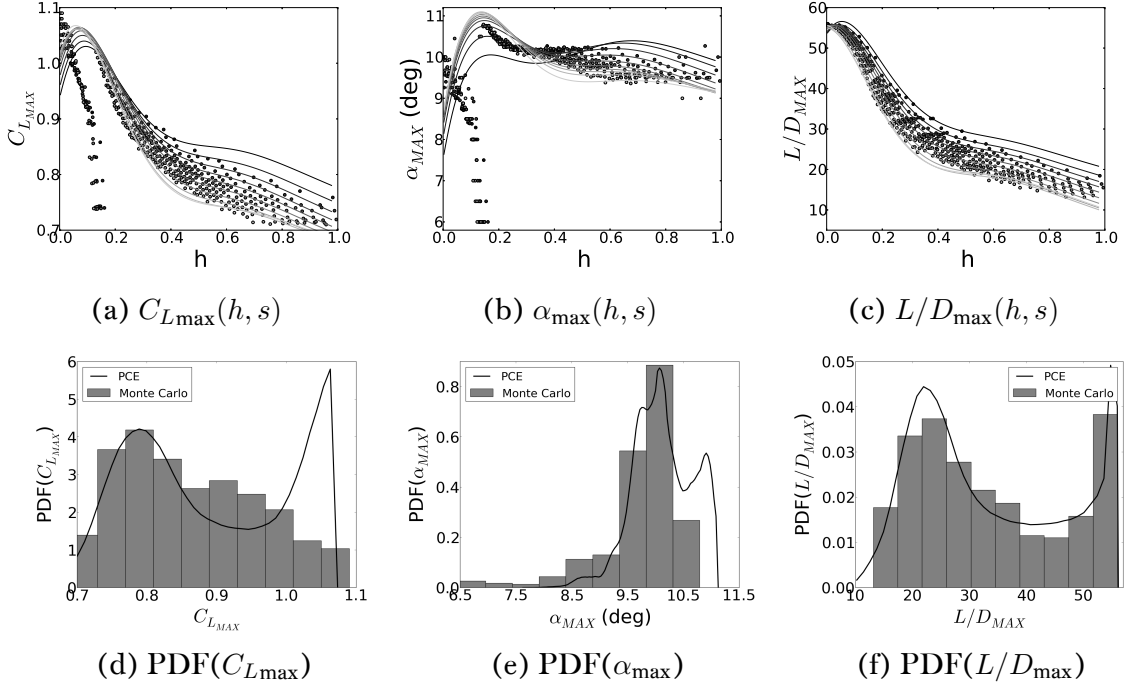


Figure 3.8: Comparisons of PCE surrogate and quasi Monte Carlo results, including output maps (*top*) and statistics (*bottom*) for horn uncertainty.

It is also important to point out that the half-Gaussian distribution that we are using does not belong to any of the basic types of distributions typically used with polynomial chaos expansions (i.e., Gaussian, uniform, beta, etc.). To apply our methodology, we must represent the half-Gaussian distribution in terms of our chosen basis functions. More

Table 3.3: Comparison of Statistical Moments for Monte Carlo and PCE: $h = \mathcal{N}_{1/2^+}(0, 0.5^2)$, $s = \mathcal{N}(1, 0.45^2)$

	$C_{L\max}$		α_{\max} (deg)		L/D_{\max}	
	MC	PCE	MC	PCE	MC	PCE
Mean	0.86	0.92	9.7	10.2	34	33
Variance	0.0097	0.014	0.84	0.29	170	170
Skewness	0.35	-0.56	-2.3	-0.43	0.38	0.38
Kurtosis	2.1	1.6	8.8	3.1	1.8	1.8

precisely, we need to determine the coefficients of $\xi \approx \sum_{i=1}^M a_i \phi_i(Z)$, where ξ is a random variable, the PDF of which is our custom input distribution (which is the half-Gaussian in our problem), and $\{\phi_i(Z)\}_{i=1}^M$ are the basis polynomials for one of the basic polynomial chaos distributions (e.g., uniform). This mapping can be approximated by probability space projection; for details on this procedure, see the work of Xiu and Karniadakis [104].

Table 3.3 gives quantitative comparisons of the statistics for the Monte Carlo and PCE results for this horn study. Fig. 3.8 compares the PCE results to the Monte Carlo results. The gray scale in the top row of plots is chosen to represent different values of s (dark to light transition indicates increasing values of s). As can be clearly seen in Fig. 3.8, the dominant parameter is the horn height scale (h); variations with horn separation distance (s) are smaller by comparison. Unlike the ridge ice cases, there are discontinuities in the maps for both $C_{L\max}(h, s)$ and $\alpha_{\max}(h, s)$. These discontinuities may arise because we are seeking only steady solutions with our flow solver. For horn heights h around 0.18, it is likely that the natural flow exhibits unsteady vortex shedding and does not correspond to a steady solution. This conjecture could be tested by running time-resolved calculations for that region of parameter space; however, we leave this issue for future work since it does not affect the demonstration or merits of the basic UQ approach and methodology that is the focus of the present work.

The polynomial chaos expansions, which are linear combinations of smooth polynomials, obviously cannot resolve such discontinuities. This results in errors in both the

PCE surrogate maps and in the output PDFs for $C_{L\max}$ and α_{\max} . This result illustrates an important caveat in the use of our PCE approach: if discontinuities are present in the response surface, then they can significantly impact the accuracy of the results. Regardless of whether the discontinuities are physical, this raises an important question worth discussing: can anything be done to improve the PCE results when discontinuities are present? One method well suited for this problem follows the traditional notions of h -refinement. The general idea is to divide the stochastic collocation mesh into separate regions and then implement PCE separately in each of those regions (while taking care to preserve total probability). This results in piecewise-smooth expansions for our response surfaces and results in significantly better agreement between the PCE and Monte Carlo results, as shown in Fig. 3.9 and Table 3.4.

In our problem, the best approach is to divide the collocation mesh at the discontinuity revealed by the Monte Carlo results. In practice, when Monte Carlo results are not available, one would have to resort to an adaptive scheme, in which the PCE surrogate is computed using a coarse mesh, and then this mesh is sequentially divided if local sensitivities (or total variance) exceed some preset threshold value. For more information on this approach, see the work of Wan and Karniadakis [98].

Table 3.4: Comparison of Statistical Moments for Monte Carlo and Multi-Element PCE:
 $h = \mathcal{N}_{1/2^+}(0, 0.5^2)$, $s = \mathcal{N}(1, 0.45^2)$

	$C_{L\max}$		α_{\max} (deg)	
	MC	MEPCE	MC	MEPCE
Mean	0.86	0.85	9.7	9.9
Variance	0.0097	0.010	0.84	0.71
Skewness	0.35	0.50	-2.3	-2.5
Kurtosis	2.1	2.3	8.8	9.0

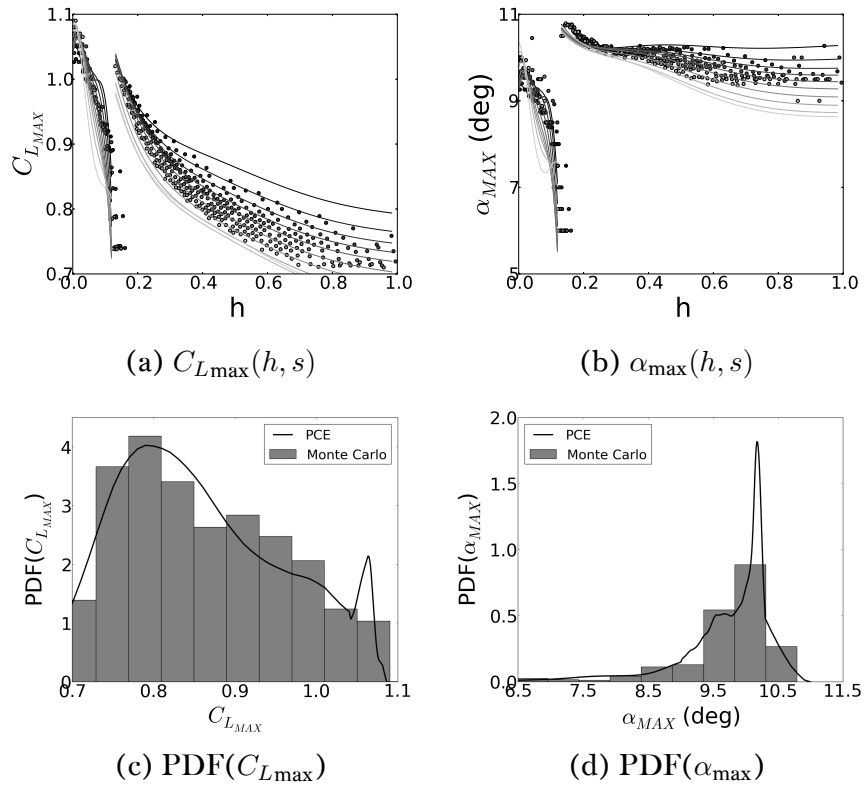


Figure 3.9: Comparisons of multi-element PCE surrogate and quasi Monte Carlo results, including output maps (*top*) and statistics (*bottom*) for horn uncertainty.

3.4 Summary

Wing icing is not only dangerous to pilots, it is a complex physics problem which is subject to a large amount of uncertainty. Quantifying the exact effects of this uncertainty on airplane performance is hence of great importance to airplane safety.

This chapter demonstrated the utility of polynomial chaos expansion (PCE) methods as a fast and accurate method for quantifying the effects on airfoil performance of ice shape uncertainty. The main advantage of this approach is speed and efficiency; each of the PCE results in this paper required an order of magnitude fewer samples than those required in the Monte Carlo-based schemes because of the efficiency of Gaussian quadrature. It is hoped that improvements in icing UQ can contribute toward improved safety regulations and protocols for pilots and a mitigation of icing-related accidents.

Chapter 4

Data-Driven UQ

4.1 Introduction

The previous chapter provides us with convincing evidence that PCE methods can be applied to efficiently and accurately quantify uncertainty in aerodynamic performance, given parameterized uncertainty in the ice shape. As we noted, however, the shape parameters utilized in the previous chapter were devised on the basis of the engineering intuition and on the collective experience of the icing community. One advantage of this approach is that it enables us to identify the effects of the scaling/translation parameters that dominate the aerodynamic characteristics without spending too much effort modeling actual shape variations. However, this can also be seen as a weakness, in the sense that the ice shapes studied do not reflect actual experimental/computational ice shapes, which raises questions about whether the entire UQ method can be extended to analyze “real” data.

In this chapter, we intend to address these concerns by deriving sets of ice shape parameters from databases of actual ice shapes. In this way, we shift toward a “data-driven” approach, in which we apply techniques from low-dimensional modeling to capture as much of the variation in a given dataset of ice shapes as possible with as few parameters as possible. Given a database of ice shapes, we first identify a low-dimensional set of shape

parameters by using the Proper Orthogonal Decomposition (POD) of the dataset. This gives us both a method for generating and studying artificial ice shapes as well as a way to assess the statistical variation of our dataset.

Based on the variation in the data, we select a parameterized range in POD space to perform uncertainty quantification (UQ). We generate artificial ice shapes by taking linear combinations of the POD modes, and study the effects on aerodynamics by meshing the resultant airfoil and computing its aerodynamic properties using our in-house flow solver (FLO103). In order to perform the UQ efficiently, we use either adaptive sparse grid polynomial chaos or Latin Hypercube sampling (LHS).

We show how the spatial modes generated by POD can be linked to physical information about the underlying icing conditions (e.g., temperature, liquid water content). This provides a means for statistically correlating the icing physics to the ice shape. The result of this is a statistical model of ice accretion, entirely driven by data, which may be used to benchmark or improve deterministic numerical ice accretion codes.

We will also explore the concept of applying clustering tools to the icing context. This can be useful if, for example, there is “too much” variation in a database – that is, the ice shapes contained in a database could vary so much from one another that many shape parameters are required to faithfully represent them. We will find that there sometimes exist subsets of ice shapes hidden within a database that can be grouped together on the basis of mutual similarity. Segregating a large database of disparate shapes into smaller subsets of closely related shapes by means of clustering algorithms will prove to be both an effective method for reducing the number of shape parameters, and a tool for empirical ice shape classification.

This chapter is organized as follows. First, we show how to identify low-dimensional models for icing datasets. These models will form the input parameter space for our UQ studies. Next, we apply the shape modeling approach to a computational dataset, and then perform a UQ study on the resulting parameter space. Next, we apply the shape

modeling approach to an experimental ice shape database. We link physical information to this database and produce a purely data-driven model of ice accretion. We then explore the concept of clustering the database into smaller subgroups. We then show how this may be done, and perform UQ on the resulting subgroups. We note that some of this material can be found in our recent AIAA paper [29].

4.2 Low-Dimensional Modeling using POD

The space of all possible ice shapes contains an infinite number of degrees of freedom. Aside from the fact that such a space is impossible to completely study, it is also not within our interests to study each and every shape perturbation that could ever possibly occur; rather, we wish to restrict our study and the parameters that govern it to *likely* ice shapes. The approach taken here is to consider a subspace of possible ice shapes, determined from data, either from simulations (§4.3) or experiments (§4.4). In particular, for a given dataset of ice shapes, we determine a low-dimensional subspace that optimally spans the data using Proper Orthogonal Decomposition (POD) [42].

Denote the vector of parameters governing a particular ice shape by $x \in \mathbb{R}^N$ (we will have more to say about what N is and how we get it soon). We wish to approximate any x as a linear combination of some basis vectors ψ_i , called POD modes:

$$x \approx \sum_{i=1}^P a_i \psi_i \quad (4.1)$$

POD gives us one way of generating the basis $\{\psi_i\}$ from the data. A key property is that the basis identified by POD will be able to represent the dataset better than any other linear basis, in the sense of projection error (using the standard Euclidean norm). The POD modes are determined as follows. Let x_j , $j = 1 \dots M$ denote the elements in our

dataset, and let $\mathbf{X} \in \mathbb{R}^{N \times M}$ be the matrix with x_j as its columns:

$$\mathbf{X} = \begin{bmatrix} | & & | \\ x_1 & \cdots & x_M \\ | & & | \end{bmatrix} \quad (4.2)$$

The POD modes ψ_i are then given by the left singular vectors of the singular value decomposition $\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$ (i.e., the columns of \mathbf{U}). The POD eigenvalues (i.e., the diagonal elements of $\mathbf{\Sigma}$) give an indication of how much of the statistical variation of the dataset is accounted for by each of the modes. When \mathbf{X} is large (as it can be in our applications), we can use the “method of snapshots” [83, 42] to efficiently calculate the POD modes/eigenvalues.

Using POD to model the ice dataset has effectively generated a (low-dimensional) parameter space to explore, along with coordinates of the data points in that space. In this way, we have automatically converted a database of ice shapes into a parameterized UQ problem. Casting this as a UQ problem has several potential benefits and uses:

- A linearly parameterized description of the ice gives us an easy and systematic method of generating new ice shapes that effectively interpolate the shapes present in our database. This makes it possible to produce and study a wide range of shapes.
- UQ tools allow us to compute and analyze the statistical relationships between our responses (aerodynamics) and our inputs (ice shapes). For example, we can look at the effect of different input distributions (e.g. uniform, Gaussian, etc.), correlations between POD modes and lift coefficients, output statistics, etc.
- We can use UQ tools to produce a surrogate model of the input-output behavior – using PCE, we obtain a polynomial mapping between the POD modes and the aerodynamics. This can be advantageous as a predictive tool: if one wishes to know the aerodynamics of a particular ice shape that has not been studied yet, one could

compute it by simply evaluating the surrogate model (assuming this ice shape is in the span of the POD basis).

Furthermore, using POD to generate our parameterization of the ice is advantageous for a few reasons:

- The POD basis outperforms any other possible linear basis of the same dimension for representing our data in the sense of projection error. In this way, it is an optimal parameterization.
- The POD coordinates are linearly uncorrelated (since the modes are orthonormal). This justifies an assumption of mutual independence amongst the parameters in our UQ study, which underlies the UQ methods we will be using.
- The POD is a general method for data-processing, which makes it amenable to analyzing other ice shape databases that we have not yet considered. Thus, our approach could be applied to other datasets as well.

As previously noted, the UQ methods that we will be using need to be efficient, since we have moderately high dimensional parameter spaces to explore. In this chapter, we use a combination of sparse grid PCE and Latin Hypercube sampling to perform UQ.

4.3 Simulation Database

In this section, we give our first example of how the techniques just discussed may be applied to an ice shape dataset from the literature. For this example, our data consists of cross-sections from an icing simulation performed on a 3D swept wing. This data and the research related to it is described in detail in Broeren et. al. [21] and is shown in Figure 4.1.

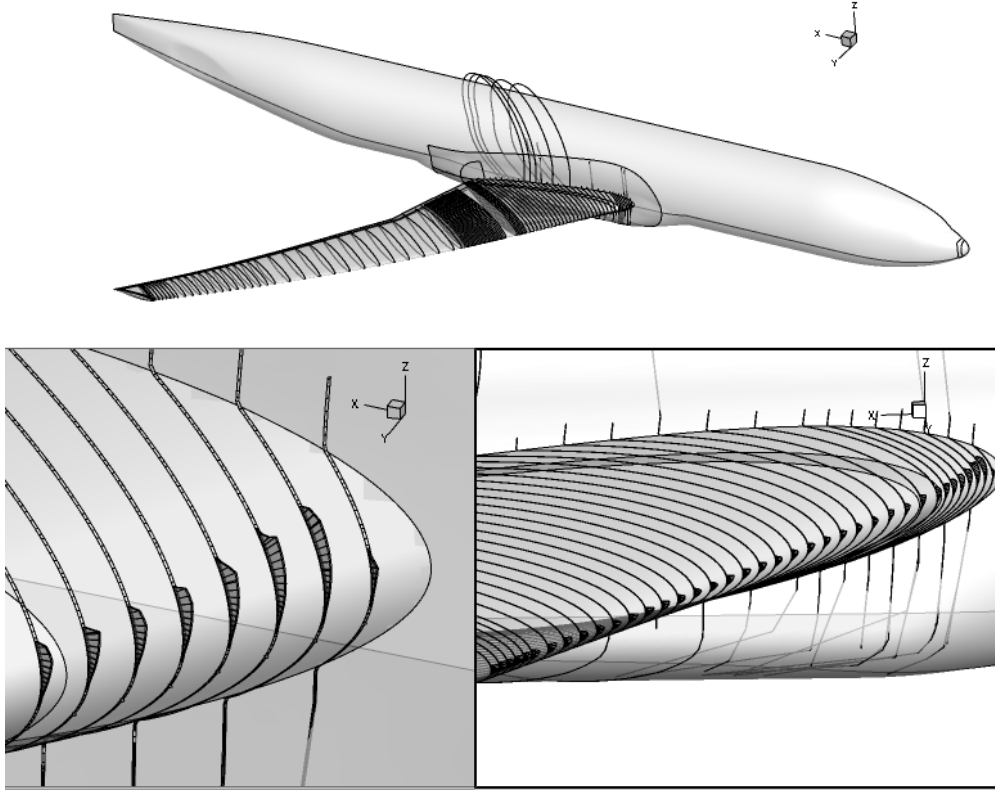


Figure 4.1: Horn ice accretion case on the NASA CRM, from Broeren et. al.[21].

The model geometry used in this data set was the NASA Common Research Model (CRM) at 65% scale. The ice accretion model used was LEWICE3D, which is a NASA's 3D icing code. This study represents 45 minutes of accretion time at an altitude of 10,000 ft and free stream velocity of 232 knots. The static temperature was -4°C , the mean volumetric diameter (MVD) was $20\text{ }\mu\text{m}$, and the liquid water content (LWC) was 0.55 g/m^3 .

4.3.1 Modeling using POD

Our first objective is to apply the POD machinery to the data set consisting of 95 individual horn ice cross sections on the CRM wing, with the objective of identifying the most important ice shape features. We will first do a preprocessing step on our data – we will map all of the airfoil cross sections into s -coordinates (arc length coordinates, relative to

the airfoil leading edge), so that the ice shapes may be represented as a function of a single variable.

Let us denote the horn ice height (in s -coordinates, normal to the airfoil) as $N_k(s)$, where k indexes the cross-sections. A POD representation of this dataset will take the following expansion form:

$$N_k(s) = \overline{N}(s) + \sum_{i=1}^M c_{i_k} \psi_i(s), \quad k = 1, \dots, S. \quad (4.3)$$

Here, $\overline{N}(s)$ is the height (in s -coordinates) of the mean ice shape, c_i is the i^{th} POD coefficient, and $\psi_i(s)$ is the i^{th} POD mode, and S is the number of ice shapes.

Fig. 4.2a shows the dataset. By inspection, we see that much of the variability in shape is due to differences in width, position, and height. Physically, these differences are due to the spanwise geometric variation of the 2D cross-sections of CRM wing, as well as 3D crossflow along the wing and boundary condition effects (i.e., effects of having a wall at the root and downwash at the tip). We can account for the spanwise geometric variations by scaling/shifting each of the individual ice shapes to fit a template shape, which produces Fig. 4.2b. The point of doing this is to separate variations in *scaling* from differences in *shape*. As we will see, this will be reflected in our parameterization of the ice, in which three of the parameters specify scaling, and the two POD modes specify shape perturbations.

If we pre-process the data with these scalings/shiftings, the POD expansion will be:

$$N_k(s) = h_k \left(\overline{N}(a_k s + b_k) + \sum_{i=1}^M c_{i_k} \psi_i(a_k s + b_k) \right), \quad k = 1, \dots, S. \quad (4.4)$$

In this framework, uncertainty could be accounted for by perturbing both the scaling parameters and the POD coefficients:

$$\begin{aligned}
a_k &\mapsto a_k + \xi_a \\
b_k &\mapsto b_k + \xi_b \\
h_k &\mapsto \xi_h h_k \\
c_{i_k} &\mapsto c_{i_k} + \xi_{c_i} \quad \text{for } k = 1 \dots M
\end{aligned} \tag{4.5}$$

Here, the first three parameters are perturbations on the nominal positions, widths, and heights of all of the individual ice shapes. The last M parameters are global perturbations on each of the POD eigenmodes. The first three parameters were chosen by scaling/shifting each individual shape so that the transformed shape most closely matches a symmetric Gaussian template $G(s)$ (which is close to the mean of the unshifted/unscaled data):

$$(a_k, b_k) = \arg \min_{a,b} \|N_k(as + b) - G(s)\|_2^2, \quad k = 1, \dots, S.$$

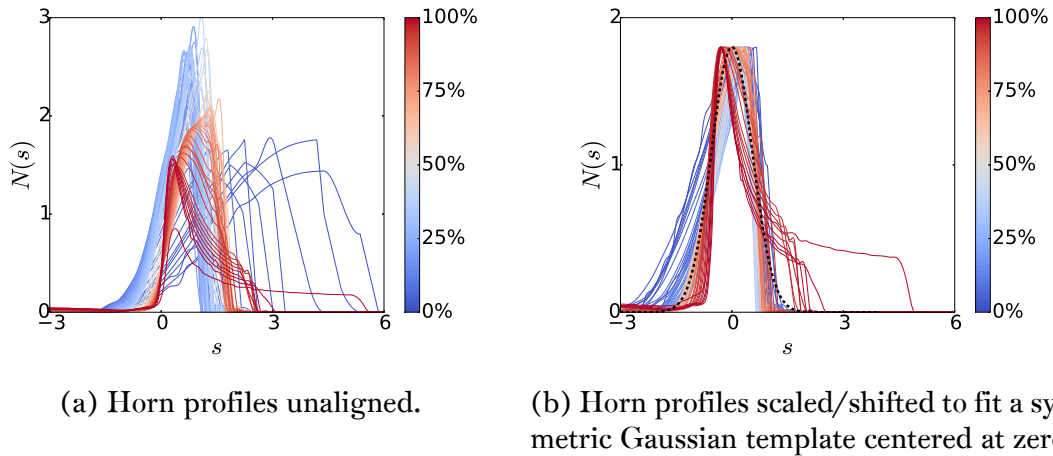


Figure 4.2: Horn profiles, unaligned and aligned. The color map denotes spanwise position along the CRM wing from root (0%) to tip (100%).

The absolute values of the POD eigenvalues are shown in Figure 4.3. As can be seen, there is a sharp drop-off in the magnitude of the scaled/shifted POD eigenvalues at mode

2. This implies that the first two modes capture much of the important features. We can visualize this by reconstructing the dataset with one and two POD modes and examining how well the shapes match the originals. Shown in Figure 4.4 are the original ice shapes along with their reconstructions using one and two POD modes. It can be observed that no skewness is able to be represented until two POD modes are used. Qualitatively speaking, the two-mode reconstructions show good agreement with the original data.

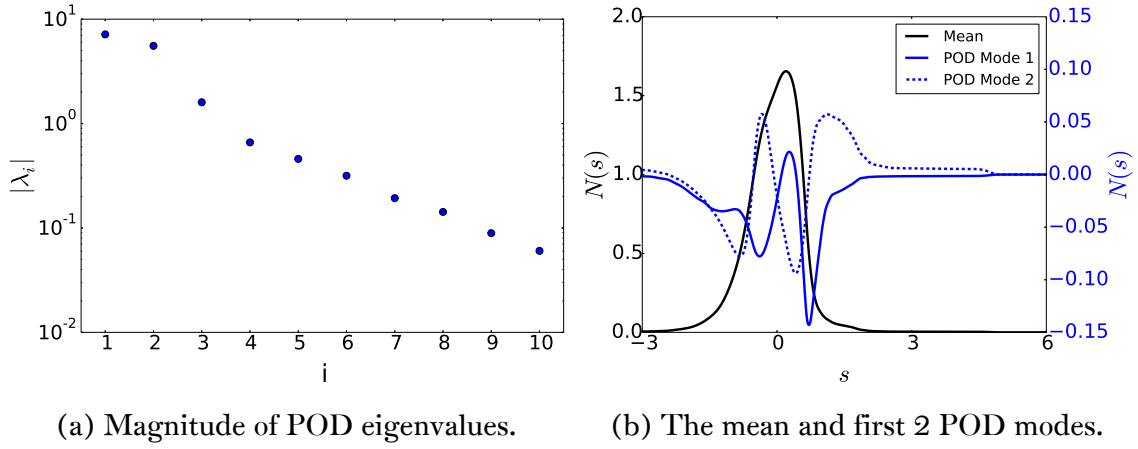


Figure 4.3: POD eigenvalues and modes for the scaled/shifted data.

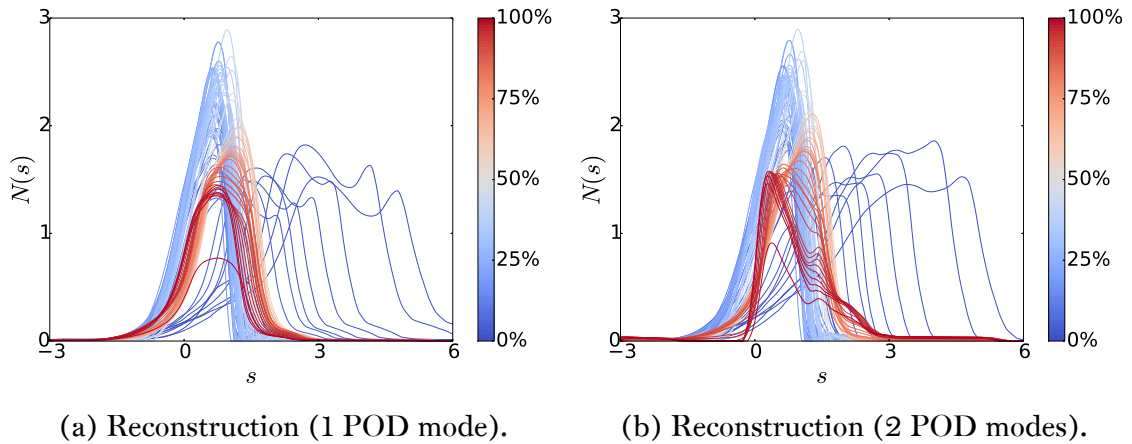


Figure 4.4: POD reconstructions of the ice. The color map denotes spanwise position along the CRM wing from root (0%) to tip (100%).

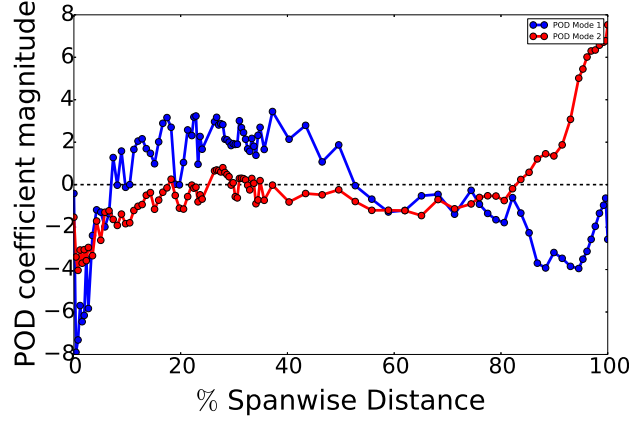
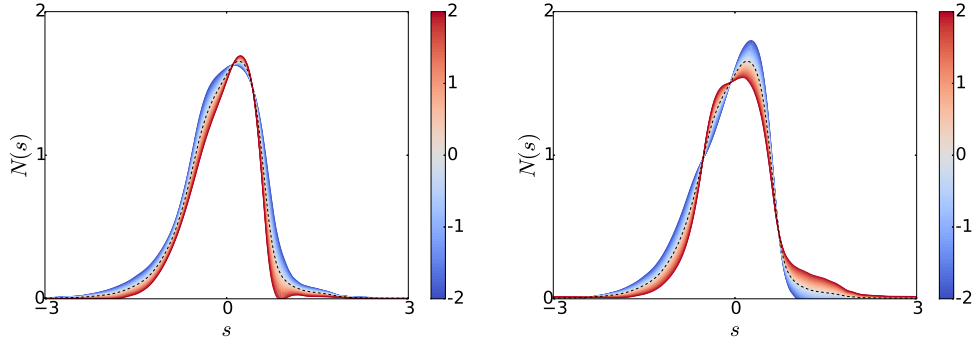


Figure 4.5: Spanwise variation of the POD modes.



(a) Variation of the 1st POD mode $\in (-2, 2)$. (b) Variation of the 2nd POD mode $\in (-2, 2)$.

Figure 4.6: Variation of the POD modes on the mean (dashed black).

The next step in identifying the relevant parameters for a UQ study is to study the spanwise variation of the POD coefficients. Figures 4.5 and 4.6 demonstrate these properties. As can be seen, both modes are most pronounced at the boundaries, since this is where ice shapes deviate the most from the mean. The first mode has the effect of making the ice profile wider and skewed left; hence, it is largest in magnitude on the inboard portions of the wing. The second mode has the effect of skewing the ice profile right; hence, it is largest in magnitude on the outboard portions of the wing.

4.3.2 Airfoil Icing UQ: Five Parameter Scenario

Now that we have generated a POD representation of our horn shape data, we can investigate the effects of uncertainty in the POD coefficients. As shown in Eq. (4.4), we have 3 scaling parameters (height, width, and position), and 2 POD coefficients, so our parameter space is 5 dimensional. We assume that all 5 of our parameters are uniformly distributed between some bounds. The ranges that we choose, as well as their independent effects on the horn shape, are displayed in Fig. 4.7. We wish to limit the parameters to ranges observed in the data; therefore, we set the upper and lower bounds of the parameters to plus/minus one standard deviation from the mean (where the means and standard deviations are calculated from the dataset distributions). This effectively produces a uniform distribution approximate to the actual distribution in the dataset.

In this UQ study, we will quantify uncertainty in the two response metrics C_L and C_D . The airfoil cross-section used here was chosen to be the cross-section at 50% semispan of the CRM, and the aerodynamic coefficients were determined at a Reynolds number of 5×10^6 , Mach number of 0.4, and angle of attack $\alpha = 3^\circ$.

Table 4.1: Data correlations

	C_L	C_D
C_L	1.00	-0.94
C_D	-0.94	1.00
a	0.09	-0.05
b	-0.78	0.82
h	-0.28	0.31
POD 1	-0.28	0.26
POD 2	0.33	-0.34

Fig. 4.8 presents the statistics for a Latin Hypercube sampling of the PCE surrogate with 10^6 samples. This surrogate required 1,103 flow solver evaluations to converge, which

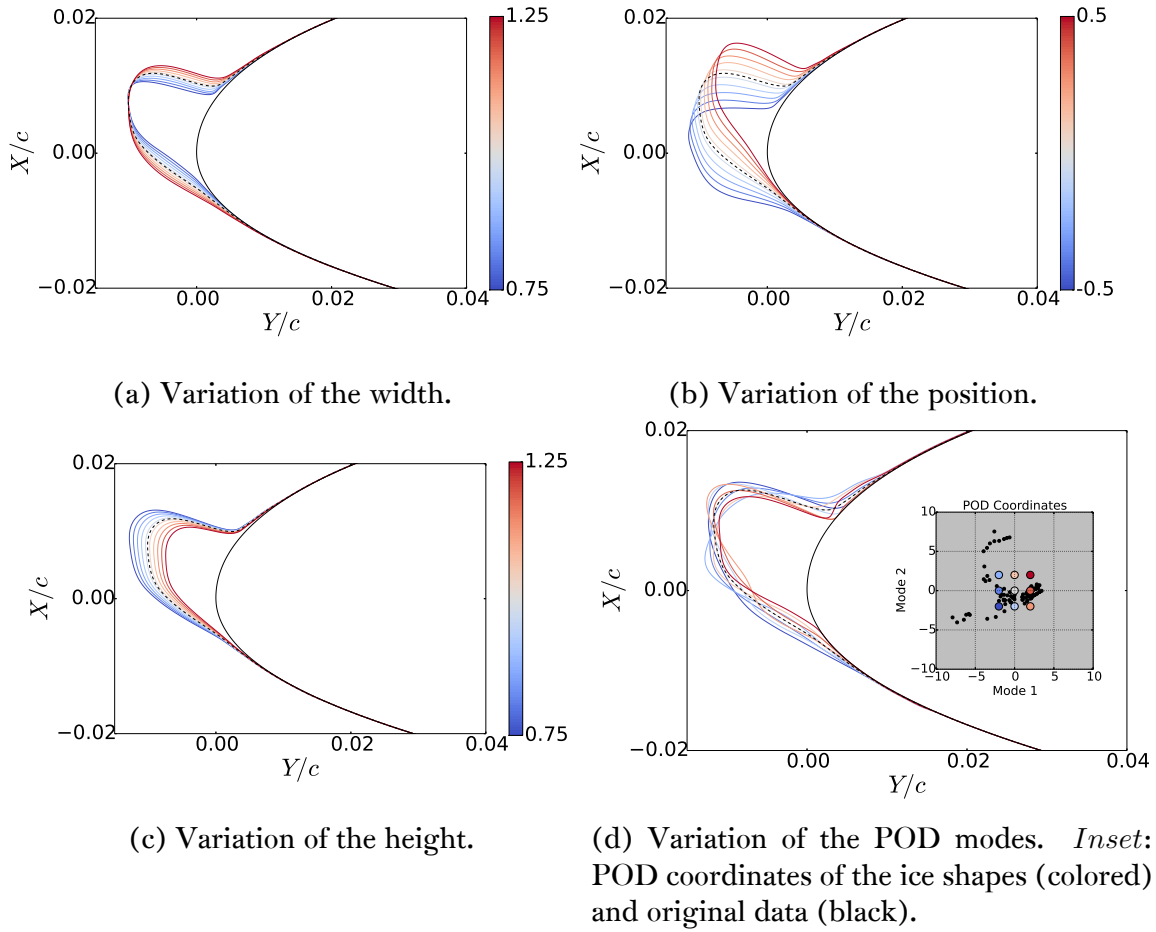


Figure 4.7: Depiction of the ranges of each of the parameters in the 5-dimensional parameter space and their effects on the horn shape (colorbars indicate parameter ranges). Dashed black ice shape in each subplot represents the mean shape used.

Table 4.2: Sobol Indices (Single Parameter)

	a	b	h	POD 1	POD 2
T	0.03	0.69	0.15	0.11	0.14

is reasonable for a 5 dimensional parameter space. Convergence is based on the change in the L_2 norm of the surrogate response covariance matrix falling below some threshold (which we set to be equal to 10^{-4}).

The first thing to note is that we can easily see that C_L and C_D correlate very strongly ($\text{corr}(C_L, C_D) = -0.94$), which indicates that over our parameter range, we only need to

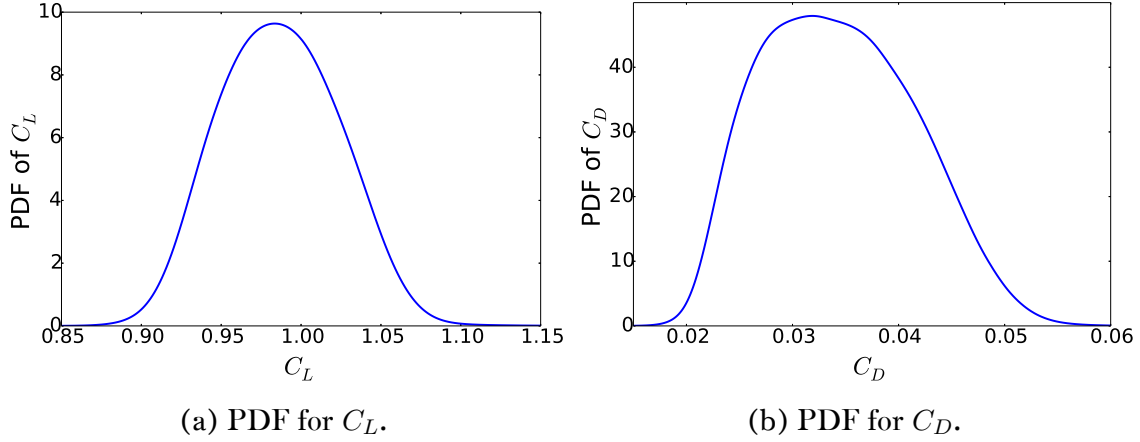


Figure 4.8: Probability density functions for two response metrics in the 5 parameter UQ study. These results required 1,103 samples in the adaptive sparse grid algorithm.

examine one of the two in order to understand the effects of our inputs on aerodynamic performance.

In order to examine the independent relative contributions of each of our 5 input parameters to the variance of our responses, we examine both the data correlations and the Sobol indices (defined previously in Chapter 2). Loosely speaking, the Sobol index gives a measure of how much, on average, a parameter (or a combination of parameters) contributes to the total variance of the response. It is clear from these metrics that variations in horn position, b , contribute the most to the variance of our response and hence b is the “most important” parameter. The caveat of this statement, of course, is that it only applies over the limited parameter range we have chosen. Had we chosen to investigate larger variations in height, for example, then height could very well be the dominant parameter.

The sign of the correlation of horn position with our responses indicates that performance degrades (i.e., lower lift, higher drag) as the horn moves closer to the upper surface. The physical explanation for this is intuitive, and can be ascertained by inspecting Fig. 4.10. The upper surface horn is a more obtrusive flow obstacle, and therefore promotes a larger steady-state separation bubble aft of the horn than the equally-sized (but less obtrusive) lower horn. This phenomenon agrees with similar findings in the previous chapter of this thesis.

Although horn position is the most dominant parameter, the other parameters do affect performance; this can be revealed by examining the correlations in Table 4.1. As expected, taller horns give worse performance than shorter ones. Performance is relatively insensitive to variations in width, but there is a slight tendency for narrower horns to give worse performance than wider horns. This is because narrower horns come to a sharper point and hence promote larger leading edge separation bubbles, whereas wider horns have more rounded points and hence are less severe. Perhaps not as immediately clear is the effect of the POD modes on performance. One way to gain some insight into this is to project the surrogate onto the two-parameter space of POD coefficients. This can be approximated by sampling the surrogate, and then locally averaging out the three scaling parameters. This produces Fig. 4.11, which demonstrates that the POD modes can interact in such a way as to produce a distinct skew to the horn. Depending on the direction of this skew, this can either help or hurt aerodynamic performance, since the length of the leading edge separation bubble depends on the horn shape details.

Integrating all of these analyses together gives a clean, intuitive picture of the effects of our 5 parameters on the flow. We find that the most dangerous horn shapes in our parameter space are tall, narrow, upper surface horns that have a sharp upper skew shape; the most benign ones are short, wide and rounded, located on the lower surface, and have gentle downward skew (or no skew at all). This is affirmed by examining Fig 4.9, which shows clear statistical clustering of the horn shapes in parameter space that produce the best and worst aerodynamic performance.

4.4 Experimental Database

We now turn our attention to a different source of ice shape data – namely, ice shapes generated from experiments in icing wind tunnels. The particular nature of the ice shapes

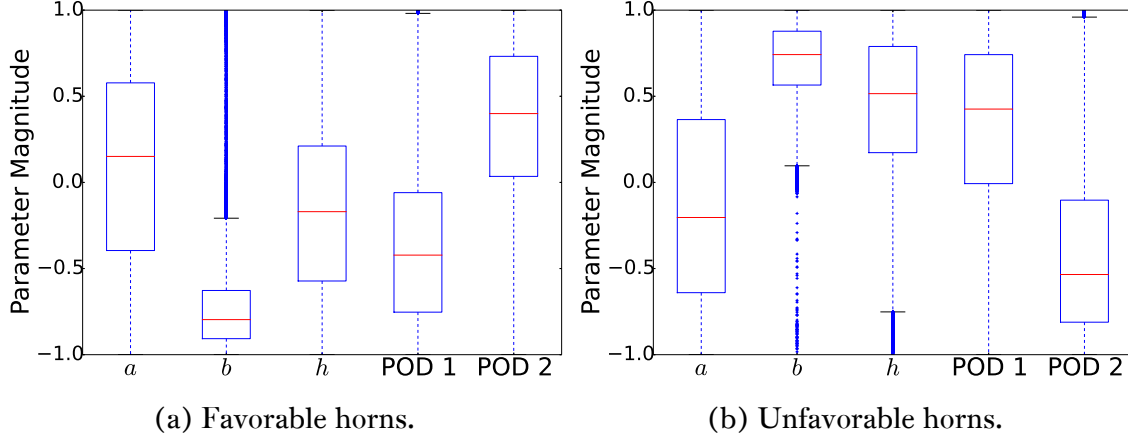


Figure 4.9: Statistical clustering of “good” (bottom 10% of C_L) and “bad” (top 10% of C_L) horns in parameter space, based on 10^6 Latin Hypercube samples of the surrogate. The parameter magnitudes have all been linearly scaled to lie between ± 1 .

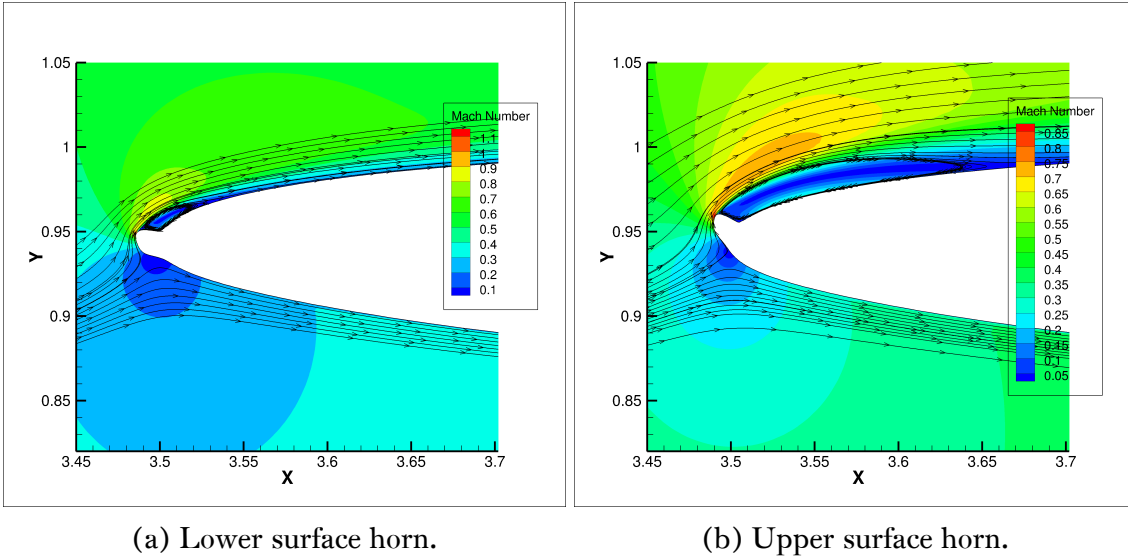
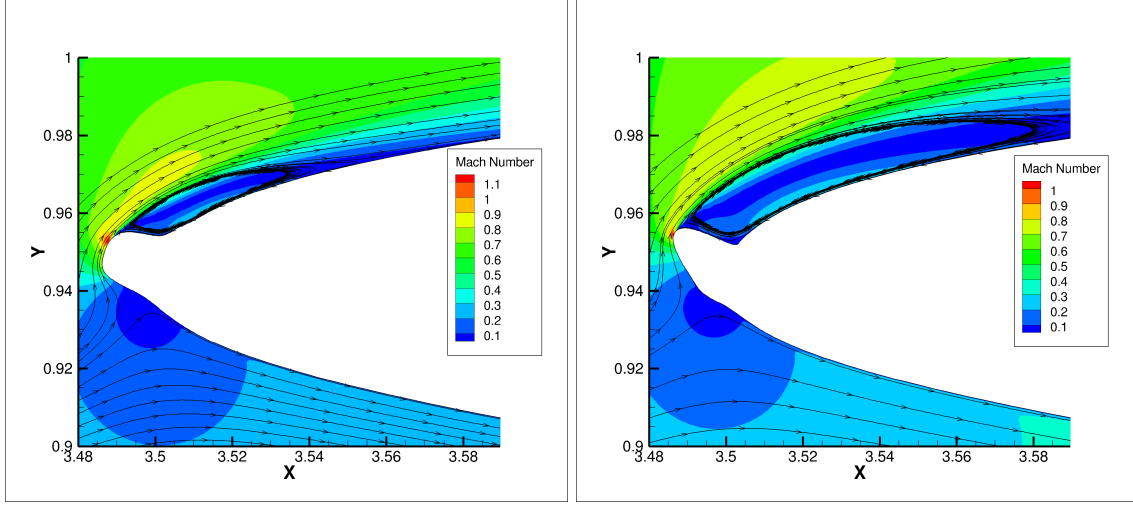


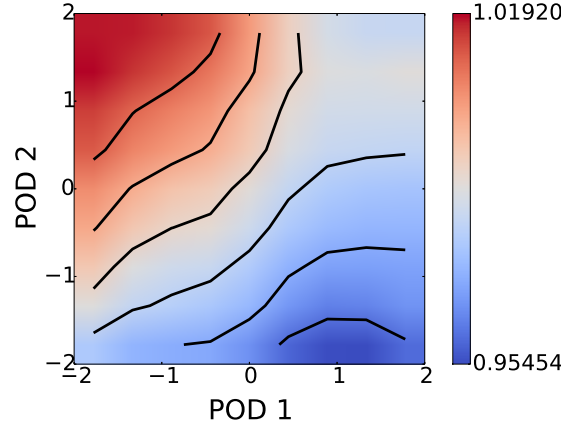
Figure 4.10: Flow field contours (of Mach number) for two horns of equal size and shape that differ only in their relative positions. The horn in (b) is more normal to the freestream flow than the horn in (a), and hence generates a larger scale leading edge separation bubble.

in this dataset presents an opportunity for us to address several issues that were irrelevant in the previous case study. Our goal in the preceding example was to use a dataset in which the individual ice shapes exhibited only modest variation from the mean shape. This ensured that we could obtain a faithful representation of the ice using only a modest number of parameters (five). In this section, we are interested in applying the same techniques to



(a) Favorably skewed horn.

(b) Unfavorably skewed horn.



(c) Contours of $\mathbb{E}(C_L|\text{POD1}, \text{POD2})$

Figure 4.11: Effect of POD coefficients on performance. Part (c) displays contours of C_L as a function of the POD coefficients, obtained by sampling the PCE surrogate and locally averaging over b and h . Parts (a-b) display two horns that are the same size/position, but differ in their POD coefficients. This gives rise to favorable/unfavorable skewness in the shape.

a dataset whose entries represent a much wider range of physical conditions, and hence a much wider range of shapes. As we will see, doing this comes at the cost of having to retain more POD modes to accurately represent the ice, which translates into a much larger UQ study. This will motivate the need for us to augment our current approach with a data clustering preprocessing step. In addition to that, we will use this data as a means to demonstrate how to analyze ice shapes that are not convex and hence cannot be cast into

arc-length coordinates. Lastly, we show how to combine physical information about icing with the POD ice shape model, which gives us a data-driven model of ice accretion.

Two resources that provide an excellent sampling of 2D ice shapes for different airfoil geometries subjected to different icing conditions are Addy [3], and Wright and Rutkowski [103]. For the studies used in this example, we use the shapes found in Addy. The shapes found in this database were generated in a wind tunnel by exposing different airfoils to a wide range of icing conditions. The conditions used reflect the guidelines and standards for atmospheric

icing conditions as defined by the Federal Aviation Administration (Federal Aviation Regulations 25 Appendix C). In that database, three clean airfoil geometries are used – one which represents a business jet, one which represents a commercial transport, and one which represents a general aviation aircraft. A plot of all of these shapes together on the same airfoil is shown in Fig. 4.13. As can be observed, there is significant variation in the size and shape details of the ice. These shapes were generated from variations in the following range of icing conditions:

- Mach number $\in [0.28, 0.39]$
- Airspeed $\in [175, 250]$ knots
- Attitude $\in [1.5, 6.0]^\circ$
- Free-stream temperature $\in [-27.8, -0.7]^\circ C$
- Surface temperature $\in [-31.6, -5.0]^\circ C$
- MVD $\in [15, 160] \mu m$
- LWC $\in [0.310, 1.6] \frac{g}{m^3}$

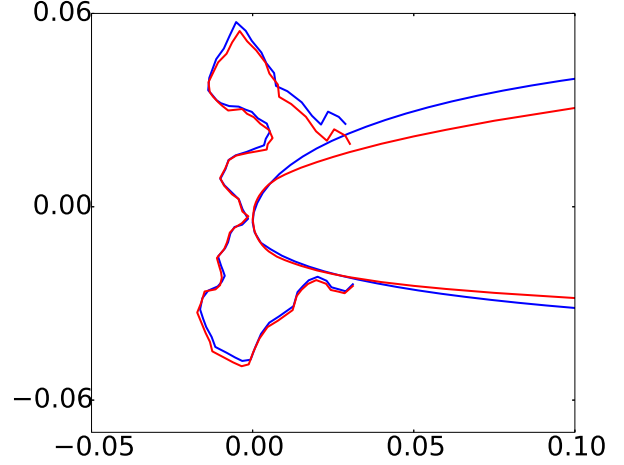


Figure 4.12: Scaling transformation example. *Blue*: original shape and airfoil, *red*: scaled ice and base airfoil.

- Exposure time $\in [0.7, 45]$ min

See Addy [3] (pg. 40) for further details.

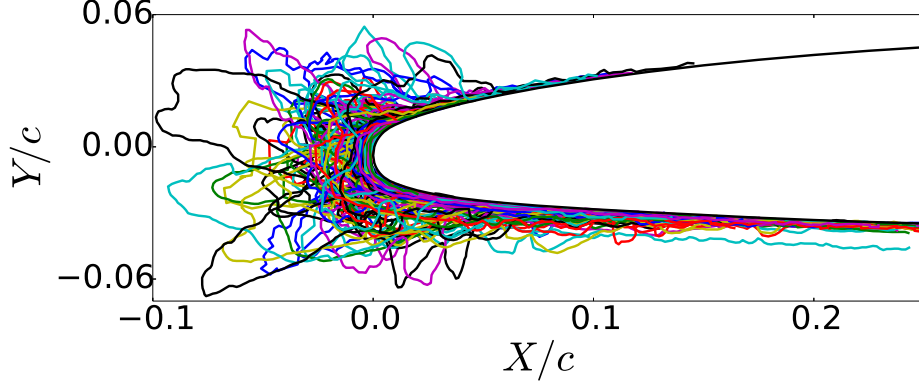


Figure 4.13: Ice shape dataset, from Addy[3]. c denotes the chord (set to 1 in all examples).

There are 145 total shapes present in the dataset that includes all base airfoils (business, commercial, and general aviation). It is here that we encounter a problem that must be solved before we are able to use this dataset – the ice shapes present were generated on three different clean airfoil geometries, and it is not immediately clear how to account for that. The approach we take to address this is to select one of the three clean geometries as a base shape, find the transformations which scale the other two clean geometries onto that baseline, and then apply these scalings to the data as appropriate. An example showing the effect of this transformation is shown in Fig. 4.12. As can be seen, the example ice shape shown (like most of the ice shapes in our database) exists at the tip of the leading edge of the airfoil and hence is not affected strongly by the scaling transformation.

4.4.1 POD of the Ice Shape Data

In order to apply the methods above to our ice shape problem, we need to first determine an appropriate vector space representation of the ice shapes. There is not necessarily one unique way of doing this. One approach would be to cast all ice shapes into arc-length coordinates (where ice shape height is measured normal to the clean base). However,

a drawback of this approach is that it is not possible to represent ice shapes which are multi-valued in arc-length coordinates. While many of the ice shapes in this database could be represented accurately in arc-length coordinates, a few of the shapes are suspect. Moreover, one of the contributions we wish to make here is to demonstrate an alternative method for vectorizing ice shapes which can handle any ice shape with no restrictions.

This method works as follows. We first find a rectangular window of space in which all of the ice shapes in Fig. 4.13 fit, and we overlay this space with a static Cartesian mesh. For a particular ice shape, we assign a value of 1 to a particular grid point if that grid point is inside/on the body of the ice, or a value of 0 if it is not. It should be noted that any points inside the clean airfoil were excepted from the grid, so that our mesh consists entirely of points located either in the free-stream, or in the ice. This process produces a “pixelation” of the ice shapes, and hence is a shape parameterization method that can handle non-convex ice shapes (which would be multi-valued in arc-length coordinates). The background mesh consists of roughly $N = 7 \times 10^5$ points. An example of this process is shown in Fig. 4.14.

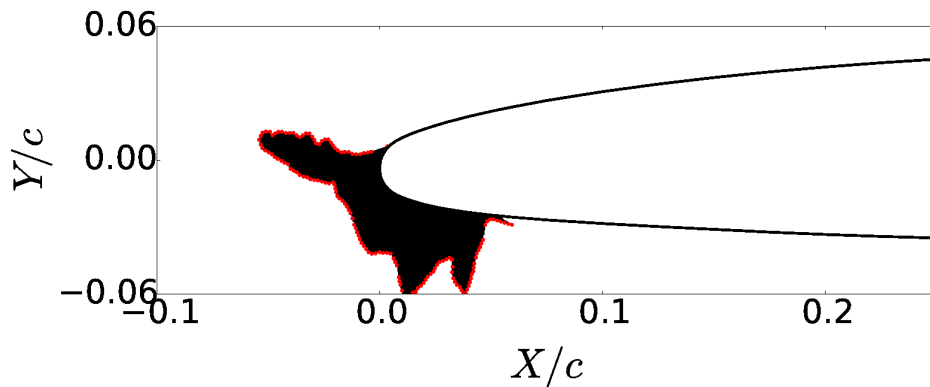


Figure 4.14: Illustration of how ice shapes are defined for POD. Each ice shape is defined on a static Cartesian background mesh, the bounds of which form the rectangular window of this figure. For a particular ice shape, a grid point is assigned a value of 1 if that point is located inside the ice boundary. The ice boundary is shown in dotted red; points on the ice are shown in dotted black.

Having cast all of our ice shapes in the same N -dimensional vector space, we are now in a position to compute the POD modes. Because we will ultimately be doing UQ on the POD modes, we would like to retain as few modes as possible while still having accurate representation power in that basis. This is the classic tradeoff between economy and accuracy in low-dimensional modeling. In order to make an informed decision on how many modes to keep, we look at the POD eigenvalues, shown in Fig. 4.15.

We make the decision to truncate the expansion at order 10, where the magnitudes of the eigenvalues have decayed by about one order of magnitude. A glance at the cumulative sum of the eigenvalues in Fig. 4.15 shows that at 10 modes, the cumulative sum reaches almost exactly two-thirds of its final value. A statistical interpretation of this is that two-thirds of the variance exhibited by the dataset POD coefficients is retained by the first 10 modes [42]. This is a quantitative basis for discussing how accurate the representation power of our model is. Certainly we could improve the fidelity of our reconstructions by retaining more modes, but this would present a more computationally laborious UQ study.

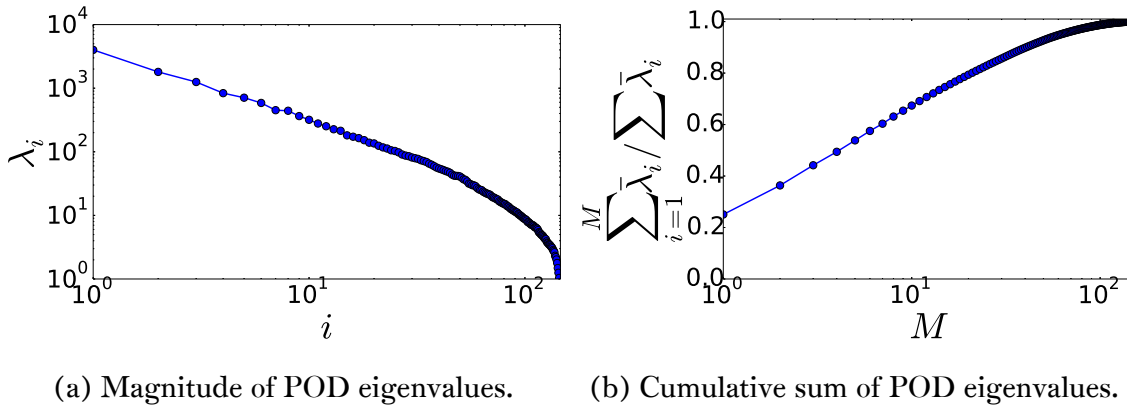


Figure 4.15: POD eigenvalues.

We display the mean shape and first 10 modes in Fig. 4.16. As can be seen, the lowest order mode has the most effect on the underside of the airfoil, where – in our dataset – there is a high probability of having ice (which is likely due to the fact that these airfoils were tested at an angle of attack, making some amount of underside accretion

relatively likely). The apparent function of the second mode is to simply add mass at “intermediate” locations. The third mode clearly acts as a switch between upper surface horn accretions and lower surface rime accretions. The fourth mode similarly discriminates between rime-like accretions that protrude outward along the centerline of the airfoil, and upper/lower surface horns. The higher order modes represent more detailed and extreme shape excursions from the mean.

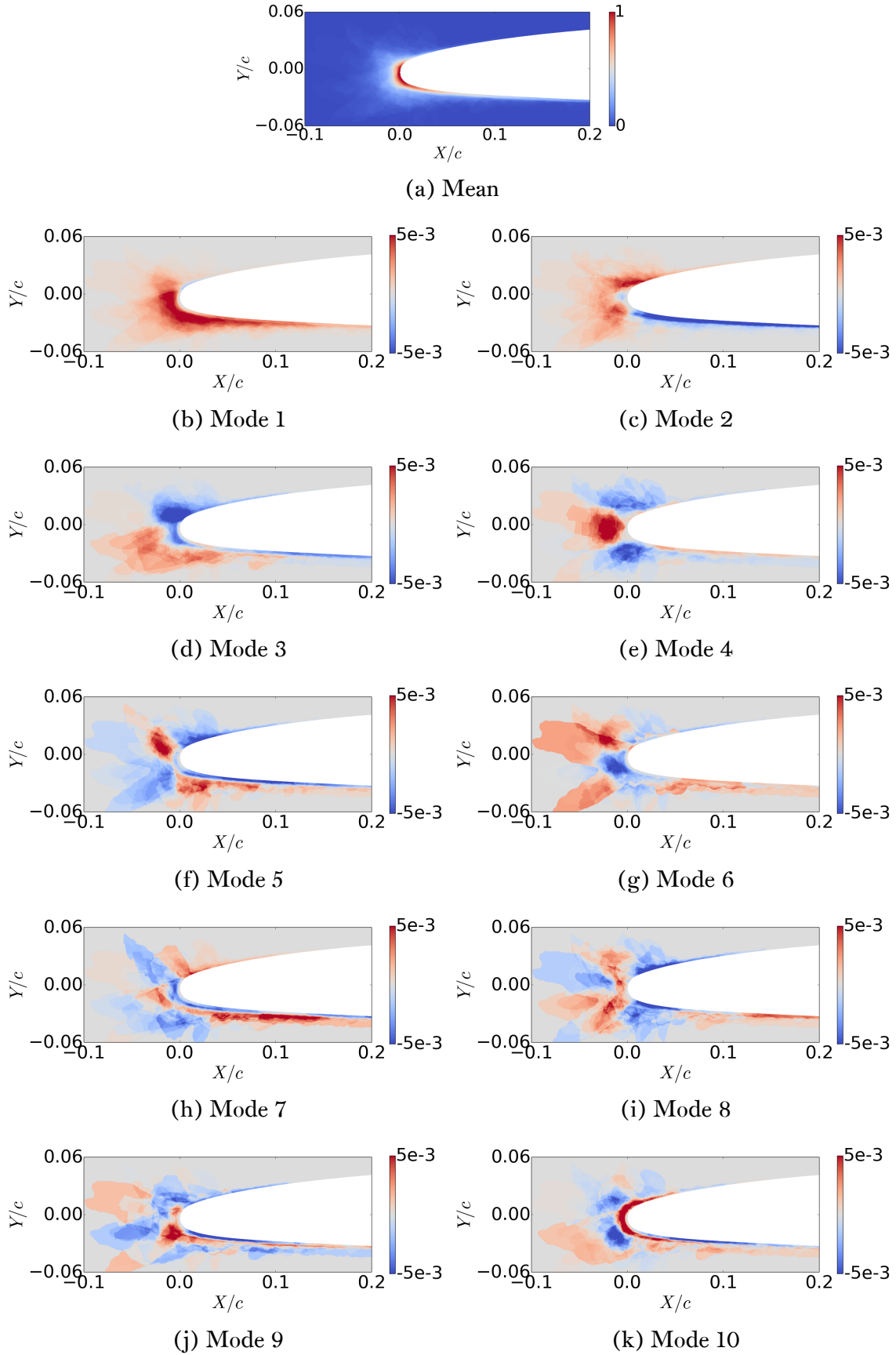
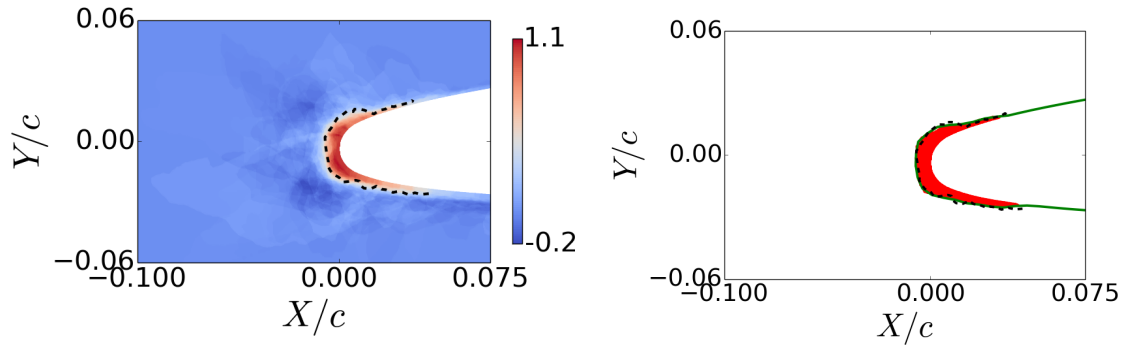


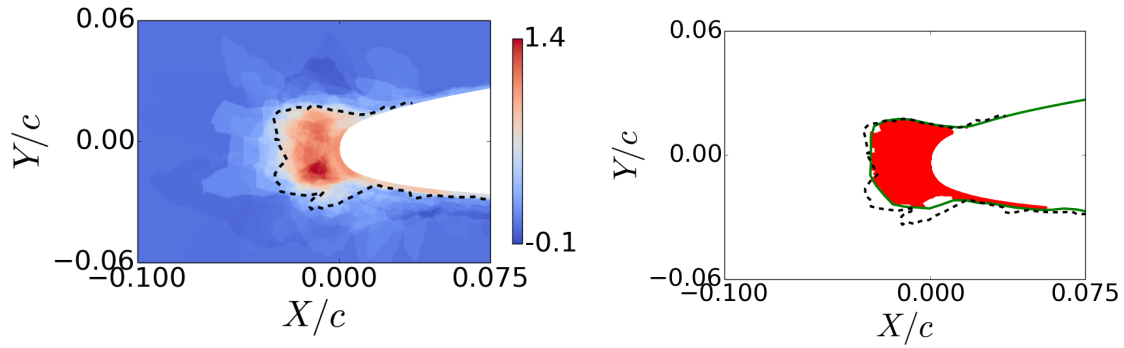
Figure 4.16: Mean and POD Modes.

We have already given some indication of the accuracy of our 10-mode ice shape model by quantifying how much of the POD coefficient variance we retain. We can do a more direct (and more visual) investigation of this question by actually projecting the ice shapes onto the truncated POD basis and comparing results. It is important to realize that our POD reconstructions will need to be filtered *a posteriori*. This is because the reconstructions will consist of grid point values that exist on a continuum on some range (around 0 to 1). However, our ice shapes are binary in nature – a particular grid point should either be 1 if it is on the ice, or 0 if it is not.

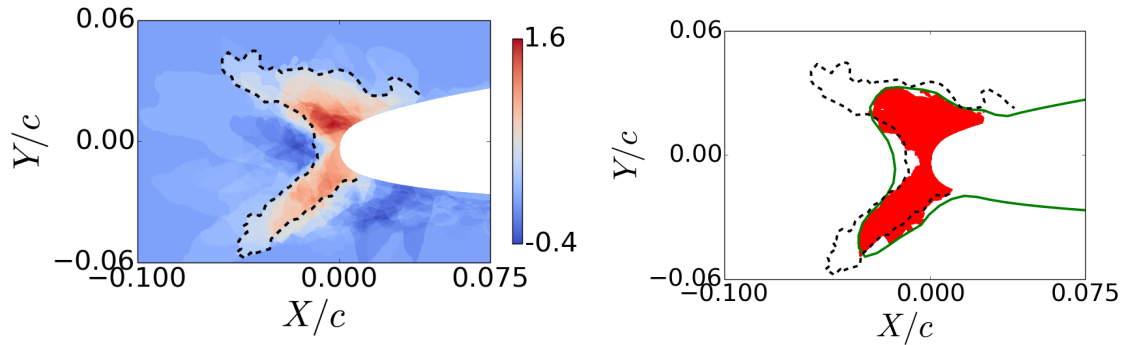
To rectify this issue, we use the following simple filtering algorithm. For a particular ice shape, we compute the maximum and minimum values of the entire grid, and we round everything greater than half of the range to 1 (and, similarly, everything less than half to 0). Demonstrations of this process are shown in Fig. 4.17 for several ice shapes. We observe that the quality of the reconstructions generally depends on how close to the mean a particular ice shape is. We can easily compute this distance with the Euclidean norm in POD coordinate space. Fig. 4.17 reveals that shapes that are close to the mean have great reconstructions, while the agreement for more extreme shapes far from the mean is less satisfactory. This agrees with our observation that the more exotic/extreme shape perturbations exist only in the higher (not lower) modes, since they account for a relatively small fraction of the dataset.



(a) Unfiltered reconstruction. (b) Filtered reconstruction.
POD coordinate distance: 8th percentile



(c) Unfiltered reconstruction. (d) Filtered reconstruction.
POD coordinate distance: 76th percentile



(e) Unfiltered reconstruction. (f) Filtered reconstruction.
POD coordinate distance: 94th percentile

Figure 4.17: *Left:* Projection of ice shape (dashed black line) onto 10 POD modes. *Right:* Filtered reconstruction (red), fitted boundary (green), and original shape (dashed black). Shown below each subfigure group is the POD coordinate distance of each respective ice shape from the dataset mean, expressed as a percentile of the distribution of dataset coordinate distances.

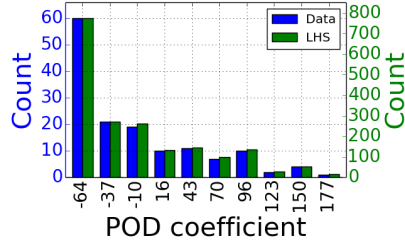
4.4.2 Airfoil Icing UQ: 10 Parameter Scenario

In accordance with the UQ approach we have followed thus far, the next step would be to perform PCE-based UQ for the 10-dimensional parameter space, with output metrics of lift/drag at some angle of attack. However, such an approach proves problematic, for two reasons. First, the distributions of dataset POD coefficients do not conform well to any of the “standard” distributions used in polynomial chaos (e.g., normal or uniform). Of course, we could adopt the same procedure as we did in our last study, whereby we approximate the actual distributions with uniform (or normal) distributions whose width is controlled by the standard deviation of the actual distribution. However, this approach is not very satisfactory, if our goal is to obtain output results whose statistics reflect the actual data distributions.

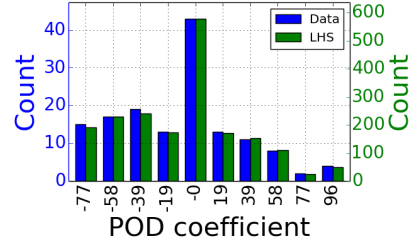
The second issue with the PCE approach to this study is that, quite simply, it does not seem to work well when applied to this particular problem. We know this because we actually did attempt to implement the method on this problem; however, even after running several thousands of simulations with anisotropic sparse grids, the resulting PCE surrogate was clearly not converged to an accuracy sufficient enough to be believable (to illustrate, a significant portion of the surrogate statistics indicated negative drag, a phenomenon which was nowhere supported in the data). We conjecture that the reason for this is polynomial overshoot in the surrogate, caused by non-smoothness in the parameter space (akin to Gibbs overshoot).

This illustrates an important lesson about choosing the appropriate UQ method for the problem at hand. Polynomial chaos, while useful in many scenarios, is no panacea. Indeed, it is predicated on the fundamental assumption that the response surface of interest is smooth, and it can certainly fail if this assumption does not hold. The UQ method for this problem must, at minimum, be able to address that reality. Additionally, we would like for the input process statistics to reflect the actual distribution of POD coordinates in

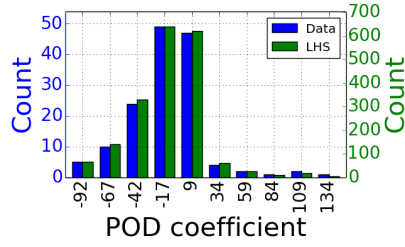
the dataset. As a result of these constraints, we choose to use Latin Hypercube Sampling (LHS) for this problem.



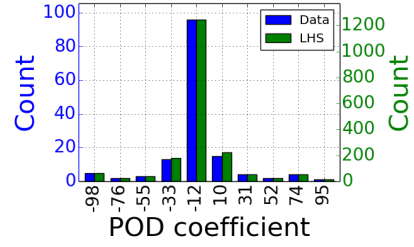
(a) Mode 1



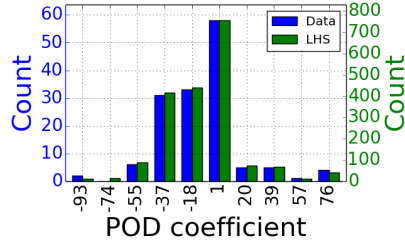
(b) Mode 2



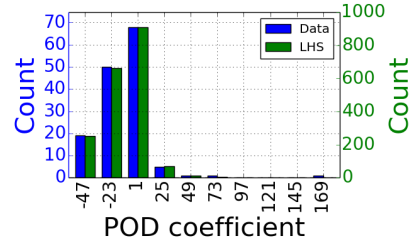
(c) Mode 3



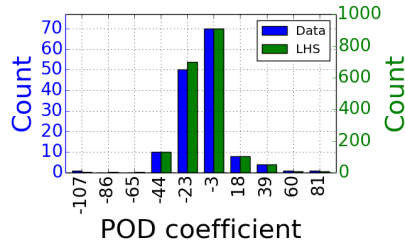
(d) Mode 4



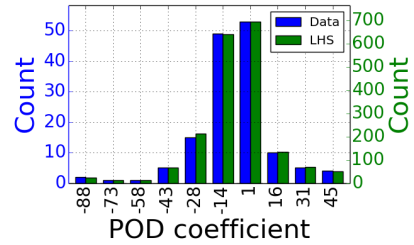
(e) Mode 5



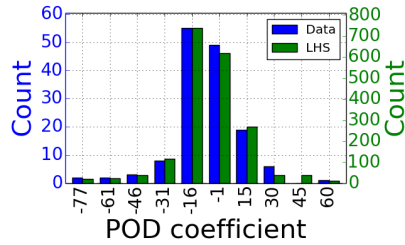
(f) Mode 6



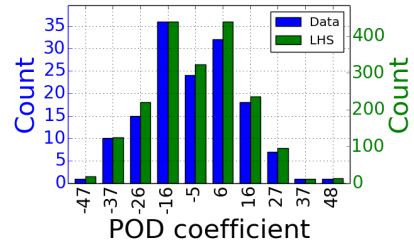
(g) Mode 7



(h) Mode 8



(i) Mode 9



(j) Mode 10

Figure 4.18: Distribution of POD coefficients for the original dataset (*blue*) and LHS (*green*).

It is not our intention to give a detailed exposition of Latin Hypercube sampling; we only review the basic concepts. The interested reader is referred to the literature on the subject [60, 43, 40, 88]. LHS is a sampling technique in which the range of each individual parameter is first divided into N bins, where N is the number of user-specified samples. The relative lengths of these bins are controlled by a user-specified distribution of the respective parameters, such that bins are shorter near areas of higher probability density. A single sample is drawn from each of the N bins for every individual parameter, and then these values are shuffled to create N parameter vectors. A defining characteristic of this method is that every row and column of the parameter space hypercube contains exactly one sample.

For this 10-parameter study, we choose to perform LHS with around 2,000 samples. Fig. 4.18 displays the distributions of POD coefficients for the original dataset and the LHS samples; the statistical agreement is excellent. As mentioned earlier, the geometry we use as the base clean airfoil is the business jet airfoil from Addy [3]. We set the Reynolds number of the flow to 5 million, the Mach number to 0.3, and the angle of attack to 3 degrees.

The resulting shapes – colored according to lift coefficient – are displayed in Fig. 4.19, and the corresponding statistics are shown in Fig. 4.20. We see that the lift statistics show a clear peak around $C_L = 0.4$; this is due to the fact that a large subgroup of the LHS sample shapes are smaller rime accretions for which there is not much deterioration in lift (we will have more to say about this fact in the next section on database partitioning). The statistics have a clear tail toward lower values of C_L , indicating that performance deteriorates by a relatively large amount for the more extreme shapes in the distribution.

We can analyze the database of LHS samples for statistical trends in several ways. One interesting question to pose is, what are the most “dangerous” (i.e., low lift, high drag) regions of space for ice accretion to exist, according to the data we have generated and studied? If we simply view the ice shapes shown in Fig. 4.19 as points in space, where all

points on the same shape have the same value of lift coefficient, then we are in a position to compute local spatial statistics as an answer to that question. We only have to define how we compute spatial subdomains. Fig. 4.21 displays the results obtained by using a quadtree decomposition of the ice shape points, with a minimum bin size specified in order to allow the cell sizes to reflect the local spatial point density. Fig. 4.21a shows the spatial average, from which it is clear that lower surface rime accretions are relatively benign, even if they are reasonably massive. Fig 4.21b shows the spatial variance. As would be expected, this plot contains more noise than Fig. 4.21a (since averaging is a smoothing operation); however, we can clearly see a horn outlined by the border of the areas of low variance. The interpretation is that performance is very sensitive to upper/lower surface horn accretions. Of course, this is widely known, but with the data we can quantify the horn shape that gives the most aerodynamic sensitivity (for the range of shapes that can be represented by our parameter space).

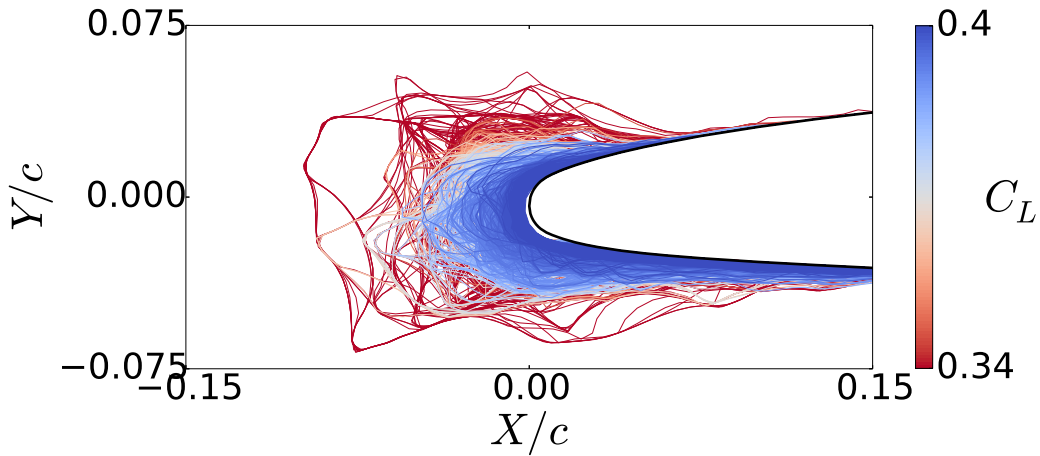


Figure 4.19: 1,921 Latin Hypercube samples.

POD of a database of ice shapes creates a set of spatial modes that optimally explain the database. Having statistically studied a wide range of shapes generated from these modes,

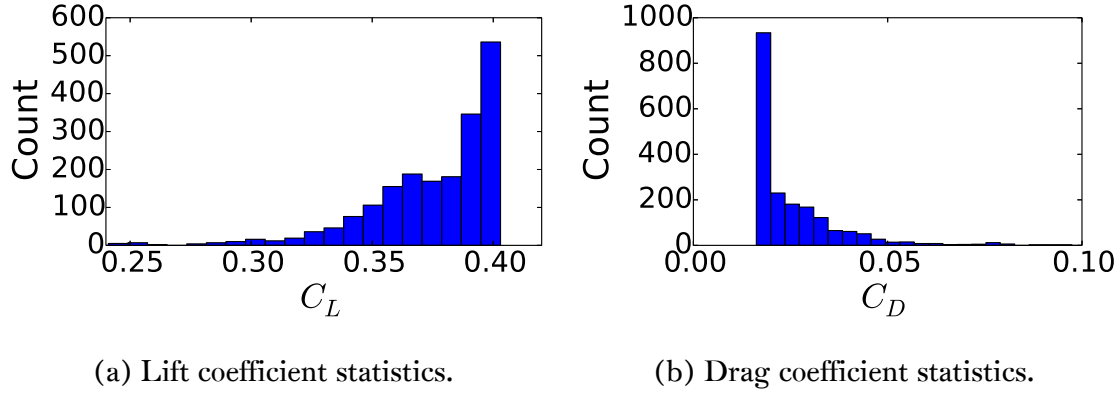
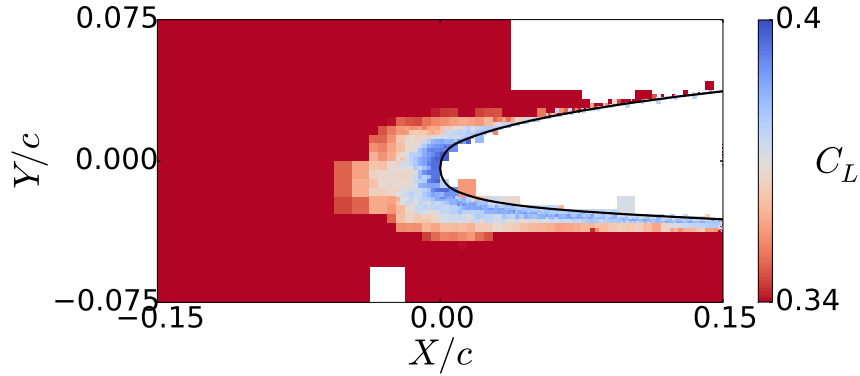


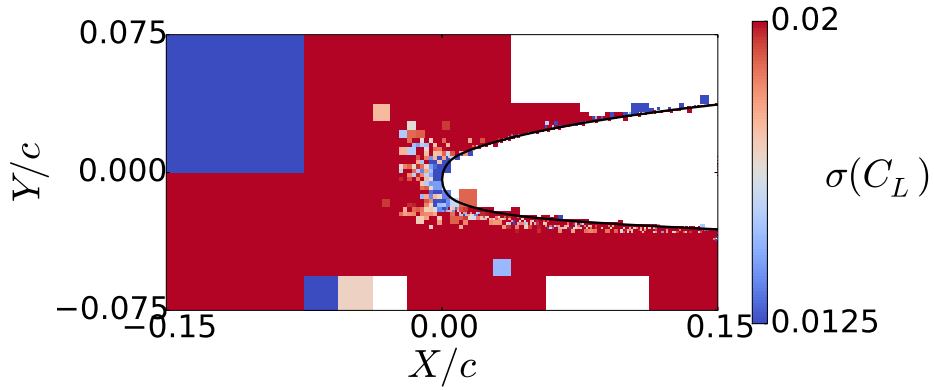
Figure 4.20: Statistics computed using 1,921 LHS samples.

we consider the next natural question: what are the statistical effects of these modes on aerodynamic performance? That is, we wish to understand performance in terms of the dominant spatial characteristics of the ice.

We can understand the general effects of the POD modes by examining the coefficients of those shapes in the upper and lower deciles of C_L . Fig 4.22 displays these statistics, along with the statistics of the original LHS distributions, for the first four modes. As we mentioned previously, the effect of modes 1 and 2 is to add ice mass, and since the top decile of C_L is populated with low ice accretions, the statistics for that decile skews heavily negative. On the other hand, the lower decile of C_L is populated with large, massive shapes, which is reflected in a slight positive skew in mode 2 (relative to the original LHS distribution). Mode 3 acts as a switch between lower and upper surface accretion. The top decile of C_L skews slightly right of the original LHS distribution for this mode, which reflects the notion that lower surface accretions are more benign than upper surface ones. Surprisingly, however, there does not appear to be any relative skew in the distribution of the lower decile, which might be expected on the same intuitive grounds. Mode 4 acts as a switch between large rime accretions along the chordline and large horns on the upper/lower surface. As expected, the lower decile of C_L tends to skew toward range giving horns, while the upper decile is tightly distributed around zero.



(a) Spatial average of C_L .



(b) Spatial variance of C_L .

Figure 4.21: Spatial statistics of C_L , computed from LHS sample shapes.

4.4.3 A Data-Driven Ice Accretion Model

It might be objected that we have not learned anything fundamentally new about icing physics from these observations; while this is true, it misses the point of what we are doing. The approach followed in this chapter has been to build and study icing models purely from *data*, with no appeal to the underlying physics. We have already seen the first usage of this in terms of uncertainty quantification – that is, we utilize a database of observations to build a model which we can study statistically. This allows us to quantify the range of aerodynamic performance corresponding to the range of variation expressed by the underlying data.

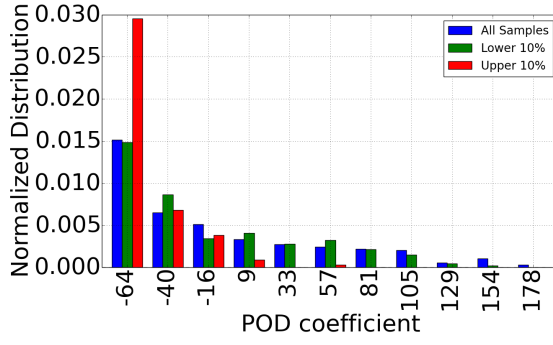
A second benefit of our approach may be realized if we link the spatial features (i.e., modal coordinates) of our data to the physical conditions under which these shapes were generated. We have data on the wind tunnel conditions under which all elements of the database were generated, including temperature, liquid water content, and accretion time, but until now, we have not made use of it. Melding physical information into our database introduces the possibility of making a “quasi-physical” model, by which we mean a data-derived POD model whose coefficients depend on physical conditions. With this approach, we are attempting to construct a statistical model of ice accretion, where the inputs are physical conditions (accretion time, temperature, and LWC) and the outputs are shape/aerodynamics. This would be useful in two ways. First, it would provide a means for comparing, benchmarking, and even improving physics-based numerical codes. Second, the data-driven model could be non-deterministic, in the sense that the mapping from physical parameters to POD coordinates need not be deterministic (indeed, the data shown in Fig. 4.23 suggests that that mapping might not be deterministic). This could account for non-deterministic sources of uncertainty in ice accretion, which is not currently accounted for in computational codes.

We begin by examining the relationships between the modal coordinates of our database and their physical conditions. Fig. 4.23 displays 2D projections of the POD coordinates, colored according to accretion time, static temperature, and liquid water content. The trend which stands out most obviously is the relationship between the modal coordinates and accretion time. Fig. 4.23a reveals a very distinct clustering (which, as we will see in the subsequent section, could actually have been ascertained with no prior knowledge of accretion time as well). Fig. 4.23b shows similar segregation between low-time accretions, which are clustered around the origin of the figure, and high-time accretions, which are spread over larger numerical values of the coordinates. As we have already noted, modes 1 and 2 simply add ice mass, and so it should be no surprise that they clearly separate the smaller accretions from the larger ones.

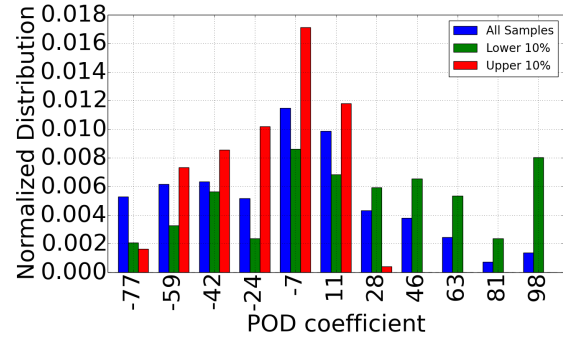
Fig. 4.23c, 4.23d show the temperature dependence of the coordinates, while Fig. 4.23e, 4.23f show dependence on LWC. Interestingly, there does not appear to be any discernible relationships between first two modes and either temperature or LWC. This is perhaps because those modes simply differentiate “big” from “small” shapes, which is mostly just a function of accretion time. However, there are apparent separations in the large ice shapes in mode 4 coordinates. We see that the large shapes with colder temperatures and lower LWC tend to have positive mode 4 coordinates, while those with warmer temperatures and higher LWC tend to have negative mode 4 coordinates. This provides further confirmation that mode 4 is essentially a switch between rime-like conditions and glaze-like conditions.

Having imported physical information into our model, we are now in a position to provide some encouraging evidence in favor of a purely data-driven icing model which respects icing physics. Shown in Fig. 4.24 are (unfiltered) ice shapes which were generated by specifying physical condition ranges. To create them, we filter the database to select only that subset of shapes that fit our criteria. We can then select the POD coefficients by examining the modal coefficient statistics of that subset. For example, the shapes in Fig 4.24 were created using the mean values of the modal coefficients of the downselected subgroups. As can be seen, we obtain qualitatively convincing horns and rime accretions simply by specifying the physical conditions appropriately.

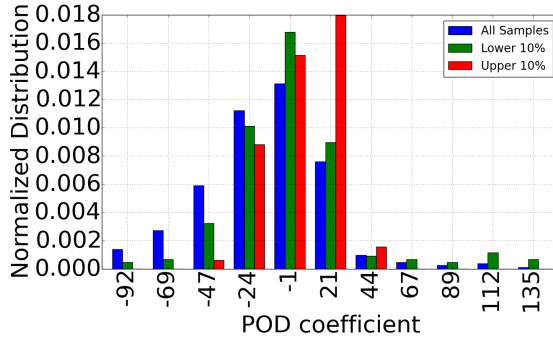
We can take this one step further and make a “random” ice shape generator which is purely data-based. Fig. 4.25 shows 100 shapes for both rime and glaze icing conditions. These shapes were generated by filtering the database for the shapes that match our criteria, creating a histogram of the POD coefficients of those shapes, and drawing 100 random samples from that distribution. The resulting shapes clearly are clustered about the mean shapes shown in Fig. 4.24, and demonstrate a significant amount of variation while still being generally “believable”, given the physical conditions used to create them.



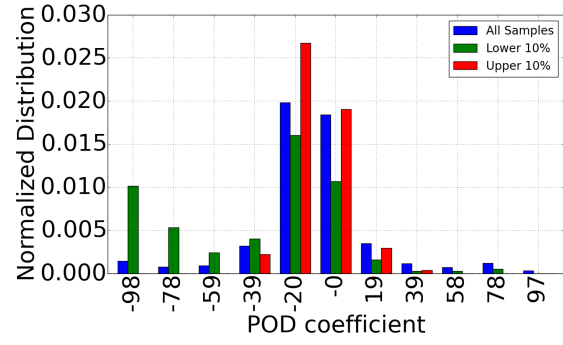
(a) Mode 1.



(b) Mode 2.



(c) Mode 3.



(d) Mode 4.

Figure 4.22: Statistics for the top/bottom deciles of C_L , along with the original LHS distributions.

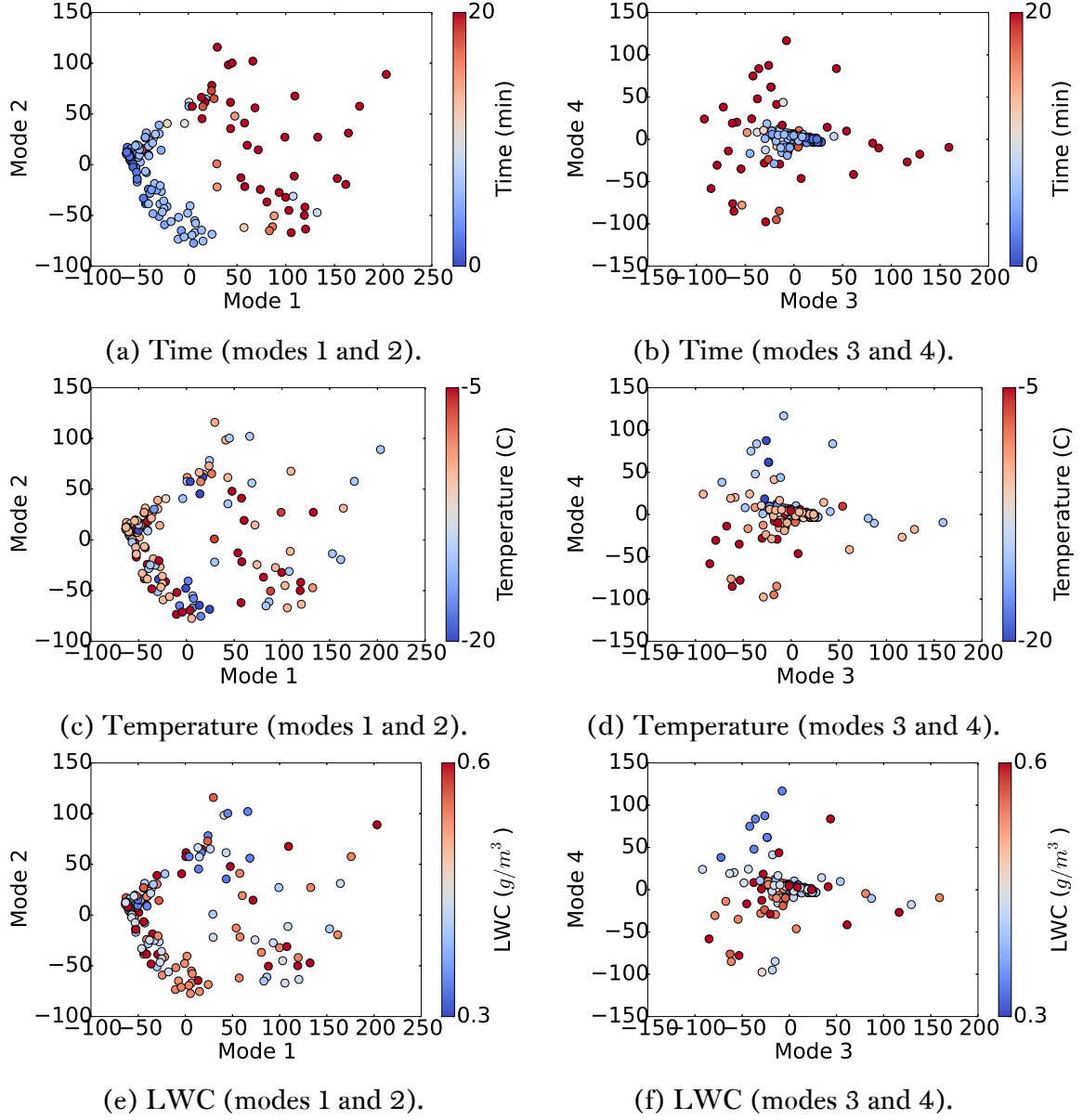
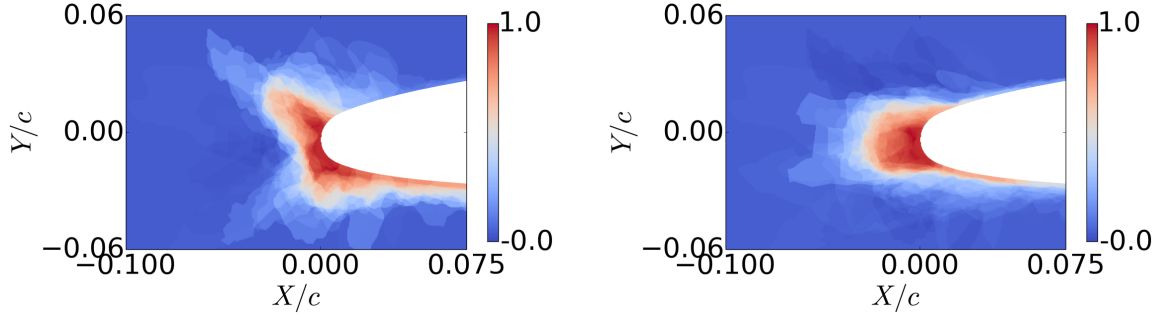
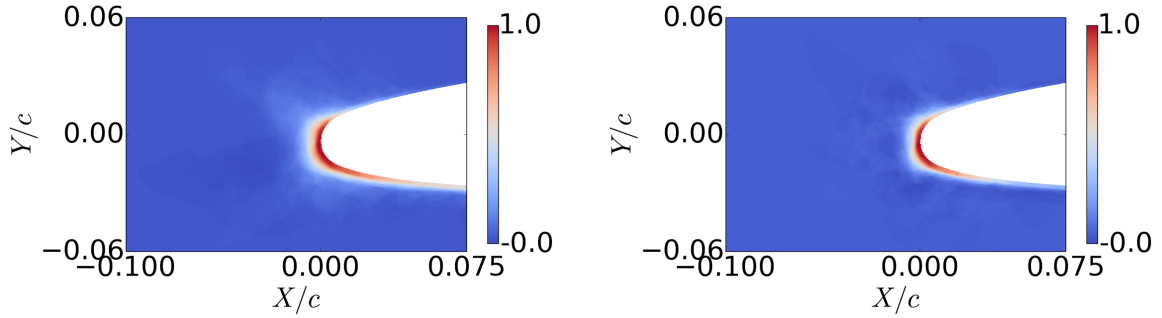


Figure 4.23: Distribution of POD coordinates, colored according to physical parameters (accretion time, static temperature, and liquid water content).

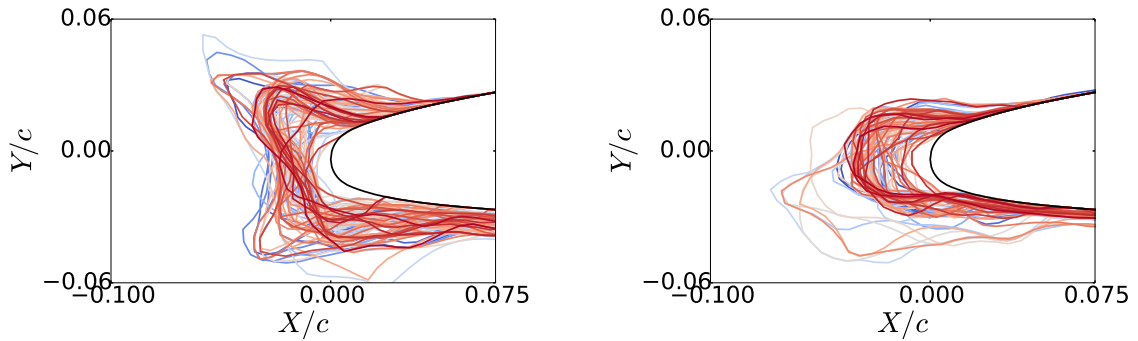


(a) Time > 10 min, temperature > -10°C , LWC > 0.45 g/m^3 . (b) Time > 10 min, temperature < -10°C , LWC < 0.45 g/m^3 .



(c) Time < 10 min, temperature > -10°C , LWC > 0.45 g/m^3 . (d) Time < 10 min, temperature < -10°C , LWC < 0.45 g/m^3 .

Figure 4.24: Purely data-driven model ice shapes.



(a) Time > 10 min, temperature > -10°C , LWC > 0.45 g/m^3 . (b) Time > 10 min, temperature < -10°C , LWC < 0.45 g/m^3 .

Figure 4.25: Random ice shapes (100 samples each) for physical conditions matching horn (*left*) and rime (*right*) accretion.

4.5 Database Partitioning

One of the observations we made in the last section regarded the idea that the shapes in our database can be classified with respect to physical conditions. For example, we saw that accretion time seemed to give a very stark division between smaller and larger shapes. Motivated by this, we use this section to expound on the topic of clusters within our database. We have already seen how shapes may be classified and the database filtered on the basis of physical conditions, so that topic needs no further discussion. Rather, our focus in this section is how clusters in the database may be deduced without using physical information at all. The clustering we perform in this section is done solely on the basis of the shape of the ice. This approach could be useful in terms of improving the POD model: if we separate a database into groups based on maximizing in-group shape similarity, then we could potentially obtain two highly accurate POD models (one for each group) as opposed to one model of lower accuracy. It might also be useful as an empirical analysis tool: we often assume that ice shapes grouped together on the basis of similar physical conditions will look similar, but to what extent is that true? This method of grouping shapes can help answer that question.

In order to cluster the shapes in our database into self-similar groupings, we must first have some means for quantifying how similar two shapes are. As we described last section, an individual ice shape element of our database consists of a vector of binary elements, which indicate whether or not there is ice at a particular location on a 2D background mesh. Defining our ice data in this way makes several similarity/dissimilarity metrics possible. We investigate the use of two different similarity metrics, each with two different approaches to clustering. The first involves defining similarity using the “exclusive or” (XOR) of two different ice shapes, computing the weighted graph Laplacian of the ice database under this metric, and then finding an optimal graph partition using the Fiedler vector. In the second method, we define similarity as the distance between the

POD coordinates of two different ice shapes, and then apply k -means clustering on the POD coordinates. As we will see, both methods yield similar clustering results.

4.5.1 The XOR Similarity Metric

Our present goal is to first connect the ice shapes of our database by using the XOR of two ice shapes as a measure of similarity. We interpret the ice shape database as an undirected graph, consisting of a collection of vertices and edges (V, E) , corresponding to the ice shapes and their associated similarities. In order to define similarity between two ice shapes, we begin by computing the XOR of two ice shapes (an example of which is shown in Fig. 4.26).

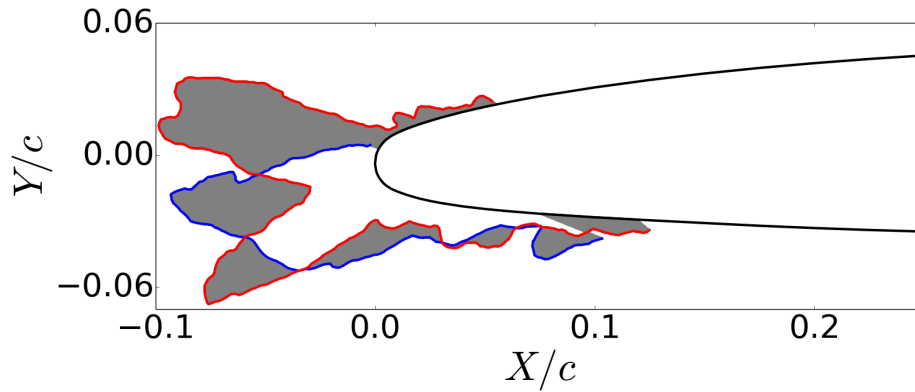


Figure 4.26: Example depiction of the XOR of two ice shapes.

The sum of this vector yields a measure of distance – two ice shapes are identical if the XOR of them equals zero, and dissimilar if it is large. Visually, this distance measure we are using corresponds to the total volume of ice that two ice shapes do not share in common (shaded grey in Fig 4.26). Finally, we define similarity between two ice shapes

by applying a Gaussian kernel function to the XOR summation:

$$w_{ij} = \exp\left(-\frac{1}{2} \frac{d_{ij}^2}{\sigma^2}\right)$$

$$d_{ij} = \sum_k^{N_G} [\text{XOR}(x_i, x_j)]_k$$
(4.6)

here, x_i denotes an ice shape defined on the 2D background mesh, and N_G denotes the total number of grid points on that mesh, d_{ij} denotes the distance between two shapes, and w_{ij} denotes the associated similarity. In general, there are no solid theoretical guidelines for choosing the user-defined parameter σ . Intuitively, it must be on the order of the length scale one expects to observe in connected clusters. This can be estimated from the database itself: compute the vertex-to-vertex distances between all vertices in the dataset, and choose a value of σ to represent some measure of locality that reflects the resulting statistics. We display the database distance statistics in Fig. 4.27.

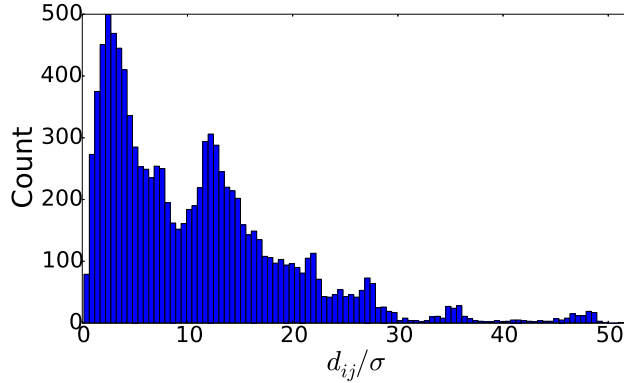


Figure 4.27: Distribution of point-to-point XOR distances, normalized by $\sigma = 3000$.

In Fig. 4.27, we have chosen $\sigma = 3000$. Notice that this choice of σ is large enough that the first (and largest) peak of the distance distribution is within about 5σ , and small enough that the remainder is outside of that range. Thus, we have implicitly defined locality based upon the first/largest peak in the distance distribution. Finally, in order to sparsify the distance matrix, we round any value of w_{ij} corresponding to a distance d_{ij} greater than 3σ in magnitude to zero. It might be objected that this choice of σ appears too small, given the

range of the distance statistics in Fig. 4.27. Choosing σ is a bit of a balancing act: too small of a choice gives low graph connectivity (which defeats the purpose of making a connected graph); too large of a choice gives high graph connectivity, which makes partitioning and identifying clusters more difficult (since, in such a case, all elements are closely connected to one another). Aware of this, we experimented with several values of σ , and found that this value yields a good trade-off between low and high graph connectivity.

4.5.2 Graph Laplacian and Spectral Partitioning

Equipped with both distance and similarity metrics, we are now in a position to calculate the graph Laplacian and use it to partition the graph. We do not wish to divert focus to a lengthy description of spectral partitioning; we briefly note the basics here. The interested reader is referred to the literature for details [58, 97, 31].

The Laplacian is defined as:

$$L = D - W \tag{4.7}$$

where W is the similarity matrix with entries w_{ij} (defined, for our setting, in Eq. 4.6), and D is the degree matrix with entries $d_i = \sum_{j=1}^N w_{ij}$ (N is the number of database elements). The eigenspectrum of this matrix provides useful information concerning the connectivity of the graph. In particular, consider an ordering of the eigenvalues $\lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_N$. The k eigenvalues equal to zero indicate the number of connected components in the graph, and the eigenvector corresponding to λ_{k+1} (referred to as the Fiedler vector) gives a grouping of the vertices which indicates connected subgraphs.

The ordered eigenvalues of the Laplacian are shown in Fig. 4.28. As can be clearly seen, the first 49 eigenvalues are zero (to machine precision), with a clear jump to a nonzero eigenvalue λ_{50} . One way to visualize the clustering produced by the Fiedler vector is to

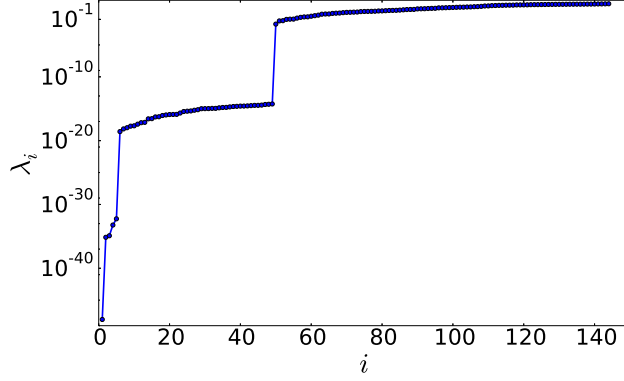


Figure 4.28: Eigenvalues of the Laplacian (ordered by magnitude).

sort the vector components by magnitude and re-order the distance (or similarity) matrix by this indexing; this is shown in Fig. 4.29.

The eigenvectors corresponding to $\lambda = 0$ give disconnected clusters within our database. As we mentioned, there are many of these. This is caused by our choice of the parameter σ in defining the shape similarity. What we find as a result of this choice is that many of the large accretions are dissimilar enough from any of the other shapes that they effectively form their own unconnected subclusters. However, there is one large cluster of 94 elements, formed by the smaller ice accretions, which is revealed by one of the zero eigenvectors and is shown in Fig. 4.29c. The subpartitioning of this cluster according to the Fiedler vector (Fig 4.29f) is fairly trivial – it simply involves cutting off the single most dissimilar horn. The eigenvector corresponding to the next smallest eigenvalue partitions off a larger subgroup of accumulations (Fig. 4.29i).

As we stated earlier, one of the potential uses of this clustering scheme is to demonstrate a means for doing empirical ice classification, as opposed to classifying shapes based on the underlying physical conditions. We can compare the modal coordinates of the clusters generated to the modal coordinates colored by physical conditions (see Fig. 4.23) to get a sense of how similar/dissimilar these schemes are. Unsurprisingly, the first graph partitioning (formed by the zero eigenvector) corresponds almost exactly to the partitioning that would occur on the basis of accretion time (compare Fig. 4.29d to Fig. 4.23a). Inter-

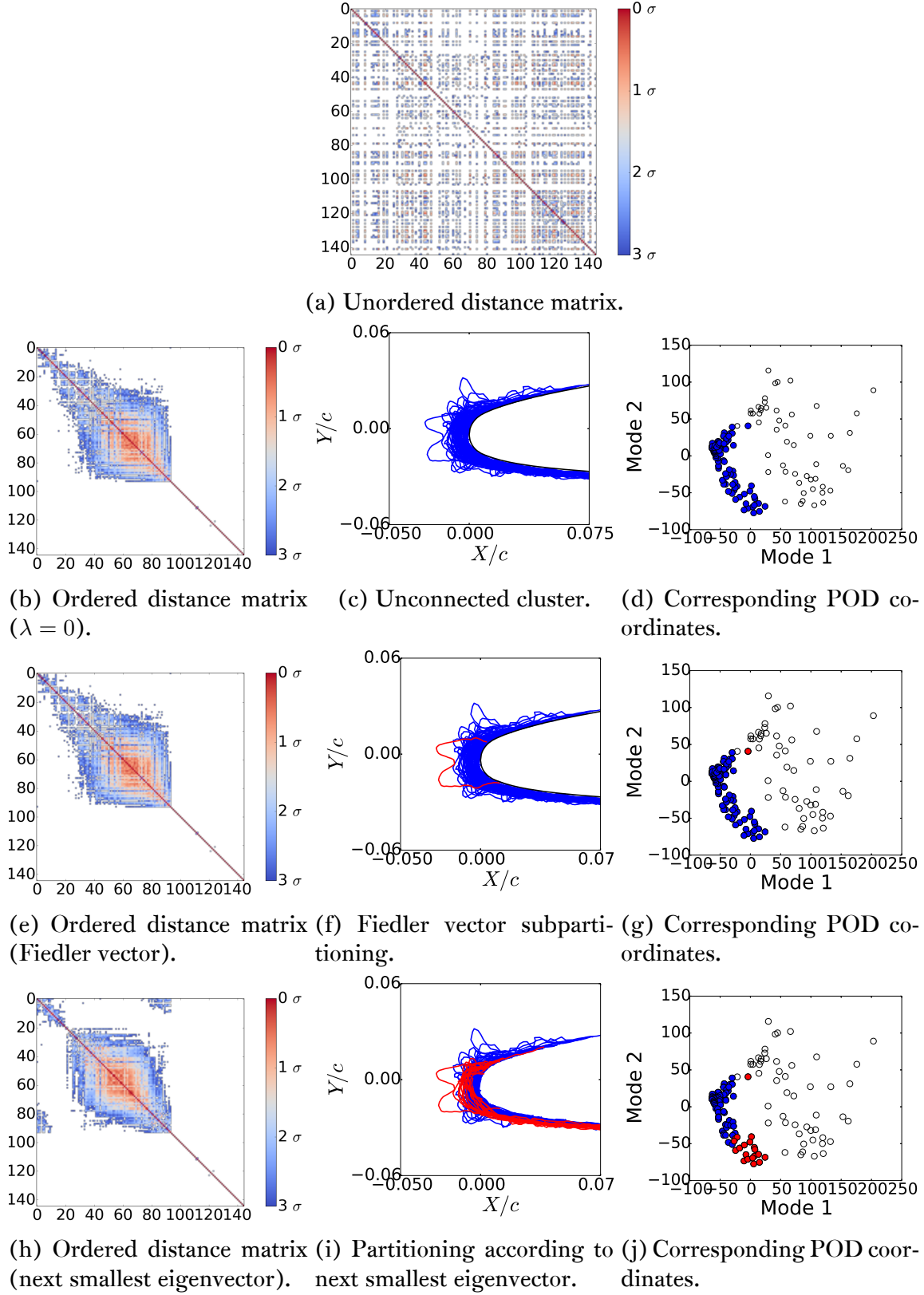


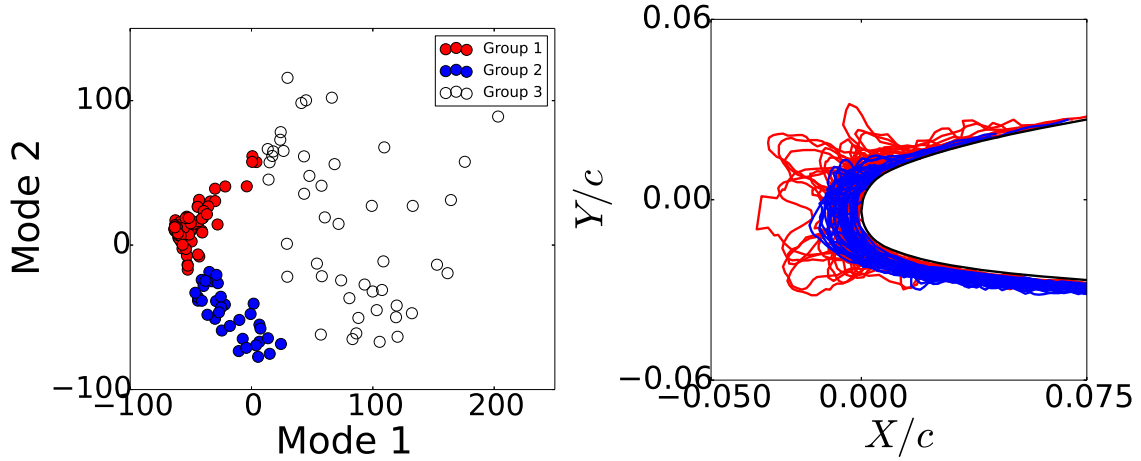
Figure 4.29: Clusters computed using the eigenspectrum of the Laplacian, as well as the associated POD coordinates.

estingly, the subpartitioning of this group does not appear to correlate much with time or temperature, although it does seem to be related to reasonably high values of LWC (compare Fig. 4.29j to Fig. 4.23e). This subpartition of shapes appears to be a combination of rime-like accretions and smooth upper-surface horn accretions that are clearly visually differentiated from the rest of the group, although that would not necessarily be expected on the basis of the temperature/LWC combinations present. It is important to recall that our database ice shapes were originally generated on one of three different airfoils (before being scaled onto a base airfoil for our analysis), so it is possible that this might account for the subdivision we have observed. Notwithstanding that, however, these observations demonstrate the utility of this empirical taxonomy scheme: sometimes, the rime/glaze accretion behaviors might not follow the “rules of thumb” that one would expect, and so if we wish to separate similar shapes in some general dataset, it might benefit us to do so using a partitioning scheme which is independent of the physical conditions.

4.5.3 Clustering in POD coordinates

We have already demonstrated one method for partitioning the graph of ice shapes using a similarity metric and the associated graph Laplacian. Another option would be to perform clustering in POD coordinate space (which we have already computed). The distance metric in this case would simply be the Euclidean distance. The clustering algorithm we choose to use is k -means [57, 56, 39].

The results of applying this clustering technique are shown in Fig. 4.30. In this figure, we have applied 2-means clustering twice in a recursive fashion: we apply it once to separate the database into two groups, and then apply it again to one of the groups. The first application of 2-means separates the small from the large ice shapes in almost the same way as the Laplacian spectral clustering does, with the notable exception of a few large horns and one large rime shape that do not visually appear to fit with the rest of the group. The second round of 2-means separates this group into two subgroups. As



(a) Clustering of POD coefficients in leading two POD modes.

(b) Corresponding ice shapes.

Figure 4.30: K-means clustering of ice shapes in POD coordinates. Groups 1 and 2 were produced as a single group after applying 2-means; applying it again yields the blue/red division.

with the subdivision produced with spectral clustering, this subdivision seems to separate the rime-like accretions from the horn-like shapes. Interestingly, there do not appear to be any distinctions between these subdivisions on the basis of physical information (i.e., accretion time, temperature, or LWC). Again, it is possible that different original base airfoils might partially explain this phenomenon, but nonetheless it is still illustrative of the use of the empirical classification scheme, which might find data trends unrevealed by physical conditions.

Comparing the two clustering methods, we see advantages and disadvantages to both. Spectral clustering produces a better separation between the large and small ice accretions, but it could be argued that visually, k -means produces a qualitatively better subdivision between the horn and rime-like accretions within the group of small shapes. Regardless, these differences are slight, and it seems that either method produces satisfactory clusters.

4.5.4 Cluster Modeling

We have already seen how clustering can help classify shapes based on similarity. A useful consequence of this is that it is often easier to model a cluster than it is to model the entire database, since the cluster has less variation. We briefly demonstrate this here on the cluster of small ice shapes.

Modeling is straightforward; we simply apply POD to this group of shapes. The first advantage of limiting ourselves to only the smaller shapes is that because these shapes are less extreme, they can be cast into arc-length coordinates (which cannot in general be done with the larger shapes, since some would be multi-valued in arc-length coordinates). Doing this is advisable, because it effectively reduces the spatial dimensionality of the ice, which makes it possible to capture more of the ice features with fewer modes.

We show the transformation of the first cluster into arc-length coordinates in Fig. 4.31. The mean and leading POD modes, as well as the cumulative sum of the POD eigenvalue magnitudes, are shown.

As can be seen, the first four POD modes captures about 80% of the total variation in the dataset, and the first eight capture about 90%; diminishing returns occur after retaining higher order modes in the model. For the purposes of performing UQ on the shape variation, there is a trade-off between retaining more POD modes to ensure a well-resolved model, and retaining less modes to keep the dimensionality of the parameter space low. We therefore choose to truncate the expansion at 5 modes.

4.5.5 Cluster Uncertainty Quantification

One of the benefits of doing the cluster analysis is that when we restrict our analysis to the cluster of closely related shapes, we are more likely to be able to apply the machinery of PCE (as opposed to the LHS we were forced to use last section), and in so doing, we

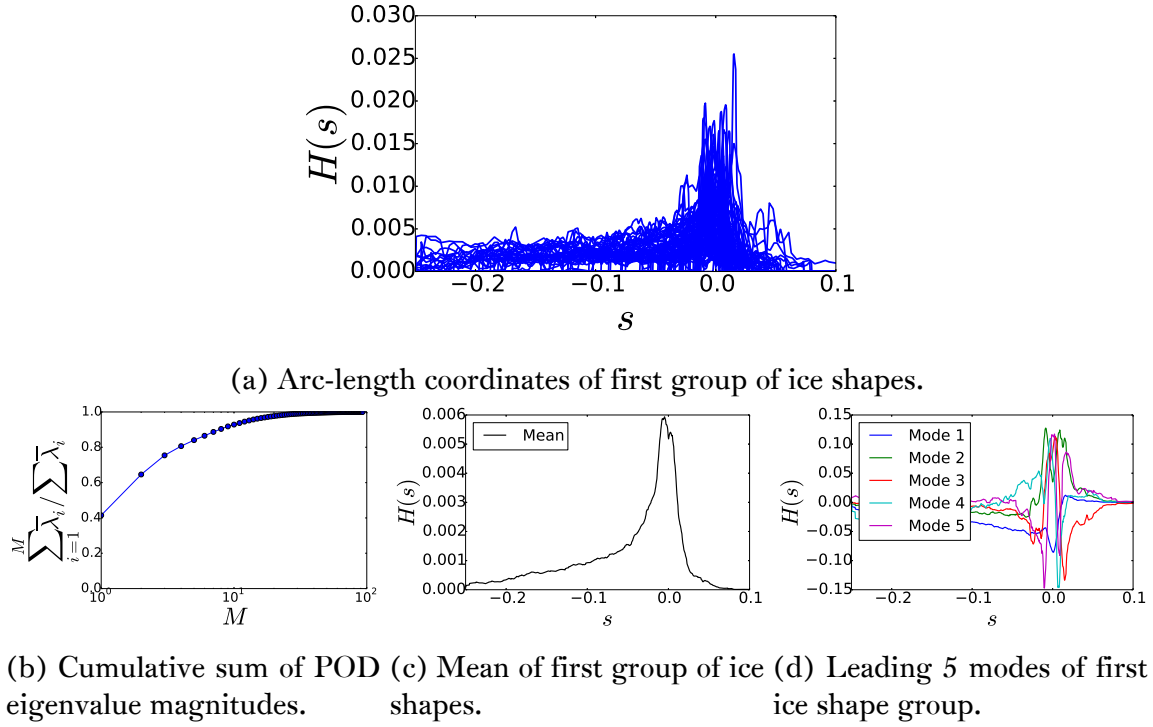


Figure 4.31: Modeling of the first group of ice shapes.

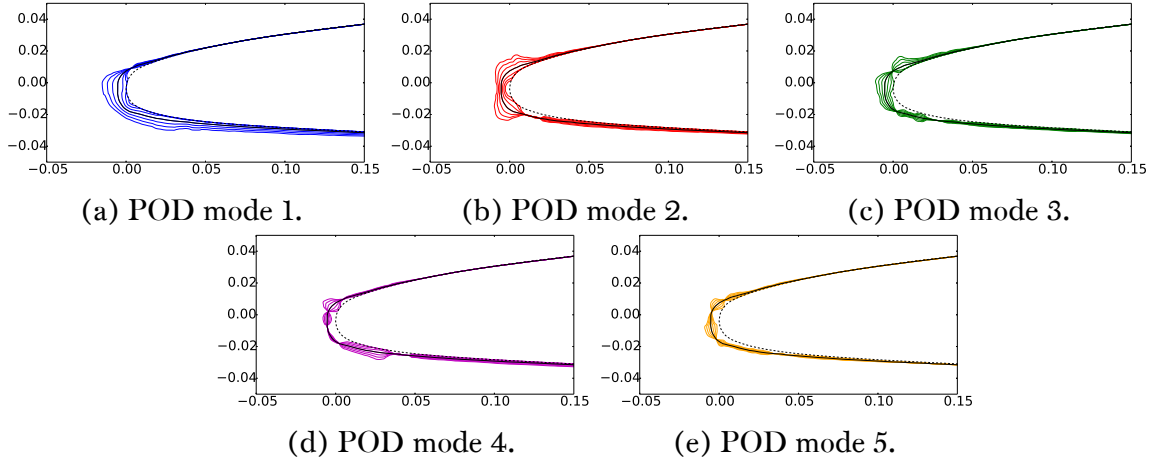


Figure 4.32: Shapes corresponding to varying each of the individual POD modes about the mean shape, in increments of one standard deviation, up to ± 3 standard deviations.

retain all of the benefits of that approach (i.e., numerical surrogate, easy global and local sensitivity computations, etc.). Here we use the PCE approach to study the statistical effects that ice shape variation (as derived from the cluster of small shapes) has on airfoil aerodynamics. As our entire work rests on a data-based approach, we wish for our results to reflect the data we have. Thus, we should attempt to use a PCE input distribution that is in decent agreement with the empirical statistics from the data.

We choose to fit a normal distribution to each of the empirical POD coefficient statistics, and thus we use the Hermite polynomials in the PCE method. Fig. 4.33 reveals that, although not perfect, the normal distribution fit is a reasonable choice.

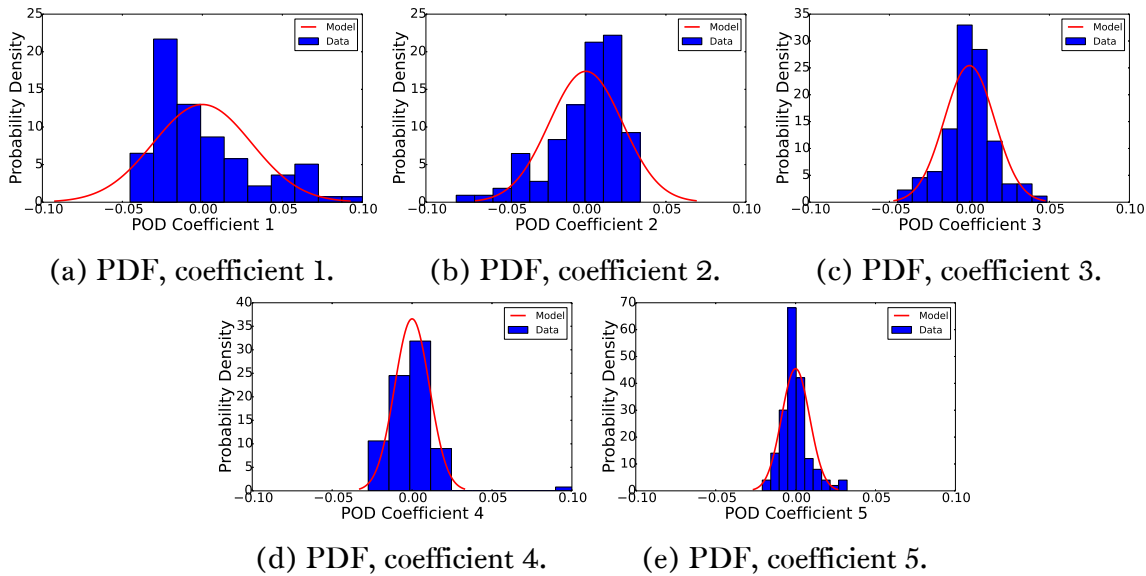


Figure 4.33: Statistics for the leading POD coefficients (for the first cluster of ice shapes). The blue histograms represent the statistics of the dataset; the red curves represent the result of fitting a Gaussian distribution to those statistics.

We perform PCE on the 5-dimensional parameter space described above using adaptive sparse grid sampling, which is implemented in DAKOTA. The clean airfoil shape is the business jet geometry, the angle of attack is set at $\alpha = 3^\circ$, and the Reynolds number is set at $Re = 7.5 \times 10^6$. These latter two conditions are chosen to be consistent with the flight conditions represented in the original dataset. There is indeed variation in both the angle of attack and Reynolds number in that dataset; however, the main goal of our UQ study

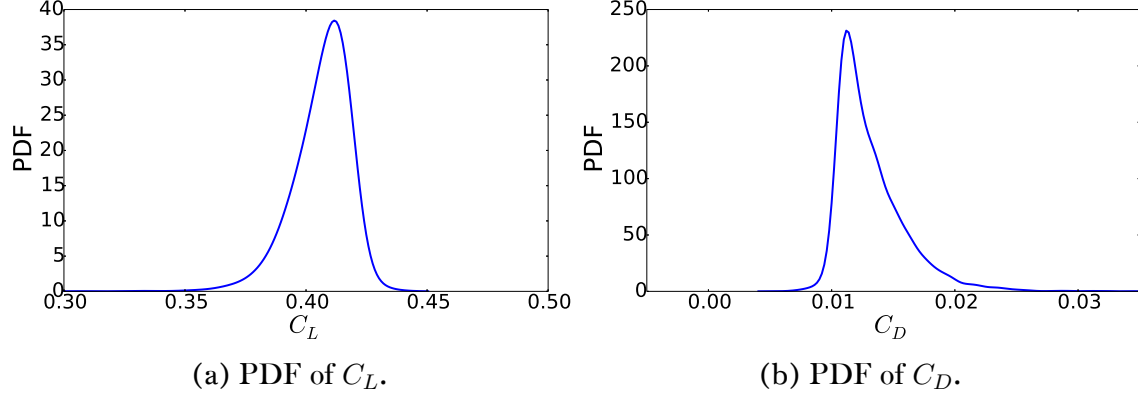


Figure 4.34: Probability density functions for output metrics, approximated using 10^4 Monte Carlo samples of the PCE surrogates and Gaussian kernel smoothing.

is to quantify the effects that ice shape variation has, hence, we do not allow either angle of attack or Reynolds number to be uncertain parameters. We quantify the uncertainty in the output metrics C_L and C_D .

Fig. 4.34 displays the output statistics that result from sparse grid PCE using 487 function evaluations. This is the number of evaluations needed to converge the change in the \mathcal{L}_2 norm of the response covariance matrix to a pre-set tolerance of 1×10^{-4} . Of course, if higher resolution is desired, more evaluations may be performed.

We can also examine the surrogate for input-output sensitivities, correlations, and dependencies. One way to do this is to Monte Carlo sample the surrogate, rank the results according to a metric (e.g., C_L), and then look at the statistics of the extremities. This method is a global measure of sensitivity, in the sense that the entire parameter space is surveyed and analyzed. This information is displayed in Fig. 4.35a, 4.35b. We see that modes 2 and 3 have a strong coupling effect on C_L . If we examine the shapes corresponding to the mean values of the top/bottom deciles of C_L (Fig. 4.35c), we conclude that modes 2 and 3 couple to produce upper/lower surface horns, which give the deterioration in performance observed.

Another means of examining sensitivity – which is local in nature – is to evaluate the gradient of the PCE surrogate at some location in parameter space (usually the mean), and

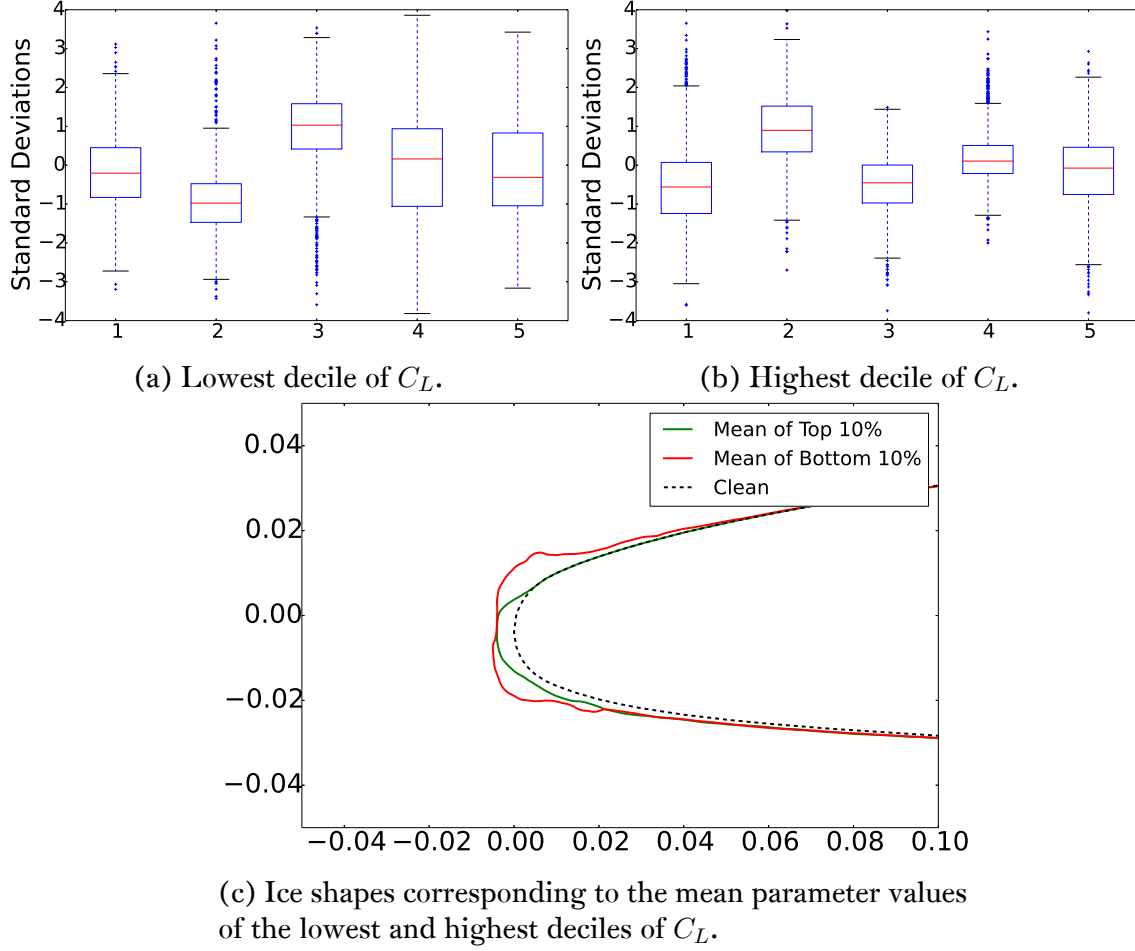


Figure 4.35: Global parameter space analysis: lowest/highest deciles of C_L . Boxplots show clustering in parameter space of the lowest and highest deciles of C_L . Data consists of 10^4 Monte Carlo surrogate evaluations.

examine the shape perturbations that result. This information is displayed in Fig. 4.36. As can be inferred, both the global and local measures of sensitivity reveal almost the same trends: lift decreases as ice mass is added to form lower and upper surface horns, and increases as ice mass decreases (particularly on the upper surface).

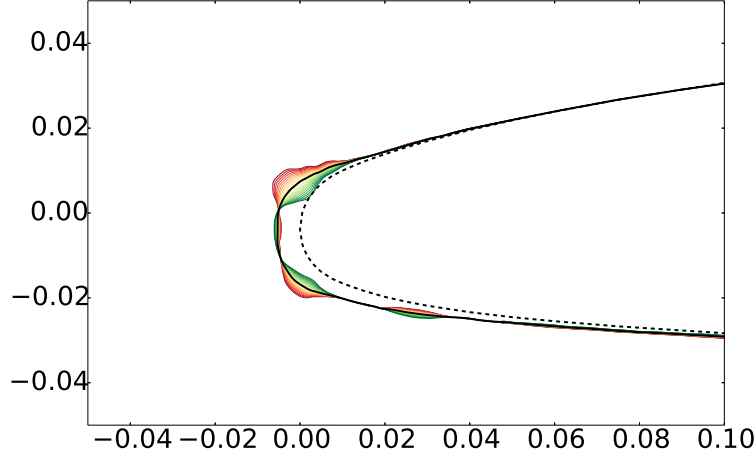


Figure 4.36: Local sensitivity: ice shapes corresponding to the gradient of C_L with respect to the uncertain parameters (evaluated at the uncertain parameter means). Red to green transition indicates lower to higher C_L .

4.6 Summary

The purpose of this chapter was to introduce a new perspective on the topic of airfoil icing, and apply it to the problem of ice modeling and uncertainty quantification. The motivation for this approach is apparent from a brief survey of the existing literature. What we find, broadly speaking, is that an extensive amount of research has already been done in two arenas – physical/computational and experimental. Many papers have been published that model the physics of ice accretion and build deterministic computational models from that (as we will discuss next chapter), and many experiments have been performed at various icing conditions in icing wind tunnels. However, no one has yet attempted to build ice accretion models purely from data, and study the variations produced by these models. This can be viewed as a “middle ground” approach between purely computational and purely experimental, and it has several benefits.

The first benefit we observed is that we can distill a set of spatial shape parameters from a database of ice shapes. This was the initial motivation for the data-based modeling: to derive a set of ice shape parameters that are no longer heuristic (as was the case in Chapter

3), but instead are “realistic” in the sense that they are produced using real data. In this way, it becomes possible to make the subsequent UQ studies similarly realistic.

We encountered the second benefit of this approach when we connected the modal coordinates of ice shapes in a database to the underlying physical conditions used to produce the shapes in that database. This made it possible to examine statistical trends between physical parameters and resulting shapes. The ultimate product of this was a purely data-driven, statistical model of ice accretion, whereby it is possible to input a desired range of physical icing conditions and generate a random sampling of ice shapes corresponding to those conditions. The implications of this could be profound in the icing community, as no one has heretofore attempted to introduce experimental data into ice accretion models. We foresee that this data-driven modeling could be used to benchmark and improve numerical codes.

Lastly, we saw how we can derive empirical classifications from the variations present in a dataset. This is useful as an evidentiary tool for exploring what, if any, links exist between physical icing conditions and ice shape. Sometimes, we saw that the clusters produced in a dataset trended cleanly with intuitive parameters (such as accretion time, which differentiates large and small shapes). Other times, we produced visually self-similar clusters within our dataset that, surprisingly, would not have been expected on the basis of physical conditions. This evidence-based method of separation gives a tool to icing researchers that can help either support or undermine the traditional classifications of ice accretion that are well-known.

Going forward, there are many extensions to this research that might be explored. First, the main concern with any data-based approach is always the quality of the data itself – results are only as “good” as the underlying data. The experimental dataset we made use of in this chapter is reasonably comprehensive, in the sense that it represents a diverse range of conditions, but it certainly could be improved by adding more shapes. For instance, a large number of the shapes in that database were low-time accretions, which does bias the

results/models toward those types of shapes. Second, an interesting avenue of research would be to explore how one might fuse the data-driven models we have produced with numerical icing codes. It might be possible to add empirical corrections to these numerical codes that are based on predictions from the data-based models.

Chapter 5

Computational UQ

The purpose of this chapter is to approach the icing UQ problem from a fundamentally different perspective than those which have been presented thus far. The last two chapters dealt with quantifying uncertainty present in the actual shape of the ice; however, the *source* of that uncertainty was only briefly addressed. In this chapter, we seek to close that gap by considering uncertainties which are more fundamental to the icing process. The parameters involved are physical quantities that govern the ice growth, and hence our goal is to quantify the effects of uncertainties in governing macroscopic physical parameters on iced airfoil performance. Research of this nature is particularly germane for certification procedures, as the parametric trends and statistics generated can inform a decision regarding whether flight is “safe” under a given set of circumstances.

In order to investigate these questions, we must first have the appropriate tools for doing so. We choose to adopt a strictly computational approach, for which there is already an extensive body of literature [96, 32, 102, 94, 81, 64, 100, 65]. The semi-empirical models described in these works form the basis for a computational icing code that we develop in this chapter. In a UQ setting, hundreds or thousands of different icing scenarios would need to be evaluated quickly, easily, and inexpensively. For this reason, it is particularly

important that we use a method for quantifying uncertainties that is both efficient and accurate.

The structure of this chapter is as follows. We first introduce some necessary background material on the physics of airfoil ice accretion. Next, we introduce the computational ice accretion code CATFISH (Coupled Aero-Thermodynamics For Icing Simulations), which we developed in-house as a part of this thesis, and which we will be using to perform parametric UQ studies. We first briefly sketch the structure of CATFISH from a system-level perspective, and then delve into greater detail on the individual component sub-modules of the software. After benchmarking this code against published results in the literature, we proceed to design and perform several UQ studies on the physics of airfoil ice accretion, and comment on their relevance to airplane safety.

5.1 Overview of the Icing Process

We have already presented a detailed account of the various categories of airfoil ice in Chapter 3. We need not repeat that information here; instead, we focus on a brief description of the process of icing, which is necessary to understand how our icing code (CATFISH) is structured.

Airfoil ice accretion exists due to the presence of micro-scale liquid water droplets present in the atmosphere. The size of an individual droplet can be highly variable (diameters ranging from around $10\mu m$ to greater than $1000\mu m$ are possible); however, it is possible to describe the statistical distribution of size for different icing conditions [6]. The temperature of the droplets depends on the ambient temperature, which may be near the freezing mark or well below it (e.g., -20°C). Droplets may exist as supercooled water in this state, due to lack of a nucleation site and insufficient heat transfer mechanisms to completely remove the latent heat of freezing. The amount of water mass present in the

atmosphere is measured as a bulk quantity and is referred to as the Liquid Water Content (LWC). Typical values of LWC tend to range from $0.1g/m^3$ to $1g/m^3$.

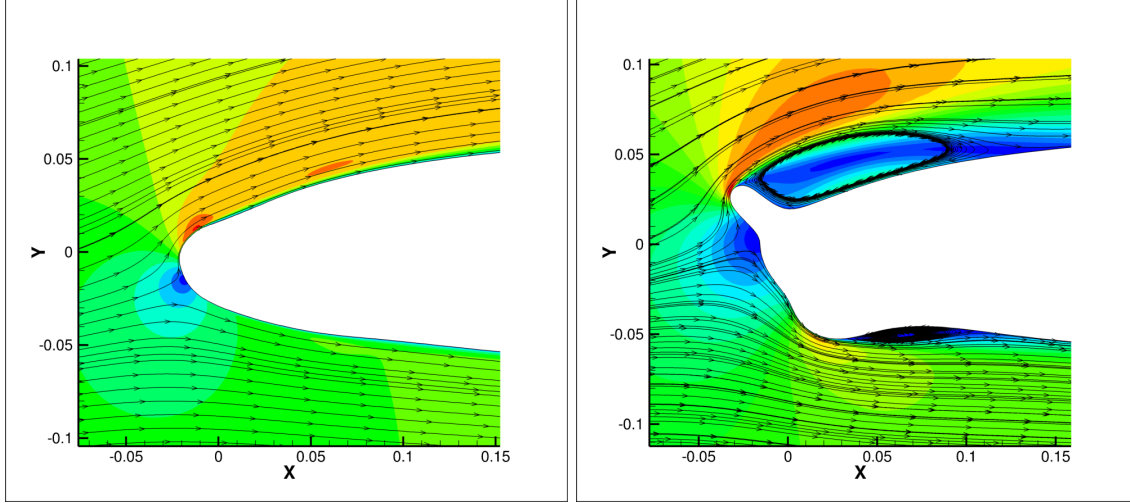
Impingement of the droplets onto the airfoil surface is the first step in ice accretion. The exact trajectory of an individual droplet is determined by a combination of the local flowfield surrounding the airfoil and the mass of the droplet. As would be expected on physical grounds, smaller, less massive droplets are more easily deflected away from the airfoil by the pressure gradients near the airfoil leading edge, whereas larger, more massive droplets have more inertia and hence tend to follow more ballistic trajectories, relatively unaffected by the local airfoil flowfield. Once a droplet strikes the airfoil surface, several phenomena may occur: the droplet may rebound off of the surface and continue advecting downstream away from the airfoil, or it may spread into a thin film on the surface, or it may splash, depositing a fraction of the mass onto the surface and ejecting the rest. Semi-empirical research and modeling [32, 96] suggests that the prevailing mechanism is determined by a combination of the Weber number (which measures the ratio of inertial to surface tension forces) and Reynolds number (which measures the ratio of inertial to viscous forces).

Once droplets impinge on the surface, they may freeze immediately, merge into a thin lubricating film, or may coalesce into beads held together by surface tension effects [37, 66, 67]. The first case is observed in conditions which are relatively cold with relatively low LWC - in such scenarios, it is cold enough that all of the impinging water freezes immediately, and the resulting ice shape closely follows the airfoil profile (referred to in the literature as “rime” accretion; see the description in Chapter 3). If, instead, the temperature and LWC combination is such that there is not enough heat transfer to completely remove the latent heat of freezing of the incoming water, a thin film may form on the surface. This film is sufficiently thin that it is generally assumed to convect along the airfoil surface driven by the aerodynamic shear stress at the wall [10]. This transport process redistributes water mass along the surface of the airfoil and provides a partial

explanation for why ice shapes do not always perfectly match the impingement profile. Some fraction of this film will freeze into ice; exactly how much depends entirely on the heat transfer mechanisms present to remove the latent heat of freezing. As will be discussed later, several such mechanisms exist, but the one which dominates is aerodynamic cooling by convection [37]. This cooling mechanism has an unequal distribution over the surface of the airfoil, and hence (under certain conditions) can give rise to unequal rates of freezing, producing ice “horns” (see the description in Chapter 3). As noted, not all unfrozen water necessarily merges into the thin film; some may coalesce into micro-scale beads. The formation and freezing of micro-scale beads has the effect of creating surface roughness, which energizes the boundary layer and promotes turbulence. The main effect of this phenomenon (from an icing viewpoint) is to increase the local heat transfer by convection, since increasing turbulence promotes greater mixing within the boundary layer [37].

Aerodynamically, the effects of icing depend heavily on the type of icing that occurs, which in turn depends on the governing physical parameters. As we have noted already, horn ice is generally more dangerous than rime ice, simply because of the geometric effects on the flow (see Fig. 5.1 for an illustration of the flowfield differences between the two).

In summary, there are two main physical processes that take place in concert to determine the ice shape for a particular set of icing conditions. The first is the advection and impingement of atmospheric liquid droplets, the details of which are determined by the flowfield surrounding the airfoil. The second process is the surface thermodynamics, which determines how much of the impinging water freezes, and how much of it is advected back along the surface. These two processes form the basis of the structure of CATFISh.



(a) Rime ice. $T_\infty = 250.37K$, $LWC = 0.55g/m^3$, $MVD = 20\mu m$, $Re = 7.76 \times 10^6$, $M = 0.32$, $T = 5$ min.
(b) Glaze ice. $T_\infty = 262.04K$, $LWC = 0.86g/m^3$, $MVD = 20\mu m$, $Re = 5.33 \times 10^6$, $M = 0.21$, $T = 5$ min.

Figure 5.1: Comparison of flowfield differences between rime and glaze accretions (calculated using CATFISH). Contours represent velocity magnitude. Conditions noted are free-stream temperature (T_∞), Reynolds number (Re), Mach number (M), liquid water content (LWC), mean volumetric diameter (MVD), and accretion time (T).

5.2 Code Structure

In this section, we briefly outline the structure of the computational airfoil icing code CATFISH, which we developed as a part of this thesis for the purpose of investigating the statistical effects of uncertainties in the physics of icing.

CATFISH integrates an existing aerodynamic code (mesh generator/flow solver) with a droplet advection module and a thermodynamic module in order to solve for time-dependent ice growth. A system-level depiction of these components is given in Fig. 5.2. The mesh/flow solver used has been described previously in this thesis, and consists of FLO103 along with an in-house hyperbolic grid generator. The droplet advection module advects a screen of particles from the free-stream at the wing using a Lagrangian particle approach. Impingement characteristics (i.e., splashing, bouncing, deposition) are simulated and a rate of water mass deposition onto the surface is calculated. The thermodynamic module solves for local rates of ice accretion on the surface by solving

mass/energy conservation partial differential equations. Finally, ice growth is calculated for a specific interval of time by adding the calculated amount of ice mass normal to the surface of the airfoil. This entire process (mesh/flow solver to incremental ice growth) can be repeated in a time-stepping fashion in order to grow ice for an arbitrarily long period of time.

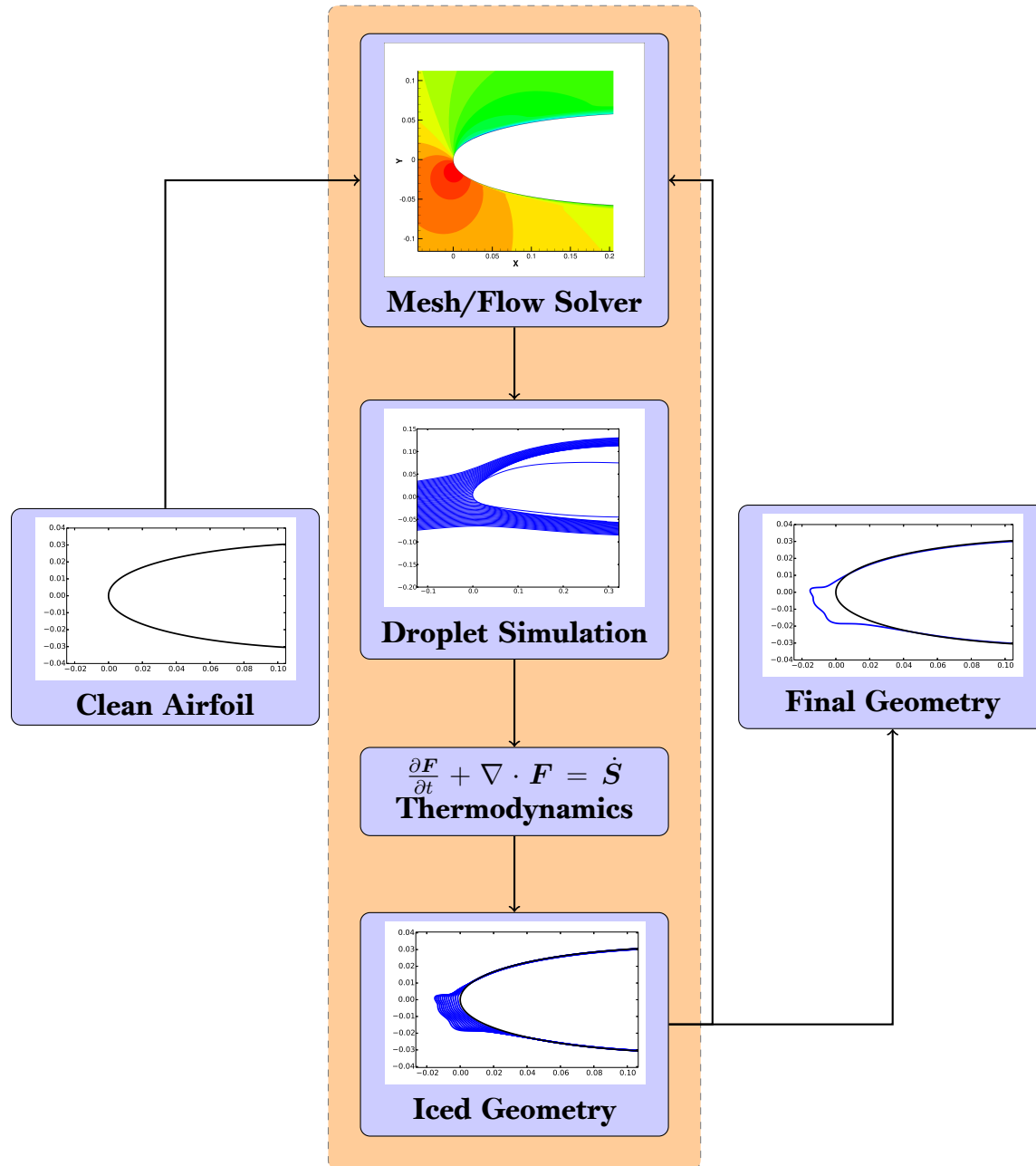


Figure 5.2: Modular structure of CATFISH

5.3 Droplet Simulation Module

The first module in CATFISh handles all of the physics related to droplet advection. Relevant inputs include droplet size distribution, LWC, free-stream temperature, and the appropriate mesh/flow solutions.

Some decisions and assumptions must be made in order to simulate the droplet advection in order to make the problem computationally tractable. To illustrate this point, we begin by performing a simple estimate the number of particles impinging on an airfoil surface given a set of “typical” aerodynamic/icing conditions. Let us assume the following conditions: $LWC = 0.5 \text{ g/m}^3$, $MVD = 20 \text{ }\mu\text{m}$, $U_\infty = 100 \text{ m/s}$. Assuming perfectly spherical droplets, we calculate that there are approximately 1.2×10^8 droplets per cubic meter in ambient conditions. Given the free-stream velocity of the airfoil, we further estimate a flux of approximately 1.2×10^{10} (i.e., 12 billion) droplets per cubic meter per second. It is also important to note that total ice accretion times of interest routinely range from tens of seconds up to 10 minutes.

5.3.1 Droplet Advection

It should be evident from this simple calculation that a full-scale, time-resolved simulation of all of the impinging particles for an icing computation is not feasible. Therefore, some simplifying assumptions must be made. The first assumption that is made is that all of the water which impinges on the airfoil may be accurately modeled by a continuous thin-film distribution of water. This assumption is valid for calculating the macroscopic distribution of impinging water, but (as noted in the previous section) ignores potential micro-scale physics involved with the accumulation of water beads held together by surface tension. One way to model some of this micro-scale physics is to introduce a model for surface roughness into the turbulence model of the aerodynamic calculation; this option will be discussed in the next section. In any event, this assumption of a surface thin-film shifts the

goal of this module from simulating each and every single droplet to calculating a flux of water across the surface of the airfoil. This flux is measured in the icing community as a fraction of the free-stream water droplet flux, and is referred to as the collection efficiency.

There exist (at the time of this writing) two main approaches for modeling droplet physics – an Eulerian formulation [10, 9], and a Lagrangian formulation [32, 96, 94]. The former method seeks to model the air as essentially a two-phase continuum (dry air plus water content) and proceeds by supplementing the mass/momentum conservation equations for the dry air (i.e., Navier-Stokes equations) with those for water content [10, 9]:

$$\begin{aligned}\frac{\partial \alpha}{\partial t} + \nabla \cdot (\alpha \mathbf{u}_d) &= 0 \\ \frac{\partial \mathbf{u}_d}{\partial t} + \nabla \cdot (\mathbf{u}_d) &= \mathbf{F}_d\end{aligned}\tag{5.1}$$

here, the variables α , \mathbf{u}_d , and \mathbf{F}_d are, respectively, mean values of the nondimensional water volume fraction, droplet velocity, and droplet forces (or equivalently, droplet velocity sources/sinks) over a small fluid element at a specific spatio-temporal location. Solving these conservation PDEs yields the water content distribution over the surface of the airfoil, which can be converted to a collection efficiency.

In contrast to the Eulerian method, the Lagrangian approach begins by assuming that the collection efficiency may be approximated by advecting a screen of uniformly-spaced particles at the airfoil and observing the resulting distribution of impacts on the surface. Fig. 5.3 shows a cartoon depiction of this procedure. Assuming no mass loss due to splashing/bouncing, it is straightforward to see (by conservation of mass) that the local collection efficiency (the ratio of the free stream droplet flux to the flux over the airfoil surface) is given by the ratio of the spacings in the figure; that is,

$$\beta(s^*) = \frac{\Delta y(s^*)}{\Delta s(s^*)}\tag{5.2}$$

where s^* denotes an arbitrary location on the surface.

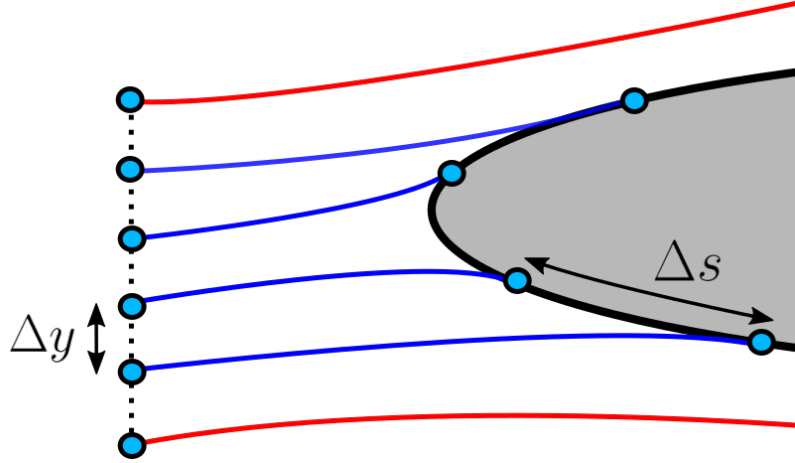


Figure 5.3: Illustration of the calculation of the collection efficiency via a Lagrangian method.

CATFISH uses the Lagrangian method to calculate the collection efficiency. The droplet volume fraction is generally small in icing scenarios; that is, the ratio of the bulk density of the droplets to that of the surrounding air is on the order of 10^{-3} . Usually, if a two-phase flow has a volume fraction loading less than 10^{-1} , then the assumption that the heavier gas phase is unaffected by the lighter phase (one-way coupling) is justified [24]. Hence, we can separate the aerodynamic flowfield calculations from the droplet trajectory calculations. A screen of uniformly-spaced particles is initiated upstream of the airfoil, and subsequently advected by the following equations of motion [32, 96]:

$$\begin{aligned} \frac{d\mathbf{x}_d}{dt} &= \mathbf{v}_d \\ m_d \frac{d\mathbf{v}_d}{dt} &= \frac{1}{2} \rho_g C_D \pi r_d^2 \|\mathbf{v}_g - \mathbf{v}_d\| (\mathbf{v}_g - \mathbf{v}_d) + m_d \mathbf{g}. \end{aligned} \tag{5.3}$$

Here, \mathbf{x}_d , \mathbf{v}_d , r_d , and m_d are, respectively, the droplet position, velocity, radius, and mass; \mathbf{v}_g and ρ_g are the gas velocity and density, \mathbf{g} denotes the gravity vector, and C_D is the droplet drag coefficient. C_D can be calculated with the Schiller and Naumann correlation [32, 96,

79]:

$$\begin{aligned} C_D &= \frac{24}{Re_d} (1 + 0.15 Re_d^{0.687}) \\ Re_d &= \frac{2\rho_g r_d}{\mu_g} \|\mathbf{v}_d - \mathbf{v}_g\| \end{aligned} \tag{5.4}$$

A choice must be made regarding how to solve Eq. 5.3. One possibility is a high-order explicit scheme (e.g., Runge-Kutta), however, the stability criterion for this method is quite restraining for small particles. An alternative method is to formulate the exact solution of Eq. 5.3 corresponding to a constant gas velocity and very small Reynolds number (i.e. Stokes drag law), and then explicitly solve that. CATFISH implements this approach, which reads [32, 96]:

$$\begin{aligned} \mathbf{x}_i^{n+1} &= \mathbf{x}_i^n + \mathbf{u}_g \Delta t + (\mathbf{v}_i^n - \mathbf{v}_g) \left(1 - \exp\left(-\frac{\Delta t}{\tau_d^n}\right) \right) \tau_d^n \\ &\quad + \left(\Delta t - \left(1 - \exp\left(-\frac{\Delta t}{\tau_d^n}\right) \right) \tau_d^n \right) \tau_d^n \mathbf{g} \\ \mathbf{v}_i^{n+1} &= \mathbf{v}_i^n + \mathbf{v}_g + \exp\left(-\frac{\Delta t}{\tau_d^n}\right) (\mathbf{v}_i^n - \mathbf{v}_g) + \left(1 - \exp\left(-\frac{\Delta t}{\tau_d^n}\right) \right) \tau_d^n \mathbf{g} \end{aligned} \tag{5.5}$$

where $\tau_d = \frac{24}{Re_d C_D} \left(\frac{2\rho_l r_d^2}{9\mu_g} \right)$ is the droplet dynamical response time (ρ_l is the liquid density).

5.3.2 Tracking Particles

The last component to droplet advection deals with determining the gas state at the location of all of the droplets. In FLO103, calculations are performed on a boundary-fitted mesh, so the problem becomes a “particle-in-cell” problem – given a particle (i.e. a droplet) location in the mesh, find that cell which it occupies.

A naive approach to this problem is clearly computationally intractable: given N_P particles and N_G cells, a brute-force algorithm (e.g., cell centroid nearest-neighbor search) requires $N_P \times N_G$ calculations (note that this must be performed each iteration).

CATFISH uses a custom algorithm to quickly and accurately solve the particle-in-cell problem. The algorithm exists in two parts. First, a calculation of the initial cell locations of the droplets is required. To solve this, a quadtree search object is created of the cell centroids (see Fig. 5.4 for a visualization). The basic premise at work in a quadtree is to subdivide a 2D spatial domain into four rectangular quadrants (i.e., northeast, northwest, southeast, and southwest). This division is then recursively iterated upon each of the subdomains until no subdivision contains more than a prespecified number of points (in our setting, grid cell centroids). Given a query point, the nearest cell centroid can be calculated by recursively searching the tree of subdivisions, and then performing a nearest-neighbor calculation once there are no subdivisions left. This tree is searched for all particles in the mesh, providing initial cell locations.

Once the droplets are advected, new cell locations must be computed, and this is where the second component of the tracking algorithm comes into play. Although the quadtree could in theory be iteratively re-searched each iteration, this strategy is actually prone to error, particularly near the airfoil surface. The fundamental reason for this is that the cells near the boundary wall are extremely anisotropic in aspect ratio (i.e., the width-to-height ratio can be 100 or more). Because quadtree spatial divisions occur on rectangles of aspect ratio close to one, this geometric phenomenon can lead to incorrect computations of cell location.

The solution we have implemented in CATFISH takes advantage of precomputed grid metrics and small time-steps to search only those cells that neighbor the previous cell location. First, each time-step is chosen such that it is not possible for any particle to move a distance which would bring it outside of the neighborhood formed by the eight cells that border the current cell location. This means that a nine-cell search is required for each particle. In order to ensure that the cell anisotropy does not lead to incorrect calculations, the nine-cell neighborhood is scaled by the local Jacobian. This mapping transforms the center cell exactly into a square. The neighboring eight cells are not in

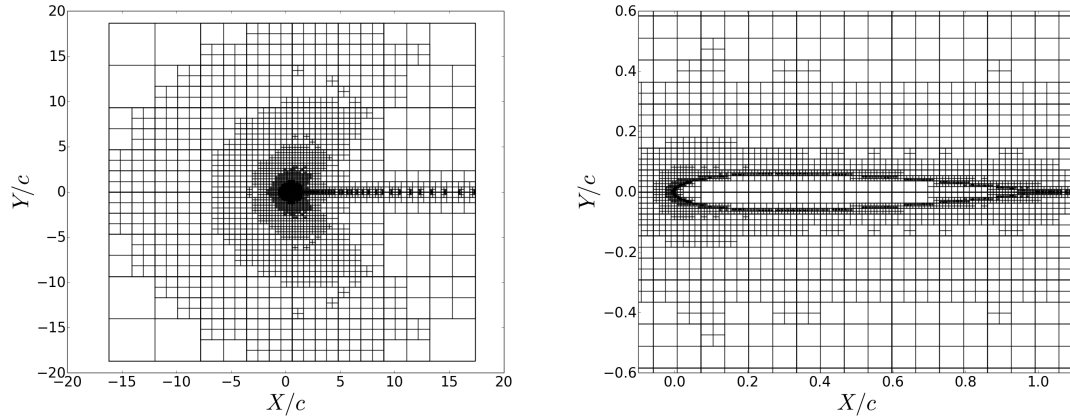


Figure 5.4: Quadtree illustration. *LEFT*: farfield view. *RIGHT*: close-up view of airfoil.

general transformed into exact squares, but instead are general quadrangles. However, in practice, this does not seem to affect the accuracy of the calculations.

5.3.3 Impingement Modeling: Bouncing, Deposition, and Splashing

Once droplets impinge upon the airfoil surface, several modes of behavior have been empirically observed [96, 32, 102] to occur. As noted previously, these include bouncing, depositing into a thin film, or splashing. CATFISH uses empirical correlations from the literature [96, 32, 94] to first predict which of the mechanisms will occur for each impacting droplet and then simulate its effects. These empirical models are briefly reviewed here.

The first step in determining which of the regimes a particular impinging droplet will experience involves computing relevant non-dimensional parameters. The Weber number, which measures the strength of the droplet's inertia relative to surface tension forces, the Reynolds number, which measures the strength of the droplet's inertia relative to viscous forces, and the Ohnesorge, which measures the strength of viscous forces relative to inertial

and surface tension forces, are all important and can be written as:

$$\begin{aligned} Re &= \frac{\rho_l v_n (2r_d)}{\mu_l} \\ We &= \frac{\rho_l (2r_d) v_n^2}{\sigma_l} \\ Oh &= \frac{\sqrt{We}}{Re} \end{aligned} \quad (5.6)$$

where v_n denotes the normal velocity of the impacting droplet, r_d is the droplet radius, and ρ_l , μ_l , σ_l are the liquid density, viscosity, and surface tension.

Using these non-dimensional quantities, we have the following empirical model for droplet impact:

$$\begin{aligned} K &\leq K_{b0} f_b(R, \delta) \quad \mapsto \quad \text{Bounce} \\ K_{b0} f_b(R, \delta) &\leq K \leq K_{s0} f_s(R, \delta) \quad \mapsto \quad \text{Deposition} \\ K &> K_{s0} f_s(R, \delta) \quad \mapsto \quad \text{Splash} \end{aligned} \quad (5.7)$$

In these equations, $K = We \times Oh^{-2/5}$ is the Cossali number. K_{b0} and K_{s0} are threshold values, $R = \frac{h_r}{2r_d}$ is the dimensionless wall roughness height (h_r is the dimensional roughness height), and $\delta = \frac{h_f}{2r_d}$ is the dimensionless film thickness (h_f is the dimensional film thickness). The f_b , f_s are correction factors to account for the effects of wall roughness and film thickness, and they may be represented as:

$$\begin{aligned} f_s(R, \delta) &= \frac{1 + \tilde{R}^2}{1 + \omega_{sr} \tilde{R}^2} \frac{1 + \delta^2}{1 + \omega_{sf} \delta^2} \\ f_b(R, \delta) &= \frac{1 + \tilde{R}^2}{(1 + \omega_{sr} \tilde{R}^2)(1 + \omega_{br} \tilde{R}^4)} \frac{1 + \delta^2}{1 + \omega_{bf} \delta^2} \end{aligned} \quad (5.8)$$

where $\tilde{R} = \frac{R^2}{R + \delta}$ is a wall roughness, adjusted to account for the presence of a liquid film. Eq. 5.8 also has several empirical constants. The values of these constants used in CAT-FISh are: $K_{s0} = 3000$, $K_{b0} = 600$, $\omega_{sr} = 20/3$, $\omega_{sf} = 5/6$, $\omega_{br} = 32$, and $\omega_{bf} = 1$.

Once the proper impingement regime has been calculated for a particular droplet, mass loss via ejection must be calculated. This is trivial for the deposition (mass loss equals zero) and bouncing (all of the mass is ejected) regimes, but for the splashing regime, empirical models must be implemented. The mass loss due to splashing is calculated using the following law:

$$\begin{aligned}\frac{m_s}{m_0} &= \max \left(a(\theta) - \frac{f_s(R, \delta) K_{s0}}{K} b(\theta), 0 \right) \\ a(\theta) &= 1 - 0.3 \sin(\theta) \\ b(\theta) &= \frac{1}{8} (1 + 3 \cos(\theta))\end{aligned}\tag{5.9}$$

where m_s and m_0 are the ejected mass and the initial mass (respectively), and θ is the angle of incidence between the impinging droplet velocity vector and the tangent vector at the wall surface.

The splashing process creates child droplets, which are ejected into the freestream and continue to advect downstream with the local velocity field. Models exist in the literature [96, 81, 64] which describe in detail statistical models for ejection droplet size and velocity. However, after experimenting numerically with some of these models, we have found that splashing tends to create many particles, which slows the collection efficiency computation appreciably. Additionally, most of the ejected droplets are swept aft of the airfoil entirely by the external flow anyway, making tracking them unnecessary. Therefore, CATFISH does not presently track ejected splash particles (i.e., it assumes that all droplets ejected through splashing do not re-impinge on the airfoil).

5.4 Thermodynamics Module

The previous subsection dealt with the droplet simulation module, the main purpose of which is the calculation of a water flux distribution over the surface of the airfoil. The next step in the icing code is to use this information to calculate a distribution of ice growth over

the surface of the airfoil. This subsection describes in detail the methodology implemented in CATFISH for accomplishing this. The approach described here is modeled after the work done by Messinger [61], Habashi [10], and NASA Glenn [100].

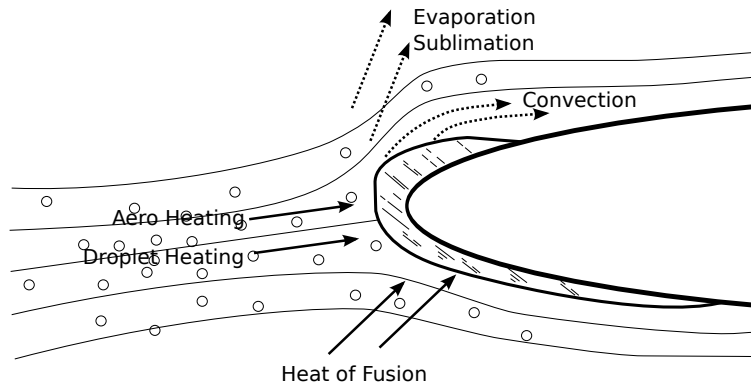


Figure 5.5: Illustration of heat transfer mechanisms in airfoil icing.

Fig. 5.5 illustrates the various dominant heat transfer mechanisms that drive the ice accretion process. Before discussing those mechanisms in detail, it is important start the discussion with a precise account of into/from what, exactly, heat is being transferred. The answer to this question depends on the nature of the ice accretion. If the icing conditions are such that rime accretion exists (i.e., where all impinging water immediately freezes upon impact), then all heat involved is transferred to/from the ice surface, which has no liquid film on top of it. If, instead, conditions are such that glaze accretion occurs, then there exists a thin (i.e., $20\ \mu\text{m}$ or less) film of water on top of the ice surface, and heat is transferred in/out of this film.

These heat transfer phenomena are accompanied by mass transfer mechanisms. We are already aware of the obvious transfer of mass that occurs by means of the droplets from the free-stream impinging on the airfoil surface. In rime accretions, this is the only mass transfer mechanism that is possible. In glaze accretions, the thin film of surface water can advect aft along the surface of the wing, driven by the local shear stress at the surface, providing a means for redistributing the incoming water.

5.4.1 Mass/Energy Conservation

Taking all of this into consideration, it is possible to write mass/energy conservation laws for the film on the surface:

$$\begin{aligned}\rho_l \left(\frac{\partial h_f}{\partial t} + \frac{\partial}{\partial s}(\bar{u}_f h_f) \right) &= \dot{m}_{imp} - \dot{m}_{evap} - \dot{m}_{ice} \\ \rho_l \left(\frac{\partial h_f c_l T}{\partial t} + \frac{\partial}{\partial s}(\bar{u}_f h_f c_l T) \right) &= \dot{e}_{imp} - \dot{e}_{evap} + \dot{e}_{ice} + \dot{e}_{conv}\end{aligned}\tag{5.10}$$

The right-hand-side of these equations represent the sources and sinks for mass (first equation) and energy (second equation) that we have just discussed. Specifically, \dot{m}_{imp} , \dot{m}_{evap} , and \dot{m}_{ice} respectively denote mass sources/sinks to account to impingement, evaporation/sublimation, and ice accretion; \dot{e}_{imp} , \dot{e}_{evap} , \dot{e}_{ice} , and \dot{e}_{conv} denote energy sources/sinks that are generated by impingement, evaporation/sublimation, ice accretion, and convection/kinetic heating. The left-hand-side represents the convective derivative of film mass ($\rho_l h_f$) and energy ($\rho_l h_f c_l T$), which advect with the film speed \bar{u}_f . h_f is the film height, c_l is the specific heat transfer coefficient of water, T is the temperature of the film/ice/airfoil surface, and s is an arc-length coordinate (with the origin at the stagnation point).

The expressions for sources/sinks in Eq. 5.10 read thus:

$$\begin{aligned}\dot{m}_{imp} &= U_\infty \text{LWC} \beta \\ \dot{m}_{evap} &= \frac{0.7 c_h}{c_{air}} \left(\frac{P_{v,p} - H_{r,\infty} P_{v,\infty}}{P_s} \right) \\ \dot{e}_{imp} &= \left(c_l T_\infty + \frac{\|u_d\|^2}{2} \right) \dot{m}_{imp} \\ \dot{e}_{evap} &= \frac{1}{2} (L_{evap} + L_{sub}) \dot{m}_{evap} \\ \dot{e}_{ice} &= (L_{fus} - c_{ice} T) \dot{m}_{ice} \\ \dot{e}_{conv} &= c_h (T_{rec} - T)\end{aligned}\tag{5.11}$$

In these equations, c_h is the convective heat transfer coefficient, c_{air} is the specific heat capacity of air, $P_{v,p}$ is the saturation vapor pressure at the airfoil surface, $P_{v,\infty}$ is the sat-

uration vapor pressure of the ambient air, $H_{r,\infty}$ is the relative humidity, P_s is the static pressure at the boundary layer edge, T_∞ is the free-stream temperature, u_d is the droplet velocity, L_{evap} , L_{sub} and L_{fus} are the latent heats of evaporation, sublimation and fusion, c_{ice} is the specific heat capacity of ice, and T_{rec} is the recovery temperature.

We can also determine the film speed, \bar{u}_f , in Eq. 5.10 in terms of known flow quantities. Because the film is thin, we can assume it to be Newtonian, i.e. $\tau = \mu_l \frac{du_f}{dy}$, where τ is the wall shear stress and y is the coordinate in the direction normal to the wall. Imposing zero velocity at the wall as a boundary condition immediately yields:

$$u_f = \frac{y}{\mu_l} \tau \quad (5.12)$$

Averaging this equation across the film thickness gives:

$$\bar{u}_f = \frac{1}{h_f} \int_0^{h_f} u_f(y) dy = \frac{h_f}{2\mu_l} \tau \quad (5.13)$$

5.4.2 Closure Constraints

At this point, we have two equations, and three unknowns (h_f , T , and \dot{m}_{ice}); the system is underdetermined. Extra information is needed to close these relations, and is provided by the following constraints:

$$\begin{aligned} h_f &\geq 0 \\ \dot{m}_{ice} &\geq 0 \\ h_f T &\geq 0 \\ \dot{m}_{ice} T &\leq 0 \end{aligned} \quad (5.14)$$

These compatibility relations are physically intuitive: neither the film height nor ice growth rate can be negative; if there is a water film, it cannot be below freezing in equilibrium; if there is ice growth, the ice cannot be above freezing in equilibrium.

5.4.3 Convective Heat Transfer Computation

Before proceeding any further, a brief digression is in order regarding how we choose to calculate the convective heat transfer, \dot{e}_{conv} . This is the most important term in the energy equation, Eq. 5.10, which can be seen by a simple order of magnitude estimate. The dominance of \dot{e}_{conv} explains why many ice shapes (i.e., horns) strongly resemble the underlying distribution of convective heat transfer across the airfoil surface.

The convective heat transfer is a subgrid scale phenomenon, in that it depends heavily on the nature of the boundary layer (laminar or turbulent) as well as turbulent mixing. The latter of these phenomena is physically dominated by the presence of small-scale roughness elements on the surface of the ice, which act as vortex generators that enhance Reynolds stresses (see the review of ice roughness in Chapter 3). It is here that we encounter a dilemma which is familiar to the computational fluids community: one of the most important physical mechanisms in our problem (that is, surface roughness and boundary layer turbulence) is also one of the most challenging to model. We obviously cannot compute these mechanisms directly, because they exist at subgrid scales, and so we resort to approximating them with subgrid models.

The model implemented in FLO103 is the rough-wall extension to the Spalart-Allmaras turbulence model [86, 8, 11]. This model consists of the one-equation turbulence model described by the classic Spalart-Allmaras model, plus a term that accounts for surface roughness. The main aerodynamic effect of this parameter is to increase eddy viscosity and promote mixing (thereby increasing convective heat transfer). The Spalart-Allmaras model computes the turbulent viscosity $\tilde{\nu}$ and reads thus:

$$\begin{aligned} \frac{\partial \tilde{\nu}}{\partial t} + u_j \frac{\partial \tilde{\nu}}{\partial x_j} = & \frac{1}{\sigma \text{Re}_\infty} \left[\frac{\partial}{\partial x_k} (\nu + \tilde{\nu}) \frac{\partial \tilde{\nu}}{\partial x_k} + c_{b2} \frac{\partial \tilde{\nu}}{\partial x_k} \frac{\partial \tilde{\nu}}{\partial x_k} \right] + c_{b1} (1 - f_{t2}) \tilde{S} \tilde{\nu} \\ & - \left[c_{w1} f_w - \frac{c_{b1}}{\kappa^2} f_{t2} \right] + \frac{1}{\text{Re}_\infty} \left(\frac{\tilde{\nu}}{d} \right)^2 + \text{Re}_\infty (f_{t1} \Delta U^2). \end{aligned} \quad (5.15)$$

Here, d is the distance from the wall, \tilde{S} is the modified vorticity, c_{b1} , c_{b2} , c_{w1} , κ , and σ are closure coefficients, f_w is a closure function, and f_{t1} and f_{t2} are functions that allow the user to set the location of the laminar/turbulent transitioning.

The rough wall extension modifies the wall distance d based on a Nikuradse sand grain roughness k_s that is presumed known to the user:

$$d = d_{min} + 0.03k_s \quad (5.16)$$

This modification changes the boundary condition for the turbulent viscosity from Dirichlet ($\tilde{\nu}_{wall} = 0$) to Neumann ($(\partial\tilde{\nu}/\partial n)_{wall} = \tilde{\nu}_{wall}/d$).

The convective heat transfer \dot{e}_{conv} is obtained by solving the RANS equations, with the turbulent eddy viscosity computed using the model just described. An open question involved in doing this is, what boundary condition should be used for the energy equation? FLO103 was originally implemented using adiabatic boundary conditions, which obviously cannot be correct for icing applications since, by definition, such a condition guarantees zero wall heat flux. The consensus of the icing community seems to be that imposing a constant clean airfoil temperature above the freestream temperature is appropriate [10, 92]. In CATFISH/FLO103, we set this (Dirichlet) boundary condition on the wall temperature to the freezing temperature (273.15 K).

5.4.4 Solution Method

CATFISH discretizes Eq. 5.10 with a finite volume method. Explicitly, the conversion of Eq. 5.10 to a finite volume scheme is achieved by integrating over the spatial domain, and

then applying the divergence theorem:

$$\begin{aligned}
\frac{\partial U}{\partial t} + \frac{\partial F}{\partial s} &= \dot{S} \\
\int_{\mathcal{B}_k} \left(\frac{\partial U}{\partial t} + \frac{\partial F}{\partial s} \right) ds &= \int_{\mathcal{B}_k} \dot{S} ds \\
\int_{\mathcal{B}_k} \frac{\partial U}{\partial t} ds + (F_{k+1/2} - F_{k-1/2}) &= \int_{\mathcal{B}_k} \dot{S} ds \\
\frac{\partial \bar{U}_k}{\partial t} &= \underbrace{\frac{1}{\Delta s_k} \int_{\mathcal{B}_k} \dot{S} ds}_{\delta_u} - \Delta F_k \\
\bar{U}_k^{t+\Delta t} &= \bar{U}_k^t - \Delta t \delta_u
\end{aligned} \tag{5.17}$$

Here, we are using general notation, where U represents the relevant flow variable, $F_k = (\bar{u}_f U)_k$ is the (body-centered) flux in cell k , \dot{S} is the sum of all sources/sinks, \mathcal{B}_k is a control volume cell, Δs_k is the length of cell k , and \bar{U}_k is the cell average of the flow variable. Fig. 5.6 displays the general schematic at play here.

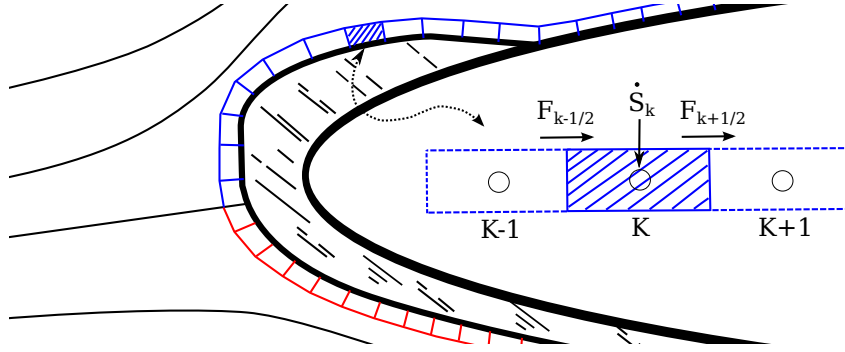


Figure 5.6: Illustration of the finite volume method. The stagnation line divides the upper surface cells (blue) from the lower surface cells (red).

Note that the film speed (\bar{u}_f) may reverse sign (as is the case if there are separation bubbles aft of a horn); this can lead to discontinuities in the solution, and necessitates some form of upwinding. Therefore, a Roe scheme [89, 78] is used to calculate the fluxes, which gives:

$$F_{k+1/2} = \frac{1}{2}(F_k + F_{k+1}) - \frac{1}{2}|\bar{u}_{f_{k+1/2}}|(U_{k+1} - U_k) \tag{5.18}$$

It should be noted - as shown in Fig. 5.6 - that the airfoil surface is divided into two sections (upper and lower surface), based on where the stagnation point is. This is because the boundary conditions for the thermodynamic variables are applied at the stagnation point ($h_f(s = 0) = 0$, $T(s = 0) = 0^\circ C$). Hence, Eq. 5.10 is solved independently on the upper and lower surfaces.

CATFISH makes a quasi-steady state assumption in solving these equations. That is, Eq. 5.10 is driven to steady-state ($\partial U / \partial t = 0$), and the resulting solution for the ice accretion rate (\dot{m}_{ice}) is used to calculate an incremental addition of ice, corresponding to a short-time march of Δt . Then, the entire grid is re-meshed, a new flowfield is solved, Eq. 5.10 is once again driven to steady-state, a new ice accretion rate is computed, and another layer of ice is added. This process is iterated upon in order to produce time-dependent ice growth, and is displayed in Alg. 3.

5.4.5 Ice Addition

The last topic that must be addressed involves how the ice is added to the airfoil surface. A naive approach is to simply add the appropriate amount for a given point in the direction of the local normal vector. However, there are two distinct problems with this method. First, the ice shape produced in this manner will actually be more massive than it should be, owing to the curvature of the underlying surface. A depiction of this phenomenon is shown in Fig. 5.7. Second, adding ice without first smoothing the profile in some fashion will result in shapes that are unrealistically jagged.

The problem of area obliteration has been observed in the NASA Glenn icing code LEWICE, is rectified using a simple geometric algorithm [101]: the areas of the red triangles in Fig. 5.7 are calculated, and a new local ice height is computed such that the updated ice area is equal to the difference between the blue rectangle and the red triangle. This scheme starts on the underside of the airfoil and proceeds in a marching fashion to

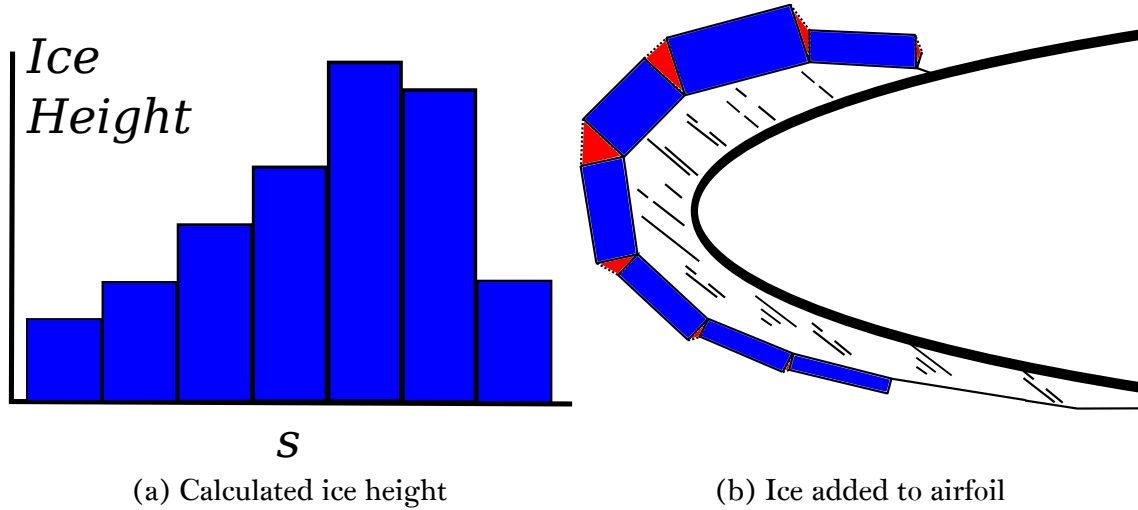


Figure 5.7: Illustration of the area oblation effect. When the calculated ice height (blue rectangles) is simply added along local normal vectors, extra mass (red triangles) is unintentionally added due to the curvature of the underlying airfoil/ice surface.

the upperside. Our numerical experiments with this method confirm that it provides a good correction for the area oblation effect.

We solve the related problem of ice jaggedness by passing the ice shape through a Laplacian smoothing function. That is, we solve the discretized version of the following operator for the smoothed coordinates:

$$\tilde{y} - \epsilon \nabla^2 \tilde{y} = y \quad (5.19)$$

where y and \tilde{y} are, respectively, the non-smoothed and smoothed coordinates, and ϵ is a user-defined smoothing parameter.

5.5 Validations

In this section, we present several studies that benchmark CATFISH and its calculations against published computational and experimental data.

Algorithm 3: CATFISh Thermodynamics Solver

Input: Flowfield and droplet calculations

Output: Ice growth rate \dot{m}_{ice} , film height h_f , surface temperature T

(1) *Initialization:* select initial guesses for h_f , T , and \dot{m}_{ice}

while $\partial U / \partial t > (\text{convergence tolerance})$ **do**

 (2) *Iterate:* forward step mass/energy equations for h_f and T

 (3) *Constraints:* apply constraints to h_f and T

 (4) *Ice rate:* solve energy equation for \dot{m}_{ice}

end

5.5.1 Droplet Simulation

The first matter which should be discussed regarding benchmarking droplet simulations is that there are two possible types of calculations we could perform - monodispersed, or polydispersed. A monodispersed simulation, as the name implies, is one in which all droplets in the freestream have the same diameter. Conversely, a polydispersed simulation is one in which the sizes of droplets advected obey some statistical distribution.

The relative tradeoffs to consider in using one or the other are exactly what one would expect - computational efficiency versus precision. Atmospheric studies [75] and current FAA safety regulations [6] agree that, in reality, cloud droplets are distributed rather than monodispersed (although different distributions are possible). The typical approach in a polydispersed simulation is to perform several independent monodispersed simulations, each drawn from the underlying droplet distribution, and then weight average them appropriately. Polydispersed droplet simulations in airfoil icing typically make use of either 10-bin or 27-bin distributions (i.e., either 10 or 27 weight-averaged monodispersed simulations). A monodispersed approach would essentially replace the droplet distribution with a delta function at the mean volumetric diameter (MVD) of the distribution. The polydispersed approach is, of course, the more realistic method, but a 27-bin polydispersed simulation requires 27 times as much work as a monodispersed simulation; this is the computational tradeoff that must be considered (especially in our UQ setting, where we must perform many icing simulations corresponding to different conditions).

Comparisons of droplet simulations and collection efficiency calculations for a number of droplet sizes and distributions are shown in Fig. 5.8. This figure presents droplet calculations for diameters equal to 20, 111, and 236 μm . The top row uses a NACA 0012 airfoil, with free-stream conditions of $Re_\infty = 4.14 \times 10^6$ and $M_\infty = 0.32$ at a 2.5° angle of attack, and the “experiment” against which CATFISH is benchmarked is numerical computation from LEWICE [10]. The bottom two rows of the figure use a NACA 23012 airfoil, with free-stream conditions of $Re_\infty = 4.6 \times 10^6$ and $M = 0.23$ at a 4° angle of attack, and the experiments against which CATFISH is benchmarked were generated at the NASA Glenn Icing Research Tunnel [75]. In all three cases, CATFISH was run using both 27-bin distributions (see Appendix B) and a single monodispersed run.

A few comments are in order regarding 5.8. First, the agreement is excellent between experiment, monodispersed, and polydispersed results for small droplet size (MVD = $20\mu\text{m}$), while there is slight disagreement between the three for larger droplet sizes (MVD = $111\mu\text{m}$ and MVD = $236\mu\text{m}$). This is most likely related to the fact that smaller droplets have less mass and hence smaller Reynolds/Cossali numbers (as defined previously in Equations 5.6 and 5.7). As a consequence of this, most of those either deposit directly onto the airfoil or simply bounce off; there is relatively little splashing physics present in the case of smaller droplets. In contrast, the larger droplets are more likely to have a high enough Cossali number that splashing will occur, which makes the modeling more challenging.

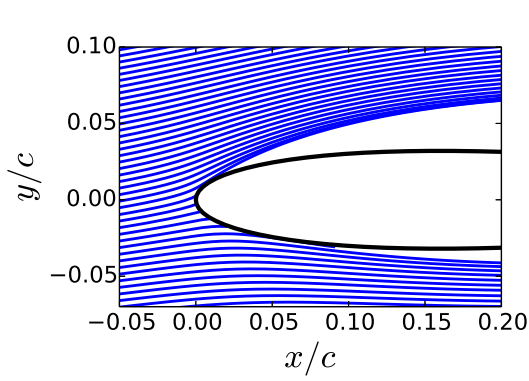
Second, the disagreement between the poly/mono-dispersed results does not seem significant enough in any of the droplet sizes to warrant the added computational burden involved with running polydispersed results. Again, it must be stressed that this added expense is quite significant in a UQ context, where new polydispersed calculations would be required each timestep of each simulation.

5.5.2 Icing Calculations

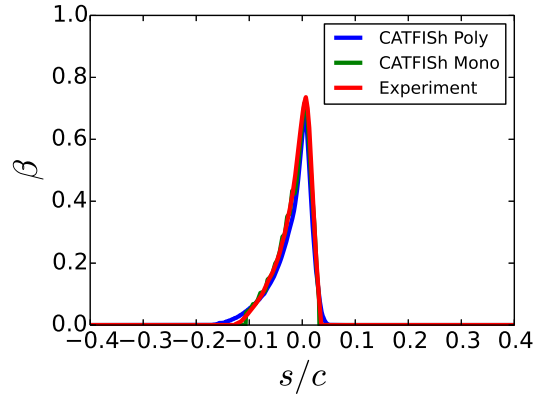
Here, we compare the ice accretion calculations of CATFISH to both experimental and numerical (LEWICE) results [103]. Specifically, we examine the ice shapes that grow on a NACA 0012 airfoil under 20 different sets of icing conditions. The test matrix detailing those conditions is given in Appendix C. Fig. 5.9 displays the final accretion results for these tests.

Examining the results in Fig. 5.9, we see qualitatively mixed results, ranging from excellent agreement between CATFISH and experiments (e.g., runs 425, 405, 424, 429) to poor agreement between CATFISH and experiments (e.g., runs 206, 427). However, it is important to keep a few things in mind when assessing these results. First, agreement between experimental and computational shapes for glaze horns is notoriously difficult to achieve, as noted by several researchers [10, 47, 93]. This is partially attributable to the modeling assumptions and simplifications made on the computational side (e.g., not fully resolving the micro-scale surface roughness physics), and partially attributable to a lack of ice shape repeatability on the experimental side. Second, most of the results demonstrate that the CATFISH is, at minimum, able to identify the correct regime of icing: if the experimental shape is a glaze horn, for example, then CATFISH generally predicts a horn, notwithstanding particular differences in height, angle, and shape details. Most of the test cases for which there is poor agreement in Fig. 5.9 fall into this category – they are horn accretions for which CATFISH predicts a different horn angle.

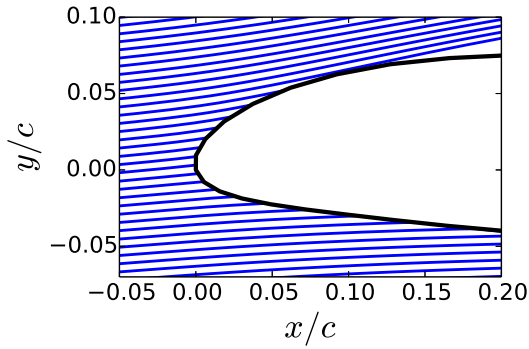
Example rime/glaze solutions are shown in Fig. 5.10. These solutions are initialized with zero for each variable, and driven to steady-state as described in Alg. 3. As can be seen, the temperature profile converges relatively slowly compared to the film and ice profiles. The final solutions are well-converged and appear to satisfy all physical constraints.



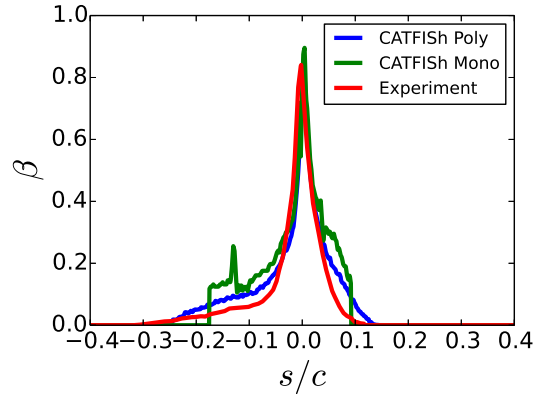
(a) Droplet advection trajectories corresponding to droplet diameter $20\ \mu\text{m}$.



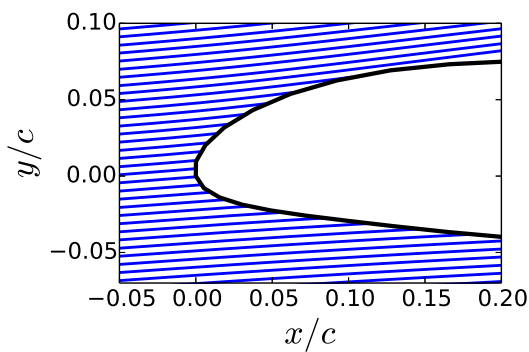
(b) Collection efficiency; MVD = $20\ \mu\text{m}$.



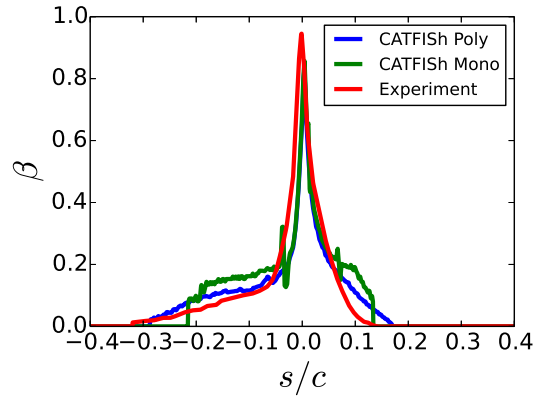
(c) Droplet advection trajectories corresponding to droplet diameter $111\ \mu\text{m}$.



(d) Collection efficiency; MVD = $111\ \mu\text{m}$.



(e) Droplet advection trajectories corresponding to droplet diameter $236\ \mu\text{m}$.



(f) Collection efficiency; MVD = $236\ \mu\text{m}$.

Figure 5.8: Droplet trajectories and collection efficiencies for various droplet sizes. On the right, monodispersed solutions are shown in green; polydispersed solutions are shown in blue.

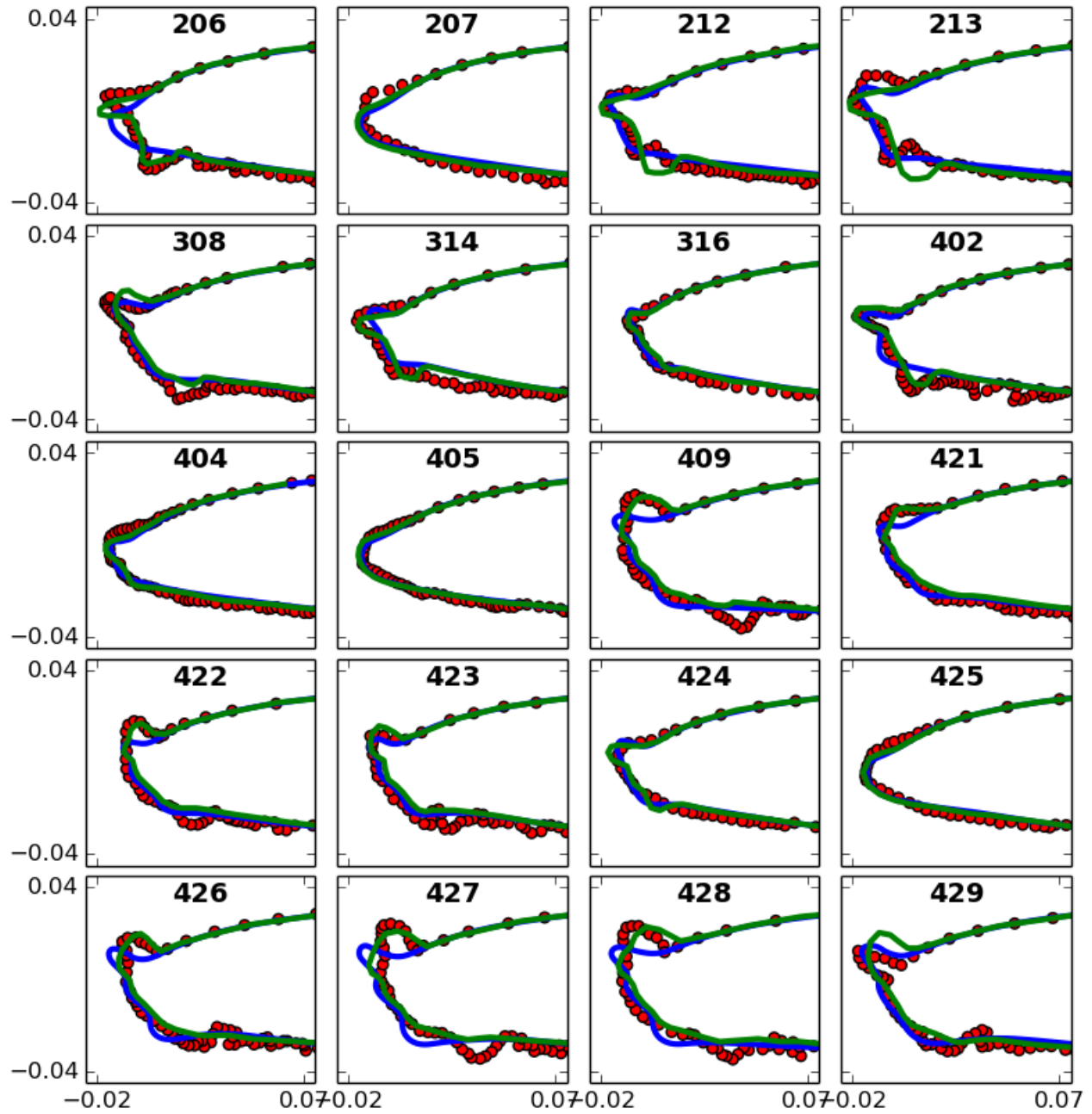
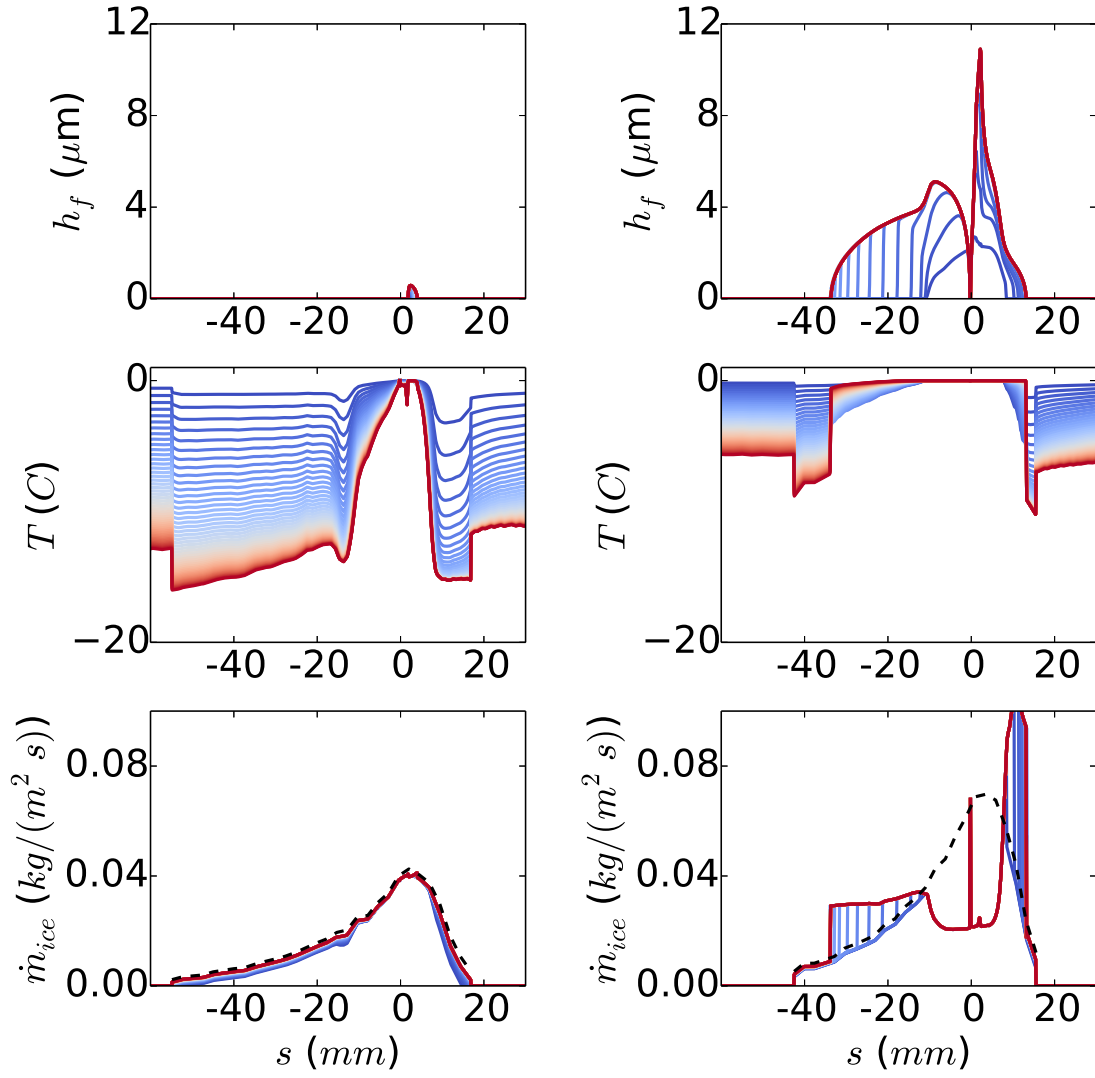


Figure 5.9: Benchmark test cases for CATFISH (titled with corresponding experiment number). Experiments are shown as red dots, LEWICE computations are shown in green, and CATFISH computations are shown in blue.



(a) Rime accretion case (run 405).

(b) Glaze accretion case (run 308).

Figure 5.10: Illustration of driving the thermodynamics to steady-state.

5.6 UQ Results

Having validated CATFISh against pre-existing results from the literature, we now turn our attention to the ultimate goal of this chapter - the quantification of uncertainty in aerodynamic performance due to uncertainty in physical parameters.

To begin, we must first identify the important governing parameters in the icing process. Toward that end, we note that there are aerodynamic parameters and thermodynamic parameters that both interact to determine the ice shape. The aerodynamic parameters that are especially relevant include the free-stream velocity, Mach and Reynolds numbers, and angle-of-attack. Important thermodynamic parameters include the free-stream temperature, liquid water content, droplet diameter, accretion time, and the roughness parameter (in the Spalart-Allmaras turbulence model). In terms of UQ methodology, we choose to use sparse grid polynomial chaos to build surrogate models globally over a given parameter space, and then analyze the resulting statistics of the surrogate for useful trends and information.

Each icing simulation is fairly expensive. Wall clock times of 2-3 hours can sometimes be necessary for 5 minute simulations (with flow/ice restarts every 1 minute). To make the UQ manageable, we run all simulations on our own in-house server, which has 128 processors. Job submission to these processors is handled by TORQUE PBS [2]. The individual subtasks of each icing simulation (e.g., interfacing with DAKOTA input/output, running FLO103/CATFISh executables) are handled by custom bash shell scripts.

The UQ studies that follow have, at largest, a parameter space dimension of three. Of course, there are more than three governing parameters in ice accretion; however, we did not attempt any studies of higher dimension. The reasoning for this was practical: such high dimensional spaces will require many simulations (potentially thousands or more), and every now and then human intervention is required (for example, grid/flow/icing solvers might fail, or the server might go down, either from a power loss or unintended

consequences from software updates). This can result in total UQ runtimes that are significantly longer than one might expect from a “back of the envelope” calculation. Hence, we limit ourselves to, at most, three parameters. With more processors, improvements to the robustness of our grid/flow/icing solvers, and improvements to the “failsafe” robustness of the server itself, higher dimensional parameter space UQ studies could certainly be possible.

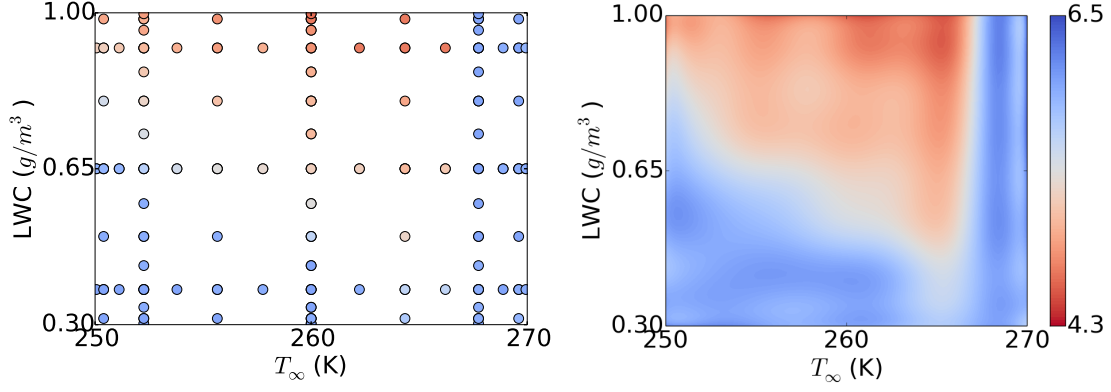
As free-stream temperature and LWC are two dominant parameters in ice accretion, we proceed to study the effects these parameters have, either alone or in conjunction with one other parameter.

5.6.1 Two Parameters: T_∞ and LWC

We begin with a 2-D parameter space consisting of the free-stream temperature (T_∞) and liquid water content (LWC) of the air. These two parameters are arguably the most important thermodynamic parameters, as they play a critical role in determining whether the accretion type is rime or glaze [93, 37]. As output, we choose to examine the coefficient of lift (C_{L_α}) at a particular angle-of-attack. This metric is important from the perspective of aircraft safety, because it provides a local description of exactly how much control a pilot has in recovering airplane lift in response to lift deterioration caused by ice accretion. We select bounds for the variables based on the conditions used in the experimental databases generated by NASA Glenn [103] - specifically, we use $T_\infty \in [250, 270]$ K and $\text{LWC} \in [0.3, 1.0]$ g/m³. We equip each of these variables with uniform distributions, a choice which corresponds to no prior statistical knowledge about which sections of parameter space are more likely (this can be intuitively interpreted as a “worst-case” statistical analysis, since the introduction of statistical knowledge through a weighted distribution would likely have the effect of lowering the output variance). The PCE basis polynomials corresponding to our choice of distribution are the Legendre polynomials.

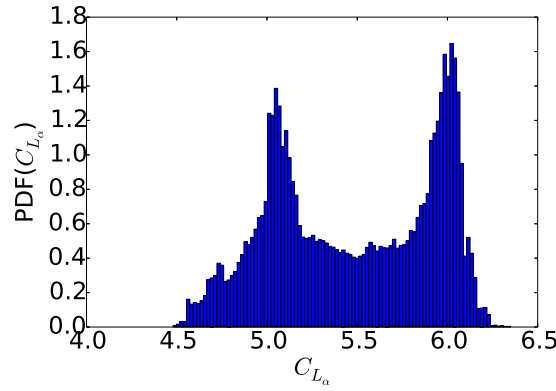
The other governing aero/thermo-dynamic parameters are held constant for this study. Again, they are chosen to be consistent with the ranges of values explored by NASA Glenn. A precise description of these parameters is as follows: $Re_\infty = 4.14e6$, $M_\infty = 0.32$, $U_\infty = 102.8$, $\alpha = 4^\circ$, $MVD = 20 \mu m$, and accretion time = 5 min. The roughness parameter (i.e., average wall roughness element height) in the Spalart-Allmaras turbulence model is set to 0.55 *mm*. The airfoil profile chosen is a NACA 0012, with a chord of 0.5334 *m*.

Fig. 5.11 shows the quadrature points used in this study, the PCE surrogate computed from these quadrature points (colored according to the lift slope value), and the statistics (computed by Monte Carlo sampling of the surrogate). We observe a sharp division between two regions in parameter space, which can be explained in terms of rime/glaze accretion. Rime accretions, by definition, occur when there is enough heat transfer to remove all (or almost all) of the latent heat of freezing needed to freeze all (or almost all) of the incoming water mass over the entire surface of the airfoil. As would be expected, this regime of icing occurs in colder conditions with lower LWC. Conversely, glaze shapes occur when there is not enough heat transfer to completely freeze all incoming water mass, resulting in a redistribution of the incoming water over the surface of the airfoil and the familiar “horn” shapes. This regime tends to occur at higher temperatures and LWC. As has been pointed out in multiple places in this thesis, horns are generally more dangerous aerodynamically than rime accretions, and this explains the trends shown in Fig. 5.11. Fig. 5.12 displays a selection of shapes from the parameter space that are associated with the benign and dangerous regions. It is also worth noting that the construction of a PCE surrogate allows us to quantitatively describe, with high precision, the exact locations in parameter space corresponding to the rime/glaze arenas, as opposed to simply describing the governing trends qualitatively.



(a) Quadrature points (colorscale identical to (b)).

(b) PCE surrogate.



(c) PCE surrogate statistics.

Figure 5.11: Quadrature points, PCE surrogate, and statistics for the 2 parameter (T_∞ and LWC) study on lift slope.

5.6.2 Three Parameters: T_∞ , LWC, and ΔT

We can extend the results of the previous section to include accretion time as a third parameter. In this way, we can quantify the time-dependent degradation of aerodynamic performance. We choose to study a uniformly distributed accretion time: $\Delta T = \mathcal{U}[1, 5]$ minutes. All other governing aero/thermo-dynamic parameters retain the same values as in the previous study.

Figures 5.14a, 5.14c and 5.14d display the quadrature points used in this study, the PCE surrogate computed from these quadrature points (colored according to the lift slope value), and the statistics (computed by Monte Carlo sampling of the surrogate). As ex-

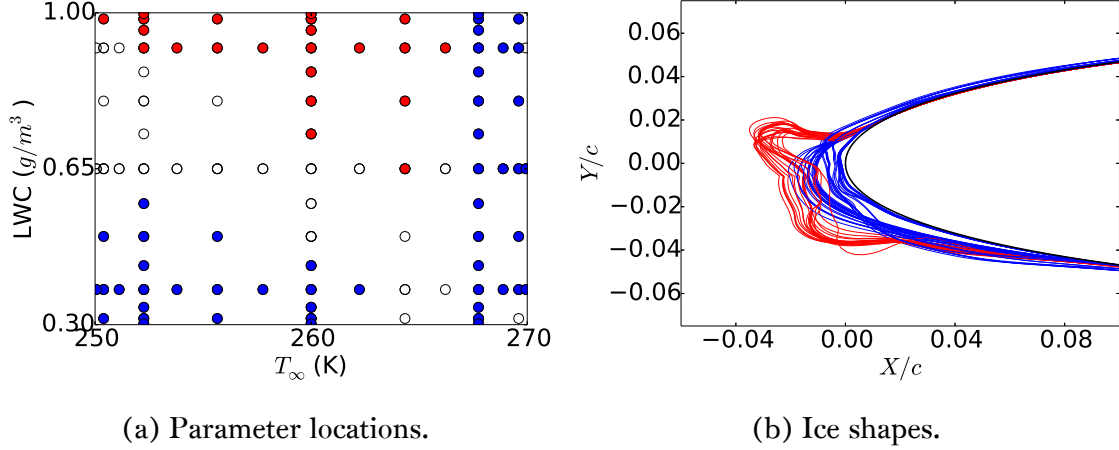


Figure 5.12: Select points in parameter space corresponding to more and less dangerous regimes of icing, as well as the ice shapes they produce.

pected, the deterioration of C_{L_α} is monotonically worse with increasing accretion time (given specific values of T_∞ and LWC). Statistically, we still have a bimodal distribution, corresponding to the rime and glaze regimes; however, this distribution is now skewed toward the peak of higher C_{L_α} . This is because horns require some amount of time to grow before they cause the large scale leading-edge separation bubbles that cause the extreme performance degradation.

One potential use of the PCE surrogate involves estimating conditional expectations of the output. In this particular setting, it may benefit aircraft safety to calculate what the lift slope is as a function of time, with the other parameters (temperature and LWC) averaged out. With PCE, this is straightforward - simply compute the output on a number of constant-time “slices” in parameter space, and average the output on each of these slices. Similarly, we can compute other statistics of each of these slices, such as percentile ranges. Fig. 5.14e displays the conditional expectation and percentile ranges of lift slope; these statistics show how quickly performance deteriorates as a function of time with statistical variations in T_∞ and LWC included. We observe that the performance range for a single value of time increases as time increases, as the differences between benign rime and dangerous horn accretions becomes more and more pronounced. Note that many of the rime

accretions, even larger ones, appear not to degrade the lift slope relative to the clean airfoil value (which is near the thin airfoil theory theoretical value of 2π). This phenomenon has been observed in other studies [17], and is due to the fact that smooth rime accretions tend to act as if they were leading edge extensions.

5.6.3 Three Parameters: T_∞ , LWC, and MVD

Now, we take droplet size into account, and perform a UQ study to see how it interacts with temperature and LWC. As has been noted, droplet size can have a reasonably significant impact on the calculated distribution of water that impinges on the airfoil. One objective of this study is to assess just how important this parameter is relative to T_∞ and LWC, which have already been established as important governing parameters. We choose a uniform distribution for the droplet size: $MVD = \mathcal{U}[10, 100] \mu\text{m}$. All simulations performed are monodispersed. Accretion time is set to 5 min; all other governing parameters are the same as before.

As can be seen in Fig. 5.15a and 5.15c, there is a small but noticeable effect of MVD, as witnessed by the conditional expectation shown in Fig. 5.15e. We observe a sharp average increase in performance as we lower MVD below about $20\mu\text{m}$, as well as a modest average increase in performance at diameters above approximately $90\mu\text{m}$.

These effects can be explained by examining Fig. 5.13. The collection efficiency for $MVD = 10\mu\text{m}$ has a much lower peak than those corresponding to the larger sizes. This results in lower impinging water mass, which results in an ice shape that is closer to rime accretion than a glaze horn. The difference between intermediate and large values of MVD is more subtle. Upon close examination of Fig. 5.13a, 5.13b, 5.13c, it is apparent that the upper surface collection efficiency (relative to the appropriate stagnation point) is slightly greater for the intermediate droplet size than it is for the large droplet size. This is most likely due to splashing losses, which are greater for larger and more massive droplets. The

effect is slight, but it is enough to produce a marginally larger upper-surface horn for the intermediate droplet size.

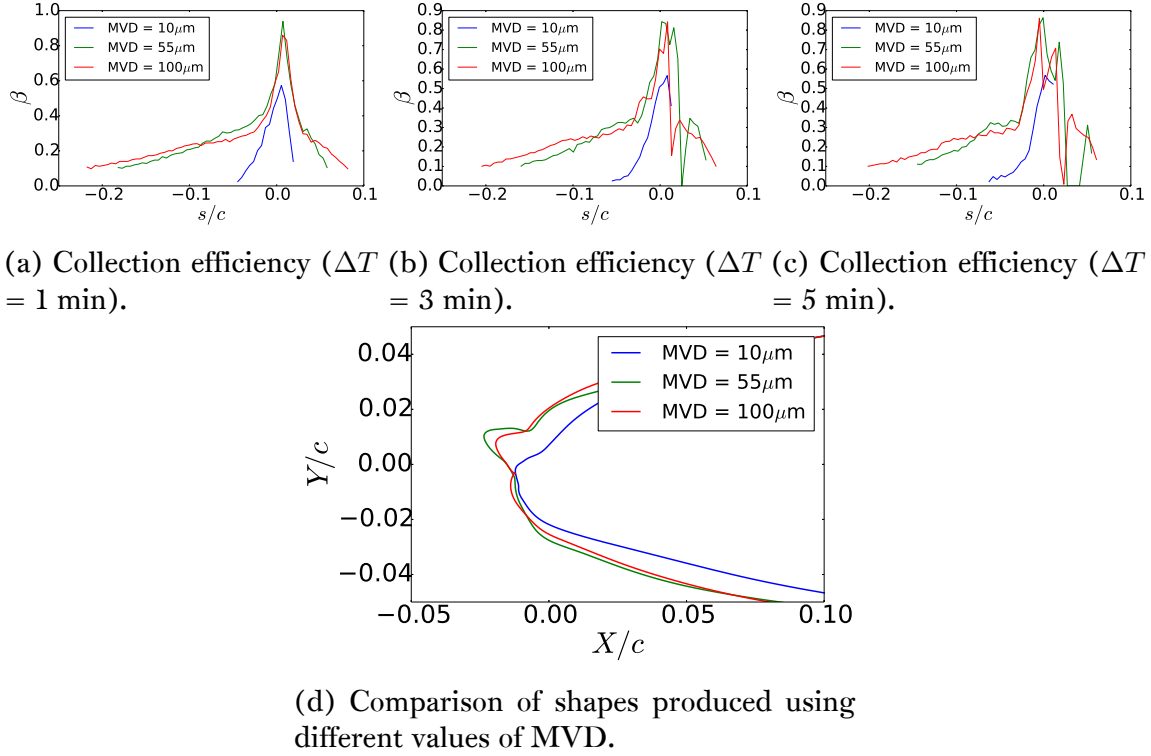


Figure 5.13: Comparison of collection efficiency evolution and ice shapes produced using different droplet sizes.

The overall implications of these observations are mixed. It is clear lowering droplet diameters below $20\mu\text{m}$ is more likely to produce benign ice shapes with less impact on performance; however, performance is relatively unaffected by increasing droplet size all the way to around $90\mu\text{m}$. The modest recovery in performance observed after that point is relatively small, if it exists at all. The importance of MVD relative to temperature and LWC can be quantified by examining the total Sobol indices (see Chapter 2), which provide a measure of the relative fraction of the total variance attributable to each individual parameter. The Sobol indices for T_∞ and LWC are 0.5 and 0.6, respectively, while the Sobol index for MVD is 0.1, meaning that the total influence of MVD is only 10% of the total variance. Assuming we believe that our droplet advection module gives a faithful representation of the true physics, then these findings may be interpreted as evidence that

MVD is a relatively unimportant parameter in icing. However, our model is imperfect, and droplet models are still an area of active research. Additionally, we have used monodispersed (rather than the more physical polydispersed) droplet distributions; it would be interesting to examine whether these findings change in a polydispersed setting.

5.6.4 Three Parameters: T_∞ , LWC, and Roughness

As we have noted, the roughness parameter in the Spalart-Allmaras turbulence model can heavily influence the convective heat transfer, and therefore the ice shape. Research has been done to develop empirical correlations relating the roughness height to aerodynamic/icing parameters; however, the local value of the roughness height can be non-uniform over the airfoil surface. Surface roughness heights have been observed to vary over an order of magnitude in some studies. Given how important of a parameter roughness height is in controlling convective heat transfer, quantifying the variation of the aerodynamics over a range of roughness values and icing conditions would be a useful study.

In this study, we aim to see how important the roughness parameter is over a range of temperature/LWC combinations. All other icing/aerodynamic parameters are kept constant at the values they have assumed in the previous studies.

What stands out most clearly from the statistical trends of Fig. 5.16 is the sharp decline in performance that occurs as k_s is decreased below 0.25mm for some combinations of temperature/LWC. By examining the ice shapes depicted in Fig 5.16b, we can deduce the physical explanation for this occurrence: the horns that give the worst performance are the ones that have the highest inter-horn angle. As we have observed in previous parts of this thesis, inter-horn separation produces horns that are more normal to the free-stream flow, resulting in a larger leading edge separation bubble and hence worse performance. The reason for this effect is the general concept that rougher surfaces promote greater turbulent mixing within the boundary layer than smoother surfaces do; hence, convective heat transfer is less in the area surrounding the stagnation point for smoother surfaces

than for rougher ones. This pushes the horns farther aft, resulting in the increased separation distance. Of course, the areas in parameter space for which the temperature is cold enough and the LWC is low enough that rime accretion exists are mostly unaffected by this phenomenon.

These results confirm our suspicion that roughness can play a pivotal role in determining the computed ice shape. It is therefore possible that existing icing codes would benefit from greater research in the subgrid roughness models.

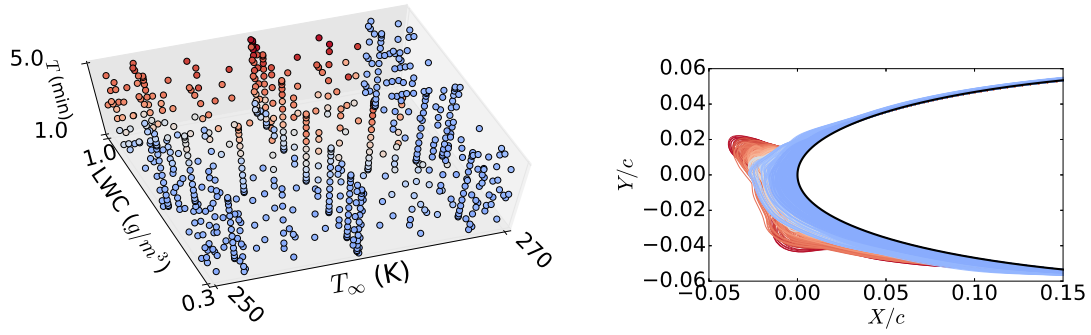
5.7 Summary

The intent of this chapter was to introduce one final perspective on the ice accretion UQ problem. The chapters prior to this one did not extensively investigate the effects of uncertainties in the governing physical parameters that produce airfoil ice shapes. This chapter sought to close that gap by developing computational codes and studying their input/output statistics.

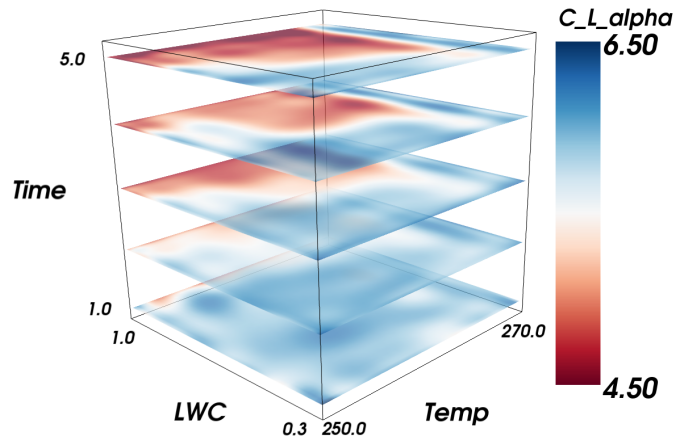
Specifically, we developed from scratch an in-house ice accretion code, which interfaces with our pre-existing, in-house flow solver. This icing code uses a hybrid of techniques from other icing codes, and it contains three modules: droplet advection (which calculates the droplet trajectories and impinging water mass from the free-stream), thermodynamics (which discretizes and solves the governing mass/energy balances to calculate a rate of ice accretion), and ice addition (which applies filtering/smoothing routines to the calculated ice rate and adds ice to the airfoil surface). We then performed a series of UQ experiments to understand and quantify the effects of different icing parameters on aerodynamic performance.

There are still many open avenues for this research in the future. For example, we did not include aerodynamic parameters (such as Reynolds/Mach number) in our uncertain

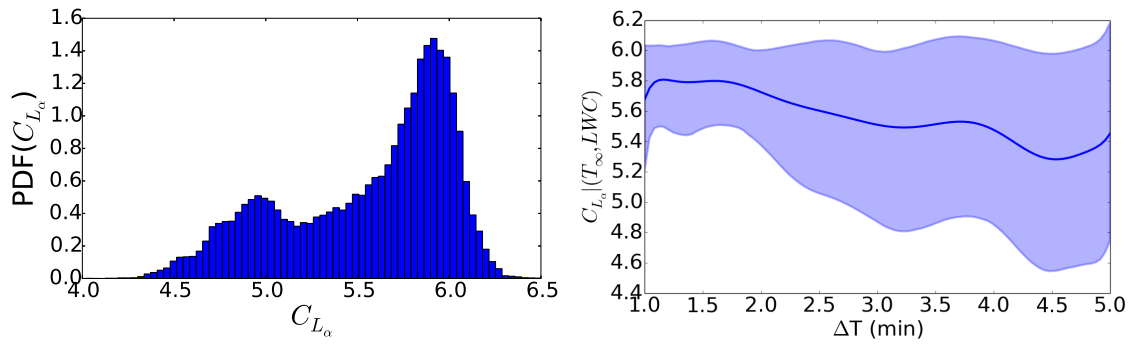
parameter space; these could have a significant impact on the calculated ice shapes, and it would be interesting to investigate those effects. It would also be interesting to investigate whether or not the purely data-driven icing models from Chapter 4 could be fused with CATFISH and used to improve its calculations, perhaps by deriving some correction terms in the ice addition step that account for any discrepancies between CATFISH and the data-driven model predictions.



(a) Quadrature points (colorscale identical to (c)). (b) Ice shapes at quadrature points (colorscale identical to (c)).



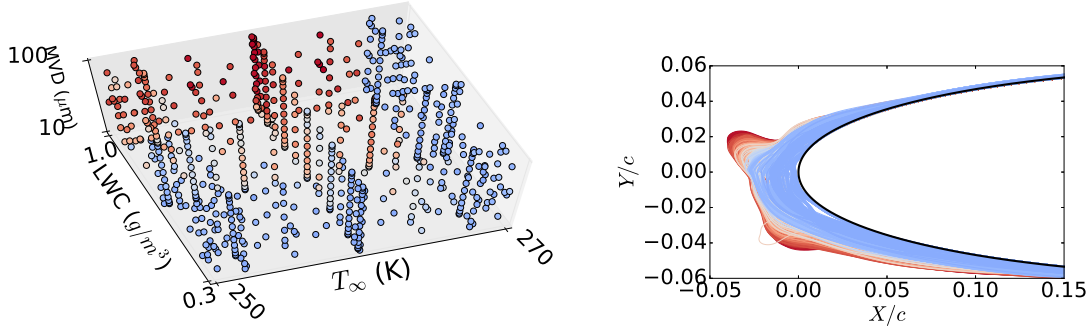
(c) PCE surrogate (parameter units identical to (a)).



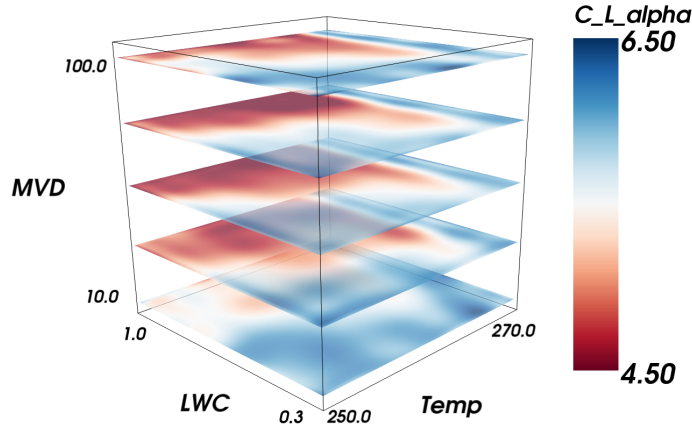
(d) PCE surrogate statistics.

(e) Statistics of C_{L_α} as a function of ΔT . Conditional expectation is solid; shaded region denotes 25th and 75th percentiles.

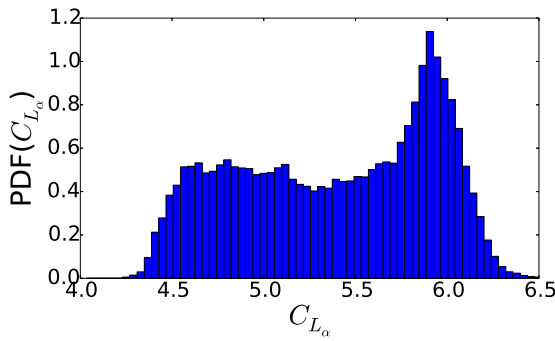
Figure 5.14: Quadrature points, PCE surrogate, and statistics for the 3 parameter (T_∞ , LWC and ΔT) study on lift slope.



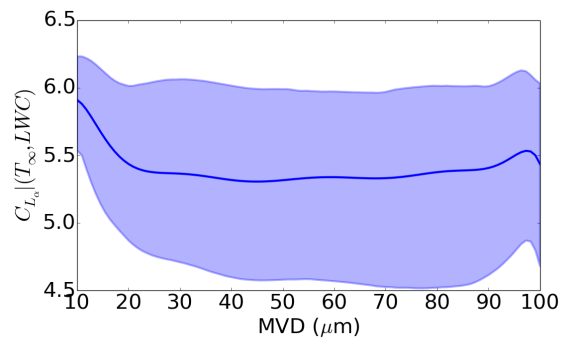
(a) Quadrature points (colorscale identical to (c)). (b) Ice shapes at quadrature points (colorscale identical to (c)).



(c) PCE surrogate (parameter units identical to (a)).

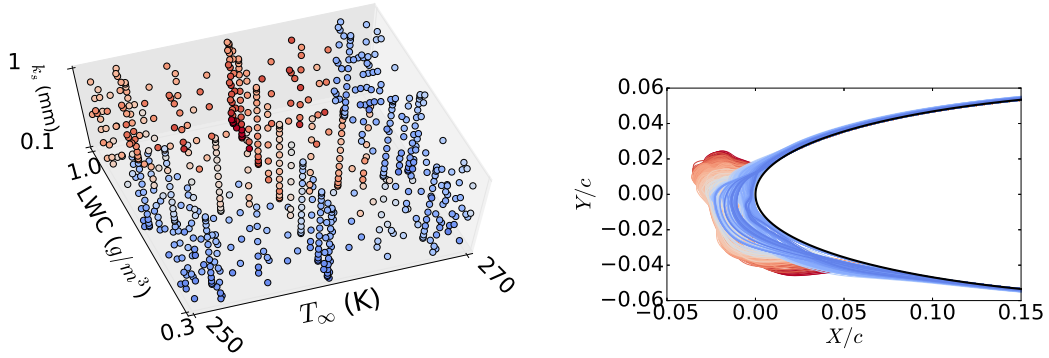


(d) PCE surrogate statistics.

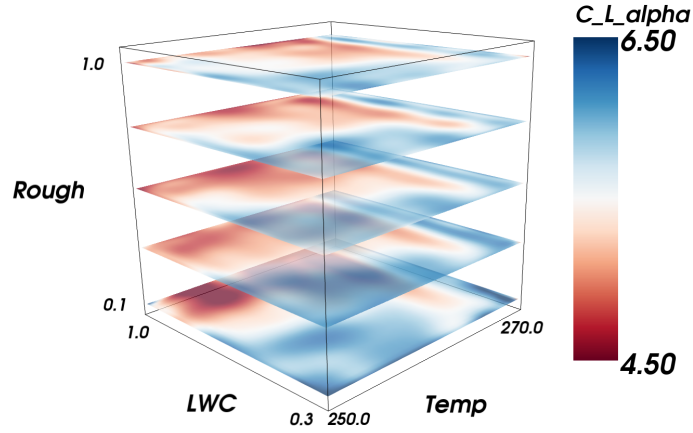


(e) Statistics of C_{L_α} as a function of MVD. Conditional expectation is solid; shaded region denotes 25th and 75th percentiles.

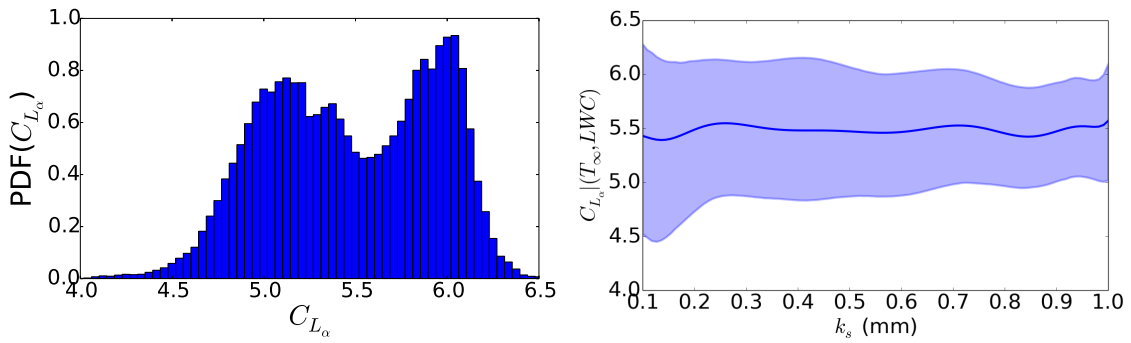
Figure 5.15: Quadrature points, PCE surrogate, and statistics for the 3 parameter (T_∞ , LWC and MVD) study on lift slope.



(a) Quadrature points (colorscale identical to (c)). (b) Ice shapes at quadrature points (colorscale identical to (c)).



(c) PCE surrogate (parameter units identical to (a)).



(d) PCE surrogate statistics.

(e) Statistics of C_{L_α} as a function of k_s . Conditional expectation is solid; shaded region denotes 25th and 75th percentiles.

Figure 5.16: Quadrature points, PCE surrogate, and statistics for the 3 parameter (T_∞ , LWC and roughness) study on lift slope.

Chapter 6

Conclusion

The fundamental intent of this dissertation was to contribute to airplane safety by addressing a problem long recognized by the FAA as a serious safety concern. The accumulation of ice on aircraft wings has been documented as the main cause for many catastrophic in-flight failures that have tragically resulted in the loss of many lives. No category of airplane is inherently any more robust against this problem – icing affects small Cessnas, medium sized business jets, and large commercial transports.

We sought to approach airfoil icing from a different perspective than those which have already been published in the community. The existing literature contains a thorough treatment of icing from experimental, computational, and physical perspectives; however, there have been few efforts to systematically quantify the effects of parametric uncertainty in the icing process on aerodynamic performance. There are several benefits of studying icing through the lens of UQ. The most essential of these is that we can not only describe qualitative trends in parameter space, but we can quantify the statistical distribution of airfoil performance, given some *a priori* statistical knowledge of the uncertainty in the underlying governing icing parameters. This could be useful in the context of flight certification, where an airplane would need to be certified as “safe” to fly in a set of uncertain atmospheric conditions, to a predefined statistical level of certainty (e.g., “we are 95%

confident that this aircraft passes all performance requirements, given this atmospheric uncertainty”).

The work presented in this thesis can be classified as belonging to one of three general categories of UQ: user-informed, data-driven, or computational. The first of these topics represents our first attempt at applying the UQ method of Polynomial Chaos Expansions (PCE) to the airfoil icing problem. Our objective was to (a) describe canonical ice shape classifications, (b) show how these classifications could be represented by a template shape and a few scaling/translation operations, and (c) confirm by experiment that PCE provides an efficient and accurate way of quantifying uncertainty over a wide range of ice shapes. We accomplished these objectives by studying parameterized shape uncertainty in ridge and horn ice cases, and showed how the lift/drag of the airfoil vary statistically.

The next topic presented was the data-driven UQ method. The motivation for considering this approach was to attempt to construct ice shape perturbations/parameters that we more “realistic” (i.e., derived from data) than those we had previously considered in the user-informed UQ. We began by introducing Proper Orthogonal Decomposition (POD) as the tool that could be used to derive an optimal set of ice shape parameters from a database of ice shapes. We applied POD with success to two independent databases of ice shapes, and performed UQ on the resulting parametric spaces. We next showed how we could link physical information (i.e., temperature, LWC, accretion time) to the POD ice shape modes, and used this link to construct a purely data-driven model of ice accretion, which was capable of accepting physical condition inputs and producing random ice shapes corresponding to those conditions as outputs. We commented on the particular usefulness of this data-driven model, as a novel means for benchmarking and improving existing numerical icing codes. Finally, we elaborated on the concept of identifying and segregating self-similar clusters in an ice shape database, and commented on its utility as an empirical classification tool and as a means for simplifying a complex dataset into a few relatively simple datasets.

Our final topic in this dissertation is computational icing UQ. The purpose of this topic was to apply the UQ machinery to understanding the statistical effects of physical parameters. To do this, we developed from scratch an in-house ice accretion code (CATFISH). This numerical code represents a hybrid of techniques from literature, and is built in three modules: a droplet advection module, a thermodynamic module, and an ice addition module. CATFISH interfaces with our pre-existing in-house mesh/flow solvers (FLO103). We first validated this code against several published results, and then proceeded to study the effects of parameterized physical uncertainty on the computed ice shape.

As we have noted, there are multiple exciting avenues for further extensions to the research we have presented. In particular, the data-driven icing model we developed could be improved by the addition of more ice shapes that represent a diverse range of physical conditions. It would also be interesting to examine whether the data-driven icing model could be blended with CATFISH and used as an empirical correction tool.

In summary, the motivation for our work has been introduce tools that can bring a degree of statistical certitude to a problem which has long been viewed as elusively uncertain. Rather than attempt to make incremental contributions to physical models, we aimed for the more general goal of quantifying the effects of uncertainty in the process of icing. It is our hope that the contributions made in this dissertation could be used to improve the continued safety of aircraft flight.

Appendix A

UQ for Cargo Hold Fires

This appendix contains a paper [27] I co-authored on the subject of applying PCE techniques to the problem of uncertainty in cargo hold fires. It is closely related in style and methodology to Chapter 3.

A.1 Introduction

Federal aviation regulations require that all large passenger aircraft have fire detection and suppression systems in all cargo compartments. Several different detection methods are generally used together, such as sensors for temperature, carbon monoxide, smoke particulate, radiation, and optical detection. These sensors are required to detect the fire within 60 seconds of fire ignition [95]. Certification of these systems currently requires expensive ground and in-flight testing. Current fire detection certification focuses on experiments using a small fire in empty cargo holds, such as the narrow-body Boeing 707 fuselage located at the Federal Aviation Administration William J. Hughes Technical center in Atlantic City, New Jersey[68]. Simulating a single fire case is a well-posed problem and relatively straightforward, but of limited utility. Due to the costs associated with these

types of experiments, testing a wide variety of fire sources, positions, and compartment cargo cluttering is not feasible.

CFD tools that can accurately simulate heat and particulate transfer in fire-induced flow in cargo holds can potentially reduce these certification costs by reducing the amount of experimental work necessary. Simulations can then be used to assess the effectiveness of a particular detector placement, as well as optimize their placement in a given cargo hold. The allure of CFD tools is the reduction of monetary costs associated with certification tests; however, a drawback is the associated computational expense. In light of this, an issue that needs to be addressed is how to accurately quantify the uncertainty associated with randomly variable boundary conditions (e.g., fire source location or temperature) while using the least amount of CFD simulations possible.

This work has two main objectives. The first is to establish efficient and accurate CFD tools that can be used to simulate cargo fires over a wide range of parameters. For these simulations we develop an in-house high-order accurate discontinuous Galerkin (DG)[41] flow solver on unstructured meshes. The DG scheme approach is well-suited for computing the turbulent, vorticity-dominated buoyancy-driven flows observed in cargo hold, and unstructured meshes allows one to compute on a complex domain such as those encountered in cluttered cargo holds.

The second objective is to apply techniques of uncertainty quantification to explore the statistical effects of parameterized boundary condition uncertainty with the ultimate goal of optimizing the placement of fire detection systems. In particular, we will be using Polynomial Chaos Expansions (PCE) to achieve this, as this method is efficient and accurate. In order to assess the feasibility of these methods to the problem at hand, we are restricting the problem to the 2-dimensional cross-section of the cargo hold.

A.2 Simulation methodology

A.2.1 Discontinuous Galerkin simulation tool

It is well known that traditional low-order $O(\Delta x^2)$ flow solvers are excessively dissipative for vorticity-dominated flows such as those seen in fires. Adequate resolution of vorticity convection far from its generation source typically requires either a prohibitively fine mesh or a higher-order representation of the flow solution. The in-house simulation tool used in this work is a nodal discontinuous Galerkin (DG) flow solver for the compressible Navier-Stokes equations with buoyancy effects, discretized with an unstructured mesh suitable for complex geometries and arbitrarily-high order of accuracy. The spatial discretization used here follows that detailed by Hesthaven and Warburton[41], and is briefly summarized here.

For a multi-dimensional conservation law of quantity u , flux \mathbf{f} , and source Ψ

$$\frac{\partial u(\mathbf{x}, t)}{\partial t} + \nabla \cdot \mathbf{f}(u(\mathbf{x}, t), \mathbf{x}, t) = \Psi(\mathbf{x}, t) \quad (\text{A.1})$$

the quantities can be approximated by an expansion

$$u(\mathbf{x}, t) \approx u_h(\mathbf{x}, t) = \sum_{i=1}^{N_p} u_h(\mathbf{x}_i, t) l_i(\mathbf{x}) \quad (\text{A.2})$$

where $l_i(\mathbf{x})$ is the multidimensional Lagrange polynomial defined by grid points \mathbf{x}_i , and N_p is the number of nodes in the element, $N_p = (N + 1)(N + 2)/2$ for a triangular element of polynomial order N .

Taking the product of this with the same Lagrange polynomial l_j serving as a test function and integrating by parts on the spatial component over an element V with surface S yields

$$\int_V \left(\frac{\partial u_h}{\partial t} l_j(\mathbf{x}) - \mathbf{f}_h \cdot \nabla l_j(\mathbf{x}) - \Psi_h l_j \right) dV = - \int_S \mathbf{f} l_j(\mathbf{x}) \cdot \mathbf{n} dS \quad (\text{A.3})$$

where flux \mathbf{f}^* is the numerical flux, uniquely defined at element interfaces. In this work the inviscid components of flux are computed using the local Lax-Friedrichs flux splitting, and the viscous flux components use a centered average.

Time integration is performed using the implicit 3rd order backward difference formula

$$\frac{du}{dt} \approx \left(u^{n+1} - \frac{18}{11}u^n + \frac{9}{11}u^{n-1} - \frac{2}{11}u^{n-2} \right) / \left(\frac{6}{11}\Delta t \right) \quad (\text{A.4})$$

where Δt is the discrete time step size and n the time step index.

This discretization leads to a non-linear system of algebraic equations to be solved at each time step. The non-linear system can be written as $\mathbf{F}(\mathbf{u}) = 0$, and Newton's method can be used with the iterative step index k ,

$$\mathbf{F}(\mathbf{u}^{k+1}) = \mathbf{F}(\mathbf{u}^k) + \mathbf{F}'(\mathbf{u}^k)(\mathbf{u}^{k+1} - \mathbf{u}^k) \quad (\text{A.5})$$

resulting in a sequence of linear systems

$$\mathbf{J}(\mathbf{u}^k)\delta\mathbf{u}^k = -\mathbf{F}(\mathbf{u}^k), \quad \mathbf{u}^{k+1} = \mathbf{u}^k + \delta\mathbf{u}^k \quad (\text{A.6})$$

for the Jacobian $\mathbf{J} = \mathbf{F}'(\mathbf{u})$. The Jacobian matrix \mathbf{J} is a very large sparse matrix which can be prohibitively expensive to store in computer memory. Fortunately the Krylov subspace methods for the solution of linear algebraic systems do not require this matrix itself, but only the matrix-vector product. This can be approximated by a finite difference

$$\mathbf{J}\delta\mathbf{u} \approx [\mathbf{F}(\mathbf{u} + \epsilon\delta\mathbf{u}) - \mathbf{F}(\mathbf{u})]/\epsilon \quad (\text{A.7})$$

for a small ($\sim 10^{-6}$) parameter ϵ . In this work the restarted GMRES algorithm is used for the solution of the linear systems at each Newton iteration, with the Newton method progressing until a desired convergence tolerance is reached and the physical time step is advanced. This approach for solving non-linear systems by coupling matrix-free Krylov

iterative methods for linear systems with Newton iterations is known as a “Jacobian-free Newton-Krylov” (JFNK) method, and is detailed in the review paper by Knoll & Keyes[48].

A.2.2 Cargo hold geometry and boundary conditions

The geometry of interest here is the forward cargo compartment of a Boeing 707. This geometry shortly after a small fire is started in the center can be seen in figure A.1a. The flow is entirely driven by buoyant effects due to the local heating produced by the fire. We note that a short distance away from the fire source, there is no longer a significant effect on the dynamics of the flow due to the actual chemical combustion process taking place. This type of flow can therefore be accurately modeled as a heat source addition into non-reactive air, freeing us from the need to tackle the computationally expensive details of the combustion problem. Experimental results of the full 3D case and background on this problem can be found in work by Oztekin et al[68, 69].

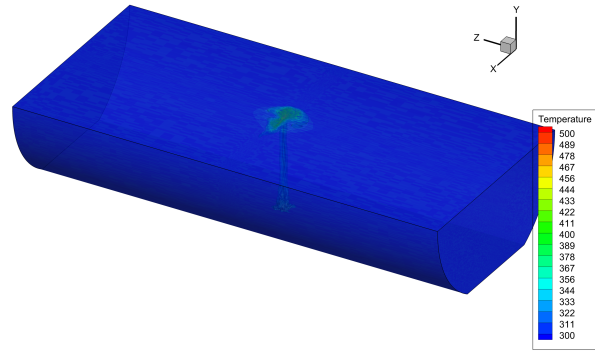
A typical simulation of a 2D cross-section of the cargo geometry (computed using our in-house DG code) can be seen in figure A.1b. A turbulent plume rising from the heat sources drives vortical flow around the compartment feeding back into itself at the bottom. We note recirculation regions in both upper corners leading to stagnation regions where streamlines are separating, indicating a sensor position there would be less effective than at other locations. The turbulent, buoyant flow is instantaneously asymmetric, but statistically averaged is largely symmetric due to symmetric boundary conditions. The base of the geometry is $1.107m$ wide, and the ceiling is $2.286m$ wide and $1m$ tall.

In this work, we restrict the analysis to a 2D cross-section of a cargo hold. All boundary conditions are isothermal, with the majority of the wall boundary fixed to the initial bulk temperature non-dimensionalized to $T_\infty = 1$. A $0.1m$ wide section of the floor is then set to an isothermal condition at a multiple of the bulk temperature in order to model a heat source. The temperature source T_s is examined in the range between $T_s = 1.2$ and 1.5 , and

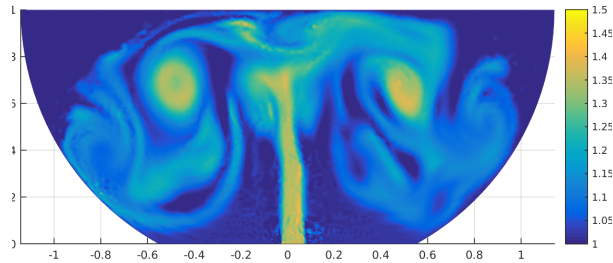
the temperature location x_s in the range between $x_s = 0.0$ (centerline) and $0.503m$ (the rightmost possible location for $0.1m$ wide source.) Due to symmetry, sources need only be placed to one side of the geometry in order to analyze sources at a reflected point along the floor.

All DNS simulations here are performed using cubic ($N = 3$) elements, with the 2D meshes consisting of approximately 1500 triangular cells. This results in 10 nodes per cell for each of the 4 quantities (density, x and y momentum, and energy) to be solved, for a total of $\sim 60,000$ degrees of freedom.

Sample flowfield snapshots of temperature are displayed in Figures A.2, A.3, and A.4. These figures illustrate the wide range of spatio-temporal flow behaviors that are possible when the fire source location and temperature are varied, and motivates a study aimed at quantifying the statistics of some measure of the flow given parameterized uncertainty in the fire source location and temperature.



(a) Temperature field after start-up of a small fire in the center.



(b) Flow driven by a heat source in a 2D cross-section. Colormap shown is temperature normalized by the initial bulk temperature.

Figure A.1: Example flowfields of buoyancy-driven flow in Boeing 707 cargo hold geometry.

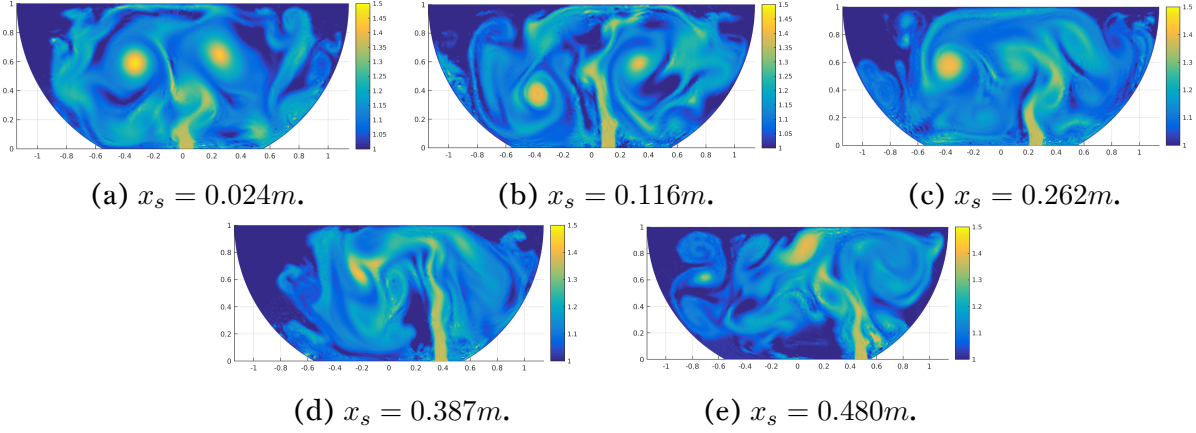


Figure A.2: Temperature fields for $T_s = 1.486$ source at the 5 source locations, time $t = 10s$ after startup.

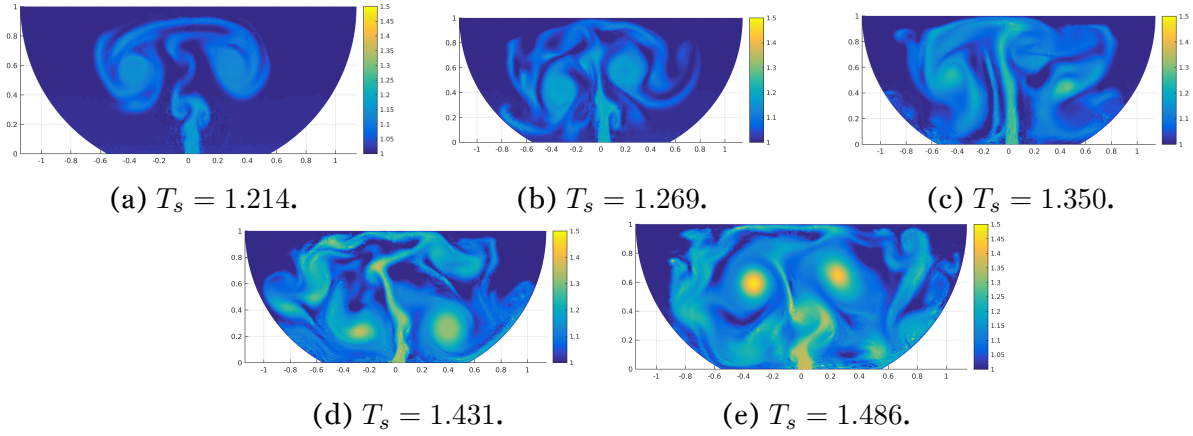


Figure A.3: Temperature fields at $x_s = 0.024m$ for the 5 values of temperature source, time $t = 10s$ after startup.

A.3 Case Study: 2-D

Cargo Hold Fire with Uncertain Location/Temperature

In this section, we apply the tools discussed to study a test problem in which both the fire source location and temperature are independent, uncertain parameters with some joint probability distribution $\rho(Z)$. We choose to equip both parameters with a uniform distribution. We assume that the range of possible fire source locations consists of the right half of the cargo hold floor. This is done in order to study the effect of spatial asymmetry

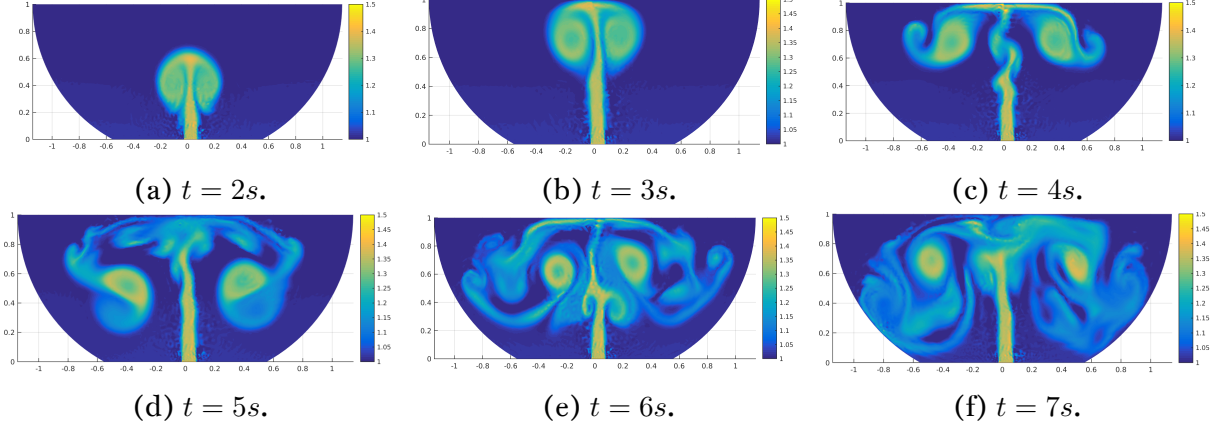


Figure A.4: Temperature field time evolution for $T_s = 1.486$, $x_s = 0.024$ case.

on the UQ problem. We assume that the range of possible fire source temperatures is given by the interval $[1.2, 1.5] \times T_\infty$.

Given that both of our parameters are uniformly distributed, our PC basis consists of the Legendre polynomials. We choose to truncate the PC expansion (2.1) at total order $N = 4$. This implies that we use a 5×5 grid of collocation points in the parameter space to evaluate the projection integrals (2.5), corresponding to the tensor product of the five zeros of the fifth order Legendre polynomials (suitably shifted/scaled) with themselves. These nodes are given in Table A.1. These are the collocation points that specify the fire source locations/temperatures that we will simulate using our DG code.

Temperature strength T_s , 5×5	1.214, 1.269, 1.350, 1.431, 1.486
Temperature location $x_s(m)$, 5×5	0.024, 0.116, 0.252, 0.387, 0.480

Table A.1: Discrete simulation parameters for uncertainty quantification study. The parameter sweep is performed using a tensor product of these values.

Quantifying an entire field quantity $u(x, t; Z)$ using PCE is difficult. This is because the spatio-temporal behavior of the flow can vary significantly with fire source location and temperature, which makes it difficult to interpolate in parameter space accurately using 4^{th} order polynomials. Therefore, we focus on a set of observables more amenable to our techniques, corresponding to the temperature along a 1-D segment near the cargo hold

ceiling. This observable vector is highly relevant from an engineering standpoint, since it informs the choice of fire sensor placement.

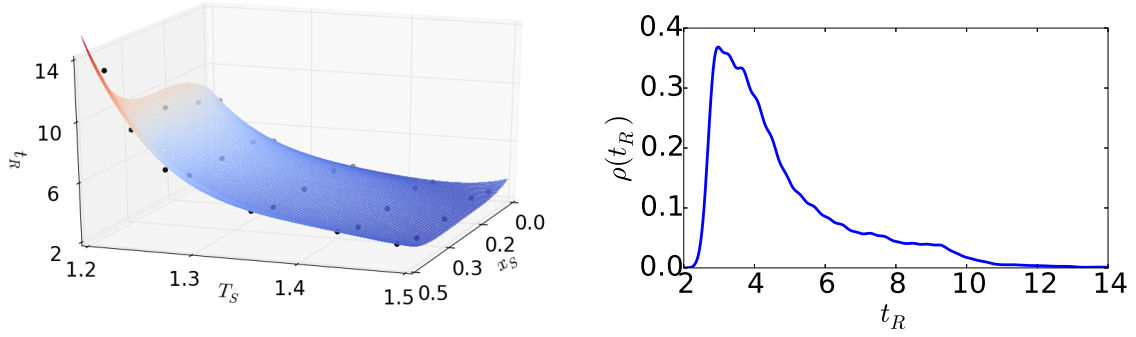
We denote the temperature along the line segment at height $y = 0.95$ in the cargo hold as $T_C(x, t; Z)$. Intuitively, one would expect a certain characteristic rise time $t_R(Z)$ of the buoyant plume from the fire source, which should be dominated by the source temperature (and possibly affected by source location if the plume interacts with the cargo walls). We define $t_R(Z)$ as the time required from the start of the fire to detection at any point on the ceiling. For early fire detection, we are interested in the ceiling temperature distribution averaged over a short period of time beginning at $t_R(Z)$ (for some choice of Z). Therefore, we define the time-averaged ceiling temperature distribution:

$$\overline{T}_C(x; Z) = \frac{1}{\Delta t} \int_{t_R(Z)}^{t_R(Z) + \Delta t} T_C(x, t; Z) dt \quad , \quad (\text{A.8})$$

and we quantify uncertainty in the observables $\overline{T}_C(x; Z)$ and $t_R(Z)$. We use an averaging time period of $\Delta t = 1s$. Note that $t_R(Z)$ is a scalar quantity and hence has one PC expansion associated with it, whereas $\overline{T}_C(x; Z)$ is a function in x (which is discretized as a vector at discrete locations) and hence has one PC expansion for each location in x we choose to measure. The units of $t_R(Z)$ will be seconds; as noted previously, $\overline{T}_C(x; Z)$ is temperature normalized by the initial bulk temperature T_∞ .

We first examine the rise time $t_R(Z)$. The PCE surrogate model for rise times $t_R(Z)$ is shown in Figure A.5, along with statistical quantities in Table A.2. Examination of Figure A.5 and Table A.2 confirms our hypothesis that the characteristic rise time t_R is dominated by the source temperature. As shown, source temperature has a Sobol index of 0.95, which means that 95% of the variance in the distribution of t_R can be attributed to source temperature (either acting alone or interacting with location). The only portion of parameter space that really is affected strongly by source location appears to be the “corner” area of parameter space where the source is very close to the cargo hold wall

and the source temperature is very low. The result of this combination of variables is that the initial buoyant plume “rolls over” toward the center of the cargo hold and falls back downward toward the floor before reaching a height of $y = 0.95$ (where we are observing ceiling temperature). This time-dependent behavior is illustrated in Figure A.6. It is not until several seconds after this has occurred that subsequent buoyant plumes finally touch the ceiling. This combination of low temperature with a wall effect is what accounts for the tail of the distribution of t_R .



(a) PCE surrogate map of $t_R(Z)$, together (b) Probability density function $\rho(t_R(Z))$ with the values at the 25 quadrature nodes. (approximated using 10,000 random samples of the PCE surrogate).

Figure A.5: PCE surrogate for $t_R(Z)$.

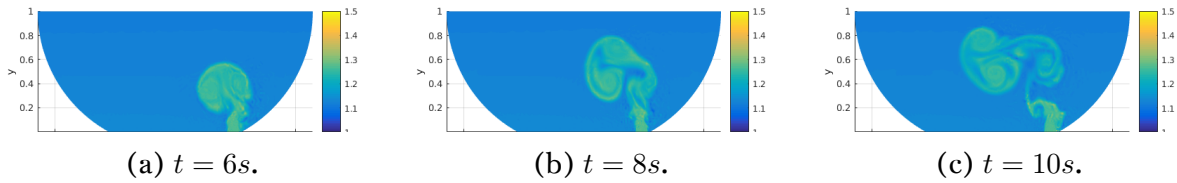


Figure A.6: Temperature field snapshots with $Z = (0.48, 1.21)$. The initial plume falls toward the floor without ever touching the ceiling, explaining the unusually long rise time.

We next turn our attention to the time-averaged ceiling temperature distribution $\overline{T_C}(x; Z)$. The time-averaged ceiling temperature distributions at the quadrature nodes

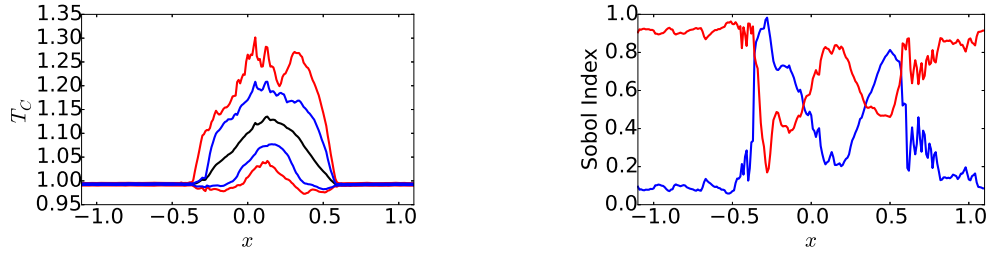
Mean	4.8
Variance	3.9
Sobol Index 1 (Location)	0.08
Sobol Index 2 (Temperature)	0.95

Table A.2: Statistical quantities of interest for $t_R(Z)$. are shown in Figure A.9. The mean distribution along with confidence intervals – computed from Monte Carlo samples of the PCE surrogate – is shown in Figure A.7a. The accuracy of the PC model for ceiling temperature can be verified by comparing the PC interpolation to data at various points in parameter space. This is done in Figure A.10, which confirms that our the PC model provides reasonably accurate interpolation.

We can also examine the total Sobol indices as a function of x for the ceiling temperature observable, which indicate which of the two uncertain parameters best explains the variance in the ceiling temperature. These Sobol indices are displayed in Figure A.7b. As can be seen, source temperature is the dominant parameter in the area around the maximum of the mean profile. The peripheral areas are dominated by source location. The explanation of this phenomenon is natural: source temperature controls the intensity of the temperature fluctuations observed on the ceiling where they are hottest, but source location determines whether or not temperature fluctuations are actually observed at all in the peripheral areas.

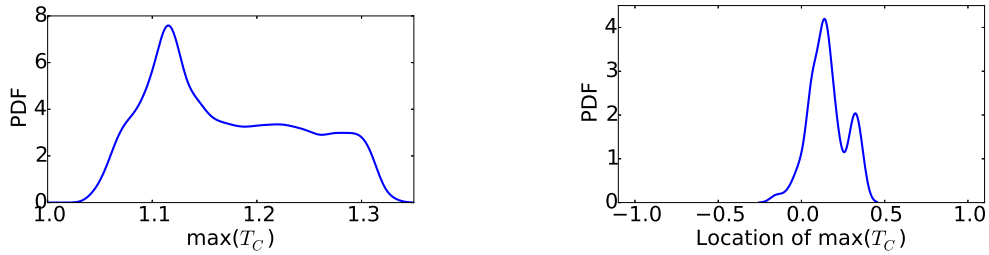
Having a PCE surrogate for ceiling temperature also means that we can compute the statistics of any quantity derived from it. Two particularly relevant examples of this include the maximum value of $\overline{T_C}(x; Z)$ as well as its location along the ceiling. We display these statistics in Figure A.8. We see that a wide range of maximum ceiling temperatures are possible, with a skew toward lower maximum values. We also see a clear skew in the location of the maximum ceiling temperature to the right of the center (as would be expected from the asymmetry in the source location). Computing correlations between these output quantities and our uncertain parameters confirms what one would expect – source

temperature dominates the maximum value of the ceiling temperature, whereas source location dominates its location.



(a) *Black*: mean time-averaged ceiling temperature profile. *Blue*: 68% confidence interval. *Red*: 95% confidence interval. (b) Sobol indices for ceiling temperature at points along the ceiling. *Blue*: source location. *Red*: source temperature.

Figure A.7: Statistical quantities of interest for time-averaged ceiling temperature.



(a) Distribution of the maximum value of the time-averaged ceiling temperature. (b) Distribution of the location of the maximum value of the time-averaged ceiling temperature.

Figure A.8: Distributions of maximum ceiling temperature value and location. Computed from 10,000 Monte Carlo samples of the PCE surrogate.

Combining all of this information together gives a clear and insightful view of the physics of our cargo hold problem. We see that the main effects of increasing temperature are to increase the maximum ceiling temperature, and to decrease the rise time. The main effect of location is to influence whether or not fluctuations in ceiling temperature are observed in the peripheral regions of the ceiling. The fact that we observe these intuitive trends in our surrogate model gives us further validation of the claim that PCE methods

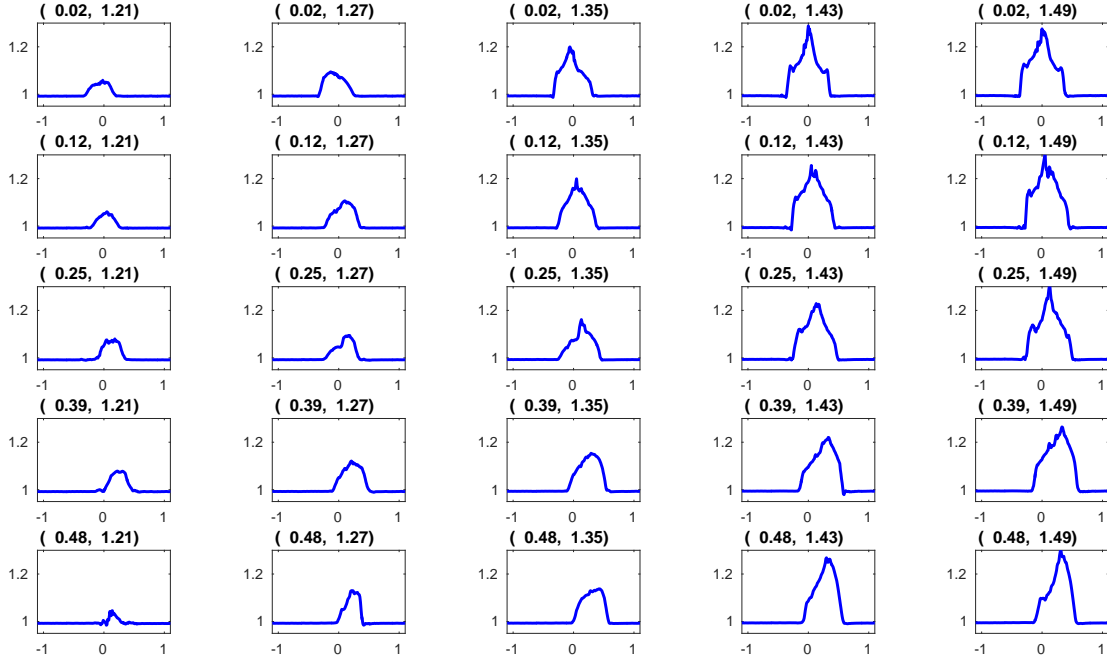


Figure A.9: Time-averaged ceiling temperature distributions collected at the 25 quadrature nodes. Each subtitle corresponds to the parameter pair (x_S, T_S) .

may provide a method for UQ which is not only efficient, but also *accurate* for this class of problems.

Of course, the main usage of these UQ tools is not just to confirm intuition, but to *quantify* it. We see that, on average, we can expect a ceiling temperature distribution which is roughly symmetric between the limits $y \in [-0.35, 0.60]$, with a maximum around $y = 0.125$. We can also give confidence intervals on the mean ceiling temperature distribution (Figure A.7), and estimate the probability distributions for the value and location of the maximum ceiling temperature (Figure A.8).

A.4 Summary

The purpose of this paper was establish a framework for performing efficient, accurate investigations of the statistical variations in cargo hold fires that occur due to parameterized

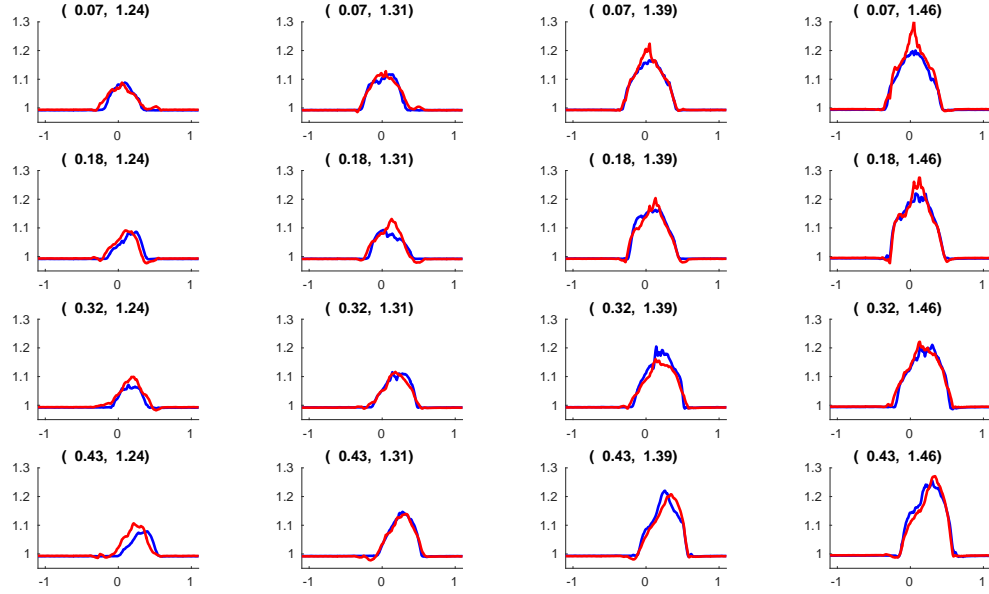


Figure A.10: Time-averaged ceiling temperature distributions collected at points on the 4x4 mesh which is dual to the 5x5 mesh. Data are displayed in blue; PC models are displayed in red. Each subtitle corresponds to the parameter pair (x_s, T_s) .

uncertainty in the boundary conditions. We address two related problems – increasing the numerical accuracy of the CFD simulation, and uncertainty quantification. Higher order numerical accuracy is necessary because traditional finite-volume schemes require a prohibitively fine mesh in order to resolve the vortex-dominated flows seen in cargo hold fire solutions. The need for uncertainty quantification stems from the fact that the boundary conditions of the cargo hold fire will always be fundamentally unpredictable, since one can never know *a priori* exactly where the fire will start, how hot it will be, how much luggage clutter there is, etc.

In order to provide greater simulation accuracy, we developed an in-house discontinuous Galerkin (DG) flow solver for the compressible Navier-Stokes equations with buoyancy effects. This code also features an unstructured mesh suitable for complex geometries. To make uncertainty quantification feasible, we first reduced the problem from quantifying the full flow field to quantifying *measures* of the flow field – a characteristic rise time

of the buoyant flow, and a time-averaged ceiling temperature distribution. This made the problem amenable to treatment with spectral expansion methods, and so we used PCE as the tool to efficiently and accurately quantify the effects of fire source location and temperature. A case study of a 2D cargo hold geometry in which the fire source location and temperature were uncertain confirmed that PCE tools provide a viable UQ approach, and keep the number of required CFD simulations to a minimum.

We are currently working to extend these methods to 3D cargo hold fire configurations. We are also planning to investigate methods for accounting for geometric uncertainty in cargo hold luggage clutter, which was not accounted for in our empty cargo hold geometries.

A.5 Acknowledgments

This research was supported under the Federal Aviation Administration (FAA) Joint University Program (JUP). The authors would also like to thank Ezgi Oztekin for helpful discussions on previous cargo hold fire research as well as a tour of the cargo hold fire testing facilities at the FAA Tech Center.

Appendix B

Droplet Distributions

Table B.1: 27-Bin Droplet Distributions

Bin	Weight	MVD $20\mu\text{m}$	MVD $111\mu\text{m}$	MVD $236\mu\text{m}$
1	4.75	3.77	10.87	15.91
2	4.75	8.42	24.52	45.34
3	4.75	10.07	29.64	74.84
4	4.75	11.55	34.96	102.0
5	4.75	12.98	44.74	122.6
6	4.75	14.30	58.34	141.6
7	4.75	15.50	70.67	160.5
8	4.75	16.65	81.29	178.4
9	4.75	17.68	91.19	197.7
10	4.75	18.61	100.9	218.0
11	4.75	19.54	110.6	240.8
12	4.75	20.51	119.5	271.0
13	4.75	21.51	128.8	320.0
14	4.75	22.51	140.1	393.5
15	4.75	23.58	152.8	455.5
16	4.75	24.73	165.9	494.6
17	4.75	25.98	179.4	534.1
18	4.75	27.47	193.7	578.0
19	4.75	29.32	207.2	624.0
20	4.75	31.85	219.7	670.9
21	1.00	33.81	227.4	701.1
22	1.00	34.83	230.1	713.6
23	1.00	36.22	237.8	728.3
24	0.50	37.47	250.5	742.1
25	0.50	38.74	264.2	752.7
26	0.50	40.67	279.5	763.2
27	0.50	44.37	312.6	1046.8

Appendix C

Ice Test Matrix

Table C.1: Ice Accretion Test Matrix

Run	P_{∞} (Pa)	T_{∞} (K)	U_{∞} (m/s)	M_{∞}	Re_{∞}	Chord (m)
409	90760	265.07	67.1	0.21	5.60e6	0.5334
402	90760	263.71	102.8	0.32	7.76e6	0.5334
405	90760	250.37	102.8	0.32	7.76e6	0.5334
404	90760	256.56	102.8	0.32	7.76e6	0.5334
421	90760	268.4	67.1	0.20	5.21e6	0.5334
422	90760	268.4	67.1	0.20	5.21e6	0.5334
423	90760	265.07	67.1	0.21	5.21e6	0.5334
424	90760	259.51	67.1	0.21	5.53e6	0.5334
425	101510	244.51	67.1	0.21	6.16e6	0.5334
426	101520	265.07	67.1	0.21	5.33e6	0.5334
427	101520	265.07	67.1	0.21	5.33e6	0.5334
428	101520	265.07	67.1	0.21	5.33e6	0.5334
429	101550	262.04	102.8	0.21	5.33e6	0.5334
308	90760	262.04	102.8	0.21	7.76e6	0.5334
314	90760	262.04	102.8	0.21	7.76e6	0.5334
316	90760	262.04	102.8	0.21	7.76e6	0.5334
206	101510	265.37	102.8	0.21	5.21e6	0.5334
207	101510	256.49	102.8	0.21	5.21e6	0.5334
212	101510	262.04	102.8	0.21	5.21e6	0.5334
213	101510	262.04	102.8	0.21	5.21e6	0.5334

Table C.2: Ice Accretion Test Matrix (continued)

Run	MVD (μm)	LWC (g/m^3)	Time (min)
409	30e-6	1.30e-3	6
402	20e-6	0.55e-3	7
405	20e-6	0.55e-3	7
404	20e-6	0.55e-3	7
421	20e-6	1.00e-3	6
422	20e-6	1.00e-3	6
423	20e-6	1.00e-3	6
424	20e-6	1.00e-3	6
425	20e-6	1.00e-3	6
426	30e-6	1.06e-3	6
427	30e-6	1.30e-3	6
428	30e-6	1.60e-3	6
429	40e-6	0.86e-3	5
308	20e-6	1.00e-3	4
314	15e-6	0.60e-3	6
316	20e-6	0.55e-3	4
206	20e-6	0.34e-3	12
207	20e-6	0.34e-3	12
212	30e-6	0.44e-3	9
213	40e-6	0.48e-3	8

Bibliography

- [1] B. M. Adams, W. J. Bohnhoff, K. R. Dalbey, J. P. Eddy, M. S. Eldred, D. M. Gay, K. Haskell, P. D. Hough, and L. P. Swiler. DAKOTA, a multilevel parallel object-oriented framework for design optimization, parameter estimation, uncertainty quantification, and sensitivity analysis: Version 5.3 user's manual. Sandia Technical Report SAND2010-2183, 2013.
- [2] Inc. Adaptive Computing Enterprises. TORQUE resource manager: administrator guide 4.2.6. Available from <http://docs.adaptivecomputing.com/torque/4-2-6/torqueAdminGuide-4.2.6.pdf>, 2013.
- [3] H. E. Addy. Ice accretions and icing effects for modern airfoils. Technical Report TP 2000-210031, NASA, April 2000.
- [4] H. E. Addy Jr., D. R. Miller, and R. F. Ide. A study of large droplet ice accretions in the NASA-Lewis IRT at near-freezing conditions; part 2. NASA TM-107424, April 1997.
- [5] H.E. Addy Jr., A.P. Broeren, J.G. Zoeckler, and S. Lee. A wind tunnel study of icing effects on a business jet airfoil. AIAA Paper 2003-0727 and NASA TM-2003-212124, 2003.
- [6] Federal Aviation Administration. Airplane and engine certification requirements in supercooled large drop, mixed phase, and ice crystal icing conditions; final rule. *Federal Register*, 79(213):65508–65540, November 2014.
- [7] D.N. Anderson and J. Shin. Characterization of ice roughness from simulated icing encounters. AIAA Paper 97-0052, 1997.
- [8] B. Aupoix and P.R. Spalart. Extensions of the Spalart-Allmaras turbulence model to account for wall roughness. *International Journal of Heat and Fluid Flow*, pages 454–462, 2003.
- [9] M.P. Barzi. Multidisciplinary optimization of in-flight electro-thermal ice protection systems. Ph.D. dissertation, McGill University, October 2014.
- [10] H. Beaugendre, F. Morency, and W. G. Habashi. Development of a second-generation in-flight icing simulation code. *Transactions of the ASME*, 128:378–387, 2006.

- [11] H. Beaugendre, F. Morency, and W.G. Habashi. Roughness implementation in FENSAP-ICE: model calibration and influence on ice shapes. *Journal of Aircraft*, (6):1212–1215, 2003.
- [12] L.A. Blumenthal. Surface pressure measurement on a three-dimensional ice shape. MS Thesis, Department of Aerospace Engineering, University of Illinois Urbana-Champaign, 2005.
- [13] National Transportation Safety Board. In-flight icing encounter and loss of control; Simmons Airlines, D.B.A. American Eagle flight 4184 Avions de Transport regional (ATR) model 72-212, n401am; Roselawn, Indiana, October 31, 1994. NTSB Aircraft Accident Report NTSB/AAR-96/01, 1996.
- [14] National Transportation Safety Board. In-flight icing encounter and uncontrolled collision with terrain; Comair flight 3272, Embraer EMB-120rt, N265CA; Monroe, Michigan, January 9, 1997. NTSB Aircraft Accident Report NTSB/AAR-98/04, 1998.
- [15] National Transportation Safety Board. NTSB press release: Acting chairman hart announces slight drop in 2013 transportation fatalities in most categories; rail deaths rise. Available from NTSB, <http://www.nts.gov/news/press-releases/Pages/PR20150202.aspx>, 2015.
- [16] D.T. Bowden. Effect of pneumatic de-icers and ice formations on aerodynamic characteristics of an airfoil. NACA TN 3564, February 1954.
- [17] M. B. Bragg, A. P. Broeren, and L. A. Blumenthal. Iced-airfoil aerodynamics. *Progress in Aerospace Sciences*, 41:323–362, 2005.
- [18] M.B. Bragg and G.M. Gregorek. Aerodynamic characteristics of airfoils with ice accretions. AIAA Paper 82-0282, 1982.
- [19] M.B. Bragg and G.M. Gregorek. Wind tunnel investigation of airfoil performance degradation due to icing. AIAA Paper 82-0582, 1982.
- [20] M.B. Bragg and E. Loth. Effects of large-droplet ice accretion on airfoil and wing aerodynamics and control. FAA/DOT/AR-00/14, April 2000.
- [21] A. Broeren, M. Potapczuk, J. Riley, P. Villedieu, F. Moens, and M. Bragg. Swept-wing ice accretion characterization and aerodynamics. AIAA Paper 2013-2824, 2013.
- [22] A.P. Broeren, H.E. Addy, and M.B. Bragg. Flowfield measurements about an airfoil with leading-edge ice shapes. AIAA Paper 2004-0059, 2004.
- [23] R.E. Brumby. Wing surface roughness – cause effect. D.C. Flight Approach, January 1979.

- [24] C. Crowe. Review – numerical models for dilute gas-particle flows. *Journal of Fluids Engineering*, pages 297–303, 1982.
- [25] M.J. Cummings. Airfoil boundary-layer transition due to large isolated 3-D roughness elements in a favorable pressure gradient. MS Thesis, Department of Aeronautical and Astronautical Engineering, University of Illinois Urbana-Champaign, 1995.
- [26] M.J. Cummings and M.B. Bragg. Boundary-layer transition due to isolated 3-D roughness on an airfoil leading edge. *AIAA Journal*, pages 1949–52, 1996.
- [27] A. DeGennaro, M. Lohry, L. Martinelli, and C.W. Rowley. Uncertainty quantification for cargo hold fires. AIAA Paper 2016-1419, January 2016.
- [28] A. DeGennaro, C. W. Rowley, and L. Martinelli. Uncertainty quantification for airfoil icing using polynomial chaos expansions. *Journal of Aircraft*, pages 1404–1411, 2015.
- [29] A. DeGennaro, C.W. Rowley, and L. Martinelli. Data-driven low-dimensional modeling and uncertainty quantification for airfoil icing. AIAA Paper 2015-3383, 2015.
- [30] L. Devroye. *Non-Uniform Random Variate Generation*. Springer-Verlag, New York, 1986.
- [31] M. Fiedler. A property of eigenvectors of nonnegative symmetric matrices and its applications to graph theory. *Czechoslovak Mathematical Journal*, pages 619–633, 1975.
- [32] Nicolas Garcia Rosa, Philippe Villedieu, and Gerard Lavergne. A statistical model for droplet-wall interaction. In *2nd Colloque INCA*, October 2008.
- [33] T. Gerstner and M. Griebel. Numerical integration using sparse grids. *Numerical Algorithms*, 18:209–232, 1998.
- [34] R. G. Ghanem and P. Spanos. *Stochastic Finite Elements: A Spectral Approach*. Springer-Verlag, New York, 1991.
- [35] G. H. Golub and J. H. Welsch. Calculation of Gauss quadrature rules. *Mathematics of Computation*, 23:221–230, 1969.
- [36] H.M. Gurbaki. Ice-induced unsteady flowfield effects on airfoil performance. Ph.D. dissertation, Department of Aeronautical and Astronautical Engineering, University of Illinois Urbana-Champaign, 2003.
- [37] R. J. Hansman and S. Turnock. Investigation of surface water behavior during glaze ice accretion. *Journal of Aircraft*, 26(2):140–147, 1989.

- [38] R. J. Hansman, K. Yamaguchi, B. Berkowitz, and M. Potapczuk. Modeling of surface roughness effects on glaze ice accretion. *Journal of Thermophysics and Heat Transfer*, 5(1):54–60, 1991.
- [39] J.A. Hartigan and M.A. Wong. Algorithm AS 136: a k-means clustering algorithm. *Journal of the Royal Statistical Society, Series C*, pages 100–108, 1979.
- [40] J.C. Helton and F.J. Davis. Sampling-based methods for uncertainty and sensitivity analysis. Technical Report SAND99-2240, Sandia National Laboratories, Albuquerque, NM, 2000.
- [41] J.S. Hesthaven and T. Warburton. *Nodal Discontinuous Galerkin Methods: Algorithms, Analysis, and Applications*. Texts in Applied Mathematics. Springer, 2008.
- [42] P. Holmes, J. Lumley, G. Berkooz, and C. Rowley. *Turbulence, Coherent Structures, Dynamical Systems and Symmetry*. Cambridge University Press, New York, 2012.
- [43] R.L. Iman and M.J. Shortencarier. A Fortran 77 program and user’s guide for the generation of Latin Hypercube samples for use with computer models. Technical Report NUREG/CR-3624, SAND83-2365, Sandia National Laboratories, Albuquerque, NM, 1984.
- [44] M.F. Kerho. Effect of large distributed roughness near an airfoil leading edge on boundary-layer development and transition. Ph.D. dissertation, Department of Aeronautical and Astronautical Engineering, University of Illinois Urbana-Champaign, 1995.
- [45] M.F. Kerho and M.B. Bragg. Airfoil boundary-layer development and transition with large leading-edge roughness. *AIAA Journal*, pages 75–84, 1997.
- [46] H.S. Kim and M.B. Bragg. Effects of leading-edge ice accretion geometry on airfoil aerodynamics. AIAA Paper 99-3150, 1999.
- [47] R. J. Kind. Ice accretion simulation evaluation test. NATO RTO Technical Report RTO-TR-038, 2001.
- [48] D.a. Knoll and D.E. Keyes. Jacobian-free Newton–Krylov methods: a survey of approaches and applications. *Journal of Computational Physics*, 193(2):357–397, jan 2004.
- [49] B. Landsberg. Aircraft icing. Safety Advisor, Weather No. 1, Aircraft Owners and Pilots Association Air Safety Foundation, 2002.
- [50] S. Lee. Effects of supercooled large-droplet icing on airfoil aerodynamics. Ph.D. dissertation, Department of Aeronautical and Astronautical Engineering, University of Illinois Urbana-Champaign, 2001.
- [51] S. Lee and M.B. Bragg. Effects of simulated-spanwise ice shapes on airfoils: experimental investigation. AIAA Paper 99-0092, 1999.

- [52] S. Lee and M.B. Bragg. Experimental investigation of simulated large-droplet ice shapes on airfoil aerodynamics. *Journal of Aircraft*, pages 844–50, 1999.
- [53] S. Lee and M.B. Bragg. Investigation of factors that influence iced-airfoil aerodynamics. AIAA Paper 2000-0099, 2000.
- [54] S. Lee and M.B. Bragg. Investigation of factors affecting iced-aircraft aerodynamics. *AIAA Journal*, pages 499–508, 2003.
- [55] O. LeMaitre. *Spectral Methods for Uncertainty Quantification*. Springer, 2010.
- [56] S.P. Lloyd. Least squares quantization in PCM. *IEEE Transactions on Information Theory*, pages 129–137, 1982.
- [57] J.B. MacQueen. Some methods for classification and analysis of multivariate observations. Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability, 1967.
- [58] J. Malik and J. Shi. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 888–905, 2000.
- [59] L. Martinelli and A. Jameson. Validation of a multigrid method for the Reynolds Averaged Equations. AIAA paper 88-0414, 1988.
- [60] M.D. McKay, R.J. Beckman, and W.J. Conover. A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*, pages 239–245, 1979.
- [61] B.L. Messinger. Equilibrium temperature of an unheated icing surface as a function of air speed. *Journal of Aeronautical Sciences*, pages 29–42, 1953.
- [62] I. Mezic. Uncertainty: Some conceptual thoughts and a good sampling method. USA/South America Symposium on Stochastic Modeling and Uncertainty Quantification, August 2011.
- [63] D. R. Miller, H. E. Addy Jr., and R. F. Ide. A study of large droplet ice accretions in the NASA-Lewis IRT at near-freezing conditions. AIAA Paper 96-0934, 1996.
- [64] C. H. R. Mundo, M. Sommerfeld, and C. Tropea. Droplet-wall collisions: experimental studies of the deformation and break-up process. *International Journal Multiphase Flow*, 21(2):151–173, 1995.
- [65] T. G. Myers. Extension to the Messinger model for aircraft icing. *AIAA Journal*, 39(2), 2001.
- [66] W. Olsen and E. Walker. Close up motion pictures of the icing process. NASA Lewis Research Center Film, 1983.
- [67] W.A. Olsen and E. Walker. Experimental evidence for modifying the current physical model for ice accretion on aircraft structures. NASA TM 87184, May 1986.

- [68] E. S. Oztekin. Heat and mass transfer due to a small-fire in an aircraft cargo compartment. *International Journal of Heat and Mass Transfer*, 73:562–573, 2014.
- [69] Ezgi S Oztekin, Ge-cheng Zha, and Richard E Lyon. Flow induced by a small fire in an aircraft cargo compartment. In *50th AIAA Aerospace Sciences Meeting*, number January, Nashville, TN, 2012.
- [70] M. Papadakis, S. Alansatan, and M. Seltmann. Experimental study of simulated ice shapes on a NACA 0011 airfoil. AIAA Paper 99-0096, 1999.
- [71] M. Papadakis, S. Alansatan, and S. Wong. Aerodynamic characteristics of a symmetric NACA section with simulated ice shapes. AIAA Paper 2000-0098, 2000.
- [72] M. Papadakis, S. Alansatan, and H.W. Yeong. Aerodynamic performance of a T-tail with simulated ice accretions. AIAA Paper 2000-0363, 2000.
- [73] M. Papadakis, B. E. Gile-Lafin, G. M. Youssef, and T. P. Ratvasky. Aerodynamic scaling experiments with simulated ice accretions. AIAA Paper 2001-0833, 2001.
- [74] M. Papadakis and B.E. Gile-Lafin. Aerodynamic performance of a tail section with simulated ice shapes and roughness. AIAA Paper 2001-0539, 2001.
- [75] M. Papadakis, A. Rachman, S. Wong, H. Yeong, E. Kuohsing, T. Giao, and C. S. Bidwell. Water droplet impingement on simulated glaze, mixed and rime ice accretions. NASA/TM Paper 2007-213961, 2007.
- [76] M. Papadakis, H.W. Yeong, R. Chandrasekharan, M. Hinson, T.P. Ratvasky, and J. Giriunas. Experimental investigation of simulated ice accretions on a full-scale T-tail. AIAA Paper 2001-0090, 2001.
- [77] O. Reynolds. On the extent and action of the heating surface for steam boilers. *Proc. Manchester Lit. Phil. Soc.*, pages 7–12, 1874.
- [78] P. L. Roe. Approximate Riemann solvers, parameter vectors and difference schemes. *Journal of Computational Physics*, 43:357–372, 1981.
- [79] L. Schiller and A. Naumann. A drag coefficient correlation. *Zeitschrift des Vereins Deutscher Ingenieure*, pages 318–320, 1935.
- [80] H. Schlichting. *Boundary Layer Theory*. McGraw-Hill, 7 edition, 1979.
- [81] R. Schmehl, H. Roskamp, M. Willmann, and S. Wittig. CFD analysis of spray propagation and evaporation including wall film formation and spray/film interactions. *International Journal of Heat and Fluid Flow*, 20(5):520–529, 1999.
- [82] J. Shin. Characteristics of surface roughness associated with leading edge ice accretion. AIAA Paper 94-0799, 1994.
- [83] L. Sirovich. Turbulence and the dynamics of coherent structures, i-iii. *Quarterly of Applied Mathematics*, pages 561–582, 1987.

- [84] S. A. Smolyak. Quadrature and interpolation formulas for tensor products of certain classes of functions. *Soviet Math. Dokl.*, 4:240–243, 1963.
- [85] P. Spalart and S. Allmaras. A one-equation turbulent model for aerodynamic flows. AIAA Paper 92-0439, 30th AIAA Aerospace Sciences Meeting, Reno, NV, January 1992.
- [86] P.R. Spalart. Trends in turbulence treatments. AIAA Paper 2000-2306, 2000.
- [87] B. Sudret. Global sensitivity analysis using polynomial chaos expansion. *Reliability Engineering and System Safety*, 2007.
- [88] L.P. Swiler and G.D. Wyss. A user’s guide to Sandia’s Latin Hypercube sampling software: LHS UNIX library and standalone version. Technical Report SAND04-2439, Sandia National Laboratories, Albuquerque, NM, 2004.
- [89] J. C. Tannehill, D. A. Anderson, and R. H. Pletcher. *Computational fluid mechanics and heat transfer*. Taylor and Francis, Philadelphia, 2nd edition, 1997.
- [90] S. Tatsumi, L. Martinelli, and A. Jameson. Flux limited schemes for the compressible Navier-Stokes equations. *AIAA Journal*, 33(2):252–261, 1995.
- [91] S. Tatsumi, L. Martinelli, and A. Jameson. A new high resolution scheme for compressible viscous flows with shocks. AIAA paper 95-0466, January 1995.
- [92] D.S. Thompson and B.K. Soni. ICEG2D – a software package for ice accretion prediction. Documentation prepared under grant NAG3-2235, available with LEWICE software, 2002.
- [93] P. Tran, M. T. Brahim, I. Paraschivoiu, A. Pueyo, and F. Tezok. Ice accretion on aircraft wings with thermodynamic effects. AIAA Paper 94-0605, 1994.
- [94] M. F. Trujillo, S. Mathews, C. F. Lee, and J. E. Peters. Modelling and experiments of impingement and atomization of a liquid spray on a wall. *International Journal of Engine Research*, 1(1):87–105, 2000.
- [95] Chapter 1 Section 25 US Code of Federal Regulations, Title 14. *Airworthiness Standards: Transport Category Airplanes*. 2016.
- [96] P. Villedieu, P. Trontin, D. Guffond, and D. Bobo. SLD Lagrangian modeling and capability assessment in the frame of ONERA 3D icing suite. AIAA paper 2012-3132, June 2012.
- [97] U. von Luxburg. A tutorial on spectral clustering. *Statistics and Computing*, pages 395–416, 2007.
- [98] X. Wan and G. E. Karniadakis. An adaptive multi-element generalized polynomial chaos method for stochastic differential equations. *Journal of Computational Physics*, 209:617–642, 2005.

- [99] Earl Weener. Lessons from icing accidents and incidents. National Transportation Safety Board (NTSB) Presentation, 2011.
- [100] W. Wright. User's manual for LEWICE version 3.2. Technical Report 214255, NASA/CR, 2008.
- [101] W. B. Wright, T. G. Keith, and K. J. DeWitt. Numerical simulation of icing, deicing, and shedding. AIAA Paper 91-0665, January 1991.
- [102] W. B. Wright and M. G. Potapczuk. Semi-empirical modeling of SLD physics. AIAA paper 2004-412, January 2004.
- [103] W. B. Wright and A. Rutkowski. Validation results for LEWICE 2.0. Technical Report CR 1999-208690, NASA, January 1999.
- [104] D. Xiu. *Numerical Methods for Stochastic Computations: A Spectral Method Approach*. Princeton University Press, 2010.