NUMERICAL SIMULATION OF TWO FLOW CONTROL APPROACHES FOR LOW REYNOLDS NUMBER APPLICATIONS

Daniel A. Reasor Jr.
University of Kentucky, dareas0@gmail.com

Recommended Citation
https://uknowledge.uky.edu/gradschool_theses/461

This Thesis is brought to you for free and open access by the Graduate School at UKnowledge. It has been accepted for inclusion in University of Kentucky Master's Theses by an authorized administrator of UKnowledge. For more information, please contact UKnowledge@lsv.uky.edu.
ABSTRACT OF THESIS

NUMERICAL SIMULATION OF TWO FLOW CONTROL APPROACHES FOR LOW REYNOLDS NUMBER APPLICATIONS

Current research in experimental and computational fluid dynamics is focused in the area of flow control. Flow control devices are usually classified as either “passive” or “active”. Plasma actuators are “active” flow control devices that require input from an external power source. Current efforts have modeled the effects of plasma actuators as a body force near the electrode. The research presented herein focuses on modeling the fluid-plasma interaction seen in dielectric barrier discharge plasma actuators as a body force vector in the region above the embedded electrode using computational fluid dynamics (CFD). This body force is modeled as the product of the gradient of the potential due to the electric field and the net charge density. In a passive flow control study, two-dimensional simulations using CFD are done with a smooth and bumpy Eppler 398 airfoil with laminar, transition, and turbulent models in an effort to improve the understanding of the flow over bumpy airfoils and to quantify the advantages or disadvantages of the bumps.

KEYWORDS: Flow Control, CFD, Plasma Actuators, Bumpy Wings, Low Reynolds Number

Daniel A. Reasor Jr.
July 30, 2007
NUMERICAL SIMULATION OF TWO FLOW CONTROL APPROACHES FOR
LOW REYNOLDS NUMBER APPLICATIONS

By

Daniel A. Reasor Jr.

Raymond P. LeBeau Jr., Ph.D.
Director of Thesis

Lyndon S. Stephens, Ph.D.
Director of Graduate Studies

July 30, 2007
RULES FOR THE USE OF THESES

Unpublished theses submitted for the Master’s degree and deposited in the University of Kentucky Library are as a rule open for inspection, but are to be used only with due regard to the rights of the authors. Bibliographical references may be noted, but quotations or summaries of parts may be published only with the permission of the author, and with the usual scholarly acknowledgments.

Extensive copying or publication of the thesis in whole or in part also requires the consent of the Dean of the Graduate School of the University of Kentucky.

Name

Date
THESIS

Daniel A. Reasor Jr.

The Graduate School
University of Kentucky
2007
NUMERICAL SIMULATION OF TWO FLOW CONTROL APPROACHES FOR
LOW REYNOLDS NUMBER APPLICATIONS

THESIS

A thesis submitted in partial fulfillment of the requirements for the degree of Master of Science in Mechanical Engineering in the College of Engineering at the University of Kentucky

By

Daniel A. Reasor Jr.

Director: Dr. Raymond P. LeBeau Jr., Assistant Professor of Mechanical Engineering

Lexington, Kentucky

2007
MASTER’S THESIS RELEASE

I authorize the University of Kentucky Libraries to reproduce this thesis in whole or in part for purposes of research.

Signed: Daniel A. Reasor Jr

Date: July 30, 2007
Acknowledgments

I would like to thank Dr. Raymond P. LeBeau Jr. for his guidance and instruction throughout the last year and for accepting me as a graduate student when Dr. George P. Huang took the chair position at Wright State University. Dr. LeBeau has been a resource, not only for scientific knowledge, but also for decisions important to life. I would also like to thank Dr. Huang for his guidance throughout my undergraduate education and for inspiring me to pursue advanced degrees. Dr. Suzanne Weaver Smith deserves recognition for giving me the opportunity to develop my skills as a leader in the BIG BLUE Project and for all her guidance in both academics and in life. Dr. Jamey D. Jacob deserves acknowledgment for being my undergraduate advisor, for initiating my interests in aerodynamics, and for initiating my interests in the flow over bumpy wings as a research topic. I would like to thank all my committee members for taking the time to review my thesis and for participating at my defense. Arvind Santhanakrishnan deserves recognition for advise regarding plasma actuators and for providing data for experimental comparisons.

Personally, I would like to acknowledge the support of my family during my education. Without the support of my family, both monetarily and emotionally, my education would not have been possible. I am also grateful for all the friends that have made my experience at the University of Kentucky bearable when school seemed overwhelming.

This work was funded, in part, by the Kentucky NASA EPSCoR through the Kentucky Space Grant Consortium Fellowship.
Table of Contents

Acknowledgments iii
List of Tables vi
List of Figures vii
List of Files xi

Chapter 1 Introduction 1
1.1 Flow Control .................................................. 5
  1.1.1 Active Flow Control ........................................ 6
  1.1.2 Passive Flow Control ....................................... 9
1.2 Objectives .................................................. 10
1.3 Framework of Thesis .......................................... 10

Chapter 2 Background and Previous Work 12
2.1 Introduction to Plasma Discharges .......................... 12
2.2 Introduction to the Physics of Plasma ...................... 15
  2.2.1 Electromagnetic Waves and Plasma ...................... 16
  2.2.2 Debye Shielding and the Debye Length .................. 17
2.3 Experimental Studies of Plasma Actuators .................. 18
2.4 Numerical Studies of Plasma and Plasma Actuators ....... 25
2.5 Characterization of Flow over Airfoils at Low-Re .......... 27
2.6 Flow Control for Low-Re Number Flight Regimes ......... 32
  2.6.1 Numerical Simulations of Low-Re Airfoils with Flow Control ... 36

Chapter 3 Computational Tools 38
3.1 Governing Equations .......................................... 38
3.2 UNCLE .................................................. 39
  3.2.1 Cell-Centered Pressure Method based on SIMPLE Algorithm ... 41
  3.2.2 Rhie and Chow Momentum Interpolation Method .......... 42
  3.2.3 Gauss–Seidel Solver used in UNCLE .................... 42
  3.2.4 Grid Generation for UNCLE ............................. 43
3.3 GHOST .................................................. 44
  3.3.1 TDMA Solver in GHOST .................................... 44
  3.3.2 Grid Generation in GHOST ............................... 45
3.4 Turbulence Models .......................................... 45
3.5 Transition Model ........................................... 48
3.6 Computational Resources .................................... 51

Chapter 4 Implementation of Flow Control with Plasma Actuators in a Navier–Stokes Solver 55
4.1 Mathematical Formulation of Suzen–Huang Model ........ 55
4.2 Additions to UNCLE for Modeling Plasma Actuators .... 60
  4.2.1 Blocking .................................................. 60
  4.2.2 Numerical Scheme Details ............................... 61
  4.2.3 Pseudo-Code Algorithm ................................... 62
  4.2.4 Discretization Details .................................... 62
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.3</td>
<td>Test Case I: Two-Dimensional Quiescent Flow with Linear SDBD Plasma</td>
<td>65</td>
</tr>
<tr>
<td>4.3.1</td>
<td>Numerical Comparisons</td>
<td>66</td>
</tr>
<tr>
<td>4.3.2</td>
<td>Experimental Comparisons</td>
<td>68</td>
</tr>
<tr>
<td>4.4</td>
<td>Test Case II: Two-Dimensional Quiescent Flow with L-PSJA</td>
<td>74</td>
</tr>
<tr>
<td>4.4.1</td>
<td>Experimental Comparisons</td>
<td>78</td>
</tr>
<tr>
<td>4.4.2</td>
<td>Parameter Study</td>
<td>80</td>
</tr>
<tr>
<td>4.4.3</td>
<td>Quantitative Comparisons</td>
<td>88</td>
</tr>
<tr>
<td>4.4.4</td>
<td>Wall Effects</td>
<td>90</td>
</tr>
<tr>
<td>4.5</td>
<td>Concluding Remarks on Plasma Actuators</td>
<td>93</td>
</tr>
<tr>
<td>Chapter 5</td>
<td>Numerical Simulations of a “Bumpy” Eppler 398 Airfoil</td>
<td>95</td>
</tr>
<tr>
<td>5.1</td>
<td>Numerical Setup</td>
<td>95</td>
</tr>
<tr>
<td>5.2</td>
<td>Model Effects</td>
<td>102</td>
</tr>
<tr>
<td>5.3</td>
<td>Lift and Drag Oscillation Study</td>
<td>105</td>
</tr>
<tr>
<td>5.4</td>
<td>Simulation Results for Reynolds Number=25,000</td>
<td>114</td>
</tr>
<tr>
<td>5.4.1</td>
<td>Separation</td>
<td>121</td>
</tr>
<tr>
<td>5.4.2</td>
<td>Vortex Shedding and Lift Oscillations</td>
<td>130</td>
</tr>
<tr>
<td>5.5</td>
<td>Simulation Results for Reynolds Number=200,000</td>
<td>136</td>
</tr>
<tr>
<td>5.6</td>
<td>Concluding Remarks on Bumpy Airfoil Simulations</td>
<td>146</td>
</tr>
<tr>
<td>Chapter 6</td>
<td>Concluding Remarks</td>
<td>147</td>
</tr>
<tr>
<td>6.1</td>
<td>Summary and Conclusions</td>
<td>147</td>
</tr>
<tr>
<td>6.2</td>
<td>Future work</td>
<td>149</td>
</tr>
<tr>
<td>Appendix</td>
<td></td>
<td>152</td>
</tr>
<tr>
<td>Bibliography</td>
<td></td>
<td>162</td>
</tr>
<tr>
<td>Vita</td>
<td></td>
<td>170</td>
</tr>
<tr>
<td>Table</td>
<td>Description</td>
<td>Page</td>
</tr>
<tr>
<td>-------</td>
<td>-----------------------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>5.1</td>
<td>Summary of the number of grid points for each grid used in bumpy and smooth airfoil simulations.</td>
<td>96</td>
</tr>
<tr>
<td>5.2</td>
<td>Results for grid blocking comparisons at $Re=25000$ (laminar).</td>
<td>98</td>
</tr>
<tr>
<td>5.3</td>
<td>Results for grid resolution comparisons at $Re=25000$ (laminar).</td>
<td>100</td>
</tr>
<tr>
<td>5.4</td>
<td>Laminar simulation results for $St_c$ of smooth E398 airfoil at $Re=25,000$ and $Re=200,000$.</td>
<td>115</td>
</tr>
<tr>
<td>5.5</td>
<td>Results for $St_c$ of Bumpy E398 airfoil at $Re=25,000$.</td>
<td>115</td>
</tr>
<tr>
<td>5.6</td>
<td>Results for $St_c$ of Bumpy E398 airfoil at $Re=200,000$.</td>
<td>115</td>
</tr>
<tr>
<td>5.7</td>
<td>$C_l$ and $C_d$ results for smooth airfoil at $Re=25,000$ (laminar).</td>
<td>117</td>
</tr>
<tr>
<td>5.8</td>
<td>$C_l$ and $C_d$ results for bumpy airfoil at $Re=25,000$ (laminar).</td>
<td>117</td>
</tr>
<tr>
<td>5.9</td>
<td>$C_l$ and $C_d$ results for bumpy airfoil at $Re=25,000$ (Trans).</td>
<td>119</td>
</tr>
<tr>
<td>5.10</td>
<td>$C_l$ and $C_d$ results for bumpy airfoil at $Re=25,000$ (SST).</td>
<td>119</td>
</tr>
<tr>
<td>5.11</td>
<td>$C_l$ and $C_d$ results for smooth airfoil at $Re=200,000$ (laminar).</td>
<td>139</td>
</tr>
<tr>
<td>5.12</td>
<td>$C_l$ and $C_d$ results for bumpy airfoil at $Re=200,000$ (laminar).</td>
<td>139</td>
</tr>
<tr>
<td>5.13</td>
<td>$C_l$ and $C_d$ results for bumpy airfoil at $Re=200,000$ (SST).</td>
<td>140</td>
</tr>
</tbody>
</table>
List of Figures

1.1 The regions of the turbulent boundary layer. ........................................... 6
1.2 Classification of flow-control strategies redrawn from Gad-el-Hak[1]. ........................ 7
2.1 Schematic of a simple plasma discharge redrawn from Lieberman [2]. ........................ 14
2.2 Schematic of a linear dielectric barrier discharge plasma actuator. .......................... 19
2.3 Schematic of a synthetic jet and a plasma synthetic jet. ....................................... 24
2.4 Flight regime Reynolds number and flight speed redrawn from Lissaman[3]. .................. 29
2.5 Separation due to an adverse pressure gradient. .................................................. 30
2.6 Schematic of the laminar separation bubble redrawn from Horton [4]. ......................... 31
2.7 MIAV inflatable wings with bumpy NACA 4318 profile. ......................................... 34
2.8 Smoke-wire visualization results for the bumpy and smooth Eppler 398 airfoil at \( Re = 25000, \alpha = 0^\circ \) [5]. .............................................................. 35
2.9 Autonomous flight of an inflatable wing UAV. .................................................... 36
3.1 Left hand rule coordinate system for control volumes used in UNCLE. ....................... 40
3.2 Schematic of the concept of Reynolds decomposition redrawn from McDonough[6]. ........ 46
4.1 Schematic of boundary conditions for uncoupled Equations. .................................... 58
4.2 Demonstration of change in charge density with input voltage. .................................. 59
4.3 Most refined computational grid used in linear SDBD plasma actuator simulations. .......... 66
4.4 Comparison of numerical results for \( \phi/\phi_{\text{max}} \) for a linear SDBD plasma actuator in quiescent flow. ................................................................. 67
4.5 Comparison of numerical results for \( \rho_c/\rho_{c_{\text{max}}} \) for a linear SDBD plasma actuator in quiescent flow. ................................................................. 67
4.6 Comparison of numerical results for \( |f_b|/\phi_{\text{max}} \rho_{c_{\text{max}}} \) for a linear SDBD plasma actuator in quiescent flow. ................................................................. 68
4.7 Comparison of \( u \) velocity profiles to experimental results. .................................. 69
4.8 Comparison of \( v \) velocity profiles to experimental results. .................................. 70
4.9 \( \rho_c/\rho_{c_{\text{max}}} \) distributions for two larger values of \( \lambda_d \). ................................. 71
4.10\( u \) velocity profiles for two larger values of \( \lambda_d \). ............................................ 72
4.11\( u \) velocity profile comparisons for \( \lambda_d=0.0017 \) m, \( \rho_{c_{\text{max}}}=0.000625 \) Cm\(^{-3}\). .... 72
4.12\( v \) velocity profile comparisons for \( \lambda_d=0.0017 \) m, \( \rho_{c_{\text{max}}}=0.000625 \) Cm\(^{-3}\). .... 73
4.13 Computational grid used in L-PSJA simulations. ............................................... 74
4.14 Flow field boundary conditions for L-PSJA simulations in quiescent flow. ................. 75
4.15 Results for normalized electric potential, \( \phi \), and normalized net charge density, \( \rho_c \), for L-PSJA simulations in the vicinity of the electrodes. .......................... 76
4.16 \( |f_b|/\phi_{\text{max}} \rho_{c_{\text{max}}} \) for L-PSJA simulations in the vicinity of the electrodes. ....... 77
4.17 Comparisons of the two different meshes used for L-PSJA computations. .................. 79
4.18 Normalized velocity jet width comparisons for different downstream locations from dielectric surface for L-PSJA in quiescent flow. ........................................... 80
4.19 Vorticity contour comparison with streamlines for L-PSJA in quiescent flow. .............. 80
4.20 $\rho_c/\rho_c^{\text{max}}$ distributions for different values of $\lambda_d$. 82
4.21 Jet comparisons for different downstream locations from dielectric surface for L-PSJA in quiescent flow with six different values of $\lambda_d$ (note: $\lambda_d^{\text{base}}=0.00187$ m). 83
4.22 Normalized velocity jet width comparisons for different downstream locations from dielectric surface for L-PSJA in quiescent flow for $\lambda_d=0.000187$ m. 83
4.23 Normalized velocity jet width comparisons for different downstream locations from dielectric surface for L-PSJA in quiescent flow for $\lambda_d=0.000204$ m. 84
4.24 Normalized velocity jet width comparisons for different downstream locations from dielectric surface for L-PSJA in quiescent flow for $\lambda_d=0.000221$ m. 84
4.25 Normalized velocity jet width comparisons for different downstream locations from dielectric surface for L-PSJA in quiescent flow for $\lambda_d=0.000255$ m. 85
4.26 Normalized velocity jet width comparisons for different downstream locations from dielectric surface for L-PSJA in quiescent flow for $\lambda_d=0.00034$ m. 85
4.27 Centerline velocity comparison for L-PSJA in quiescent flow under steady operation with different values of $\rho_c^{\text{max}}$. 86
4.28 Trend line plot for $V_{\text{peak}}$ for L-PSJA under steady operation with $\rho_c^{\text{max}}=0.0075$ Cm$^{-3}$ and ranging $\lambda_d$ values with linear and exponential best fit lines. 87
4.29 Trend line plot for $V_{\text{peak}}$ for L-PSJA under steady operation with $\lambda_d=0.00017$ m and ranging $\rho_c^{\text{max}}$ values with linear and power best fit lines. 88
4.30 Centerline velocity comparison for L-PSJA in quiescent flow under steady operation with $\rho_c^{\text{max}}=0.00375$ Cm$^{-3}$. 89
4.31 Jet width comparisons at 6 mm and 10 mm from dielectric surface for L-PSJA in quiescent flow under steady operation with $\rho_c^{\text{max}}=0.00375$ Cm$^{-3}$. 89
4.32 Jet width comparisons at 26 mm and 30 mm from dielectric surface for L-PSJA in quiescent flow under steady operation with $\rho_c^{\text{max}}=0.00375$ Cm$^{-3}$. 90
4.33 Normalized half width ($x_{1/2}/d_i$) of jet plotted against normalized distance from the wall $y/d_i$ with $\rho_c^{\text{max}}=0.00375$ Cm$^{-3}$. 91
4.34 $v$ velocity contours and vorticity contours with streamlines. 92
4.35 Centerline $v$ velocity and jet widths at different distances from the surface. 92
4.36 $u$ velocity profiles at different locations from L-PSJA centerline. 93
5.1 Boundary conditions with numbered grids for two-dimensional airfoil simulations. 97
5.2 Three different size structured grids used. 98
5.3 $C_l$ results for grid resolution comparisons at $Re=25000$ (laminar). 100
5.4 $C_p$ curves for the different bumpy and smooth meshes for $Re=25000$, $\alpha=0^\circ$. 101
5.5 $\gamma$ contour for the bumpy airfoil transitional simulation at $Re=25000$, $\alpha=20^\circ$. 103
5.6 $\gamma$ contours for bumpy E398 airfoil at $Re=25,000$, for several angles of attack. 104
5.7 Laminar simulation results for $C_l$, $C_d$ vs. $t^*$ for smooth E398 airfoil at $Re=25,000$. .......................................................... 106
5.8 Laminar simulation results for $C_l$, $C_d$ vs. $t^*$ for bumpy E398 airfoil at $Re=25,000$. .......................................................... 107
5.9 S.–H. intermittency transition model simulation results for $C_l$, $C_d$ vs. $t^*$ for bumpy E398 airfoil at $Re=25,000$. ................. 108
5.10 Turbulent SST simulation results for $C_l$, $C_d$ vs. $t^*$ for bumpy E398 airfoil at $Re=25,000$. ...................................................... 109
5.11 Laminar simulation results for $C_l$, $C_d$ vs. $t^*$ for smooth E398 airfoil at $Re=200000$. ......................................................... 111
5.12 Laminar simulation results for $C_l$, $C_d$ vs. $t^*$ for bumpy E398 airfoil at $Re=200000$. ......................................................... 112
5.13 SST simulation results for $C_l$, $C_d$ vs. $t^*$ for bumpy E398 airfoil at $Re=200000$.113
5.14 $St_c$ vs. $\alpha$ for the two sets of simulations. ............................ 116
5.15 $C_l$ and $C_d$ vs. $\alpha$ for bumpy and smooth airfoils at $Re=25,000$ (laminar). 118
5.16 $C_l$ and $C_d$ vs. $\alpha$ for all simulations at $Re=25,000$. ............ 120
5.17 $C_l$-$C_d$ polar for all simulations at $Re=25,000$. ....................... 120
5.18 $C_l$-$C_d$ polar for all simulations at $Re=25,000$. ....................... 121
5.19 Time averaged streamlines at $Re=25,000$, $\alpha=7^\circ$. ......... 123
5.20 Time averaged results for smooth E398 airfoil at $Re=25,000$, $\alpha=7^\circ$ (Laminar). .......................................................... 124
5.21 Time averaged results for bumpy E398 airfoil at $Re=25,000$, $\alpha=7^\circ$ (Laminar). .......................................................... 125
5.22 Time averaged results for bumpy E398 airfoil at $Re=25,000$, $\alpha=7^\circ$ (Transition). .......................................................... 126
5.23 Time averaged results for bumpy E398 airfoil at $Re=25,000$, $\alpha=7^\circ$ (SST). 127
5.24 Time averaged velocity profiles at $Re=25,000$, $\alpha=7^\circ$ (SST). 128
5.25 $\gamma$ profiles for the bumpy airfoil transitional simulation at $Re=25,000$, $\alpha=7^\circ$. 128
5.26 Time averaged streamlines at $Re=25,000$, $\alpha=20^\circ$. ........... 130
5.27 Time averaged results for smooth E398 airfoil at $Re=25,000$, $\alpha=20^\circ$ (Laminar). .......................................................... 131
5.28 Time averaged results for bumpy E398 airfoil at $Re=25,000$, $\alpha=20^\circ$ (Laminar). .......................................................... 132
5.29 Time averaged results for bumpy E398 airfoil at $Re=25,000$, $\alpha=20^\circ$ (Transition). .......................................................... 133
5.30 Time averaged results for bumpy E398 airfoil at $Re=25,000$, $\alpha=20^\circ$ (SST).134
5.31 Time averaged velocity profiles at $Re=25,000$, $\alpha=20^\circ$ (SST).135
5.32 $\gamma$ profiles for the bumpy airfoil transitional simulation at $Re=25,000$, $\alpha=20^\circ$. ...................................................... 135
5.33 $C_l$ vs. $t^*$ for smooth and bumpy E398 airfoil at $Re=25,000$, $\alpha=10^\circ$ 137
5.34 Streamlines for smooth and bumpy E398 airfoil at $Re=25,000$, $\alpha=10^\circ$. 137
5.35 Vorticity contours overlaid with streamlines for smooth and bumpy E398 airfoil at $Re=25,000$, $\alpha=10^\circ$ .......................... 138
5.36 $C_l$-$C_d$ for bumpy and smooth airfoils at $Re=200,000$. .......................... 140
5.37 $C_l$-$C_d$ polar for all simulations at $Re=200,000$. ....................... 141
5.38 Time averaged velocity profiles at $Re=200,000$, $\alpha=20^\circ$. .......... 142
5.39 Time averaged results for smooth E398 airfoil at $Re=200,000$, $\alpha=20^\circ$ (Laminar) ................................................................. 143
5.40 Time averaged results for bumpy E398 airfoil at $Re=200,000$, $\alpha=20^\circ$ (Laminar) ................................................................. 144
5.41 Time averaged results for bumpy E398 airfoil at $Re=200,000$, $\alpha=20^\circ$ (SST). 145
6.1 Coarse grid and results for Eq. 4.9, 4.14 and $\frac{|f_b|}{\phi_{\text{max}} e^{\text{max}}}$. .................................................. 150
List of Files

DReasor.pdf ................................................................. 47MB
Chapter 1
Introduction

Ludwig Prandtl’s work near the turn of the 20th Century was the beginning of flow control research. The concept of influencing flow via interactions with the boundary layer or the bulk flow is the phenomenon known as flow control and is one of the most heavily investigated fields in fluid dynamics and aerodynamics. Flow control consists of manipulating the flow over an object near the surface in an effort to reduce separation, drag, noise and friction while improving various other characteristics of the flow. The basis of this work is to investigate two types of flow control, passive and active. More specifically, it focuses on numerical simulations of active flow control through the use of dielectric barrier discharge (DBD) plasma actuators and passive flow control for aerodynamic applications through implementation of “large-scale surface roughness” or “bumps” on an airfoil surface.

The equations that describe the motion of fluid are known as the Navier–Stokes (N.–S.) equations after Claude Louis Marie Henri Navier (1785-1836) and George Gabriel Stokes (1819-1903). Mathematicians have tried to prove various aspects such as existence and uniqueness of weak or strong solutions to these equations while scientists and engineers have tried to simplify the equations so that analytical results can be obtained for comparison to experimental results. Presently, no one has derived an analytical solution to the full N.–S. equations, but many solutions exist for simplified flows. These equations are the conservation of mass, momentum and energy and are commonly written in Cartesian tensor notation (or Einstein’s summation notation) in differential form as

\[ \frac{\partial \rho}{\partial t} + \frac{\partial (\rho u_k)}{\partial x_k} = 0 \]  \hspace{1cm} (1.1)

\[ \rho \frac{\partial u_j}{\partial t} + \rho u_k \frac{\partial u_j}{\partial x_k} = -\frac{\partial p}{\partial x_j} + \frac{\partial}{\partial x_j} \left( \lambda \frac{\partial u_k}{\partial x_k} \right) + \frac{\partial}{\partial x_i} \left[ \mu \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \right] + \rho f_i \]  \hspace{1cm} (1.2)

\[ \rho \frac{\partial e}{\partial t} + \rho u_k \frac{\partial e}{\partial x_k} = -p \frac{\partial u_k}{\partial x_k} + \frac{\partial}{\partial x_j} \left( k \frac{\partial T}{\partial x_j} \right) + \lambda \left( \frac{\partial u_k}{\partial x_k} \right)^2 + \mu \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \frac{\partial u_j}{\partial x_i} \]  \hspace{1cm} (1.3)
where \( u_i = (u_1, u_2, u_3)^T = (u, v, w)^T \) is the velocity vector, \( \rho \) is density, \( p \) is static pressure, \( \lambda \) is the second viscosity, \( \mu \) is dynamic viscosity, \( f_i \) is a body force (usually gravity), \( e \) is internal energy per unit mass, \( k \) is thermal conductivity and \( T \) is temperature. The second viscosity is often removed through Stokes’s Relation where the bulk viscosity \( (K, \text{not seen in the form above}) \) is zero and the second viscosity is related to the dynamic viscosity via \( \lambda = -\frac{2}{3}\mu \). The terms associated with the second viscosity may also be removed by introducing the incompressibility assumption where the velocity vector is divergence free (i.e. \( \partial u_i / \partial x_i = 0 \)). Many times, equations of state are used when solving the \( \text{N.-S.} \) equations to address the closure problem (when there are more unknowns than equations); some common equations of state are the incompressibility equation of state where the material derivative of density is equal to zero (i.e. \( D\rho / Dt = 0 \)), the barotropic equation of state where pressure is a function of density (i.e. \( p = f(\rho) \)) and the perfect gas equation of state where pressure is related to density and temperature through the ideal gas relation \( p = \rho RT \). With the use of these equations of state the closure problem no longer exists except for the ideal gas equation of state where there are five equations and six unknowns (unless \( e = C_v T \) is assumed).

For the simulations presented in this work the incompressible assumption is used. Mentioned previously, this assumption yields a divergence free velocity vector, but the strict definition of compressibility is given by

\[
\tau = \frac{1}{\rho} \frac{\partial \rho}{\partial p} = -\frac{1}{\nu} \frac{\partial \nu}{\partial p},
\]

where \( p \) is the pressure and \( \rho \) is the density, and \( \nu \) is the specific volume. In order for the flow to be incompressible \( \tau \) must be small. This implies that the change in density must be small with respect to change in pressure. Many times in fluid mechanics we term fluids that require very large changes in pressure to cause minor changes in density as an incompressible fluid. However, fluid applications that have changes in density, such as stratified flows, can be characterized as incompressible even though \( \tau \) would, strictly speaking, have a significant value. Most of the time compressible flows are characterized by high Mach numbers, where the Mach number is given as the ratio of a local or free stream velocity to the speed of sound in the fluid (i.e. \( M = U/a \)). We will not make use of the Mach number in this work since our flows are incompressible or the Mach number is assumed to be less than 0.3 everywhere in the computational domains.
presented herein.

For the studies presented in this work pertaining to airfoil simulations we will make use of the dimensionless quantity known as the Reynolds number \( \text{Re} \). The Reynolds number is given as

\[
\text{Re} = \frac{\rho U L}{\mu} = \frac{U L}{\nu},
\]

where \( \rho \) is the density, \( U \) is velocity, \( L \) is the characteristic length scale, \( \mu \) is the dynamic viscosity, and \( \nu = \mu/\rho \) is the kinematic viscosity. The Reynolds number has commonly been described as the dimensionless quantity that relates the inertial effects of the flow to the dissipative effects of the flow. In the case of many low-\( Re \) flows the viscous effect is non-negligible and serves to smooth the flow; many low-\( Re \) flows are laminar flows. As \( Re \) increases to a critical value the inertial effects become more important to the flow. These inertial contributions are from the nonlinear terms in the N.–S. equations. Eventually as the \( Re \) increases the flow moves from a steady state flow to a quasi-time dependent flow. As the \( Re \) is increases further, the flow eventually becomes turbulent. The \( Re \) for which the flow becomes turbulent varies with many different parameters such as those that characterize the geometry and the fluid.

Despite being a well known set of equations, the N.–S. equations lack a general analytical solution. Many analytical solutions have been obtained for a small class of problems that we encounter, but are only obtained through simplifications that many deem appropriate (while others may protest to be controversial). Solutions using computational fluid dynamics (CFD) are, in general, the closest and only obtainable approximations to the complete set of equations given above for interesting flows or flows over complex geometries. CFD is the concept of taking the N.–S. equations and through various techniques of linearization and discretization making them solvable on a digital computer through iterative methods that combine both implicit and explicit computations. Furthermore, mechanical engineers, more specifically aerodynamicists and fluid dynamists, commonly use CFD to solve the full N.–S. equations, the incompressible N.–S. equations, and the Euler equations for use in the development of airplanes, automobiles and heat exchangers. Hence, without the use of CFD, many of the commercial and military aircraft, spacecraft and automobiles of today would have never been built, or at least would not be as efficient or as aerodynamic.
Experimentalists commonly try to change the performance of many designs by implementing flow control devices. This process can be relatively easy in a laboratory setting depending upon the application, but data collection and repeatability are many times quite challenging. The use of accurate CFD simulations makes understanding the influences that flow control devices have on the bulk flow or boundary layer clearer than interpretations of experimental results alone. Also, through the use of CFD, there is a large variety of data that can be obtained given correct implementations. This includes but is not limited to: skin friction, pressure coefficients, lift coefficients, drag coefficients, vorticity, velocity, pressure, density, shock locations, expansion fan locations, and boundary layer thickness. Many commercial and government CFD codes have reached a point where their solutions match experimental observations quite accurately. Hence, solutions obtained from CFD simulations with these well-known and verified codes are becoming more and more essential in the development of aircraft. The fact that roughly 100,000 computer hours will go into the CFD simulations used to design the Boeing 787 Dreamliner is an example of its widespread acceptance in today’s aerospace industry.

As previously mentioned, CFD is comprised of the techniques that discretize the N.–S. equations so that they can be solved on a digital computer. The use of numerical schemes to solve discretized versions of complex partial differential equations is not only used in fluid dynamics. Electrical engineers use numerical methods to solve Maxwell’s equations, mathematicians create numerical schemes many times robust enough to solve very general partial differential equations of certain types and physicists rely heavily on numerical calculations for computing solutions to complex partial differential equations such as those found in plasma physics, electro-mechanics and magnetism. Some of the most classical partial differential equations studied numerically include the wave equation, the heat equation, the Schrödinger equation, Laplace’s equation, Poisson’s equation, the Euler–Tricomi equation, the Ginzburg–Landau equation and Burger’s equation. It is common among many commercial CFD codes that they include multiphysics capabilities and some even allow the user to input additional equations that are solved in parallel to the CFD computations or that can be coupled with the flow field properties. The process of solving the N.–S. equations with additional PDEs that at-
tempt to model the physics associated with dielectric barrier discharge plasma actuators in an in-house CFD code is the basis for one of the major studies in this work.

1.1 Flow Control

Early flow control efforts by Prandtl involved implementing surface roughness on a flat plate to trip the flow into transition to turbulence. By tripping the flow into turbulence, the basic structure of the boundary layer is manipulated. The no-slip boundary condition states that the velocity on the surface of an object is zero in both the direction normal and tangential to the surface. The boundary layer consists of the region that starts on the no-slip boundary condition on the surface; the velocity then increases monotonically, constantly increasing, until it reaches the free stream flow except when separation is present. When flow control devices interact with laminar boundary layers they typically do so in two ways: by thinning the boundary layer by means of injecting momentum tangent to the surface or by tripping the flow into transition or turbulence and fundamentally changing the structure of the boundary layer.

Theory pertaining to the turbulent boundary layer has undergone rigorous investigations by numerous physicists and engineers. It is commonly agreed that the structure of the turbulent boundary layer consists of a viscous sublayer where molecular viscosity dominates, a buffer layer or blending region, a turbulent log-law region, and a layer described by the law of the wake. What is understood is that these regions are representative of the boundary layer that forms over a flat plate for fully-developed flow (see Figure 1.1 for a schematic of the turbulent boundary layer structure). The locations of these different regions are commonly described in terms of a dimensionless distance from the wall commonly given as

$$y^+ = \frac{yu_\tau}{\nu}$$

(1.6)

where $u_\tau = \sqrt{\tau_w/\rho}$ is the friction velocity. The dimensionless velocity value is commonly given as a function of $y^+$ (i.e. $u^+ = f(y^+)$) as seen in Fig. 1.1.

In general, flow control devices are classified into two categories, passive or active (see Fig. 1.2). Passive flow control devices do not require or include control systems, external input or feedback loops. Active flow control devices are given a predetermined
setting where they function according to user input and do not necessarily react to the flow. Reactive active devices do possess the capability to adapt depending on flow characteristics and are integrated into more complex control systems that adjust to different flight or flow characteristics. For example, feed forward flow control devices typically receive a signal from a sensor and actuate to modify the flow in a preprogrammed manner. An example of a feed forward application is where the pressure or velocity is measured at an upstream location and the flow control devices actuate according to rules set forward by the designer based on these measurements. In turn, this actuation influences the flow field downstream of the actuator location. Conversely, closed-loop feedback flow control devices typically receive a feed-forward signal from a comparator then send a measured or controlled variable back to a feedback element that communicates with the comparator. These closed-loop flow control devices can control the flow downstream of its location much more efficiently since they get feedback as to how much or little they influence the flow.

1.1.1 Active Flow Control

Gad-El-Hak [1] states that “active airflow control consists of manipulating a flow to affect a desired change”. Many times active flow control devices are used to manipulate flow into transition from laminar to turbulent inside the boundary layer to prevent or to reduce the amount of separation. This elimination or reduction of separation will
reduce the amount of drag, increase the amount of lift, or could eliminate unwanted instabilities that cause vibrations, noise, or other forms of energy loss. It is easily seen why applications of this sort are of prime importance to industry, more specifically aeronautics.

In order to manipulate the flow, there are three main phenomena of interest in all flow control devices. These phenomena are laminar-turbulent transition, separation, and turbulence. Delaying laminar-turbulent transition within the boundary layer has many advantages. In some instances the drag associated with laminar flow may be an order of magnitude less than that of turbulent flows. For aircraft, reduced drag means reduced fuel costs or an improvement in fuel economy, faster flight speeds, and more flight endurance. Other characteristics like the maximum lift coefficient and the stall angle are of significant importance to aircraft design and performance. For instance, increasing the amount of lift on a wing through the use of active flow control devices can drastically reduce the amount of fuel used during takeoff and landing. The lift at a given angle of attack for a given airfoil can be increased by increasing the camber of the airfoil. However, the maximum achievable lift is limited by the ability of a flow to follow the curvature of the airfoil (i.e. it will separate after the camber has reached a certain value). Ways to prevent the flow from detaching from the airfoil is to use
a leading edge slat, trailing edge flaps, or wall jets. All of these devices have been studied extensively, and leading edge slats and trailing edge flaps are still the most common flow control devices seen on commercial aircraft. The last phenomenon of importance is turbulence. An increase in turbulence can lead to greater mixing of the flow. A decrease in turbulence can therefore play a fundamental role in the decrease of aerodynamic noise. In many instances, the delay of the laminar-turbulent transition is done using wall jets. The turbulent boundary layer is much more resistant to adverse pressure gradients that can eventually lead to separation which can be detrimental to the performance of devices such as airfoils or turbine blades. Based on the application, the presence of turbulent flow or a turbulent boundary layer can be good or bad. In the case where separation reduction is desired, a turbulent boundary layer is advantageous. If friction drag reduction is needed then a laminar boundary layer is more attractive.

It is well known that many of the mechanical devices used in active flow control are effective, but they do have drawbacks. In particular, many of the mechanical active flow control devices are relatively complicated, add considerable weight, require an interior volume to be integrated, and are common sources for noise and vibration. Furthermore, they consist of mechanical parts that can wear and can stop functioning after a period of operation. Among the many types of active flow control methods used today, a new and promising technology known more specifically as the dielectric barrier discharge (DBD) plasma actuator do not have many of the drawbacks of current flow control devices. Plasma actuators make use of the discharge-induced electric wind within the boundary layer to modify its properties and thusly modify the airflow. In most cases, these actuators are composed of two electrodes mounted flush to a wall. These electrodes are then supplied a high voltage (e.g. $O(10^4 V)$) resulting in the generation of a relatively cool sheet of plasma. These plasma actuators accelerate the airflow tangentially and very close to the wall where the boundary layer exists. Since they require an electric signal input, they can be easily integrated into complex control systems. These devices operate on time scales orders of magnitude less than common bulk flows which make them advantageous for control systems. Most of the work that has been published on DBD plasma actuators has been experimental, but several models have been developed in efforts to simulate their effects in CFD. Thus, most of the efforts from
both experimentalists and computational scientists have been to try to understand the physics that cause the effects that DBD plasma actuators have on the flow. It is agreed that these effects are due to neutral-electron interactions, but experimental data of this phenomena is difficult to obtain, which has led to the difficulties associated with a comprehensive understanding of this multi-physics system and modeling for use in N.-S. computations.

1.1.2 Passive Flow Control

The effects of surface roughness on the laminar-turbulent transition has gained importance due to the interest in low Reynolds number airfoils for modern aerodynamic applications. Roughness not only affects the laminar-turbulent transition, but also effects pressure gradients, the skin friction and the Mach number. In general, wall roughness favors the laminar-turbulent transition since, under otherwise equal conditions, the transition would occur at smaller Reynolds numbers for a rough wall than for a smooth wall. More precisely, the roughness produces additional relatively large amplitude disturbances in the laminar flow. Results from nonlinear perturbation theory show that the critical Reynolds number for transition is reduced[7]. Roughness has also been observed[8] to change the fundamental slope of the $\alpha$-$C_l$ (angle of attack vs. lift coefficient) curve for a given airfoil.

One of the most common passive flow control devices used today in aerodynamic applications are riblets, or longitudinally grooved surfaces, that have been shown to reduce skin-friction drag on $\mathcal{O}(5-10\%)$ in turbulent boundary layers. These riblets reduce the amount of skin friction drag while simultaneously increasing the surface area. Currently, the exact mechanism by which they reduce drag is still controversial. It is assumed that the reduction of drag is likely to originate in the quiescent regions within the crevices of the roughness[1]. Similarly, the addition of “bumps” on the surface of inflatable wings serves as a large scale version of surface roughness. Previous work has been published on their effectiveness as a passive flow control device[9, 10], but the precise mechanism is still not understood much like that of riblets. It has been observed[10] that there exists a nearly quiescent region within the bumps similar to that observed with riblets. Experimental literature suggests that these pertubations serve as a passive means to thin the boundary layer on the upper surfaces of airfoils.
at relatively low Reynolds numbers\cite{9}. Santhanakrishnan \textit{et al.} \cite{5, 9, 11} suggest that these perturbations or “bumps” on the upper and lower surface yield results similar to that of other traditional (passive and active) flow control devices.

1.2 Objectives

The present effort is the numerical investigation of two flow control mechanisms through the use of two CFD codes. The first flow control device studied is the DBD plasma actuator. The first part of this work will consist of material relevant to the field of plasma actuators as it applies to bulk flow contributions and their interaction with the boundary layer. This will be done through implementing additional equations into the unstructured grid based N.–S. solver titled “UNCLE”. The second half of this work will focus on numerical simulations of the “regular perturbations” or “bumps” on the upper and lower surfaces of an Eppler 398 airfoil. This study will be done using the structured grid based N.–S. solver titled “GHOST”.

1.3 Framework of Thesis

Chapter 2 includes a background into the physics of plasmas focusing on plasma discharges. Then an overview of state-of-the-art research in the area of plasma actuators. Included are discussions on experimental studies of plasma actuators and their applications and numerical simulations of plasma actuators. This will also include a discussion of the experimental investigations of annular plasma synthetic jet actuators (A-PSJAs) or PSJAs. Also included is a summary of work done on aerodynamic analysis of low Reynolds number airfoils including both experimental and numerical studies. Additional discussion pertaining to experimental investigations of “bumpy” wings for low-$Re$ applications will be included.

After presenting a literature review of current trends pertaining to both of the flow control devices, a discussion of the computational tools used for the numerical investigations will be presented. A brief description of each code is included with additional comments on turbulence and transition models previously implemented into the codes that were used in this work. There will also be a brief inclusion of some of the details used to implement the additional equations that attempt to model the physics associated with DBD plasma actuators.
Following the establishment of a framework for this study and computational tools used, we proceed with the specific discussion on the solution procedures developed for the application of plasma actuators into the N.–S. computations in the CFD code UNCLE. This begins in Chapter 4, with a detailed mathematical derivation of the model used to that capture the effects due to the incorporation of a plasma actuator(s) on a flat surface. Included is a formal discussion on how the basic laws of electromagnetics and definitions seen in plasma physics combine to reveal the model used for the computations[12]. A discussion describing the implementation of boundary conditions for the additional equations is also included. Following the framework of the numerical scheme, details of the test cases studied are presented. Numerical results including comparisons with previous numerical computations done with GHOST and experimental data are then presented in detail. Additional results will be presented that are unique to the implementation of this model in the unstructured grid N.–S. solver.

In Chapter 5 we discuss the problem associated with the flow over smooth airfoils at relatively low Reynolds numbers. A presentation of grid studies corresponding to the effects of blocking are discussed. A detailed analysis associated with the simulations of the “bumpy” wings is presented and conclusions are made relevant to their feasibility for low $Re$ applications. Lift and drag results for various angles of attack at different Reynolds numbers are also included with suggestions for their use and comments about the laminar-turbulent transition that is assumed to be present in the flow over these irregular surfaces. This will also include a discussion of temporal effects discussed via Strouhal number analysis.

A conclusion is presented in Chapter 6 with a summary of findings and conclusions drawn from various studies along with proposals for future investigations. Further comments will be made for guidance in implementation of different models for the simulation of plasma actuators and for geometry change considerations for “bumpy” airfoils or inflatable wings.
In this chapter an introduction into plasma discharges will be discussed followed by a section presenting some of the physics associated with plasmas and how they apply to the study of DBD plasma actuators. A survey of current experimental and numerical research discusses various applications and phenomena associated with these devices. Following the discussion of plasma actuators, a discussion on low-$Re$ airfoils will be presented starting in section 2.5 with a survey of literature relevant to experiments, which will include discussion on bumpy airfoils, and numerical investigations of low-$Re$ airfoils with different flow control applications.

2.1 Introduction to Plasma Discharges

For the last few decades non-thermal atmospheric pressure plasmas have been studied for numerous industrial applications such as ozone generation, pollutant removal and surface treatment. These non-thermal plasmas may be produced by a variety of different electrical discharges and have relatively low energy cost because the majority of the electrical energy goes into the production of energetic electrons, not to heating the surrounding gas.

A corona discharge is a weakly luminous discharge, which usually appears at atmospheric pressure near the ends of thin wires or tips of conducting material where the electric field, $\vec{E}$ is relatively large. Corona discharges are typically classified as a Townsend discharge or a negative glow discharge; the classification depends on the magnitude of the electric field and electric potential distribution[13]. Corona discharges generally consist of two electrodes exposed to air at atmospheric pressure and temperature where an AC voltage is applied. Controlling the geometric shape of one of the electrodes can generate an asymmetric electric field where the electric field in the vicinity of one electrode is much greater than the electric field separating the two electrodes. Externally these discharges are seen through a glowing light, plasma, and a loud hissing sound[14]. The effects of the discharge is strongly governed by the shape of the electrodes, the size of the gap separating them, and the gas of the working fluid in the vicinity of operation. Therefore, determining the optimum configuration requires inves-
tigation into a relatively large parameter space. Some applications of corona discharges include the removal of unwanted electrical discharges from the surface of aircraft in flight to eliminate their influence on avionics systems, manufacturing of ozone ($O_3$), scrubbing particles from air in HVAC systems, removing unwanted volatile organics such as pesticides, treating surfaces of polymer films for printing applications, and photocopying. Other applications include electro-hydrodynamic thrusters (EHDs), lifters, and ionic wind devices. If the electric field is strong near the cathode then it is a negative corona discharge. Conversely, if the electric field is strong near the anode then it is a positive corona discharge.

Arc discharges generally consist of an electrical breakdown of a gas producing an ongoing plasma discharge, resulting from a current flowing through a normally non-conductive media; this many times includes noble gases and air at atmospheric pressure and temperature. These arc discharges are produced in the gap between two electrodes. When at high temperature, the plasma is capable of melting or vaporizing many industrial materials such as steel. Therefore, these arc discharges can be used for welding, plasma cutting and electrical discharge machining. When the temperature or pressure is reduced, these electric arcs are used for lighting, plasma screen displays, camera flash lamps, and neon signs.

Glow discharge plasma forms by passing a current at 100-1000V through a gas, typically noble gases. The simplest type of glow discharge is a direct-current (DC) glow discharge. A typical device to generate this type of discharge consists of two electrodes in a low pressure enclosure, filled with a noble gas, with a potential of several hundred to thousands of volts applied between the two electrodes. Within the enclosure, a relatively small population of atoms is ionized via random collisions. The ions are driven toward the cathode by the electric potential, and the electrons are driven toward the anode by the same potential difference. The initial population of ions and electrons collide with other atoms, and the result is ionization. If the potential is held constant at a sufficient level, a population of ions and electrons will remain to populate the enclosure. This process is typically found in devices such as fluorescent lights, plasma-screen televisions and in analytical chemistry applications.

Plasma actuators describe a broad set of devices based on using atmospheric pres-
sure electrical discharges. This set of discharges may include corona discharges, glow discharges, and dielectric barrier discharges (DBDs). A schematic of a simple plasma discharge can be seen in Fig. 2.1. It consists of a voltage source that drives current through a low pressure gas between two parallel conducting plates or electrodes. The gas “breaks down” to form a plasma, usually weakly ionized. The term weakly ionized refers to the plasma density being only a small fraction of the gas density. The formation of these plasma discharges is due to what is known as the Townsend mechanism, or an electron avalanche which corresponds to the manipulation of some primary electrons in cascade ionization. The electron avalanche develops due to the multiplication of electrons proceeding along their drift or path from the cathode to the anode. A discharge current is then created due to this phenomenon. The discharges primarily used for airflow control are usually atmospheric pressure corona discharges and DBDs.

![Figure 2.1 Schematic of a simple plasma discharge redrawn from Lieberman [2].](image)

DBD plasma actuators, of specific interest herein, typically consist of an asymmetric arrangement of two electrodes, one exposed to the atmosphere and the other embedded in a dielectric that separates the two. The input of high AC voltage at high frequency causes a region of DBD plasma in the interfacial air gap above the embedded electrode to form. This plasma region drives the residual fluid in the form of a horizontal wall jet created from the working fluid. This horizontal wall jet is controlled by the input voltage amplitude and multiple frequency inputs. Two fundamental frequencies are used in DBD plasma actuator devices; they include the actuation frequency $f_{ac}$ and the pulsing frequency $f_p$. The actuation frequency is the high frequency corresponding to the high voltage AC input that typically range from the high hundreds to high
thousands of Hertz. The pulsing frequency describes the low frequency in which the fundamental high frequency is on during operation; these frequencies typically range from 1-100Hz. Another temporal input is the duty cycle. The duty cycle describes the amount of time the high voltage input at the actuation frequency is on (typically ranging from 5-50%). Both the intensity and spread of the jet is influenced by this input as well as the plasma intensity. Plasma actuators can be readily employed as active flow control devices, and have been shown to control boundary layer separation through their addition of near-wall flow momentum[16].

2.2 Introduction to the Physics of Plasma

As Lieberman states[2]: “A plasma is a collection of free charged particles moving in random directions that is neutral (electrically) on average.” Chen[17] states that “a plasma is a quasi neutral gas of charged and neutral particles which exhibits collective behavior”. The plasmas of interest in the discussion of DBD plasma actuators are those known as weakly ionized plasma discharges.

Many of the equations that define the physics of plasma are commonly seen in electrodynamics, more specifically, the part of electrodynamics that deals with electro-motive forces. The application of Ohm’s law applies to the forces that are sometimes significant in the vicinity of the region where plasma exists. In this introduction a description of the state known as “plasma” will be given with the addition of some fundamental concepts that are the basis of understanding how they apply to DBD plasma actuators.

To make charges move in a conductive material, they have to be “pushed” or “pulled”. The speed in which these particles move is strongly dependent on the properties of the material in which they are located. For most substances studied in electrodynamics this is expressed as

\[ \vec{J} = \sigma \vec{f}. \]  

(2.1)

In this relationship the current density is given as \( \vec{J} \) and is proportional to the force exerted per unit charge, \( \vec{f} \). The proportionality factor \( \sigma \) is an empirical constant that varies from one material to another; it is commonly called the conductivity. Many express the conductivity as the resistivity which is its reciprocal (i.e. \( 1/\sigma \))[18]. In
Eq. (2.1) the current density is expressed as a function of the force exerted per unit charge. This equation is often times seen as the following expression that contains both the electric and magnetic field:

$$\vec{J} = \sigma (\vec{E} + \vec{v} \times \vec{B}).$$

(2.2)

In this relationship the electric field is $\vec{E}$, velocity is $\vec{v}$, and the magnetic field is $\vec{B}$. For most electrical applications the cross product that involves the velocity and the magnetic field is ignored, but for plasma applications this is not the case. If this second term is ignored Eq. (2.2) is commonly referred to as Ohm’s Law. These preceding equations are the basis for the study of the effects of plasma actuators. While a calculation of the individual molecules is the precise way to formulate the effects due to a cool plasma formation or a plasma discharge, it is very expensive computationally as it requires simulations including dense particle tracking which can be exceedingly demanding.

### 2.2.1 Electromagnetic Waves and Plasma

In general, free electrons in conductive media are not bounded to any particular atom or molecule. Rather, they can move about within the material. Under these conditions, many of the same models seen in electromagnetic waves still apply[18]. For instances associated with a dilute plasma or ionized gas, the damping associated with the electromagnetic waves is negligible, and the proportionality factor, $\sigma$, is purely imaginary. The proportionality factor for plasma applications is given as

$$\sigma = i \left( \frac{N f q^2}{m \omega} \right).$$

(2.3)

where $N$ is the number of molecules per unit volume, $f$ is the number of free electrons per molecule, $q$ is an elementary charge, $m$ is mass of a molecule, and $\omega$ is the wave frequency. Under the assumption that the permittivity of the medium is approximately the relative permittivity of the medium where the plasma exists (i.e. $\varepsilon \cong \varepsilon_o$), then general equation that describes the wave number is given as

$$\kappa^2 = \varepsilon \varepsilon_o \omega^2 + i \sigma \varepsilon \omega.$$  

(2.4)

As previously mentioned, the proportionality factor is purely imaginary for plasma applications. Therefore the Eq. (2.4) can be rewritten as

$$\kappa^2 = \frac{1}{\varepsilon^2} \left( \omega^2 - \omega_p^2 \right),$$

(2.5)
where
\[ \omega_p = q \sqrt{\frac{N_f}{m \varepsilon_o}}. \] (2.6)

Equation (2.6) is known as the plasma frequency. If the wave frequency is below \( \omega_p \) then the plasma is opaque. For these frequencies the wave number is purely imaginary from Eq. (2.5). The remainder of this discussion assumes that the operating frequencies of a given plasma are well below that of the plasma frequency. Experimental observation [16] has suggested that the effects of plasma actuators are most prevalent when the plasma density is opaque and the plasma intensity is relatively large. Plasma frequencies are commonly seen to be \((10^6 - 10^9)\)Hz whereas the input frequencies for the AC voltage sources used to drive the dielectric barrier plasma actuators discussed in following sections range from the upper hundreds to the high thousands of Hertz. Since the operating frequency of the input voltage is at least two orders of magnitude less than that of the plasma frequency, this assumption seems appropriate.

2.2.2 Debye Shielding and the Debye Length

Chen [17] states that Debye shielding is the ability of a plasma to shield out electric potentials that are applied to it. It is of importance to introduce this because the formation of the plasma sheath and the distribution of plasma is found to be directly related to the effectiveness of plasma actuators. Lieberman[2] defines the Debye length as the distance scale in which charge densities can exist spontaneously. The Debye length as a measure of the shielding distance or the thickness of the plasma sheath[17]. The Debye length is commonly given as
\[ \lambda_d \equiv \left( \frac{\varepsilon_o K T_e}{ne^2} \right)^{1/2}, \] (2.7)

where \( n \) is the number density, \( e \) is the elementary unit charge of one electron, \( \varepsilon_o \) is the permittivity of free space, \( K \) is Boltzmann’s constant \((K=1.38 \times 10^{-23}J^oK)\), and \( T_e \) is the electron temperature in Volts. From Eq. (2.7) we can see that as the \( \lambda_d \) increases the density \( n \) must decrease; this corresponds to the number of electrons relative to the sheet of plasma decreasing per unit volume, assuming that \( KT_e \) remains relatively constant. Conversely, as the density is increased the Debye length is decreased; this corresponds to each layer of plasma having more electrons. One of the criteria for an ionized gas to be a plasma is that it be dense enough that \( \lambda_d \) is much smaller than the
characteristic length of the system (i.e. $\lambda_d \ll L$). The characteristic length of a typical system, such as a dielectric barrier discharge, containing plasma would be on the order of an electrode length (i.e. $O(L_e)$). The numerical model used later in this work will make use of the Debye length as a parameter that dictates where significant charge density exists.

2.3 Experimental Studies of Plasma Actuators

One of the main advantages that the plasma actuators have is that they directly convert electrical energy to mechanical energy without the use of moving mechanical parts. They also have a very short response time which makes them an attractive device for high frequency control systems. However, the process of converting the electrical energy to mechanical energy is not very efficient[15]. Many have argued that the functionality of plasma actuators is based on the electric wind. As stated by Robinson[19] “the phenomena variously known as the electric wind, corona wind and electric aura refers to the movement of gas induced by the repulsion of ions from the vicinity of a high voltage electrode”. This was reported for the first time in 1709 by Hauksbee and the first explanation was given by Faraday in 1838. The electric wind is therefore due to the collisions between the ions that drift and the neutral particles in the electrode gap region. The electron velocities are much higher than the ion velocities, but the the role of the electrons is typically assumed negligible due to their small mass relative to the neutral particles[15].

Much of the current research in plasma actuators is done with linear single dielectric barrier discharge (SDBD) plasma actuators. Figure 2.2 shows the basic setup for a plasma actuator of this type. As previously mentioned, this device is composed of a set of electrodes slightly offset, separated by a thin dielectric material. The opaque plasma forms in the region above the embedded electrode, but does not exist in the region beyond the embedded electrode and does not extend, vertically, above the thickness of the exposed electrode. The opaque plasma is formed above the embedded electrode region as a result of the series of discharges as electrons are transferred to and from the dielectric surface. However, the presence of significantly charged particles can exist outside of the opaque plasma region undetected by visual spectrum photography.

Dielectric barrier discharge plasma actuators have a large number of industrial ap-
Applications because they can operate in air at atmospheric pressure and do not need a sophisticated pulsed power supply. It should be noted that the efficiency and effectiveness of SDBD actuators has shown to increase when an additional pulsing frequency is added to the high frequency input power source. They may be excited by a sinusoidal high voltage input delivered by a single transformer.

In the late 1990’s Roth’s research group perfected and developed a new atmospheric pressure dielectric barrier discharge plasma actuator[20, 21]. It was a surface DBD that established itself in air between at least two electrodes placed in asymmetrically on each side of a dielectric material. Roth et al. [21] shows that the discharge induces a secondary airflow of several meters per second tangentially to the wall, such that the resulting force increases with the applied voltage up to a maximum value. This secondary airflow can modify the free stream, resulting in drag modification.

In Enloe et al. [22], the researchers take the light emissions from the plasma as a surrogate for the plasma density (or the electron density). The authors then clarify that “using light emission to infer plasma density assumes that the recombination time of the plasma is short compared to the timescale of the discharge, but this assumption is confirmed by the observations and is consistent with plasma lifetimes reported in the literature”[23]. They verify that the discharge ignites near the edge of the exposed
electrode and that the extent of the plasma in the chordwise direction increases in time until the discharge quenches, at which point the AC voltage is at a local maximum. They also confirm that near the edge of the exposed electrode, the plasma density generally increases in time.

Post et al. [24] demonstrates that a DBD plasma actuator can thin the boundary layer, reduce the amount of separation, increase lift and can increase the stall angle of an airfoil. In the experiment described in detail in [24], the plasma actuators were mounted at \( x/c \approx 0.8 \) which was seen as an optimum location determined from previous experimental results. This was the furthest downstream location where the actuators could be mounted without weakening the trailing-edge of the wing. The findings from this experiment were that the amount of lift that the plasma actuators generate varied linearly with the dissipated power of the plasma actuator, the lift improvements also varied linearly with increases in low Reynolds numbers, and that they have a dual effect on lift which includes addition of momentum to the flow and flow control interaction in the existing viscous part of the flow field. The results of the plasma actuators were also compared to that of equivalent wing flaps. The authors note that the plasma actuator employs a constant force or moment rather than the constant lift coefficient of a flap. Since plasma actuators employ a constant force, it may have advantages over the wing flaps in control development since the designer would be able to develop a controller based on a constant force rather than a constant lift or moment coefficient.

There have also been more focused experiments[25, 26] on the effects that plasma actuators have on the flow over airfoils. As mentioned previously, the ability to be integrated on complex surfaces is an attractive feature of plasma actuators and example of such an application is a NACA 0015 airfoil. In the experiments of Post et al. [25, 26] a standard NACA 0015 airfoil is stalled at the relatively high angle of attack of 16° without the use of actuators, but with the implementation of the actuator on the leading edge the flow tends to be more streamlined. The fact that the airfoil without the plasma actuator is stalled implies that there is an adverse pressure gradient above the top surface, in this case, beginning at \( \sim 30\%c \) and propagating to the trailing edge. The airfoil with the plasma actuator shows significantly less separation. The reduction of separation above the upper surface tends to increase the amount of lift generated by
the airfoil up to the stall angle. It also delays the drop off in the lift coefficient that is
typical when increasing the angle of attack past the stall angle. Another result is that
the stall angle of this airfoil was able to be increase from 14° to 18°. In essence, the
actuator was able to deter the dynamic stall vortex and eliminated the sharp drop in
the lift coefficient when the airfoil began to stall.

The incorporation of plasma actuators at various locations on a flying-wing UAV
such as that mentioned in Patel et al. [27] increases the lift coefficient in various ranges of
angles of attack. This research concludes that the incorporation of plasma actuators on
this particular wing results in the most relative improvement of lift coefficients at high
angles of attack. The researchers also concluded that a continuous shift in the lift curve
can be obtained for a wide range of angles of attack by using a system of distributed
plasma actuators. Furthermore, the incorporation of plasma actuators serve as a means
to replace conventional control surfaces that would be integrated on the trailing edges
of the 1303 UAV. At high angles of attack, the actuator is most effective when used at
the leading edge slightly on the windward surface, in close proximity to the separation
point[27].

Plasma actuators have also been incorporated into geometries other than that of
wings or airfoils. One current geometry of interest is the flow over cylinders as it
applies to wheels and other blunt bodied objects. One of the main objectives of current
research[28, 29] is to reduce landing gear noise for commercial transport aircraft through
streamlining the flow over the landing gear. By integrating plasma actuators around
the downstream surface of a cylinder, the amount of vortex shedding can be reduced.
However, at higher Reynolds numbers the authority of the stationary plasma actuators
is lost; this result could lead to the research of mobile actuators that adjust position
based on Reynolds number.

Not only have there been investigations of the effects that plasma actuators have
on the flow field in the vicinity of the electrodes and their effects downstream of the
actuator, but there have also been rigorous investigations into the effects that the wave-
form plays in the effectiveness of the actuator[30, 31, 32]. This research deals with the
amount of dielectric barrier discharge on the surface above the embedded electrodes.
The papers also discuss the effects due to the ion-induced secondary electron emis-
isions. These types of investigations discuss in greater detail the effects that the large voltage sources have on the flowing media and in a sense obtain results that describe the phenomena that cause the electrical hydro-dynamic force on the surface above the embedded electrode.

It is apparent that there have been numerous investigations of the effects that SDBD plasma actuators have on the boundary layer, but Enloe et al. [33] sought to investigate the effects of the AC input frequency on the effectiveness of the SDBD actuator more rigorously. During typical operation of plasma actuators the input signals are either sinusoidal, sawtooth or square waves that vary in frequency from 1-10 kHz. Furthermore Enloe et al. applied low frequency duty cycles of 2% to 50% of the common input frequency. What was shown was that this low frequency pulsing caused a “backward stroke”, a “quenching”, and a “forward stroke”. When the duty cycle is increased to its largest value, namely 50%, it was shown that the SDBD was seen as a continuous source of heat and momentum in terms of the bulk fluids response. It was also shown that the force that the plasma actuator applied to the flow was strongly proportional to the input frequency (i.e. \( f_b = C f \)). Also in this study, it was concluded that the momentum imparted to the fluid in any single discharge, per pulse, depends on the characteristics of the applied voltage waveform, that, in turn, affect the structure of the plasma morphology itself.

Experiments have been done to investigate the effects of plasma actuators on lift enhancement and roll control. These actuators have been mounted near the trailing edge, where traditional control surfaces would be mounted. The application of plasma actuators could be beneficial on applications where control surfaces can be restricting. For example, current research in UAV’s has been done using inflatable wings. The requirement to package these wings in a compact structure before deployment makes plasma actuators an attractive means for roll control[34]. Traditional ailerons cannot be easily folded into a compact enclosure unlike a plasma actuator.

The use of synthetic (zero-net mass flux) jets are a popular topic of flow control research which began before the widespread popularity of plasma actuators. A basic schematic of a synthetic jet actuator is given in Fig. 2.3a. In Glezer et al. [35] the effects of the synthetic jet are examined in quiescent and cross flow regimes. A synthetic jet
is produced by the interactions of train vortices that are formed by the alternating motion of a diaphragm inside an orifice below the surface of interest. Since the fluid used for the jet comes from the residual fluid in the flow, the devices net mass flux is zero. These jets can be produced to interact with flows on different length and time scales. However, since these devices require an orifice within the device in which they are mounted, they are difficult to mount where geometric tolerances are limited.

An effort to reproduce the effects of synthetic jets with cascades of annular dielectric barrier discharge plasma actuators is a current topic of research for various flow control applications. Experimental studies have shown that the amount of force due to a single plasma actuator cannot produce the same effects as that seen in the aforementioned synthetic jets made popular by Glezer. However, by cascading and sequencing annular plasma actuators, similar flow fields have been produced[16]. Although research thus far has not demonstrated that plasma synthetic jets can produce jet velocities of the same magnitude as synthetic jets, they have been shown to be much greater than those of single linear plasma actuators. Current research at Oklahoma State University[36], the University of Kentucky[37, 38], and the National Institute of Advanced Industrial Science and Technology in Tsukuba, Japan with the University of Nottingham, UK [39] focuses on the development of these annular plasma actuators/plasma synthetic jets. A basic schematic of a side view of a plasma synthetic jet is given in Fig. 2.3b.

The premise of developing a plasma synthetic jet actuator (PSJA) is to combine the useful effects seen in research dealing with synthetic jets and that of DBD plasma actuators. Most of the research pertaining to PSJAs was done with quiescent flow experiments. Within the research presented on this topic[16] vortex ring structures were studied while varying the pulsing frequency of the input AC voltage. Fundamentally different results were obtained from the different pulsing frequencies that range approximately from 1-100 Hz. The basic action of the PSJA is to attract the fluid adjacent to the surface. In doing so, the residual fluid is ejected outward normal to the surface in the form of a jet. The formation of this jet is composed of different vortex structures that include fundamental vortex structures that align with the jet and tertiary vortex structures that form above the actuators. It was found in Santhanakrishnan et al. [16] that the longevity of the jet was dependent on the presence of the starting vortex.
Figure 2.3  Schematic of a synthetic jet and a plasma synthetic jet.

(a) Synthetic jet redrawn from Glezer[35].  (b) Plasma synthetic jet.

ring that is primarily a function of the pulsing frequency of the PSJA. The streamwise extent of the jet was controlled by the interactions of the vortex rings and the strength and uniformity of the plasma itself.

Santhanakrishnan et al. [37] discusses the use of linear plasma synthetic jets (L-PSJAs) and PSJAs for flow control. In this paper several test cases are discussed relevant to L-PSJAs and PSJAs in quiescent and in crossflow under steady and unsteady actuation. A detailed summary of streamwise variation in local axial velocity for both devices is discussed with additional discussion on the influences of these devices in three different cross flow configurations over a flat plate. In this study it was found that the PSJA penetrates the cross flow more effectively that the L-PSJA. Additional studies of the influences of PSJAs and L-PSJAs have on cylinder flow are also discussed. Data from this study will be used in comparisons to computational results to follow. Santhanakrishnan et al. [38] discusses SDBDs as background into the discussion of PSJAs and L-PSJAs. Results from this work will be used in comparisons of SDBD plasma actuators under steady operation in quiescent flow in the chapter discussing the results from the computations with a SDBD plasma actuator.

Bolitho et al. [36] discusses various arrangements of PSJAs for aerodynamic flow control. This study combines various PSJA arrangements using both blowing and
sucking to increase the effects beyond that of a single PSJA. In this study it was found that by combining PSJAs of various size the jet width and streamwise velocity magnitude can be increased significantly. Future work is being done on these types of arrangement to increase the effectiveness of these devices to the level that many traditional synthetic jets has reached.

2.4 Numerical Studies of Plasma and Plasma Actuators

There have been several efforts to adequately describe the phenomenon associated with fluid plasma interaction in CFD codes and codes that monitor the ion-neutral interaction. Orlov et al. [40, 41] uses a space-time lumped-element circuit model to simulate the aerodynamic plasma actuator if the volume of the plasma was known for the particular applied voltage conditions. This model has predicted that the power dissipated in the plasma resistive element increases with the $7/2$ power of the applied voltage. This was in agreement with Enloe et al. [42] and Post[24], which showed that induced thrust and maximum velocity generated by the asymmetric electrode arrangement of the SDBD plasma actuator varied with the input voltage (i.e. $V^{7/2}$). This method’s intention was to model the ionization process to provide predictions of the body force for a range of parameters that are a function of the input voltage and frequency.

In He et al. [43], an investigation of flow separation control over a wall-mounted hump model and its control using a linear DBD plasma actuator was studied. Presented in this paper was a brief discussion of the model developed by Orlov and Corke[40]. The test case was the hump model chosen from the 2004 NASA Langley CFD validation workshop since numerous experimental and numerical data was obtainable. The numerical study was done by implementing Orlov’s model into Fluent, the commercial CFD code, and was tested with various RANS turbulence models to see which most accurately predicted the flow fields and skin friction. When the plasma actuator model was implemented, it demonstrated the ability to control the separation over the hump.

Boeuf et al. [31] state that the force acting on the neutral gas in a DBD actuator is due to electron-molecule and ion-molecule collisions. They also state that the force per unit volume on the gas molecules is equal to the momentum transferred per unit volume and per unit time from charged particles to neutral particles. This statement is consistent with the previous explanations for the basic principles for which DBD
actuators work. In this work the authors decompose the force generated into two distinct forces: one based on ions, one based on electrons.

Font et al. [32] uses a particle-in-cell and a Monte-Carlo (PIC-DSMC) method to numerically simulate the interaction of negative and positive ions with a fluctuating electric field. They conclude that if pure oxygen is used in the simulation, the negative oxygen molecules change the force production from that previously seen with pure nitrogen simulations and experiments. The negative oxygen molecules diminish the net force from the generated plasma, but still give better results than that simulated with pure nitrogen and also demonstrate better results from experiments conducted with pure nitrogen. Their experimental results also demonstrate that the force is greater for pure oxygen that for pure nitrogen.

Roy et al. [44] developed a two-dimensional three-species collisional plasma-sheath model for asymmetric DBD plasma actuators. In this model they solved equations for charge continuity, the charge momentum, and a potential based on Poisson’s equation for pure helium. However, in a later study[45] they use an asymmetric DBD model for real gas using eight species. This is done with a self-consistent multi-body system of plasma. The equations governing the motion of charged neutral species are solved with a Poisson equation finite element solver making use of a Galerkin weak formulation. In this study, a separate model is used for both nitrogen and oxygen with equations taking the form of Poisson’s equation for the electrons, \( N, N_2, N_2^+, O_2, O, O^-, O_2^+ \), and the potential \( \phi \), where \( \phi \) is defined by the relationship for the electric field, \( \vec{E} = -\nabla \phi \). The results for the time-averaged streamwise force show that most of the acceleration is above the DBD actuator, but there is also a small decelerating force downstream of the powered electrode and induces a fluctuation in the time dependent or temporal evolution of the streamwise velocity.

Shyy et al. [46] develops a model that does not explicitly account for the chemistry seen in other models. The model discussed is developed for DBD plasma actuators and includes the use of the electric field computed between two electrodes, a body force calculated that accounts for the net charge density and a constant that accounts for collision efficiency. This formulation also accounts for the change in amplitude and frequency of the applied voltage. This body force is then included in the N.–S. equations
as a source term. This article concludes that as the frequency and the amplitude of the voltage source are increased so is the body force in the vicinity of the electrode. The results also conclude that the model generates a jet that is up to five times the free stream velocity for small free stream values and roughly equal to the free stream velocity for values of $10\text{ ms}^{-1}$. Studies were conducted for input voltages of 3, 4 and 5 kV and 2, 3, 4, and 6 kHz with free stream velocities ranging from 2-10 ms$^{-1}$.

In Suzen et al. [12, 47], a two equation model was developed based on the basic principles of Maxwell’s equations. One of the equations in this model is in the form of Laplace’s equation that models the potential due to the electric field which is used to simulate effects of the input voltage (i.e. $\nabla \cdot (\varepsilon \nabla \phi) = 0$). The second equation attempts to model the net charge density above the embedded electrode by solving an equation similar to Poisson’s equation (i.e. $\nabla \cdot (\varepsilon \nabla \rho_c) = \rho_c / \lambda_D^2$) which includes an archival value for the Debye length (the length scale of the plasma). In these papers, a prescribed boundary condition for the net charge density is assumed above the embedded electrode in either a full or half-Gaussian distribution. This model was used to simulate the effects of separation reduction due to the addition of plasma actuators on the surface of low-pressure turbine blades.

In 2007, Suzen et al. revised the model previously discussed to make it more applicable to more complex geometry. In this model[48] a prescribed distribution of charge density is no longer used above the embedded electrode. Instead, the embedded electrode is assumed to be a source for the net charge density. The same equation used in the previous model is used assuming the Debye length of the plasma in the dielectric is relatively large making the source term in the second equation essentially zero inside the dielectric. This model will be discussed in further detail later in the chapter discussing the results of the numerical simulations and will be the basis of the computations with DBD plasma actuators to follow.

2.5 Characterization of Flow over Airfoils at Low-$Re$

At the sharp trailing edge of an airfoil, the flow changes dramatically. Due to viscous effects, the air is unable to flow around the sharp trailing edge of a typical airfoil. Most often, we see a vortex formation at this trailing edge which is many times called a starting vortex. The stagnation point in the flow moves toward the trailing edge and in
this process the lift increases progressively. The circulation around an airfoil increases until the flows from the upper and lower surfaces combine at the trailing edge. Thus, the amount of circulation around a wing and the resultant lift are initiated by the starting vortex which is due to the effects of viscosity\cite{viscosity}. This fundamental relationship is what we observe when we see lift and drag data for many airfoils. At high chord-based Reynolds numbers ($Re_c = \frac{Uc}{\nu}$) this phenomenon happens so quickly in time that we often do not recognize the actual phenomenon associated with lift. However, at low Reynolds numbers we can see this phenomenon more readily which makes flow over low Reynolds number airfoils interesting. Commercial aircraft that transport passengers and cargo typically fly in a flight regime that is characterized by Reynolds numbers in the millions to tens of millions ($10^6$-$10^7$). Therefore, the flow over the wings of these large aircraft is mostly turbulent and does not exhibit many of the effects just mentioned.

Traditional approaches have been to change the surface to encourage attached flow. Reduction of the leading edge radius on airfoils is an example of such a change in surface shape. Other attempts to prevent or delay separation have been accomplished by moving surfaces. Such an example is morphing upper surfaces, oscillating diaphragms, and movement of walls tangent to the flow field such as a belt. If the wall tangent to the free stream velocity moved at the same velocity as the wall, the boundary layer could, theoretically, be eliminated altogether. Flow in the boundary layer can be accelerated by blowing a plane jet through small orifices that directly interact with the external flow field. Vortex generation by means of actuated diaphragms (such as those seen in synthetic jets) in such orifices has also been seen as an effective means to provide energy to the boundary layer and to inject momentum to accelerate the retarded flow in the boundary layer. Suction has been proved as an effective means to delay separation by assisting in maintaining attached flow. However, the implementation of such devices can be cumbersome for complex geometries.

Low Reynolds number aerodynamics is of important interest for micro-aerial-vehicles (MAVs) and unmanned-aerial-vehicles (UAVs). Flight regimes typical of UAV flight are seen in Fig. 2.4 which are characterized by chord based Reynolds number, $Re_c$, and the flight speed. In this figure we can see that UAVs or “model airplanes” can be
characterized by Reynolds numbers ranging from $10^3$ to $10^5$ and by flight speeds in the 1-100 ms$^{-1}$ range. Many of these UAVs operate in high and low altitudes where the Reynolds number can vary by orders of magnitude because of the increase in kinematic viscosity, $\nu$, that results from a decrease in density, $\rho$. In high altitudes the Reynolds numbers can be quite low (e.g. 25000); this introduces the need for low-$Re$ airfoils. When the Reynolds number is low the viscous effects are large. This can cause an increase in drag relative to higher $Re$ flows and can also limit the amount of lift for a given airfoil. According to Lissaman[3] relatively rough airfoils tend to perform better at lower Reynolds numbers than smooth ones. For airfoils operating at $Re$ of $O(10^6)$, the adverse pressure gradient that forms on the upper surface of the airfoil occurs after transition to turbulence. Since a turbulent boundary layer is much more resistive to adverse pressure gradients than a laminar boundary layer, its presence can help prevent separation. Furthermore, a low-$Re$ airfoil must be designed in such a way that it can either resist separation via its fundamental geometry or by other means such as a flow control device.

![Flight regime Reynolds number and flight speed redrawn from Lissaman[3].](image)

Figure 2.4  Flight regime Reynolds number and flight speed redrawn from Lissaman[3].

Separation happens relatively close to the wall where the no-slip boundary condition is present. Therefore, the dominant terms in the N.-S. equations near the wall (where
the velocity components are small) are the pressure gradient term and the shear stress term given as
\[
\frac{\partial^2 u}{\partial y^2} \text{|}_{\text{wall}} = \frac{1}{\mu} \frac{\partial p}{\partial x},
\]
and \(u_i \approx 0\) in this region due to the no-slip boundary condition. When the pressure decreases, the second derivative of the velocity with respect to \(y\), on the left hand side of the equation, is negative. This means that the velocity within the boundary layer has to increase in order to match the free stream velocity. When the pressure gradient is adverse (i.e. \(\partial p/\partial x > 0\)) the left hand side of the equation must be positive. What we also know is that the shear stress term must be negative at the edge of the boundary layer where the velocity is equal to that of the free stream conditions. Thusly, there is an inflection point for the velocity where the second derivatives of the velocity change signs from positive to negative. This inflection of the velocity causes the flow to separate and drastically increases the amount of drag associated with pressure. A schematic of separation due to an adverse pressure gradient is given in Fig. 2.5.

![Figure 2.5](image)

Figure 2.5  Separation due to an adverse pressure gradient.

According to Lissaman[3], at relatively low Reynolds numbers, around 30000, complete laminar flow can occur for small angles of attack on certain airfoils, but as the angle of attack in increased, adverse pressure gradients become more severe and laminar
separation occurs. When this separation occurs, like that at high Reynolds numbers, lift is decreased and drag is increased. In many instances the laminar separated shear layer can transition into turbulence. This is an important feature to outline since airfoil simulations are often done assuming that the flow is fully laminar or fully turbulent. Simulations that assume fully laminar flow over an airfoil can produce misleading results since it is known that most times when reattachment occurs it does so as a turbulent boundary layer. The separation on the upper surface of a low-$Re$ airfoil is commonly characterized by the laminar separation bubble. A schematic of the laminar separation bubble is seen in Fig. 2.6. In this figure we can see that the structure of the laminar separation consists of four distinct regions including the laminar boundary layer, the separated laminar shear layer which forms above the leading half of the separation bubble, the separated turbulent shear layer which forms above the separation bubble and the redeveloping turbulent boundary layer which appears after the flow reattaches. Within the separation bubble there exists a region where dead or stagnant air exists with a vortex of reversed flow. In order to accurately observe these regions, techniques such as particle-image-velocimetry (PIV) must be used in experimental studies; these regions can be studied in CFD simulations as well.

![Figure 2.6](image)

Figure 2.6  Schematic of the laminar separation bubble redrawn from Horton [4].

The phenomena associated with the boundary layer on the upper surface of an
airfoil as it transitions from laminar to turbulent are given in Bertin [8] as: unstable flow containing two-dimensional Tollmien–Schlichting (T–S) waves, a region where three-dimensional unstable waves and hairpin eddies develop, a region where vortex breakdown produces locally high shear, a region with fluctuating, three-dimensional flow due to cascading vortex breakdown, and a region where turbulent spots form. Stability theory indicates that these two-dimensional T–S waves travel in the mean flow direction and experimental observations have confirmed these predictions. One way of simulating this transition region is the use of intermittency transport transition models. Most of these intermittency transition models used to model the transition from laminar to turbulent flow do so with by means of solving additional PDEs such as an intermittency equation along with Reynolds-Averaged Navier–Stokes (RANS) turbulence models. Once these equations are solved, the turbulent viscosity $\mu_T$ is altered with a local value of $\gamma$. Simulating transition with these models results in fairly accurate predictions of mean flow characteristics with relatively good results for quantitative comparison such as skin friction, drag, lift and surface pressure. Hence, simulations using intermittency transport models do not capture all the physics associated with the laminar-turbulent transition, but do capture most of the important flow characteristics used for aerodynamic design considerations. Simulations for low-$Re$ airfoils presented in this work will include the use of the Suzen–Huang Intermittency model in an effort to predict separation location/size and to predict more accurate values of lift and drag.

The use of surface roughness or the presence of an adverse pressure gradient results in a by-pass of some of the aforementioned steps in the laminar-turbulent transition. Bertin[8] calls such devices by-pass mechanisms. It is these by-pass mechanisms that are of particular interest because by helping the laminar boundary layer transition to turbulence we form a more resilient turbulent boundary layer that will not separate as easily due to an adverse pressure gradient.

2.6 Flow Control for Low-$Re$ Number Flight Regimes

The previous discussion highlighted that the formation of the separation bubble on the upper surface due to an adverse pressure gradient is inherent to flight at low Reynolds numbers. After the separation bubble forms, the pressure tries to recover; this results in the laminar boundary layer tending to separate from the surface. This
separation is the source for the large increases in pressure drag. Depending on the specific configuration, the flow can continue to evolve into a number of different possibilities. In some instances it can re-attach and form a turbulent boundary layer and in other cases the boundary layer remains unattached[9]. Active flow control efforts have included devices such as piezoelectric actuators, plasma actuators, and morphing wings[37, 49, 50]. These active flow control devices can be implemented into control systems that alter the flow by means of moving the surface or injecting momentum into the boundary layer.

In passive turbulent boundary layer control, a few of the devices used to reduce the wall shear stress are grooved surfaces and surface roughness. Surface roughness increases the amount of parasitic drag, but also can serve as a means to decrease the production of turbulent stresses in the boundary layer. The reduction in the amount of skin friction is due to the formation of stagnant regions of air between bumps which will in turn reduce the skin friction by effectively reducing the magnitude of the velocity near the surface where the turbulent boundary layer would otherwise exist.

There have been recent investigations into the uses of inflatable wings for various applications that benefit from a low packed volume to high inflated volume ratio such as that of certain small UAVs[51]. The missions envisioned for small UAVs include surveillance for homeland security, military applications, and extraterrestrial exploration of other planets such as Mars[52, 53, 54, 55]. One lightweight approach employs inflatable wings that inherently possess bumps which modify their baseline profile and are a by-product of the manufacturing techniques used to construct them. Not only do inflatable wings possess the ability to be packaged in a relatively small space[56], they have the ability to implement wing warping technology as a method of roll control[57].

The discussion herein focuses on the effects of bumps on the airfoil surface. These bumps are a passive boundary layer control method. It has been shown that the “bumps” reduce the amount of separation on the later half of the airfoil[9], but the flow has not been classified as laminar, transitional, or turbulent. The observed aerodynamic effects due to the bumps on the surface of these wings has been shown to be favorable in terms of wing performance, as it applies to a reduction in the separation region, at Reynolds numbers in the range of 10000 to 200000 [5, 9, 11]. These favor-
able results have been primarily qualitative in nature. Thus far, there have been two primary wing profiles investigated. One is based on the Eppler 398 profile and consists of bumps that have a radius $\sim 2\%c$. The second is based on the NACA 4318 profile and consists of bumps with a radius of $\sim 1.5\%c$. Examples of the MIAV (NACA 4318) inflatable wings, manufactured by ILC Dover, can be seen in Fig. 2.7.

![MIAV inflatable wings with bumpy NACA 4318 profile.](image)

In early developments regarding inflatable wing technology at the University of Kentucky, Usui presented a thesis[11] titled *Aeromechanics of Low Reynolds Number Inflatable/Rigidizable Wings* that focused on the selection of an airfoil for low-$Re$ applications by using Xfoil[58] and the University of Illinois Urbana Champagne database[59]. This resulted in the selection of the Eppler 398 airfoil, in large part due to manufacturing considerations. Additional stress analysis was done on wing models using finite element analysis (FEA) with ANSYS. This stress analysis was done for several different loadings corresponding to different flight characteristics and/or dynamic pressures. Further analysis was done for different spans to determine an optimal or maximum span that could be used. The stress analysis was followed by wind tunnel testing of the inflatable rigidizable wing. This testing was done with prototype test sections using smoke-wire flow visualization[60]. Lift and drag data was obtained for Reynolds numbers of 156000, 200000, and 250000.

Subsequently, the profiles of the inflatable wings available for testing moved from the Eppler 398 profile to a NACA 4318 profile. ILC Dover used the Eppler 398 profile for inflatable/rigidizable wings, but the UK-FASM wing is constructed of rugged Vectran surplus from construction of NASA’s Mars Lander air bags. Inside the Vectran outer shell is a polyurethane bladder that, without the rigidity of the Vectran shell, could not withstand the nominal inflation pressures or 27 psig needed to make the wings fully rigid. The MIAV wings are based on the same NACA 4318 profile as the FASM wings,
but are made of polyurethane-coated rip-stop nylon and are a fraction of the cost. The MIAV wings are also considerably lighter in weight, and, at nominally 6 psig, are not designed for the same inflation pressures used for the FASM wings. Both of these wings possess the ability to be stored in small enclosures and can withstand the dynamic pressures that are required for flight of small UAVs in low altitudes.

The reason for the interest in numerical simulations of airfoils with bumpy profiles is a phenomenon observed from experimental results. Previously, there have been wind tunnel experiments that focused on the effects of the bumps on the Eppler 398 airfoil[9, 11]. PIV instrumentation was used to gather flow data. Below in Fig. 2.8a,b we can see the difference in the flow between the ideal and bumpy profiles. Comparing these figures, the location of the point of separation on the bumpy wing is further downstream than that of the smooth wing. These photos display the results from the use of smoke-wire visualization which give only an instantaneous view of the flow. This technique is good for observing bulk flow characteristics, but does not possess the ability to give quantitative results for the bulk flow or for the crevice regions between the bumps. Qualitative results like those seen in the figures are the primary reason for investigating the bumpy wing profiles numerically.

![Figure 2.8 Smoke-wire visualization results for the bumpy and smooth Eppler 398 airfoil at Re = 25000, α = 0°[5].](image_url)
wings with the NACA 4318 airfoil. The NACA 4318 airfoil wings have flown test beds ranging in GTOW (gross take-off weight) from 8 lbs. to 38 lbs.; these test beds vary with wing weight and wing construction. Some of these wings have been tapered while others have been rectangular in planform. Recent research efforts at the University of Kentucky have focused on flying inflatable wings in an applicable test bed autonomously. In Fig. 2.9 an example of a test bed with GTOW 16.0 lbs. flying autonomously using a Cloudcap Technology Piccolo II autopilot.

![Autonomous flight of an inflatable wing UAV.](image)

**Figure 2.9** Autonomous flight of an inflatable wing UAV.

### 2.6.1 Numerical Simulations of Low-Re Airfoils with Flow Control

In Innes[65] several steady numerical simulations of bumpy airfoils were done using Fluent. The author also presents a MATLAB script used to generate bumpy airfoils for based on the NACA 4-digit series. Results are compared for the bumpy airfoils to their smooth counterparts. Also included were simulations of the bumpy Eppler 398 airfoil used in the experimental observations at the University of Kentucky[5, 9, 11]. In this study, the author could not validate numerical results with the experimental results available for velocity profiles. For the laminar simulations, this was due to separation seen at chord locations where experimental results exhibited little or no separation; for fully turbulent simulations, using the one-equation Spalart–Allmaras[66] model, flatter velocity profiles were observed. Innes indicated that transition is likely to occur on the upper surface of the airfoil, but was not investigated further. His study extending to simulations of bumpy NACA 4-digit series airfoils included the effects of bumps on NACA 0010, 1412, and 2411 airfoils. Innes simulated the NACA series airfoils at \( Re = 300000 \) for a range of angles of attack. The author concluded that, for positive angles of attack up to nine degrees, the bumpy airfoils demonstrated less lift than the smooth airfoils. Innes also did a study that included several simulations of different
bump heights and concluded that as the bump height increased the ratio of lift-drag decreased significantly.

There have been numerous studies done within the UK CFD Group pertaining to Low-Reynolds number airfoils. Katam[67] presented a thesis entitled *Simulation of Low-Re Flow Over a Modified NACA 4415 Airfoil with Oscillating Camber* that was used in his Master’s degree thesis. Katam’s work also included validation of the structured grid based code GHOST for high-Re cases with experimental data and against other CFD codes such as NASA’s CFL3D and UMD. The results were in good agreement when running the one-equation Spalart–Allmaras (SA) turbulence model and Menter’s two-equation SST turbulence model against Menter’s SST model in GHOST. This study validated GHOST for several two-dimensional airfoil test cases and also concluded that there was little or no difference for results that included a grid that spanned $4c$ or greater above and below the airfoil respectively. This study also included the Suzen–Huang transition model[68, 69, 70] which produced results that presented no significant difference to the laminar test cases, not unreasonable for low-Re flow over a smooth airfoil. However, the addition of the bumps leads to a potentially different result. Additional studies of this work can be found elsewhere by Katam *et al*. [71, 72] and Pern *et al*. [73]. Simulations of morphing airfoils are still an ongoing project of the UK Cluster Fluid Dynamics Group in coordination with Nan Jou Pern of the UK Fluid Mechanics Group.

Various studies have also been done with Genetic Algorithms for optimizing the placement of blowing and suction jets with Menter’s SST turbulence model for steady and unsteady cases using GHOST[74, 75, 76, 77, 78, 79]. In these simulations a baseline lift and drag coefficient are used to generate a baseline fitness value. The genetic algorithm then changes inputs for jet configurations including angle, amplitude and locations. Using genetic algorithms the baseline fitness value was increased by approximately 5%. These studies were performed with a NACA 0012 airfoil. Additional simulations have been done for low-Re with airfoils such as the NACA 4414 and the NACA 0012 for a range of Reynolds numbers and angles of attack[80].
Chapter 3
Computational Tools

3.1 Governing Equations

Two codes were used to conduct studies on the “bumpy” wing profiles while only one was used to conduct studies of various models of plasma actuators. Without plasma actuators, both codes assume that there are no body forces present. The $f_i$ term in the momentum equation below is representative of the body force due to the implementation of the DBD plasma actuator model. Therefore, the general governing equations used to develop UNCLE and GHOST are given here using Einstein’s index summation notation in integral form.

**Conservation of Mass**

$$\frac{d}{dt} \int_V \rho dV = -\oint_S \rho u_i n_i dS \quad (3.1)$$

**Conservation of Momentum**

$$\frac{d}{dt} \int_V \rho u_j dV = -\oint_S \rho u_i n_i u_j dS - \oint_S p n_j dS + \oint_S \tau_{ij} n_i dS + \oint_V f_i dV \quad (3.2)$$

**Conservation of Energy**

$$\frac{d}{dt} \int_V \rho E dV = -\oint_S \rho u_i n_i E dS - \oint_S p u_j n_j dS + \oint_S u_j \tau_{ij} n_i dS. \quad (3.3)$$

where the specific total energy is $E = e + \frac{1}{2}(u_i^2)$. Here, $\rho$ is the density, $u_i$ are the velocity vector components, $n_i$ is the unit normal vector of the interface, $p$ is the static pressure, and $\tau_{ij}$ is the shear stress tensor.

Equations 3.1, 3.2 and 3.3 can be simplified with the assumption that the flow is incompressible. This assumption holds true for flows with low compressibility or that are characterized by less than 30% of the speed of sound in the respective fluid and when no large changes in temperature are present. The incompressible assumption is common in almost all liquid flows and in many relatively low speed gas flows. For the results presented in this work, the density is assumed constant. Consequently, the velocity vector is divergence free (i.e. $\nabla \cdot \bar{u} = 0$). For flows sufficiently far away from walls, the N.–S. equations reduce to the Euler equations where all the transport terms are neglected, but the computations presented in this work do not neglect these terms associated with the viscousity seen in the momentum and energy equations given.
3.2 UNCLE

One of the in-house CFD codes for the flow field calculations is titled “UNCLE”. UNCLE was originally written by P.G. Huang while at the University of Kentucky. The primary reason for the development of an unstructured grid based solver is to meet the challenges associated with difficult geometries and boundary conditions. A detailed description of the code and the validation procedures used is given elsewhere[81]. UNCLE is a two/three-dimensional, finite-volume, unstructured, incompressible N.–S. solver for steady and unsteady flows. This code relies on a cell-centered pressure-based method that is similar to the SIMPLE algorithm[82] with second order accuracy in time and space. In order to compute the flux on the interfaces of each finite volume, a second order upwind scheme is adopted for the advection terms and a second order central difference scheme is used for diffusion terms. A collocated grid system with the Rhie and Chow momentum interpolation method[83] is employed[81] to avoid the checkerboard solution of the pressure based scheme. Fluxes on the volume faces are determined through interpolation of cell-centered values. This is somewhat more complex in unstructured codes because the number of neighboring cells can change and the numbering scheme for the cells is not straightforward. The time discretization for UNCLE is a second-order fully implicit scheme. UNCLE also has the capability to handle multiple element types such as triangular, quadrilaterals, tetrahedral, and hexahedral. Optimization and further verification of UNCLE can be found elsewhere in a study by Gupta[84] titled *Performance Evaluation and Optimization of the Unstructured CFD Code UNCLE* and in additional publications[85, 86, 87].

In Fig. 3.1 we can see a common representation of a face in UNCLE. In this figure $P_1$ and $P_2$ represent cell centers and $V_1$ and $V_2$ represent the two vertex points of a two-dimensional face. Here $(\xi, \eta)$ represent the coordinates in the direction of the cell centers $(P_1, P_2)$ and the face vertex points $(V_1, V_2)$. Two-dimensional elements that UNCLE can handle include triangles and quadrilaterals; three-dimensional elements such as tetrahedral and hexahedral are also included. This code follows a left-hand rule convention for both two and three-dimensional elements. The calculations of the volumes and the areas of the faces vary with each element. In order to calculate the flow properties at the cell interfaces a Taylor Series expansion is done. The equations
take the following forms:

\[
\phi^{LHS} = \phi_{P_1} + \frac{\partial \phi}{\partial x}{|}_{P_1} (x_f - x_{P_1}) + \frac{\partial \phi}{\partial y}{|}_{P_1} (y_f - y_{P_1}) + \frac{\partial \phi}{\partial z}{|}_{P_1} (z_f - z_{P_1}) + H.O.T.\ (3.4)
\]

\[
\phi^{RHS} = \phi_{P_2} + \frac{\partial \phi}{\partial x}{|}_{P_2} (x_f - x_{P_2}) + \frac{\partial \phi}{\partial y}{|}_{P_2} (y_f - y_{P_2}) + \frac{\partial \phi}{\partial z}{|}_{P_2} (z_f - z_{P_2}) + H.O.T.\ (3.5)
\]

where \( \phi \) represents any of the scalar quantities and the velocity vector components, while the subscripts \( P_1 \) and \( P_2 \) represent the node points of a face and \( V_1 \) and \( V_2 \) correspond to the vertices of that same face. The vertices follow the left hand rule convention as well. The superscripts \( LHS \) and \( RHS \) denote the left \( (P_1) \) and right \( (P_2) \) hand sides of the face or interface and H.O.T. stands for the higher order terms in the Taylor series expansion. The equation that can be used to obtain the flow properties at the face is given as

\[
\phi_f = \frac{1}{2}(\phi^{LHS} + \phi^{RHS}) - \frac{1}{2}\text{sign}(1, \dot{m})(\phi^{LHS} + \phi^{RHS}). \tag{3.6}
\]

Gauss’s Divergence Theorem is used to calculate the gradients at the cell centers via

\[
\oint_{\mathcal{V}} \frac{\partial \phi}{\partial x_i} d\mathcal{V} = \oint_{\mathcal{A}} \phi n_i d\mathcal{A}, \tag{3.7}
\]

or rather

\[
\frac{\partial \phi}{\partial x_i} \approx \sum_{k=1}^{N_{face}} \frac{\phi n_{i,k} A_k}{V}, \tag{3.8}
\]

where \( \phi \) is, again, the flow property calculated at the cell center, \( V \) is the volume of the cell, \( \mathcal{A} \) is the area of the face, \( n_i \) is the unit normal vector of the interface, and \( N_{face} \) is the total number of faces corresponding to the cell for which the flow property is being
calculated. For diffusive fluxes, gradients at the interfaces can be evaluated using the following equations:

\[
\frac{\partial \phi}{\partial x} = \frac{\partial \phi}{\partial \xi} \frac{\partial \xi}{\partial x} + \frac{\partial \phi}{\partial \eta} \frac{\partial \eta}{\partial x} + \frac{\partial \phi}{\partial \zeta} \frac{\partial \zeta}{\partial x},
\]
(3.9)

\[
\frac{\partial \phi}{\partial y} = \frac{\partial \phi}{\partial \xi} \frac{\partial \xi}{\partial y} + \frac{\partial \phi}{\partial \eta} \frac{\partial \eta}{\partial y} + \frac{\partial \phi}{\partial \zeta} \frac{\partial \zeta}{\partial y},
\]
(3.10)

\[
\frac{\partial \phi}{\partial z} = \frac{\partial \phi}{\partial \xi} \frac{\partial \xi}{\partial z} + \frac{\partial \phi}{\partial \eta} \frac{\partial \eta}{\partial z} + \frac{\partial \phi}{\partial \zeta} \frac{\partial \zeta}{\partial z}.
\]
(3.11)

Gauss’s Divergence Theorem given in Eq. (3.7) is also used to evaluate the second derivatives at the cell centers and is similar to that of the first derivatives:

\[
\frac{\partial^2 \phi}{\partial x_i^2} \approx \frac{\sum_{k=1}^{N_{face}} \frac{\partial \phi}{\partial x_i} n_{i,k} A_k}{V}.
\]
(3.12)

3.2.1 Cell-Centered Pressure Method based on SIMPLE Algorithm

As mentioned previously, UNCLE uses the SIMPLE algorithm. The SIMPLE algorithm is a pressure correction method. The acronym SIMPLE stands for “semi-implicit method for solving pressure-linked equations”. The essence of the algorithm is as follows:

1. Guess the values of \((p^*)^n\) at all the cell centers. Also, arbitrarily set values of \((\rho u^*_i)^n\) at the proper velocity grid points.

2. Solve for \((\rho u^*_i)^{n+1}\) at all the appropriate internal grid points.

3. Substitute these values of \((\rho u^*_i)^{n+1}\) in and solve for \(p'\) at all the interior grid points.

4. Calculate \(p^{n+1}\) at all internal grid points via \(p^{n+1} = (p^*)^n + p'\).

5. Use the values for \(p^{n+1}\) to solve the momentum equations again. For this, we designate \(p^{n+1}\) obtained in step 4 as the updated values for \((p^*)^n\) to be used. With this interpretation we return to step 2 and repeat 2 through 5 until convergence is achieved. The number of iterations per time step usually ranges from five to ten depending on the current status of solution convergence.

When convergence is achieved, the velocity distribution satisfies the continuity equation[88]. The aim of this algorithm is to calculate the velocity distribution from the momentum equations that will eventually satisfy continuity. By adding pressure to the continuity equation we directly link the momentum and continuity equations with \(u_i\) and \(p\).
3.2.2 Rhie and Chow Momentum Interpolation Method

In the work of Rhie and Chow[83] a method for solving for the remaining unknown pressure is constructed by combining the continuity and momentum equations. The first assumption is that the $u$ and $v$ velocity components are obtained from

\begin{equation}
    u_p^* = \sum_{k=1}^{N} a_u u_k^* + S_u^k - \frac{y_f \Delta \xi \Delta \eta p_x^*}{a_p} + \frac{y_f \Delta \xi \Delta \eta p_y^*}{a_p},
\end{equation}

\begin{equation}
    v_p^* = \sum_{k=1}^{N} a_v v_k^* + S_v^k + \frac{x_f \Delta \xi \Delta \eta p_x^*}{a_p} - \frac{x_f \Delta \xi \Delta \eta p_y^*}{a_p},
\end{equation}

where $S_u^k$ and $S_v^k$ are the error from the generic source terms ($S_u, S_v$) in the $u$ and $v$ momentum equations after the pressure gradient terms have been extracted from them. The superscript * for $u$ and $v$ denotes that they are based on the estimated pressure field $p^*$. In general, $u^*$ and $v^*$ will not satisfy continuity. This leads to a net mass source term. To remove this mass, the velocity components are assumed to be corrected by the following relations:

\begin{equation}
    u = u^* - \frac{y_f \Delta \xi \Delta \eta p_x^*}{a_p} + \frac{y_f \Delta \xi \Delta \eta p_y^*}{a_p},
\end{equation}

\begin{equation}
    v = v^* + \frac{x_f \Delta \xi \Delta \eta p_x^*}{a_p} - \frac{x_f \Delta \xi \Delta \eta p_y^*}{a_p},
\end{equation}

where $p'$ is the pressure correction which is related to the pressure $p$ according to

\begin{equation}
    p = p^* + p'.
\end{equation}

The final result is that if the solution converges, the correction terms at the final converged state should vanish, and the converged solution should satisfy Eqs. (3.13), (3.14).

3.2.3 Gauss–Seidel Solver used in UNCLE

One of the advantages of using unstructured grid codes is that the effort that goes into the grid generation process is often much simpler for exceedingly complex geometries, but can be more demanding than structured code generation for simple geometries including airfoils, rectangular enclosures, and cylinders. However, computational effort associated with computing solutions on unstructured grids can increase significantly. Not only does the effort involved in computing the solution differ, but the methods used are considerably different.
UNCLE makes use of the delta formulation to solve the equations that govern the fluid flow given in Eqs. (3.1), (3.2), (3.3). For this method an initial value of a physical property is given as \( \varphi \) and a small change \( \Delta \varphi \) is added to the initial value at every iteration until the solution is sufficiently converged. Therefore, we define the advancement of the solution from time level \( n \) to \( n+1 \) as

\[
\varphi^{n+1} = \varphi^n + \Delta \varphi^n. 
\] (3.18)

In order to calculate the \( \Delta \varphi \) at one cell, all the neighboring cells \( \Delta \varphi \) must be calculated. This updating process is usually done with an underrelaxation factor to help filter out the large oscillations that are present when solving from initial conditions.

The subroutines used throughout the code to solve the different physical properties makes use of a point Gauss–Seidel matrix solver. It contains a DO loop over cell centers. At the end of each complete sweep over all the cells the velocity, pressure, and other properties are updated. To ensure stable convergence and accuracy there is an iterative loop over the cells sometimes multiple times[84]. This inner iterative loop is given as specified by the user. The number of inner iterations depends on the test case. The initial values used can sometimes require more iterations of this process when a test case starts, but then can be reduced as the solution becomes more converged. Likewise, the underrelaxation factors can be increased when the solution becomes more converged as well.

3.2.4 Grid Generation for UNCLE

Grid generation was done using the commercial software Gambit, commonly used with the CFD code Fluent. An in-house file reader is used to read .msh files generated by Gambit and produce appropriate data for use with UNCLE consisting of cell, vertex, and face data. After the cell, vertex, and face data is determined from the mesh, an in-house partition program that includes the METIS partitioning algorithm[89] is used to prepare the data for parallel computing. The parallel construction of the code is done using the message passing interface (MPI) protocols. One of the main advantages of using METIS is that it is extremely efficient at breaking up the unstructured grid into smaller grids while still maintaining load balance between parallel computers.
3.3 GHOST

The second CFD code used is titled “GHOST” and was also originally written by P.G. Huang. It is a structured two-dimensional, finite-volume, incompressible Navier–Stokes solver for steady and unsteady flows. This code makes use of the one-equation SA and the two-equation SST turbulence model. The quadratic upwind interpolation for convective kinematics (QUICK) scheme[90] is applied to discretize the advection terms with second order accuracy. A second order centered difference scheme is used for the diffusion terms. For the turbulence models, the QUICK or a Total Variation Diminishing (TVD) scheme[91, 92] can be used for the advection terms. The time discretization is second order upwind and uses the delta-form sub-iterative scheme. This code has also undergone vast optimization techniques for minimizing memory usage and L2 cache misses. GHOST also includes the Suzen–Huang Transition model[68] and has the capability to use overset grids. Both UNCLE and GHOST are written in FORTRAN90/95 and include the capabilities to utilize MPI parallel computing on both 32 bit and 64 bit architectures.

3.3.1 TDMA Solver in GHOST

Being a structured grid code, the flow properties are stored at each cell center location and are reference by an \((i,j)\) index scheme. Most of the calculations done in GHOST are done using a series of bi-directional sweeps in nested loops. These sweeps vary to ensure that the solution is converged. The large matrices that are formed due to the discretization methods listed previously are generated with an Alternating-Direction-Implicit (ADI-type) decomposition and are tri-diagonal. ADI-type matrices are solved by calculating the advanced time step values in two different steps[93]. The first step involves only \(x\) derivatives at the advanced time level and the second step involves only \(y\) derivatives. In each sweep the advanced time values are calculated implicitly. After these matrices are solved, the Rhie and Chow momentum interpolation method[83] is used to extract the pressure field. Being a second order structured grid code, its solver is based on the Thomas Algorithm[94] or tri-diagonal matrix algorithm (TDMA). The basic functionality of the TDMA solver in GHOST is forward substitution backward solution.
3.3.2 Grid Generation in GHOST

Two in-house grid generation codes are used to generate airfoil grids for GHOST. The first grid generation code is called “gridgen”. Gridgen, not to be confused with the popular commercial grid generation software, is a program written in C++ which takes airfoil vertex data and creates a dense grid off of the airfoil to a user specified thickness. The second grid generation code used is called “g-modified” and it is used to generate background grids. It is written in FORTRAN90/95 and has various user inputs in which the stretching and grid density are specified by the user. After generating the background grids, it writes separate files for the individual grids for parallel computing. The node balancing is left up to the user when inputting which grids will be computed on which nodes.

3.4 Turbulence Models

Turbulence is many times referred to as the last great unsolved problem of classical physics. No straightforward method exists for obtaining stochastic solutions of these non-linear partial differential equations. For now, a statistical approach, in which temporal, spatial or ensemble average is defined and the equations of motion are written for the various moments of the fluctuations about the mean. Unfortunately, the nonlinearity of the Navier–Stokes equations guarantees that the process of averaging to obtain moments results in an open systems of equations in which the number of unknowns is always greater than the number of equations. This is known as the closure problem and makes obtaining direct solutions to the (averaged) full equations of motion impossible[1].

Both of the turbulence models used for the results presented herein are Reynolds-Averaged-Navier–Stokes (RANS) turbulence models. Each of the physical properties in the RANS equations is assumed to be composed of a mean term and a fluctuating term. We write the fluctuations associated with the pressure and the velocities as: $u_i(x_i, t) = \bar{u}_i(x_i) + u'_i(x_i, t)$, $p_i(x_i, t) = \bar{p}_i(x_i) + p'_i(x_i, t)$. Figure 3.2 represents a velocity decomposed into a mean and fluctuating part where the mean of the fluctuating part is identically zero.

The fluctuating terms are given such that the time averages of these values go to
zero (i.e. $\bar{u}'_i = 0, p'_i = 0$). The mean velocities are computed as follows:

$$\bar{u}(x, y, z) = \lim_{T \to \infty} \frac{1}{T} \int_0^T u(x, y, z) \, dt$$

(3.19)

The time average of the mean velocity is the same time-averaged value, i.e.

$$\bar{u} = \lim_{T \to \infty} \frac{1}{T} \int_0^T \bar{u}(x, y, z) \, dt = \bar{u}(x, y, z)$$

(3.20)

However, as stated previously, the time-averaged value of the fluctuating terms is identically zero, i.e.

$$\bar{u}' = \lim_{T \to \infty} \frac{1}{T} \int_0^T [u(x, y, z, t) - \bar{u}(x, y, z)] \, dt = \bar{u}(x, y, z) - \bar{u}(x, y, z) = 0$$

(3.21)

This leads to the averaged equations of motion for continuity and momentum given as:

**Continuity Equations**

$$\frac{\partial \bar{u}}{\partial x} + \frac{\partial \bar{v}}{\partial y} + \frac{\partial \bar{w}}{\partial z} = 0$$

(3.22)

$$\frac{\partial u'}{\partial x} + \frac{\partial v'}{\partial y} + \frac{\partial w'}{\partial z} = 0$$

(3.23)

**Momentum Equation**

$$\frac{\partial \bar{u}_i}{\partial t} + \bar{u}_j \frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial (u_i u_j)}{\partial x_j} = -\frac{1}{\rho} \frac{\partial p}{\partial x_i} - g \delta_{i3} + \nu \frac{\partial^2 \bar{u}_i}{\partial x_j \partial x_j}.$$  

(3.24)

Both codes have the capability to use either the one-equation SA or the two-equation SST turbulence model. However, simulations presented in this work only make use of the two-equation SST model; thus, the details of the SA model will not be presented.
Two-Equation SST Turbulence Model

Menter’s SST model\[95\] was used to determine the effectiveness of a turbulence model on the bumpy airfoil test cases that are presented later. This two-equation turbulence model is a combination of the two-equation $k-\omega$ and the two-equation $k-\varepsilon$ models.

The common starting point for most two-equation models is the Boussinesq approximation (or Boussinesq Hypothesis) which implies that the Reynolds-Stress tensor takes the following form:

$$\tau_{ij} = 2\nu_T S_{ij} - \frac{2}{3} k \delta_{ij}, \tag{3.25}$$

where $S_{ij}$ is the mean strain-rate tensor.

As mentioned, Menter’s SST turbulence model is a combination of the two-equation $k-\omega$ and the two-equation $k-\varepsilon$ models. It makes use of a blending function given as $F_1$. The details of the model are given as follows:

$k$-equation (kinetic energy)

$$\frac{\partial \rho k}{\partial t} + \frac{\partial \rho u_j k}{\partial x_j} = p_k - 0.09 \rho \omega k + \frac{\partial}{\partial x_j} \left[ (\mu + \sigma_k \mu_T) \frac{\partial k}{\partial x_j} \right], \tag{3.26}$$

$\omega$-equation (dissipation)

$$\frac{\partial \rho \omega}{\partial t} + \frac{\partial \rho u_j \omega}{\partial x_j} = \frac{c}{\nu_T} p_k - \beta \rho \omega^2 + \frac{\partial}{\partial x_j} \left[ (\mu + \sigma_\omega \mu_T) \frac{\partial \omega}{\partial x_j} \right] + 2 \rho (1 - F_1) \sigma_\omega \frac{1}{\omega} \frac{\partial k}{\partial x_j} \frac{\partial \omega}{\partial x_j}. \tag{3.27}$$

The constants seen in the above equations given as $c$, $\beta$, $\sigma_k$, and $\sigma_\omega$, are given by the following expression

$$\phi = F_1 \phi_1 + (1 - F_1) \phi_2, \tag{3.28}$$

where $\phi$ represents any one of these constants; $\phi_1$ represents any constant in the $k-\omega$ model and $\phi_2$ represents the corresponding constant in the $k-\varepsilon$ model. These constants are defined as

$k-\omega$

$$\sigma_{k1} = 0.85, \sigma_{\omega1} = 0.5, \beta_1 = 0.075, c_1 = 0.553, \tag{3.29}$$

$k-\varepsilon$

$$\sigma_{k2} = 1.0, \sigma_{\omega2} = 0.856, \beta_2 = 0.0828, c_2 = 0.44. \tag{3.30}$$
The production term is given as

\[ p_k = \tau_{ij} \frac{\partial u_i}{\partial x_j}, \]  

(3.31)

where \( \tau_{ij} \) is given as the shear stress tensor:

\[ \tau_{ij} = \mu T \left[ \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} + \frac{2}{3} \frac{\partial u_k}{\partial x_k} \delta_{ij} \right] - \frac{2}{3} \rho k \delta_{ij}. \]  

(3.32)

The blending function \( F_1 \) is defined as

\[ F_1 = \tanh(\text{arg}_1^4), \]  

(3.33)

with

\[ \text{arg}_1 = \min \left\{ \max \left\{ \sqrt{k}, \frac{500 \nu}{0.09 \omega d}, \frac{4 \rho \sigma \omega^2 k}{C D_{k\omega} \omega^2} \right\} \right\}, \]  

(3.34)

where \( d \) is the distance to the closest wall and \( C D_{k\omega} \) is the positive portion of the cross-diffusion term in Eq. (3.30):

\[ C D_{k\omega} = \max \left[ 2 \rho \sigma \omega^2 \frac{1}{\omega} \frac{\partial k}{\partial x_j} \frac{\partial \omega}{\partial x_j}; 10^{-20} \right]. \]  

(3.35)

The kinematic eddy viscosity is defined as

\[ \nu_T = \frac{a_1 k}{\max(a_1 \omega; \Omega F_2)}, \]  

(3.36)

where \( \Omega \) is the magnitude of vorticity and \( a_1 = 0.31 \). The function \( F_2 \) is given by the following relationship:

\[ F_2 = \tanh(\text{arg}_2^2), \]  

(3.37)

with

\[ \text{arg}_2 = \max \left\{ \frac{2 \sqrt{k}}{0.09 \omega d}, \frac{500 \nu}{d \omega} \right\}. \]  

(3.38)

3.5 Transition Model

The transition model used in GHOST is that developed by Suzen and Huang [68, 69, 70]. This model is a relatively new transport model for intermittency. The use of this intermittency model to determine lift and drag for the bumpy airfoils was done since the precise flow physics is still not understood, but the flow is likely to be neither fully laminar nor fully turbulent. The influence of the bumps on the wings and the level of free stream turbulence are the two most influential factors on the transitional behavior
for the flow over the bumpy wings. Many CFD simulations deal with transition by switching on a turbulence model (or turbulent eddy viscosity) at an experimentally predetermined transition location[68]. Simulations of this type essentially ignore the region where transition is present and treat the two or more regions as fully laminar or fully turbulent. The development of the S.–H. intermittency transition model was done to simulate low-pressure turbines where the transitional region can span a fairly large portion of the surface. The Reynolds numbers for with the S.–H. model have been verified for turbine blades range from 25000 to 200000. Although the model has not been validated for the flow over airfoils, the Reynolds number range in which it has been verified matches the range for the low-\( Re \) simulations to follow in the discussion of bumpy wing simulations.

The version of the S.–H. Intermittency Transport model in GHOST uses the SST turbulence model to compute the value for the eddy viscosity, \( \mu_T \), and other turbulent quantities. The behavior of the transitional flows is modeled by modifying the eddy viscosity with an intermittency factor, \( \gamma \). This is done so via \( \mu^*_T = \gamma \mu_T \).

The production term used in this model is a mix of the generation terms used in previous models by Steelant and Dick[96] and Cho and Chung[97]. \( T_o \) is a value that aims to reproduce the intermittency distribution and is given by

\[
T_0 = C_0 \rho \sqrt{u^2 + v^2} \beta(s),
\]

along the streamline direction, \( s \), where

\[
\beta(s) = 2f(s)f'(s).
\]

The function \( f(s) \) is a distributed-breakdown function given as

\[
f(s) = \frac{as^4 + bs^3 + cs^2 + ds' + e}{gs^3 + h},
\]

with coefficients

\[
\begin{align*}
a &= \sqrt{n\sigma} \\
b &= -0.4906 \\
c &= 0.204 \left( \frac{n\sigma}{U} \right)^{-1/2} \\
d &= 0.0 \\
e &= 0.04444 \left( \frac{n\sigma}{U} \right)^{-3/2} \\
h &= 10e \\
g &= 50.
\end{align*}
\]
These coefficients are the same as those used by the Steelant and Dick model except \( a \) and \( g \). This approach does not use conditioned N–S. equations for these values, but rather adjusted values for a shorter distance for the distribution breakdown and faster response to the flow variables. The \( T_1 \) term is the same as the production of kinetic energy,

\[
T_1 = C_1 \gamma \frac{P_k}{k} = C_1 \gamma \frac{\partial u_i}{\partial x_j} \tau_{ij},
\]

with the shear stresses given as

\[
\tau_{ij} = \mu_T \left[ \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} - \frac{2}{3} \frac{\partial u_k}{\partial x_k} \delta_{ij} \right] - \frac{2}{3} \rho \kappa \delta_{ij}.
\]

The \( T_2 \) term represents the production term resulting from the interaction of the mean velocity and the intermittency values; it is given by

\[
T_2 = C_2 \gamma \rho \frac{k^{3/2}}{\epsilon} \frac{u_i}{(u_k u_k)^{1/2}} \frac{\partial u_i}{\partial x_j} \frac{\partial \gamma}{\partial x_j}.
\]

The production terms \( T_0 \) and \( (T_1 - T_2) \) are blended using a function \( F \) to assist in the switching from Steelant and Dick’s \( T_0 \) to Cho and Chung’s \( (T_1 - T_2) \) inside the region where transition occurs,

\[
P_\gamma = (1 - F)T_0 + F(T_1 - T_2).
\]

Suzen and Huang use the non-dimensional parameter, \( k/S\nu \), to correlate the blending function \( F \), where \( k \) is the turbulent kinetic energy and \( S \) is the magnitude of the strain rate. Simulations pertaining to turbine blades have shown that this parameter increases rapidly with the distance away from the wall in the transition region. The equation that relates \( \gamma \) and \( k/S\nu \) along the line that separates the region that is separated by a diagonal cut between the value where \( \gamma = 1 \) and where \( \gamma = 0 \) at the edge of the boundary layer is given as

\[
\frac{k}{S\nu} = 200(1 - \gamma^{0.1})^{0.3}.
\]

This line separates the use of the Steelant and Dick and the Cho and Chung models. This means that \( T_0 \) is active below this line and \( (T_1 - T_2) \) is active above it. To help ensure that the switching between the two approaches is gradual, Suzen and Huang use the following blending function

\[
F = \tanh^4 \left[ \frac{k/S\nu}{200(1 - \gamma^{0.1})^{0.3}} \right].
\]
For fully developed turbulent flow, the model switches to the Cho and Chung model except in the region very near the wall.

The previous production terms are mostly unaffected by diffusion. Thusly, the use of a diffusion related production term (proposed by Cho and Chung) is needed; it is given as

\[ T_3 = C_3 \rho \frac{k^2}{\epsilon} \frac{\partial \gamma}{\partial x_j} \frac{\partial \gamma}{\partial x_j}. \]  

(3.50)

This \( T_3 \) term is active throughout the flow field and no blending is applied to this term.

The diffusion of the intermittency parameter \( \gamma \) is given as

\[ D_\gamma = \frac{\partial}{\partial x_j} \left\{ (1 - \gamma) \gamma \sigma_{\gamma l} \mu + (1 - \gamma) \sigma_{\gamma T} \mu T \right\} \frac{\partial \gamma}{\partial x_j}. \]  

(3.51)

The final form of the S.-H. model is;

\[ \frac{\partial \rho \gamma}{\partial t} + \frac{\partial \rho u_j \gamma}{\partial x_j} = (1 - \gamma) \left[ (1 - F)T_0 + F(T_1 - T_2) \right] + T_3 + D_\gamma, \]  

(3.52)

or alternatively given in a more complete form (including the details for the production terms) as

\[ \frac{\partial \rho \gamma}{\partial t} + \frac{\partial \rho u_j \gamma}{\partial x_j} = (1 - \gamma) \left[ (1 - F)C_0 \rho \sqrt{u^2 + v^2} \beta(s) 
+ F \left( \frac{C_1 \gamma}{k} \frac{\partial u_j}{\partial x_j} - C_2 \gamma \rho \frac{k^{2/3}}{\epsilon} \frac{u_i}{(u_k u_k)^{1/2}} \frac{\partial u_i}{\partial x_j} \frac{\partial \gamma}{\partial x_j} \right) \right] 
+ C_3 \rho \frac{k^2}{\epsilon} \frac{\partial \gamma}{\partial x_j} \frac{\partial \gamma}{\partial x_j} 
+ \frac{\partial}{\partial x_j} \left\{ (1 - \gamma) \gamma \sigma_{\gamma l} \mu + (1 - \gamma) \sigma_{\gamma T} \mu T \right\} \frac{\partial \gamma}{\partial x_j}, \]  

(3.53)

with modeling constants

\[ \sigma_{\gamma l} = \sigma_{\gamma T} = 1.0 \quad C_0 = 1.0 \quad C_1 = 1.6 \quad C_2 = 0.16 \quad C_3 = 0.15. \]  

(3.54)

The initial conditions for \( \gamma \) is zero throughout the entire flow field. On the solid wall boundaries \( \gamma \) is kept at zero, in the free stream a zero gradient of \( \gamma \) is assumed, and on the outer boundaries \( \gamma \) is extrapolated from the inside of the domain to the boundary points.

3.6 Computational Resources

The computational results presented herein were obtained through the use of Kentucky Fluid Clusters (KFCs). The KFC clusters are commodity clusters that are built
out of current consumer (or PC grade) hardware. The clusters are constructed by building individual computing nodes and server machines in-house and then connecting them via an internal network. Work done by the UK CFD Group on the development and testing of commodity hardware for supercomputing has been presented\[86, 85, 98, 99, 100, 101, 102\] at numerous technical conferences. One of the research interests of the UK CFD group is to test current computer technology as it applies to scientific computing. The use of commodity clusters is a relatively low-cost alternative to the traditional approach of the high cost supercomputers. The computational resources used for the studies presented herein are done primarily on KFC3, KFC5, and KFC6A. However, the details of all the systems will be presented to inform the reader of the efforts of the UK Cluster Fluid Dynamics Group to keep up with the trends in commodity hardware.

In 2001 the UK CFD group began building clusters for their own use with consumer grade parts. The first machine was called KFC1 (1 conveniently built in 2001, the others follow with corresponding years as well). This cluster contained 20 dual processor AMD nodes. The processors were 1.4 GHz Athlon MPs (32 bit) processors with 384 MB of RAM per processor with 40 GB of disk space per dual processor node connected with a 100 Mbs\(^{-1}\) network. In 2002 KFC2 was constructed using Athlon XP 2000+ (32 bit) processors with 256 MB of RAM per node. The original configuration for KFC2 was 50 nodes and was connected with four 48-port 100 Mbs\(^{-1}\) switches with four network cards per node. Currently, KFC2 operates with 24 nodes due to degrading parts and since current technology makes maintenance of the older systems less of a priority. In 2003 KFC3 was purchased from Dell. This is the only cluster that was not built in house by the CFD group. This cluster is composed of 32 2.8 GHz Intel Pentium 4 (32 bit) processors with 256MB of RAM and 40GB of hard drive storage per node. It is split into two separate clusters that operate independently with a 100 Mbs\(^{-1}\) network. KFC4, built in 2004, is constructed with AMD 2500+ 1.826 GHz (32 bit) processors with 512 MB of RAM per node. It currently has 47 nodes linked by two separate 48-port switches. Connecting the nodes to one another is a singe Gigabit switch and connecting the nodes to the server is a single 100 Mbs\(^{-1}\) switch. In 2005 KFC5 was constructed out of 47 AMD Athlon 3200+ 2.0 GHz (64 bit) processors with 512 MB of RAM per node. This was the first machine that was built with the new (at the
time) 64 bit architecture which increased the performance of each node significantly. KFC5 is connected through a single 48-port 1000 Mbs\(^{-1}\) or Gigabit switch. A somewhat obstructed view of KFC2-KFC5 housed in RGAN 214 can be seen in Fig. 3.3a.

![Cluster Room (RGAN 214) and KFC6I](image)

Figure 3.3 Computing resources used by the University of Kentucky Cluster Fluid Dynamics Group.

In the Fall-Winter of 2006, KFC6 began testing and construction. This cluster is composed of two different machines. KFC6I (seen in Fig. 3.3b) is composed of 23 Intel Core 2 Duo e6400 2.13 GHz dual core (64 bit) processors with 1 GB of RAM per node. KFC6A is composed of 23 AMD Athlon 64x2 4600+ 2.4 GHz dual core (64 bit) processors with 1 GB of RAM per node. Both KFC6A and KFC6I are built with the relatively new dual core technology developed in parallel by Intel and AMD. The design and size of the L2 cache, the pathway for data to travel from the RAM to the processor, is one of the fundamental differences between the nodes in KFC6I and KFC6A. Intel’s technology for L2 cache is “shared” where both cores of the processor share the available cache whereas AMD’s technology allocates cache for each core. The size of the L2 cache for the Intel processors is double (2 MB shared) that of the AMD processors (2x512 kB). Before building KFC6, testing was done with several different processors from both AMD and Intel with the in-house CFD codes used by the members of the UK CFD group. In this study it was found that the highly optimized codes, GHOST and EPIC, benefited more from the higher clock speed of the AMD processor (2.4 GHz) than from the larger cache of the Intel processor. Conversely, the somewhat optimized code, UNCLE, saw many more benefits from the larger cache of the Intel machine. It
should also be noted that both KFC6 clusters benefit from new RAM technology that essentially doubles the speed that data moves from the RAM to the cache. Following these preliminary tests, it was decided among the group that purchasing the parts for two separate, smaller, clusters would be more beneficial to the group since seldom does one member occupy more than 23 nodes on a single cluster.

My contributions to the computing resources used by the UK CFD group included pricing of individual components for test nodes and for the clusters, building test nodes, servers (all previously for KFC6), archive machines (1+ TB of storage mirrored via RAID1), and a machine dedicated to grid generation and post processing (dual core AMD 4200+ with 4 GB of RAM). Other responsibilities included the installation and maintenance of compilers and kernels for the Linux operating systems of the clusters. Typical maintenance issues were also part of my responsibilities including troubleshooting and fixing “down” nodes from KFC4 and KFC5 and moving old (previous member) data from the cluster servers to the archive machines.
Chapter 4
Implementation of Flow Control with Plasma Actuators in a Navier–Stokes Solver

This chapter will discuss my research on simulations of plasma actuators. The basis of this work is recent experimental literature and previous implementations of plasma actuators[16, 37, 38, 103] by Y.B. Suzen and P.G. Huang into the structured grid CFD code GHOST[12, 47, 48]. The adaption of plasma actuators into the N.–S. equations was done by incorporating the product of the charge density and the electric field as a body force vector in the location above the embedded electrode on the surface where the dielectric material meets the fluid. This model was tested using two test cases. The first is the single linear DBD plasma actuator and the second is the L-PSJA under steady operation.

4.1 Mathematical Formulation of Suzen–Huang Model

Let us start from the basics of electrostatics. Assume we have many point charges \(q_1, q_2, \ldots, q_n\) at distances \(r_1, r_2, \ldots, r_n\) from a point charge \(Q\). If we refer to the principle of superposition the total force, \(\vec{F}\) on \(Q\) is given as:

\[
\vec{F} = \vec{F}_1 + \vec{F}_2 + \cdots = \frac{Q}{4\pi \varepsilon_o} \left( \frac{q_1 r_1}{r_1^2} + \frac{q_2 r_2}{r_2^2} + \cdots \right) = Q \vec{E},
\]

where \(\vec{E}\) is the electric field and \(\varepsilon_o\) is the permittivity of free space. Equation 4.1 describes the total force on a charge of interest, \(Q\). This is the basis for the body force vector that is used in the model of plasma actuators. If we consider the charge on a per volume basis we can express it as charge density, \(\rho_c\) with units of Coulombs per unit volume. This is consistent with the units for body force terms. Therefore, the body force per unit volume due to the effects of the single dielectric barrier plasma actuator may be written as

\[
\vec{f}_b = \rho_c \vec{E}.
\]

From Maxwell’s Equations we know that

\[
\nabla \times \vec{E} = -\mu_o \frac{\partial \vec{H}}{\partial t},
\]
where $\mu_o$ is the permittivity of free space and $\vec{H}$ is the magnetic field. If we assume that the time derivative of the magnetic field, $\frac{\partial \vec{H}}{\partial t}$, is very small, a valid assumption for plasma applications\[2\], then Eq. (4.3) yields $\nabla \times \vec{E} \sim 0$. This implies that $\vec{E}$ is the gradient of a scalar potential

$$\vec{E} = -\nabla \Phi.$$  \hfill (4.4)

This is consistent with the fact that, by definition, the curl of any gradient is zero. Assuming that the permittivity $\varepsilon$ has a non-zero spatial derivative, Gauss’s law becomes

$$\nabla \cdot (\varepsilon \vec{E}) = \rho_c.$$  \hfill (4.5)

Combining Eq. (4.4) and Eq. (4.5) yields

$$\nabla \cdot (\varepsilon \nabla \Phi) = -\rho_c.$$  \hfill (4.6)

The permittivity can further be expressed as $\varepsilon = \varepsilon_r \varepsilon_o$ where $\varepsilon_r$ is the relative permittivity of the medium of interest and $\varepsilon_o$ is the permittivity of free space. It is of interest to introduce the characteristic length of a plasma which is the Debye length, $\lambda_d$. The Debye length is the distance scale on which significant charge densities can spontaneously exist. We should also note that our smallest grid spacing should be no larger than this length scale in the vicinity of the plasma above the embedded electrode. We now introduce the following relationship for the Debye length \[22\]:

$$\rho_c/\varepsilon_o = (-1/\lambda_d^2)\Phi.$$  \hfill (4.7)

Using superposition we can break the scalar potential, $\Phi$, into a scalar potential due to the electric field $\phi$ and a second scalar potential for the charge density $\varphi$ as

$$\Phi = \phi + \varphi.$$  \hfill (4.8)

If we assume that the Debye length is small and the charge on the dielectric surface is not large, the distribution of charged species in the domain is governed by the potential caused by the electric charge on the dielectric surface and is largely unaffected by the external electric field\[48\]. Therefore, we can reasonably write two independent equations in terms of the two potentials. The partial differential equation for the potential due to the electric field generated by the applied voltage is

$$\nabla \cdot (\varepsilon_r \nabla \phi) = 0.$$  \hfill (4.9)
Therefore, the partial differential equation associated with the second potential needs to satisfy
\[ \nabla \cdot (\varepsilon_r \nabla \varphi) = -\frac{\rho_c}{\varepsilon_o}. \] (4.10)

From Eq. (4.7) we get the expression
\[ \frac{\Phi}{\lambda_d^2} = \frac{\phi + \varphi}{\lambda_d^2} = \frac{\rho_c}{\varepsilon_o} \rightarrow -\frac{\rho_c \lambda_d^2}{\varepsilon_o} + \phi = \varphi \] (4.11)

We now substitute this into Eq. (4.10) yielding
\[ \nabla \cdot [\varepsilon_r \nabla \left( -\frac{\rho_c \lambda_d^2}{\varepsilon_o} + \phi \right)] = -\frac{\rho_c}{\varepsilon_o}. \] (4.12)

We have already assumed that \( \phi \) satisfies the version of Laplace’s equation given in Eq. (4.9) so the previous equation reduces to
\[ \nabla \cdot \left[ \varepsilon_r \nabla \left( -\frac{\rho_c \lambda_d^2}{\varepsilon_o} \right) \right] + \nabla \cdot (\varepsilon_r \nabla \phi) = 0 = -\frac{\rho_c}{\varepsilon_o}. \] (4.13)

Since \( \varepsilon_o \) and \( \lambda_d \) are constants we can remove them from the differential operators and cancel terms without any loss of generality. The result is
\[ \nabla \cdot (\varepsilon_r \nabla \rho_c) = \frac{\rho_c}{\lambda_d^2}, \] (4.14)

which is the PDE for the net charge density. Once we have solved the PDE describing the potential due to the change in the electric field and the PDE describing the change in the net charge density we can calculate the body force vector that will be inserted into the Navier–Stokes computations as
\[ \vec{f}_b = \rho_c \vec{E} = \rho_c (\nabla \phi). \] (4.15)

The result for the body force vector seen in Eq. (4.15) can only be calculated after both the potential due to the electric field, \( \phi \), and the net charge density, \( \rho_c \), are computed. Currently, the solution to these equations is not directly dependent on the flow variables, meaning that it can be solved independently of flow field, excepting the influence of the flow on the plasma boundary conditions.

The boundary conditions chosen for Eq. (4.9) are to set the normal derivative of the potential due to the electric field to zero on the outer boundaries of the numerical domain. The second boundary condition is to set \( \phi = \phi(t) \) on the exposed electrode;
this represents the applied AC voltage. The implementation of different waveforms into a multi-physics solver allows this time dependent boundary condition to function similar to the input voltage of experimental setups and, given a sufficiently small time step in a numerical simulation, can capture the effects of the duty-cycle and the excitation frequency. For the embedded electrode, \( \phi = 0 \) corresponds to the ground of the circuit.

A summary of the boundary conditions for \( \phi \) can be seen in Fig. 4.1a.

\[
\begin{align*}
\text{Fluid Region:} & \quad \nabla \cdot (\varepsilon_r \nabla \phi) = 0 \\
\text{Outer Boundaries:} & \quad \frac{\partial \phi}{\partial n} = 0 \\
\text{Fluid Region:} & \quad \nabla \cdot (\varepsilon_r \nabla \rho_e) = \rho_e / \lambda_d^2 \\
\text{Outer Boundaries:} & \quad \rho_e = 0
\end{align*}
\]

(\( a \)) Boundary conditions for Eq. 4.9.

(a) Boundary conditions for Eq. 4.9.  

(b) Boundary conditions for Eq. 4.14

Figure 4.1  Schematic of boundary conditions for uncoupled Equations.

The boundary conditions chosen for Eq. (4.14) are to set the net charge density to a prescribed value of zero on the outer boundaries; this corresponds to the net charge density being very small far away from the actuator. In this model the embedded electrode is a source term for the net charge density and is time dependent. From Eq. (4.14) we can see that the value of the Debye length plays a critical role in the amount of charge that propagates into the air medium. For the air region, an empirical value of the Debye length is used. The Debye length is assumed to approach infinity inside the dielectric material. A summary of the boundary conditions for Eq. (4.14) can be seen in Fig. 4.1b.

Since the work of Forte et al. [104] the fundamental effects of the force have generally been seen as a push-push or a pull-pull phenomenon. Since the model above does not take into consideration the push-push or pull-pull effect, a time dependent term of the net charge density was added that has the opposite sign of the input voltage and also varies with a given input signal. This implementation results in a body force that always points downstream of the actuator and that essentially “pulses”, never
changing direction in $x$ (see Fig. 4.2). The net charge density above the embedded electrode will be opposite in sign from the input voltage because the positive input voltage will attract electrons leaving a net positive charge density in the vicinity above the embedded electrode. When the input voltage is negative, the opposite is true. What is important is to note that when the input voltage is positive, the gradient is negative in the plasma region, while the opposite is true for negative input voltage. Therefore, the body force will always point downstream of the actuator. The result of this addition is also consistent with the fact that the net charge will be conserved over one period. For unsteady simulation capabilities various waveforms are constructed using the Fourier series expansions. This implementation also allows for adjustments with the lag associated with charge density as its time scale may vary from the applied input voltage. The maximum charge density, $\rho_c^{\text{max}}$, appearing in the model is, in theory, the only parameter that needs to be changed to match experimental results. The above equations are solved in non-dimensional form where the distributions of $\phi/\phi^{\text{max}}$ and $\rho_c/\rho_c^{\text{max}}$ range from zero to unity. The maximum value of the potential is the amplitude of the applied voltage and the maximum value of net charge density is the parameter that currently is “tuned” to match experimental results on a case-by-case basis.

![Figure 4.2](image.png)  
Figure 4.2  Demonstration of change in charge density with input voltage.
In the Suzen–Huang model several values are prescribed, but most pertain to empirical values or material constants. They are given as follows. The permittivity of free space is \( \varepsilon_o = 8.8542 \times 10^{-12} \text{C}^2\text{N}^{-1}\text{m}^{-2} \), the relative permittivity of air is \( \varepsilon_1 = 1.0 \), the relative permittivity of Kapton is \( \varepsilon_2 = 2.3 \), the Debye length is \( \lambda_d = 0.00017 \text{m} \), \( \phi_{max} = 5 \text{kV} \), and the model parameter used in previous studies[48] to match velocity magnitude for the linear DBD actuator is \( \rho_{c_{max}} = 0.00750 \text{Cm}^{-3} \).

4.2 Additions to UNCLE for Modeling Plasma Actuators

In order to implement the S.–H. DBD model into UNCLE several upfront bookkeeping steps must be taken for the additional variables include the properties at the cell centers, vertices, and faces. This also includes gradients at the cell centers. For use in parallel computing, additional subroutines were included to ensure that additional passing from computing nodes was not confused with other data sharing associated with the main flow field solver. Blocking of different parts of the numerical domain was used to implement different material properties including relative permittivity and density and to ensure that the flow field was not solved within the dielectric material. Additional subroutines for solving the two scalar potentials were added. Lastly, implementation of the scalar potential results into the flow solver calculations as a body force term captures the modeled DBD actuator effects.

4.2.1 Blocking

In order to deal with the different materials that are present in the simulation of a dielectric barrier discharge plasma actuator, a blocking scheme was implemented to specify properties, constraints, and to minimize computational effort. Similar schemes were previously implemented in GHOST for similar applications[12, 47, 48], but the inherent nature of unstructured grids increases the difficulty of the implementation process.

One of the most useful aspects of cutting the domain into blocks, although not the primary purpose here, is for code optimization. By cutting the larger partitions that are generated for each node used, the smaller blocks more closely match the cache size for a given processor. There has been much effort implementing these types of schemes in GHOST and UNCLE. However, by breaking the sections into smaller blocks the nature
of solving the matrices that describe the momentum and other properties in the domain fundamentally changes. The initial process of breaking the grid into several partitions also changes the way in which the solution is calculated, but doing so on a smaller scale (sub-blocking) has significantly more impact since it must be done on a much finer level due to processor L2 cache size constraints. Furthermore, this “sub-blocking” is not a large problem in structured grids where the matrices being solved are tri-diagonal for the most part. The problem with unstructured grids is that fact that the solver used is a point solver. Therefore, breaking the grid into cache size blocks does not guarantee connectivity. The fact that some of the cache size blocks can contain many grid points that may not share faces makes the nature of the solver much different. This change in the process of solving the entire grid changes the solution is some instances and also effects the rate of convergence.

The blocking process used in UNCLE starts with the grid generation process. When grids are generated in Gambit, sections of the domains (faces in two-dimensional grids) must have a specified type of fluid. This is a constraint inherent for grid generation with Fluent which was utilized for our purposes. When generating the sections of the numerical domain for the flow field the medium is named “Fluid ZONE_1”. Similarly when the section is dielectric material it is specified as “Fluid ZONE_2”. The electrodes are specified as “electrode embedded” and “electrode exposed” respectively. The sub-routine “read_node_date” in UNCLE reads these titles and trims them so that the zone (e.g. Fluid, electrode) is written in four characters; the block type (e.g. ZONE_1, ZONE_2, embedded and exposed) is also trimmed to six characters to differentiate the fluid type. Once the grid is specified into the different blocks different parameters are specified such as those seen in the model and the boundary condition sections discussed previously.

4.2.2 Numerical Scheme Details

For the applications with plasma actuators, the potential due to the electric field and the potential due to the net charge density are implemented with the same numerical scheme as the scalar equations present in the SST turbulence model seen in UNCLE. A blocking method is also implemented for use with the plasma actuators. Since it is not necessary to compute both potentials in the entire domain, just in the vicinity of the
electrodes, this saves computational resources, but poses more effort when integrating into the existing code. The potential due to the electric field is computed throughout the entire blocked numerical domain. The net charge density is implemented in areas outside of the electrodes and dielectric material regions in that same blocked domain.

4.2.3 Pseudo-Code Algorithm

Here we outline the process in which the two additional PDEs for the DBD plasma actuator model are solved in UNCLE. There are numerous other additions not mentioned here associated with bookkeeping and MPI protocols for parallelization, but the essential steps in solving the PDEs is given in the pseudo-code algorithm below.

1. Initialize the values for the distribution of $\phi$ and $\rho_c$ with smooth initial data or a previously calculated distribution from a restart file.

2. Set the boundary conditions for the additional potential equations.

3. Interpolate vertex values for $\phi$ and $\rho_c$ from initial data or previous iteration values.

4. Calculate the gradients of the potentials using vertex and cell centered values.

5. Solve the partial differential equations for $\phi$ and $\rho_c$ using a point Gauss-Seidel solver with the previously calculated values for gradients and cell and vertex data.

6. Incorporate results from the two additional PDEs into the N.–S. computations as a source term.

7. Repeat steps 3-6 until desired number of iterations is complete.

4.2.4 Discretization Details

In the subroutine “CAL_PHI_2D”, given in the Appendix, the following calculations solve the additional equations added to represent the effects of the DBD plasma actuators in two-dimensions, but similar implementations were done for three-dimensions.

The diffusion coefficient is the relative permittivity of the material in which the points or vertices exist. Therefore, a volume-weighted average is taken at each face.
This is done for both equations although the same values for the permittivities are used in both:

\[ \nu_\phi = \frac{\varepsilon_\phi^1 \nu_\phi^1 + \varepsilon_\phi^2 \nu_\phi^2}{\nu_\phi^1 + \nu_\phi^2}, \quad (4.16) \]

\[ \nu_\varphi = \frac{\varepsilon_\varphi^1 \nu_\varphi^1 + \varepsilon_\varphi^2 \nu_\varphi^2}{\nu_\varphi^1 + \nu_\varphi^2}, \quad (4.17) \]

where \( V \) is the volume of the respective cell. The formulation of the basic numerical schemes in UNCLE is that the \( \xi \) direction is in the direction connecting cell centers in the grid. The values of the gradients associated with this direction does not consider the difference in spatial values at the two points. This is done for both of the scalar potentials corresponding to two separate equations:

\[ \frac{\partial \phi}{\partial \xi} = \phi^{p2} - \phi^{p1}, \quad (4.18) \]

\[ \frac{\partial \varphi}{\partial \xi} = \varphi^{p2} - \varphi^{p1}. \quad (4.19) \]

In the normal direction connecting neighboring vertices we use the \( \eta \) direction previously displayed in Ch. 3 (Fig. 3.1) describing the formulations done in UNCLE. Again, we neglect the spatial variation as it will be related through the appropriate transformation:

\[ \frac{\partial \phi}{\partial \eta} = \phi^{v2} - \phi^{v1}, \quad (4.20) \]

\[ \frac{\partial \varphi}{\partial \eta} = \varphi^{v2} - \varphi^{v1}. \quad (4.21) \]

Here we introduce the transformation back to the \( x \) and \( y \) coordinate system in the same way we saw the velocity components in Eq. (3.11):

\[ \frac{\partial \phi}{\partial x} = \frac{\partial \phi}{\partial \xi} \frac{\partial \xi}{\partial x} + \frac{\partial \phi}{\partial \eta} \frac{\partial \eta}{\partial x}, \quad (4.22) \]

\[ \frac{\partial \varphi}{\partial x} = \frac{\partial \varphi}{\partial \xi} \frac{\partial \xi}{\partial x} + \frac{\partial \varphi}{\partial \eta} \frac{\partial \eta}{\partial x}, \quad (4.23) \]

\[ \frac{\partial \phi}{\partial y} = \frac{\partial \phi}{\partial \xi} \frac{\partial \xi}{\partial y} + \frac{\partial \phi}{\partial \eta} \frac{\partial \eta}{\partial y}, \quad (4.24) \]

\[ \frac{\partial \varphi}{\partial y} = \frac{\partial \varphi}{\partial \xi} \frac{\partial \xi}{\partial y} + \frac{\partial \varphi}{\partial \eta} \frac{\partial \eta}{\partial y}. \quad (4.25) \]

The fluxes \( \mathcal{F} \) at each of the faces are calculated by using the previously calculated diffusion coefficients, gradients, and face areas \( A \) by

\[ \mathcal{F}_\phi = -\nu_\phi \frac{\partial \phi}{\partial x} A_1 - \nu_\phi \frac{\partial \phi}{\partial y} A_2, \quad (4.26) \]
\[ F_\phi = -\nu_\phi \frac{\partial \phi}{\partial x} A_1 - \nu_\phi \frac{\partial \phi}{\partial y} A_2. \] (4.27)

The right hand side terms of \( p1 \) and \( p2 \) are updated with the flux and has an outward pointing normal via the coordinate system choice previously prescribed:

\[ \text{RHS}_{\phi}^{p1} = \text{RHS}_{\phi}^{p1} - F_\phi, \] (4.28)
\[ \text{RHS}_{\phi}^{p2} = \text{RHS}_{\phi}^{p2} - F_\phi, \] (4.29)
\[ \text{RHS}_{\varphi}^{p1} = \text{RHS}_{\varphi}^{p1} - F_\varphi, \] (4.30)
\[ \text{RHS}_{\varphi}^{p2} = \text{RHS}_{\varphi}^{p2} - F_\varphi. \] (4.31)

The off-diagonal terms for the matrix formation are updated with the coefficients for the previous flux calculations via

\[ a_{\phi}^{p1} = a_{\phi}^{p2} = \nu_\phi \left( \frac{\partial \xi}{\partial x} A_1 + \frac{\partial \xi}{\partial y} A_2 \right), \] (4.32)
\[ a_{\varphi}^{p1} = a_{\varphi}^{p2} = \nu_\varphi \left( \frac{\partial \xi}{\partial x} A_1 + \frac{\partial \xi}{\partial y} A_2 \right). \] (4.33)

The main diagonal terms are then calculated by summing the neighboring cells off-diagonal terms:

\[ a_{p,\phi} = a_{p,\phi} + \sum_{k=1}^{N} a_{\phi,k}, \] (4.34)
\[ a_{p,\varphi} = a_{p,\varphi} + \sum_{k=1}^{N} a_{\varphi,k}. \] (4.35)

Additional terms are included in the sum \( a_{p,\phi} \) and \( a_{p,\varphi} \) respectively in the case where additional source terms are added to the right hand side coefficients. This is done to make the matrix more stable during the calculations to assist convergence if the initial conditions cause large oscillations. To further assist in stability an underrelaxation factor \( \beta \) is also used to make the main diagonal more dominant.

\[ a_{p,\phi} = \frac{a_{p,\phi}}{\beta_\phi}, \] (4.36)
\[ a_{p,\varphi} = \frac{a_{p,\varphi}}{\beta_\varphi}. \] (4.37)

These values typically range from \( 0.1 < \beta \leq 1 \), \( \beta=1 \) meaning no underrelaxation. For the simulations presented in this study an initial underrelaxation of 0.3 was used for the 64
first 500 time steps and then an underrelaxation of 0.8 was used for following iterations until the solution converged. Once all the values for the main diagonal, off-diagonal and right hand side terms are calculated, the point Gauss–Seidel point solver is used to calculate $\Delta \phi$ and $\Delta \varphi$. Once these values are obtained the values of $\phi$ and $\varphi$ are updated for the next time step via

$$\phi^{n+1} = \phi^n + \Delta \phi^n,$$  \hspace{1cm} (4.38)  

$$\varphi^{n+1} = \varphi^n + \Delta \varphi^n$$ \hspace{1cm} (4.39)

The update process is typically done several times for each outer iteration for the main flow solver and for the additional plasma equations. For the simulations presented here, the plasma equations typically do ten iterations in the G.–S. solver for each outer iteration. For each time step, twenty outer iterations are done. Therefore, the addition of the two scalar equations for the DBD plasma actuator simulations requires two hundred additional calls to the G.–S. solver for each time step taken.

### 4.3 Test Case I: Two-Dimensional Quiescent Flow with Linear SDBD Plasma Actuator

The initial actuator simulated consists of two 1 cm wide, 100 $\mu$m thick conductive copper strips as electrodes which are separated by a 100 $\mu$m thick Kapton dielectric with a relative permittivity of $\varepsilon_r = 2.7$. Streamwise spacing of electrodes is 500 $\mu$m. The relative permittivity of air is $\varepsilon_r = 1.0$. Several coarse grids were used in preliminary studies, but a fine grid with $\Delta y \approx 2 \times 10^{-6}$ m, $\Delta x \approx 2.5 \times 10^{-7}$ m was used for this study. Two views of the finest grid are given in Fig. 4.3. The total number of grid points for the mesh seen below is approximately 120,000. Due to the limitations in Gambit, the grids were constructed in centimeters and then scaled to meters for the computations before UNCLE calculated the values associated with the cell geometry.

In the experiments the lower electrode was grounded and plasma region was generated using a square wave with frequency of, $\omega=4.5$ kHz and amplitude of $\phi^{\text{max}}=5$ kV. There is no external flow - all flows are generated by the action of the plasma actuator. From the experiment it was observed that the flow was drawn into the surface region above the embedded electrode by the plasma induced body force. This resulted in a jet issuing to the right of the actuator with a maximum velocity of approximately 1 m s$^{-1}$. 
4.3.1 Numerical Comparisons

In this section we will compare the results for the SDBD geometry between two different numerical simulations of the Suzen–Huang model implemented in UNCLE and GHOST. Previous studies\[12, 47, 48\] by Suzen and Huang have discussed the implementation of this model into the structured grid code GHOST. The model parameters for this comparison were
\[
\rho_c^{\text{max}} = 0.0075 \text{ C/m}^3, \quad \lambda_d = 0.00017 \text{ m}, \quad \varepsilon_r^1 = 1.0 \quad \text{and} \quad \varepsilon_r^2 = 2.7
\]
which were used in previous studies conducted by Suzen \textit{et al.} [48]; the results from that paper will be presented again for the purpose of comparison with the UNCLE simulation results.

In Fig. 4.4, we can see that the distribution of the normalized potential due to the electric field is similar in both simulations. The biggest difference in this plot is the streamlines. The streamlines in the UNCLE results indicate that the body force is influencing the fluid in the $x$ direction more than the GHOST simulations further away from the dielectric surface. More discussion on the velocity will be included in the experimental comparisons with $u$ and $v$ velocity profiles.

In Fig. 4.5 we can see that the distribution of the normalized net charge density for the two different simulations are similar. However, the solution presented from GHOST appears to be different from that of UNCLE in that the normalized net charge density propagated further from the dielectric surface (see Fig. 4.5). This could result in differences in the magnitude of the force vector and ultimately properties of the flow.
Figure 4.4  Comparison of numerical results for $\phi/\phi^{\text{max}}$ for a linear SDBD plasma actuator in quiescent flow. A comprehensive grid independence study has not been done for the simulations in UNCLE. Ideally, creating a grid identical to that used in the GHOST simulations would be a better means to compare the solutions. It should be noted that the grids have similar $y$ grid resolution near the electrodes, but the grid used in GHOST for the electro-magnetic equations has more $x$ resolution.

Figure 4.5  Comparison of numerical results for $\rho_c/\rho_c^{\text{max}}$ for a linear SDBD plasma actuator in quiescent flow.

The magnitude of the computed normalized body force vector is approximately 600 for both simulations and the majority of the body force is located in the gap between the two electrodes and the maximum value of the body force for the two simulations is at the edge of the embedded electrode (Fig. 4.6). The value for the
normalized body force is slightly different for the two simulations due to differences in the charge field distribution. This most noticeable between the two electrodes, where the extended distribution of the charge density in the UNCLE simulation causes the body force to be slightly stronger, especially near the edge of the leading electrode. The slight discontinuity in the contour near the exposed electrode for the UNCLE results is due to the values of the gradients being solely stored at the cell centers. There is an artificial boundary that cuts through the grid at that location that is inherent to the parallelization process. Since $\rho_c$ and $\phi$ are stored at the cell centers and at the vertices for the flux and gradient calculations to be done, this is not seen in Fig. 4.4,4.5. Overall, the distributions in all three sets of numerical comparisons are in good agreement.

![Figure 4.6](image)

(a) UNCLE  
(b) GHOST[48]

Figure 4.6  Comparison of numerical results for $|f_b|/\phi_{\text{max}} \rho_{c_{\text{max}}}$ for a linear SDBD plasma actuator in quiescent flow.

4.3.2 Experimental Comparisons

In Fig. 4.7 two sets of results for the velocity profiles are given at different upstream and downstream locations of the actuator for a linear SDBD plasma actuator in quiescent flow. For the experimental results (Fig. 4.7a), the maximum value of the horizontal velocity $u$ shown for the locations extracted from the flow field is approximately 0.95 ms$^{-1}$ and is at a downstream location 1.6 cm from the exposed electrode edge. The flow essentially starts to accelerate once it has reached the trailing edge of the exposed electrode ($x=0.0$). As the flow passes over the grounded electrode, the flow near the surface continues to accelerate and entrain more fluid, creating a growing jet-like region.
This process continues even after the flow moves beyond the ground, with the thickest jet corresponding to the 1.6 cm location with a height at this point approaching 0.5 cm. The results for the model simulation has a distinctly different distribution (Fig. 4.7b). While the maximum velocity for the simulation (1.5 ms$^{-1}$) is of the proper order (and could be further corrected by proper tuning of the value of $\rho_c^{\text{max}}$), the location of this maximum velocity in the simulation is about 0.07 cm downstream of the actuator edge, just past the leading edge of the ground, similar to the GHOST simulations[48]. From this point, the peak velocity of the near-surface jet decreases as the jet width widens. By the 1.6 cm location, the UNCLE simulation peak velocity is approximately 0.5 ms$^{-1}$. The UNCLE jet is somewhat thinner than the experimental results at that location downstream of the interface. The thickness of the velocity profiles from the experimental results suggest that more fluid is being entrained in the experiment than what is captured in the model. The acceleration which occurs beyond the region where the embedded electrode is located is not captured by the model.

![Graph](attachment:image.png)

(a) Experimental[38]

(b) UNCLE

Figure 4.7  Comparison of $u$ velocity profiles to experimental results.

The experimental results for the $v$ velocity profile suggest that the location where the $v$ velocity is the strongest and downward is at the end of the exposed electrode(see Fig. 4.8). This is also true for the numerical simulations carried out in both codes. The magnitude of the velocity is similar to the experimental results in the UNCLE simulation, but not a precise match. The maximum downward velocity from UNCLE is at the interface and is approximately 0.3 ms$^{-1}$ which is roughly 0.105 ms$^{-1}$ less.
than that observed in the experiments. This lack of downward flow is consistent with the thinner boundary jet in the simulations, as less flow is drawn down into the jet at the start. The simulations also indicate a “bounce” where at $x=0.4$, the vertical velocity near the surface becomes positive in the simulations, something that happens in the experiments only at about 0.8 cm. The magnitude of the maximum values of the velocities can be matched at any specified location in the $x$ by changing $\rho_{e}^{\text{max}}$, but the magnitude of $v$ at the interface and the increased velocity and fluid entrainment beyond the embedded electrode cannot simultaneously be achieved by simply modifying this parameter.

Another input parameter that can be altered, other than that of $\rho_{e}^{\text{max}}$, to better match the experimental data is the length scale of the plasma, $\lambda_d$. Since the equation that governs the propagation of charge density relies on this length scale, the distribution of significant charge density will vary in the region of the actuator as this constant changes. Moreover, as this length is increased, the charge density will propagate further into the region where the N.-S. computations are employed. In the extreme case where the length scale is assumed to be very large, or where the source term in the charge density equation is very small, the charge density propagates into the fluid domain in such a way that the body force distribution extends well beyond the height of the exposed electrode. This would yield a body force present outside of the opaque plasma region. Experimental results presented in Fig. 4.7 suggest that the jet thickness approaches a
value five times thicker than the exposed electrode height. This would imply that the body force may extend into regions where the plasma is transparent to the naked eye. This is not to suggest that the Debye length is unphysically long; rather, it suggests that the charge density distribution must be more expansive than what is being generated by the current model configuration.

In Fig. 4.9 we can see different plots for the normalized net charge density $\rho_c/\rho_c^{max}$ for different values of the Debye length. In these plots we can see as the Debye length is increased from the value used by Suzen et al. [48], the domain in the vicinity of the electrode where significant charge density exists increases. This leads to a considerably larger body force distribution resulting in larger induced velocities for a constant $\rho_c^{max}$ and $\phi^{max}$. Large charge distributions like those seen in Fig. 4.9 also yield fundamentally different velocity profiles than those observed in the experiment (see Fig. 4.10). Therefore, by increasing the Debye length we can change the normalized charge distribution and generate thicker jets than the original values used, but not necessarily model the experimental results more accurately.

![Figure 4.9](image)

**Figure 4.9** $\rho_c/\rho_c^{max}$ distributions for two larger values of $\lambda_d$.

However, by altering both the Debye length and the maximum net charge density similar jet profiles can be obtained. In Fig. 4.11,4.12 we again see the experimental results[38] and results from UNCLE when $\lambda_d=0.0017$ m. $\rho_c^{max}=0.000625$ Cm$^{-3}$. This is the same $\lambda_d$ that produced velocities near 3 ms$^{-1}$ when $\rho_c^{max}$ was kept at its default value of 0.0017 Cm$^{-3}$. In Fig. 4.11 the $u$ velocity is less than that recorded in the experiment, which can be tweaked by increasing $\rho_c^{max}$, but the jet thickness is now more
representative of experimental results at the downstream location of 1.6 cm. The trend of the $u$ velocity magnitudes is similar in both where the large increase in magnitude is not obtained until 0.4 cm past the exposed electrode edge. What is different is that the computational results do not increase in velocity beyond the electrode, but the increase in net charge density due to the increase in the Debye length, has made the losses less than that seen for the original parameter configuration. In Fig. 4.12 we see the $v$ velocity profiles for this configuration. In these plots we see that the trends associated with the $v$ velocity profiles are consistent, but the computations yield a smaller $v$ velocity at the electrode interface than the experiments. These results are evidence that changing the model parameters can give fairly good results describing the effective wall jet created by the DBD plasma actuators.

$\lambda_d=0.0017\text{m}$

Figure 4.10  $u$ velocity profiles for two larger values of $\lambda_d$.

$\lambda_d=0.017\text{m}$

Figure 4.11  $u$ velocity profile comparisons for $\lambda_d=0.0017\text{ m}$, $\rho_{\text{max}}=0.000625\text{ Cm}^{-3}$.
Figure 4.12  $v$ velocity profile comparisons for $\lambda_d=0.0017$ m, $\rho_{e,max}^{\text{Cm}}=0.000625$ Cm$^{-3}$.
4.4 Test Case II: Two-Dimensional Quiescent Flow with L-PSJA

The L-PSJA simulated consists of two 1 cm wide, 100 µm thick conductive copper strips as exposed electrodes which are separated from the embedded electrode that is 1.3 cm wide by a 100µm thick Kapton dielectric with a relative permittivity of $\varepsilon_r = 2.7$. Stream wise spacing of the electrodes is 500 µm. The relative permittivity of air is $\varepsilon_r = 1.0$. A grid with $\Delta y \approx 1 \times 10^{-6} \text{m}, \Delta x \approx 2 \times 10^{-5} \text{m}$ was used for this study. Two views of the grid are given in Fig. 4.13. The total number of grid points for the mesh seen below is approximately 120,000. The flow field boundary conditions are given in Fig. 4.14 where the outer boundaries are specified as outflow with $\partial u_i/\partial n_i = 0$ and the surfaces are specified to satisfy the no-slip condition, $u_i = 0$.

![Figure 4.13](image)

(a) Full View  
(b) Detailed View  

Figure 4.13  
Computational grid used in L-PSJA simulations.

After the study of the SDBD plasma actuator at steady operations was the study of the L-PSJA developed by Jacob and Santhanakrishnan[16, 38, 103, 37]. Using the grid seen in Fig. 4.13, initial simulations were conducted using the S.–H. model. In the previous test case for the SDBD we saw contours for the electric potential $\phi$ and the net charge density $\rho_c$. The results for these distributions for the L-PSJA are given in Fig. 4.15a, 4.15b. A contour plot for the normalized body force is seen in Fig. 4.16. As mentioned previously, the body force is given as the product of the gradient of the potential due to the electric field and the net charge density (i.e. $f_b = -\rho_c \nabla \phi$). Again, we see the discontinuity in the body force distribution near the left electrode which is similar to that seen in the contour for the body force for the SDBD study. As a
reminder, this is due to the gradients of $\phi$ (i.e. $d\phi/dx_i$) being defined at the cell centers and not at the vertex points. When data is extracted for plotting, the vertex points are used with cell connectivity data to generate the contours. Since the body force is added to the N.–S. computations only at the cell centers, this discontinuity is not representative of the actual force driving the flow field near the left exposed electrode. When the two exposed electrodes have a voltage applied to them they generate the electric field distribution seen in Fig. 4.15a which has the characteristics of having negative $x$ gradients going from the left exposed electrode to the center of the ground and positive $x$ gradients from the right exposed electrode to the ground. This results in two separate $x$ body forces that oppose each other which generate a zero-net-mass-flux jet above the center of the ground electrode. The $y$ components of the gradient are both positive in this configuration where the negative sign in the body force expression results in fluid being gathered from above the exposed electrodes and injected to the center of the L-PSJA device. This is what is observed in the experiments[103] as well.

In the previous section it was demonstrated that the model can yield similar qualitative and quantitative results for the velocity profiles when above the embedded electrode. Thus, initial predictions assumed that the model would yield better results for the L-PSJA than previously demonstrated for the SDBD device since the jet exists above the embedded electrode.

Preliminary results revealed that the distance between the dielectric interface and

Figure 4.14  Flow field boundary conditions for L-PSJA simulations in quiescent flow.
Figure 4.15  Results for normalized electric potential, $\phi$, and normalized net charge density, $\rho_c$, for L-PSJA simulations in the vicinity of the electrodes.
the upper outflow boundary is critical in these quiescent flow computations. The low-
Re or low velocity jets generated by the L-PSJA devices are extremely sensitive to
the outflow boundary conditions above. As previously stated, the upper boundary is
assumed to be outflow where the normal derivatives of the velocity components are
zero (i.e. $\partial u_i / \partial n_i = 0$). This is a common boundary condition for aerodynamic flows
such as airfoil and turbine blade simulations. The effects due to the outflow boundary
condition are characterized by highly distinguishable jet asymmetry and creep as the jet
approaches the upper boundary. This could possibly be due to the lack of refinement
in the grid at the upper boundary due to the stretching nature of the grid. Once it
was found that the size of the original mesh was influencing the development of the jet
due to the L-PSJA, a second larger mesh was generated that with twice the geometric
distance from the dielectric surface (see Fig. 4.17a,c). This mesh has the same horizontal
dimensions as the previous mesh and uses the same boundary conditions. In Fig. 4.17b,d
the influence that the upper boundary has on the flow field is apparent. In these plots
there are significant problems with the smaller mesh upper boundary preventing the
flow from fully developing and ultimately influencing the jet. When the upper boundary
was increased, the influence was no longer seen. In Fig. 4.17d there are standing vortex
structures approximately 15 cm above the dielectric surface, but these structures do
not influence the symmetry of the jet below the $\approx 8$ cm height above the surface. As
a result of this observation, the remainder of the L-PSJA simulations are conducted using the mesh with the 20 cm upper boundary. It is important to note that in the experiments, a 15 cm upper boundary was used which has a wall boundary condition. Future simulations will be conducted to see if an upper no-slip boundary will affect the jet velocity values at heights of interest to our study which fall in the PIV windows of less than 3.9 cm above the dielectric surface.

4.4.1 Experimental Comparisons

The first study was with the original values for $\lambda_d$ and $\rho_c^{\text{max}}$ from the previous investigation of the SDBD plasma actuators which were 0.00017 m and 0.0075 Cm$^{-3}$ (Coulombs per unit volume) respectively. Using experimental results from Santhanakrishnan et al. [103] we compare the jet width at different locations from the dielectric surface with local normalized velocities. In Fig. 4.18 the results for four different locations are compared. In Fig. 4.18a we can see that the results from the computational result predict a thinner jet than observed in the experiment. We can also see that the computational results tend to predict a larger negative $v$ velocity component near the edge of the jet. Further from the dielectric surface (Fig. 4.18b) we can see that the computations again predict a thinner jet than observed experimentally. However, the trend of negative velocity at the edge of the jet is reversed; at these locations the experimental results suggest a larger negative $v$ component than the computations. Although these velocity profiles do not match perfectly, they certainly demonstrate that the model can predict similar normalized velocity jet profiles at different downstream locations from the dielectric surface. Overall, the model does a good job generating jet profiles similar to that seen in the experiments. This is promising for implementation on surfaces of airfoils or cylinders to modify values such as lift, drag, skin friction, and surface pressure.

In Fig. 4.19 we see vorticity contours with streamlines from the experiment[103] and from the computations done with UNCLE. In this plot we see that the relatively strong concentrations in vorticity for the experiment and computations agree in location. The streamlines remind us that in the previous plots (Fig. 4.18) we see a thinner jet for the computations than what was observed experimentally. These streamlines also show that the particle paths for the two results are not identical as the computations
Figure 4.17  Comparisons of the two different meshes used for L-PSJA computations.
Figure 4.18 Normalized velocity jet width comparisons for different downstream locations from dielectric surface for L-PSJA in quiescent flow.

have a stronger horizontal component of entraining fluid from the left and right near the dielectric surface. The computations also demonstrate that the fluid entrained approaches the edge of the exposed electrodes in a stronger fashion than what was observed in the experiment.

Figure 4.19 Vorticity contour comparison with streamlines for L-PSJA in quiescent flow.

4.4.2 Parameter Study

The initial results revealed that the model is capable of producing fairly accurate results for the normalized velocity jet thickness at different locations from the dielectric surface. This prompted a parameter study to see if tweaking the Debye length, $\lambda_d$, and the maximum net charge density, $\rho_c^{\text{max}}$, could yield more accurate results. The first
study conducted kept $\rho_c^{max}$ fixed to the default value of $0.0075 \text{ Cm}^{-3}$ and varied the Debye length which has been shown to change the distribution of charge density.

Figure 4.20 shows different distributions of $\rho_c/\rho_c^{max}$ for six different $\lambda_d$ parameters. In this study the baseline value for $\lambda_d$ was computed followed by 110%, 120%, 130%, 150%, and 200% of the baseline value. We can see as the value of $\lambda_d$ is increased from the baseline value the distribution of significant charge density increases from the dielectric surface. This increase in charge density leads to larger regions of body force. This will inherently affect the width of the jet that originates from the center of the embedded electrode.

The results for the jet widths with $v$ velocity magnitudes is given in Fig. 4.21. In Fig. 4.21a the jet width at 6 mm from the surface is largely unaffected by the increase in $\lambda_d$ from the base value of $0.000187 \text{ m}$. As we get further from the dielectric surface at 10 mm in Fig. 4.21b we do not see a significant difference in the jet width either. The same trend is seen at 26 mm and 30 mm from the dielectric surface in Fig. 4.21c,d. This result is different than that seen in the comparison of the linear actuator, where the value of $\lambda_d$ changed the thickness of the jet near the wall. What we can gather from the plots in Fig. 4.21 is that an increase in $\lambda_d$ results in an increase in the maximum centerline jet velocity. This is also consistent with an increase in $\lambda_d$ from the linear actuator comparisons. Normalized comparisons of the jets are given in Fig. 4.22-4.26. In these plots we see further demonstration of the trends seen in Fig. 4.21 which demonstrate no significant demonstration of jet width increase with increase of $\lambda_d$. The effect of increasing the value of $\lambda_d$ will have the same effects as increasing the relative permittivity of the dielectric material. Numerically, the values of the relative permittivity are included in the flux calculations for each iteration while the value for $\lambda_d$ is included as a source term on the right hand side of the net charge density equation.

In Fig. 4.27 the effects in peak centerline velocity with the change in $\rho_c^{max}$ is shown. In this plot we can see that the default value from the previous numerical investigations[48] predicted peak velocities near $0.95 \text{ cms}^{-1}$. Three other values were chosen to investigate the effects in the centerline velocity; the values chosen $0.00375 \text{ Cm}^{-3}$, $0.003 \text{ Cm}^{-3}$, and $0.001875 \text{ Cm}^{-3}$ which correspond to 50%, 40% and 25% of the
Figure 4.20 $\rho_c/\rho_c^{\text{max}}$ distributions for different values of $\lambda_d$. 

(a) (baseline) $\lambda_d=0.00017$ m  
(b) $\lambda_d=0.000187$ m  
(c) $\lambda_d=0.000204$ m  
(d) $\lambda_d=0.000221$ m  
(e) $\lambda_d=0.000255$ m  
(f) $\lambda_d=0.00034$ m
Figure 4.21  Jet comparisons for different downstream locations from dielectric surface for L-PSJA in quiescent flow with six different values of $\lambda_d$ (note: $\lambda_d^{base}=0.00187$ m).

Figure 4.22  Normalized velocity jet width comparisons for different downstream locations from dielectric surface for L-PSJA in quiescent flow for $\lambda_d=0.000187$ m.
Figure 4.23 Normalized velocity jet width comparisons for different downstream locations from dielectric surface for L-PSJA in quiescent flow for $\lambda_d=0.000204$ m.

Figure 4.24 Normalized velocity jet width comparisons for different downstream locations from dielectric surface for L-PSJA in quiescent flow for $\lambda_d=0.000221$ m.
Figure 4.25 Normalized velocity jet width comparisons for different downstream locations from dielectric surface for L-PSJA in quiescent flow for $\lambda_d = 0.000255$ m.

Figure 4.26 Normalized velocity jet width comparisons for different downstream locations from dielectric surface for L-PSJA in quiescent flow for $\lambda_d = 0.00034$ m.
default value $\rho_c^{max}$ value. In Fig. 4.27 the line corresponding to $\rho_c^{max} = 0.00375$ Cm$^{-3}$ is in relatively good agreement with the experimental results for a $y$ value of approximately 10 mm.

![Image](image_url)

Figure 4.27 Centerline velocity comparison for L-PSJA in quiescent flow under steady operation with different values of $\rho_c^{max}$.

Using the results from the simulations presented above best fit lines were generated to determine values of $\lambda_d$ and $\rho_c^{max}$ needed to generate peak velocities with a fixed input voltage $\phi^{max} = 5$ kV. In Fig. 4.28 a plot of $V_{peak}$ against a range of $\lambda_d$ values is given with $\rho_c^{max}$ fixed at 0.0075 Cm$^{-3}$. This figure includes two best fit lines, exponential and linear. The equation for the exponential best fit line is

$$V_{peak} \approx e^{3221\lambda_d - 0.7956} \quad (R^2 \approx 0.9914).$$  \hfill (4.40)

The equation for the linear best fit line is

$$V_{peak} \approx 3402\lambda_d + 0.1813 \quad (R^2 \approx 0.9779).$$  \hfill (4.41)

Here we can see that the data from the simulations fits the exponential curve better with a $R^2$ value of 0.9914. More simulations may lead to slightly different coefficients for the corresponding fit lines, but the trend in the range of $\lambda_d$ from 0.00017 m to 0.00022 m has a fundamentally different slope than the peak velocity values ranging in
\( \lambda_d \) from 0.00022 m to 0.00034 m. Even though the Debye length is not directly in the calculation of the body force in the S.-H. model we would expect that as it increases we should decrease the value of \( \rho_c^{\text{max}} \) in order to maintain a velocity comparable to the experimental results. This is consistent with the definition of the Debye length presented earlier in Eq. (2.7).

In Fig. 4.29 a plot of \( V_{\text{peak}} \) against a range of \( \rho_c^{\text{max}} \) values is given with \( \lambda_d \) fixed at 0.00017 m. This figure includes two best fit lines as well, power and linear. The equation for the power best fit line is

\[
V_{\text{peak}} \approx e^{0.8189 \ln(\rho_c^{\text{max}})+3.781} \quad (R^2 \approx 0.9993).
\]  

(4.42)

The equation for the linear best fit line is

\[
V_{\text{peak}} \approx 96.61 \rho_c^{\text{max}} + 0.08167 \quad (R^2 \approx 0.9997).
\]

(4.43)

Both best fit lines in this plot match the data from the simulations well with an \( R^2 \) value >0.999. This plot also implies that with a fixed \( \rho_c^{\text{max}} \), \( V_{\text{peak}} \) will vary linearly with input voltage \( \phi_c^{\text{max}} \) since the normalized body force is proportional to both of these values (i.e. \( |f_b|/\phi_c^{\text{max}} \rho_c^{\text{max}} \)).

Figure 4.28  
Trend line plot for \( V_{\text{peak}} \) for L-PSJA under steady operation with \( \rho_c^{\text{max}}=0.0075 \text{ Cm}^{-3} \) and ranging \( \lambda_d \) values with linear and exponential best fit lines.
4.4.3 Quantitative Comparisons

In this section we compare the results from the numerical simulations to experimental results quantitatively (not non-dimensionally as presented in the previous discussion). The previous plots demonstrated that the numerical results are in relatively good agreement for jet width. In Fig. 4.31 the jet width and \( v \) velocities are compared for the L-PSJA at 6 mm and 10 mm from the dielectric surface. In this plot we can see that the computations are in good agreement for both of the \( y \) values. The peak velocity for the computations are greater than the experiment, but from the previous plot (Fig. 4.27) the value of \( \rho_{c}^{\text{max}} \) can be changed to match the peak velocity exactly. The computations do demonstrate a larger negative \( v \) velocity approximately 4 mm from the center for both \( y \) values. In Fig. 4.32 we compare the \( v \) velocity at 26 mm and 30 mm from the dielectric surface. Here we see that the jet has a stronger \( v \) velocity in the computations than the experiment, but the width is similar and the \( v \) velocity 5 mm from the center is also closer to the experimental value than what was observed at 6 and 10 mm from the surface. The results for 26 mm and 30 mm from the surface are consistent with the plot in Fig. 4.27 as the model does not predict the same loss in \( v \) velocity as the jet gets further from the dielectric surface.
Figure 4.30  Centerline velocity comparison for L-PSJA in quiescent flow under steady operation with $\rho_{c}^{max} = 0.00375 \text{ Cm}^{-3}$.

Figure 4.31  Jet width comparisons at 6 mm and 10 mm from dielectric surface for L-PSJA in quiescent flow under steady operation with $\rho_{c}^{max} = 0.00375 \text{ Cm}^{-3}$.
Figure 4.32  Jet width comparisons at 26 mm and 30 mm from dielectric surface for L-PSJA in quiescent flow under steady operation with $\rho_c^{max}=0.00375$ Cm$^{-3}$.

In common free jet analysis the spread of the jet is characterized by the half-width. The half-width corresponds to the width away from the centerline where the velocity is half the peak velocity value. For the jet results demonstrated in Fig. 4.30, 4.31, 4.32 the half width is given in Fig. 4.33. In this plot we see that the half width of the jet normalized with the interior diameter of the actuator (roughly the embedded electrode width) is linear with the normalized distance from the surface up to approximately $3y/d_i$. This is the typical proportionality of the jet half-width that is characterized by similarity solutions of a turbulent free-wall jet.

4.4.4 Wall Effects

Since the numerical results presented above do not demonstrate the loss observed in $v$ velocity as the jet moves away from the dielectric surface, a wall effect study was conducted. In this study a similar grid was constructed as that of the previous L-PSJA simulations with the addition of a wall boundary condition 15 cm above the dielectric surface. In this simulation the dielectric used is alumina ceramic with a relative permittivity of 9.4 with an input voltage of 5 kV under steady operation.

In Fig. 4.34a $v$ velocity contours are given for a window demonstrative of the experimental PIV window. This figure does not demonstrate any significant losses in $v$ due
Figure 4.33  Normalized half width ($x_{1/2}/d_i$) of jet plotted against normalized distance from the wall $y/d_i$ with $\rho_{c}^{max}=0.00375$ Cm$^{-3}$.

to the wall boundary condition above. In Fig. 4.34b vorticity contours with streamlines are given for the same simulation result. In this figure there are not any noticeable effects due to the addition of the boundary condition. In Fig. 4.35a the centerline $v$ velocity is plotted against the distance from the wall. Here we see that the velocity decreases from approximately $0.38$ ms$^{-1}$ to $0.26$ ms$^{-1}$ which is not as severe as the $v$ velocity loss seen in the experiments. In Fig. 4.35b the $v$ velocity is plotted against $x$ for several $y$ locations above the dielectric surface. This plot shows the same trends observed in the numerical simulations in the previous sections. In Fig. 4.36 the $u$ velocity profiles are plotted for several locations to the left and right of the L-PSJA centerline. In this plot we see that the $u$ velocity is nearly zero at the centerline and that the simulation produces results that are highly symmetrical. These plots conclude that the wall boundary condition does not affect the numerical results nor yields a better match with the experiments.
(a) $v$ velocity contours

(b) Vorticity contours with streamlines

Figure 4.34  $v$ velocity contours and vorticity contours with streamlines.

---

(a) Centerline $v$ velocity

(b) Jet widths

Figure 4.35  Centerline $v$ velocity and jet widths at different distances from the surface.
4.5 Concluding Remarks on Plasma Actuators

The S.–H. model was demonstrated to be successfully integrated into the unstructured grid code UNCLE. Subtle differences between the results obtained in GHOST and the results obtained in UNCLE for \( \rho_c \) and ultimately \( |f_b|/\rho_c^{max}\phi^{max} \) are good sources for future focus in the verification process. More refined grids must also be used to carry out similar simulations to demonstrate “true” grid independence for these quasi-steady test cases.

The results presented in this chapter for the simulations of SDBD plasma actuators demonstrated that the S.–H. DBD model was capable of generating a body force in the vicinity of the electrode with a maximum downward velocity at the electrode interface and the maximum horizontal velocity above the embedded electrode. In these results the lack of jet growth past the embedded electrode demonstrates a weakness of the model. However, these solutions do point out that the inability of the model to generate a significant force distribution beyond the electrode is a by-product of the lack of a significant net charge density and horizontal gradient in the electric field.

The shortcomings of the model pertaining to the lack of body force beyond the embedded electrode prompted an investigation of the L-PSJA plasma actuator. In this configuration the entire body force is located above the embedded electrode; therefore it was more likely to simulate good comparisons with experimental results. For the L-PSJA, results comparing the non-dimensional velocity jets were in very good agreement with the experimental results; the only difference being a slight difference in jet width.
The initial value of $\rho_c^{\text{max}}$ was determined to be too large to match quantitative values from the experiment. A parameter study expanding to both the Debye length and the net charge density resulted in a simulation that matched the peak jet velocity, jet width and jet spreading up to approximately 10 mm from the dielectric surface. Most of the numerical DBD simulations conducted by others tend not to compare results to quiescent flow experiments, but rather with aerodynamic applications. The drawback to this approach is that any significant body force production tangent to the surface in the vicinity of the electrode will demonstrate improvements in aerodynamic performance. Comparing results where the device dictates the flow physics is a more precise and convincing way to demonstrate the capabilities of a model, which was a key aim of this study.
Chapter 5
Numerical Simulations of a “Bumpy” Eppler 398 Airfoil

This chapter provides a discussion of research involved in simulations of bumpy profile airfoils. The inspiration for this work was the profiles of the inflatable wings designed by ILC Dover and constructed for use in the BIG BLUE Project at the University of Kentucky. The simulations presented in this chapter consider the effects of bumps on the surface of an Eppler 398 airfoil. The bumpy E398 airfoil is the base profile for the inflatable/rigidizable wings used in BIG BLUE I and BIG BLUE II and was used in the previously mentioned experimental observations[5, 9, 11]. The presentation which follows includes a discussion of the numerical setup for the simulations, observed oscillations in lift and drag, mean results for lift and drag, effects of transition and turbulent models, and separation reduction.

5.1 Numerical Setup

The setup for the two-dimensional airfoil simulations includes five boundary conditions. The inlet boundary condition is a prescribed constant velocity value. The outlet boundary condition or “outflow” boundary includes setting the velocity gradient to zero or the normal derivative of the velocity to zero. The upper and lower extremities of the grid are set to free stream. On the airfoil surface is the no-slip boundary condition where both components of the velocity are zero. The boundary conditions are summarized in Fig. 5.1.

The mesh is separated into ten grids, including nine background grids and one dense grid around the airfoil overset on the background mesh (numbered in Fig. 5.1). The number of total grid points in the bumpy airfoil mesh is \( \approx 275,000 \). The mesh surrounding the airfoil is the most dense of the ten grids used; it contains \( \approx 100,000 \) grid points. For the smooth airfoil simulations a total of \( \approx 210,000 \) grid points was used with the airfoil grid containing \( \approx 38,000 \) grid points. A summary of the number of grid points for each of the ten meshes used in the simulations is given in Table 5.1. The same background mesh was used for both simulations, which is one of the advantages of using overset grids; changing the density of the overset grid, the dense airfoil grid in this case, does not affect the grid generation process of the background grid. This
feature makes the use of overset grid codes advantageous for optimization projects or moving grid projects where multiple grids are used where, in some cases, more than half of the mesh does not have to be regenerated. The simulations presented in this work range from four days to three weeks for each individual Re-α combination. The laminar simulation results require less computing time since fewer equations are solved than those using Menter’s SST turbulence model and the Suzen–Huang intermittency transport transition model that also uses the SST turbulence model in region where γ >0.

Table 5.1 Summary of the number of grid points for each grid used in bumpy and smooth airfoil simulations.

<table>
<thead>
<tr>
<th>Grid</th>
<th>Bumpy</th>
<th>Smooth</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>102608</td>
<td>37994</td>
</tr>
<tr>
<td>2</td>
<td>7701</td>
<td>7701</td>
</tr>
<tr>
<td>3</td>
<td>60551</td>
<td>60551</td>
</tr>
<tr>
<td>4</td>
<td>15251</td>
<td>15251</td>
</tr>
<tr>
<td>5</td>
<td>4131</td>
<td>4131</td>
</tr>
<tr>
<td>6</td>
<td>32481</td>
<td>32481</td>
</tr>
<tr>
<td>7</td>
<td>8181</td>
<td>8181</td>
</tr>
<tr>
<td>8</td>
<td>4131</td>
<td>4131</td>
</tr>
<tr>
<td>9</td>
<td>32481</td>
<td>32481</td>
</tr>
<tr>
<td>10</td>
<td>8181</td>
<td>8181</td>
</tr>
</tbody>
</table>

As part of the preparation for these simulations, a study was done to determine if the smaller grid (see Fig. 5.2a) demonstrated blocking due to the upper and lower boundary conditions. To do so, two additional grids were constructed with upper, lower and rear boundaries further away from the airfoil geometry (Fig. 5.2b). The grids include vertical heights of approximately 4c, 10c and 12c (c denotes chord length of airfoil). In this construction the number of grid points for the background meshes were increased from the 4c mesh to the values given in Table 5.1 for the 10c and 12c meshes to account for the stretching to larger dimensions. The results obtained for $C_l$ and $C_d$ on the smooth and bumpy Eppler airfoil at Re=25000, $\alpha = 0^\circ$ are given in Table 5.2. In this study we define the lift and drag coefficients as

$$C_l = \frac{L}{1/2\rho U_\infty^2} = \mathbf{j} \cdot \oint_S \left[ -\left( p - p_\infty \right) \mathbf{n} + \tau_{w,t} \right] dS$$

(5.1)
Figure 5.1  Boundary conditions with numbered grids for two-dimensional airfoil simulations.

\[ C_d = \frac{D}{1/2\rho U^2_\infty} = \mathbf{i} \cdot \frac{\oint_S [- (p - p_\infty) \mathbf{n} + \tau_w \mathbf{t}] dS}{1/2\rho U^2_\infty} \]  \hspace{1cm} (5.2)

where \( p \) is the static pressure, \( \mathbf{i} \) is the unit normal in the \( x \) direction, \( \mathbf{j} \) is the unit normal in the \( y \) direction, \( \mathbf{n} \) is the unit normal to the surface, \( \mathbf{t} \) is the unit normal tangent to the surface, and \( \tau_w \) is the shear stress at the surface. These values are obtained by evaluating the integrals around the surface of the airfoil from the flow field results. These results are averaged over, roughly, the last ten non-dimensional time steps. From these values we can see that the smallest grid does show evidence of blocking, increasing the free stream velocity over the top surface and, in this case, effectively increasing the amount of lift of both airfoils. Previous studies[67] show similar results for low-\( Re \) simulations of a NACA4415 airfoil. In that study it was concluded that giving at least 6\( c \) between the upper and lower boundary conditions yields results of similar magnitude and variance for \( C_l \) and \( C_d \).

In addition to a blocking study, a test to see if a significant increase in the airfoil grid density would have any significant effect on the lift and drag results. The bumpy airfoil grid (grid 1) was essentially doubled from roughly 100,000 to 200,000 grid points. This study was done after seeing noise in the pressure coefficient \( C_p \) curves. It was found that by increasing the grid points, the noise in the \( C_p \) curve could be removed near the leading edge of the bumpy airfoil, but was still present in certain areas of the

97
Figure 5.2  Three different size structured grids used.

Table 5.2  Results for grid blocking comparisons at $Re=25000$ (laminar).

<table>
<thead>
<tr>
<th>Grid</th>
<th>Airfoil</th>
<th>$\alpha$</th>
<th>$C_l$</th>
<th>$C_d$</th>
<th>$C_l/C_d$</th>
</tr>
</thead>
<tbody>
<tr>
<td>4c</td>
<td>Bumpy</td>
<td>0°</td>
<td>-0.059</td>
<td>0.067</td>
<td>-0.881</td>
</tr>
<tr>
<td>10c</td>
<td>Bumpy</td>
<td>0°</td>
<td>-0.102</td>
<td>0.063</td>
<td>-1.619</td>
</tr>
<tr>
<td>12c</td>
<td>Bumpy</td>
<td>0°</td>
<td>-0.105</td>
<td>0.063</td>
<td>-1.667</td>
</tr>
<tr>
<td>4c</td>
<td>Smooth</td>
<td>0°</td>
<td>0.236</td>
<td>0.057</td>
<td>4.140</td>
</tr>
<tr>
<td>10c</td>
<td>Smooth</td>
<td>0°</td>
<td>0.105</td>
<td>0.059</td>
<td>1.779</td>
</tr>
<tr>
<td>12c</td>
<td>Smooth</td>
<td>0°</td>
<td>0.103</td>
<td>0.055</td>
<td>1.8727</td>
</tr>
</tbody>
</table>
airfoil. This is due to the change in the grid element shape in the crevice of the bumps. Due to the ADI-type solving technique used in GHOST, the change in shape and size of the elements was too different for the Rhie and Chow momentum interpolation scheme to remove all signs of a checkerboard solution. In unstructured grid codes, such as UNCLE, the Rhie and Chow momentum interpolation is done more carefully since the solution is solved via a Gauss–Seidel point solver. This type of solver allows each point to consider the size and shape differences of all the surrounding cells each time it is solved. In the ADI-type TDMA solver, the upstream and downstream cells in one coordinate direction are considered during each sweep through the matrix. However, even with the checkerboard solution, the values for \( C_l \) and \( C_d \) are very similar to the values where there is not a checkerboard solution present. Using the smaller (less refined) grid to do the simulations saves time and computational effort without a lot of sacrifice in accuracy. Also, doing a “true” grid independence study for the unsteady flow associated with low-\( Re \) simulations of bumpy wings requires averaging the solutions for flow properties over long time cycles to ensure that unique events do not significantly change the solution. This makes grid independence hard to determine conclusively. For this study, resulting values in \( C_l \) and \( C_d \) that are close is good enough to determine whether or not the addition of the bumps is beneficial or not to the Eppler 398 airfoil at low to moderately low Reynolds numbers. For the smooth E398 grid it was found that some irregularity in the grid spacing near the leading edge also added noise to the \( C_p \) plot similar to the effects due to the bumps. Therefore, a more precise construction was done to ensure grid spacing was more gradual. This second grid actually contained fewer grid points than the previous grid, but was constructed more effectively, resulting in the removal of the unwanted noise in the \( C_p \) curve and removing the checkerboard solution in the pressure and velocity fields. The results comparing the lift and drag for this study are found in Table 5.3 and the comparison of the lift curves for the different grids can be found in Fig. 5.3.

In Fig. 5.4 we see the differences in the pressure coefficient, defined as

\[
C_p = \frac{p - p_\infty}{1/2 \rho U_\infty^2}.
\] (5.3)

In Fig. 5.4a a comparison of the two smooth grids used in this study is presented. The figure demonstrates the effects of poor grid construction near the leading edge of the
Table 5.3 Results for grid resolution comparisons at $Re=25000$ (laminar).

<table>
<thead>
<tr>
<th>Grid Points</th>
<th>Airfoil</th>
<th>$\alpha$</th>
<th>$C_l$</th>
<th>$\sigma_{C_l}$</th>
<th>$C_d$</th>
<th>$\sigma_{C_d}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>102608</td>
<td>Bumpy</td>
<td>0°</td>
<td>-0.102</td>
<td>0.080</td>
<td>0.063</td>
<td>0.003</td>
</tr>
<tr>
<td>205146</td>
<td>Bumpy</td>
<td>0°</td>
<td>-0.113</td>
<td>0.097</td>
<td>0.065</td>
<td>0.006</td>
</tr>
<tr>
<td>37994</td>
<td>Smooth</td>
<td>0°</td>
<td>0.105</td>
<td>0.077</td>
<td>0.059</td>
<td>0.006</td>
</tr>
<tr>
<td>34606</td>
<td>Smooth</td>
<td>0°</td>
<td>0.115</td>
<td>0.065</td>
<td>0.055</td>
<td>0.003</td>
</tr>
</tbody>
</table>

Figure 5.3 $C_l$ results for grid resolution comparisons at $Re=25000$ (laminar).
smooth airfoil. The different grid sizes in the mesh lead to the oscillations for grid 1 (the original grid). The results for the second grid (reconstructed grid) show no signs of oscillation. Therefore, careful grid construction can remove the oscillations for the smooth airfoils. If the oscillations are filtered out of the original grid the result is remarkably close to that of the second grid. Therefore, the results for lift and drag for the smooth results are assumed to be approximately equal. In Fig. 5.4b a comparison of the two bumpy grids is given. In this plot the oscillations for the first grid are worse than those in the second grid. However, all of the oscillations are not removed despite the efforts of making a more precisely spaced grid with twice the density. By filtering the small oscillations out of the original bumpy grid we get very close to the same result for the pressure distribution on the lower surface of the airfoil. The upper surface actually experiences less suction for grid 1 than for grid 2. This difference in the pressure coefficient is directly related to the decrease in lift seen in Table 5.3 and Fig. 5.3. This is due to the fact that the lift coefficient can be calculated from the pressure distribution via

$$C_l = \int_{LE}^{TE} C_p^{\text{lower}}(x) - C_p^{\text{upper}}(x) dx.$$  \hspace{1cm} (5.4)

These relatively short wavelength, small amplitude oscillations in the $C_p$ curves presented above are due to the non-uniform growth in the cell size around the airfoil. This directly affects the pressure interpolation scheme given in the details of the Rhie

Figure 5.4 $C_p$ curves for the different bumpy and smooth meshes for $Re=25000$, $\alpha=0^\circ$. 

(a) $C_p$ for smooth grids  \hspace{1cm} (b) $C_p$ for bumpy grids
and Chow method discussed in Ch. 3. Since the pressure interpolation scheme uses the
cell volumes (or areas in two-dimensions) as a weighting constant in the interpolation,
the changes in cell size from small to large and back to small allow for a converged
solution that is oscillatory and non-physical. However, by filtering the $C_p$ curves these
oscillations do not greatly affect the trends seen in the lift or the effects in the pressure
distribution due to the addition of the bumps. A further study was not conducted to
see if a grid with $\approx 200,000$ grid points that are poorly distributed would give a smaller
lift value as that see in the $\approx 100,000$ grid point mesh. Therefore, the loss in lift cannot
be specifically attributed to the grid spacing.

5.2 Model Effects

In this section a discussion of the effects of the flow regime assumptions is presented.
Using GHOST there is the option to choose between a fully laminar simulation, a
transitional simulation using the Suzen–Huang intermittency model, and fully turbulent
simulations using either Menter’s SST turbulence model or the SA turbulence model.
For the fully turbulent simulations the SST model was used. The flow over the bumpy
airfoils is more complex than the flow over smooth airfoils making characterization of
the flow difficult. Part of this study is focused on determining a range of lift and
drag values that can bound the values corresponding to the actual values if the flow
is truly more complex than fully laminar or fully turbulent flow. In this effort, the
S.–H. transition model is used to determine if a correlation-based intermittency model
can produce results that are reasonable for a flow that is characterized by a marginally
large transition region near the surfaces of the bumpy airfoil at $Re=25,000$. Therefore,
laminar, transitional, and turbulent simulations are conducted for the bumpy airfoil at
$Re=25,000$. However, the flow over the bumpy airfoil at $Re=200,000$ is assumed to be
fully turbulent. Future work could include simulations using the S.–H. transition model
at this Reynolds number as well.

In Fig. 5.5 we see an example when the transition model predicts a large region on
the upper surface of the airfoil that has values for $\gamma$ that are at an intermediate value
(i.e. $0.1 < \gamma < 0.9$). The value for $\gamma$ is used to characterize when the model predicts
that the flow is laminar or turbulent. When $\gamma < 0.1$ then the flow is near fully laminar.
Conversely, when $\gamma > 0.9$ then the flow is near fully turbulent. When the values of $\gamma$
fall in intermediate values the flow is being modeled as in transition between laminar and turbulent. The velocity profiles for when the value of $\gamma$ is in these intermediate values will be discussed later. When $\gamma > 0.1$ we will see that the model will begin to smooth the solution in a similar manner that the SST turbulence model does. This is a by-product of the transition model using the SST model for the turbulent quantities such as the eddy viscosity $\mu_T$.

In Fig. 5.6, results for $\gamma$ contours are given for a set of the simulations at $Re=25000$. In Fig. 5.6a we can see that values for $\gamma > 0$ are present along the length of the airfoil on the lower surface and are not present until the separation region on the upper surface. In Fig. 5.6b the angle of attack is increased slightly to $2^\circ$ and the results are similar except that the region where significant values of $\gamma$ is increased above the upper surface as the separation region expands. With an increase of $\alpha$ to $7^\circ$ (see Fig. 5.6c) the intermittency model predicts that a localized section of the separation region is nearly fully turbulent as well as a thin region on the lower surface of the airfoil. In Fig. 5.6d the angle of attack is increased dramatically to $20^\circ$. In this figure we see that the intermittency model predicts transition from the leading edge to the beginning of the separation region. The bulk of the separation region behind the airfoil turns back to $\gamma$ values representative of laminar flow. This is most likely due to the essentially stagnant air in that region. The plots in Fig. 5.6 serve to demonstrate the fact that the S.–H. transition model predicts that the flow is in transition above and below significant portions of the airfoil which agrees with the presumption that the flow over these airfoils is not characterized by fully laminar or fully turbulent flow.

![Figure 5.5](image)

**Figure 5.5** $\gamma$ contour for the bumpy airfoil transitional simulation at $Re=25000$, $\alpha=20^\circ$. 
Figure 5.6  $\gamma$ contours for bumpy E398 airfoil at $Re=25,000$, for several angles of attack.
5.3 Lift and Drag Oscillation Study

Values for $C_l$ and $C_d$ oscillate due to inherent unsteadiness in the flow over these airfoils at relatively low $Re$. This discussion includes plots of $C_l$ and $C_d$ against non-dimensional time (i.e. $t^* = (U_\infty t)/c$). The results for $Re=25,000$ for the smooth E398 airfoil are given in Fig. 5.7 where the values for $C_l$ and $C_d$ oscillate about a mean. This oscillation changes with the angle of attack, typically increasing in magnitude and number of harmonics. As the smooth airfoil reaches stall at 20°, we see very peculiar oscillations in $C_d$; in fact, we see a very large spike at $t^* \sim 12$. This large spike happens later in the cycle as well, at around $t^* \sim 17$ (not seen on the plot). Similar trends are seen in Fig. 5.8 for the laminar bumpy airfoil simulations at $Re=25,000$. In this plot, a large spike in $C_d$ is present around $t^* \sim 12$, but at $\alpha=17^\circ$. Since data was not saved for every time step, the exact phenomenon creating these large spikes in drag are unknown. Previous restart files are available which could be a starting point for new simulations, but due to the unsteadiness in the flow field at high angles of attack, reproducing the phenomena is not guaranteed. In Fig. 5.9 the results for the bumpy airfoil simulated with the S.–H. transition model at $Re=25,000$ are presented. This plot demonstrates oscillations that are much smoother with clearer fundamental frequencies than the simulations computed with the fully laminar assumption. Small sudden changes in the relatively smooth data are also present for a few angles of attack, but have not been investigated further. What is demonstrated in the transition model results is that the oscillations about the mean values for $\alpha$ at 0° and 2° are larger than those at 5-10°; this trend is not observed in the laminar simulations. It is suggested that this smoothing is related to the amount of turbulence present in the separation of the bumpy airfoil, and thus a larger region where the SST model is used. In Fig. 5.10 the results for the bumpy E398 airfoil simulated under the fully turbulent assumption at $Re=25,000$ are given. In these plots the SST turbulence model filters out most of the oscillations in the lift and drag curves (inherent to the averaging in RANS models) for moderate angles of attack. It is not until $\alpha$ reaches 17° that an oscillation of significant magnitude relative to the mean is noticeable.

In Fig. 5.11 the results for the smooth E398 airfoil computed under the fully laminar assumption at $Re=200,000$ are presented. In this plot oscillations in lift and drag are
Figure 5.7 Laminar simulation results for $C_l, C_d$ vs. $t^*$ for smooth E398 airfoil at $Re=25,000$. 
Figure 5.8  Laminar simulation results for $C_l$, $C_d$ vs. $t^*$ for bumpy E398 airfoil at $Re=25,000$. 
Figure 5.9  S.–H. intermittency transition model simulation results for $C_l$, $C_d$ vs. $t^*$ for bumpy E398 airfoil at $Re=25,000$. 

(a) $C_l$ vs. $t^*$

(b) $C_d$ vs. $t^*$
Figure 5.10 Turbulent SST simulation results for $C_l$, $C_d$ vs. $t^*$ for bumpy E398 airfoil at $Re=25,000$. 
fairly uniform until $\alpha=10^\circ$. At the high angles of attack, large variations form at multiple frequencies. This is consistent with the simulations at $Re=25,000$ for the smooth airfoil. Furthermore, the lift value peaks at $\alpha=15^\circ$, but the drag increases indicating that the airfoil has reached stall. In Fig. 5.12, results for the bumpy airfoil simulated at $Re=200,000$ under the fully laminar assumption are presented. In these figures both the lift and drag coefficients are much more complex than the laminar results at $Re=25,000$. These oscillations suggest that the flow at $Re=200,000$ over the bumpy airfoil is not fully laminar, making the solution very unstable. In Fig. 5.13 we see the results for the bumpy airfoil simulations assuming that the flow is fully turbulent. In this figure we see that the oscillations are relatively smooth for both $C_l$ and $C_d$; this is consistent with the results at $Re=25,000$.

Previous work[105] has characterized the unsteadiness in the flow by the Strouhal number based on the airfoil chord length (i.e. $St_c = f_c/U_\infty$). In the previous experimental study of bumpy airfoils[9] it was stated that the flow over the bumpy airfoils was more unsteady than that over the smooth airfoils. Here we want to quantify this unsteadiness via $St_c$ for two sets of simulations at Reynolds numbers of 25,000 and 200,000. In tables 5.4, 5.5, 5.6 the results for $St_c$ are given for the two sets of simulations. In Fig. 5.14 these results are plotted for comparison.

In Fig. 5.14a ($Re=25,000$), the results for the chord based Strouhal number plotted against angle of attack for the bumpy simulations is similar to that of the smooth simulation with the exception of the S.–H. intermittency computations. The Strouhal number for the smooth simulation decreases with angle of attack until it reaches the approximate stall angle at $15^\circ$. Once stall is reached the Strouhal number then decreases again to the last test case of $\alpha=20^\circ$. The laminar bumpy simulation decreases at the approximate slope of the smooth simulation, but at $\alpha=12^\circ$ the slope changes but does not increase. The Strouhal number does not increase with angle of attack for the laminar simulation until the $20^\circ$ simulation. This also assures us that the oscillations in the SST results are not purely numerical, but driven by the unsteadiness of the flow field. The simulation for the S.–H. transition model is similar to the previous simulations until $\alpha=5^\circ$; afterward, it remains approximately constant for $\alpha=5^\circ$-$17^\circ$ then increases to approximately the same value as the turbulent and laminar simulations for the bumpy
Figure 5.11  Laminar simulation results for $C_l$, $C_d$ vs. $t^*$ for smooth E398 airfoil at $Re=200000$. 

(a) $C_l$ vs. $t^*$

(b) $C_d$ vs. $t^*$
Figure 5.12  Laminar simulation results for $C_l$, $C_d$ vs. $t^*$ for bumpy E398 airfoil at $Re=200000$. 

(a) $C_l$ vs. $t^*$

(b) $C_d$ vs. $t^*$
Figure 5.13  SST simulation results for $C_l$, $C_d$ vs. $t^*$ for bumpy E398 airfoil at $Re=200000$. 
airfoil. When the bumpy airfoil is simulated under the fully turbulent assumption the fundamental slope of the St-α curve is approximately the same as the two laminar cases until 12°; the Strouhal number then increases at the 15° mark then decreases to the 20° point. The Strouhal number for the turbulent case is not driven by the same large oscillations in lift and drag as that seen in the laminar and transitional cases; yet, the values are have similar magnitude and slope. This suggests that the SST model is truly smoothing the flow to a mean value removing most of the unsteadiness, but not to an extent where it is undetectable.

In Fig. 5.14b (Re=200,000), the results for the bumpy and smooth simulations are quite different. The two bumpy simulations are similar with the exception of the simulation at α=5°. From these simulations we can see that the oscillations in lift have a fundamentally higher frequency for the smooth simulations than the bumpy until α=12°. This suggests that the flow over the bumpy airfoil is less unsteady than the flow over the smooth airfoil for moderate angles of attack. The Strouhal number results for the bumpy airfoils are just slightly higher than those observed at Re=25,000 whereas the results for the smooth airfoil increase by more than a factor of two. Therefore, if the oscillation in the lift is due to vortex shedding from the trailing edge of the airfoil, it is much more rapid for the smooth airfoil than the bumpy. However, if smoke-wire visualization was used to qualify the unsteadiness, the movement in the upper region of separation can be different than the vortex shedding off the trailing edge. A more focused approach of visualizing unsteadiness in the shear layer that forms at the top of the separation region must be done to quantify the flow as more unsteady from the same viewpoint of smoke-wire visualization.

5.4 Simulation Results for Reynolds Number=25,000

In this section we will discuss the results obtained for the bumpy and smooth airfoils at Re=25,000. The smooth E398 airfoil was simulated under the assumption that the flow was completely laminar. Again, since the nature of flow over the bumpy profiles is still not completely understood and is likely to be partially laminar and partially turbulent, the simulations for the bumpy profiles include laminar, transitional, and SST results in an effort to bracket the possible range of behaviors.

In Table 5.8 the quantities for mean $C_l$, $C_d$ and their standard deviations are given
Table 5.4  Laminar simulation results for $St_c$ of smooth E398 airfoil at $Re=25,000$ and $Re=200,000$.

<table>
<thead>
<tr>
<th>$\alpha$</th>
<th>$St_c$, $Re=25k$</th>
<th>$St_c$, $Re=200k$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$-2^\circ$</td>
<td>3.38</td>
<td>4.90</td>
</tr>
<tr>
<td>0$^\circ$</td>
<td>2.24</td>
<td>4.99</td>
</tr>
<tr>
<td>2$^\circ$</td>
<td>2.21</td>
<td>5.10</td>
</tr>
<tr>
<td>5$^\circ$</td>
<td>1.87</td>
<td>5.46</td>
</tr>
<tr>
<td>7$^\circ$</td>
<td>1.82</td>
<td>5.43</td>
</tr>
<tr>
<td>10$^\circ$</td>
<td>1.27</td>
<td>4.00</td>
</tr>
<tr>
<td>12$^\circ$</td>
<td>0.78</td>
<td>2.24</td>
</tr>
<tr>
<td>15$^\circ$</td>
<td>1.38</td>
<td>1.84</td>
</tr>
<tr>
<td>17$^\circ$</td>
<td>1.10</td>
<td>1.63</td>
</tr>
<tr>
<td>20$^\circ$</td>
<td>0.62</td>
<td>2.61</td>
</tr>
</tbody>
</table>

Table 5.5  Results for $St_c$ of Bumpy E398 airfoil at $Re=25,000$.

<table>
<thead>
<tr>
<th>$\alpha$</th>
<th>$St_c$ (Laminar)</th>
<th>$St_c$ (Trans)</th>
<th>$St_c$ (SST)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$-2^\circ$</td>
<td>1.94</td>
<td>-</td>
<td>2.16</td>
</tr>
<tr>
<td>0$^\circ$</td>
<td>2.02</td>
<td>2.29</td>
<td>2.24</td>
</tr>
<tr>
<td>2$^\circ$</td>
<td>1.93</td>
<td>2.20</td>
<td>2.10</td>
</tr>
<tr>
<td>5$^\circ$</td>
<td>1.74</td>
<td>0.49</td>
<td>1.95</td>
</tr>
<tr>
<td>7$^\circ$</td>
<td>1.63</td>
<td>0.56</td>
<td>1.92</td>
</tr>
<tr>
<td>10$^\circ$</td>
<td>1.33</td>
<td>0.56</td>
<td>1.66</td>
</tr>
<tr>
<td>12$^\circ$</td>
<td>0.82</td>
<td>0.61</td>
<td>1.08</td>
</tr>
<tr>
<td>15$^\circ$</td>
<td>0.72</td>
<td>0.60</td>
<td>1.15</td>
</tr>
<tr>
<td>17$^\circ$</td>
<td>0.73</td>
<td>0.57</td>
<td>1.05</td>
</tr>
<tr>
<td>20$^\circ$</td>
<td>0.86</td>
<td>0.88</td>
<td>0.89</td>
</tr>
</tbody>
</table>

Table 5.6  Results for $St_c$ of Bumpy E398 airfoil at $Re=200,000$.

<table>
<thead>
<tr>
<th>$\alpha$</th>
<th>$St_c$ (Laminar)</th>
<th>$St_c$ (SST)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0$^\circ$</td>
<td>3.42</td>
<td>2.80</td>
</tr>
<tr>
<td>2$^\circ$</td>
<td>-</td>
<td>2.79</td>
</tr>
<tr>
<td>5$^\circ$</td>
<td>1.57</td>
<td>2.53</td>
</tr>
<tr>
<td>7$^\circ$</td>
<td>-</td>
<td>2.36</td>
</tr>
<tr>
<td>10$^\circ$</td>
<td>1.94</td>
<td>2.02</td>
</tr>
<tr>
<td>12$^\circ$</td>
<td>-</td>
<td>1.59</td>
</tr>
<tr>
<td>15$^\circ$</td>
<td>1.41</td>
<td>1.37</td>
</tr>
<tr>
<td>17$^\circ$</td>
<td>-</td>
<td>1.25</td>
</tr>
<tr>
<td>20$^\circ$</td>
<td>1.65</td>
<td>0.98</td>
</tr>
</tbody>
</table>
for the bumpy airfoil for laminar flow at $Re=25,000$ at multiple angles of attack. Similar data is provided for the smooth airfoil in laminar flow in Table 5.7. The results are repeated graphically in Fig. 5.15, with the error bars being their respective standard deviations of the amplitude variation. The results indicate that in all cases the trend is for greater variation in the lift and drag at higher angles of attack, indicating the growing importance of the separation region. Comparing the two laminar results, the values for $C_d$ for both airfoils are quite similar for the range of angles of attack simulated. The slope of the $C_l$-$\alpha$ curve also is similar up to the point of stall. What is different for the two airfoils is the intersection for zero lift. For the bumpy airfoil the intersection is at approximately $1^\circ$ whereas for the smooth airfoil it is approximately at $-1^\circ$ which results in the bumpy airfoil demonstrating less lift for the angles of attack studied under the laminar assumption.

A comparison of the simulations using the transition (Table 5.9) and turbulent models (Table 5.10) is seen in Fig. 5.16. In this figure we can see that the intersection of zero lift for the turbulent simulation is at approximately $-1^\circ$ and that the slope of the $C_l$-$\alpha$ curve is smaller for the turbulent simulations than that of the laminar simulations. The result is that the turbulent assumption results in higher values of lift at lower angles of attack and lower values of lift at higher angles of attack with the crossover occurring at about $10^\circ$. The amount of drag from these two simulations is similar for angles of attack less than $7^\circ$, but at the higher angles of attack the laminar simulations result in higher values of drag. The simulations using Suzen–Huang intermittency transitional
Table 5.7  $C_l$ and $C_d$ results for smooth airfoil at $Re=25,000$ (laminar).

<table>
<thead>
<tr>
<th>$\alpha$</th>
<th>$C_l$</th>
<th>$\sigma_{C_l}$</th>
<th>$C_d$</th>
<th>$\sigma_{C_d}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$-2^\circ$</td>
<td>-0.141</td>
<td>0.121</td>
<td>0.061</td>
<td>0.004</td>
</tr>
<tr>
<td>$0^\circ$</td>
<td>0.105</td>
<td>0.077</td>
<td>0.059</td>
<td>0.006</td>
</tr>
<tr>
<td>$2^\circ$</td>
<td>0.355</td>
<td>0.059</td>
<td>0.069</td>
<td>0.004</td>
</tr>
<tr>
<td>$5^\circ$</td>
<td>0.629</td>
<td>0.089</td>
<td>0.100</td>
<td>0.013</td>
</tr>
<tr>
<td>$7^\circ$</td>
<td>0.703</td>
<td>0.092</td>
<td>0.126</td>
<td>0.015</td>
</tr>
<tr>
<td>$10^\circ$</td>
<td>0.830</td>
<td>0.127</td>
<td>0.172</td>
<td>0.035</td>
</tr>
<tr>
<td>$12^\circ$</td>
<td>1.143</td>
<td>0.232</td>
<td>0.264</td>
<td>0.055</td>
</tr>
<tr>
<td>$15^\circ$</td>
<td>1.535</td>
<td>0.191</td>
<td>0.328</td>
<td>0.086</td>
</tr>
<tr>
<td>$17^\circ$</td>
<td>1.495</td>
<td>0.100</td>
<td>0.317</td>
<td>0.045</td>
</tr>
<tr>
<td>$20^\circ$</td>
<td>1.577</td>
<td>0.285</td>
<td>0.451</td>
<td>0.133</td>
</tr>
</tbody>
</table>

Table 5.8  $C_l$ and $C_d$ results for bumpy airfoil at $Re=25,000$ (laminar).

<table>
<thead>
<tr>
<th>$\alpha$</th>
<th>$C_l$</th>
<th>$\sigma_{C_l}$</th>
<th>$C_d$</th>
<th>$\sigma_{C_d}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$-2^\circ$</td>
<td>-0.242</td>
<td>0.098</td>
<td>0.069</td>
<td>0.007</td>
</tr>
<tr>
<td>$0^\circ$</td>
<td>-0.102</td>
<td>0.080</td>
<td>0.063</td>
<td>0.003</td>
</tr>
<tr>
<td>$2^\circ$</td>
<td>0.084</td>
<td>0.114</td>
<td>0.075</td>
<td>0.004</td>
</tr>
<tr>
<td>$5^\circ$</td>
<td>0.334</td>
<td>0.148</td>
<td>0.099</td>
<td>0.006</td>
</tr>
<tr>
<td>$7^\circ$</td>
<td>0.435</td>
<td>0.158</td>
<td>0.116</td>
<td>0.009</td>
</tr>
<tr>
<td>$10^\circ$</td>
<td>0.625</td>
<td>0.203</td>
<td>0.164</td>
<td>0.019</td>
</tr>
<tr>
<td>$12^\circ$</td>
<td>0.903</td>
<td>0.030</td>
<td>0.227</td>
<td>0.037</td>
</tr>
<tr>
<td>$15^\circ$</td>
<td>1.145</td>
<td>0.299</td>
<td>0.313</td>
<td>0.041</td>
</tr>
<tr>
<td>$17^\circ$</td>
<td>1.259</td>
<td>0.309</td>
<td>0.373</td>
<td>0.084</td>
</tr>
<tr>
<td>$20^\circ$</td>
<td>1.453</td>
<td>0.141</td>
<td>0.270</td>
<td>0.041</td>
</tr>
</tbody>
</table>
model yielded results that generally fell in between the fully turbulent and fully laminar results. It is notable that the transitional lift curve more closely follows the laminar curve, while the drag nearly matches the turbulent curve. In all the bumpy simulations there is no stall region visible up to angles of attack of 20°, unlike the smooth airfoil where stall starts around 15°. Therefore, the numerical simulations indicate that the addition of the bumps on the airfoil surface tend to increase the stall angle of the Eppler 398 airfoil at this low Reynolds number. In Fig. 5.17 we see the $C_l$-$C_d$ polar for all the simulations at $Re=25,000$. In this plot the laminar simulation tends to squeeze the line resulting in less lift at a given drag value. The last point in the bumpy laminar simulation indicates an increase in lift with a decrease in drag which is an unusual result since the induced drag is typically relatively high at corresponding lift values. However, the S.–H. model predict very similar values to the smooth airfoil up to a $C_d$ value of 0.1, afterward the transition model predicts advantages in the lift for a given drag up to $C_d\approx0.29$. The simulations assuming that the flow was fully turbulent also predicted that the bumpy airfoil performed as well as the smooth up to a $C_d$ of approximately 0.1. Afterward, the SST results predicted a gain in performance at corresponding drag values up to approximately 0.225. This is due to the small amounts of drag predicted
from the SST model for the angles of attack simulated.

### Table 5.9 $C_l$ and $C_d$ results for bumpy airfoil at $Re=25,000$ (Trans).

<table>
<thead>
<tr>
<th>$\alpha$</th>
<th>$C_l$</th>
<th>$\sigma_{C_l}$</th>
<th>$C_d$</th>
<th>$\sigma_{C_d}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$0^\circ$</td>
<td>-0.017</td>
<td>0.032</td>
<td>0.054</td>
<td>0.004</td>
</tr>
<tr>
<td>$2^\circ$</td>
<td>0.013</td>
<td>0.027</td>
<td>0.055</td>
<td>0.006</td>
</tr>
<tr>
<td>$5^\circ$</td>
<td>0.369</td>
<td>0.043</td>
<td>0.068</td>
<td>0.004</td>
</tr>
<tr>
<td>$7^\circ$</td>
<td>0.481</td>
<td>0.072</td>
<td>0.080</td>
<td>0.006</td>
</tr>
<tr>
<td>$10^\circ$</td>
<td>0.650</td>
<td>0.075</td>
<td>0.104</td>
<td>0.007</td>
</tr>
<tr>
<td>$12^\circ$</td>
<td>0.865</td>
<td>0.045</td>
<td>0.126</td>
<td>0.007</td>
</tr>
<tr>
<td>$15^\circ$</td>
<td>1.053</td>
<td>0.073</td>
<td>0.174</td>
<td>0.018</td>
</tr>
<tr>
<td>$17^\circ$</td>
<td>1.156</td>
<td>0.079</td>
<td>0.209</td>
<td>0.024</td>
</tr>
<tr>
<td>$20^\circ$</td>
<td>1.251</td>
<td>0.078</td>
<td>0.289</td>
<td>0.028</td>
</tr>
</tbody>
</table>

### Table 5.10 $C_l$ and $C_d$ results for bumpy airfoil at $Re=25,000$ (SST).

<table>
<thead>
<tr>
<th>$\alpha$</th>
<th>$C_l$</th>
<th>$\sigma_{C_l}$</th>
<th>$C_d$</th>
<th>$\sigma_{C_d}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$-2^\circ$</td>
<td>-0.074</td>
<td>0.000</td>
<td>0.054</td>
<td>0.000</td>
</tr>
<tr>
<td>$0^\circ$</td>
<td>0.084</td>
<td>0.000</td>
<td>0.053</td>
<td>0.011</td>
</tr>
<tr>
<td>$2^\circ$</td>
<td>0.206</td>
<td>0.001</td>
<td>0.055</td>
<td>0.000</td>
</tr>
<tr>
<td>$5^\circ$</td>
<td>0.403</td>
<td>0.001</td>
<td>0.065</td>
<td>0.000</td>
</tr>
<tr>
<td>$7^\circ$</td>
<td>0.495</td>
<td>0.000</td>
<td>0.078</td>
<td>0.000</td>
</tr>
<tr>
<td>$10^\circ$</td>
<td>0.651</td>
<td>0.002</td>
<td>0.101</td>
<td>0.000</td>
</tr>
<tr>
<td>$12^\circ$</td>
<td>0.772</td>
<td>0.004</td>
<td>0.113</td>
<td>0.000</td>
</tr>
<tr>
<td>$15^\circ$</td>
<td>0.899</td>
<td>0.002</td>
<td>0.145</td>
<td>0.021</td>
</tr>
<tr>
<td>$17^\circ$</td>
<td>1.007</td>
<td>0.009</td>
<td>0.176</td>
<td>0.021</td>
</tr>
<tr>
<td>$20^\circ$</td>
<td>1.114</td>
<td>0.013</td>
<td>0.232</td>
<td>0.002</td>
</tr>
</tbody>
</table>

In Fig. 5.18 we compare the set of simulations for the set of angles of attack by plotting $C_l/C_d$ vs. $\alpha$. This plot reveals that the smooth airfoil actually has a better lift to drag ratio over most of the angles of attack studied when compared to the laminar bumpy simulations. However, due to the reduction in drag in the fully turbulent and transitional simulations the bumpy profile actually has a better lift to drag ratio assuming turbulent effects for higher angles of attack. Both the transitional and turbulent curves exhibit a peak around $12^\circ$, after which they fall off steadily. These results suggest that the more turbulence the bumps are able to create at these low Reynolds numbers, the greater the reduction in drag and the higher the lift-drag ratio at low Reynolds numbers. However, if the flow remains largely laminar or the angle of attack is low, the
Figure 5.16 $C_l$ and $C_d$ vs. $\alpha$ for all simulations at $Re=25,000$.

Figure 5.17 $C_r$-$C_d$ polar for all simulations at $Re=25,000$. 
bumpy airfoil will tend to cause a reduction in aerodynamic performance.

![Graph showing Cl/Cd for bumpy and smooth airfoils at Re=25,000.](image)

**Figure 5.18** $C_l/C_d$ for bumpy and smooth airfoils at $Re=25,000$.

5.4.1 Separation

As mentioned briefly in the introduction, wind tunnel experiments demonstrated a reduction in the amount of separation on the upper surface due to the addition of the bumps. It is also good to note that looking at streamlines that are more attached should not necessarily imply that the performance of the airfoil is improved in terms of lift and drag; although, often it is the case. Reducing separation is certainly an advantage in when implementing control surfaces for example. A set of simulation results presenting streamlines, vorticity contours, $u/U_\infty$ contours, $u/U_\infty$ profiles, and gamma profiles is presented in this section.

In Fig. 5.19, time-averaged streamlines are plotted for the bumpy and smooth airfoil at $Re=25,000$ and $\alpha=7^\circ$. In this figure we can see the difference in the size of the separation region as characterized by the streamlines. The smooth and bumpy laminar simulations have a similar size separation region above the upper surface. The streamlines from the S.-H. transition model simulations and the streamlines from the SST turbulent simulations are similar. The separation region predicted by the two latter
models is smaller than predicted by the laminar simulation. The strength of the vortex structures are implied by the vorticity plots given in Fig. 5.20, 5.21, 5.22, 5.23. The magnitudes of the velocities in the regions where profiles were not extracted can be inferred from the $u/U_\infty$ contour plots that accompany the vorticity contour plots previously listed. In all these plots we see a well defined shear layer at the upper part of the separation region with additional concentrations where strong vortices are present. In the turbulent simulation the shear layer is smeared and does not form as far downstream as that seen in the other bumpy airfoil simulations. Time averaged velocity profiles at 10-50\%c are given in Fig. 5.31 which plot normalized distance from the airfoil surface, $y/c$, and normalized velocity, $u/U_\infty$. In Fig. 5.24 we can see velocity profile comparisons extracted from the time averaged flow fields at 10-50\%c locations. In this plot we can see that the thickness of the boundary layer near the leading edge at 10-40\%c for the bumpy and smooth airfoils are similar in thickness when fully laminar flow is assumed for the bumpy airfoil. At 50\%c the bumpy airfoil does not exhibit any separation (or flow reversal), but the smooth airfoil does exhibit separation. The results for the velocity profiles for the transition and turbulent model are strikingly similar. This implies that the transition model is using the SST turbulence model near the surface. In Fig. 5.25 we can see the values for $\gamma$ at the different chord locations at the same $y/c$ distances from the surface as the velocity profiles in Fig. 5.24. In this plot we can see that the maximum values of $\gamma$ for each chord location happen near the surface. The highest value for $\gamma$ is recorded at the 50\%c location, but all five locations have $\gamma > 0.25$ near the surface; values of $\gamma$ of this magnitude imply that the flow is not fully turbulent nor fully laminar within the boundary layer.

In Fig. 5.26 time averaged streamlines are plotted for the bumpy airfoil at $Re=25,000$ and $\alpha=20^\circ$. In this figure the three models used can be compared by their prediction of the separation region above the upper surface which is present at this moderately large angle of attack. We can also see that the smooth airfoil has a well defined separation region near the leading edge of the airfoil. The flow then reattaches downstream past mid chord. When the bumpy airfoil is simulated with the fully laminar assumption it results in a smaller separation region near the leading edge and a formation of a vortex near half chord. The simulation with the S.–H. transition model predicts a large
Figure 5.19  Time averaged streamlines at $Re=25,000$, $\alpha=7^\circ$. 
Figure 5.20  Time averaged results for smooth E398 airfoil at $Re=25,000$, $\alpha=7^\circ$ (Laminar).
Figure 5.21  Time averaged results for bumpy E398 airfoil at $Re=25,000$, $\alpha=7^\circ$ (Laminar).
Figure 5.22  Time averaged results for bumpy E398 airfoil at $Re=25,000$, $\alpha=7^\circ$ (Transition).
Figure 5.23  Time averaged results for bumpy E398 airfoil at $Re=25,000$, $\alpha=7^\circ$ (SST).
Figure 5.24  Time averaged velocity profiles at $Re=25,000$, $\alpha=7^\circ$.

Figure 5.25  $\gamma$ profiles for the bumpy airfoil transitional simulation at $Re=25,000$, $\alpha=7^\circ$. 
separation region with a distinct laminar separation bubble followed by vortex shedding from the trailing edge. The fully turbulent simulation with Menter’s SST model smears the separation region and does not predict the vortex formation on the upper surface of the airfoil near mid-chord. In this figure we can clearly see that the height of the region where the flow is reversed is less for the bumpy airfoil than the smooth airfoil under the fully laminar assumption at all chord locations. In fact we can see the evidence of reattachment on the bumpy profile at the 40% and 50% locations. When the S.–H. intermittency transition model is used, the flow reversal is not present at the 10% and 20% locations and the velocity profiles also show that the flow does not undergo as severe changes as it does in the laminar profiles. The SST results are similar to the S.–H. results in predicting the location where separation occurs, but the velocity profiles have a slightly flatter slope.

In Fig. 5.32 we see the values of $\gamma$ for the chord locations discussed in the previous plot. We see that the value for $\gamma$ reaches its peak value for the 50% location. We can also see that the peak value for $\gamma$ decreases as the chord location approaches the leading edge. This implies that the transition model predicts the flow is more turbulent the further down the airfoil we extract the locations. The fact that the velocity profiles for the turbulent and transition model are so similar is a product of $\gamma$ having a significant value for all the chord locations considered in the boundary layer region. The strength of the vortex structures are implied by the vorticity plots given in Fig. 5.27, 5.28, 5.29, 5.30. The magnitudes of the velocities in the regions where profiles were not extracted can be inferred from the $u/U_\infty$ contour plots that accompany the vorticity contour plots. For the smooth simulation, Fig. 5.27, we see a very peculiar vortex formation past the trailing edge of the airfoil. The vorticity plot for this simulation assures that this structure is relatively weak compared to that of the shear layer and the concentration of vorticity at the trailing edge. The vorticity contour for the bumpy laminar simulation in Fig. 5.28 demonstrates that the standing vortex structure on the upper surface does not have relatively strong values of vorticity, but the velocity contour implies that flow is reversed (not stagnant) in that region on average. The plots for the transition model and the turbulence model are similar in Fig. 5.29, 5.30 with the turbulent model smearing the separation region to a larger extent; this is demonstrated
in all three plots.

Figure 5.26  Time averaged streamlines at $Re=25,000, \alpha=20^\circ$.

5.4.2 Vortex Shedding and Lift Oscillations

In this section the vortex shedding phenomena and vorticity concentrations will be discussed as it relates to the oscillation in the lift of a bumpy and smooth airfoil at $Re = 25,000, \alpha = 10^\circ$. In Fig. 5.33 we see that four locations are labeled in each of the $C_l$ vs. $t^*$ curves for the smooth and bumpy airfoils. At these locations different phenomena associated with the vortex formations and concentrations in vorticity are observed. Both of the simulations are fully laminar so that we can compare the vortex shedding without any smearing due to significant values of $\gamma$ or assuming the flow field in fully turbulent.

In Fig. 5.34, plots of the streamlines for these two airfoils are seen from 25%$c$ to 115%$c$ into the wake of the airfoil. In Fig. 5.34 we see that when the lift is at its local minimum we have a vortex at the trailing edge of both airfoils. As the lift increases
Figure 5.27  Time averaged results for smooth E398 airfoil at $Re=25,000$, $\alpha=20^\circ$ (Laminar).
Figure 5.28  Time averaged results for bumpy E398 airfoil at $Re=25,000$, $\alpha=20^\circ$ (Laminar).
Figure 5.29  Time averaged results for bumpy E398 airfoil at $Re=25,000$, $\alpha=20^\circ$ (Transition).
Figure 5.30  Time averaged results for bumpy E398 airfoil at $Re=25,000$, $\alpha=20^\circ$ (SST).
Figure 5.31  Time averaged velocity profiles at $Re=25,000$, $\alpha=20^\circ$.

Figure 5.32  $\gamma$ profiles for the bumpy airfoil transitional simulation at $Re=25,000$, $\alpha=20^\circ$. 
we see the absence of that vortex. As the lift increases to its local maximum, only
two relatively large vortex structures are seen on the upper surface of both the smooth
and bumpy airfoils. This implies that the lift variation is directly related to the vortex
shedding that is observed for simulations of airfoils at low Reynolds numbers. As the
lift reaches its local maximum, the vortex structures are largely the same as they are
in the intermediate location between the minimum and maximum location. When the
lift begins to decrease the formation of a vortex at the trailing edge is observed on the
smooth airfoil; the uppermost streamline on both the smooth and bumpy airfoils is now
detached from the trailing edge as well. This phenomena is also true when the lift is
at its local minimum. From the uppermost streamline we can see that the region of
separation for the two airfoils is largest when they experience the least amount of lift.
We can also see that the separation region for the bumpy airfoil is less than that of the
smooth aifoil for all locations which is consistent with the experimental observations
mentioned previously. It has been noted in previous work [105] that the vortex shedding
is a mechanism that serves to reattach the flow to the airfoil surface and when the
uppermost streamline reattaches to the airfoil surface as seen in the points labeled “2”
and “3”, we see an increase in lift. Figure 5.35 demonstrates the strength of the vortices
that form on the upper surface of both the smooth and bumpy airfoils. This confirms
that the strength of the vorticities near the trailing edge of these airfoils is strong relative
to the values of vorticity that are seen in the shear layer that originates at the point of
separation.

5.5 Simulation Results for Reynolds Number=200,000

Increasing the Reynolds number by, roughly, an order of magnitude changes the flow
over these types of airfoils. By increasing the Reynolds number, the affect of decreasing
significant altitude in flight can be compared. In Table 5.11 we see the results for the
simulation of the smooth airfoil for a range of angles of attack at $Re=200,000$. Here we
see that the standard deviation associated with $C_l$ and $C_d$ again generally increases with
angle of attack. We also see that the smooth airfoil reaches stall in the neighborhood
of 17-20° angle of attack. The laminar (Table 5.12) and turbulent (Table 5.13) bumpy
simulations are shown graphically in Fig. 5.36. We see that in the bumpy simulations
the $C_l$ is greater for the laminar simulations, but there is not a cross over as seen in the
Figure 5.33  $C_l$ vs. $t^*$ for smooth and bumpy E398 airfoil at $Re=25,000$, $\alpha = 10^\circ$

Figure 5.34  Streamlines for smooth and bumpy E398 airfoil at $Re=25,000$, $\alpha = 10^\circ$
Figure 5.35  Vorticity contours overlaid with streamlines for smooth and bumpy E398 airfoil at $Re=25,000$, $\alpha = 10^\circ$
$Re=25,000$ simulations. The amount of drag generated for both simulations is similar. However, in Table 5.13 the bumpy airfoil demonstrates an increase in the amount of drag up to the largest angle of attack studied, while $C_l$ does not decrease. In Fig. 5.37 we see the lift-drag polar for all the simulations at $Re=200,000$. In this plot the additions of the bumps tend to squeeze the relationship between $C_l$ and $C_d$. This effectively reduces the amount of lift at any given drag value in the plots. By simulating the bumpy wings with the SST turbulence model, the line is squeezed more than the laminar simulation due to the predicted loss in lift.

Table 5.11  $C_l$ and $C_d$ results for smooth airfoil at $Re=200,000$ (laminar).

<table>
<thead>
<tr>
<th>$\alpha$</th>
<th>$C_l$</th>
<th>$\sigma_{C_l}$</th>
<th>$C_d$</th>
<th>$\sigma_{C_d}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$-2^o$</td>
<td>0.275</td>
<td>0.044</td>
<td>0.018</td>
<td>0.003</td>
</tr>
<tr>
<td>$0^o$</td>
<td>0.401</td>
<td>0.042</td>
<td>0.025</td>
<td>0.004</td>
</tr>
<tr>
<td>$2^o$</td>
<td>0.648</td>
<td>0.041</td>
<td>0.030</td>
<td>0.005</td>
</tr>
<tr>
<td>$5^o$</td>
<td>0.867</td>
<td>0.024</td>
<td>0.046</td>
<td>0.004</td>
</tr>
<tr>
<td>$7^o$</td>
<td>0.913</td>
<td>0.024</td>
<td>0.072</td>
<td>0.005</td>
</tr>
<tr>
<td>$10^o$</td>
<td>1.134</td>
<td>0.052</td>
<td>0.110</td>
<td>0.014</td>
</tr>
<tr>
<td>$12^o$</td>
<td>1.603</td>
<td>0.073</td>
<td>0.125</td>
<td>0.023</td>
</tr>
<tr>
<td>$15^o$</td>
<td>1.807</td>
<td>0.071</td>
<td>0.153</td>
<td>0.025</td>
</tr>
<tr>
<td>$17^o$</td>
<td>1.859</td>
<td>0.076</td>
<td>0.185</td>
<td>0.028</td>
</tr>
<tr>
<td>$20^o$</td>
<td>1.782</td>
<td>0.088</td>
<td>0.288</td>
<td>0.036</td>
</tr>
</tbody>
</table>

Table 5.12  $C_l$ and $C_d$ results for bumpy airfoil at $Re=200,000$ (laminar).

<table>
<thead>
<tr>
<th>$\alpha$</th>
<th>$C_l$</th>
<th>$\sigma_{C_l}$</th>
<th>$C_d$</th>
<th>$\sigma_{C_d}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$0^o$</td>
<td>0.277</td>
<td>0.092</td>
<td>0.049</td>
<td>0.010</td>
</tr>
<tr>
<td>$5^o$</td>
<td>0.764</td>
<td>0.106</td>
<td>0.075</td>
<td>0.015</td>
</tr>
<tr>
<td>$10^o$</td>
<td>1.005</td>
<td>0.102</td>
<td>0.118</td>
<td>0.023</td>
</tr>
<tr>
<td>$15^o$</td>
<td>1.394</td>
<td>0.144</td>
<td>0.179</td>
<td>0.051</td>
</tr>
<tr>
<td>$20^o$</td>
<td>1.533</td>
<td>0.057</td>
<td>0.232</td>
<td>0.042</td>
</tr>
</tbody>
</table>
Table 5.13  \( C_l \) and \( C_d \) results for bumpy airfoil at \( Re=200,000 \) (SST).

<table>
<thead>
<tr>
<th>( \alpha )</th>
<th>( C_l )</th>
<th>( \sigma_{C_l} )</th>
<th>( C_d )</th>
<th>( \sigma_{C_d} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0(^\circ)</td>
<td>0.127</td>
<td>0.006</td>
<td>0.041</td>
<td>0.001</td>
</tr>
<tr>
<td>2(^\circ)</td>
<td>0.261</td>
<td>0.009</td>
<td>0.045</td>
<td>0.001</td>
</tr>
<tr>
<td>5(^\circ)</td>
<td>0.466</td>
<td>0.013</td>
<td>0.055</td>
<td>0.001</td>
</tr>
<tr>
<td>7(^\circ)</td>
<td>0.557</td>
<td>0.025</td>
<td>0.070</td>
<td>0.001</td>
</tr>
<tr>
<td>10(^\circ)</td>
<td>0.712</td>
<td>0.027</td>
<td>0.092</td>
<td>0.002</td>
</tr>
<tr>
<td>12(^\circ)</td>
<td>0.850</td>
<td>0.011</td>
<td>0.102</td>
<td>0.001</td>
</tr>
<tr>
<td>15(^\circ)</td>
<td>0.988</td>
<td>0.022</td>
<td>0.134</td>
<td>0.003</td>
</tr>
<tr>
<td>17(^\circ)</td>
<td>1.095</td>
<td>0.026</td>
<td>0.164</td>
<td>0.004</td>
</tr>
<tr>
<td>20(^\circ)</td>
<td>1.256</td>
<td>0.047</td>
<td>0.227</td>
<td>0.011</td>
</tr>
</tbody>
</table>

Figure 5.36 \( C_l/C_d \) for bumpy and smooth airfoils at \( Re=200,000 \).
The time average streamlines are given for the $Re=200,000$ case for the smooth airfoil (laminar) and the bumpy airfoil simulations (laminar and turbulent) in Fig. 5.39, 5.40, 5.41. In Fig. 5.39, a demonstration of the laminar separation bubble is present near the leading edge. The vorticity contour hints at low frequency vortex shedding from the shear layer rolling up at approximately half chord. The velocity profiles for this case plotted in Fig. 5.38 indicate that separation happens in the vicinity of $30\%c$. The laminar bumpy velocity profiles indicate that there is leading edge separation at $10\%c$, but the turbulent bumpy profiles do not indicate significant separation at any of the chord lengths where the velocity was extracted. The vorticity contours for the laminar bumpy simulations (Fig. 5.40) also hint at the vortex shedding observed in the laminar smooth simulations. The turbulent vorticity contours and velocity fields demonstrate a well defined separation region.
Figure 5.38  Time averaged velocity profiles at $Re=200,000, \alpha=20^\circ$. 
Figure 5.39  Time averaged results for smooth E398 airfoil at $Re=200,000$, $\alpha=20^\circ$ (Laminar).
Figure 5.40  Time averaged results for bumpy E398 airfoil at $Re=200,000$, $\alpha=20^\circ$ (Laminar).
Figure 5.41 Time averaged results for bumpy E398 airfoil at $Re=200,000, \alpha=20^\circ$ (SST).
5.6 Concluding Remarks on Bumpy Airfoil Simulations

The results presented in this chapter demonstrated that the amount of separation present on the upper surface was reduced due to the additions of the bumps on the Eppler 398 airfoil which is consistent with experimental observations. However the reduction in separation on the upper surface of the airfoil by the addition of the bumps did not increase the performance of the airfoil from a perspective of increasing the lift and reducing drag. For all the test cases studied the bumpy airfoils demonstrated less lift than its smooth counterpart, but situations arose where drag was similar or lower. The S.–H. intermittency transport equation transition model predicted a transition region on the upper and lower surface of the bumpy airfoil for a range of angles of attack at $Re=25,000$. This demonstrated that the model predicted a neither fully laminar nor fully turbulent flow over the airfoil. In all the simulations presented, the transition model and turbulent model results reduced the amount of lift and drag of the bumpy airfoils from the fully laminar simulations. The oscillations in lift were characterized by the chord based Strouhal number for the simulations presented. From these results, the flow over the bumpy airfoil tended to exhibit smaller values of $St_c$ which implies that the vortex shedding frequency associated with lift oscillations is less for the bumpy airfoils than for the smooth airfoils. However, unsteadiness in the shear layer due to the bumps could possibly cause the flow over the bumpy airfoils to appear more unsteady. The use of inflatable wings that possess this bumpy airfoil technology have been proved capable of flying at Reynolds numbers $O(100,000)$ in low altitudes. Therefore, if the lift generated by the current inflatable wing technology is sufficient for UAV flight, then additional weight added to make the surface smooth is unnecessary. The separation reduction on the upper surface due to the bumps produces a flow field where conventional roll control devices can be used with good authority (perhaps better than that of the smooth airfoil).
In this chapter, a summary of the present research will be given. We will then present a list of conclusions drawn from the research. Finally, a brief note on future work that can be done as a consequence of this research is included.

6.1 Summary and Conclusions

In chapter 1 an introduction into the topic of flow control was presented. An outline of the differences between active and passive flow control were discussed and presented in graphical fashion. In addition to the discussion on flow control a discussion of some of the concepts used in the following chapters were discussed.

Chapter 2 discussed background information regarding plasma discharges and included a discussion about some of the basic principles of plasma physics. A literature review was then presented based on experimental studies using linear dielectric barrier discharge plasma actuators. Newer plasma actuator devices such as plasma synthetic jet actuators and linear plasma synthetic jet actuators were then discussed. Recent numerical efforts to model these DBD plasma actuators was presented. In addition to the discussion of DBD plasma actuators, literature relevant to the study of low-$Re$ airfoils was presented. The previous research of experimentalists dealing with smooth and bumpy Eppler 398 airfoils was discussed. Concluding this chapter was a discussion of the research done by the UK CFD Group as it applies to active flow control devices for low-$Re$ airfoil flow control.

In the third chapter the computational tools were discussed. This included a detailed description of both codes used in this work. It also included details associated with the discretization of the unstructured code. Details of Menter’s SST turbulence model and the Suzen–Huang intermittency transport model were also presented. The computer resources of the University of Kentucky Cluster Fluid Dynamics group used for these studies concluded this chapter.

The fourth chapter begins with the derivation of the Suzen–Huang DBD plasma actuator model. After the details of the governing equations of model is given, the boundary conditions are discussed. The input parameters used in previous research of
Suzen and Huang[48] is included. The additions to UNCLE are then given, including blocking for different materials and domains, numerical scheme details for the additional partial differential equations solved, and a pseudo-code algorithm. The main subroutines added to UNCLE to implement the model are included in the appendix. Different from results presented by other authors is a comparison of quiescent flow configurations where the plasma actuator devices drive the fluid flow. The first test case is the single linear dielectric barrier discharge plasma actuator. The numerical results for the potential due to the electric field and the net charge density distributions are compared to the previous results of Suzen and Huang[48]. The results from UNCLE closely match the results of Suzen and Huang’s previously published results with GHOST. For the linear actuator, the study concluded that the model produced relatively good results for the SDBD plasma actuator effects for locations above the embedded electrode, but did not capture the effects downstream of the device. The results are good enough to demonstrate flow control for low-Re simulations on aerodynamic surfaces such as airfoils or turbine blades due to the addition of momentum at the dielectric surface.

The second test case presented focused on comparisons to the experimental results of the linear plasma synthetic jet actuator (L-PSJA). This study compared numerical and experimental results via normalized $v$ velocity jet widths and vorticity contours. A parametric study was then conducted to see if the model could produce jet thickness values closer to the experimental results than the baseline configuration. This study was successful in producing numerical results that matched experiments well up to a height of 10 mm above the dielectric surface. The experimental results predicted greater loss in centerline velocity than the numerical computations at greater heights above the dielectric surface. Thus, a study including a mesh with a wall boundary condition 15 cm above the devices was done more closely resembling the experiments. This study did not yield interactions between the upper wall boundary condition and the centerline jet velocity or jet width.

The fifth chapter discussed the simulations of a bumpy airfoil characteristic of an inflatable Eppler 398 wing. This chapter begins with a boundary condition blocking study of the the bumpy and smooth airfoils used. A detailed discussion of the results for lift and drag followed for the two Reynolds numbers studied. In this study, the effects
of the bumps on the upper and lower surface of the E398 airfoil tended to decrease the amount of separation while also decreasing the amount of lift. The amount of drag for the bumpy airfoils was not significantly higher than that of the smooth airfoil, contrary to the effects of conventional surface roughness. The oscillations in the, laminar and transition model, lift and drag curves were due to relatively low frequency vortex shedding off of the trailing edge of the airfoils at the low Reynolds numbers studied. The oscillations in the fully turbulent results for lift and drag, although extremely small in amplitude, are also due to the flow physics since chord based Strouhal numbers were similar to the laminar simulations.

6.2 Future work

Current efforts to implement the Suzen–Huang Model in three-dimensions in UNCLE have resulted in preliminary results for a coarse three-dimensional grid. This grid is an extrusion of a coarse two-dimensional grid. The Navier–Stokes computations were not computed due to the lack of refinement in the grid, so only the electromagnetic equations are solved. These results demonstrate that the model has been implemented in three-dimensions for hexahedral cells. One of the advantages of this model is its ability to be implemented in three-dimensions since it does not contain any boundary conditions that are, strictly speaking, two-dimensional. More simulations will be computed for the linear, three-dimensional SDBD plasma actuator when the two-dimensional version is verified more thoroughly with experimental results similar to those in the previous discussion. The primary goal of implementing the S–H. model into UNCLE is to simulate three-dimensional test cases such as those seen in the PSJAs where two-dimensional simulations will not suffice in predicting the amount of mass entrained by the annular configuration.

The current implementation and of the S.–H. plasma actuator model into the unstructured grid code UNCLE allows for future studies of DBD plasma actuators in any fundamentally two-dimensional aerodynamic configuration. Copying of the subroutines added to the version of UNCLE with the S.–H. DBD model are implemented to a newer version of UNCLE is needed for three-dimensional simulations. The bookkeeping associated with the added variables for the solver and the MPI passing protocols are also needed when adding to a new version of UNCLE.
Figure 6.1  Coarse grid and results for Eq. 4.9, 4.14 and $\frac{|f_{b}|}{\rho_{c}^{max} \frac{\phi_{max}}{\rho_{c}^{max}}}$. 
Future simulations with bumpy wing profiles in two dimensions and bumpy wings in three dimensions will be ongoing with the production of experimental data for quantitative comparison. The NACA 4318 airfoil is the baseline airfoil for many of the more current inflatable wings used in the BIG BLUE project. Grid generation and several test cases at $Re=18000$ and $Re=36000$ have been simulated using UNCLE and the MATLAB script written by Innes[65]. The grid generation for any additional bumpy NACA-4 digit airfoil can be done using the MATLAB script given in the study by Innes[65], based on data obtained by measurements, or from the molds used to manufacture the wings. Grid generation for the smooth NACA 4318 airfoil has been done for use in GHOST. Bumpy NACA 4318 airfoil grid generation is needed for GHOST because of the difference in computational effort required and since GHOST contains the S.–H. intermittency transport transition model which was used for the bumpy Eppler 398 in this study.

This work demonstrates the capabilities and the limitations of the S.–H. DBD plasma actuator model while used in UNCLE. The implementation of this model into UNCLE is a solid foundation for future work dealing with plasma actuator devices. Further validation and comparisons with experimental data can be done in coordination with experimentalists in the field. The simulations of the bumpy Eppler 398 airfoil in GHOST using the S.–H. transition model are the first set of simulations using a transition model on inflatable wing airfoils. This model has been validated for use with low-pressure turbine blades at Reynolds numbers in the same range, but not for airfoils.
Appendix

Additions to UNCLE for S.–H. DBD Plasma Actuator Model

As mentioned in the pseudo-code algorithm in Chapter 4 the gradients of the two potential are first calculated after the node and vertex data is determined from initial values or from a restart file. The gradients for two dimensional simulations are calculated using the following subroutine, “gradients_phi_2d”:

```fortran
SUBROUTINE gradients_phi_2d(internal, node, nnode, cell, cell_phi, ncell, vertex, bc, nbc, my_rank)
    INTEGER :: iface, nbc, ibc, v1, v2, v3, v4, p1, p2, ncell, icell, nbblock, iblock, &
               nnode, itr, ntransfer, ii, istart, iend, &
               my_rank, request_no, ierr, to_process, from_process, tag_send, tag_receive
    TYPE (cell_faces), DIMENSION (:) :: bc
    TYPE (points), DIMENSION (: ) :: node
    TYPE (cells), DIMENSION (: ) :: cell
    TYPE (cell_voltage), DIMENSION (:) :: cell_phi
    TYPE (vpoints), DIMENSION (: ) :: vertex
    TYPE (cell_faces) :: internal
    REAL (high) :: a1, a2, a3, ex, ey, nx, ny, dphi1de, dphi1dn, dphi2de, dphi2dn
    REAL (high) :: phi1_f, phi2_f, dphi1, dphi2, phi, x1, x2, xc, y1, y2, yc
    type(transfer_var_4), dimension(:), pointer :: data_var, data_var_send
    type(transfer_var_2), dimension(:), pointer :: data_var1, data_var1_send
    INTEGER status (mpi_status_size)
    LOGICAL :: logic, logic1

    cell_phi(1:ncell)%dphi1dx = 0._high
    cell_phi(1:ncell)%dphi1dy = 0._high
    cell_phi(1:nnode)%phi1_max = node(1:nnode)%phi1
    cell_phi(1:nnode)%phi1_min = node(1:nnode)%phi1
    cell_phi(1:ncell)%dphi2dx = 0._high
    cell_phi(1:ncell)%dphi2dy = 0._high
    cell_phi(1:nnode)%phi2_max = node(1:nnode)%phi2
    cell_phi(1:nnode)%phi2_min = node(1:nnode)%phi2

    DO iface = 1, internal%nfaces
        p1 = internal%face(iface)%p1
        p2 = internal%face(iface)%p2
        v1 = internal%face(iface)%v(1)
        v2 = internal%face(iface)%v(2)
        a1 = internal%face(iface)%a(1)
        a2 = internal%face(iface)%a(2)
        phi1_f = (node(p1)%phi1*cell(p2)%vol+node(p2)%phi1*cell(p1)%vol)/&
                 (cell(p1)%vol+cell(p2)%vol)
        cell_phi(p1)%dphi1dx = (cell_phi(p1)%dphi1dx+phi1_f*a1)
        cell_phi(p1)%dphi1dy = (cell_phi(p1)%dphi1dy+phi1_f*a2)
        cell_phi(p2)%dphi1dx = (cell_phi(p2)%dphi1dx-phi1_f*a1)
        cell_phi(p2)%dphi1dy = (cell_phi(p2)%dphi1dy-phi1_f*a2)
        cell_phi(p1)%phi1_max = max(cell_phi(p1)%phi1_max, node(p2)%phi1)
        cell_phi(p1)%phi1_min = min(cell_phi(p1)%phi1_min, node(p2)%phi1)
        cell_phi(p2)%phi1_max = max(cell_phi(p2)%phi1_max, node(p1)%phi1)
        cell_phi(p2)%phi1_min = min(cell_phi(p2)%phi1_min, node(p1)%phi1)
        phi2_f = (node(p1)%phi2*cell(p2)%vol+node(p2)%phi2*cell(p1)%vol)/&
                 (cell(p1)%vol+cell(p2)%vol)
        cell_phi(p1)%dphi2dx = (cell_phi(p1)%dphi2dx+phi2_f*a1)
        cell_phi(p1)%dphi2dy = (cell_phi(p1)%dphi2dy+phi2_f*a2)
        cell_phi(p2)%dphi2dx = (cell_phi(p2)%dphi2dx-phi2_f*a1)
        cell_phi(p2)%dphi2dy = (cell_phi(p2)%dphi2dy-phi2_f*a2)
    END DO
```

152
cell_phi(p2)%dphi2dy = (cell_phi(p2)%dphi2dy-phi2_f*a2)
cell_phi(p1)%phi2_max = max( cell_phi(p1)%phi2_max, node(p2)%phi2)
cell_phi(p1)%phi2_min = min( cell_phi(p1)%phi2_min, node(p2)%phi2)
cell_phi(p2)%phi2_max = max( cell_phi(p2)%phi2_max, node(p1)%phi2)
cell_phi(p2)%phi2_min = min( cell_phi(p2)%phi2_min, node(p1)%phi2)
END DO

DO ibc = 1, nbc
  IF(INDEX(bc(ibc)%title,'peri')/=0 .or. index(bc(ibc)%title, 'pass') /=0)THEN
    logic1 = .TRUE.
  ELSE
    logic1 = .FALSE.
  ENDIF
  IF (index(bc(ibc)%title, 'inte') /=0)THEN
    logic = .TRUE.
  ELSE
    logic = .FALSE.
  ENDIF
  DO iface = 1, bc(ibc)%nfaces
    p1 = bc(ibc)%face(iface)%p1
    p2 = bc(ibc)%face(iface)%p2
    v1 = bc(ibc)%face(iface)%v(1)
    v2 = bc(ibc)%face(iface)%v(2)
    a1 = bc(ibc)%face(iface)%a(1)
    a2 = bc(ibc)%face(iface)%a(2)
    IF(logic1.or.logic)THEN
      phi1_f=(node(p1)%phi1*cell(p2)%vol+node(p2)%phi1*cell(p1)%vol)/( &
        cell(p1)%vol+cell(p2)%vol)
      cell_phi(p1)%phi1_max=max( cell_phi(p1)%phi1_max, node(p2)%phi1)
      cell_phi(p1)%phi1_min=min( cell_phi(p1)%phi1_min, node(p2)%phi1)
      phi2_f=(node(p1)%phi2*cell(p2)%vol+node(p2)%phi2*cell(p1)%vol)/( &
        cell(p1)%vol+cell(p2)%vol)
      cell_phi(p1)%phi2_max=max( cell_phi(p1)%phi2_max, node(p2)%phi2)
      cell_phi(p1)%phi2_min=min( cell_phi(p1)%phi2_min, node(p2)%phi2)
    ELSE
      phi1_f=node(p1)%phi1
      phi2_f=node(p1)%phi2
    ENDIF
  END DO
END DO

cell_phi(1:ncell)%dphi1dx=cell_phi(1:ncell)%dphi1dx/cell(1:ncell)%vol
cell_phi(1:ncell)%dphi1dy=cell_phi(1:ncell)%dphi1dy/cell(1:ncell)%vol
cell_phi(1:ncell)%dphi2dx = cell_phi(1:ncell)%dphi2dx / cell(1:ncell)%vol

IF(ntransfer_max>0) THEN
    allocate(data_var_send(ntransfer_max), data_var(ntransfer_max1))
ENDIF

iend = 0
DO ibc = 1, nbc
    IF (index(bc(ibc)%title, 'peri') /= 0 .OR. index(bc(ibc)%title, 'pass') /= 0 ) THEN
        ntransfer = bc(ibc)%nfaces
        istart = iend + 1
        iend = iend + ntransfer
        call send_to_dphidx_2d(bc(ibc)%face, ntransfer, cell_phi, data_var_send(istart:iend))
        to_process = bc(ibc)%to_zone - 1
        tag_send = bc(ibc)%to_zone * 10 + my_rank + 8760
        CALL mpi_isend (data_var_send(istart:iend), ntransfer, data_t_mpi_4, to_process, &
                        & tag_send, mpi_comm_world, request_no, ierr)
        CALL mpi_request_free (request_no, ierr)
    ELSE
        DO iface = 1, bc(ibc)%nfaces
            v1 = bc(ibc)%face(iface)%v(1)
            v2 = bc(ibc)%face(iface)%v(2)
            p1 = bc(ibc)%face(iface)%p1
            p2 = bc(ibc)%face(iface)%p2
            ex = bc(ibc)%face(iface)%e(1)
            ey = bc(ibc)%face(iface)%e(2)
            nx = bc(ibc)%face(iface)%n(1)
            ny = bc(ibc)%face(iface)%n(2)

            dphi1de = node(p2)%phi1 - node(p1)%phi1
            dphi1dn = vertex(v2)%phi1 - vertex(v1)%phi1
            dphi2de = node(p2)%phi2 - node(p1)%phi2
            dphi2dn = vertex(v2)%phi2 - vertex(v1)%phi2

            cell_phi(p1)%dphi1dx = ex * dphi1de + nx * dphi1dn
            cell_phi(p1)%dphi1dy = ey * dphi1de + ny * dphi1dn
            cell_phi(p1)%dphi2dx = ex * dphi2de + nx * dphi2dn
            cell_phi(p1)%dphi2dy = ey * dphi2de + ny * dphi2dn
        ENDDO
    ENDIF
ENDDO

DO ibc = 1, nbc
    IF (index(bc(ibc)%title, 'peri') /= 0 .OR. index(bc(ibc)%title, 'pass') /= 0 ) THEN
        ntransfer = bc(ibc)%nfaces
        from_process = bc(ibc)%to_zone - 1
        tag_receive = my_rank*10 + bc(ibc)%to_zone + 8760
        CALL mpi_irecv (data_var(1:ntransfer), ntransfer, data_t_mpi_4, &
                        & from_process, tag_receive, mpi_comm_world, request_no, ierr)
        CALL mpi_wait (request_no, status, ierr)
        DO itr=1,ntransfer
            ii = data_var(itr)%i
            cell_phi(ii)%dphi1dx = data_var(itr)%var(1)
            cell_phi(ii)%dphi1dy = data_var(itr)%var(3)
            cell_phi(ii)%dphi2dx = data_var(itr)%var(2)
            cell_phi(ii)%dphi2dy = data_var(itr)%var(4)
IF(ntransfer_max>0)THEN
  deallocate(data_var_send,data_var)
ENDIF

cell_phi(:,)%phi_phi1 = 1._high
cell_phi(:,)%phi_phi2 = 1._high

DO iface = 1, internal%nfaces
  p1 = internal%face(iface)%p1
  p2 = internal%face(iface)%p2
  xc = internal%face(iface)%xc
  yc = internal%face(iface)%yc
  x1 = node(p1)%x(1)
  y1 = node(p1)%x(2)
  x2 = node(p2)%x(1)
  y2 = node(p2)%x(2)

  dphi1 = cell_phi(p1)%dphi1dx*(xc-x1)+cell_phi(p1)%dphi1dy*(yc-y1)
  IF(dphi1 > 0._high)THEN
    phi = min(1._high, (cell_phi(p1)%phi1_max-node(p1)%phi1)/dphi1)
  ELSE IF(dphi1 < 0._high)THEN
    phi = min(1._high, (cell_phi(p1)%phi1_min-node(p1)%phi1)/dphi1)
  ELSE
    phi = 1._high
  ENDIF
  cell_phi(p1)%phi_phi1 = min(cell_phi(p1)%phi_phi1,phi)

  dphi2 = cell_phi(p1)%dphi2dx*(xc-x1)+cell_phi(p1)%dphi2dy*(yc-y1)
  IF(dphi2 > 0._high)THEN
    phi = min(1._high, (cell_phi(p1)%phi2_max-node(p1)%phi2)/dphi2)
  ELSE IF(dphi2 < 0._high)THEN
    phi = min(1._high, (cell_phi(p1)%phi2_min-node(p1)%phi2)/dphi2)
  ELSE
    phi = 1._high
  ENDIF
  cell_phi(p1)%phi_phi2 = min(cell_phi(p1)%phi_phi2,phi)

  dphi1 = cell_phi(p2)%dphi1dx*(xc-x2)+cell_phi(p2)%dphi1dy*(yc-y2)
  IF(dphi1 > 0._high)THEN
    phi = min(1._high, (cell_phi(p2)%phi1_max-node(p2)%phi1)/dphi1)
  ELSE IF(dphi1 < 0._high)THEN
    phi = min(1._high, (cell_phi(p2)%phi1_min-node(p2)%phi1)/dphi1)
  ELSE
    phi = 1._high
  ENDIF
  cell_phi(p2)%phi_phi1 = min(cell_phi(p2)%phi_phi1,phi)

  dphi2 = cell_phi(p2)%dphi2dx*(xc-x2)+cell_phi(p2)%dphi2dy*(yc-y2)
  IF(dphi2 > 0._high)THEN
    phi = min(1._high, (cell_phi(p2)%phi2_max-node(p2)%phi2)/dphi2)
  ELSE IF(dphi2 < 0._high)THEN
    phi = min(1._high, (cell_phi(p2)%phi2_min-node(p2)%phi2)/dphi2)
  ELSE
    phi = 1._high
  ENDIF
  cell_phi(p2)%phi_phi2 = min(cell_phi(p2)%phi_phi2,phi)
ELSE
  phi = 1._high
ENDIF

cell_phi(p2)%phi_phi2 = min(cell_phi(p2)%phi_phi2,phi)
ENDDO
DO ibc = 1, nbc
  IF (index(bc(ibc)%title, 'peri') /= 0 .or. index(bc(ibc)%title, 'pass') /= 0 ) THEN
    logic1 = .TRUE.
  ELSE
    logic1 = .FALSE.
  END IF
  IF (index(bc(ibc)%title, 'inte') /= 0) THEN
    logic = .TRUE.
  ELSE
    logic = .FALSE.
  END IF
  DO iface = 1, bc(ibc)%nfaces
    p1 = bc(ibc)%face(iface)%p1
    p2 = bc(ibc)%face(iface)%p2
    xc = bc(ibc)%face(iface)%xc
    yc = bc(ibc)%face(iface)%yc
    x1 = node(p1)%x(1)
    y1 = node(p1)%x(2)
    x2 = node(p2)%x(1)
    y2 = node(p2)%x(2)
    IF(logic1.or.logic)THEN
      dphi1=cell_phi(p1)%dphi1dx*(xc-x1)+cell_phi(p1)%dphi1dy*(yc-y1)
      if(dphi1 > 0._high)then
        phi=min(1._high, (cell_phi(p1)%phi1_max-node(p1)%phi1)/dphi1)
      else if(dphi1 < 0._high)then
        phi=min(1._high, (cell_phi(p1)%phi1_min-node(p1)%phi1)/dphi1)
      else
        phi=1._high
      endif
    cell_phi(p1)%phi_phi1=min(cell_phi(p1)%phi_phi1,phi)
    dphi2=cell_phi(p1)%dphi2dx*(xc-x1)+cell_phi(p1)%dphi2dy*(yc-y1)
    if(dphi2 > 0._high)then
      phi=min(1._high, (cell_phi(p1)%phi2_max-node(p1)%phi2)/dphi2)
    else if(dphi2 < 0._high)then
      phi=min(1._high, (cell_phi(p1)%phi2_min-node(p1)%phi2)/dphi2)
    else
      phi=1._high
    endif
    cell_phi(p1)%phi_phi2=min(cell_phi(p1)%phi_phi2,phi)
  ENDDO
  DO ibc = 1, nbc
    IF (index(bc(ibc)%title, 'peri') /= 0 .or. index(bc(ibc)%title, 'pass') /= 0 ) THEN
      logic1 = .TRUE.
    ELSE
      logic1 = .FALSE.
    END IF
    IF (index(bc(ibc)%title, 'inte') /= 0) THEN
      logic = .TRUE.
    ELSE
      logic = .FALSE.
    END IF
    DO iface = 1, bc(ibc)%nfaces
      p1 = bc(ibc)%face(iface)%p1
      p2 = bc(ibc)%face(iface)%p2
      xc = bc(ibc)%face(iface)%xc
      yc = bc(ibc)%face(iface)%yc
      x1 = node(p1)%x(1)
      y1 = node(p1)%x(2)
      x2 = node(p2)%x(1)
      y2 = node(p2)%x(2)
      IF(logic1.or.logic)THEN
        dphi1=cell_phi(p1)%dphi1dx*(xc-x1)+cell_phi(p1)%dphi1dy*(yc-y1)
        if(dphi1 > 0._high)then
          phi=min(1._high, (cell_phi(p1)%phi1_max-node(p1)%phi1)/dphi1)
        else if(dphi1 < 0._high)then
          phi=min(1._high, (cell_phi(p1)%phi1_min-node(p1)%phi1)/dphi1)
        else
          phi=1._high
        endif
      cell_phi(p1)%phi_phi1=min(cell_phi(p1)%phi_phi1,phi)
      dphi2=cell_phi(p1)%dphi2dx*(xc-x1)+cell_phi(p1)%dphi2dy*(yc-y1)
      if(dphi2 > 0._high)then
        phi=min(1._high, (cell_phi(p1)%phi2_max-node(p1)%phi2)/dphi2)
      else if(dphi2 < 0._high)then
        phi=min(1._high, (cell_phi(p1)%phi2_min-node(p1)%phi2)/dphi2)
      else
        phi=1._high
      endif
      cell_phi(p1)%phi_phi2=min(cell_phi(p1)%phi_phi2,phi)
    ENDDO
  ENDDO

  dphi1 = cell_phi(p2)%dphi1dx*(xc-x2)+cell_phi(p2)%dphi1dy*(yc-y2)
  if(dphi1 > 0._high)then
    phi = min(1._high, (cell_phi(p2)%phi1_max-node(p2)%phi1)/dphi1)
  else if(dphi1 < 0._high)then
    phi = min(1._high, (cell_phi(p2)%phi1_min-node(p2)%phi1)/dphi1)
  else
    phi = 1._high
  endif
  dphi2 = cell_phi(p2)%dphi2dx*(xc-x2)+cell_phi(p2)%dphi2dy*(yc-y2)
  if(dphi2 > 0._high)then
    phi = min(1._high, (cell_phi(p2)%phi2_max-node(p2)%phi2)/dphi2)
  else if(dphi2 < 0._high)then
    phi = min(1._high, (cell_phi(p2)%phi2_min-node(p2)%phi2)/dphi2)
  else
    phi = 1._high
  endif
  dphi3 = cell_phi(p2)%dphi3dx*(xc-x2)+cell_phi(p2)%dphi3dy*(yc-y2)
  if(dphi3 > 0._high)then
    phi = min(1._high, (cell_phi(p2)%phi3_max-node(p2)%phi3)/dphi3)
  else if(dphi3 < 0._high)then
    phi = min(1._high, (cell_phi(p2)%phi3_min-node(p2)%phi3)/dphi3)
  else
    phi = 1._high
  endif
Once the gradients are calculated for the two additional PDEs the main solver for
the equations is called. For the two dimensional simulations this subroutine is called “cal\_phi\_2d” and used the point G.-S. solver that is used for other scalar variables such as pressure.

```fortran
SUBROUTINE cal_phi_2d(internal, bc, nbc, vertex, BLOCK, nBLOCK, node, coef, cell, &
  & cell_v, cell_phi, ncell, nnode, receive, give, n_receive, n_give, &
  & dt, res, res_phi2, old_var, my_rank)
  INTEGER :: iface, nbc, ibc, v1, v2, v3, v4, p1, p2, ncell, icell, nnode, nbblock, &
  & my_rank, n_receive, n_give, ierr, ii, iblock
  TYPE (cell_faces) :: internal
  TYPE (cell_faces), DIMENSION(:) :: bc
  TYPE (cell), DIMENSION(:) :: cell
  TYPE (cell_voltage), DIMENSION(:) :: cell_phi
  TYPE (cell_vel), DIMENSION(:) :: cell_v
  TYPE (coeffs), DIMENSION(:) :: coef
  TYPE (vpoints), DIMENSION(:) :: vertex
  TYPE (block_t), DIMENSION(:) :: block
  TYPE (points), DIMENSION(:) :: node
  TYPE (recei_d), DIMENSION(:), POINTER :: receive
  TYPE (give_d), DIMENSION(:), POINTER :: give
  TYPE (points_old), DIMENSION(:) :: old_var
  REAL (high) :: a1, a2, ex, ey, nx, ny, dphi1dx, dphi1dy, &
  & dphi2dx, dphi2dy, flux, dt, mass, up, &
  & um, res, res_phi2, dux, duy, dxd, dyd, vol, phi1, phi2, &
  & dph1, dph2, xc, yc, x1, y1, x2, y2, k_m, k_p, e_m, e_p, &
  & flux_phi1, flux_phi2, dw, vis, c, beta, ss, diss, time_term, lambda
  LOGICAL :: logic, logic1
  REAL(hign), DIMENSION(nnode) :: dt

  DO icell=1,ncell
    dphi1dx = cell_phi(icell)%dphi1dx
    dphi1dy = cell_phi(icell)%dphi1dy
    dphi2dx = cell_phi(icell)%dphi2dx
    dphi2dy = cell_phi(icell)%dphi2dy
    vol=cell(icell)%vol
    phi1 = node(icell)%phi1
    phi2 = node(icell)%phi2
    lambda = cell_phi(icell)%lambda
    coef(icell)%rhs_u = 0.0_high
    coef(icell)%ap_u = 0.0_high
    coef(icell)%rhs_v = -vol*phi2/(lambda**2)
    coef(icell)%ap_v = vol/(lambda**2)
  ENDDO

  cell(:)%isur = 0._high
  dt(:)=0._high

  DO iface = 1, internal%nfaces
    mass= 0._high
    v1 = internal%face(iface)%v(1)
    v2 = internal%face(iface)%v(2)
    p1 = internal%face(iface)%p1
    p2 = internal%face(iface)%p2
    a1 = internal%face(iface)%a(1)
    a2 = internal%face(iface)%a(2)
    ex = internal%face(iface)%e(1)
```

```
ey = internal%face(iface)%e(2)
nx = internal%face(iface)%n(1)
ny = internal%face(iface)%n(2)

xc=internal%face(iface)%xc
yc=internal%face(iface)%yc
x1=node(p1)%x(1)
y1=node(p1)%x(2)
x2=node(p2)%x(1)
y2=node(p2)%x(2)

tvis = (cell_phi(p1)%eps*cell(p2)%vol+cell_phi(p2)%eps*cell(p1)%vol)/ &
& (cell(p1)%vol+cell(p2)%vol)

dphi1de = node(p2)%phi1 - node(p1)%phi1
dphi2de = node(p2)%phi2 - node(p1)%phi2
dphi1dn = vertex(v2)%phi1 - vertex(v1)%phi1
dphi2dn = vertex(v2)%phi2 - vertex(v1)%phi2
dphi1dx = ex * dphi1de + nx * dphi1dn
dphi1dy = ey * dphi1de + ny * dphi1dn
dphi2dx = ex * dphi2de + nx * dphi2dn
dphi2dy = ey * dphi2de + ny * dphi2dn
difphi1=tvis
difphi2=tvis

flux_phi1 = -difphi1 * dphi1dx * a1 - difphi1 * dphi1dy * a2
flux_phi2 = -difphi2 * dphi2dx * a1 - difphi2 * dphi2dy * a2
coef(p1)%rhs_u = coef(p1)%rhs_u - flux_phi1
coef(p2)%rhs_u = coef(p2)%rhs_u + flux_phi1
coef(p1)%rhs_v = coef(p1)%rhs_v - flux_phi2
coef(p2)%rhs_v = coef(p2)%rhs_v + flux_phi2

cell(p1)%isur = cell(p1)%isur + 1
cell(p2)%isur = cell(p2)%isur + 1
coef(p1)%an_u(cell(p1)%isur) = difphi1 * (ex*a1+ey*a2)
coef(p2)%an_u(cell(p2)%isur) = difphi1 * (ex*a1+ey*a2)
coef(p1)%an_v(cell(p1)%isur) = difphi2 * (ex*a1+ey*a2)
coef(p2)%an_v(cell(p2)%isur) = difphi2 * (ex*a1+ey*a2)

END DO

DO ibc = 1, nbc
DO iface = 1, bc(ibc)%nfaces
  IF(index(bc(ibc)%title,'peri')/=0.or.index(bc(ibc)%title,'pass')/=0)THEN
    logic1 = .TRUE.
  ELSE
    logic1 = .FALSE.
  END IF

  IF(index(bc(ibc)%title,'inte')/=0)THEN
    logic = .TRUE.
  ELSE
    logic = .FALSE.
  END IF

  v1 = bc(ibc)%face(iface)%v(1)
v2 = bc(ibc)%face(iface)%v(2)
p1 = bc(ibc)%face(iface)%p1
```
p2 = bc(ibc)%face(iface)%p2
a1 = bc(ibc)%face(iface)%a(1)
a2 = bc(ibc)%face(iface)%a(2)
ex = bc(ibc)%face(iface)%e(1)
ey = bc(ibc)%face(iface)%e(2)
nx = bc(ibc)%face(iface)%n(1)
ny = bc(ibc)%face(iface)%n(2)
dw=(node(p1)%yw*cell(p2)%vol+node(p2)%yw*cell(p1)%vol)/ &
  (cell(p1)%vol+cell(p2)%vol)

IF (logic1) THEN
  tvis = (cell_phi(p1)%eps*cell(p2)%vol+cell(p2)%eps*cell(p1)%vol)/ &
          (cell(p1)%vol+cell(p2)%vol)
ELSE
  tvis = cell_phi(p1)%eps
ENDIF

dphi1de = node(p2)%phi1 - node(p1)%phi1
dphi2de = node(p2)%phi2 - node(p1)%phi2
dphi1dn = vertex(v2)%phi1 - vertex(v1)%phi1
dphi2dn = vertex(v2)%phi2 - vertex(v1)%phi2
dphi1dx = ex * dphi1de + nx * dphi1dn
dphi1dy = ey * dphi1de + ny * dphi1dn
dphi2dx = ex * dphi2de + nx * dphi2dn
dphi2dy = ey * dphi2de + ny * dphi2dn
difphi1 = tvis
difphi2 = tvis

IF(logic)THEN
  flux_phi1 = -difphi1 * dphi1dx * a1 - difphi1 * dphi1dy * a2
  flux_phi2 = -difphi2 * dphi2dx * a1 - difphi2 * dphi2dy * a2
  coef(p1)%rhs_u = coef(p1)%rhs_u - flux_phi1
  coef(p1)%rhs_v = coef(p1)%rhs_v - flux_phi2
  cell(p1)%isur = cell(p1)%isur + 1
  coef(p1)%an_u(cell(p1)%isur) = difphi1 * (ex*a1+ey*a2)
  coef(p1)%an_v(cell(p1)%isur) = difphi2 * (ex*a1+ey*a2)
END IF

flux_phi1 = -difphi1 * dphi1dx * a1 - difphi1 * dphi1dy * a2
flux_phi2 = -difphi2 * dphi2dx * a1 - difphi2 * dphi2dy * a2
coef(p2)%rhs_u = coef(p2)%rhs_u + flux_phi1
coef(p2)%rhs_v = coef(p2)%rhs_v + flux_phi2
cell(p2)%isur = cell(p2)%isur + 1
coef(p2)%an_u(cell(p2)%isur) = difphi1 * (ex*a1+ey*a2)
coef(p2)%an_v(cell(p2)%isur) = difphi2 * (ex*a1+ey*a2)
END DO
END DO

res_phi1=0.0_high
res_phi2=0.0_high

DO iblock =1, nblock
  DO icell=block(iblock)%n_begin, block(iblock)%n_end
    coef(icell)%ap_u = coef(icell)%ap_u + sum(coef(icell)%an_u(:))
    coef(icell)%ap_v = coef(icell)%ap_v + sum(coef(icell)%an_v(:))
  res_phi1 = res_phi1 + abs(coef(icell)%rhs_u)
  res_phi2 = res_phi2 + abs(coef(icell)%rhs_v)
END DO
END DO
coef(1:ncell)%ap_u = coef(1:ncell)%ap_u/urfphi
coef(1:ncell)%ap_v = coef(1:ncell)%ap_v/urfphi

dt=0._high
DO ii=1,outer_iter_phi
    CALL point_GS_solver (cell, dt, coef(:)%rhs_u, coef(:)%ap_u, coef, &
                       & ncell, BLOCK, nblock, inner_iter_phi,1)
    IF(outer_iter_phi/=ii)CALL set_bc_iv (node, cell_v, dt, bc, nbc,my_rank)
ENDDO
node(1:ncell)%phi1 = abs(node(1:ncell)%phi1 + dt(1:ncell))

dt=0._high
DO ii=1,outer_iter_phi
    CALL point_GS_solver (cell, dt, coef(:)%rhs_v, coef(:)%ap_v, coef, &
                       & ncell, BLOCK, nblock, inner_iter_phi,2)
    IF(outer_iter_phi/=ii)CALL set_bc_iv (node, cell_v, dt, bc, nbc,my_rank)
ENDDO
node(1:ncell)%phi2 = abs(node(1:ncell)%phi2 + dt(1:ncell))

node(1:ncell)%fb = sqrt((node(1:ncell)%phi2*cell_phi(1:ncell)%dphi1dx)**2 + &
                (node(1:ncell)%phi2*cell_phi(1:ncell)%dphi1dy)**2)/densit

CALL set_bc_phi(node, cell_phi, bc, nbc,give,receive,n_give,n_receive,my_rank)

END SUBROUTINE cal_phi_2d
Bibliography


77. L. Huang, P. Huang, R. LeBeau, and T. Hauser. Optimization of blowing and suction control on NACA 0012 airfoil using EARND genetic algorithm with diversity control. Accepted by AIAA Journal of Aircraft.


168


Vita

Daniel A. Reasor Jr.
Born October 17th, 1982 in Corbin, KY.

Education
Bachelor of Science, Mechanical Engineering, August 2001 - May 2006
University of Kentucky, Lexington, KY, USA.

Professional Positions Held
Research Assistant, Fall 2005 - Summer 2007
Computational Fluid Dynamics Group
Department of Mechanical Engineering, University of Kentucky, Lexington, KY.

Lean Apprentice, August 2004 - December 2004
Lean Manufacturing Systems Group
Center for Manufacturing, University of Kentucky, Lexington, KY.

Co-op, August 2003 - December 2003
Market Quality - Interior Parts Group
Honda of America Manufacturing Inc., Marysville, OH.

Scholastic Honors and Certifications
Science, Mathematics and Research for Transformation (SMART) Scholarship/Fellowship,
Fall 2007 - Spring 2011
United States Department of Defense with the Naval Postgraduate School, Washington, D.C.

Presidential Fellowship Recipient, Fall 2007 - Spring 2008
Georgia Institute of Technology, Atlanta, GA.

Kentucky Space Grant Consortium Fellowship, Fall 2006 - Spring 2007
University of Kentucky, Lexington, KY.

Certified Engineer in Training (EIT), October 2006
Commonwealth of Kentucky, Frankfort, KY.

Graduate School Annual Year Fellowship, Fall 2005 - Fall 2006
University of Kentucky, Lexington, KY.

ASME Bluegrass Section Outstanding Scholar, 2006
University of Kentucky, Lexington, KY.

Certified in Lean Manufacturing Systems, August 2005
University of Kentucky, Center for Manufacturing, Lexington, KY.

Outstanding Senior in Mechanical Engineering, Fall 2004 - Spring 2005
University of Kentucky, Lexington, KY.

*Outstanding Junior in Mechanical Engineering*, Fall 2003 - Spring 2004
University of Kentucky, Lexington, KY.

*Outstanding Sophomore in Mechanical Engineering*, Fall 2002 - Spring 2003
University of Kentucky, Lexington, KY.

**Professional Society Membership**
American Institute for Aeronautics and Astronautics

**Papers and Presentations**


4. D. Reasor and R. LeBeau, “Numerical Investigation of the Effects of Bumps on Inflatable Wing Profiles,” AIAA Student Paper, AIAA Region III Student Conference, South Bend, IN, March 2007 (Received 2nd Place in Master’s Paper/Presentation Division).

