AN ABSTRACT OF THE DISSERTATION OF

Cody W. Ray for the degree of Doctor of Philosophy in Mechanical Engineering presented on April 19, 2012.

Title: Modeling, Control, and Estimation of Flexible, Aerodynamic Structures

Abstract approved: Belinda A. Batten

Engineers have long been inspired by nature’s flyers. Such animals navigate complex environments gracefully and efficiently by using a variety of evolutionary adaptations for high-performance flight. Biologists have discovered a variety of sensory adaptations that provide flow state feedback and allow flying animals to feel their way through flight. A specialized skeletal wing structure and plethora of robust, adaptable sensory systems together allow nature’s flyers to adapt to myriad flight conditions and regimes. In this work, motivated by biology and the successes of bio-inspired, engineered aerial vehicles, linear quadratic control of a flexible, morphing wing design is investigated, helping to pave the way for truly autonomous, mission-adaptive craft. The proposed control algorithm is demonstrated to morph a wing into desired positions. Furthermore, motivated specifically by the sensory adaptations organisms possess, this work transitions to an investigation of aircraft wing load identification using structural response as measured by distributed sensors. A novel, recursive estimation algorithm is utilized to recursively solve the inverse problem of load identification, providing both wing structural and aerodynamic
states for use in a feedback control, mission-adaptive framework. The recursive load identification algorithm is demonstrated to provide accurate load estimate in both simulation and experiment.
Modeling, Control, and Estimation of Flexible, Aerodynamic Structures

by
Cody W. Ray

A DISSERTATION
submitted to
Oregon State University

in partial fulfillment of
the requirements for the
degree of
Doctor of Philosophy

Presented April 19, 2012
Commencement June 2012

APPROVED:

Major Professor, representing Mechanical Engineering

Head of the School of Mechanical, Industrial, and Manufacturing Engineering

Dean of the Graduate School

I understand that my dissertation will become part of the permanent collection of Oregon State University libraries. My signature below authorizes release of my dissertation to any reader upon request.

Cody W. Ray, Author
ACKNOWLEDGMENTS

The author expresses special appreciation to his wife, Jenilee Ray, for her help, both technical and otherwise, during his entire Ph.D. experience and dedicates this dissertation to her. The author also acknowledges his parents for their consistent love and support over the years and always being there like good friends whenever needed.

The author would like to acknowledge his advisor, Belinda Batten, and express genuine appreciation for both the support she offered and the challenges she insisted he endure during his graduate career. She never ceased to teach, regardless of disagreements, and the author is indebted to her for forcing him to “learn to learn,” which is fundamentally the most important skill one may wield throughout life.

Further acknowledgement goes to Roberto Albertani, who served as a strong academic force for the latter part of the author’s Ph.D. program. His attendance at Oregon State University has made markedly positive changes to all those working with him, especially the author.

Also acknowledged for their intermittent support and useful discussions throughout the author’s academic career, in no particular order, are Trenton Carpenter, Jorn Cheney, Chris Patton, Jasmine Chuang, Joshua Merrit, and Drs. Timothy Kennedy, Brian Bay, Vrushali Bokil, Nathan Gibson, Ben Dickinson, and John Singler. The author thanks you all and hopes to be surrounded by folks like you throughout his entire career.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Introduction ..........................</td>
</tr>
<tr>
<td>2 Biological Inspiration: Morphing Wings and Sensors ...............</td>
</tr>
<tr>
<td>2.1 Morphing and Flexible Wing Designs ..........................</td>
</tr>
<tr>
<td>2.2 Sensors: Closing the Loop ..........................</td>
</tr>
<tr>
<td>3 Literature Review: Morphing Wings and Load Identification ...............</td>
</tr>
<tr>
<td>3.1 Morphing Wing Designs ..........................</td>
</tr>
<tr>
<td>3.1.1 Morphing Wings and Smart Materials ..........................</td>
</tr>
<tr>
<td>3.1.2 Membrane Wing Designs ..........................</td>
</tr>
<tr>
<td>3.2 Aerodynamic Load Identification: A Review of Existing Methodologies ...............</td>
</tr>
<tr>
<td>3.2.1 A Chronology of Aerodynamic Load Identification ...............</td>
</tr>
<tr>
<td>3.2.2 Disturbance Observer Design: A Potentially Real-Time Solution Built for Control ...............</td>
</tr>
<tr>
<td>3.3 Inspiration and Review Summary, Hypotheses, and Dissertation Contributions ...............</td>
</tr>
<tr>
<td>4 Model Development ..........................</td>
</tr>
<tr>
<td>4.1 Thin Plate Model ..........................</td>
</tr>
<tr>
<td>4.1.1 Moment Resultant Computation: Generalization to Anisotropic Materials ..........................</td>
</tr>
<tr>
<td>4.1.2 Derivation of Dynamic Equations and Boundary Conditions ...............</td>
</tr>
<tr>
<td>4.1.3 Piezocomposite Actuators and Sensors ..........................</td>
</tr>
<tr>
<td>4.2 Linear Membrane Model ..........................</td>
</tr>
<tr>
<td>Section</td>
</tr>
<tr>
<td>---------</td>
</tr>
<tr>
<td>4.2.1</td>
</tr>
<tr>
<td>4.2.2</td>
</tr>
<tr>
<td>5</td>
</tr>
<tr>
<td>5.1</td>
</tr>
<tr>
<td>5.2</td>
</tr>
<tr>
<td>5.3</td>
</tr>
<tr>
<td>5.4</td>
</tr>
<tr>
<td>5.4.1</td>
</tr>
<tr>
<td>5.4.2</td>
</tr>
<tr>
<td>5.4.3</td>
</tr>
<tr>
<td>5.4.4</td>
</tr>
<tr>
<td>5.4.5</td>
</tr>
<tr>
<td>6</td>
</tr>
<tr>
<td>6.1</td>
</tr>
<tr>
<td>6.2</td>
</tr>
<tr>
<td>6.2.1</td>
</tr>
<tr>
<td>6.2.2</td>
</tr>
<tr>
<td>6.2.3</td>
</tr>
<tr>
<td>6.3</td>
</tr>
<tr>
<td>6.3.1</td>
</tr>
<tr>
<td>Section</td>
</tr>
<tr>
<td>------------------------------------------------------------------------</td>
</tr>
<tr>
<td>6.3.2 Convergence of Regularization Parameter $\lambda$</td>
</tr>
<tr>
<td>6.3.3 Evidence of Consistency</td>
</tr>
<tr>
<td>6.4 Quasi-Static Load Identification for a Simulated Membrane Wing</td>
</tr>
<tr>
<td>6.4.1 Load One</td>
</tr>
<tr>
<td>6.4.2 Load Two</td>
</tr>
<tr>
<td>6.4.3 Load Three</td>
</tr>
<tr>
<td>6.5 Preliminary Experimental Results</td>
</tr>
<tr>
<td>6.5.1 Experimental Setup</td>
</tr>
<tr>
<td>6.5.2 Estimation Results</td>
</tr>
<tr>
<td>7 Conclusions</td>
</tr>
<tr>
<td>Bibliography</td>
</tr>
<tr>
<td>Appendices</td>
</tr>
<tr>
<td>A. Derivation of Weak Form of an Anisotropic, Thin Plate Model</td>
</tr>
<tr>
<td>B. Derivation of Weak Form of a Linear Membrane Model</td>
</tr>
<tr>
<td>B.1 Finite Element Approximation of Stretch Sensor Model</td>
</tr>
</tbody>
</table>
# List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Wing morphing of a Violet-green Swallow</td>
<td>8</td>
</tr>
<tr>
<td>2.2</td>
<td>Apparent wing twist of a Black-billed Magpie</td>
<td>8</td>
</tr>
<tr>
<td>2.3</td>
<td>Bird wing schematic</td>
<td>9</td>
</tr>
<tr>
<td>2.4</td>
<td>Bat wing schematic</td>
<td>9</td>
</tr>
<tr>
<td>2.5</td>
<td>Bat wing plagiopatagium</td>
<td>9</td>
</tr>
<tr>
<td>2.6</td>
<td>Simplified muscle spindle</td>
<td>12</td>
</tr>
<tr>
<td>3.1</td>
<td>Comparison of $C_L$ vs. AOA for rigid and membrane wings</td>
<td>24</td>
</tr>
<tr>
<td>3.2</td>
<td>Membrane wing deforming due to 20g mass</td>
<td>25</td>
</tr>
<tr>
<td>3.3</td>
<td>Top view of elliptical planform of wing used in [1]</td>
<td>25</td>
</tr>
<tr>
<td>4.1</td>
<td>Plate model geometric definitions</td>
<td>37</td>
</tr>
<tr>
<td>4.2</td>
<td>Force resultants acting on an infinitesimal plate element</td>
<td>38</td>
</tr>
<tr>
<td>4.3</td>
<td>Moment resultants acting on an infinitesimal plate element</td>
<td>38</td>
</tr>
<tr>
<td>4.4</td>
<td>Forces acting on a deformed infinitesimal membrane element</td>
<td>53</td>
</tr>
<tr>
<td>5.1</td>
<td>Wing twist</td>
<td>63</td>
</tr>
<tr>
<td>5.2</td>
<td>Wing camber</td>
<td>64</td>
</tr>
<tr>
<td>5.3</td>
<td>Wing bending</td>
<td>65</td>
</tr>
<tr>
<td>5.4</td>
<td>Morphing patch-plate system geometry</td>
<td>66</td>
</tr>
<tr>
<td>5.5</td>
<td>180-element mesh for patch-plate system</td>
<td>80</td>
</tr>
<tr>
<td>5.6</td>
<td>720-element mesh for patch-plate system</td>
<td>80</td>
</tr>
<tr>
<td>Figure</td>
<td>Description</td>
<td>Page</td>
</tr>
<tr>
<td>--------</td>
<td>-------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>5.7</td>
<td>2880-element mesh for patch-plate system</td>
<td>80</td>
</tr>
<tr>
<td>5.8</td>
<td>First bending mode of plate system</td>
<td>81</td>
</tr>
<tr>
<td>5.9</td>
<td>First torsional mode of plate system</td>
<td>81</td>
</tr>
<tr>
<td>5.10</td>
<td>Second bending mode of plate system</td>
<td>81</td>
</tr>
<tr>
<td>5.11</td>
<td>Second torsional mode of plate system</td>
<td>81</td>
</tr>
<tr>
<td>5.12</td>
<td>Numerical results: bending gain convergence for (P_{z1})</td>
<td>85</td>
</tr>
<tr>
<td>5.13</td>
<td>Numerical results: bending gain convergence for (P_{z4})</td>
<td>85</td>
</tr>
<tr>
<td>5.14</td>
<td>Numerical results: bending gain convergence for (P_{z7})</td>
<td>85</td>
</tr>
<tr>
<td>5.15</td>
<td>Numerical results: velocity gain convergence for (P_{z1})</td>
<td>86</td>
</tr>
<tr>
<td>5.16</td>
<td>Numerical results: velocity gain convergence for (P_{z4})</td>
<td>86</td>
</tr>
<tr>
<td>5.17</td>
<td>Numerical results: velocity gain convergence for (P_{z7})</td>
<td>86</td>
</tr>
<tr>
<td>5.18</td>
<td>Wing twist: control input time history</td>
<td>87</td>
</tr>
<tr>
<td>5.19</td>
<td>Wing twist: wing-tip corner deflection time history</td>
<td>87</td>
</tr>
<tr>
<td>5.20</td>
<td>Wing twist: final surface position</td>
<td>89</td>
</tr>
<tr>
<td>5.21</td>
<td>Wing twist: surface deformation time snapshots</td>
<td>89</td>
</tr>
<tr>
<td>5.22</td>
<td>Wing twist: final (\epsilon_{xx}) field</td>
<td>89</td>
</tr>
<tr>
<td>5.23</td>
<td>Wing twist: final (\epsilon_{yy}) field</td>
<td>89</td>
</tr>
<tr>
<td>5.24</td>
<td>Wing twist: final (\epsilon_{xy}) field</td>
<td>89</td>
</tr>
<tr>
<td>5.25</td>
<td>Wing camber: control input time history</td>
<td>90</td>
</tr>
</tbody>
</table>
## LIST OF FIGURES (Continued)

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.26</td>
<td>Wing camber: wing-tip corner deflection time history</td>
<td>90</td>
</tr>
<tr>
<td>5.27</td>
<td>Wing camber: final surface position</td>
<td>91</td>
</tr>
<tr>
<td>5.28</td>
<td>Wing camber: surface deformation time snapshots</td>
<td>91</td>
</tr>
<tr>
<td>5.29</td>
<td>Wing camber: final $\epsilon_{xx}$ field</td>
<td>92</td>
</tr>
<tr>
<td>5.30</td>
<td>Wing camber: final $\epsilon_{yy}$ field</td>
<td>92</td>
</tr>
<tr>
<td>5.31</td>
<td>Wing camber: final $\epsilon_{xy}$ field</td>
<td>92</td>
</tr>
<tr>
<td>5.32</td>
<td>Wing bending: control input time history</td>
<td>93</td>
</tr>
<tr>
<td>5.33</td>
<td>Wing bending: wing-tip corner deflection time history</td>
<td>93</td>
</tr>
<tr>
<td>5.34</td>
<td>Wing bending: final surface position</td>
<td>94</td>
</tr>
<tr>
<td>5.35</td>
<td>Wing bending: surface deformation time snapshots</td>
<td>94</td>
</tr>
<tr>
<td>5.36</td>
<td>Wing bending: final $\epsilon_{xx}$ field</td>
<td>95</td>
</tr>
<tr>
<td>5.37</td>
<td>Wing bending: final $\epsilon_{yy}$ field</td>
<td>95</td>
</tr>
<tr>
<td>5.38</td>
<td>Wing bending: final $\epsilon_{xy}$ field</td>
<td>95</td>
</tr>
<tr>
<td>6.1</td>
<td>Simulated membrane system geometry</td>
<td>99</td>
</tr>
<tr>
<td>6.2</td>
<td>Laboratory membrane wing geometry</td>
<td>100</td>
</tr>
<tr>
<td>6.3</td>
<td>72-element mesh for membrane system</td>
<td>125</td>
</tr>
<tr>
<td>6.4</td>
<td>288-element mesh for membrane system</td>
<td>125</td>
</tr>
<tr>
<td>6.5</td>
<td>1152-element mesh for membrane system</td>
<td>125</td>
</tr>
<tr>
<td>6.6</td>
<td>First mode of membrane system</td>
<td>126</td>
</tr>
<tr>
<td>Figure</td>
<td>Page</td>
<td></td>
</tr>
<tr>
<td>--------</td>
<td>------</td>
<td></td>
</tr>
<tr>
<td>6.7</td>
<td>126</td>
<td></td>
</tr>
<tr>
<td>6.8</td>
<td>126</td>
<td></td>
</tr>
<tr>
<td>6.9</td>
<td>126</td>
<td></td>
</tr>
<tr>
<td>6.10</td>
<td>128</td>
<td></td>
</tr>
<tr>
<td>6.11</td>
<td>129</td>
<td></td>
</tr>
<tr>
<td>6.12</td>
<td>129</td>
<td></td>
</tr>
<tr>
<td>6.13</td>
<td>129</td>
<td></td>
</tr>
<tr>
<td>6.14</td>
<td>129</td>
<td></td>
</tr>
<tr>
<td>6.15</td>
<td>129</td>
<td></td>
</tr>
<tr>
<td>6.16</td>
<td>130</td>
<td></td>
</tr>
<tr>
<td>6.17</td>
<td>130</td>
<td></td>
</tr>
<tr>
<td>6.18</td>
<td>133</td>
<td></td>
</tr>
<tr>
<td>6.19</td>
<td>133</td>
<td></td>
</tr>
<tr>
<td>6.20</td>
<td>133</td>
<td></td>
</tr>
<tr>
<td>6.21</td>
<td>134</td>
<td></td>
</tr>
<tr>
<td>6.22</td>
<td>134</td>
<td></td>
</tr>
<tr>
<td>6.23</td>
<td>135</td>
<td></td>
</tr>
<tr>
<td>6.24</td>
<td>135</td>
<td></td>
</tr>
<tr>
<td>6.25</td>
<td>135</td>
<td></td>
</tr>
<tr>
<td>Figure</td>
<td>Description</td>
<td>Page</td>
</tr>
<tr>
<td>----------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-------</td>
</tr>
<tr>
<td>6.26</td>
<td>Final membrane position for load one, 5% noise</td>
<td>135</td>
</tr>
<tr>
<td>6.27</td>
<td>Estimated load distribution for load two, 1% noise</td>
<td>137</td>
</tr>
<tr>
<td>6.28</td>
<td>Estimated load distribution for load two, 5% noise</td>
<td>137</td>
</tr>
<tr>
<td>6.29</td>
<td>Exact load two: $P_z = 200 \sin(2\pi x) \sin(\pi y)$</td>
<td>137</td>
</tr>
<tr>
<td>6.30</td>
<td>$L^2$ relative load estimate error for load two, 1% noise</td>
<td>138</td>
</tr>
<tr>
<td>6.31</td>
<td>$L^2$ relative load estimate error for load two, 5% noise</td>
<td>138</td>
</tr>
<tr>
<td>6.32</td>
<td>$L^2$ relative position estimate error for load two, 1% noise</td>
<td>138</td>
</tr>
<tr>
<td>6.33</td>
<td>$L^2$ relative position estimate error for load two, 5% noise</td>
<td>138</td>
</tr>
<tr>
<td>6.34</td>
<td>Final membrane position for load two, 1% noise</td>
<td>139</td>
</tr>
<tr>
<td>6.35</td>
<td>Final membrane position for load two, 5% noise</td>
<td>139</td>
</tr>
<tr>
<td>6.36</td>
<td>Estimated load distribution for load three, 1% noise</td>
<td>140</td>
</tr>
<tr>
<td>6.37</td>
<td>Estimated load distribution for load three, 5% noise</td>
<td>140</td>
</tr>
<tr>
<td>6.38</td>
<td>Exact load three: $P_z = 200 \sin(\pi x) \sin(2\pi y)$</td>
<td>140</td>
</tr>
<tr>
<td>6.39</td>
<td>$L^2$ relative load estimate error for load three, 1% noise</td>
<td>141</td>
</tr>
<tr>
<td>6.40</td>
<td>$L^2$ relative load estimate error for load three, 5% noise</td>
<td>141</td>
</tr>
<tr>
<td>6.41</td>
<td>$L^2$ relative position estimate error for load three, 1% noise</td>
<td>141</td>
</tr>
<tr>
<td>6.42</td>
<td>$L^2$ relative position estimate error for load three, 5% noise</td>
<td>141</td>
</tr>
<tr>
<td>6.43</td>
<td>Final membrane position for load three, 1% noise</td>
<td>142</td>
</tr>
<tr>
<td>6.44</td>
<td>Final membrane position for load three, 5% noise</td>
<td>142</td>
</tr>
<tr>
<td>Figure</td>
<td>Description</td>
<td>Page</td>
</tr>
<tr>
<td>--------</td>
<td>-----------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>6.45</td>
<td>Wing test sample fabrication</td>
<td>143</td>
</tr>
<tr>
<td>6.46</td>
<td>Experimental wind tunnel test setup</td>
<td>144</td>
</tr>
<tr>
<td>6.47</td>
<td>Wing sample during testing</td>
<td>144</td>
</tr>
<tr>
<td>6.48</td>
<td>Time-varying lift resultant estimate</td>
<td>145</td>
</tr>
<tr>
<td>6.49</td>
<td>Estimated load distribution on membrane wing</td>
<td>146</td>
</tr>
<tr>
<td>6.50</td>
<td>Estimated and measured membrane deformation</td>
<td>146</td>
</tr>
</tbody>
</table>
## LIST OF TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.1</td>
<td>Essential and natural plate boundary conditions</td>
<td>45</td>
</tr>
<tr>
<td>5.1</td>
<td>Normalized morphing shape functions</td>
<td>65</td>
</tr>
<tr>
<td>5.2</td>
<td>Plate system material parameters</td>
<td>67</td>
</tr>
<tr>
<td>5.3</td>
<td>Plate system Kelvin-Voigt and viscous damping parameters</td>
<td>67</td>
</tr>
<tr>
<td>5.4</td>
<td>Plate system: first four undamped modal frequencies</td>
<td>81</td>
</tr>
<tr>
<td>5.5</td>
<td>Plate system: eigenvalues</td>
<td>82</td>
</tr>
<tr>
<td>6.1</td>
<td>Membrane system material parameters and dimensions</td>
<td>100</td>
</tr>
<tr>
<td>6.2</td>
<td>Membrane system: first four undamped modal frequencies</td>
<td>127</td>
</tr>
<tr>
<td>6.3</td>
<td>Membrane system: eigenvalues</td>
<td>127</td>
</tr>
</tbody>
</table>
To Jenilee
1 INTRODUCTION

A goal of recent research in scientific, engineering, military, and law enforcement communities is the development of small-scale air vehicles that exhibit high performance, efficiency, and are capable of executing missions autonomously. Such craft would undoubtedly be useful for a variety of tasks including, but not limited to, reconnaissance, search and rescue, and even combat missions. Consider a fleet of inexpensive, but effective, micro air vehicles (MAVs) capable of the same maneuverability seen in nature’s flyers, entering a dense urban setting to seek and destroy a known deadly threat. Such a scene was once limited to the imagination, but aircraft capable of performing such a task are now on the horizon. The motivation for continuing development of such craft is sound and ethical, for it will potentially reduce or even prevent loss of life on the battlefield or save lives during disaster. Investigations related to the MAV initiative are taking place in a variety of scientific fields and are already generating new and useful results. More importantly, like many scientific endeavors, the gains in knowledge and understanding will not necessarily be limited to small air vehicles or merely air vehicles, but will likely be applicable to aircraft of all sizes as well as problems involving control of fluids and morphing structures, and even in the fields of renewable energy (wind turbines) or automotive engineering – generally anywhere one wishes to improve efficiency and/or control design.

Investigations involving morphing and flexible wing designs, largely inspired by biology, are yielding aerial vehicles at the scale of < 15cm that operate with great performance and efficiency – performance perhaps even beginning to compare to their biological counterparts. Although morphing and flexible wing designs have long been of interest in aerodynamics (indeed the Wright Flyer is considered to have been of a morphing design), only relatively recently have such designs been investigated at
the scale of MAVs. This is partly due to the inertia of tradition in aircraft design and science in general, but also to technological limitations only recently begun to be addressed. While flight at the larger scales, which was of primary interest during the 20th century, requires large, stiff wings for support, and efficient flight at these scales is characterized by steady laminar flow, flight at smaller scales is more prone to a variety of undesirable aerodynamic effects. Although such problems are present at the scale of larger craft, the effects are generally of a smaller scale than the aircraft itself, thus such effects become pronounced when the aircraft is of small scale. Scientists and engineers are looking to biology to overcome such obstacles in the creation of morphing and flexible wing MAV designs.

Through millions of years of evolution, trial and error, stability and instability, nature has created a huge variety of flyers. Birds, bats, flying (gliding) squirrels, insects, and their extinct ancestors have taken to the skies and largely freed themselves from terrestrial locomotion. It has been observed that many flying animals, especially birds, utilize wing morphing for flight: bending, twisting, and sweeping their wings about during aerodynamic maneuvers. It has also been found that for certain animals that have highly flexible wings, such as bats, the flexible nature of the wing itself provides many desirable, passive flight effects. Hence, biologists are now directly interacting with aerodynamicists, engineers, and mathematicians in an effort to better understand the flight of these animals and how one might replicate their abilities to improve current technological designs. For example, the Air Force Office of Scientific Research Multidisciplinary University Research Initiative (MURI) project, led by Kenny Breuer of Brown University and including collaboration with Oregon State University, Massachusetts Institute of Technology, and University of Maryland, involves studying the aerodynamics, structural dynamics [2, 3], control, and fluid-structure interaction of bat flight [4], flexible wing designs [5, 6], as well as specialized
flow sensor designs [7–10]. A related MURI project at University of Michigan, involving University of Florida, consisted of similar collaborative efforts but largely regarded studies of bird and insect flight [11, 12]. The Defense Advanced Research Projects Agency (DARPA), who strongly endorses multidisciplinary research as the foundation for creating the innovations of the future, has also expressed interest in wing morphing, facilitating the DARPA 2011 Morphing Aircraft Structure program whose end goal is a wing that manifests significant planform geometry changes during flight by utilizing advanced actuators [13].

Not surprisingly, sensor networks within biological models are generating much enthusiasm amongst control and estimation engineers. The ease by which nature’s flyers navigate through confined, obstacle-cluttered environments, simultaneously rejecting severe disturbances, has long inspired engineered aerial vehicle design. Organisms such as bats exhibit a variety of sensory mechanisms for flight control, including muscle and hair cell sensory feedback. It is hypothesized that including distributed sensors into MAV designs will enhance flight and onboard control by accounting for unsteady aerodynamic effects. Distributed sensor networks could provide both structural and flow state knowledge, instantaneously accounting for structural vibration, deformation, flow separation, and general disturbances. An active wing morphing design must utilize sensory feedback to address shape tracking error and “close the loop.” MAV designs utilizing distributed sensory feedback will greatly expand the operational envelope of current MAV flight.

This dissertation is a synthesis of investigations regarding optimal control, morphing, and state estimation as they pertain to flexible, aerodynamic structures. First, optimal control is utilized to bend and twist a thin, flexible, wing-like structure into aerodynamically inspired shapes. It is found, however, that pressure disturbances on said wing are, at times, too great to reject without direct knowledge of the dis-
turbances themselves. Thus, motivated by this reason and biology, the problem of estimating such disturbances in a general disturbance estimation framework is investigated. It is found that basic stretch sensor models indeed provide useful information in flight control design. To verify this and compare to simulation results, the load identification approach is tested on real data collected during wind tunnel testing.

This document begins with biological inspiration for all aspects of research; the first section regards morphing and flexible wing designs and the flight consequences of such, which is followed by a discussion of sensors in biology. Relevant engineering literature is then reviewed. Thin plate and membrane models used in chapters 5 and 6 are derived in chapter 4. These are infinite-dimensional, distributed parameter systems and, therefore, necessitate an approximation scheme for simulation and control/estimation purposes. An appendix provides weak form computations for both the plate and membrane model. A final section in the appendix details the finite element approximation of the stretch sensor model developed late in chapter 4. Utilizing the weak forms of the plate and membrane models, a finite element approximation scheme is used to discretize the systems in space, allowing for investigations of wing morphing control and aerodynamic load identification in chapters 5 and 6, respectively.
2 Biological Inspiration: Morphing Wings and Sensors

Birds, bats, and other species that fly have been of interest to biologists for years. Active study of such models in the engineering community has been increasing rapidly as engineers look to biology for inspiration regarding flight. Biological flyers tend to exhibit high levels of morphing and flexibility, allowing them to maneuver and exhibit outstanding efficiency greatly exceeding that of engineered aircraft. It is generally accepted by biologists and scholars of other disciplines that more species than merely Homo sapiens possess and utilize sensors in the body that aid in successful locomotion. This chapter will highlight some of birds’ and bats’ flight characteristics alluded to above as well as some of the biological sensors that may be employed for feedback and subsequent motor control.

2.1 Morphing and Flexible Wing Designs

An intriguing biological fact is that most species of animals fly [14, 15]. In this dissertation, the focus will be birds and bats, or from an engineering perspective, fixed-wing plate models that support bending moment and fixed-wing membrane models that do not support bending moment. Whether plate- or membrane-like, both possess the ability to morph and flex their wings, maneuver with skill, and maintain efficiency. Of course they have to, for evolution has ensured competition between flyers just as it has for every organism. While it is well known that birds and bats bend and twist their wings during flight, just how and why they are doing so has only recently begun to rigorously be addressed. Answers to such questions might seem simple, but are actually difficult as they generally rely upon highly complex fluid-structure interactions, interactions between multiple body systems, etc.
When one looks at the flight characteristics of biological flyers, extreme gust tolerance, speed control, target tracking, high roll rate, and a variety of other desirable characteristics and abilities are seen [16]. These successes are even more pronounced when considering the low Reynolds numbers ($10^5$ or lower)$^1$ at which these abilities are demonstrated, for destabilizing laminar bubbles, flow separation, and boundary layer growth are all common at such Reynolds numbers [17]. During flight in a Reynolds number range of $10^3 - 10^4$, flow separation around an airfoil can lead to sudden increases in drag and loss of efficiency; the result of which can be seen when comparing the flight of larger birds, which can soar for extended periods of time, with smaller birds that must flap continuously to remain aloft. The Reynolds number of larger bird species is generally $> 10^4$, compared with hummingbirds, for example, which operate at an approximate Reynolds number of $10^3$.

To relate to engineering for a moment, these effects are marked and difficult to overcome via traditional means of wing actuation at the scale of MAVs. Thus, engineers are looking to biology for inspiration in the development of a new class of flying machines. These flight vehicles are envisioned as having, like birds and bats, continuously morphing and highly flexible wing designs allowing them to maneuver more like biological flyers. To actualize such flight vehicles, science and engineering must strive to better understand just how biological flyers achieve such feats and how an engineered system might be constructed to achieve the same.

As hinted at earlier, wings are classified by engineers as fixed, flapping, or rotating; in this work, only the first is considered. For a great introduction to the biology and

---

$^1$Reynolds number is a nondimensional characterization of flight, allowing comparison between geometric scales, velocities, and densities of objects (esp. wings), specifically giving a measure of the ratio of inertial forces to viscous forces, and consequently quantifying the relative importance of these two types of forces for given flow conditions. It is given by the formula $Re = \frac{\rho v L}{\mu}$, where $\rho$ is fluid (air) density, $v$ is fluid velocity, $L$ is characteristic length (wing chord), and $\mu$ is dynamic viscosity.
physics of flapping flight, consult [16]. Also recall that fixed does not necessarily imply rigid, as many fixed wings in biology are indeed flexible. Bird and bat wings are fundamentally different structures, particularly when it comes to modeling them. A bird wing is an example of a plate that is clamped along one edge (in the bird, clamped to fused bones down the middle of the body for support, stability, and strength) with the feathers forming and maintaining a smooth surface during flight. A bat wing is an example of a membrane that is, again, clamped on one edge (to the body) but supported around the remaining edges by flexible arm, hand, and greatly elongated finger bones. From an engineering structures viewpoint, such differences are in fact encapsulated by thin plate and membrane theory, respectively.

It is well known that birds morph their wings to maneuver by utilizing a variety of strategies including flexing to control speed and direction, folding to reduce lift, and flaps to accommodate gusts and to adjust for landing; see the examples in figures 2.1 and 2.2. Bird wings are made of multiple layers of interconnected feathers, themselves relatively flexible, which allows for planform adjustments for particular flight modes [16]. This physiology allows birds to rapidly transition from cruising to quick maneuvering or landing. The anatomy of birds allows for a wide range of wing configurations, each of which is useful for a particular flight task [18].

Wing behavior, or function, during flight is consequential of its form. The humerus is a relatively short and powerful bone (interestingly, also part of the respiratory system), bearing the main stresses during up and downstrokes. The elbow joint is where the humerus meets the ulna and smaller radius, which together support the mid-wing and allow for twist during flight. Towards the wing-tip, one finds the wrist bones, a digit, fused hand bones, and the remaining digits. Also important to note is the patagial tendon and patagium. These are included in the discussion for their cruciality to bird wing anatomy and proper functioning during flight as well as their
likely role in proprioception (discussed in the next section). The patagium of a bird wing is a thin, flexible, expandable, membranous fold of skin. Similar to the patagium of a flying squirrel which acts like a parachute, catching the air and aiding in gliding flight, or that of a bat, the patagium of a bird aids in flight as well. Together, with the twist-allowing arm bones, the feathers of the bird wing form a complete and continuous \textit{plate-like} surface when compared with a bat wing membrane. See figure 2.3 for a side by side comparison of a bird and bat wing.

Bats morph their wings differently than birds, a direct result of their structural differences [16]. Unlike birds, bat species have flexible bones that taper towards the wing-tip to reduce mass and allow the wing overall to exhibit much greater flexibility. The wing is essentially a mammalian arm attached to a hand with quite elongated fingers, all connected by flexible membrane tissue. The bat wing consists of more than two dozen independently controlled joints (the elbow, wrist, and finger joints), allowing a high degree of articulation [19, 20]. A simplified bat wing schematic is illustrated in figure 2.4.

The uniqueness of bats’ wings allows them to perform extraordinary feats during flight. The membrane tissue between the fingers is highly specialized for flight, being
Semi-rigid feathers

Flexible membrane

9

highly anisotropic and varying up to 1000:1 in spanwise vs. chordwise stiffness [21]. Such anisotropy allows for development of specific camber during flight [22]. It is hypothesized that flexibility, anisotropy, and active membrane control are the primary reasons bats are able to perform with seemingly unequaled flight capabilities in terms of both aerodynamic performance and efficiency [23, 24]. On the subject of “active membrane control,” bats may very well sense (elaborated on in the next section) and change the tension in their wing membrane by utilizing muscles located within the membrane itself, as pictured in figure 2.5 and suggested by Gupta in [25].

Figure 2.3: Bird wing schematic

Figure 2.4: Bat wing schematic

Figure 2.5: Bat wing plagiopatagium (largest membrane section of wing) muscle bands connected to stiff fibers running spanwise. Image courtesy of Swartz Lab, Brown University
Bats have even been referred to as nature’s flight specialists [26], as many recent studies indicate membrane wings exhibit very desirable characteristics when compared to their more rigid counterparts. For instance, bats can land upside down, and do so rather gracefully [27]. Bats are capable of slow-speed-turning flight as well as quick, sharp changes in flight direction, and the ability to navigate rapidly through cluttered environments [28]. Because the membrane is stretchy and has some pretension, bats can flex their wings a little, reducing the span by about 20%, but they cannot flex their wings too much or the wing membrane will go slack [16]. Slack membranes are inefficient because they are prone to flutter [29].

Common to both birds and bats, however, is the ability to alter their wing span, which serves to decrease wing area and either increase forward velocity or reduce drag during an upstroke [29].

The consensus of the research on biological animals is, very generally, that morphing and flexibility are the primary explanations for their abilities. It is, therefore, not surprising that engineers have long been fascinated by such abilities and try to construct aircraft using biology as a guide.

2.2 Sensors: Closing the Loop

This section is devoted to an overview of sensors found in birds and bats that are most likely employed during locomotion. The focus is on skeletal muscle sensors, or muscle spindles, common to birds, and hair cell sensors hypothesized to be utilized by bats [8, 10]. Engineers may be able to duplicate such systems, and, as the reader will see in later chapters, this dissertation addresses a possible means of identifying and responding to pressure distribution on a wing using sensors inspired by the information presented here.
Before delving into this subject, note that, even though they are members of the same Mammalian class as humans who unconsciously employ muscle spindle sensors during motor control, it is not yet known for certain what function the muscles in the bat membrane perform (see figure 2.5). Research on that subject is being done at Brown University at the time of this writing. What is known, however, about muscle, or “stretch,” sensors is presented here. Since such sensors are common to vertebrates, and as bats are the second most widely distributed order of mammals, it is not illogical to presume the bat also utilizes feedback from such structures in its wings. Sensors are essential components to the existence of life – all animals have developed a plethora of sensory systems that capture information from their physical environment, ultimately ensuring their survival [30].

Proprioception is one of, if not the most important sense experienced by an organism: the sense of the relative position(s) of adjacent parts of a whole (body or wing e.g.) and the extent of strength being exerted during movement. Proprioceptors such as muscle spindles, joint receptors, and cutaneous receptors collaboratively sense different states of a limb, the force on it, speed of, position of, and adjacent joints’ stiffness, and this information is relayed to the brain. Muscle spindles are located within muscles and run parallel to muscle fibers. They detect muscle length and contractile velocity, or change in and rate of change of length. Muscle spindles are encapsulated, wedge-shaped, and tapered on either end (see figure 2.6). Cutaneous, or tactile, sensory receptors, on the other hand, are located in the dermis or epidermis in the form of hair cells that provide aerodynamic feedback and aid in control during flight [31, 32], as supported by evidence from the engineering literature [8, 10]. A system containing a diverse set of sensor types results in enhanced stimulus discrimination [33]. It should be emphasized, too, that the work in this dissertation deals
primarily with unconscious proprioception, as a bird or bat flying is equivalent to a human walking or running [34].

Figure 2.6: Simplified muscle spindle

Sensory feedback occurs when centers in the brain compare what is happening with what was expected to happen, and it is on this basis that compensatory adjustments are made. Following this two-way communication, subsequent motor control results, which is defined as desired postures and movements [34].

Recall that a bird or bat flying is equivalent to a human walking or running – learned actions that have, over some amount of time, become automatic. As soon as a given action has been or even just begun to be learned a plan of motor action, or program, is executed every time that action is to be carried out. So proprioception is key in muscle memory, meaning training greatly improves this sense [34]. This “motor memory” not surprisingly involves both how it feels to the subject to perform the action as well as what is achieved. Thus, motor learning requires aspects of motivation, alertness, concentration, balance, and other senses such as sight and hearing. It is accomplished in successive stages of giving selective responses to a stimulus, transitioning from non-specific responses to highly selective associations [34]. Most motor actions are therefore generated from both real-time, sensed information and programs based on past experience. These programs can be applied in a vast variety of situations because the central nervous system adapts to its environment. Furthermore, there also exists a process within the body that keeps muscle spindles pre-tensed and regulates sensor gain, so the sensors remain adaptive to most if not all loading situa-
tions, static or otherwise [34]. All of the above is discussed in detail in, among many others, [34–39].

Coupling between neural information processing and emergent mechanical behavior is inevitable as muscles, sense organs, and the brain all interact to produce coordinated movement in complex terrain as well as when confronted with unexpected perturbations [40]. Biological organisms like birds and bats have developed sensory adaptations to suit their particular and varying environments. The biological information presented in this chapter serves as inspiration for morphing and adaptive estimation and control approaches in engineering as well as strong motivation for this research.
This engineering literature review is comprised of two main sections: first, morphing and flexible wing designs, and second, load identification.

Most work done on wing morphing has been from an experimental point of view. Few investigations have been completed involving theoretical and/or optimal control approaches. This is largely due to the complexity of the wing structures of aircraft, type of actuation, and computational limitations existing until very recently. The type of actuation is obviously of integral importance for morphing a wing. Traditional actuation methods as applied to wing morphing will first be reviewed, followed by smart material actuators that show promise for achieving a truly continuous morphing wing design. And concluding the first main section of this chapter will be a review of membrane wing designs.

Strictly the problem of identifying the load and/or forces on a wing from dynamic sensory data (e.g., strain or acceleration) will be addressed in the second main section of this chapter. The formulation to directly solve a nonlinear parameter or disturbance estimation problem is generally referred to as an inverse problem, for which an immense amount of literature exists; hence a comprehensive literature review of all applicable methods is obviously beyond the scope of this dissertation. What will be discussed, however, is a chronology of aerodynamic load identification followed by a more in-depth look at the disturbance observer design as a potentially real-time control solution.
3.1 Morphing Wing Designs

While no general definition for morphing is agreed upon, within the field of aeronautical engineering it is defined as a set of technologies that improve vehicle performance by manipulating vehicle geometric state to match the current flight environment [41]. Traditionally, rigid wing geometry is engineered as a compromise between the “optimal” designs, or wing states, that exist for the particular flight regimes at which a given aircraft will be expected to operate. For instance, it is well known that a wing designed for slow, loitering flight is not optimal for a high velocity flight regime requiring great maneuverability. Two general types of wing morphing exist: wing planform changes, which include span, chord, and sweep modifications; and out-of-plane transformations, which include twist, dihedral/gull, and spanwise bending modifications. Airfoil adjustments like camber and point of maximum camber modifications can be considered a third type of morphing, but are taken to be a subset of out-of-plane morphing in this dissertation. In the context of approximating wings with thin plates, two distinct classes of deformation are differentiated: axial, or deformations tangent to the mid-plane of the wing; and transverse, or deformations in the z-direction, or out of the wing plane.

Morphing wing designs have existed since the dawn of engineered flight, and certainly before in the imaginations of ancient scientists and engineers. Morphing, by the definition above, for flight control was first utilized on the Wright Flyer in 1903. Cables were attached to allow the pilot to twist the wings to achieve a desired configuration [42]. However, due to power requirements of actuators to change wing shape, wing morphing methods were largely abandoned [43]. Generally speaking, large shape changes of a wing usually have associated design penalties such as added weight or complexity. If such penalties did not exist, morphing designs would always
be desired [44]. As Bowman et al. showed however, there is a crossover point where a fuel penalty for not morphing exceeds the morphing weight penalty [45].

Evidence is mounting that wing morphing enhances the aerodynamic performance and robustness of unmanned aerial vehicles (UAVs) and MAVs and a variety of performance gains are made by allowing a wing to deform in flight [11,16,46–52]. Indeed, it has repeatedly been demonstrated that use of a morphing wing, whose geometry varies according to external aerodynamic loads, results in increased aerodynamic performance during cruise and maneuvers, provided the wing can be morphed to optimize the flow at every point during the mission [53–55].

Roth et al. [56] showed morphing could have a large impact on fleet size for the U.S. Coast Guard patrol by allowing craft to operate in multiple flight regimes effectively. This, among other studies, has sparked great interest in manned, morphing aircraft. For example, an estimated 1% reduction in drag could save the U.S. fleet of wide-body transport aircraft up to $140 million/year (studies at NASA Dryden) along with noise reduction [44]. A variety of other studies have demonstrated the benefits of morphing, or mission-adaptive, wings for larger craft, for instance Smith et al. [57], Wittmann et al. [58], and Rodriguez [59]. However, at the scale of small UAVs and MAVs (5cm - 1m wingspan), improved maneuverability and performance are of primary concern, but fuel is obviously an important factor as well. Considering expendable craft at this scale, fuel becomes even less a concern compared with performance. It is the author’s intention hereafter to limit the review to UAV- and MAV-scale craft, or at least to work applicable at this scale.

Just under a century after the Wright brothers, Munday and Jacob [60] demonstrated separation could be reduced significantly by oscillating the camber of a wing in a specific way, compared with a static wing at the same angle of attack (AOA). A couple years later, Abdulrahim et al. [61] and Stanford et al. [62] directly applied
a few very basic wing morphing techniques. They utilized Kevlar cables and torque rods attached directly to the underside of a flexible wing MAV as well as servo motors and investigated both wing curling and twisting as means of flight control. It was successfully demonstrated that a wing made out of flexible material can be morphed with little power.

Concurrent with the previous study, Abdulrahim [63] and Garcia et al. [64] investigated wing morphing via a 24-inch craft capable of gull-like wing motions by means of two hinged spar structures on either wing. At the fuselage, a single hinge coordinates both wings’ movement to form a symmetric actuation scheme. The relative deflection of each wing is thus controlled by this linear actuator in the fuselage. The outer hinge joint on each wing behaves passively in response to the fuselage hinge. This design allows the craft to maintain position under any wing loading without energy consumption or control effort. The biologically inspired capability of folding its wings like a gull had an effect on this MAV’s flight performance, including improved climb rate, glide angle, and stall reduction. Additionally, remote control pilots stated flying and handling the craft was a substantial improvement over traditional designs.

Guiler and Huebsch also investigated wing morphing utilizing torque rods to achieve wing twist in [65]. They demonstrated such morphing provides adequate control forces and moments to control a UAV. Although torque rods have repeatedly been shown to be successful morphing/twisting mechanisms, there are drawbacks to such a design. As stated earlier, weight becomes an issue, for extra servo motors, rods/cables, and pulleys are required for such designs.

Barbarino et al. [44] provide a wide-ranging review of morphing aircraft, from actuation schemes such as those used in the referenced works above to smart material designs. This review captures the current state of the art in morphing designs, and
the reader wishing to learn more about many craft and studies not mentioned here is strongly encouraged to consult it.

Altogether, although many wing morphing designs have been proposed, very few have been tested in MAVs and UAVs [61]. There are many reasons for this, the primary being, until recently, technological limitations. The utilization of smart material actuators to achieve wing morphing is of particular interest in this dissertation. For this reason, the discussion is now geared more towards investigations into technological advances in smart material actuators, as they promise to greatly enhance performance of small aircraft [66,67].

As morphing wings become more complex, so does the control of such structures. In the very near future, human ground or remote pilots will be unable to fully control the craft to their full potential. Considering the interest in smart materials for wing morphing (discussed in the next subsection), it is likely systems with dozens if not hundreds of sensors and actuators could come to fruition in the not so distant future. This will require on-board control and estimation approaches, which would free the need of a human pilot, provided the field of control can “keep up with” the advancement of aircraft technology.

3.1.1 Morphing Wings and Smart Materials

Birds, bats, and other flying animals generally use antagonistic muscle pairs for wing actuation. While such an actuation type is available to engineered systems in the form of hubs, hinges, and servo motors, it has been shown to create discontinuities over the wing surface which can lead to airflow separation [68]. Contrarily, there are several benefits of continuous morphing control via active smart materials over discrete trailing edge control using conventional control surfaces. Traditional control
surfaces are less effective at the low Reynolds numbers encountered in the MAV and UAV flight regime, and such small craft cannot afford to lose energy through control surface drag because of their inherent power limitations. Continuous, full wing morphing control designs allow for direct control of circulation, drag, and wing-tip vortices. Inspired to investigate smart material sensors and actuators in flight control by sensors found in biology, the various types of smart materials available to engineers are now examined.

In the last few decades, new types of sensors/actuators have arisen, often referred to as smart materials. Examples include classical piezoelectric sensors/actuators, shape memory alloys (SMAs), and a variety of specialized types that utilize either piezo or SMA materials, all to the end of improving deformation and/or sensitivity. Furthermore, by building smart materials directly into a system in a continuous manner, the discontinuities mentioned above could be prevented, and the design greatly benefitted [69]. Of course, this is providing such smart material actuators exhibit the control authority necessary to manipulate the wing surface. Such is the general issue with smart materials: their limitation of small deformations restricts actuation ability and, therefore, application. Other limitations, in the case of piezo materials, include high voltage requirements, the fact they are generally brittle and/or difficult to conform to the system geometry to which they are being applied, and the non-linear nature of their constitutive equations, making their use in traditional feedback control designs difficult. Although much research has been devoted to SMAs in morphing designs, a detailed discussion regarding them is not presented in this work. See Pagano et al. [70] and Prahlad and Chopra [71] for further discussion.

Piezoelectric materials, on the other hand, offer relatively high force output in a wide frequency bandwidth. Although the strain output is relatively low when compared with other materials and types of actuation, the fast response of piezo materials
is useful, as that is expected for UAV and MAV wing morphing [44]. The limitations just described are largely responsible for the lack of studies using piezo materials to achieve wing morphing at the scale of MAVs. However, other uses for piezo materials have been found that embrace the small deformation limitations: piezoelectric actuators have been used for stall control applications via simple mechanisms such as buzzing [72, 73].

Interest in wing morphing using piezoelectric materials has generally been present since the early 1990s. Static control of aerodynamic surfaces began with a study of adaptive, box-wing structures for aeroelastic control, investigated by Lazarus et al. in 1991 [74]. In 1997, Pinkerton and Moses [75] investigated the use of thin-layer composite, unimorph ferroelectric sensors and actuators (also known as THUNDER actuators) to morph a wing. Although piezoelectric in nature, such actuators are inherently nonlinear in their behavior, and hysteresis was observed. Models of these actuators are discussed in detail by Smith in [76], where it is noted nonlinear models will be required to capture the physics of THUNDER. In 2001, Munday and Jacob [77] developed a wing with adaptive curvature also using THUNDER actuators. The design was capable of both static and dynamic control and allowed modification of wing thickness and maximum camber point. Bae et al. [78] demonstrated camber control of a large, flexible wing UAV using piezoelectric actuation. In addition, a large number of studies have also been devoted to helicopter blades. Although these structures are not of particular interest in this work, the overall design concepts are not completely unrelated. Relevant studies include those by Grohmann et al. [79, 80] and the references therein.

THUNDER actuators have been demonstrated as a means of controlling a morphing wing, but their nonlinear and hysteresis behavior present difficulties from a control and estimation standpoint. Because of these difficulties, engineers are seeking better
behaved, i.e. linear actuators that nonetheless provide sufficient deformation capability to be useful for a wing morphing design. A promising sub-type of smart piezo material has arisen out of NASA Langley Research Center that provides the actuation authority needed to both morph a wing as well as maintain quasi-linear behavior and hysteresis loop, simplifying control algorithm development [81–83]. These new piezocomposite actuators, macro-fiber composites (MFCs), are comprised of piezoelectric material fibers embedded in epoxy or resin. In-plane strains of up to .2% have been achieved with MFCs, which is orders of magnitude greater than standard piezoelectric actuators and materials.

While most studies involving MFCs have been limited to actuation strategies with the goal of changing the camber of rotor blade flaps, e.g., [84], more recent studies have demonstrated full wing morphing capability in addition to effectively changing the camber of a wing [85–87]. The morphing achieved in these studies also provided sufficient authority to control the studied craft, which was flown in both a wind tunnel and free flight. All electronics were powered by a standard, remote-controlled aircraft battery, thereby proving that such designs are even feasible outside of controlled laboratory conditions.

Smart material actuators are constantly being improved through advances in material science and manufacturing capability. However, at this time, few smart material actuators provide sufficient deformation to achieve wing morphing at the scale of MAVs. Those that are capable of wing morphing include SMAs, THUNDER actuators, and piezocomposite materials. Adding the associated relative difficulty in modeling such smart materials, piezocomposite actuators arise as superior, lending themselves to linear models and control. Also, piezocomposite smart materials have been demonstrated in actual flight tests to provide the much needed deformation authority for wing morphing, behave comparatively linearly, and are less massive
compared with traditional smart material actuators. For these reasons, piezocomposite materials are utilized in this dissertation to investigate morphing a wing with optimal control.

3.1.2 Membrane Wing Designs

Membrane wings are central to this dissertation, exhibit many passive morphing effects, and the evidence is mounting that lifting-surface flexibility can enhance the aerodynamic performance and robustness of MAVs [11,16,47,50–52], attracting both scientists and engineers’ attention. As mentioned in chapter 2, bats have membrane wings carrying with them a variety of interesting and useful flight characteristics. Many studies have already demonstrated that engineered membrane wings also exhibit these effects, for instance [3, 88–90] and the references therein. This section exists not to fully review the work done with membrane wings, but rather to give the reader a taste of the impressive characteristics of these highly flexible wings.

Membrane wings come in a variety of designs. While most linearly elastic wings are generally constructed of materials such as metal or carbon fiber, membrane wings are made of highly elastic or non-linearly elastic materials, allowing for much greater deformation. As the reader can probably imagine, such wings naturally billow as a result of flow pressures. Billowing increases the camber of the wing naturally, thus the wing passively contours to the pressure field unlike rigid wings that support bending moment and cannot contour so easily to the flow field. This natural behavior is the primary reason interest in membrane wings is expanding. Further examples of desirable behavior include passive adaptations to pressure distribution, flow detachment
via adaptive geometric twist or washout\(^1\), adaptive inflation for increased lift, and larger stability margins [90].

The main characteristic allowing for the behavior mentioned above is membrane wings’ inability to support a bending load moment. This reduces the inherent spatial low-pass filter property possessed by most linear elastic structures. Indeed, most structures effectively filter inputs of high spatial and temporal frequency to produce distributed deformations and vibrational responses. Thus, the deformations of membrane wings are more locally induced. As such, the effect of a point load, for example, on a membrane is drastically different than on a plate or beam, the effect being more localized and pronounced on a membrane. This makes membrane wings excellent candidates for spatial estimation and identification of pressure resultants, and this property is used in chapter 6. What this means for plates and beams is not that spatial estimation and identification of pressure resultants cannot be done, just that it may be less accurate and more sensitive to noise, in part due to substantially lower strain values encountered in such materials (especially for small MAV designs).

The lift characteristics of a membrane wing are superior to those of a rigid wing. Although the induced drag of a membrane wing increases due to the increased lift and billowing as is expected, the lift coefficient \(C_L\) nonetheless is more impressive than a rigid wing. For instance, figure 3.1 compares the coefficient of lift for a membrane vs. rigid wing. While the rigid wing begins to exhibit separation and stall at approximately 10 degrees AOA, the membrane wing dynamics delay the onset of stall, and much greater lift coefficient is exhibited.

Shyy et al. [51, 91, 92] investigated highly flexible airfoils that exhibited camber changes in response to aerodynamic loads. Flexibility in the airfoils improved un-

\(^1\)That is to say, the wing changes shape due to the local pressures on the wing such that the local angle of attack along the wing chord is conveniently approximately that required to delay stall and improve gust tolerance.
steady performance by limiting flow separation at high AOAs. However, the studied membrane wing models exhibited degraded lift-drag ratio compared with rigid wings of the same planform.

Another example of a membrane-wing MAV is one constructed using carbon fiber battens and cloth prepreg materials in the early 2000s. Ifju et al. [93] developed this MAV that exhibits much more resistance to stall than its rigid wing counterparts. A detailed review of this craft can be found in [16] (along with reviews of many of the craft mentioned in the previous section on morphing designs).

Albertani et al. characterized membrane wing dynamics in [94], and Ray and Albertani then investigated membrane wing-tip vortices and the effect of Gurney flaps on membrane wing designs in [1]. The wing used in these investigations is illustrated in figures 3.2 and 3.3, with figure 3.2 demonstrating the flexibility of such a design as it deforms due to a 20g mass placed in the center of the wing. Note the
deformation contours to the mass location itself. The deformation is far greater than a carbon fiber plate-like wing would exhibit.

![Figure 3.2: Membrane wing deforming due to 20g mass](image1.png)

![Figure 3.3: Top view of elliptical planform of wing used in [1]](image2.png)

Membranes and membrane wings are used by many organisms. Biologists are finding not only are they aerodynamically impressive, they contain sensors that are probably used for load/pressure feedback, the engineering literature for which will now be reviewed.

### 3.2 Aerodynamic Load Identification: A Review of Existing Methodologies

In this second section of this chapter, the possible approaches for utilizing sensory data to identify aerodynamic load are reviewed. As the topic of load identification is relatively broad, the organization of this chapter’s remaining subsections is: Firstly, general, non-sequential, offline approaches to load identification found in the literature are reviewed. Also grouped into this category and briefly reviewed are ad hoc approaches. Secondly, a brief overview is given of the large number of distinct recursive approaches, mostly based on Kalman estimation or Bayesian cost functions at their core. An approach stands out as the general basis for most specialized disturbance
observer designs, one that is described in many ways but most commonly referred to as the joint extended Kalman filter (JEKF). It is a general, real-time/quasi-real-time approach and was chosen for its relevance to the problem at hand in this dissertation, for expansion in chapter 6, as it allows for quasi-real-time identification of external disturbances such as loads, forces, and pressures.

3.2.1 A Chronology of Aerodynamic Load Identification

The problem of load identification from structural measurements, such as strain, position, velocity, and acceleration, has long been of interest to engineers attempting to estimate or reconstruct structural state and/or external disturbances (load, moments, etc.). The benefits of being able to monitor states that are effectively external to the structure itself are multitudinous, including disturbance rejection, modification, and simply measurement. One example, the problem at hand in this dissertation, is identifying the load distribution on a wing, be it rigid, linearly elastic, or even a highly elastic, nonlinear membrane. Estimating such a pressure distribution would be beneficial in two major ways: First, using very simple aerodynamic theory, tracking an “optimal” pressure distribution would be possible. Second, it would allow for structural health monitoring, as a damaged wing may not provide the lift expected in a certain region when compared with other regions of the wing. If an engineered, mission-adaptive wing is to come to fruition, it will likely require feedback of both structural and disturbance states.

In 1972, Pilkey and Kalinowski utilized mathematical models and system identification techniques to identify shock and vibration forces in [95]. Hillary and Ewins [96] used strain gauges for both force identification and frequency response function measurements for structures in 1984. Gregory et al. determined the forces acting on
nonrigid bodies [97] and Wang investigated force identification from structural response [98] in 1986 and 1987, respectively. Stevens presented an overview on force identification problems in 1987 [99]. Maniatty et al. studied inverse elastic and elastoviscoplastic problems using the finite element method, utilizing regularization techniques to yield smooth solutions to the problem [100]. Starkey and Merrill discussed the ill-posed nature of indirect force measurement techniques in 1989 [101]. Using structural response to estimate force is an ill-posed, and potentially ill-conditioned problem and demands regularization [102], as will be discussed in chapter 6. Park and Park estimated the force, time, and location of an impact force on a beam using wave propagation theory and strain measurements in 1994 [103]. On a similar note, in 1997, Kirby et al. recovered the shape of a beam based on strain data using an inverse method [104]. The strain field was simply directly integrated using polynomial strain field assumptions.

On a more relevant note, in 1998, Cao et al. [105] utilized multilayer neural networks to determine load-strain relationships. By training a neural network on known combinations of load and strain, general loads could be estimated. The learning parameters for the neural network were very particular, often causing failure of convergence if improperly selected. Published the same year, Johnson derived necessary and sufficient conditions to ensure convergence of the load estimated by solving the underlying inverse problem of load identification from structural response [106].

Specifically interested in aeroelastic loads, Eksteen and Raath reconstructed the time history of a load for fatigue testing by approximating the load as quasi-static in 2000 [107]. Similarly motivated in 2001, Shkarayev et al. developed an inverse formulation utilizing a finite element model to recover the loads, stresses, and displacements of aerospace vehicles from strain data [108]. In the same year, Law and Fang attempted to overcome the common weakness of large fluctuations in identified
load results (due to noise and/or inherent ill-conditioning) via a dynamic programming approach [109]. They successfully provided bounds on the estimated loads.

In 2002, Chock and Kapania utilized classical steepest decent to solve the load identification problem from sensed displacement measurements corrupted by noise [110]. They made the assumption loads could be represented by polynomial functions and found that lower order polynomials yielded better results. Li and Kapania extended this work to nonlinear finite element models in 2004 [111].

In 2005, Coates et al. investigated identifying loads under the assumption the loads can be represented by single term Fourier cosine series [112]. Specific load functions were chosen as a basis for representing general, unknown load functions. The coefficients of the Fourier series were then estimated and compared to a database of known coefficients and loads, at which an estimate could be arrived by combining basis load functions accordingly. Accuracy was lost for higher order distribution functions. Coates and Thamburaj extended this work in 2008 [113] to two independent variables using single and double Fourier series and identified the “most likely” load using this approach. Not surprisingly, they found, for a small strain data set, the expected load functions must be modeled well by the Fourier series.

Many of the approaches mixed in with this history of load identification utilize basic stress-strain-displacement relationships and direct integration or differentiation to arrive at load estimates. These approaches are teeming with assumptions regarding both strain and load distribution, but have been shown, nonetheless, to be qualitatively accurate. If one knows the expected pressure distribution function, one can effectively directly estimate such a distribution from strain on a structure, provided the structure is linearly elastic and geometrically simple. However, pressure distributions are rarely, if ever, known exactly, and even a slight difference in the functional
form of the quantities one is estimating may lead to disastrous consequences for a flight vehicle.

Many of the above approaches are only accurate for static loading conditions or must be computed offline due to computational complexity. A relatively simple but powerful set of recursive identification tools that lend themselves to control designs and allow for time-varying solutions to the load identification inverse problem are now discussed. These recursive techniques have not, to the author’s knowledge, been applied to the specific problem of load identification; many involve quite disparate inverse problems or applications. The general techniques, however, are the same and fully apply to load identification with modification.

3.2.2 Disturbance Observer Design: A Potentially Real-Time Solution Built for Control

A variety of recursive approaches exist that solve inverse problems, the most well known being the Kalman filter. Several fields are aware of Kalman filter techniques for joint state-parameter estimation inverse problems including geology and water science [114], biomedical engineering [115], speech enhancement [116], electric car/battery management systems [117], fault detection/delamination of composite materials [118], among other engineering uses of particular relevance to this dissertation.

Recursive inverse methods branch into a variety of methods such as maximum a posteriori (MAP) and maximum likelihood (ML). Depending on specific formulations and circumstances, the Kalman filter itself provides both a MAP and ML solution. A Kalman-based approach is utilized in this dissertation. This general approach branches into three different filter types: JEKF, dual extended Kalman filter (which can, itself, generate approximate solutions to a variety of cost functions), and
other Kalman filters [119]. Disturbance observer designs utilize the above filters to approximate disturbances to a system, such that a better estimate of the true system dynamics can be achieved.

Various Kalman approaches are found in the literature, as are other statistically or deterministically derived approaches such as recursive least squares (the algorithm which serves as much of the basis for the Kalman filter itself). Fundamentally though, most methods arise from the minimization of a common, general cost function (which results from specific statistical assumptions and/or circumstances) and only differ in basic assumptions. For a rigorous discussion of some of these algorithms, including derivations using a statistical framework that directly illustrates the connections between the algorithms, see Nelson and Wan et al. [119, 120].

While the specific approach utilized in this dissertation is novel, it is based on a Kalman filter framework dating to the early 1960s [121]. As early as 1963, it was suggested by Kopp and Orford [122] that state estimation theory can be used to not only estimate the states of a system, but also the unknown parameters (and therefore disturbances) of a system. In 1967, Carney introduced joint state-parameter estimation and discussed a variety of approaches, comparing the Kalman approach to least squares and ML approaches, demonstrating the equivalency of the approaches when proper weighting matrices are chosen (along with specific noise assumptions, etc.) [123]. In 1969, Basil and Mono investigated the criteria of the observability of systems with unknown inputs [124]. This was followed by the development of the continuous unknown input observer (UIO), or Kalman inverse filter, by Bayless and Brigham in 1970 [125], in which the approach was used to remove sensor reverberation from seismic sensor signals. Shortly thereafter, the approach was used again in geophysical analyses to solve the problem of deconvolution of seismic signals [126].
The Kalman inverse approach, when used in the context of state estimation for the purpose of control, is largely referred to as an UIO. Strictly referring to the UIOs, Johnson introduced a systematic approach to designing feedback control using disturbance estimates in [127], which was expanded on in [128, 129]. Here, Johnson suggested with knowledge of an external disturbance, perhaps control could be designed to harness the disturbance itself to reduce cost. The work has broad implications and is applicable in general to any persistently acting disturbances such as forces, torques, and voltages.

Since Johnson’s work, the general disturbance observer concept has been applied to a variety of problems including force and friction estimation [130–136]. The technique is catching on in even more fields such as aerodynamics and mechanical engineering, having been applied to aerodynamic coefficient and parameter estimation problems [137–139] as well as vehicle tire force and machine friction problems [140–142]. In 1990, the UIO was extended to nonlinear plants [143], and, in 1995, was used for fault estimation [144].

McAree summarized the attractions and limitations of the UIO approach in [145]. The main attractions are its causality and suitability for real-time implementation, its natural dealing with model and measurement uncertainty, and its ease of adaptability to nonlinear systems. There are also limitations to the approach such as its sensitivity to being carefully designed so as to not reduce the stability margin of a feedback law, as evidenced by investigations by Coelingh et al. [146] and Tesfayc et al. [147]. Another positive, it enables secondary, extremum-seeking control systems

---

2Fundamentally, the problem of identifying unknown inputs (for structures – forces, pressures, moments, etc.) to a system is one particular type of unknown and possibly time-varying parameter identification problem. For linear systems, a modification of a system parameter might be representable by a time-varying disturbance input to the system. For this reason, all types of unknown parameters, including model errors, mismatches, unknown inputs, colored noise, etc., are lumped under one title: unknown disturbances.
and, as Johnson suggested in [127], control cost reduction. For example, consider a membrane wing with displacement control input. If the problem calls for a certain membrane deflection, and there is a pressure present on the membrane that results in a deflection, the actuators may not be needed, or may only be needed minimally, to achieve the desired deflection, for the external pressure has done the job itself. Another limitation is the need of a plant model along with the structure of the inputs/disturbances which are, in general, unknown. Furthermore, the bandwidth to which inputs/disturbances can be reconstructed depends on the dynamics of the plant and the process and measurement uncertainties specified in the Kalman filter. An additional drawback to this approach is that a statistical characterization of the covariance of the unknown disturbance is needed in order to achieve an optimal estimate. For many disturbances, the engineer will simply lack such a characterization. It is thought that adaptive Kalman filter techniques can be used to overcome the drawback of required foreknowledge of state and measurement statistics [148], but little work has been done to develop such a technique.

The relationship between recursive solutions to an inverse problem and the disturbance observer formulation is obvious: a disturbance observer is a recursive approximation to an inverse solution that is directly used in feedback control/estimation. Thus, any recursive approach that provides both an estimate of the system and disturbance states can be used in a disturbance observer design so long as that recursive method provides sufficiently fast convergence for the time scales present in the system dynamics.

Although the UIO approach has largely been applied ad hoc – that is, the parameters of the approach include physical process, measurement noise covariance, and filter initial conditions having been “tuned” rather than rigorously measured and/or computed – it has, nonetheless, been very successful. At the time of this writing,
there has not been a direct application to distributed parameter systems. This may be due to the resulting system dimensions after discretization. However, the author hypothesizes model reduction techniques may be applied to alleviate such difficulties.

To conclude, dissimilar to the previous subsection, focused on in this subsection were recursive methods and a novel Kalman UIO approach. Again, this novel approach solves the same fundamental inverse problem but recursively, updating an estimate with each measurement taken at discrete or continuous time steps. It proves very useful from the perspective of designing system control laws since it simultaneously estimates external disturbances and system state, thus enabling feedback control. The UIO utilized hereafter in this work is a specific Kalman inverse filter approach, referred to in general as a JEKF, that relies upon basic knowledge of how a disturbance enters a system and the differential equation that “generates” it. Also, since the approach depends heavily on parameters for physical process and measurement noise covariance, and regularization, such parameters are considered tunable in that they may be manipulated for individual problems.

3.3 Inspiration and Review Summary, Hypotheses, and Dissertation Contributions

Prior to delving into the development of the models and results, it is useful to summarize this dissertation thus far as well as what will be contributed in the remaining chapters.

In chapter 2, it is discussed how birds and bats morph their wings during flight. This leads to the hypothesis that such abilities improve flight and flight control. This hypothesis is largely experimentally addressed in the engineering literature. The general consensus of which is that incorporating smart materials into flexible MAV
designs is an effective way of achieving morphing and camber control in a continuous, unobtrusive manner. The work presented in chapter 5 therefore entails developing a general approach to control design for wing morphing. Model-based control and smart material actuators are utilized to morph a wing, in simulation, into biologically inspired positions. Specifically, MFC actuators and a model-based control algorithm are utilized to achieve wing morphing similar to that found in Ray et al. [5]. While the general linear quadratic Gaussian (LQG) control solution is well documented in the literature, successfully implementing such a control algorithm in this application has yet to be investigated. It is confirmed that model-based control algorithms such as LQG are appropriate for this application and can be used in a gain scheduling approach to online, autonomous control of a thin, flexible wing. Such a control algorithm is required for a continuously morphing wing with distributed sensors and actuators.

It is found in chapter 5, however, something is missing. Such a wing control algorithm will have no knowledge of the resultant pressure distribution on the wing surface and, thus, no feedback to “close the loop” between the control system and incident flow. Looking to biology, the science of proprioception and investigations of muscle sensors and actuators suggest flying animals feel their way through flight in much the same way their ground dwelling relatives sense their movements, terrain, and interaction between the two. This includes sensing pressure distributions, lift/moment resultants, etc., such that the adaptive structures can be used to their full potential. It is well known that flightless vertebrates utilize muscle spindle sensors, among other sensors, for limb position and velocity feedback. Therefore, it is hypothesized a stretch sensor model can be used to estimate pressure distribution on a wing. It is for this reason a specialized JEKF algorithm is utilized in chapter 6 to form such an
estimate from stretch sensor feedback on a simulated membrane model, and results are compared to experimental wind tunnel test data.

The next immediate chapter entails relevant model derivations. A generalized anisotropic thin plate model is first derived, including the associated piezocomposite sensors and actuators. A membrane and stretch sensor derivation follow.
4 Model Development

In this chapter, the linear plate and membrane models used throughout the remainder of this dissertation are derived and discussed, along with relevant sensor and actuator models. For a general discussion of plate models, from which membrane models may be derived, see [76, 149] and the references therein.

4.1 Thin Plate Model

A thin plate model is a common approximation of a wing. Although linear, a thin plate model captures much of the important dynamics of a wing and allows for simulation of such a system with distributed smart material sensors and actuators.

In this work, a plate is considered purely in a bending state, i.e., only transverse loads are acting upon the plate, which cause purely transverse deflections. To begin the derivation of the plate equation of motion, the hypotheses made in classical plate theory are stated as [76]

- The plate is very thin, i.e. \( h \ll L \) and \( h \ll W \), where \( h \), \( L \), and \( W \) are the plate thickness, length, and width, respectively.

- Transverse deformations are small compared with the thickness of the plate.

- Strains in the middle-plane are negligible.

- Stresses normal to the middle-plane \( \sigma_z \) are identically zero everywhere (see figure 4.1 for definition of the middle plane).

The plate thickness and material parameters are also allowed to vary spatially, which simplifies modeling material inhomogeneities.
With the above assumptions in place, consider a plate of length $L$, width $W$, and thickness $h$. Let $\Omega \in [0 \, L] \times [0 \, W]$ denote the plate’s domain of support. The edges of the plate are counted counter-clockwise and labeled $\Gamma_i$, for $i = 1, 2, 3, 4$, beginning with the edge along the line $y = 0$. A total of $N_{pz}$ piezocomposite patch pairs are bonded to the plate with their edges aligned to the x and y coordinate system as illustrated in figure 4.1. Each patch is designated by its local domain of support $\Omega_i$, for $i = 1, 2, ..., N_{pz}$. All of the piezocomposite actuators have thickness $h_{pz}$; Young’s modulus $Y_{1pz}$ in the direction of the piezocomposite fibers and $Y_{2pz}$ in the direction orthogonal to the piezocomposite fibers; and shear modulus $G_{pz}$.

![Figure 4.1: Plate model geometric definitions. Left: middle-plane ($z = 0$)/coordinate system definition. Right: plate and patch geometry](image)

Consider an infinitesimal plate element with force and moment resultants illustrated in figures 4.2 and 4.3, respectively. Denote transverse shear force terms as $Q_x$ and $Q_y$, moments as $M_x$, $M_{xy}$ and $M_y$, and external load (pressure) by $f_n$. Note that the orientation of $M_{xy}$ and $M_{yx}$ are taken such that $M_{xy} = M_{yx}$.

Denote the density of the plate by $\rho$ and transverse deflection by $w$. Balancing forces in the z-direction, in combination with Newton’s second law yields

$$\rho \, h \, dx \, dy \, \frac{\partial^2 w}{\partial t^2} = \left( Q_x + \frac{\partial Q_x}{\partial x} \, dx \right) \, dy - Q_x \, dy + \left( Q_y + \frac{\partial Q_y}{\partial y} \, dy \right) \, dx - Q_y \, dx - f_n \, dx \, dy,$$
which, after simplifying and taking the limit as $dx, dy \to 0$ leads to the dynamic transverse shear force equilibrium equation

\[
\rho h \frac{\partial^2 w}{\partial t^2} - \frac{\partial}{\partial x} Q_x - \frac{\partial}{\partial y} Q_y = -f_n. \tag{4.1}
\]

Balancing moments with respect to the $x$, $y$, and $z$ axes about point $O$ in figure 4.3 yields

\[
\begin{align*}
Q_x &= \frac{\partial M_x}{\partial x} + \frac{\partial M_{yx}}{\partial y} \\
Q_y &= \frac{\partial M_y}{\partial y} + \frac{\partial M_{xy}}{\partial x} \\
M_{xy} &= M_{yx},
\end{align*}
\]
respectively. Substituting the above for $Q_x$ and $Q_y$ in equation 4.1 yields

$$\rho h \frac{\partial^2 w}{\partial t^2} - \frac{\partial}{\partial x} \left( \frac{\partial M_x}{\partial x} + \frac{\partial M_{yx}}{\partial y} \right) - \frac{\partial}{\partial y} \left( \frac{\partial M_y}{\partial y} + \frac{\partial M_{xy}}{\partial x} \right) = -f_n,$$  \hspace{1cm} (4.2)

which is the equation of motion for a thin plate.

Computing the moment terms requires a strain-displacement relationship and Hooke’s law. At this point in the derivation, damping is introduced. If it is hypothesized the plate physics are viscoelastic, the material can be taken to be Kelvin-Voigt, thus stress is defined as proportional to strain as well as the time rate of change of strain. In other words,

$$\sigma(t) = E\epsilon(t) + \eta \frac{d\epsilon(t)}{dt},$$  \hspace{1cm} (4.3)

where $\sigma$ is stress, $E$ is modulus of elasticity, $\epsilon$ is strain, and $\eta$ is the viscosity, or Kelvin-Voigt damping constant. The Kelvin-Voigt constant is generally determined empirically using inverse or estimation techniques or taken to yield qualitatively accurate results in simulation if the structure is complicated, as was done here. If the presence of air, or viscous, damping is also assumed, the combination of the Voigt hypothesis in equation 4.3 and viscous damping result in a Rayleigh proportional damping model. Care must be taken here so that the boundary conditions are modeled appropriately, as the addition of a damping term to the stress-strain assumption, equation 4.3, modifies the moment resultants of the thin plate model.

4.1.1 Moment Resultant Computation: Generalization to Anisotropic Materials

Fundamentally, the moment resultants in equation 4.2 are found by integrating the internal stresses over the thickness of the plate. For small deflections, this equation
can be adapted to model anisotropic composites by modifying the moment resultants accordingly, as follows. Note that square brackets will be used to denote matrices when written in single line equations.

Consider a single layer of a composite plate, which is essentially an orthotropic plate, assuming it is composed of unidirectional carbon fiber. Consider the fibers at some angle $\alpha$ from the $x$-axis such that, under a simple rotation, the plate will align with the reference coordinate system. From elasticity theory, the strains are defined by

$$
\begin{align*}
\epsilon^k_x &= -z_k \frac{\partial^2 w}{\partial x^2} \\
\epsilon^k_y &= -z_k \frac{\partial^2 w}{\partial y^2} \\
\gamma^k_{xy} &= -2z_k \frac{\partial^2 w}{\partial x \partial y}.
\end{align*}
$$

(4.4)

The generalized stress-strain relationship for an orthotropic material in plane stress is

$$
\begin{bmatrix}
\sigma_{x'} \\
\sigma_{y'} \\
\tau_{x'y'}
\end{bmatrix} = z
\begin{bmatrix}
E'_{x'} & \nu_{y'}\nu_{y'} E'_{x'} & 0 \\
\nu_{x'}\nu_{x'} E'_{y'} & E'_{y'} & 0 \\
(1 - \nu_{x'}\nu_{y'}) & (1 - \nu_{x'}\nu_{y'}) & G'
\end{bmatrix}
\begin{bmatrix}
\epsilon_{x'} \\
\epsilon_{y'} \\
\gamma_{x'y'}
\end{bmatrix},
$$

(4.5)

with $E'_{x'}$, $E'_{y'}$ and $G'$ the Young's moduli and shear modulus of the orthotropic layer, respectively, in the $x'$ and $y'$ coordinate system. Denoting the stiffness tensor in equation 4.5 as $D^{*'}$, and extending equation 4.3 to multiple dimensions (i.e., $E$ becomes $D^{*'}$) yields

$$
\sigma' = zD^{*'}\epsilon' + z\eta^{*'}\frac{\partial}{\partial t}\epsilon',
$$
where the Voigt constant $\eta$ has been generalized to a matrix with unknown entries of the same structural form as $D^{st}$, i.e.

$$
\eta^{st'} = \begin{bmatrix}
\eta_{xx'}^{st'} & \eta_{xy'}^{st'} & 0 \\
\nu_{xy}^{st'} \eta_{xx'}^{st'} & \nu_{xy}^{st'} \eta_{yy'}^{st'} & 0 \\
(1 - \nu_{xx}^{st'} \nu_{yy}^{st'}) & (1 - \nu_{xx}^{st'} \nu_{yy}^{st'}) & 0 \\
0 & 0 & \eta_{xy}^{st'}
\end{bmatrix}.
$$

(4.6)

The equations immediately above refer to the $x'$ and $y'$ axes, which are those normal to and aligned with the fibers, respectively. For unidirectional fiber materials, rotation can realign the fibers, and therefore the natural material axes (principal directions), to the reference coordinate system. A transformation matrix rotates the fibers by an angle $\alpha$:

$$
T = \begin{bmatrix}
\cos^2 \alpha & \sin^2 \alpha & \sin \alpha \cos \alpha \\
\sin^2 \alpha & \cos^2 \alpha & -\sin \alpha \cos \alpha \\
-2 \sin \alpha \cos \alpha & 2 \sin \alpha \cos \alpha & \cos^2 \alpha - \sin^2 \alpha
\end{bmatrix}.
$$

This allows one to write

$$
\begin{bmatrix}
\epsilon_{x'} \\
\epsilon_{y'} \\
\gamma_{x'y'}
\end{bmatrix} = T \begin{bmatrix}
\epsilon_x \\
\epsilon_y \\
\gamma_{xy}
\end{bmatrix},
$$

or concisely,

$$
\epsilon' = T \epsilon.
$$

The stress equation also undergoes a similar transformation:

$$
\sigma' = T \sigma.
$$
Combining the above strain and stress equations, the transformed stress-strain matrix equation is written as

\[
\sigma = zT^{-1}D''T \epsilon + zT^{-1}\eta''T \frac{\partial}{\partial t} \epsilon \\
= zD^{*}\epsilon + z\eta^{*} \frac{\partial}{\partial t} \epsilon,
\]

where

\[
D^{*} = T^{-1}D''T \tag{4.7}
\]
\[
\eta^{*} = T^{-1}\eta''T. \tag{4.8}
\]

Combining these with the definition of strain in equation 4.4, and treating each layer as residing in a unique location in \( z \), yields the strain-displacement relations for the \( k \)th layer:

\[
\begin{bmatrix}
\sigma_x \\
\sigma_y \\
\tau_{xy}
\end{bmatrix}_k = zD^{*}_k \begin{bmatrix}
\epsilon_x \\
\epsilon_y \\
\gamma_{xy}
\end{bmatrix}_k + z\eta^{*}_k \frac{\partial}{\partial t} \begin{bmatrix}
\epsilon_x \\
\epsilon_y \\
\gamma_{xy}
\end{bmatrix}_k
\]
\[
= zD^{*}_k \begin{bmatrix}
-\frac{\partial^2 w}{\partial x^2} \\
-\frac{\partial^2 w}{\partial y^2} \\
-2 \frac{\partial^2 w}{\partial x \partial y}
\end{bmatrix} + z\eta^{*}_k \begin{bmatrix}
-\frac{\partial^3 w}{\partial t \partial x^2} \\
-\frac{\partial^3 w}{\partial t \partial y^2} \\
-2 \frac{\partial^3 w}{\partial t \partial x \partial y}
\end{bmatrix}. \tag{4.9}
\]

\( D^{*} \) in equation 4.9 must be evaluated for every layer. Representing the \( k \)th layer by the coordinates in \( z \), i.e. the \( k \)th layer is denoted by \( z \in [t_{k-1} \ t_k] \), integrating across
all layers, and summing the results yields the moment resultants:

\[
\begin{bmatrix}
M_x \\
M_y \\
M_{xy}
\end{bmatrix}_k = - \sum_{k=1}^{n} \int_{-t_{k-1}}^{t_k} z \left[ D^*_k \begin{bmatrix}
\frac{\partial^2 w}{\partial x^2} \\
\frac{\partial^2 w}{\partial y^2} \\
2 \frac{\partial^2 w}{\partial x \partial y}
\end{bmatrix} + \eta^*_k \begin{bmatrix}
\frac{\partial^3 w}{\partial t \partial x^2} \\
\frac{\partial^3 w}{\partial t \partial y^2} \\
2 \frac{\partial^3 w}{\partial t \partial x \partial y}
\end{bmatrix} \right] z^2 dz.
\]

Thus, performing the above integration, the equations of the bending and twisting moments acting on the mid-plane of the plate are

\[
\begin{bmatrix}
M_x \\
M_y \\
M_{xy}
\end{bmatrix} = - \begin{bmatrix}
D_{11} & D_{12} & D_{13} \\
D_{12} & D_{22} & D_{23} \\
D_{13} & D_{23} & D_{33}
\end{bmatrix} \begin{bmatrix}
\frac{\partial^2 w}{\partial x^2} \\
\frac{\partial^2 w}{\partial y^2} \\
2 \frac{\partial^2 w}{\partial x \partial y}
\end{bmatrix} - \begin{bmatrix}
\eta_{11} & \eta_{12} & \eta_{13} \\
\eta_{12} & \eta_{22} & \eta_{23} \\
\eta_{13} & \eta_{23} & \eta_{33}
\end{bmatrix} \begin{bmatrix}
\frac{\partial^3 w}{\partial t \partial x^2} \\
\frac{\partial^3 w}{\partial t \partial y^2} \\
2 \frac{\partial^3 w}{\partial t \partial x \partial y}
\end{bmatrix} - D \begin{bmatrix}
\frac{\partial^2 w}{\partial x^2} \\
\frac{\partial^2 w}{\partial y^2} \\
2 \frac{\partial^2 w}{\partial x \partial y}
\end{bmatrix} - \eta \begin{bmatrix}
\frac{\partial^3 w}{\partial t \partial x^2} \\
\frac{\partial^3 w}{\partial t \partial y^2} \\
2 \frac{\partial^3 w}{\partial t \partial x \partial y}
\end{bmatrix},
\]

(4.10)

where the terms of matrices D, the lamina rigidity matrix, and \( \eta \) are

\[
D_{ij} = \sum_{k=1}^{n} \frac{1}{3} (D^*_{ij})_k (t_{k}^3 - t_{k-1}^3) \quad (i, j = 1, 2, 3)
\]

\[
\eta_{ij} = \sum_{k=1}^{n} \frac{1}{3} (\eta^*_{ij})_k (t_{k}^3 - t_{k-1}^3) \quad (i, j = 1, 2, 3).
\]

(4.11)
Note the above suggests Voigt damping could be approximated conveniently by choosing a scalar constant multiplied by the rigidity matrix $D$, since the structural form is equivalent. Indeed, this is the case with most Rayleigh damping assumptions, i.e., damping is proportional to the addition of the mass and stiffness matrices, post finite element approximation.

4.1.2 Derivation of Dynamic Equations and Boundary Conditions

Once the matrix $D$ in equation 4.11 has been calculated via equations 4.7 - 4.11, the equation of motion immediately follows from substitution of the moment resultants in equation 4.10 into the plate dynamics in equation 4.2. This yields the final form of the plate equation of motion,

$$\rho h \frac{\partial^2 w}{\partial t^2} + \gamma \frac{\partial w}{\partial t} - \frac{\partial}{\partial x} \left( \frac{\partial M_x}{\partial x} + \frac{\partial M_{yx}}{\partial y} \right) - \frac{\partial}{\partial y} \left( \frac{\partial M_y}{\partial y} + \frac{\partial M_{xy}}{\partial x} \right) = -f_n, \quad (4.12)$$

and all that is left is to determine appropriate boundary conditions for this partial differential equation (PDE). Associated with the dynamic equation above, initial conditions describe the initial position and velocity of the plate, and boundary conditions describe whether a boundary is free, clamped, or simply supported. Take the initial position and velocity of the plate to be functions on domain $\Omega$ to be

$$w(t = 0, x, y) = g_1 \quad (4.13)$$

$$\dot{w}(t = 0, x, y) = g_2. \quad (4.14)$$

In this work, free and clamped boundary conditions are used. The easiest way of deriving boundary conditions is to enforce that work done along an edge is identically
zero along the entire edge. Consider a boundary tangent to the x-axis $\Gamma_i$; this could be $\Gamma_1$ or $\Gamma_3$ in the plate derivation above. The work $J$ performed due to small deformations along this edge can, for purely transverse motion, be written as

$$J = \int_{\Gamma} \left( Q_y w - M_{yx} \frac{\partial w}{\partial x} + M_y \frac{\partial w}{\partial y} \right) \partial \Gamma.$$  \hfill (4.15)

Integration by parts yields

$$J = \int_{\Gamma} \left( Q_y w + \frac{\partial M_{yx}}{\partial x} w + M_y \frac{\partial w}{\partial y} \right) \partial \Gamma - M_{yx} w \big|_{\partial \Gamma},$$  \hfill (4.16)

where $\big|_{\partial \Gamma}$ denotes evaluation at the end points of boundary $\Gamma_i$, and $w$ can be taken here to represent a virtual displacement. Grouping terms in $w$ and its derivatives into separate equations and equating the results to zero define the boundary conditions. A table of boundary conditions is provided for boundaries tangent to the x-axis (table 4.1). Analogous boundary conditions exist for boundaries tangent to the y-axis.

<table>
<thead>
<tr>
<th>Boundary Conditions</th>
<th>Clamped</th>
<th>Free</th>
<th>Simply Supported</th>
</tr>
</thead>
<tbody>
<tr>
<td>Essential</td>
<td>$w = \frac{\partial w}{\partial y} = 0$</td>
<td>$-$</td>
<td>$w=0$</td>
</tr>
<tr>
<td>Natural</td>
<td>$M_{yx} \big</td>
<td>_{\partial \Gamma} = 0$</td>
<td>$M_y = 0$</td>
</tr>
<tr>
<td></td>
<td>$Q_y + \frac{\partial M_{yx}}{\partial x} = 0$</td>
<td>$-$</td>
<td>$-$</td>
</tr>
<tr>
<td></td>
<td>$M_{yx} \big</td>
<td>_{\partial \Gamma} = 0$</td>
<td>$-$</td>
</tr>
</tbody>
</table>

Note the moment resultants in the boundary condition definitions contain both damping and elasticity terms. For instance, the condition $Q_y + \frac{\partial M_{yx}}{\partial y} = 0$ could be expanded
as follows. First, expand all relevant terms using equations 4.10 - 4.11:

\[
\frac{\partial M_y}{\partial y} = -\frac{\partial}{\partial y} \left( D_{12} \frac{\partial^2 w}{\partial x^2} + D_{22} \frac{\partial^2 w}{\partial y^2} + D_{23} \frac{\partial^2 w}{\partial x \partial y} \right) - \frac{\partial}{\partial y} \left( \eta_{12} \frac{\partial^3 w}{\partial t \partial x^2} + \eta_{22} \frac{\partial^3 w}{\partial t \partial y^2} + \eta_{23} \frac{\partial^3 w}{\partial t \partial x \partial y} \right). \tag{4.17}
\]

Likewise,

\[
\frac{\partial M_{yx}}{\partial x} = -\frac{\partial}{\partial x} \left( D_{13} \frac{\partial^2 w}{\partial x^2} + D_{23} \frac{\partial^2 w}{\partial y^2} + D_{33} \frac{\partial^2 w}{\partial x \partial y} \right) - \frac{\partial}{\partial x} \left( \eta_{13} \frac{\partial^3 w}{\partial t \partial x^2} + \eta_{23} \frac{\partial^3 w}{\partial t \partial y^2} + \eta_{33} \frac{\partial^3 w}{\partial t \partial x \partial y} \right). \tag{4.18}
\]

Now, because one can take \( M_{xy} = M_{yx} \), equation 4.18 can simply be doubled to form

\[
Q_y + \frac{\partial M_{yx}}{\partial x} = \frac{\partial M_y}{\partial y} + 2 \frac{\partial M_{xy}}{\partial x}.
\]

For an isotropic material, the above simplifies and matrix \( D \) becomes

\[
D = \frac{h^3}{12(1-\nu^2)} \begin{bmatrix} E & \nu E & 0 \\ \nu E & E & 0 \\ 0 & 0 & \frac{E(1-\nu)}{2} \end{bmatrix},
\]

which is the form of the standard isotropic plate rigidity matrix. Several terms now drop from equation 4.17, leaving

\[
\frac{\partial M_y}{\partial y} = -\frac{E h^3}{12(1-\nu^2)} \frac{\partial}{\partial y} \left( \frac{\partial^2 w}{\partial y^2} + \nu \frac{\partial^2 w}{\partial x^2} \right) - \frac{\eta h^3}{12(1-\nu^2)} \frac{\partial}{\partial y} \left( \frac{\partial^3 w}{\partial t \partial y^2} + \nu \frac{\partial^3 w}{\partial t \partial x^2} \right)
\]

\[
\frac{\partial M_{yx}}{\partial x} = -\frac{E h^3}{12(1-\nu^2)} \frac{\partial}{\partial x} \left( 1 - \nu \right) \frac{\partial^2 w}{\partial x \partial y} - \frac{\partial}{\partial x} \left( \frac{\eta}{2(1+\nu)} \frac{\partial^3 w}{\partial t \partial x \partial y} \right).
\]

Thus

\[
Q_y + \frac{\partial M_{yx}}{\partial x} = 0.
\]
becomes

\[
- \frac{E h^3}{12(1 - \nu^2)} \frac{\partial}{\partial y} \left( \frac{\partial^2 w}{\partial y^2} + (2 - \nu) \frac{\partial^2 w}{\partial x^2} \right) \\
- \frac{\eta h^3}{12(1 - \nu^2)} \frac{\partial}{\partial y} \left( \frac{\partial^3 w}{\partial t \partial y^2} + (2 - \nu) \frac{\partial^3 w}{\partial t \partial x^2} \right) = 0,
\]

which is the standard boundary condition for isotropic thin plates that states a free edge must be free of shear force. Likewise, the other boundary conditions simplify for an isotropic material.

### 4.1.3 Piezocomposite Actuators and Sensors

A basic approach for incorporating piezocomposite sensors/actuators into the thin plate model is now developed. While the model is intended to serve as an approximation to piezocomposite materials, more precise models can easily be implemented in place of the following, provided they are linear. It is assumed the piezocomposite actuators simultaneously act as sensors. Combined sensor-actuators is still an emerging field of research and development [150]; the use of piezocomposite actuators as self-sensing has not yet been investigated. It is the author’s hypothesis that such a use is possible. Additionally, it is assumed the sensors used here provide an average, non-directional strain value. This is a crude strain measurement to use, but nonetheless provides adequate feedback.

The main hypothesis in the development of the following model is, although piezocomposite actuators are more complex behaviorally than traditional piezoceramic materials, their behavior can be characterized in a way similar to the approach used for modeling carbon fiber plates in the previous subsection. To be precise, it is assumed piezocomposites can be modeled as orthotropic materials that deform primarily along
one axis and do so only via Poisson’s effects along the other. Modeling the piezocomposites this way allows their placement on a plate geometry to maximize their control authority and effectiveness. Thus, traditional piezoceramic models are utilized, extending the models where necessary to exhibit orthotropic behavior.

Note the top and bottom patches that form pairs can be treated separately if one is interested in modeling both axial and transverse deflections. Here, it is assumed, for simplicity sake, the patches are activated with exactly opposite voltages, providing solely bending moments, thus exhibiting only transverse, or out-of-plane, motion.

**Piezocomposite Sensors and Actuators**

Here, the assumption piezocomposite actuators are self-sensing is made, i.e. returning a total average strain value. If equation 4.4 describes the strain field, then the sensor output can be written simply as

\[
S_i = - \int_{\Omega_i} h \left( \frac{\partial^2 w}{\partial x^2} + \frac{\partial^2 w}{\partial y^2} \right) \partial \Omega_i, \tag{4.19}
\]

where \(\Omega_i\) is the region occupied by the \(i\)th piezocomposite patch. This model, along with the thin plate model, is approximated in the next chapter using a finite element scheme.

For low drive regimes, that is to say, electric potentials resulting in linear behavior, of piezocomposites, it is assumed the linear constitutive equations hold. Those equations are

\[
\sigma = \epsilon \epsilon + \dot{\epsilon} - \frac{d_{33}}{L_{pz}} V \\
\sigma = \epsilon \epsilon + \dot{\epsilon} - \frac{d_{33}}{L_{pz}} V \\
P = Y d_{33} \epsilon + \xi \frac{V}{h} \tag{4.20}
\]
where $Y$ and $c$ denote the effective Young’s modulus and Kelvin-Voigt damping coefficients of the piezocomposite, $L_{pz}$ is the length of the patch, $\sigma$ and $P$ are the stress and polarization fields, and $d_{33}$ is an empirically determined constant relating deformation of the piezocomposite to applied voltage. Note stress depends linearly on voltage.

Consider figure 4.1 again, in which patches are bonded at locations with support given by

$$\chi_{pz_i}(x, y) = \begin{cases} 1, & (x, y) \in \Omega_i \\ 0, & (x, y) \notin \Omega_i \end{cases}$$

with fiber direction aligned with the $x$-axis. This means the dominate deformation direction will be along the $x$-axis.

Patch material contribution modifications to the thin plate model previously derived result in changes to density and moment/force resultants. These contributions [149] are as follows.

A contribution to the density of the plate due to the patch mass forces $\rho$ in equation 4.12 to become a piecewise function of space. Redefining $\rho(x, y)$ as the composite density, with $\rho$ the original density of the plate, results in

$$\rho(x, y) = \rho h + 2 \sum_{i=1}^{N_{pz}} \chi_{pz_i}(x, y) \rho_A h_{pz},$$

where $\rho_A$ is the average density of the piezocomposite material and $h_{pz}$ is the thickness of the patches.

Due to linearity, the moment contributions of the patches to equation 4.12 can be represented as the sum of plate and patch moments. Thus, applying a similar analysis to moment contributions of the patches, and assuming a patch is placed on both the top and bottom of the plate forming a pair of patches, patch material contributions to the plate system can be written in terms of moments that are added to those in
equation 4.10 (taken, with modifications for orthotropy, from [76]):

\[
(M_x)_{pz} = \frac{2Y_{1pz}c_3}{1 - \nu_{12pz}\nu_{21pz}} (\kappa_x + \nu_{21pz}\kappa_y) \sum_{i=1}^{N_{pz}} \chi_{pz_i}(x, y) \tag{4.23}
\]

\[
(M_y)_{pz} = \frac{2Y_{2pz}c_3}{1 - \nu_{12pz}\nu_{21pz}} (\kappa_y + \nu_{12pz}\kappa_x) \sum_{i=1}^{N_{pz}} \chi_{pz_i}(x, y) \tag{4.24}
\]

\[
(M_{xy})_{pz} = 2G_{pz}c_3\kappa_{xy} \sum_{i=1}^{N_{pz}} \chi_{pz_i}(x, y), \tag{4.25}
\]

where \(c_3 = \int_{h/2}^{h/2+h_{pz}} z^2 dz\). The external moment contribution to the plate system due to an active patch is (also from [76])

\[
M_{x_{ext}}^i = \frac{-Y_{1pz}d_{33}c_2}{L_{pz}(1 - \nu_{12pz})} \sum_{i=1}^{N_{pz}} V_i \chi_{pz_i}(x, y)
\]

\[
M_{y_{ext}}^i = \frac{-Y_{2pz}d_{33}c_2}{L_{pz}(1 - \nu_{21pz})} \sum_{i=1}^{N_{pz}} V_i \chi_{pz_i}(x, y), \tag{4.26}
\]

where \(c_2 = \int_{h/2}^{h/2+h_{pz}} z dz\). Note upon activation, the external moments generated in each direction are proportional to the Young’s modulus in each direction of the patch. In reality, such a relationship may not be completely accurate for a strongly orthotropic piezocomposite, but, assuming such a model has been “fit” to laboratory data, such a relationship should yield sufficiently accurate characterization for small deformations of the plate system and low activation regimes of the piezoceramic device. Should the piezocomposite actuator be replaced by traditional piezoceramic material utilizing piezoelectric coefficient \(d_{31}\), one simply changes the above, replacing \(d_{33}\) with \(d_{31}\), and \(L_{pz}\) with \(h_{pz}\).

Equations 4.23 - 4.26 are then added to the original system moment resultant equations to yield the combined patch-plate dynamic model. The moment contri-
butions account for the stiffness and damping contributions of the piezo to the thin plate model. The Voigt constant of the piezo material is taken to be the same as the plate, simplifying the model. The combined model can be written as

\[
\rho h \frac{\partial^2 w}{\partial t^2} + \gamma \frac{\partial w}{\partial t} - \frac{\partial}{\partial x} \left( \frac{\partial M_x}{\partial x} + \frac{\partial M_{yx}}{\partial y} \right) - \frac{\partial}{\partial y} \left( \frac{\partial M_y}{\partial y} + \frac{\partial M_{xy}}{\partial x} \right) = -f_n + M_{ext}, \quad (4.27)
\]

where

\[
M_x = M_{xplate} + M_{xpz} \\
M_y = M_{yplate} + M_{ypz} \\
M_{xy} = M_{xyplate} + M_{xypz} \\
M_{ext} = \frac{\partial^2}{\partial x^2} (M_{x_{ext}}) + \frac{\partial^2}{\partial y^2} (M_{y_{ext}}) \quad (4.28)
\]

The resulting equation, equation 4.27, is discretized in space using a finite element approximation procedure, resulting in a form appropriate for simulation and control design. The membrane model that will be used in chapter 6 is now introduced, including a derivation of stretch sensors.

4.2 Linear Membrane Model

Membrane models can be arrived at via a variety of approaches, including limiting procedures using thin plate equations and direct force balancing using Newton’s laws. The latter is the choice derivation for most applications (including the present) since it describes in detail how the model arises and, in the linear case, simplifies to the classical wave equation. Similar to the plate model derivation in the previous section, consider a membrane infinitesimal, but unlike the plate derivation which began from
an undeformed state, this derivation begins with a membrane in a deformed state. This allows for easy derivation of a nonlinear model that can be linearized, illustrating precisely how and when Poisson’s equation arises in the modeling process.

In 1907, Föpple derived equilibrium equations for a membrane plate, which turned out to be the Von Karman plate equations [151]. Also arriving at the membrane model by utilizing a limiting procedure on plate theory, Hencky [152] investigated an initially planar membrane with circular boundary conditions inflated by a uniform pressure. Cambell [153] researched the response of an inflated membrane with a given initial tension. These results have been demonstrated experimentally many times, including in an article comparing a classically linear membrane model to a nonlinear membrane model [154]. The problem of studying a membrane undergoing deflection due to uniform hydrostatic force is commonly referred to as the Hencky-Campbell problem, due to the history just described.

To be considered a membrane, a structure must satisfy the following conditions [155]:

1. The boundaries are free from transverse shear forces and moments. Loads applied to the boundaries must lie in planes tangent to the middle surface.

2. The normal displacements and rotations at the edges are unconstrained, i.e. these edges can displace freely in the normal direction to the middle surface.

3. A membrane must have a smoothly varying, continuous surface.

4. The components of the surface and edge loads must also be smooth and continuous functions of the coordinates.
5. Membranes do not have any flexural rigidity and therefore cannot resist any bending loads. Membranes can only sustain tensile loads. Their inability to sustain compressive loads leads to the phenomenon known as wrinkling.

Consider the infinitesimal membrane element in figure 4.4 with tension $T_x$ in the $x$-direction and $T_y$ in the $y$-direction. Also consider a face load/pressure $P_z$ acting on the membrane strictly in the normal direction. The membrane section itself rotated by some degree about the centroid of the section. The tension forces acting on the element are not necessarily aligned with the principle coordinate directions due to the rotation of the element. Denote these angles of rotation due to deflection as $\theta$ and $\psi$ as is indicated in figure 4.4. The tension forces can then be represented using trigonometry. Shear forces are neglected in this model, as they contribute very little to the overall membrane dynamics [156]. Using figure 4.4 as a guide, sum the forces

![Figure 4.4: Forces acting on a deformed infinitesimal membrane element](image-url)
in the z-direction:

\[
\sum F_z = \left( T_x + \frac{\partial T_x}{\partial x} \right) \sin \left( \theta + \frac{\partial \theta}{\partial x} dx \right) dy - T_x \sin (\theta) dy + \left( T_y + \frac{\partial T_y}{\partial y} \right) \sin \left( \psi + \frac{\partial \psi}{\partial y} dy \right) dx - T_y \sin (\psi) dx + P_z \frac{dx dy}{\cos (\theta) \cos (\psi)}. \tag{4.29}
\]

Now, expanding the first sine term using an angle addition formula,

\[
\sin \left( \theta + \frac{\partial \theta}{\partial x} dx \right) = \sin (\theta) \cos \left( \frac{\partial \theta}{\partial x} dx \right) + \cos (\theta) \sin \left( \frac{\partial \theta}{\partial x} dx \right),
\]

and making the assumption \( \theta \ll 1 \) and \( \frac{\partial \theta}{\partial x} \ll 1 \), simplify the above using Taylor series, yielding

\[
\sin \left( \theta + \frac{\partial \theta}{\partial x} dx \right) \approx \sin (\theta) + \cos (\theta) \frac{\partial \theta}{\partial x} dx.
\]

Similarly, for the second angle addition term in equation 4.29,

\[
\sin \left( \psi + \frac{\partial \psi}{\partial y} dy \right) \approx \sin (\psi) + \cos (\psi) \frac{\partial \psi}{\partial y} dy.
\]

Substituting these approximate angle addition formulas into equation 4.29, expanding, simplifying, and dropping higher order terms yields

\[
\sum F_z = \frac{\partial T_x}{\partial x} \sin (\theta) dx dy + T_x \frac{\partial \theta}{\partial x} \cos (\theta) dy dx + \frac{\partial T_y}{\partial y} \sin (\psi) dx dy + T_y \frac{\partial \psi}{\partial y} \cos (\psi) dx dy + P_z \frac{dx dy}{\cos (\theta) \cos (\psi)}. \tag{4.30}
\]
Apply Newton’s second law, where the mass of the infinitesimal is proportional to the approximate area, to arrive at

\[
\frac{\rho h}{\cos(\theta)\cos(\psi)} \frac{\partial^2 w}{\partial t^2} = \frac{\partial T_x}{\partial x} \sin(\theta) + T_x \frac{\partial \theta}{\partial x} \cos(\theta) + \frac{\partial T_y}{\partial y} \sin(\psi) + T_y \frac{\partial \psi}{\partial y} \cos(\psi) + \frac{P_z}{\cos(\theta)\cos(\psi)}.
\]  

(4.31)

This equation assumes the membrane thickness does not vary due to deformation, an assumption that may not be satisfactory for very large deflections. If $\theta \ll 1$ and $\phi \ll 1$, which are reasonable assumptions for a membrane wing, equation 4.31 simplifies to

\[
\rho h \frac{\partial^2 w}{\partial t^2} = T_x \frac{\partial \theta}{\partial x} + T_y \frac{\partial \psi}{\partial y} + P_z.
\]

When slope in the x- and y-directions is approximated by $\tan(\theta)$ and $\tan(\psi)$, respectively, one obtains

\[
\rho h \frac{\partial^2 w}{\partial t^2} - T_x \frac{\partial^2 w}{\partial x^2} - T_y \frac{\partial^2 w}{\partial y^2} = P_z.
\]

(4.32)

Statements have not yet been made regarding the tension terms $T_x$ and $T_y$. These terms may be defined as functions of the state $w$, leading to a nonlinear equation that captures the change in tension due to deformation, or they may be taken to be constants arising simply due to pre-strain. Assuming the latter, equation 4.32 becomes the classical Poisson’s equation for a vibrating membrane.

Similar to the plate model in the previous section, variational arguments regarding virtual work can be used to generate boundary conditions. Rather than repeat the mathematics, it is simply stated that the boundary conditions relevant to this dissertation are solely zero Dirichlet, as all membranes simulated and studied in the lab will have fully fixed edges. For Poisson’s equation, a fixed edge simply corresponds
simply to the natural boundary condition (written for general boundary \( \Gamma \))

\[
|w|_\Gamma = 0. \tag{4.33}
\]

There are no restrictions on the derivative of \( w \).

Associated with equation 4.32, initial conditions describe the initial position and velocity of the membrane. Take the initial position and velocity of the membrane to be functions on domain \( \Omega \) as follows

\[
w(t = 0, x, y) = g_1 \tag{4.34}
\]
\[
\dot{w}(t = 0, x, y) = g_2. \tag{4.35}
\]

### 4.2.1 Computation of Tension Parameters and Damping

It is still to be determined how to calculate the tension values, \( T_x \) and \( T_y \). The classical Poisson’s equation calls for internal tensions with units of force per unit length. During tensioning of a membrane, a specific strain field is applied to the membrane that explicitly determines the tension field in \( x \)- and \( y \)-directions. Hooke’s model is presented here, as it is relevant for in-plane prestress of an isotropic material:

\[
\begin{bmatrix}
T_x \\
T_y \\
T_{xy}
\end{bmatrix} = \frac{Eh}{1-\nu^2} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{(1-\nu)}{2} \end{bmatrix} \begin{bmatrix} \epsilon_x \\ \epsilon_y \\ \gamma_{xy} \end{bmatrix}, \tag{4.36}
\]

where \( E \) is the Young’s modulus of the membrane material, \( h \) is the thickness of the membrane, and \( \nu \) is Poisson’s ratio. In general, one applies a specified percent prestrain to a membrane to achieve a tension field determined by equation 4.36. In a laboratory setting, the tension values can be approximated from the full-field
strain to verify a membrane has been pre-tensed to the desired degree. The material parameters of the membrane, present in equation 4.36, were determined through laboratory testing and are summarized in chapter 6 when a specific membrane model is chosen for investigation.

Damping can be included in the same manner as it was for the plate model: Kelvin-Voigt by assuming equation 4.36 satisfies the Kelvin-Voigt hypothesis in equation 4.3. The final membrane equation of motion is stated here without derivation:

\[
\rho_h \frac{\partial^2 w}{\partial t^2} + \gamma \frac{\partial w}{\partial t} - \eta \left( \frac{\partial^3 w}{\partial t \partial x^2} + \frac{\partial^3 w}{\partial t \partial y^2} \right) - T_x \frac{\partial^2 w}{\partial x^2} - T_y \frac{\partial^2 w}{\partial y^2} = P_z. \quad (4.37)
\]

In this context, the terms \( \gamma \) and \( \eta \) are, again, referred to as the viscous and Kelvin-Voigt damping parameters, respectively. Equation 4.37 is an accurate membrane model for small deflections. For a discussion of the accuracy and comparison with nonlinear models, see [154].

### 4.2.2 Generalized Stretch Sensors

To model what an organism might perceive as stretch information being generated by muscle spindles, stretch sensors are developed in this subsection that will provide such general stretch feedback to be utilized in chapter 6. Unlike plate theory, in which a well defined strain model was used to derive the general equations, no such models exist for membrane theory. Generalized concepts such as relative changes in curve length or surface area are necessary to consider in order to arrive at a model for stretch sensors.

Consider a membrane undergoing deflection and a small stretch (or strain) sensor covering a local region. Much like the muscle bands in the bat membrane, such a
sensor will yield a cumulative response due to deformation. Regardless of the model chosen to capture the physics of such a membrane, the deformation itself is what gives rise to the stretch (or change in area) measured by a given sensor, be it a bat wing muscle or strain gauge. For this reason, the author considers estimating what a properly empirically tuned sensor might sense, having been bonded to a membrane that is now undergoing a general deflection. The qualitative sensor model that arises is intimately related to Lagrange strain, which is itself nonlinear. Although applied to a linear membrane model, a compromise in accuracy only results due to the assumed membrane model and not the sensor model itself. Indeed, the model for stretch/strain proposed here can be used in conjunction with any dynamic model for membrane or plate deflections, so long as the sensors provide Lagrange strain as feedback.

To begin, consider an initially flat membrane. If such a membrane undergoes a very small deformation to arrive at a surface described by \( w = f(x, y) \), one can determine the initial and final change in surface area of a small section, whose support is denoted by \( \Omega_s \). Basic calculus provides a useful equation that describes surface area, denoted by \( A' \):

\[
A' = \int_{\Omega_s} \sqrt{1 + \left( \frac{\partial w}{\partial x} \right)^2 + \left( \frac{\partial w}{\partial y} \right)^2} \, d\Omega_s. \tag{4.38}
\]

\( \frac{\partial w}{\partial x} \ll 1 \) and \( \frac{\partial w}{\partial y} \ll 1 \) for very small deflections; therefore one can apply the binomial theorem to equation 4.38 to simplify the square root term, yielding an approximate expression for \( A' \),

\[
A' \approx \int_{\Omega_s} 1 + \frac{1}{2} \left( \frac{\partial w}{\partial x} \right)^2 + \frac{1}{2} \left( \frac{\partial w}{\partial y} \right)^2 \, d\Omega_s, \tag{4.39}
\]
where the higher order terms have been neglected. Notice the first term in the integral in equation 4.39 can be integrated to arrive at

\[ A' \approx A_0 + \int_{\Omega_s} \frac{1}{2} \left( \frac{\partial w}{\partial x} \right)^2 + \frac{1}{2} \left( \frac{\partial w}{\partial y} \right)^2 \, d\Omega_s, \]  

(4.40)

where \( A_0 \) is the initial, unperturbed surface area of the region \( \Omega_s \). At this point, a percent change in area can be defined as follows: subtract the initial area \( A_0 \) to the left hand side of equation 4.40, divide both sides by the initial area \( A_0 \), and denote the percent change in area as \( S \), yielding

\[ \frac{A' - A_0}{A_0} \approx \frac{1}{2A_0} \int_{\Omega_s} \left( \frac{\partial w}{\partial x} \right)^2 + \left( \frac{\partial w}{\partial y} \right)^2 \, d\Omega_s \]

\[ S = \frac{\alpha}{2A_0} \int_{\Omega_s} \left( \frac{\partial w}{\partial x} \right)^2 + \left( \frac{\partial w}{\partial y} \right)^2 \, d\Omega_s \]  

(4.41)

for some sensor constant of proportionality \( \alpha \).

Note \( S \geq 0 \) for all \( w(x, y) \), which agrees with both intuition and, more importantly, the definition of a membrane. Stretch is positive regardless of deformation and orientation. Thus an obvious limitation of this sensor model is its inherent inability to discern between positive and negative deformations. This becomes important when one attempts to estimate a pressure field on a membrane solely via stretch sensor information, as such a measurement may return the correct magnitude, but not the correct sign of pressure if the load is highly spatially varying. In general, because a membrane deforms locally to applied loads, the sign of the pressure is worked out due to the interaction of the other sensors.

The sensor constant of proportionality \( \alpha \) is included to capture any magnitude discrepancies between feedback from a real sensor and the model. The stretch sensor
model in this work is assumed to, at the very least approximately, capture the output from a piezo film or another cumulative stretch/strain sensor. Within the linear range of behavior, it is likely a similar relationship could be determined for specific sensor types.

This sensor model can be approximated by the finite element method (see Appendix B.1). It is nonlinear, however, and therefore necessitates specialized estimation approaches, as discussed in chapter 6.

The formulation above for stretch sensors is equivalent to “large” strain, i.e., Lagrange or Green strain, taken from [157] and given by

\[
\begin{align*}
\epsilon_x &= \frac{\partial u}{\partial x} + \frac{1}{2} \left[ \left( \frac{\partial v}{\partial x} \right)^2 + \left( \frac{\partial w}{\partial x} \right)^2 \right] \\
\epsilon_y &= \frac{\partial v}{\partial x} + \frac{1}{2} \left[ \left( \frac{\partial u}{\partial x} \right)^2 + \left( \frac{\partial w}{\partial y} \right)^2 \right].
\end{align*}
\]

If the system is constrained to deflect only in the z-direction, i.e. \( u = v = \epsilon_z = 0 \), the formulation becomes

\[
\begin{align*}
\epsilon_x &= \frac{1}{2} \left( \frac{\partial w}{\partial x} \right)^2 \\
\epsilon_y &= \frac{1}{2} \left( \frac{\partial w}{\partial y} \right)^2,
\end{align*}
\]

which agrees with the individual terms of the integrand in equation 4.41. Thus, when seeking the total average Lagrange strain over a rectangular region, the change in area of that region due to deformation is being sought.

This concludes the model derivation chapter. In the next chapter, chapter 5, an investigation of wing morphing using piezocomposite actuators and linear quadratic control is performed utilizing the thin plate model derived in this chapter. Following
that investigation, chapter 6 utilizes the membrane model to propose and investigate a novel load identification algorithm.
Wing morphing using a LQG approach is now investigated. A desired morphed position of a wing is described using a 3D surface that can be tailored to describe any surface. Three basic, canonical shapes were directly inspired by the morphing wing review and biology and include wing twist, camber, and bending deformations. In general, arbitrary shapes chosen by a naive user will not satisfy the plate physics, i.e., the plate may not be able to take precisely the shape chosen.

With the desired shapes chosen, the patch-plate system is approximated via a finite element approach. A generalized quadratic tracking control problem is then introduced, discussed, and solved for the approximated system, which yields a feed-forward signal that prescribes the correct voltage input to each actuator. The numerical results of simulation for the three canonical shapes are examined. Furthermore, as with all finite element approximations of infinite-dimensional systems, it is important to demonstrate convergence as the discretization is refined. For this reason, numerical convergence is investigated, and both graphical and eigenvalue evidence is presented that, indeed, suggests the approximation is converging to approximate the infinite-dimensional, continuous system.

5.1 Biologically Inspired Morphing

Wing morphing can be classified into three general types: planform alteration (changes in span, chord, and sweep), out-of-plane transformation (wing twist, dihedral, and bending modifications), and airfoil adjustment (wing camber and thickness modifications) [44]. In this work, only out-of-plane deflections are tracked utilizing the thin plate model developed in chapter 4. Again, the three general morphing shapes chosen
are wing twist, camber, and bending. They are ubiquitous in nature and in engineered models, each providing a useful aerodynamic effect, and, together, forming the basis for much of the aerodynamic prowess of flying animals.

The first type of morphing investigated is wing twist. A wing capable of twisting via smart materials would be beneficial to flight. It is well known that modifying the twist of one or both wings of an aircraft generates roll moment; when one wing of a craft is twisted while the other is twisted in the opposite direction (or not at all), a net roll moment results which causes the craft to roll in a direction directly related to the twisting wings; and causes a change in lift resultant on that wing. Define a positive wing twist as resulting from an increase in AOA. A positive twist then results in a roll in the direction of positive AOA. A general twisted wing position is depicted in figure 5.1.

![Figure 5.1: Wing twist](image)

The second type of wing morphing investigated is camber modification. Modifying wing camber results in a change in the lift coefficient of a wing. For a thin wing, camber can be described as the degree of chordwise bending, that is, the greater the average curvature in the chordwise direction, the greater the camber. By definition, a perfectly flat plate has zero camber while a plate bent as if to become a cylindrical
section has positive camber. Actively changing camber allows a craft to operate efficiently at different velocities. A large camber benefits a slow moving, loitering craft, while a small camber benefits the same craft traveling at high velocities. Thus, a change in camber will likely be necessary if the aircraft transitions between loitering, observational flight, and an attack or kill mode. Figure 5.2 illustrates a wing undergoing a change in camber. The original position is planar, but the morphed surface assumes a position that results in the mid-chord section being elevated above the leading and trailing edges. This position is referred to as the point of maximum camber.

![Figure 5.2: Wing camber](image)

The third and final type of wing morphing investigated is wing bending, or dihedral modification. Modification of wing dihedral changes roll moment stability which depends on the direction of dihedral [158]. Positive bending increases stability, while negative decreases stability but improves roll rate. Figure 5.3 illustrates a bending wing whose original surface is planar and resulting surface bent downwards. Such a downward position is observed in fast, maneuverable biological flyers, such as the Violet-green Swallow in figure 2.1.
Describing these shapes mathematically is simple. Interpolate between the important features of each surface using polynomials as functions of x and y. For instance, the twisted surface is formulated by choosing a linear plate-tip edge at some angle to the horizontal and then interpolating back to the origin in a linear fashion. Functions for camber and bending are likewise created; these three functions generated figures 5.1 - 5.3 above. Table 5.1 provides function definitions for twist, camber, and bending morphing types. These equations are utilized in the next section to develop feedforward control inputs to the patches within a feedback control design loop.

Table 5.1: Normalized morphing shape functions

<table>
<thead>
<tr>
<th>Type</th>
<th>Equation</th>
<th>a</th>
</tr>
</thead>
<tbody>
<tr>
<td>Twist</td>
<td>$z_d(x, y) = \frac{a}{L}(1 - \frac{2}{W}x)(L - y)$</td>
<td>.01</td>
</tr>
<tr>
<td>Camber</td>
<td>$z_d(x, y) = \frac{-2a}{W^2L}x(x - W)(L - y)$</td>
<td>.01</td>
</tr>
<tr>
<td>Bending</td>
<td>$z_d(x, y) = -\frac{a}{L^2}(y - L)^2$</td>
<td>.1</td>
</tr>
</tbody>
</table>
5.2 System Geometry and Approximation Scheme

A simple, not completely ad hoc system geometry was chosen, sufficient to generate useful results for utilizing optimal control to morph a wing (see figure 5.4). The plate is cantilevered with a clamped boundary condition along the edge $y = L$, or boundary $\Gamma_3$, and thus free along all other boundaries, as illustrated in figure 4.1 and readily apparent in figures 5.1 - 5.3. The plate has length $L = 0.6$ m and width $W = 0.15$ m, yielding an aspect ratio of $L/W = 5$. The $\Gamma_3$ boundary, of course, approximates attachment to an aircraft fuselage or body of a flyer.

![Figure 5.4: Morphing patch-plate system geometry](image)

The carbon fiber battens, in black in figure 5.4, add support to an otherwise very flexible system that would not withstand aerodynamic load. The carbon fiber is a four-layer orthotropic material with a Young’s modulus of 20 GPa, and thus greater stiffness, along the y-axis, or spanwise direction, and 1.5 GPa along the x-axis, or
chordwise direction. These material parameters and fiber directionality are used for computation of the plate rigidities in equations 4.4 - 4.11. The interior plate material, in dark gray, is rather flexible, with a Young’s modulus of 1 GPa, corresponding to a rubber-like material under sufficient pretension so that its behavior can be described by linear plate theory. Nine piezocomposite patches are placed on the top and bottom of the wing structure, resulting in 18 total pairs of patches. For the purposes of this investigation, all of the patch pairs are taken to be activated out of phase of one another, resulting in pure bending moment along each patch’s location [76]. The self-sensing actuators, in light gray, are positioned along each major batten section via geometric constraints alone, only supported by carbon fiber battens. Relevant material parameters are provided in table 5.2.

Table 5.2: Plate system material parameters

<table>
<thead>
<tr>
<th></th>
<th>$\rho$ ($\frac{kg}{m^2}$)</th>
<th>$E_1$ (GPa)</th>
<th>$E_2$ (GPa)</th>
<th>$G$ (GPa)</th>
<th>$\nu_1$</th>
<th>$\nu_2$</th>
<th>$d_{33}$ (m/V)</th>
<th>h (mm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Plate</td>
<td>1000</td>
<td>10</td>
<td>10</td>
<td>7.5</td>
<td>.33</td>
<td>.33</td>
<td>n/a</td>
<td>.5</td>
</tr>
<tr>
<td>Battens</td>
<td>1500</td>
<td>4.86</td>
<td>3.79</td>
<td>3</td>
<td>.27</td>
<td>.0345</td>
<td>n/a</td>
<td>2</td>
</tr>
<tr>
<td>Piezos</td>
<td>1780</td>
<td>15</td>
<td>45</td>
<td>3</td>
<td>.3</td>
<td>.1</td>
<td>$500 \times 10^{-12}$</td>
<td>.5</td>
</tr>
</tbody>
</table>

Relevant damping parameters used in the numerical studies are provided in table 5.3. The values yield qualitatively accurate results and are of verified and appropriate magnitude for materials such as carbon fiber.

Table 5.3: Plate system Kelvin-Voigt and viscous damping parameters

<table>
<thead>
<tr>
<th></th>
<th>$\eta_1$ (Pa $\cdot$ s)</th>
<th>$\eta_2$ (Pa $\cdot$ s)</th>
<th>$\gamma$ (Pa $\cdot$ s/m)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Plate</td>
<td>0.0001</td>
<td>0.0001</td>
<td>5</td>
</tr>
<tr>
<td>Battens</td>
<td>0.0001</td>
<td>0.0001</td>
<td>5</td>
</tr>
<tr>
<td>Piezos</td>
<td>0.00005</td>
<td>0.00005</td>
<td>5</td>
</tr>
</tbody>
</table>

All of the material inhomogeneities and discontinuities are aligned with the nodes
of the finite element discretization, discussed next. Such alignment demonstrates, in practice, expedited convergence and maintains accuracy.

The plate model system equation, equation 4.27, combined with the distributed, external moments generated by the patches, equation 4.28, and boundary conditions is a PDE. The objective of an approximation scheme is to approximate the model and control solution such that they converge to the true solution of the infinite-dimensional problem.

In this work, a Galerkin approximation scheme approximates the patch-plate model using spline expansions. From appendix A, the transverse displacements are quantified by the weak model formulation

$$\int_{\Omega} \left( \rho h \frac{\partial^2 w}{\partial t^2} \Phi + \gamma \frac{\partial w}{\partial t} \Phi + M_x \frac{\partial^2 \Phi}{\partial x^2} + M_y \frac{\partial^2 \Phi}{\partial y^2} + 2M_{xy} \frac{\partial^2 \Phi}{\partial x \partial y} + f_n(t) \Phi \right) \partial \Omega = 0, \quad (5.1)$$

for \( \Phi \) in the space of test functions

$$V = \{ \Phi \in H^2(\Omega) \mid \Phi(x, L) = \Phi_y(x, L) = 0 \text{ for } 0 \leq x \leq W \}, \quad (5.2)$$

where \( L \) and \( W \) are the length and width of the plate, respectively. The philosophy behind approximating the dynamics of equation 5.1 is as follows: The relation is projected onto a spline-based, dimensionally finite subspace \( V^N \subset V \) to obtain a system discrete in space, continuous in time, and appropriate for finite-dimensional control design. In this work, the spline functions are taken to be bi-cubic B-splines. For a definition of these functions, see [76].

To begin the finite element approximation, consider a partition of \( \Omega \) where \( x_m = m\Delta x \ (\Delta x > 0) \), \( y_n = n\Delta y \ (\Delta y > 0) \), \( \Delta x = W/N_x \), \( \Delta y = L/N_y \), \( m = 0, ..., N_x \), and \( n = 0, ..., N_y \). To enforce the clamped boundary condition along boundary \( \Gamma_3 \),
a modified cubic B-spline basis is used. The cubic B-splines do not naturally satisfy
zero Dirichlet or zero Neumann boundary conditions. However, a linear combination
of the spline functions near the boundary ensures the boundary conditions can be
satisfied. Details of this can be found in [76]. The product space basis elements, \( \Phi_k \),
are taken to be
\[
\Phi_k(x, y) = \phi_m(x)\phi_n(y),
\]
and the approximating subspace
\[
V^N = \text{span}\{\Phi_k\}_{k=1}^{N_w},
\]
where \( N_w = (N_x + 1)(N_y + 1) \). Thus, approximate displacements are represented by
\[
w^N(t, x, y) = \sum_{k=1}^{N_w} w_k(t) \Phi_k(x, y).
\]

It should be noted here that equation 5.1 is composed of terms that contain spacial
discontinuities due to piezo patch contributions and plate material inhomogeneities.
Aligning both actuators and geometric discontinuities with nodes, and treating all
actuator, plate, and carbon fiber batten system components as plate elements, the
contributions of these materials in terms of stiffness and mass can be accounted for
easily in the finite element assembly process itself.

To form the Galerkin finite element approximation of this system, substitute \( w^N \)
for \( w \) in equation 5.1, restricting the variational formulation 5.1 to the subspace \( V^N \),
where derivatives of \( w^N \) are computed from direct differentiation of equation 5.5. If
the moment terms in equation 5.1 are expanded via equation 4.10, the above approx-
imation for \( w \) and its derivatives can be substituted directly into the full moment
term expansion. Substituting these terms into equation 5.1, simplifying, and writing
in matrix vector form yields

\[ M \ddot{z} + K_d \dot{z} + Kz = f + bu \quad (5.6) \]

\[ y = \hat{C}z, \quad (5.7) \]

where dots indicate differentiation in time; and the term \( bu \) replaces the external moments, now representing the control inputs; \( C \) is generated from approximation of equation 4.19; and \( z = [w_1 \ldots w_N]^T \). Matrices \( K \) and \( K_d \) in equation 5.6 are defined as

\[ K = K_1 + K_2 + K_3 \quad (5.8) \]

\[ K_d = K_{d1} + K_{d2} + K_{d3} + K_{d4}; \quad (5.9) \]

where the individual finite element approximation matrices are defined as

\[ [M]_{i,j=1}^N = \int_{\Omega} \rho h \Phi^i \Phi^j d\Omega \]

\[ [K_1]_{i,j=1}^N = \int_{\Omega} \left( D_{11} \Phi^i_{xx} + D_{12} \Phi^i_{yy} + 2D_{13} \Phi^i_{xy} \right) \Phi^j_{xx} d\Omega \]

\[ [K_2]_{i,j=1}^N = \int_{\Omega} \left( D_{13} \Phi^i_{xx} + D_{23} \Phi^i_{yy} + 2D_{33} \Phi^i_{xy} \right) \Phi^j_{xy} d\Omega \]

\[ [K_3]_{i,j=1}^N = \int_{\Omega} \left( D_{12} \Phi^i_{xx} + D_{22} \Phi^i_{yy} + 2D_{33} \Phi^i_{xy} \right) \Phi^j_{yy} d\Omega \]

\[ [K_{d1}]_{i,j=1}^N = \int_{\Omega} \left( \eta_{11} \Phi^i_{xx} + \eta_{12} \Phi^i_{yy} + 2\eta_{13} \Phi^i_{xy} \right) \Phi^j_{xx} d\Omega \]
\[ [K_{d2}]_{i,j=1}^N = \int_\Omega (\eta_{13} \Phi_{xx}^i + \eta_{23} \Phi_{yy}^i + 2\eta_{33} \Phi_{xy}^i) \Phi_{yy}^j \, d\Omega \]
\[ [K_{d3}]_{i,j=1}^N = \int_\Omega (\eta_{12} \Phi_{xx}^i + \eta_{22} \Phi_{yy}^i + 2\eta_{33} \Phi_{xy}^i) \Phi_{yy}^j \, d\Omega \]
\[ [K_{d4}]_{i,j=1}^N = \int_\Omega \gamma \Phi^i \Phi^j \, d\Omega \]
\[ [b]_{i,j=1}^N = \int_\Omega \sum_{i=1}^{N_{pz}} [M_{x_{ext}}^i \Phi_{xx}^i + M_{y_{ext}}^i \Phi_{yy}^i] \, d\Omega \]
\[ [f]_{j=1}^N = \int_\Omega f_i \Phi^j \, d\Omega \]
\[ \left[ \hat{\mathcal{C}} \right]_{i,j=1}^N = \int_\Omega \sum_{i=1}^{N_{pz}} \chi_{p_{zi}} (\Phi_{xx}^j + \Phi_{yy}^j) \, d\Omega, \]

where superscripts \( i,j \) refer to matrix indices, \( M_{x_{ext}}^i \) and \( M_{y_{ext}}^i \) are defined in equation 4.26, and the material parameter matrices are populated using the procedure of chapter 4 with the parameters of tables 5.2 and 5.3. This is the finite element matrix approximation to the plate system PDE in equation 4.12. Note while summations describe multiple patch inputs and sensor outputs, all terms actually depend on patch geometry, locations, and material parameters. The relevant patch contributions to the above matrices are computed during finite element matrix assembly.

The second order form of the approximate system, equation 5.6, can be placed in first order form as:

\[
\begin{bmatrix}
K & 0 \\
0 & M
\end{bmatrix}
\begin{bmatrix}
\dot{z} \\
\ddot{z}
\end{bmatrix} =
\begin{bmatrix}
0 & K \\
-K & -K_d
\end{bmatrix}
\begin{bmatrix}
\dot{z} \\
\ddot{z}
\end{bmatrix} +
\begin{bmatrix}
0 \\
0
\end{bmatrix} u +
\begin{bmatrix}
0 \\
f
\end{bmatrix}
\]

(5.10)

\[
y = \left[ \hat{\mathcal{C}} \right] \begin{bmatrix}
\dot{z} \\
\ddot{z}
\end{bmatrix}.
\]

(5.11)
For simulation and control purposes, write this system in first order state space form (assuming the mass matrix $M$ is nonsingular) by defining $x = [z \; \dot{z}]^T$, then

$$
\begin{align*}
    \dot{x} &= Ax + Bu + F \\
    y &= Cx,
\end{align*}
$$

where

$$
A = \begin{bmatrix}
    0 & I \\
    -M^{-1}K & -M^{-1}K_d
\end{bmatrix} \tag{5.13}
$$

$$
B = \begin{bmatrix}
    0 \\
    M^{-1}b
\end{bmatrix} \quad
C = \begin{bmatrix}
    \dot{C} & 0
\end{bmatrix} \quad
F = \begin{bmatrix}
    0 \\
    M^{-1}\hat{F}
\end{bmatrix}, \tag{5.14}
$$

and initial conditions for 5.12 are

$$
x_0 = \begin{bmatrix}
    z_0 \\
    \dot{z}_0
\end{bmatrix} = \begin{bmatrix}
    \mathcal{P}g_1 \\
    \mathcal{P}g_2
\end{bmatrix}, \tag{5.15}
$$

where $g_1$ and $g_2$ are defined in equations 4.13 - 4.14 and describe the initial position and velocity of the plate, respectively. Initial conditions for the original PDE are projected into the proper subspace, $V^N$ (equation 5.4), using a projection operator, $P$, which is defined as

$$
\begin{bmatrix}
    \mathcal{P}g_1 \\
    \mathcal{P}g_2
\end{bmatrix} = \begin{bmatrix}
    S^{-1} \int g_1 \Phi \, d\Omega \\
    S^{-1} \int g_2 \Phi \, d\Omega
\end{bmatrix}, \tag{5.16}
$$

where

$$
[S]_{i,j=1}^N = \int_{\Omega} \Phi_j \Phi_i \, d\Omega. \tag{5.17}
$$
5.3 General Control Solution

In this section, a linear quadratic control problem is solved that provides a systematic means of morphing a wing into desired shapes. Specifically, LQG – again, an acronym for linear quadratic Gaussian – control is used. This is a specific type of control in which a standard linear quadratic regulator feedback law is combined with an observer, or state estimator, to provide state estimates for feedback. LQG is so named because it is assumed that Gaussian white noise corrupts the sensor measurements, thus a stochastic estimation scheme is needed. In this case, a linear Kalman-filter-based observer (not to be confused with the Kalman approach used in the next chapter) is used to estimate the full system state from measured strain provided by the self-sensing actuators. First, the linear quadratic regulator problem solution is presented, followed by that of the observer system. A full discussion of this control methodology can be found in Dorato [159].

Begin with the model in equations 5.12 - 5.14 that describes the time evolution of the finite element approximation of the thin plate model of chapter 4,

\[ \dot{x} = Ax + Bu + F, x(0) = x_0. \]

Define the tracking error, describing the differences between the actual and desired state, as \( \xi \). Making the substitution \( \xi = x - x_d \), where \( x_d \) is the desired function to track,

\[ \dot{\xi} = A\xi + Bu + F + (Ax_d - \dot{x}_d) \]

\[ \xi(0) = x(0) - x_d(0). \]
$x_d$ has itself been discretized and projected into the same finite element space as state variable $z$. For instance, if the desired position is wing twist, then $x_d$ is formed by the interpolation of the functional form of the twisted wing into a column vector of basis function coefficients resulting from projection of the desired function into space $V^N$ of equation 5.4.

Hypothetically, if $x_d$ is assumed to be a function of time, the forthcoming discussion still holds true, as it can accommodate periodic or time-varying desired position problems. That is not the case in the present work, however, as the system is merely being driven to desired state positions.

Since the matrix $A$ and functions $x_d$ and $\dot{x}_d$ in the equations above are known, the problem now becomes one of driving the tracking error $\xi$ to zero. The problem statement is thus summarized as: given the known disturbance $d(t)$, find the control $u^*(t)$ such that $u^* = \min_u J(u)$, where

$$J = \int_0^{t_f} \left( \xi^T Q \xi + u^T R u \right) dt,$$

subject to the constraints

$$\dot{\xi} = A\xi + Bu + F + d$$
$$d = Ax_d - \dot{x}_d$$
$$\xi(0) = x(0) - x_d(0).$$

The control cost function, equation 5.18, can be minimized using techniques from the calculus of variations. Necessary conditions for a minimizer in the form of differential equations arise. For details of this approach, see Bryson and Ho [160]. The necessary conditions for this problem yield a differential Riccati equation and an equation
describing an optimal feedforward signal for the tracking component of the derived control law. The differential matrix Ricatti equation and matrix equation for the feedforward signal \( b \) must be solved to obtain the optimal control law \( u(t) \). These equations can be written in general form as

\[
\dot{\Pi} = \Pi A + A^T \Pi - \Pi B R^{-1} B^T \Pi + Q \tag{5.19}
\]

\[
\dot{b} = -(A - BR^{-1} B^T \Pi)^{-T} b - \Pi d. \tag{5.20}
\]

The solution of equation 5.19, \( \Pi(t) \), provides a time-varying optimal control law. This can be simplified for certain systems, such as the one at hand, by finding a steady state solution for both equations 5.19 and 5.20. Setting \( \dot{\Pi} = 0 \) and \( \dot{b} = 0 \) yields the steady state regulator problem, in which the optimization time interval is infinite, as well as the algebraic Ricatti equation,

\[
0 = \bar{\Pi} A + A^T \bar{\Pi} - \bar{\Pi} B R^{-1} B^T \bar{\Pi} + Q \tag{5.21}
\]

\[
\bar{b} = -(A - BR^{-1} B^T \bar{\Pi})^{-T} \bar{\Pi} d. \tag{5.22}
\]

where \( \bar{\Pi} \) and \( \bar{b} \) denote the steady state solutions for \( \Pi(t) \) and \( b(t) \), respectively. Thus the control problem for the finite element system is written in terms of error state vector \( \xi \) as

\[
\dot{\xi} = A \xi + B u + F + d
\]

\[
d = A x_d - \dot{x}_d
\]

\[
u = -R^{-1} B^T \bar{\Pi} \xi - R^{-1} B^T \bar{b}.
\]
Writing equation 5.23 in a form more conducive to actual implementation,

$$\dot{x} = Ax + Bu(t) + F$$

$$u(t) = -R^{-1}B^T\Pi(x - x_d) - R^{-1}B^T\bar{b}.$$  \hspace{1cm} (5.23)

Note the solution $\bar{b}$ acts as a feedforward signal to the actuators, prescribing the input required to morph the system into the desired position, while the difference $x - x_d$ is the classical regulator about the morphed position. Here, $R^{-1}B^T\Pi$ is the optimal feedback control gain matrix.

The choice of state-weighting matrix $Q$ and control input cost matrix $R$ in equation 5.18 are problem specific. A $Q$ matrix representing the energy of the system is often utilized to minimize energy [149]. In this work, rather than using energy to generate a cost function, $Q$ is formulated such that it weights the position error in the control cost function, equation 5.18. That is, $Q$ is chosen to be

$$Q = \begin{bmatrix}
\alpha \int_{\Omega} \phi_i \phi_j \, d\Omega & 0 \\
0 & \beta \int_{\Omega} \phi_i \phi_j \, d\Omega
\end{bmatrix},$$

where $\phi_{i,j}$ are the finite element shape functions, as in equations 5.2 - 5.5, and 0 refers to a properly sized zero submatrix. $Q$ is, therefore, of size $N \times N$ for a state vector $x \in \mathbb{R}^N$. In formulating the problem this way, it is also possible to include a weighting function for particular elements or spatial regions that can be generated during the finite element assembly procedure. The simplest way is equally weighting each position state, but allowing for greater relative weighting of position than velocity and/or control in the cost function equation 5.18. In this work, it is taken that $\alpha = \beta = 1$, and, for a control input $R$ matrix, the choice was simply made by trial and error until a satisfactory voltage input was found. In practice, such tuning will be
nearly identical and just as necessary in order to safeguard against destroying piezo
devices and generate a truly optimal control solution for the problem at hand.

Alongside the control problem solved above, the problem of estimating the total
system state from noisy measurements is solved using the standard linear Kalman
filter. Assuming the process and measurement signals are corrupted by uncorrelated,
independent, and normally distributed noise with zero mean $Q_c$ and standard deviation $R_c$, the Kalman filter equations are written as

\[
\dot{\hat{x}} = A\hat{x} + Bu + K(y - C\hat{x}) \tag{5.24}
\]
\[
\dot{P} = AP + PA^T - PC^T R_c^{-1} CP + Q_c, \tag{5.25}
\]

where $\hat{x}(0) = E[x(0)]$ and $P(0) = E[(x(0) - \hat{x}(0))(z(0) - \hat{x}(0))^T]$ are the initial
conditions for the estimated state vector $\hat{x}$ and initial error covariance $P$, describing
the knowledge of the initial conditions of the system. Infinite confidence is assumed
in this initial condition, resulting in $\hat{x}(0) = x_0$ and $P(0) = 0$. The filter Riccati
equation in $P$ above can be simplified, assuming there is a steady state solution for
$P, \bar{P}$, resulting in the algebraic filter Riccati equation,

\[
0 = A\bar{P} + \bar{P}A^T - \bar{P}C^T R_c^{-1} C\bar{P} + Q_c. \tag{5.26}
\]

The equations for $\bar{P}$ and $\bar{P}$, 5.21 and 5.26, can be solved for using MATLAB
command LQR, which solves the steady state Riccati equation. MATLAB was used
to solve all Riccati equations in this work.
Combining the filter in equation 5.24 and original controlled system in equation 5.23 results in a form conducive to simulation in MATLAB:

\[
\begin{align*}
\dot{x} &= Ax + Bu + F \\
y &= Cx \\
\dot{\hat{x}} &= A\hat{x} + Bu + \bar{P}C^T R_c^{-1}(y - C\hat{x}) \\
u &= -R^{-1}B^T\bar{\Pi}(\hat{x} - x_d) - R^{-1}B^T\bar{b}.
\end{align*}
\] (5.27)

\(\bar{P}C^T R_c^{-1}\) is the Kalman gain, or observer gain, matrix. Using the system in equation 5.27, now systematically solve for the three canonical wing morphing types: wing twist, camber, and bending.

The above solution is known as the LQG controller, due to the assumptions of the noise present in the system. The resulting vector-valued, closed-loop system can then be solved by standard differential equation solvers. Throughout the remainder of this chapter, the above system is solved in this way, the results of which are then plotted by multiplying by the finite element basis functions accordingly. The MATLAB solver ODE23t was used to solve the system and, in all cases, default error tolerances were used.

5.4 Morphing Control Solution: Numerical Results

In this section, numerical convergence of system modes and eigenvalues for the approximate finite element model is addressed, followed by that of control gains, and, finally, several investigations in which the control solution of the previous section morphs a wing into desired positions. In all simulations, zero initial conditions for
both position and velocity are used so that the problem is purely one of driving the system to a desired position.

5.4.1 Numerical Convergence Investigation

Fundamentally, there are two aspects to the current finite element approximation and control problem. On one hand, the finite element approximation is just that, an approximation of the plate system containing material inhomogeneities. On the other hand, what has been developed is a control algorithm based on the discretized system, not the infinite-dimensional system derived in chapter 4. Thus, in this situation, one needs to demonstrate, or at least provide evidence, that both the simulation and control algorithm are converging to something meaningful and useful for real implementation.

Three mesh refinements qualitatively determine numerical convergence. The meshes are determined by the plate geometry, with the constraint that only square elements be used at each iteration. Each mesh was refined by a factor of two. The first mesh computed, figure 5.5, is $7 \times 31$ nodes, which corresponds to 180 elements, the second, figure 5.6, $13 \times 61$ nodes or 720 elements, and the third, figure 5.7, $25 \times 121$ nodes or 2880 elements.

Convergence of System Modes and Eigenvalues

Convergence of the approximation scheme is evidenced by convergence of the system modes and eigenvalues, or modal frequencies/shapes for repeated mesh refinements. Illustrated in figures 5.8 - 5.11 are the first four modes ordered by frequency. Note that the mode shapes are similar to those of an isotropic plate model, but differ due to stiff carbon fiber battens. The location of greatest bending in mode two
Figure 5.5: 180-element mesh for patch-plate system

Figure 5.6: 720-element mesh for patch-plate system

Figure 5.7: 2880-element mesh for patch-plate system

coincides with a carbon fiber batten and gives rise to the effect observed in figure 5.10. Close examination reveals the plate material deforms outward between battens. This deformation has become part of the modal characteristics of the plate structure, an interesting but perhaps intuitive result.

The mode shapes themselves change imperceptibly for mesh refinements, so table 5.4 of modal frequencies is included to illustrate this. There are no exact solutions available for a plate geometry this complex, therefore the decreasing, relative change between frequencies offers evidence of convergence.

Since the modal frequencies offer no evidence of damping, the first ten eigenvalues for each mesh refinement are provided in table 5.5, in which significant digit convergence in the imaginary component of the first four eigenvalues for two mesh
Figure 5.8: First bending mode of plate system

Figure 5.9: First torsional mode of plate system

Figure 5.10: Second bending mode of plate system

Figure 5.11: Second torsional mode of plate system

Table 5.4: Plate system: first four undamped modal frequencies for mesh refinements $7 \times 31$, $13 \times 61$, and $25 \times 121$

<table>
<thead>
<tr>
<th></th>
<th>$7 \times 31$</th>
<th>$13 \times 61$</th>
<th>$25 \times 121$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1st</td>
<td>10.116 Hz</td>
<td>9.777 Hz</td>
<td>9.692 Hz</td>
</tr>
<tr>
<td>2nd</td>
<td>40.902 Hz</td>
<td>40.329 Hz</td>
<td>40.099 Hz</td>
</tr>
<tr>
<td>3rd</td>
<td>50.719 Hz</td>
<td>49.940 Hz</td>
<td>49.638 Hz</td>
</tr>
<tr>
<td>4th</td>
<td>121.488 Hz</td>
<td>119.560 Hz</td>
<td>118.709 Hz</td>
</tr>
</tbody>
</table>
refinements is implied albeit slow. Convergence for systems with material inhomogeneities and piezo materials is documented in the literature as slowly occurring.

Table 5.5: Plate system: first ten eigenvalues for mesh refinements 7 × 31, 13 × 61, and 25 × 121

<table>
<thead>
<tr>
<th></th>
<th>7 × 31</th>
<th>13 × 61</th>
<th>25 × 121</th>
</tr>
</thead>
<tbody>
<tr>
<td>-0.54380</td>
<td>-0.54393 + 61.432i</td>
<td>-0.54395 + 60.899i</td>
<td></td>
</tr>
<tr>
<td>-0.39818</td>
<td>-0.39834 + 253.397i</td>
<td>-0.39841 + 251.949i</td>
<td></td>
</tr>
<tr>
<td>-0.51519</td>
<td>-0.51520 + 313.786i</td>
<td>-0.51521 + 311.887i</td>
<td></td>
</tr>
<tr>
<td>-0.38619</td>
<td>-0.38664 + 751.222i</td>
<td>-0.38690 + 745.874i</td>
<td></td>
</tr>
<tr>
<td>-0.54356</td>
<td>-0.54094 + 839.704i</td>
<td>-0.54165 + 831.026i</td>
<td></td>
</tr>
<tr>
<td>-0.38992</td>
<td>-0.39162 + 1408.335i</td>
<td>-0.39218 + 1393.131i</td>
<td></td>
</tr>
<tr>
<td>-0.67503</td>
<td>-0.71330 + 1604.847i</td>
<td>-0.73398 + 1585.292i</td>
<td></td>
</tr>
<tr>
<td>-0.46126</td>
<td>-0.27357 + 2444.335i</td>
<td>-0.29668 + 2356.906i</td>
<td></td>
</tr>
<tr>
<td>-0.20102</td>
<td>-0.46325 + 2469.112i</td>
<td>-0.46271 + 2443.910i</td>
<td></td>
</tr>
<tr>
<td>-0.85973</td>
<td>-1.01768 + 2606.222i</td>
<td>-0.08575 + 2532.346i</td>
<td></td>
</tr>
</tbody>
</table>

Convergence of Functional Control Gains

For demonstrating convergence of the control algorithm, it is useful to produce graphical representations of the control functional gains, discussed at length in Gibson and Adamian [161] with applications of the gains discussed in [162–165].

For the purposes here, it suffices to argue simply that the feedback control law has a feedback part \( R^{-1}B^{T}\Pi(\dot{x} - x_d) \) and feedforward part \( R^{-1}B^{T}\bar{b} \) (equation 5.27). For many structural control problems like this one, that are modeled by PDEs, the
feedback part, before any finite element approximations, can be written as

\[ u(t) = -K_c \hat{x}(t, s) = \int_{\Omega} (k_b \hat{z}_{ss}(t, s) + k_v \hat{z}(t, s)) ds, \tag{5.28} \]

where \( K_c = R^{-1} B^T \bar{\Pi} \). In this integral representation, the kernels \( k_b \) and \( k_v \) represent the bending gain and velocity gain for the control problem (for more information on this structure of the problem in the context of beams, see [162]. After discretization, the approximate gains \( k_b \) and \( k_v \) can be computed as [161]

\[
[k_b \ k_v]^T = \begin{bmatrix} K & 0 \\ 0 & M \end{bmatrix}^{-1} K_c^T. \tag{5.29}
\]

Demonstrating that \( k_b \) and \( k_v \) converge serves as strong evidence for convergence of the approximate control law to its infinite-dimensional counterpart. It also serves as a means of empirically determining speed of convergence and allows the engineer to determine a sufficient level of convergence of the approximation scheme.

The time required to compute the gains ranged from mere seconds to four hours on a Macbook Pro running OS X Lion with a 2.2 Ghz Core, i7 processor, and eight GB memory. The LQG solution for each mesh was obtained using MATLAB solving routine LQR, which solves the algebraic Riccati equation. The gains were then computed using equation 5.29 and separated into the bending and velocity components in accordance with equation 5.28. Because there is a bending and velocity gain solution for each patch pair, there are a total of nine gains. Rather than plot all of these, plotted here are only the bending and velocity gains associated with patches \( P_{z1}, P_{z4}, \) and \( P_{z7} \), only those on the y-z plane, \( x = W/2 \), which bisects the plate. This was done because the gains in this direction converge the slowest and are conveniently similar in magnitude for the mesh refinements chosen.
The bending gains are plotted using the second spatial derivatives of the basis functions. This is in accordance with the approach described by Gibson and Adamian [161]. The first gain convergence plot, figure 5.12, illustrates the convergence of the bending gain for $P_{z1}$. The gains are plotted for the three mesh refinements in progressively darker shades of gray to aid in visualization of convergence. Note, for progressive mesh refinements, the gains decrease in magnitude, but the values of the gains spatially closest to $P_{z1}$ have larger positive magnitude than the rest of the set. This characteristic is also reflected in figures 5.13 and 5.14 which illustrate the bending gains for patches $P_{z4}$ and $P_{z7}$, respectively.

The bending gains for $P_{z4}$ demonstrate a subtle but interesting cyclical stabilization of the gains near the location of $P_{z4}$, but a steady decrease in magnitude indicative of convergence. The bending gains for patch $P_{z7}$ also appear to converge with the same overall behavior as $P_{z1}$ and $P_{z4}$.

The second component of the gains is associated with the velocity state. Rather than plotting against the second derivative of the relevant basis functions, the basis functions themselves are used. The first figure, 5.15, depicts the velocity gains for $P_{z1}$ and convergence exhibiting little change between the second and third mesh refinements, at least at the scale containing the entire gain solution. Upon closer examination for $y \geq .2$, similar behavior is observed, but smaller in magnitude than for the gains along $y \leq .2$.

Figures 5.16 and 5.17 illustrate similar convergence for the velocity gains of $P_{z4}$ and $P_{z7}$, respectively. There are a few notable differences, however, especially for the velocity gains of $P_{z7}$. The gains appear to change very little between the first and second meshes, but abruptly for the second mesh refinement. It is hypothesized this behavior is due to the increasing distance from the free end of the plate system. Note similar behavior is observed for the $P_{z4}$ gains in figure 5.16, but less extreme.
Nevertheless, the gains demonstrate convergence in all cases, but indicate a necessarily high level of refinement is needed before considering the gains as converged.

These convergence investigations indeed suggest the approximation scheme discussed in section 5.2 and geometry therein is converging, albeit slowly. Furthermore, the gains suggest convergence of the approximate control solution discussed in 5.3. With convergence strongly suggested, numerical experimentation can now be done,
which involves demonstrating the capability of the proposed system to be morphed into the three aerodynamic shapes via optimal control and piezocomposite patch actuators.
5.4.2 Wing Twist

The morphing investigation begins with achieving wing twist. Again, the wing starts from rest and attempts to achieve a desired position described by the twist equation in table 5.1. Engineering intuition predicts the dominantly active patches to be those towards the wing root and along the edges due to the torsional mode dominating any twisting behavior of the system. Thus, the largest patches at the root, patches $P_{z8}$ and $P_{z9}$ as figured in 5.4, are expected to be the most important for twisting behavior.

The control effort for achieving wing twist is presented in figure 5.18. The effort at steady state, approximately $\pm 4000V$, corresponds to the feedforward signal $b$ computed via equations 5.22 - 5.23. The large signals correspond to patches $P_{z8}$ and $P_{z9}$ and are opposite in magnitude as expected. A majority of signals are small compared with inputs to patches $P_{z8}$ and $P_{z9}$. Again, this is as intuition suggests and is a result of the desired surface being flat except for the connection point to the clamped edge. The small nonzero values for patches $P_{z1}$ through $P_{z7}$ are likely an effort to maintain a flat surface in the presence of Poisson’s effects.

![Figure 5.18: Wing twist: control input time history](image)

![Figure 5.19: Wing twist: wing-tip corner deflection time history](image)
While, strictly speaking, this voltage requirement exceeds the accepted input range of piezocomposite devices [82], the plate system tracks the desired twisted position relatively closely as illustrated in figure 5.19. The axis limits of ±1cm correspond to the desired positions of the plate-tip corners. This deflection corresponds to a change in an AOA of approximately 4.8 degrees, as evidenced in figure 5.19. While the voltage input exceeds the accepted range, it nonetheless suggests that a change in wing-tip AOA of only one to two degrees would be possible by using an acceptable input to the patches. Due to linearity, it is therefore predicted that an input of approximately 1.6 kV would yield a wing-tip AOA of two degrees. Of course, such an accepted range depends on the materials chosen and would likely be different for a physical system, therefore these questions are not addressed further here.

Under the applied voltage, the full plate twists as illustrated in figure 5.20. The desired position function, a plane passing through $z = \pm 1\text{cm}$ along the length of the plate, bisects the plate along its center. The piezocomposite actuators drive the plate to twist in such a way that the plate-tip is nearly at the desired position. Since this final position was obtained over time, time snapshots are presented in figure 5.21 to depict the changes in the system. Note there was little, if any, overshoot due to appropriate actuation. This is obviously a desirable quality, especially for aerodynamic control systems, which are particularly prone to instability due to aerodynamic-structural interaction.

It is also interesting to observe the strain fields developed during twisting. The strain fields for the last, steady state snapshot are presented in figures 5.22 - 5.24. As expected, these indicate greatest activation of the patches along the wing root, both deforming the same magnitude but opposite in sign.

Based on these results, piezocomposite actuators can twist a wing and are capable of doing so with currently available power supplies for small UAVs and MAVs. The
Figure 5.20: Wing twist: final surface position

Figure 5.21: Wing twist: surface deformation time snapshots

Figure 5.22: Wing twist: final $\epsilon_{xx}$ field

Figure 5.23: Wing twist: final $\epsilon_{yy}$ field

Figure 5.24: Wing twist: final $\epsilon_{xy}$ field

Voltage input used, illustrated in figure 5.18, was of a scale that indicates meaningful twist would result even from substantially less input voltages.
5.4.3 Wing Camber

The second type of morphing investigated is wing camber.

Note the control input signals in figure 5.25 reach steady state within half of a second. Many signals are plotted here that follow the geometric pattern of being larger in magnitude for actuators one, four and seven, as these actuators offer greatest control of chordwise deformations.

The control input, while modifying camber, deforms the plate in a bending manner in the positive z-direction. The plate-tip corner deflection history illustrates this in figure 5.26, describing the time history of position for both free corners. Purely a change in camber would yield small or zero plate-tip deflection. However, an increase in tip deflection is observed here. This is likely due to Poisson’s effects, which cause deformation in the direction normal to the actuator fibers. The chosen desired surface, which is found in table 5.1 and causes the edge patches to activate simultaneously with the center patches, may help to alleviate such undesired deformations due to secondary effects. The modified camber and undesired bending of the surface is more
readily apparent in figure 5.27, a plot of the final deformed surface (snapshots of this solution are also provided in figure 5.28).

![Figure 5.27: Wing camber: final surface position](image1)

![Figure 5.28: Wing camber: surface deformation time snapshots](image2)

A large discrepancy is found between the desired and final position of the plate, illustrated in figure 5.27. The desired surface was chosen purely from the standpoint of achieving approximately a quadratic surface in the \( x \)-direction. Of course, a desired position can be chosen a posteriori, then the plate-patch system would track precisely the desired position. The only means of determining such a desired position is to prescribe patch inputs until the desired position is achieved, thus the approach contributes nothing general to the solution. Rather, allowing for general functions regardless of whether or not they satisfy the system dynamics is desirable.

Again, it is interesting to observe the strain fields developed. The steady state strain fields for camber are presented in figures 5.29 - 5.31 and indicate greatest activation of the patches along the battens through the wing chord. The strain \( \epsilon_{xx} \) in the \( x \)-direction is distributed throughout the chord of the wing, but nearly zero along the boundaries, indicating the desired function chosen indeed identifies the relevant
features of a cambered wing and yields a control solution that attempts to strain only the plate interior.

![Figure 5.29: Wing camber: final $\epsilon_{xx}$ field](image)

![Figure 5.30: Wing camber: final $\epsilon_{yy}$ field](image)

![Figure 5.31: Wing camber: final $\epsilon_{xy}$ field](image)

Similar to the wing twist investigation, these results suggest piezocomposite actuators can modify wing camber and are, again, capable of doing so with the power supplies currently available for small UAVs and MAVs. Since the voltage input, illustrated in figure 5.25, was of a realistic scale, piezocomposite actuators could be applied to a physical model to achieve camber control. With both wing twist and camber as viable morphing options for this system, the final type of morphing, bending, is now investigated.

### 5.4.4 Wing Bending

The third and final type of morphing investigated is wing dihedral modification, or bending in the z-direction. The desired position is reminiscent of the birds’ wings in figure 2.1. It is expected, as it was with twist, that the dominantly active patches
will be those towards the wing root and along the edges due to the bending mode in figure 5.8, so patches $P_{z8}$ and $P_{z9}$ again. However, unlike the torsional case, one expects a degree of actuation from patches located along the edge, and indeed, the greatest control effort is exhibited by patches $P_{z5}$ and $P_{z6}$, as illustrated in figure 5.32.

The effort at steady state, approximately $-5kV < V < 5kV$, corresponds to the feedforward signal $b$ computed via equations 5.22 - 5.23. The large signals correspond to patches $P_{z8}$ and $P_{z9}$, and are of the same sign and magnitude. Surprisingly, all patches were active to achieve this position.

Again, strictly speaking, the voltage required to achieve this deformation is larger than current constraints of piezocomposite materials. Nonetheless, the control algorithm achieves the desired position to a high degree of accuracy as illustrated in both the tip deflection time history, figure 5.33, and especially in figure 5.34, a plot of the final deformed surface.

Note that only the region between the wing tip and the first batten looking spanwise towards the root is not tracking to a high degree of accuracy, as illustrated in figure 5.34. This is likely due to the positioning of patches along the edge, patches...
$P_{z2}$ and $P_{z3}$ in figure 5.4. Still, the efficacy of the control algorithm to track a desired dihedral is demonstrated wholeheartedly by these results.

\begin{figure}[h]
\centering
\includegraphics[width=0.4\textwidth]{figure5.34}
\caption{Wing bending: final surface position}
\end{figure}

\begin{figure}[h]
\centering
\includegraphics[width=0.4\textwidth]{figure5.35}
\caption{Wing bending: surface deformation time snapshots}
\end{figure}

Again, time snapshots are provided in figure 5.35. Note the first few snapshots are spaced further apart than the remaining, indicative of a quick movement and abrupt deceleration into the desired position with little overshoot.

The final strain fields and confirmation of the most activated patches being those along the wing edges are presented in figures 5.36 - 5.38. Unlike the twisting case, it is evident from figure 5.37, a plot of the strain $\epsilon_{yy}$ in the y-direction, that all the patches are acting to progressively bend the plate into the desired position. There is a large negative strain in the x-direction at the plate-tip edge, most likely due to patch $P_{z1}$ reducing deformations resulting from Poisson’s effects in an effort to maintain a flat plate-tip edge.
5.4.5 Conclusions

In this chapter, a novel way of morphing a wing using desired functions and LQG was introduced. Using a finite element approximation of a plate model composed of carbon fiber battens, piezocomposite self-sensing actuators, and a flexible interior plate material, the wing was successfully morphed into three biologically inspired positions that would be useful for small UAV or MAV control, efficiency, and maneuverability. It was found that a distributed-actuator methodology can continuously morph a wing into a variety of useful shapes.

A feedback design such as this fails to reject unknown inputs to the system if such inputs cause significant bias or have statistical variance much greater than the parameters used to compute the Kalman gain matrix of equation 5.27. Indeed, it is rare that external disturbances actually fit the criteria required for such a feedback control design, that is, disturbances are not normally distributed with zero mean and
known covariance. Therefore, in the following chapter, a novel disturbance observer design capable of estimating the spatial and temporal distributions of a pressure field on a membrane wing is formulated in an effort to eliminate the constraints just discussed.
6 DISTURBANCE OBSERVER DESIGN: IDENTIFYING AERODYNAMIC LOAD FROM STRUCTURAL SENSOR DATA

In this chapter, a novel approach to aerodynamic load identification is derived and tested in numerical simulation and laboratory experiment. The derivation consists of several distinct, equally important components, each of which may seem independent until they are used in combination to effectively identify aerodynamic load on a membrane wing.

Again, the fundamental goal of this chapter is the identification of distributed aerodynamic load from structural response, or, in a more abstract form, to estimate the inputs to a dynamic system from the output of the system, which could be provided by strain, position, or velocity sensors. It is well known that this constitutes an ill-posed problem due to the non-uniqueness of the desired solutions. Coupling non-uniqueness to the ubiquitous noise that is found in real systems, this problem poses a surprising technical challenge. However, by combining approaches found in the literature and reviewed in section 3.2, the unknown input observer (UIO) system formulation and the joint extended Kalman filter (JEKF) approach, and a novel means of obtaining a regularized solution, the problem can be solved relatively accurately for quasi-static loads and potentially time-varying loads.

As with the thin plate model of chapter 5, the membrane model is a PDE and, for use in this work, must be approximated using the finite element method, yielding a system lending itself to estimation and simulation investigations. Multiple, distributed stretch sensors provide the sensed measurements. Furthermore, as it is desired that this work be directly applicable to experiment, this work is done in discrete time; therefore the system resulting from finite element approximation is
also discretized in time, resulting in a discrete time state space representation of the membrane PDE. From this point, the system is placed in a standard UIO form, and application of the extended Kalman filter (EKF) to the system results in the JEKF formulation. A novel means of introducing regularization into the JEKF framework is then derived, allowing for enforcement of smooth, realistic, recursive solutions to the inverse problem. The reader may not be aware of what the JEKF or regularization is or what they achieve; therefore, subsections are included that discuss both and provide conceptual arguments to aid in understanding. After several subsections successfully build on the proposed algorithm, numerical investigations are performed to address both quantitative and qualitative accuracy of the final algorithm. Three load identification numerical experiments are then presented. These are followed by preliminary results from actual wind tunnel tests completed in the Oregon State University wind tunnel that yielded data capable of validating this approach, strongly encouraging further work using the proposed methodology.

6.1 System Geometry and Approximation Scheme

System Geometry and Material Parameters

The membrane system geometry is depicted in figure 6.1, in which light gray denotes membrane domain and darker regions depict cumulative stretch (or nonlinear strain) sensors. Note, in this work, because a conceptual problem is being addressed using data collected for a real membrane model without sensors, the sensors shown in this figure are taken to not contribute to the system dynamics as the self-sensing strain sensors did in the previous chapter. Rather, the stretch, or strain (used interchangeably throughout this work), is simply observed. The dark regions depicted in figure
6.1 are therefore regions of strain integration, the result of which approximates the 
quantity that a stretch sensor measures and can therefore be described as virtual 
sensors. Also, the effect of time delays are ignored, and the only noise present in the 
sensor feedback is artificially prescribed for testing purposes.

![Simulated membrane system geometry: elastic membrane in light gray and stretch sensor regions in dark gray](image)

Figure 6.1: Simulated membrane system geometry: elastic membrane in light gray and stretch sensor regions in dark gray

The membrane is taken to have a width along the x-axis of 14cm and a height 
of 7cm, yielding an aspect ratio of two. This aspect ratio is reflected approximately 
in the laboratory specimen studied in section 6.5, the geometry of which is shown in 
figure 6.2, and allows for consistent sizing of finite elements so that square elements 
are maintained at all iterations. Four virtual stretch sensors are placed on the domain 
in the positions illustrated with geometry motivated by the direction and orientation 
of the muscles found in the bat plagiopatagium (see figure 2.5).

The membrane wing used in laboratory experiment is introduced concurrently 
with its simulated counterpart. The experimental membrane wing is composed of a 
steel frame surrounding and supporting two identical latex membrane sections. Each 
side of the membrane component of the experimental wing is of nearly the same 
dimension and aspect ratio as the simulated wing, offering a conveniently direct com-
parison of numerical and laboratory results. For simplicity, the following discussion 
can be thought of as applying to both the simulated and the experimental wing, as
the same finite element model was used to solve the underlying goal of estimating both membrane and aerodynamic load states.

Figure 6.2: Laboratory membrane wing geometry: elastic membrane in light gray, frame in dark gray, internal membrane width 14 cm, and internal height 7.5 cm

The membrane is pre-tensioned by applying 5% pre-strain, which corresponds to directly substituting \(0.05\) for the strain terms \(\varepsilon_x\) and \(\varepsilon_y\) and letting \(\varepsilon_{xy} = 0\) (assuming zero shear strain) in equation 4.36. Both the numerical investigation and laboratory experiment utilize membranes with 5% prestrain, and identical material parameters so that direct comparisons can be made. The material is slightly inhomogeneous, therefore the pre-strain field varies. The material characteristics of the laboratory specimen are characterized and utilized for all membrane investigations and simulations. Table 6.1 summarizes the membrane material, geometric, and damping parameters.

<table>
<thead>
<tr>
<th>(\rho) ((kg/m^3))</th>
<th>(E) (MPa)</th>
<th>(\nu)</th>
<th>(\eta) ((Ns/m^2))</th>
<th>(\gamma) ((Ns/m))</th>
<th>(h) (m)</th>
<th>(L) (m)</th>
<th>(W) (m)</th>
</tr>
</thead>
<tbody>
<tr>
<td>960</td>
<td>1.3</td>
<td>.36</td>
<td>.0001</td>
<td>5</td>
<td>.00014</td>
<td>.14</td>
<td>.07</td>
</tr>
</tbody>
</table>

**Approximation Scheme**

It is assumed the laboratory specimen’s dynamics can be described by Poisson’s equation, equation 4.32, with the tension values \(T_x\) and \(T_y\) computed using equation
4.36 and the assumption the effective pre-strain in the membrane is 5%. The goal then becomes one of approximating the resulting PDE in a way that allows for simulation and estimation of finite-dimensional quantities and states that converge to relevant infinite-dimensional counterparts upon further refinement of the spatial and temporal approximating meshes.

In general, a finite element approximation of Poisson’s equation requires $C^1$ continuous, linear basis functions, unlike the higher order basis functions used in chapter 5, which were of third order. The approach to introducing regularization in the next section necessitates using basis functions that are twice continuously differentiable; therefore, a cubic basis is used here. There are benefits to using such a basis, however, including superior convergence and accuracy. While a quadratic basis would also provide such benefits, it is expedient to utilize the same cubic B-spline finite element code as was used for the thin plate model of chapter 5 for this problem, as it has been shown to be very accurate and requires only a few mesh refinements to achieve accuracy to several significant digits.

The approximation then proceeds exactly as in section 5.2. Again, a Galerkin approximation scheme approximates the membrane model using spline expansions. From appendix B, the transverse displacements are quantified by the weak model formulation

$$
\rho \int_\Omega \frac{\partial^2 w}{\partial t^2} \Phi \partial \Omega + \gamma \int_\Omega \frac{\partial w}{\partial t} \Phi \partial \Omega - \eta \int_\Omega \left( \frac{\partial^2 w}{\partial t \partial x} \frac{\partial \Phi}{\partial x} + \frac{\partial^2 w}{\partial t \partial y} \frac{\partial \Phi}{\partial y} \right) \partial \Omega \\
+ T_x \int_\Omega \frac{\partial w}{\partial x} \frac{\partial \Phi}{\partial x} \partial \Omega + T_y \int_\Omega \frac{\partial w}{\partial y} \frac{\partial \Phi}{\partial y} \partial \Omega + \int_\Omega P \Phi \partial \Omega = 0 \quad (6.1)
$$
for $\Phi$ in the space of test functions

\[ V = \{ \Phi \in H^2(\Omega) \mid \Phi|_{\partial\Omega} = 0 \}, \]  

(6.2)

where $\Omega$ is the domain of the membrane. They are defined, as in the last chapter, as

\[ \Phi_k(x, y) = \phi_m(x)\phi_n(y), \]  

(6.3)

and the approximating subspace then becomes

\[ V^N = \text{span}\{\Phi_k\}_{k=1}^{N_w} \]  

(6.4)

for $N$ nodes on a rectangular mesh. The philosophy of approximating the dynamics of equation 6.1 is also the same. The relation is projected onto a spline-based, finite-dimensional subspace $V^N \subset V$ to obtain a system discrete in space, continuous in time, and appropriate for a finite-dimensional estimator design. The resulting vector-valued system is simulated using standard ordinary differential equation integration methods.

Now the actual finite element approximation begins by defining the following substitution (and relevant derivatives in time and space):

\[ w(t, x, y) \approx \sum_{i=1}^{N} w_i(t)\Phi_i(x, y). \]  

(6.5)

Utilizing dot notation for post-substitution time derivatives, neglecting the independent variables $(t, x, y)$ for notation simplification purposes, and substituting the above
approximations into the weak form equation 6.1,

\[ \rho h \int_{\Omega} \sum_{i=1}^{N} \tilde{w}_i \Phi_i \frac{\partial \Omega}{\partial x} + \gamma \int_{\Omega} \sum_{i=1}^{N} \tilde{w}_i \Phi_i \frac{\partial \Omega}{\partial y} - \eta \int_{\Omega} \left( \sum_{i=1}^{N} \tilde{w}_i \frac{\partial \Phi_i}{\partial x} \frac{\partial \Phi_i}{\partial x} + \sum_{i=1}^{N} \tilde{w}_i \frac{\partial \Phi_i}{\partial y} \frac{\partial \Phi_i}{\partial y} \right) \frac{\partial \Omega}{\partial x} \\
+ T_x \int_{\Omega} \sum_{i=1}^{N} w_i \frac{\partial \Phi_i}{\partial x} \frac{\partial \Phi_i}{\partial x} \frac{\partial \Omega}{\partial x} + T_y \int_{\Omega} \sum_{i=1}^{N} w_i \frac{\partial \Phi_i}{\partial y} \frac{\partial \Phi_i}{\partial y} \frac{\partial \Omega}{\partial y} + \int_{\Omega} P_z \Phi \frac{\partial \Omega}{\partial x} = 0. \]

Because the basis function coefficients, \( w_i \), are not dependent upon the spatial variables, the integration is moved inside the sums. Relabeling \( \Phi \) as \( \Phi_j \), assuming a finite set of basis functions, and writing as a vector-matrix equation, the above becomes

\[ \rho h M \ddot{z} + D \dot{z} + K_T z = p \]  \hspace{1cm} (6.6)
\[ y = H(z) = z^T C_s z, \]  \hspace{1cm} (6.7)

where \( z = [w_1 \ w_2 \ \ldots \ w_N]^T \), \( D = (\gamma M + \eta K) \), the observation equation 6.7 and matrix \( C \) have been obtained from appendix B.1, and

\[ [M]_{i,j=1}^{N} = \int_{\Omega} \Phi_i \Phi_j \frac{\partial \Omega}{\partial x} \]
\[ [K]_{i,j=1}^{N} = \int_{\Omega} \left( \frac{\partial \Phi_i}{\partial x} \frac{\partial \Phi_i}{\partial x} \frac{\partial \Omega}{\partial x} + \frac{\partial \Phi_i}{\partial y} \frac{\partial \Phi_i}{\partial y} \frac{\partial \Omega}{\partial y} \right) \]
\[ [K_T]_{i,j=1}^{N} = \int_{\Omega} \left( T_x \frac{\partial \Phi_i}{\partial x} \frac{\partial \Phi_i}{\partial x} + T_y \frac{\partial \Phi_i}{\partial y} \frac{\partial \Phi_i}{\partial y} \right) \frac{\partial \Omega}{\partial x} \]
\[ [p]_{j=1}^{N} = \int_{\Omega} P_z \Phi_j \frac{\partial \Omega}{}.

Matrix \( C \) results from approximating the stretch sensor model of chapter 4 using the same basis elements as above. Thus, the system model is linear, but the stretch sensor measurements require a quadratic sensor model, i.e., \( y = z^T C_s z \). For simulation and
control purposes, the system is written in first order state space form for \( x = [z \  \dot{z}]^T \), assuming the mass matrix \( M \) is nonsingular, as

\[
\begin{align*}
\dot{x} &= Ax + P \\
y &= x^T C x,
\end{align*}
\]

(6.8)

where

\[
A = \begin{bmatrix} 0 & I \\ -M^{-1}K & -M^{-1}D \end{bmatrix}
\]

\[
C = [C_s \ 0] \quad P = \begin{bmatrix} 0 \\ M^{-1}p \end{bmatrix}.
\]

Also, the initial conditions for equation 6.8 are

\[
x_0 = \begin{bmatrix} z_0 \\ \dot{z}_0 \end{bmatrix} = \begin{bmatrix} \mathcal{P}g_1 \\ \mathcal{P}g_2 \end{bmatrix},
\]

(6.9)

where \( g_1 \) and \( g_2 \) are defined in equations 4.34 - 4.35 and describe the initial position and velocity of the membrane, respectively. Similar to chapter 5, \( P \) is a projection operator that projects the infinite-dimensional initial conditions into the finite-dimensional subspace \( V \), defined in equation 6.4, and is defined as

\[
\begin{bmatrix} \mathcal{P}g_1 \\ \mathcal{P}g_2 \end{bmatrix} = \begin{bmatrix} S^{-1} \int_{\Omega} g_1 \Phi_j \, d\Omega \\ S^{-1} \int_{\Omega} g_2 \Phi_j \, d\Omega \end{bmatrix},
\]

(6.10)

where

\[
[S]_{i,j=1}^N = \int_{\Omega} \Phi_i \Phi_j \, d\Omega.
\]

(6.11)

The approximate system, given in equation 6.8, can now be simulated with an approximate disturbance observer design. To pursue a discrete model, it is necessary
to place the system above, which is continuous in time, into discrete-time form by calculating the matrix exponential. Although a variety of time discretization techniques exist, it was found that the matrix exponential offered the simplest solution and, of course, the technique can easily be adapted to a real-time implementation.

The approximate system above, because it is linear, has a solution

\[ x(t) = e^{A(t-t_0)}x(t_0) + \int_{t_0}^{t} e^{A(t-\tau)}P(\tau)d\tau. \]

If matrices \( A \) and \( P \) are approximately constant on a sampling time interval of interest, \( \Delta t = t_k - t_{k-1} \), then a new discrete system can be constructed as (from [166])

\begin{align*}
  x_k &= Fx_{k-1} + G \\
  y_k &= x_k^TCx_k,
\end{align*}

where

\begin{align*}
  F &= e^{A\Delta t} \\
  G &= A^{-1}[e^{A\Delta t} - I]P.
\end{align*}

Equations 6.12 - 6.13 are derived on the premise that the external loading of the system is quasi-static on a sampling interval, \( \Delta t \). For highly transitory aerodynamic loading conditions, such an approximation may not be accurate. However, with such sufficiently high sample rate, many relevant loading conditions change sufficiently slowly that the above approximation suffices. Indeed, in this work, only quasi-static loads are investigated, and, as will be demonstrated in section 6.5, the assumption is nonetheless accurate even in laboratory conditions. A very small sample time (1
ms) was used for all numerical investigations in an effort to avoid any errors resulting from these simplifying assumptions.

The discrete formulation of the approximate equations of motion for the membrane can now be used in a JEKF framework.

6.2 A Recursive Approach to Aerodynamic Load Identification

In this section, the load identification approach is derived and discussed. The UIO approach to jointly estimating system and disturbance states is first introduced, followed by the EKF algorithm, which, upon application to the joint system, defines the JEKF. Note, again, the discrete formulation of the EKF (rather than its continuous counterpart) is used because the membrane model has been discretized in both space and time to accommodate discrete data collected in laboratory experiment.

The basic approach of the general UIO, or a specific instance of the approach, the JEKF, is to generate new states that describe the disturbance input to the system and append them to the original system states in the system’s original state space realization. Thus, fundamentally, the JEKF is a direct application of the EKF to the joint system-disturbance state space model. As such, the EKF is first introduced to expedite the solution to the joint system, discussed thereafter.

It should be noted that, although a Kalman filter was utilized in the control design for chapter 5, the approach used here is entirely independent, so parallels between the chapters are limited. In chapter 5, a linear problem necessitated the use of a linear Kalman filter. In the current work, the quadratic stretch (strain) sensor model necessitates the use of nonlinear approaches, the EKF specifically, to obtain the desired solutions. For detailed treatments of the Kalman filter, the EKF, and the relationships between the two, suggested texts are Anderson and Moore [167] and
Gelb [168]. The EKF algorithm presented in this discussion is largely taken from [166] with appropriate modifications made for the local notation and problems.

6.2.1 Unknown Input Observer Form: Specialization to Load Identification

This load identification methodology makes assumptions to represent external pressure/loads as additional states of the state vector x, as in equation 6.29, and solves a joint estimation problem of both external loads and system state. Thus the approach is a disturbance observer, specialized for load identification but utilizing specialized algorithms for distributed parameter systems.

Note that the notation utilized in this subsection should be considered independent from the rest of the chapter. Special attention should be paid to subscripts and equation references so that similar notation does not cause confusion of the terms.

Here, it is assumed that the true load distribution on a wing can be represented by a linear combination of basis functions. More precisely, the true load \( P_z(t, x, y) \) in equation 6.1 can be approximated by a linear combination of basis functions (the finite element basis functions themselves) as

\[
P_z(t, x, y) \approx \sum_{i=1}^{N} f_i(t) \phi_i(x, y).
\]  

(6.14)

When 6.14 is substituted into a finite element approximation, the result is an input matrix \( \hat{p} \) multiplied by unknown basis function coefficients \( f_i \). This assumption allows generic loads to be represented in a second order system (using equation 6.6 as a baseline model),

\[
\rho h M \ddot{z} + D \dot{z} + K_T z = \hat{p} f,
\]  

(6.15)
where \( \hat{p} \) is a \( N \times L \) load input matrix and \( f \) is a \( L \times 1 \) column vector of “load” states. Henceforth, \( \hat{p} \) and \( f \) are taken to be of the same dimension as state vector \( z \), thus \( \hat{p} \) is now \( N \times N \) and \( \hat{p} = M \), where

\[
[M]_{i,j=1}^{N} = \int_{\Omega} \Phi_i \Phi_j \partial \Omega.
\]

An approach must now be developed to estimate state vector \( f \), thereby providing estimates of the distributed load, such that the error between observed and predicted measurements is minimized. This is nearly identical to the general problem of state estimation, except that not only are system states to be estimated, but states describing inputs to the system are as well. Because the system state itself is a function of the load input states, this represents a nonlinear estimation problem.

Writing equation 6.15 in first order form, as was done before for equation 6.6, the following is obtained:

\[
\dot{x} = Ax + \begin{bmatrix} 0 \\ \hat{P} \end{bmatrix} f, \quad (6.16)
\]

where \( x \in R^{2N} \), \( A \) is a \( 2N \times 2N \) plant model, and \( \hat{P} = \frac{1}{\rho h} M^{-1} \hat{p} \), an \( N \times N \) diagonal input matrix with entries \( 1/\rho h \) along the diagonal.

In the spirit of Johnson [127] and Bayless [125], assume the real load distribution \( P_z \) of equation 6.1 has, and only for the sake of this derivation, been generated by a linear system of the form

\[
P_z = \hat{p} f \quad (6.17)
\]

\[
\dot{f} = D(t) + \delta(t), \quad (6.18)
\]
where \( \hat{p} \) is taken from equation 6.15; \( D(t) \) is a matrix function of time to be determined; \( f = (f_1, f_2, \ldots, f_L) \); \( \delta = (\sigma_1, \sigma_2, \ldots, \sigma_p) \); and these are all completely unknown sequences of random intensity, randomly occurring, isolated delta functions, i.e. Gaussian white noise. Without loss of generality, take \( L = N \) such that the number of load states coincide with the number membrane position states.

Of course, the true form of \( P_z \) remains unknown, but the assumption provides a means of generating an artificial state space representation of the disturbance. The precise form of the state space representation dictates the types of disturbances that can be estimated. Note also that this assumption is very similar to that of a shaping filter. Assume the matrix function of time, \( D(t) \) in equation 6.21, can be represented by a power series of the form

\[
D(t) = D_0 + D_1 t + D_2 t^2 + \ldots ,
\]

(6.19)

where the \( D_i \) are constant matrices. Equation 6.19 is then truncated at a desired point, which determines the state space representation of the additional disturbance states. Here, equation 6.19 is truncated at and beyond \( D_0 \), suggested by setting \( \dot{f} = 0 \). This is equivalent to providing knowledge to the estimation algorithm that explicitly states the disturbance is a constant input to the system. Conveniently, this choice does not forbid estimation of a time-varying disturbance so long as the rate of change of the disturbance is very small compared with the sample rate of the estimator. It should also be noted that with each retained \( D_i \) in the truncation of equation 6.19, the dimensionality of the resulting full state space model will increase in size corresponding to the size of the disturbance state vector \( f \) in equation 6.20.

---

1See pages 84 - 87 in [169] or page 133 in [168]. A shaping filter accounts for colored noise in an estimation algorithm. By assuming the colored noise (correlated noise, unknown deterministic input, etc.) to be generated by a linear system, itself driven by white noise, the standard Kalman filter algorithm can easily be applied to the augmented system at the cost of no longer being optimal.
Choosing \( f = D_0 \) then yields the obvious \( \dot{f} = 0 \). The hypothesized disturbance-generating system, equations 6.20 - 6.21, then becomes

\[
P_z = \hat{p}f \\
\dot{f} = \delta(t),
\]

where the changes observed in \( f \) are taken to be the result of noise inputs to the system. In accordance with Bayless and Brigham [125] and using equations 6.16 - 6.21, one can now write a coupled system consisting of the original system, augmented load states, and appropriately modified measurement equation (taken directly from equation 6.13) as

\[
\begin{bmatrix}
\dot{x} \\
\dot{f}
\end{bmatrix}
= \begin{bmatrix} A & \hat{P} \\ 0 & 0 \end{bmatrix}
\begin{bmatrix} x \\ f \end{bmatrix}
+ \begin{bmatrix} \zeta \\ \kappa \end{bmatrix}
\]

\[
y = x^T C_s x + \theta,
\]

where \( \zeta, \kappa, \) and \( \theta \) are uncorrelated, zero-mean, Gaussian-white-noise, random vectors of proper size with correlation matrices

\[
E\{\zeta(t)\zeta'(\tau)\} = \Xi \delta(t - \tau)
\]

\[
E\{\kappa(t)\kappa'(\tau)\} = \Sigma \delta(t - \tau)
\]

\[
E\{\theta(t)\theta'(\tau)\} = \Theta \delta(t - \tau).
\]

Equations 6.22 - 6.23 describe the generic, first order disturbance observer design. The above system, equations 6.22 - 6.23, represents a state space model which can be used in an estimation algorithm to provide estimates for both system and disturbance states, the disturbance states themselves being encapsulated by vector \( f \).
The disturbance states correspond to unknown basis element coefficients, that is, the coefficients of the $\Phi_i$, similar to the $w_i$ in equation 6.5. Therefore, reconstruction of the identified pressure distribution on the proposed membrane model can be done exactly as is done in general when using the finite element method, by multiplying by the basis elements. Also, note the disturbance now evolves alongside the system state, the two effectively coupled dynamically. Although complicating the problem of estimating system states, the coupling results in improved system state estimates.

The augmented system model in equation 6.22 is also discretized using the matrix exponential approach, yielding a discrete joint system and load state model,

\[
\begin{align*}
\begin{bmatrix} x_k \\ f_k \end{bmatrix} &= \hat{F} \begin{bmatrix} x_{k-1} \\ f_{k-1} \end{bmatrix} + \begin{bmatrix} \zeta \\ \kappa \end{bmatrix} \\
y_k &= x_k^T C x_k,
\end{align*}
\]

where

\[
\hat{F} = \exp \left( \begin{bmatrix} A & \hat{P} \\ 0 & 0 \end{bmatrix} \Delta t \right),
\]

and noise disturbances $\zeta$ and $\kappa$ approximate their continuous counterparts appropriately. The system 6.24 - 6.25 can now be solved directly by the EKF. Doing so yields what is commonly referred to as the JEKF. The EKF is now briefly introduced.
6.2.2 Obtaining An Estimation: The Joint Extended Kalman Filter

Consider the discrete, nonlinear system with state variable vector $x$ (a general form of the system of equations in the previous subsection, equations 6.24 - 6.25):

\[
\begin{align*}
x_k &= f_{k-1}(x_{k-1}, w_{k-1}) \\
y_k &= h_k(x_k, v_k) \\
w_k &\sim (0, Q_k) \\
v_k &\sim (0, R_k),
\end{align*}
\tag{6.26}
\]

where functions $f$ and $h$ represent system dynamics and measurement matrices, respectively, and $w_k$ and $v_k$, the process and measurement noise, are taken to be Gaussian white noise characterized by zero mean and noise covariance matrices $Q_k$ and $R_k$, respectively. These random variables characterize the noisy inputs $w$ and $v$ to the system and measurement equations. Taylor series approximations of the above can be used to derive a linear approximation to the nonlinear system above. After this is completed, application of the linear Kalman filter to the time-varying approximate system yields the EKF algorithm (see [166] for details of this approach),

\[
\begin{align*}
P_k^- &= F_{k-1} P_{k-1}^+ F_{k-1}^T + L_{k-1} Q_{k-1} L_{k-1}^T \\
K_k &= P_k^- H_k^T (H_k P_k^- H_k^T + M_k R_k M_k^T)^{-1} \\
\hat{x}_k^+ &= \hat{x}_k^- + K_k [y_k - h_k(\hat{x}_k^-, 0)] \\
P_k^+ &= (I - K_k H_k) P_k^- (I - K_k H_k)^T + K_k R_k K_k^T,
\end{align*}
\tag{6.27-6.30}
\]
where

\[ F_{k-1} = \left. \frac{\partial f_{k-1}}{\partial x} \right|_{\hat{x}_{k-1}} \]  \quad (6.31)  
\[ L_{k-1} = \left. \frac{\partial f_{k-1}}{\partial w} \right|_{\hat{x}_{k-1}} \]  \quad (6.32)  
\[ H_k = \left. \frac{\partial h_k}{\partial x} \right|_{\hat{x}_k^-} \]  \quad (6.33)  
\[ M_k = \left. \frac{\partial h_k}{\partial v} \right|_{\hat{x}_k^-} \]  \quad (6.34)  
\[ \hat{x}_0^+ = E(x_0) \]  \quad (6.35)  
\[ P_0^+ = E \left[ (x_0 - \hat{x}_0^+)(x_0 - \hat{x}_0^+)^T \right]. \]  \quad (6.36)  

Subscripts \( k \) and \( k-1 \) denote the discrete time steps, i.e., time step \( k = 1 \) corresponds to initial time \( t = 0 \), and \( k = 2 \) thus denotes \( t = \Delta t \), where \( \Delta t \) is the sample time of the system (again, in all simulated cases, \( \Delta t = 1 \) ms). While equations 6.27 - 6.36 can be rather daunting to understand, the algorithm is actually quite intuitive. Stated simply, rather than attempt to construct an estimator for the nonlinear system 6.26, which, in general, explicitly forbids a linear update of covariances, the nonlinear system is turned into a time-varying linear system to which linear estimation techniques, including the Kalman filter, can be applied [166].

Execution of the above algorithm yields an estimate for state vector \( x \) for each sample time. Equations 6.27 - 6.30 are evaluated explicitly and used for feedback to drive the estimated system so that the covariance of the estimate error for each state is minimized. Specifically, equation 6.27 is the estimation error covariance update equation. For strictly linear systems, time-varying or otherwise, it describes how error covariance propagates through the linear system, i.e. the effect the system dynamics have on the process noise covariance. The Kalman gain update, equation 6.28, follows directly from combining the error covariance term and sensor matrix into
a linear system that provides an optimal update to the state estimate (equation 6.29) to drive the estimated system to minimize the error covariance, that is, the trace of matrix $P_k^+$.

Equations 6.27 - 6.36 can be directly applied to the real plant model and measurement system, equations 6.12 - 6.13, and the joint system-disturbance state space model, equations 6.24 - 6.25, to yield estimates of the system and disturbance states. This yields the JEKF approach. From this point, estimates for both system and disturbance states can be obtained, either by numerical simulation or real strain measurements obtained from a membrane in the laboratory.

However, of particular concern for the above JEKF algorithm is the requirement that process noise covariance be available for the disturbance plant model itself. The statistical nature of the force is required for true optimality of the algorithm and allows for tunability to particular problems. The statistical characterization of process and measurement mean and covariances is modified to include terms specifically for the disturbance states themselves. Consider the process noise covariance matrix $Q_k$ from equation 6.26 extended to include disturbance covariance assumptions (because of the concatenation of the system and disturbance into a joint vector), resulting in covariance matrix $\hat{Q}$,

$$\hat{Q} = \begin{bmatrix} Q & 0 \\ 0 & Q_d \end{bmatrix},$$

(6.37)

where independence between state and disturbance states is explicitly assumed, resulting in $\hat{Q}$ with zero matrices in the state-disturbance, cross-covariance locations. While the process noise for the true system states is determined or assumed for particular problems based on empirical results, the submatrix $Q_d$ in the above equation is prescribed or tuned to particular load identification problems. General engineering rules of thumb exist if it is known $f_i$ vary by some amount over a specific time interval.
In general, however, regardless of the specified values of $Q_d$, a measure of tuning will be required. Here, the rule of thumb given by Gelb on page 350 of [168] is assumed: for a parameter, or nodal load, $f_i$, which is likely to change by an amount $\Delta f_i$ over an interval $\Delta t$, the corresponding $i$th diagonal entry in $[Q_d]_{(i,i)}$ can be computed as

$$[Q_d]_{(i,i)} = \frac{\Delta f_i^2}{\Delta t}. \quad (6.38)$$

Thus, for a static problem, $Q_d$ can be taken to be zero if the load is not expected to change. For a time-varying load, however, this equation must be used to approximate $Q_d$ to reflect a changing value and allows a deterministic quantity that changes in time to be treated as stochastic. Extensive testing of the algorithm at hand revealed this rule of thumb does indeed yield better results than arbitrarily assigning values to $Q_d$.

The measurement noise covariance $R$ in equation 6.26 does not change in the joint extended formulation, but the initial error covariance, defined in equation 6.36, changes as a consequence of the additional states. Much like $Q_d$, $P_0^+$ is extended appropriately to accommodate the lack of knowledge regarding the initial error covariance, initial condition of the system, and disturbance states. In many circumstances, the exact initial condition is known for the system state. However, rarely, if ever, is the initial condition of the load (or any disturbance for that matter) known.

Early on in this work, it was realized that the resulting estimates were poor due to the inherently ill-posed nature of the problem$^2$. The reason for this can be explained thus: from a discrete number of strain measurements, a potentially infinite dimen-

---

$^2$Hadamard’s well-posedness theorem states that a problem is well posed if
- it has a unique solution
- the solution depends continuously on the given data
otherwise the problem is said to be ill-posed [170].
sional quantity is being estimated. Because there are many load distributions that result in identical strain measurements, this generally results in wildly varying identified loads, which are exceedingly vulnerable to noisy measurements and modeling errors. A common approach to preventing such undesirable effects can be found in the field of inverse problem theory, and is referred to as regularization. Incorporating regularization into such estimation approaches, such as offline, least squares schemes, is well documented. However, adaptations of such approaches to recursive estimation schemes such as the JEKF are unavailable in the literature or are ad hoc, supported only by heuristic arguments. In the next subsection, a novel means of enforcing a smooth solution to the load identification problem is presented.

6.2.3 Smooth Estimates: A Novel Implementation of Regularization

Enforcing smooth, unique solutions to an inverse problem is not, in and of itself, a new concept. However, enforcing smoothness with regularization in a JEKF framework is, as far as the author is aware, an entirely novel implementation of regularization. In fact, the means by which regularization is introduced is intuitive and not found in the literature, but nonetheless agrees with similar approaches that are found in the literature. Furthermore, regularization allows for a mathematical means of constraining the identified load, subject to the “belief” that the true aerodynamic load will exhibit a high degree of smoothness. Such beliefs, degrees of belief, or seemingly subjective knowledge, can be incorporated into the present framework if pursued from the Bayesian statistical perspective.

Consider the issues arising due to ill-posedness when estimating system state from few measurements. In general, regularization is used to force a unique solution in the presence of noise. However, it suffices to say that not only do regularization techniques
allow for improvements in estimation of inverse problems with noisy measurements,
but can also be used to enforce “smoothness” of solutions in general. This suggests
that if one can include regularization in the Kalman filter framework for parameter
identification, identified load solutions could be constrained to be smoothly varying.
Just how to include regularization into the Kalman filter framework is suggested
by the initial condition of the error covariance, $P_0^+$, in equation 6.36. It is known
that letting this value become zero or infinity dictates, statistically, what problem is
being solved. To aid in understanding the modification that must be made to the
Kalman filter equations, consider the Bayesian statistics perspective of the estimation
problem, i.e., Bayes’ theorem states that the probability of state $x$ given observation
$y$ is given by

$$p(x|y) = \frac{p(y|x)p(x)}{p(y)}, \quad (6.39)$$

which can be written as a recursive relationship for a dynamic system, provided the
measurements are statistically independent from the states, and the system (taken in
context) can be approximated by a first-order Markov process [171]. This relationship
describes the form of Bayes’ theorem relevant in this work; that is, for current time-
step $k$, Bayes’ theorem can be written as

$$p(x_k|y_k) = \frac{p(x_k|y_{k-1})p(y_k|x_k)}{p(y_k|y_{k-1})}. \quad (6.40)$$

A discussion of how to approach state estimation from the Bayesian perspective was
perhaps first provided in Ho and Lee [172], and the reader wishing to learn more
about what these terms mean should consult an introductory text on the subject,
such as Bolstad [173]. From Bolstad, Bayes’ theorem may be described intuitively by
writing

\[
posterior = \frac{prior \times likelihood}{\int prior \times likelihood}, \quad (6.41)
\]

where the posterior corresponds to the left hand side of equation 6.40; the prior corresponds to \(p(x_k|y_{k-1})\) and defines the knowledge of the model; the likelihood, \(p(y_k|x_k)\), incorporates measurement noise into the overall estimate; and the denominator is sometimes referred to as the evidence and involves an integral. The posterior represents the knowledge of the state of nature after measurements \(y\), and, by definition, contains all the information necessary for estimation. The prior represents the knowledge of the distribution of the real state \(x\) before measurements are applied, and the likelihood provides appropriate weighting and measurement knowledge for construction of the posterior.

For linear, Gaussian assumptions on the model and noise, equation 6.40 admits an analytic solution, which is the Kalman filter [171]. For nonlinear models, i.e., the system in equation 6.26, the Kalman filter algorithm can be extended using linearization and approximation of the process and measurement models, resulting in the EKF equations 6.27 - 6.36. Thus, this discussion applies directly to the JEKF discussed previously.

Taking a step back, Vogel demonstrates that regularization and the MAP estimator are intimately related in that regularization arises in the choice of prior [174]. This is the case regardless of whether or not the estimator is recursive or single-step: thus, regularization can be implemented in a recursive filter by simply changing the prior accordingly. In [172], it is demonstrated that the prior information (in the Bayesian sense) is encapsulated in a single term \(P_0^+\) corresponding exactly to the same term that describes the knowledge of the initial state estimate error, thus one is led to consider modification of the single term \(P_0^+\) to include regularization in the EKF.
algorithm. Using Bayes’ theorem to generate the cost function which the Kalman filter solves recursively, it is easy to show that a regularized solution can be found by simply modifying the initial condition $P_0^+$ due to the recursive nature of the filter algorithm.

From [172], a single stage of the Kalman filter can be derived from the Bayesian perspective as follows. Given a set of $p$ measurements, $y = (y_1, y_2, \ldots, y_p)$, the relationship

$$y = Cx + v,$$

and the statistical assumptions of the Kalman filter (statistical characterization of the process and measurement noises, i.e., the definitions in equation 6.26), the use of Bayes’ theorem, after assuming the prior and likelihood are normally distributed with appropriate (Gaussian) assumptions for process and measurement noise, results in the following probability model for the posterior:

$$p(x|y) = \beta e^{-\frac{1}{2}[(x-x)TP^{-1}(x-x)]},$$

where

$$\beta = \frac{|CP_0^+CT + R|^{\frac{1}{2}}}{(2\pi)^{n/2}|P_0^+|^{1/2}|R|^{1/2}}$$

and

$$P^{-1} = (P_0^+)^{-1} + CT R^{-1} C.$$

If one takes the natural logarithm of both sides of this equation, it can be written as

$$\log p(x|y) = \log \left( \beta e^{-\frac{1}{2}[(x-x)TP^{-1}(x-x)]} \right)$$

$$= -\frac{1}{2}(x - \bar{x})^TP^{-1}(x - \bar{x}) + c,$$
where \( c = \log \beta \). Substituting \( P^{-1} = (P_0^+)^{-1} + C^T R^{-1} C \) and expanding yields

\[
\log p(x|y) = -\frac{1}{2} (x - \bar{x})^T (P_0^+)^{-1} (x - \bar{x}) - \frac{1}{2} (x - \bar{x})^T (C^T R^{-1} C) (x - \bar{x}) + c.
\]

Consider \( E(x) = \bar{x} = 0 \), then the above becomes

\[
\log p(x|y) = -\frac{1}{2} x^T (P_0^+)^{-1} x - \frac{1}{2} x^T (C^T R^{-1} C) x + c,
\]

which is similar in structure to the classical least squares cost function (see [174]). Note that the second term, involving \( C \) explicitly, is the reason why the Kalman filter is sometimes referred to as a “temporal regularization” filter. More importantly, note that one can expand the above by arbitrarily introducing a \( P_0^+ = ((\tilde{P}_0^+)^{-1} + \lambda \Lambda)^{-1} \) to yield a cost function with secondary spatial regularization term \( \lambda \Lambda \), i.e.,

\[
\log p(x|y) = -\frac{1}{2} x^T (\tilde{P}_0^+)^{-1} x - \frac{\lambda}{2} x^T \Lambda x - \frac{1}{2} x^T (C^T R^{-1} C) x + c. \quad (6.43)
\]

Therefore, one need only modify the initial error covariance term \( P_0^+ \) appropriately to solve the more general Tihonov regularization problem. It can be shown that minimization of the above equation yields the classical Kalman filter, as in equations 6.27 - 6.36, except that \( P_0^+ \) is modified to include the regularization matrix and weighting parameter as follows:

\[
\hat{x}_0^+ = E(x_0) \\
\tilde{P}_0^+ = E [(x_0 - \hat{x}_0^+)(x_0 - \hat{x}_0^+)^T] \\
P_0^+ = ((\tilde{P}_0^+)^{-1} + \lambda \Lambda)^{-1}. \quad (6.44)
\]
Consider the recursive relationship for $P_k^+$ given in equation 6.30 with the initial condition given in equation 6.36. In general, the initial error covariance matrix $P_0^+$ describes the knowledge, or lack thereof, regarding the initial condition of the real system. If perfect knowledge of the real system’s initial state is available, then $P_0^+ = 0$, as the covariance of the error must be zero by definition. On the other hand, if little is known regarding initial error covariance, then $P_0^+ = \infty$, as is suggested by standard texts on the subject. Thus, the Kalman filter is solving the above cost function and is already equipped with a specific, optimal form of regularization. Including the modified regularization term above, equation 6.44, does not prevent an optimal solution from being found, but the solution will retain optimality only for some metric based on the modified cost function. In other words, the cost function determines the solution, which will be optimal for the given cost function that generated it.

Equivalent approaches are discussed in Kaipio [175], and a comparison of these and the approach used here are provided in [176], where derivation of the approach was accomplished in a heuristic manner rather than the comparatively rigorous approach presented in this dissertation, which is based on Bayes’ theorem, the origin of the Kalman filter. Although incorporation of regularization in the Kalman filter equations modifies the equations, as discussed in [176], it nonetheless remains an optimal filter due to the inherent cost function minimization, i.e. equation 6.43. However, when used to estimate unknown inputs or parameters in a system, the approach is not optimal. This can be seen by interpreting the disturbance observer model, equation 6.22, as an approach to modifying the Kalman filter to account for colored noise, that is, noise with nonzero mean. It is well known that the Kalman filter is no longer optimal for such cases, but can nonetheless yield excellent results through a variety of similar approaches, including the one utilized here.
Choice of Regularization Matrix $\Lambda$

Choosing $\Lambda$ is problem specific and depends on the precise goal of the estimation scheme. In the case of estimating aerodynamic load on a wing via strain, there are a few characteristics of the load that allow for a fairly general but relevant choice. Such loads, at least those that exist in nature at low Reynolds numbers, are generally continuously varying, smooth functions. Due to the ill-posedness of the load identification problem, the added constraints placed on the identified load function by enforcing “smoothness” would certainly help identify a (perhaps unique) load given the strain data. The question remains how to enforce the assumption of smoothness on identified load distributions.

It turns out that such an approach is used routinely in spline data analysis, where it is often referred to as a “roughness penalty,” enforced to create smoothly varying best-fits of data. In [177], Ramsay and Silverman utilize the second derivative of the identified function to ascertain its smoothness. Thus, by using the quadratic form formed of inner products of second derivatives of the finite element basis functions,

$$
[\Lambda]_{i,j=1}^N = \int_{\Omega} \left( \frac{\partial^2 \phi_i}{\partial x^2} \frac{\partial^2 \phi_j}{\partial x^2} + \frac{\partial^2 \phi_i}{\partial y^2} \frac{\partial^2 \phi_j}{\partial y^2} \right) d\Omega,
$$

(6.45)

the algorithm can be made to effectively penalize large second derivatives states, thereby enforcing smoother solutions.

Again, because the Kalman filter is recursive in nature, the above analysis applies to the JEKF originally introduced earlier, providing a means of obtaining a regularized estimate of external load. While it may be of some value to obtain a regularized solution for the membrane system states, that is not done here. Rather, a modified form of the regularization matrix $\Lambda$ is utilized. Specifically, since the JEKF requires
a joint matrix of error covariances for system and disturbance states, this allows for regularization solely of the load identification solution.

The general form of the error covariance matrix in equation 6.36 becomes, after the above regularization,

\[
\tilde{P}_0^+ = E[(x_0 - \hat{x}_0^+)(x_0 - \hat{x}_0^+)^T] \\
P_0^+ = (\tilde{P}_0^{-1} + \lambda \Lambda)^{-1} = \left( \begin{bmatrix} \tilde{P}_{s0}^+ & 0 \\ 0 & \tilde{P}_{d0}^+ \end{bmatrix} \right)^{-1} + \lambda \begin{bmatrix} 0 & 0 \\ 0 & \Lambda_d \end{bmatrix}^{-1},
\]

where the subscripts \(s\) and \(d\) refer to the system and disturbance states, respectively. Thus, regularization is only applied to the disturbance states using this approach. The choice of \(\lambda\) is not arbitrary and is generally determined empirically or through extensive simulations. In this work, it is determined approximately for each sensor configuration and set of parameters chosen, as the “best” value of \(\lambda\) is problem specific. A Nelder-Mead approach is used in choosing \(\lambda\) for a particular sensor/parameter configuration. It does not return an optimal value for \(\lambda\), but ensures that \(\lambda\) is chosen such that the error is minimized within user-specified criteria.

The regularization solution above is therefore implemented simply by modifying the initial condition for the error covariance. The overall algorithm can be summarized as follows: The EKF from subsection 6.2.2 is applied to the disturbance observer system in equation 6.22. The error covariance initial condition from equation 6.44 is modified to admit a regularized solution using the results in equation 6.46. This yields a simultaneous estimate for the membrane system state and load state, providing a direct, coupled estimate that can be used to reproduce the load input to the system via the basis functions in the standard finite element approach.
6.3 Convergence and Consistency: Numerical Results

Similar to the investigation in chapter 5, it is important to assess convergence and consistency of the approximate model and estimation approach. A convergent approximation scheme is one which converges to the solution of the PDE it approximates as the spatial discretization, or mesh, is refined. This contrasts with a consistent estimation scheme, which is described as providing a result convergent to the true parameter being estimated as more samples or sensors are used.

Since analytical solutions are available for Poisson’s equation on a rectangular domain, the approximate modal frequencies and eigenvalues are compared to exact values to determine convergence to the PDE solution.

To demonstrate consistency of the JEKF algorithm, a specific mesh is chosen and an increasing number of sensors is added to the domain iteratively. For each iteration, the final load estimate is recorded. A total of three iterations are performed. A consistent algorithm will yield increasingly accurate estimates as the number of sensors increases.

After convergence and consistency are addressed, three example problems are solved. A biologically inspired sensor configuration is used and demonstrated to work well for a variety of static load distributions, including non-conforming loads.

6.3.1 Convergence of System Modes and Eigenvalues

Three mesh refinements are used to qualitatively determine convergence of the modal frequencies and eigenvalues. The meshes shown in figures 6.3 - 6.5 are determined by the membrane geometry, with the constraint that only square elements be used at each iteration. The first mesh computed is $13 \times 7$ nodes, which corresponds to
72 elements, the second $25 \times 13$ nodes corresponding to 288 elements, and the third, $49 \times 25$ nodes corresponding to 1152 elements. Thus, refinement by a factor of two in each direction is used. The time required to compute meshes and eigenvalues ranged from seconds to one minute on a Macbook Pro running MATLAB R2011b on OS X Lion with a 2.2 Ghz Core, i7 processor, and 8 GB memory.

In figures 6.6 - 6.9, the mode shapes are plotted. Because the membrane is isotropic, the shapes agree well with the undamped exact solution. Interestingly, for the laboratory test discussed later in this chapter, such modes were readily visible during wind tunnel testing.

Damping is present in this model, thus the imaginary components of the approximated eigenvalues are slightly different from the exact solution. The undamped modal frequencies are the same, however, and are presented in table 6.2. Conver-
Figure 6.6: First mode of membrane system

Figure 6.7: Second mode of membrane system

Figure 6.8: Third mode of membrane system

Figure 6.9: Fourth mode of membrane system

gence is swift, especially when compared with the plate model in the previous chapter which contained inhomogeneities and discontinuities. The exact frequencies for an undamped membrane are given in the first column of table 6.2 and match to several digits of accuracy with the final mesh refinement in the last column of the table.

An exact solution for the damped case is unavailable, but the eigenvalues are demonstrated to converge to several significant figures after two mesh refinements (table 6.3). Note that the undamped modal frequencies in table 6.2 are intimately related to the imaginary portion of the damped eigenvalues of table 6.3, the latter
Table 6.2: Membrane system: first four undamped modal frequencies for mesh refinements $13 \times 7$, $25 \times 13$, and $49 \times 25$

<table>
<thead>
<tr>
<th>Mode</th>
<th>exact</th>
<th>$13 \times 7$</th>
<th>$25 \times 13$</th>
<th>$49 \times 25$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>82.14063305 Hz</td>
<td>82.140658289 Hz</td>
<td>82.1406341 Hz</td>
<td>82.14063306 Hz</td>
</tr>
<tr>
<td>2</td>
<td>103.90059556 Hz</td>
<td>103.90063548 Hz</td>
<td>103.90059613 Hz</td>
<td>103.90059557 Hz</td>
</tr>
<tr>
<td>3</td>
<td>132.44779106 Hz</td>
<td>132.44827162 Hz</td>
<td>132.44779727 Hz</td>
<td>132.44779115 Hz</td>
</tr>
<tr>
<td>4</td>
<td>151.45984364 Hz</td>
<td>151.4655282 Hz</td>
<td>151.45989864 Hz</td>
<td>151.45984442 Hz</td>
</tr>
</tbody>
</table>

being approximately equal to the former modulus $2\pi$.

Table 6.3: Membrane system: first five eigenvalues for mesh refinements $13 \times 7$, $25 \times 13$, and $49 \times 25$

<table>
<thead>
<tr>
<th></th>
<th>$13 \times 7$</th>
<th>$25 \times 13$</th>
<th>$49 \times 25$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\lambda_1$</td>
<td>-31.919407 + 515.116976i</td>
<td>-31.919399 + 515.116819i</td>
<td>-31.919399 + 515.116817i</td>
</tr>
<tr>
<td>$\lambda_2$</td>
<td>-39.910341 + 651.605852i</td>
<td>-39.910325 + 651.605605i</td>
<td>-39.910325 + 651.605601i</td>
</tr>
<tr>
<td>$\lambda_3$</td>
<td>-53.228785 + 830.492986i</td>
<td>-53.228537 + 830.490015i</td>
<td>-53.228534 + 830.489977i</td>
</tr>
<tr>
<td>$\lambda_4$</td>
<td>-63.885917 + 949.533112i</td>
<td>-63.883134 + 949.503990i</td>
<td>-63.883102 + 949.503651i</td>
</tr>
<tr>
<td>$\lambda_5$</td>
<td>-71.876851 + 1029.731486i</td>
<td>-71.874060 + 1029.704576i</td>
<td>-71.874027 + 1029.704263i</td>
</tr>
</tbody>
</table>

Because the second mesh refinement, $25 \times 13$ nodes, is used for all numerical experiments, the accuracy of the numerical results is expected to be satisfactory. Indeed, any inaccuracy resulting in the laboratory experiment is likely due to model mismatch with reality.

### 6.3.2 Convergence of Regularization Parameter $\lambda$

An important component of this work is the choice of regularization parameter $\lambda$. A simple Nelder-Mead search algorithm is used, along with simulation of the membrane
system, equation 6.12, to determine an optimal \( \lambda \) for obtaining a smooth load estimate. The MATLAB function \( \text{fminsearch} \) is used to execute this, and the results of this approach are summarized in figure 6.10.

![Figure 6.10: Approximation of optimal \( \lambda \) via search algorithm](image)

The final value of \( \lambda = 8.582470 \times 10^{14} \) was used for all simulations.

### 6.3.3 Evidence of Consistency

Now investigated are actual estimation results as pertaining to assessing consistency of the proposed load identification algorithm. The load utilized is of rectangular shape and notoriously difficult to estimate due to discontinuities. Application of the identification algorithm to a membrane under this load introduces oscillatory behavior in the identified load solution near the boundaries of the load distribution itself. However, with the regularization constant chosen above, the consistency of the approach is readily evident for this particular load, and thus it is likely similar consistency will be found for other loads.

As can be seen in figures 6.11 - 6.17, the load estimate converges approximately to the real load distribution; the rectangular distribution of the load is readily apparent.
for the final, dense sensor distribution. Note that such loads actually cannot exist in an aerodynamic setting, at least naturally occurring loads are highly unlikely to exhibit such distributions.

Figure 6.11: Consistency investigation: exact load distribution

Figure 6.12: Consistency investigation: first sensor configuration

Figure 6.13: Consistency investigation: first load estimate

Figure 6.14: Consistency investigation: second sensor configuration

Figure 6.15: Consistency investigation: second load estimate
6.4 Quasi-Static Load Identification for a Simulated Membrane Wing

In this section, three membrane loading problems are investigated numerically. The regularized JEKF algorithm is executed for a virtual membrane, estimates system state, and identifies approximately the external load distribution on the membrane.

Now that the approach has demonstrated convergence and consistency, an investigation of the identification of three loading cases with varying, artificially introduced noise levels is now pursued. The loads can be described as the first few terms from a general Fourier series and are chosen in an effort to capture many of the interesting features that aerodynamic loads may exhibit. The load distributions can be written analytically as

\[ P_1 = 200 \sin(\pi x) \sin(\pi y) \]  
\[ P_2 = 200 \sin(2\pi x) \sin(\pi y) \]  
\[ P_3 = 200 \sin(\pi x) \sin(2\pi y), \]

represent a basis of sorts for the variety of loads that may occur, and provide definite examples which can be used to demonstrate the algorithm at hand. The loads
investigated here are not time-varying, but the algorithm has demonstrated successful preliminary load identification results for time-varying loading situations, so long as sampling rate is sufficiently great enough that the loads can be approximated as steady state on relevant intervals.

The system geometry and sensor layout shown in figure 6.1 is used. The layout conceptually represents muscle stretch sensors similar to those observed in the bat wing. As the sensors are far larger than real muscle fibers, the goal is not to replicate biology in detail, rather simply through the sensor concept itself.

The following investigations can be described as follows: The original, finite element membrane model is solved in discrete time with an input corresponding to the loads in equation 6.49. As a result of load input, the membrane will deform and, because of damping, reach steady state. For each sample time, the sensor output equation 6.13 will be evaluated to yield an approximate value of cumulative strain over the designated sensor areas, as illustrated in figure 6.1. The regularized JEKF algorithm will utilize a discretized form of the joint system-disturbance model, equation 6.22 (also computed using matrix exponential in precisely the same manner as was done in subsection 6.1 for the original semi-discrete membrane state space model. Utilizing the sensed measurements, the JEKF generates estimates for both system and load states, where the load states form an estimate for the load input to the simulated plant model. This process repeats until steady state is achieved.

A comparison is provided for each load case, artificial Gaussian white noise being introduced to the sensor model, equation 6.13, in the amounts of 1% and 5% magnitude of maximum sensor output. Such a high level of noise inevitably degrades estimates, but the qualitatively important features and magnitudes are nonetheless retained in the estimated system state and identified loads. The measurement noise covariance for the JEKF algorithm is set explicitly at $1 \times 10^{-4}$ for simplicity sake and,
in general, the value depends on statistical characterization of whatever measurement system is being used.

For each load case simulated, the final load estimate is plotted for both noise cases. Also illustrated and discussed are the $L^2$ relative error norms for the load and position estimates for each time step, followed by three dimensional plots of actual and estimated membrane position. While it is difficult to discern between actual and estimated membrane position for all load and noise scenarios, this is in and of itself a result indicating the efficacy of the algorithm.

**Initial Conditions**

Initial conditions are chosen differently for the numerical tests than for the laboratory data experiment. For the simulations presented in this section, membrane initial position is taken to be a quadratic surface that achieves a maximum deformation of one cm and zero along the boundaries, with zero initial velocity. This corresponds to $g_1(x, y) = .01x(x - 1)y(y - 1)/.5^4$ and $g_2 = 0$ in equations 4.34 and 4.35, and for the load, a uniform estimate of 200 Pa pressure. These initial conditions are chosen to better approximate a membrane in steady-state (assuming presence of a load of similar magnitude), rather than starting from rest. Because each load experiment involves a different load distribution, there is a small oscillatory phase in the error, largely due to discrepancy between the initial condition and the actual steady-state position the membrane assumes due to the load. Initial conditions for the state estimate are identical to those of the simulated model.
6.4.1 Load One

The first and simplest load investigated is that of equation 6.47 (see figure 6.20). The estimated load distribution with 1% artificial noise is illustrated in figure 6.18 and 5% noise in figure 6.19. Explicitly satisfying the boundary conditions of the domain itself, load one reaches maximum pressure of 200 Pa in the center of the domain and explicitly satisfies the boundary conditions with no oscillations.

In comparing figure 6.18 to figure 6.19, it is readily apparent that, even with far more noise, the main features of the load are preserved. Note that the 1% noise case is nearly indiscernible from the exact load distribution. The 5% noise case is certainly a worse estimate of the load, however, the greatest magnitude is identified in the correction location (the very center of the domain), and it appears as if spurious load maxima occur in a pattern similar to the sensor layout itself.

![Figure 6.18: Estimated load distribution for load one, 1% noise](image)

![Figure 6.19: Estimated load distribution for load one, 5% noise](image)

![Figure 6.20: Exact load one: $P_z = 200 \sin(\pi x) \sin(\pi y)$](image)
Computing the $L^2$ relative load estimate error, illustrated in figures 6.21 - 6.22, characterizes how accurately the real load is estimated with only four cumulative stretch sensors. The initial value is large because of the load initial condition, a uniform field of magnitude 200 Pa, but decays to steady state error within $1/10$ of a second. This indicates successful load identification even in the presence of membrane vibrational dynamics. The final error is 0.0001343 which can be interpreted as .013% error in the estimate. This contrasts with the final $L^2$ error of 20% observed for the 5% noise case, illustrated in figure 6.22.

![Figure 6.21: $L^2$ relative load estimate error for load one, 1% noise](image)

![Figure 6.22: $L^2$ relative load estimate error for load one, 5% noise](image)

Similar to the load estimate error, the membrane position estimate error in time can be computed and is illustrated in figures 6.23 and 6.24. The position error for the 1% noise case achieves a steady state value of approximately 0.001%, contrasting to the 5% noise case that achieves a steady state error of .2%. The 1% noise case has similar effects, but they are not nearly as apparent as the 5% case. The 5% noise case exhibits a final error of approximately 21%, a surprisingly high result due to nonlinearity of the system and identification algorithm, not wholly unexpected.

Finally, the final position of the membrane for both noise cases is illustrated in figures 6.25 - 6.26. The final positions obviously reflect the load distribution itself. In
Figure 6.23: $L^2$ relative position estimate error for load one, 1% noise

Figure 6.24: $L^2$ relative position estimate error for load one, 5% noise

these plots, there are actually two surfaces, the exact and the estimated membrane position, but because the two are so close in value, they are visually indiscernible. Close inspection reveals slight mismatch between mesh lines, the cumulative effect of which is summarized in figures 6.23 - 6.24 as an integrated value.

Figure 6.25: Final membrane position for load one, 1% noise

Figure 6.26: Final membrane position for load one, 5% noise

For a simple load such as equation 6.47, the disturbance observer approach demonstrates highly accurate estimation capability. Such load estimates improve the esti-
mate of system state which would have been found by the standard Kalman filter approach. A second, slightly more complicated load is now investigated.

6.4.2 Load Two

The second load investigated is that of equation 6.48. This load achieves two maximum amplitudes and is illustrated in figure 6.29, along with the corresponding 1% and 5% noise cases, figures 6.27 and 6.28. Explicitly satisfying the boundary conditions of the domain itself, load two reaches maximum pressure of 200 Pa in the center of the domain.

The qualities of the exact distribution are reflected in the estimates, but there is a slight pressure distribution error at the center of the distribution, especially for the 5% noise case. It is important to remember that the stretch sensors can only return positive values. With that constraint, estimation of spatially oscillatory load distributions becomes more difficult. The algorithm must enforce load equilibrium, but such equilibrium is certainly not unique.

Again, the load estimation errors for each noise case are illustrated in figures 6.30 - 6.31. Due to the added complexity of the load, an increase in the estimate error is observed, the final value of which is approximately 12%. This contrasts with the surprising result that the final error for the 5% noise case is actually less, achieving a final value of 10%! This result is wholly unexpected, but likely a result of random chance, that is, the estimate is being forced in the right direction by the artificial noise itself. Overall, estimate errors increase in comparison to the previous load estimates, but this is somewhat expected due to the complexity of estimating a load that spatially varies with so few sensors. If more sensors are used, results similar to that of the consistency study summarized by figures 6.11 - 6.17 result.
The position error estimates for each noise case are illustrated in figures 6.32 - 6.33. Again, the errors for both cases increased, also experienced a sharp increase in magnitude with a steady-state value of approximately 0.25\% error for 1\% noise, and the 5\% noise case again demonstrated less error, with a final value of approximately 0.09\%. Such an error is small considering the scale of deformations the membrane is experiencing, $\approx 1$ cm peak deformation.

The final membrane surface is presented in figures 6.34 - 6.35. Again, the membrane deformation reflects the spatial distribution of the load, as expected. Also, again, the estimated and exact surfaces are so close in value that it is difficult to perceive a difference at the scale of the membrane deformation. However, if one looks closely at the plots, some error is perceptible as a slight offset in grid lines, especially near $(x, y) = (0.02, 0.02)$, with the offset for the 5\% noise case being even less apparent.
6.4.3 Load Three

Load three, equation 6.49 and illustrated in figure 6.38, differs from the previous loads in that it achieves a maximum value along the right boundary of the domain and thus does not satisfy the boundary conditions. Estimation of loads along edges is documented in the literature as being particularly difficult, as one must assume a different approximation space $V$ for the load and structural approximations, as in
equation 6.4. Here, again, it is assumed the loads satisfy the zero Dirichlet boundary condition as intuition would suggest real aerodynamic loads likely do.

The resulting load estimates, figures 6.36 - 6.37, present the boundary estimate issue mentioned in the previous paragraph as small, erroneous fluctuations in the estimate. Nonetheless, many of the important distributional features of the load are retained. The estimated load distribution exhibits greatest magnitude near the right boundary and decreases to zero along all edges save for the right boundary, along which the estimate exhibits the spurious behavior previously mentioned.

The $L^2$ load estimate errors for load three are illustrated in figures 6.39 - 6.40 and attain a steady-state error value of approximately 5.6% for 1% artificial noise and 5.4% for 5% noise. Surprisingly, this is a better estimate than the load two case, perhaps because, although boundary conditions are not satisfied by the load explicitly, the load itself is not spatially oscillatory to the degree of load two.

More interestingly are the position estimate errors for the load three case, illustrated in figures 6.41 - 6.42. As can be seen in the plot, the error oscillates with much greater magnitude than the previous load cases. The final error attained is .025 % for 1% noise and .12% for 5% noise. These results are surprising because, from relatively
close load estimates, fairly large differences in error arise for position estimates for the 1% and 5% noise cases. That being said, the errors are both fairly insignificant compared with the previous load and result in membrane position differences on the order of tenths of a millimeter.

The final estimated and exact surfaces of the membrane for each noise case are shown in figures 6.43 - 6.44. It is important to note that, even with fairly large oscillatory behavior in the load estimate along the right boundary of the membrane, the estimated surface is nonetheless smooth and, from this plot, it is impossible to discern between the estimated surface and the exact.

Although not addressed in this work, it is appropriate to note that the optimal regularization parameter chosen was not optimal for each noise case, but rather, was a value chosen for a noise-free case, thus superior results can be found if optimized with the knowledge of expected noise levels. Strictly speaking, the regularization
approach here is formulated to smooth the solution only as a result of the knowledge that realistic load distributions will likely be smooth, or at least not highly oscillatory. Thus, other possibilities for regularization exist and may offer improvements over the current results. Furthermore, the current regularized JEKF algorithm offers a transparent and simple means of introducing a variety of regularization approaches, expediting their respective study.

Also, the constraint that the load be quasi-static in nature can be relaxed at the cost of greater state space system dimensionality. However, it would be interesting to
apply model reduction techniques to the aforementioned system to alleviate the added computational burden. It may then be possible to execute the resulting estimation algorithm in real-time and estimate time-varying loads. Such an approach would greatly benefit current aerodynamic control systems.

6.5 Preliminary Experimental Results

For a final test of the algorithm developed in this chapter, a real membrane wing sample is constructed, a conceptual drawing of which is provided in figure 6.2. The study consists of collecting high frequency snapshots of the membrane wing during wind tunnel testing and executing the regularized JEKF algorithm directly using the noisy, laboratory data, which consists of strain information in approximately the same four regions, as used in the previous section.
6.5.1 Experimental Setup

The membrane wing frame was fabricated from steel and reinforced so that very little deflection occurred in the frame. The latex membrane was characterized in the laboratory, the parameters of which are identical to those used throughout the previous load estimation investigations. The latex membrane was pre-tensed to 5% prestrain. A photo of the speckled membrane being pre-tensed is provided in figure 6.45. The membrane was speckled so that a digital image correlation (DIC) could be used to determine the membrane deformation and strain fields in near real-time during the test. The damping parameters of the membrane were not characterized, but since the wind tunnel test was performed with the assumption of quasi-static loading and steady-state flow, damping was of little importance as the membrane was also assumed to have achieved steady-state position and strain. Furthermore, the goal was to identify the load distribution on the wing. Based on the sting balance measurements, the load resultant was steady.

![Figure 6.45: Wing test sample fabrication](image)

Photos of the laboratory experimental setup and the wing during wind tunnel testing are provided in figures 6.46 and 6.47. The AOS-manufactured high speed binocular camera system took photo snapshots at 300 Hz, and yielded a total of 1400
images (roughly 4.6 seconds), a sample size that demonstrated to be excessive as the values of strain were steady to allow swift convergence of identified loads. The cameras were positioned above the wing, on top of the wind tunnel, and were focused and calibrated through a plexiglass window so that full-field strain and displacement could be measured. The wing was tested at 18 m/s wind velocity with an AOA of 4 degrees.

After the experiment was finished, the digital images were downloaded to a PC and, using the VIC-3D 2009 software package, analyzed to yield deformation and strain field data. The strain and deformation data were then post-processed by an in-house custom MATLAB code that discards any erroneous measurements (generally indicated by the VIC software), and integrates the Lagrange strain data, computed by VIC-3D, over predetermined regions, such as those in figure 6.1 to emulate the existence of stretch (strain) sensors identical to those used in the previous section and derived in appendix B. Once the integrated strain fields were processed, the algorithm of the previous section could be applied directly to the resulting data. The resulting estimate of the load distribution was then integrated to enable a one-to-one comparison with sting balance data collected during the wind tunnel test.

Figure 6.46: Experimental wind tunnel test setup

Figure 6.47: Wing sample during testing
6.5.2 Estimation Results

Initial conditions for the estimated system were chosen by projecting the observed membrane deformation and treating the quasi-steady state membrane deformation observed in the wind tunnel as the initial condition, thus the initial velocity was taken to be zero. The initial estimated load was again assumed to have uniform value corresponding to a pressure of 200 Pa.

Figure 6.48 summarizes the estimated lift via integration of estimated load distribution, and the sting balance data. The sting balance data is displayed purely as a 95% confidence interval as the sting balance signal was not synchronized with the DIC system, thus forbidding a true comparison in time. Nevertheless, the estimated lift was within approximately 5% of the true lift measured.

![Figure 6.48: Time-varying lift resultant estimate](image)

The load distribution estimated from experimental strain data is illustrated in figure 6.49, with the resulting membrane deformation illustrated in figure 6.50. The load distribution is suspiciously localized in the center of the membrane. The author firmly believes that the explanation of this distribution, which differs from that expected by aerodynamic theory and experiment, is model mismatch between the
linear Poisson model and the real membrane, which was demonstrated to experience strains of the same order of magnitude as the pre-strain. Such similar strains indicate a linear model is not capable of capturing the dynamics of the membrane wing and nonlinear models should be investigated to extend these results.

Figure 6.49: Estimated load distribution on membrane wing

Figure 6.50: Estimated (mesh) and measured (interpolated surface) membrane deformation

The results presented here are nonetheless promising. The estimation algorithm demonstrated robustness to the unknown noise present in the measured data and provided a fairly accurate lift estimate. Such an estimate could easily be combined with other on-board sensors to improve the operation of a UAV or MAV craft. The results presented in this section, and the previous, strongly encourage further investigation and characterization of the approach to load estimation derived and presented in this chapter.
7 Conclusions

Biological inspiration has and will always be an integral component of engineering designs and approaches. However, just because nature arrived at a particular solution does not mean that solution is “optimal.” Engineers are free to pick and choose from biology in their attempts to offer solutions to problems.

Biological systems and engineering literature were reviewed as inspiration for the overall design, actuation, and sensors used in this work. Evolutionary biology’s myriad flight solutions include everything from birds with feathered wings that behave quasi-rigidly during flight but nonetheless allow for high degrees of morphing to bats with highly elastic membrane wings that allow for beneficially passive and aeroelastic flight effects. The engineering community is well aware of nature’s designs and their respective advantages, as they are readily reflected in engineering designs.

Investigations began with approximating inspiring birds’ and bats’ wings during gliding flight to plates and membranes, respectively. Orthotropic carbon fiber battens were the dominant component of the plate model, reinforcing a more flexible material. The membrane model was simply an isotropic membrane. Both models were derived from basic principles. Two types of sensors were also derived: piezocomposite, self-sensing actuators and intuitive stretch sensors.

The piezocomposite, self-sensing actuators were used in conjunction with linear quadratic control to morph a thin plate wing into three biologically inspired positions. Piezocomposites certainly provided the type of internally driven morphing that engineers abundantly agree upon as being the future of morphing wings. It is the author’s opinion that such actuators will likely continue to advance and be capable of use in such a high drive regime in the near future.
Having experimented with a morphing wing, the investigation turned to the obviously useful problem of estimating the load on a wing due to air pressure resultants caused by lift. A novel approach to load identification was proposed that accurately estimated three progressively more complex aerodynamic loads. Then applied to data collected during wind tunnel testing of a membrane wing of similar aspect ratio, it accurately estimated total lift to within 5% of sting balance measurements. Such an approach could be used to achieve a variety of ends and, most importantly, is necessary for a truly autonomous, morphing wing to adapt effectively to its flight environment in real time.

Both wing morphing, via smart materials and optimal control, and disturbance observer designs show promise for aerodynamic applications. Future work in wing morphing might include investigation of other actuator types and experimental validation of the approach used in this dissertation. Future work for this load identification approach should include a thorough analysis at the infinite-dimensional level. The approach should also be extended to time-varying disturbances, and solving the continuous version of the problem would be an interesting direction for future work as well.

The contributions of this work are summarized succinctly as follows:

- Derived and extended thin plate theory for anisotropic materials with distributed smart material sensors and actuators.
- Designed and wrote a versatile finite element code that admits any initial conditions, anisotropic material parameters, sensor and actuator positions, and external loading inputs, and solves for both open- and closed-loop system response for full state as well as partial state feedback linear quadratic control.
• While not discussed in this dissertation, the aforementioned finite element code has been and is being utilized in a variety of secondary projects and investigations, including model reduction, sensor placement, and system identification investigations.

• Derived a membrane and stretch sensor model from basic principles to investigate membrane wing dynamics.

• Wrote a second finite element code for the simulation of membrane wing dynamics in an attempt to estimation aerodynamic load inputs to a membrane wing system.

• Derived a novel load identification method for membrane wings with distributed stretch, or strain, sensors, and investigated its efficacy via numerical simulation and experimental wind tunnel tests.

Financial Support Acknowledgment

This work was supported in part by the Air Force Office of Scientific Research (AFOSR) under the Multidisciplinary University Research Initiative (MURI) grant FA9550-07-1-0540. Known informally as the “Bat MURI,” this project has greatly expanded our knowledge of bat structure and flight, simultaneously providing support to aspiring young researchers. It is a testimony to the success of AFOSR in providing funding that extends scientific knowledge, understanding, and technology, ensuring that the United States of America maintains its scientific prowess and leads the world into a thriving, safer and happier tomorrow.
Bibliography


APPENDICES
A. Derivation of Weak Form of an Anisotropic, Thin Plate Model

The derivation of the weak form of the thin plate model of chapter 4 proceeds as follows. First multiply equation 4.12 by test function $\phi(x, y)$ and integrate over the domain $\Omega$ of the plate, keeping all terms that could vary spatially under the integration for generality. Each term will be treated separately for clarity sake.

$$\int_{\Omega} \rho h \frac{\partial^2 w}{\partial t^2} \phi \, d\Omega + \int_{\Omega} \gamma \frac{\partial w}{\partial t} \phi \, d\Omega - \int_{\Omega} \frac{\partial^2 M_x}{\partial x^2} \phi \, d\Omega - \int_{\Omega} \frac{\partial^2 M_y}{\partial y^2} \phi \, d\Omega$$

$$- \int_{\Omega} \frac{\partial^2 M_{xy}}{\partial x \partial y} \phi \, d\Omega - \int_{\Omega} \frac{\partial^2 M_{yx}}{\partial y \partial x} \phi \, d\Omega = - \int_{\partial \Omega} f_n \phi \, d\Omega$$  \hspace{1cm} (1)

Integrating each term containing spatial derivatives using Green’s identity in component form (pg. 18 of [178]) yields

$$\int_{\Omega} u \frac{\partial v}{\partial x} = \int_{\partial \Omega} uv n_1 \, dx - \int_{\Omega} \frac{\partial u}{\partial x} v \, d\Omega$$

$$\int_{\Omega} u \frac{\partial v}{\partial y} = \int_{\partial \Omega} uv n_2 \, dy - \int_{\Omega} \frac{\partial u}{\partial y} v \, d\Omega,$$

where $n_1$ and $n_2$ are the first and second entries of the unit normal along boundary $\partial \Omega$. Applying these integration formulae twice to each term, one finds, for each term separately,

$$\int_{\Omega} \frac{\partial^2 M_x}{\partial x^2} \phi \, d\Omega = \int_{\partial \Omega} \frac{\partial M_x}{\partial x} \phi n_1 \, dx - \int_{\partial \Omega} M_x \frac{\partial \phi}{\partial x} n_1 \, dx + \int_{\Omega} M_x \frac{\partial^2 \phi}{\partial x^2} \, d\Omega$$  \hspace{1cm} (2)

$$\int_{\Omega} \frac{\partial^2 M_y}{\partial y^2} \phi \, d\Omega = \int_{\partial \Omega} \frac{\partial M_y}{\partial y} \phi n_2 \, dy - \int_{\partial \Omega} M_y \frac{\partial \phi}{\partial y} n_2 \, dy + \int_{\Omega} M_y \frac{\partial^2 \phi}{\partial y^2} \, d\Omega$$  \hspace{1cm} (3)
\[
\begin{align*}
\int_{\Omega} \frac{\partial^2 M_{xy}}{\partial x \partial y} \phi \partial \Omega &= \int_{\partial \Omega} \frac{\partial M_{xy}}{\partial y} \phi n_1 dx - \int_{\partial \Omega} M_{xy} \frac{\partial \phi}{\partial x} n_2 dy + \int_{\Omega} M_{xy} \frac{\partial^2 \phi}{\partial y \partial x} \partial \Omega \\
\int_{\Omega} \frac{\partial^2 M_{yx}}{\partial x \partial y} \phi \partial \Omega &= \int_{\partial \Omega} \frac{\partial M_{yx}}{\partial x} \phi n_2 dy - \int_{\partial \Omega} M_{yx} \frac{\partial \phi}{\partial y} n_1 dx + \int_{\Omega} M_{yx} \frac{\partial^2 \phi}{\partial x \partial y} \partial \Omega.
\end{align*}
\]

Including boundary terms grouped by unit normal vector entries \(n_1\) and \(n_2\), the weak form then becomes

\[
\int_{\Omega} \left( \rho \frac{\partial^2 w}{\partial t^2} + \gamma \frac{\partial w}{\partial t} + M_x \frac{\partial^2 \phi}{\partial x^2} + M_y \frac{\partial^2 \phi}{\partial y^2} + M_{xy} \frac{\partial^2 \phi}{\partial x \partial y} + M_{yx} \frac{\partial^2 \phi}{\partial y \partial x} + f_n \phi \right) \partial \Omega \\
+ \int_{\Gamma} \left( - \frac{\partial M_x}{\partial x} \phi n_1 - M_x \frac{\partial \phi}{\partial x} n_1 - M_y \frac{\partial \phi}{\partial y} n_1 + \frac{\partial M_{xy}}{\partial y} \phi n_1 \right) \partial \Gamma \\
+ \int_{\Gamma} \left( \frac{\partial M_y}{\partial y} \phi n_2 - M_y \frac{\partial \phi}{\partial y} n_2 - M_{xy} \frac{\partial \phi}{\partial x} n_2 + \frac{\partial M_{yx}}{\partial x} \phi n_2 \right) \partial \Gamma = 0,
\]

where \(\Gamma \equiv \partial \Omega\). Boundary conditions are now applied such that the above boundary terms are eliminated. A proper space \(V\) is chosen to enforce the essential boundary conditions while the natural boundary conditions are simply matched using the above equation and setting to zero, simplifying the above.

The weak form above is now tailored specifically to the boundary conditions used to emulate a wing in chapter 5, that is, a clamped edge to represent the attachment to an aircraft fuselage and free boundaries otherwise.

Take \(\phi\) in the space of test functions

\[
V = \{ \phi \in H^2(\Omega) \mid \phi(x, L) = \phi_y(x, L) = 0 \text{ for } 0 \leq x \leq W \},
\]

where \(L\) and \(W\) are the length and width of the plate, respectively. This choice ensures the boundary condition along clamped edge \(\Gamma_3\) is satisfied by the basis functions.
In extracting the boundary terms from the weak form above, take note of the unit normal vector entries associated with each term. Along boundaries that are tangent to the y-axis, \( n_2 = 0 \), and, likewise for boundaries tangent to the x-axis, \( n_1 = 0 \). Also note, however, that the right-hand rule is used in the development of these equations, therefore, for instance, the unit normal along \( \Gamma_1 \), that is, for the boundary \( y = 0, 0 \leq x \leq W \), the unit normal entry \( n_2 = -1 \). Thus, the extracted boundary conditions become

\[
\int_{\Gamma} \left( -\frac{\partial M_x}{\partial x} \phi n_1 - M_x \frac{\partial \phi}{\partial x} n_1 - M_{yx} \frac{\partial \phi}{\partial y} n_1 + \frac{\partial M_{xy}}{\partial y} \phi n_1 \right) \partial \Gamma \\
+ \int_{\Gamma} \left( \frac{\partial M_y}{\partial y} \phi n_2 - M_y \frac{\partial \phi}{\partial y} n_2 - M_{xy} \frac{\partial \phi}{\partial x} n_2 + \frac{\partial M_{yx}}{\partial x} \phi n_2 \right) \partial \Gamma = 0.
\]

It is easiest and intuitive to deal with each boundary separately. Consider boundary \( \Gamma_1 \), which is the wing-tip edge. As mentioned before, \( n_2 = -1 \) along this boundary and \( n_1 = 0 \). Extracting only the boundary terms which are nonzero,

\[
- \int_{\Gamma_1} \left( \frac{\partial M_y}{\partial y} \phi - M_y \frac{\partial \phi}{\partial y} - M_{xy} \frac{\partial \phi}{\partial x} + \frac{\partial M_{yx}}{\partial x} \phi \right) \partial \Gamma = 0. \tag{6}
\]

Since this is a free edge, it must satisfy the boundary conditions derived in chapter 4, stating zero shear force and moment along the boundary. The conditions are repeated here for convenience:

\[
Q_y + \frac{\partial M_{yx}}{\partial x} = 0 \\
M_y = 0 \\
M_{xy}|_{\partial \Gamma} = 0 \tag{7}
\]
with

\[ Q_y = \frac{\partial M_y}{\partial y} + \frac{\partial M_{xy}}{\partial x}. \]

Immediately, the boundary conditions are satisfied if the term \( M_{xy} \frac{\partial \phi}{\partial x} \) is integrated by parts once, resulting in the point-wise evaluation of \( M_{xy} \) at \( \partial \Gamma_1 \), i.e.

\[ \int_{\Gamma_1} M_{xy} \frac{\partial \phi}{\partial x} \, dx = M_{xy} \mid_{\partial \Gamma} - \int_{\Gamma_1} \frac{\partial M_{xy}}{\partial x} \phi \partial \Omega. \]  

(8)

Substitution of equation 8 into equation 6 yields

\[ - \int_{\Gamma_1} \left( \frac{\partial M_y}{\partial y} \phi - M_y \frac{\partial \phi}{\partial y} - \frac{\partial M_{xy}}{\partial x} \phi + \frac{\partial M_{yx}}{\partial x} \phi \right) \partial \Gamma - M_{xy} \mid_{\partial \Gamma} = 0, \]

in which it is obvious that the terms can be grouped by \( \phi \) and its derivatives to satisfy the boundary conditions in equation 7.

Performing similar operations for all of the boundaries, all boundary conditions are used, leaving only

\[ \int_{\Omega} \left( \rho h \frac{\partial^2 w}{\partial t^2} \phi + \gamma \frac{\partial w}{\partial t} \phi + M_x \frac{\partial^2 \phi}{\partial x^2} + M_y \frac{\partial^2 \phi}{\partial y^2} + 2 M_{xy} \frac{\partial^2 \phi}{\partial x \partial y} + f_n \phi \right) \partial \Omega = 0, \]  

(9)

where \( M_{xy} \) and \( M_{yx} \) are combined in accordance with the results of chapter 4.
B. Derivation of Weak Form of a Linear Membrane Model

The derivation of the weak form of the linear membrane model of chapter 4 proceeds as follows. Beginning with the strong form of the membrane model

$$\rho h \frac{\partial^2 w}{\partial t^2} + \gamma \frac{\partial w}{\partial t} - \eta \left( \frac{\partial^2 w}{\partial t \partial x^2} + \frac{\partial^2 w}{\partial t \partial y^2} \right) - T_x \frac{\partial^2 w}{\partial x^2} - T_y \frac{\partial^2 w}{\partial y^2} = P_z,$$

denote the domain of the membrane by $\Omega$ and multiply the above by a test function $\phi$. Now, integration over $\Omega$, yields the weak form

$$\int_{\Omega} \rho h \frac{\partial^2 w}{\partial t^2} \phi \, d\Omega + \int_{\Omega} \gamma \frac{\partial w}{\partial t} \phi \, d\Omega - \int_{\Omega} \eta \left( \frac{\partial^2 w}{\partial t \partial x^2} + \frac{\partial^2 w}{\partial t \partial y^2} \right) \phi \, d\Omega$$

$$- \int_{\Omega} T_x \frac{\partial^2 w}{\partial x^2} \phi \, d\Omega - \int_{\Omega} T_y \frac{\partial^2 w}{\partial y^2} \phi \, d\Omega = \int_{\Omega} P_z \phi \, d\Omega.$$

The second derivative terms can be integrated using Green’s identity in component form (pg. 18 of [178]), yielding

$$\int_{\Omega} T_x \frac{\partial^2 w}{\partial x^2} \phi \, d\Omega = \int_{\partial \Omega} T_x \frac{\partial w}{\partial x} \phi n_x \, dx + \int_{\Omega} T_x \frac{\partial w}{\partial x} \frac{\partial \phi}{\partial x} \, d\Omega - \int_{\Omega} T_x \frac{\partial w}{\partial x} \phi \, d\Omega$$

$$\int_{\Omega} T_y \frac{\partial^2 w}{\partial y^2} \phi \, d\Omega = \int_{\partial \Omega} T_y \frac{\partial w}{\partial y} \phi n_y \, dy + \int_{\Omega} T_y \frac{\partial w}{\partial y} \frac{\partial \phi}{\partial y} \, d\Omega - \int_{\Omega} T_y \frac{\partial w}{\partial y} \phi \, d\Omega,$$

where $n_x$ and $n_y$ denote the first and second entries of the normal vector along the boundary of the domain, $\partial \Omega$. Take $\phi \in V = \{ \phi \in H^1(\Omega) \mid v(\partial \Omega) = 0 \}$ such that Dirichlet boundary conditions are satisfied by test function $\phi$. 
The weak form then becomes, after the above integrations,

\[
\rho h \int_{\Omega} \frac{\partial^2 w}{\partial t^2} \phi d\Omega + \gamma \int_{\Omega} \frac{\partial w}{\partial t} \phi d\Omega - \eta \int_{\Omega} \left( \frac{\partial^2 w}{\partial t \partial x} \frac{\partial \phi}{\partial x} + \frac{\partial^2 w}{\partial t \partial y} \frac{\partial \phi}{\partial y} \right) d\Omega \\
+ T_x \int_{\Omega} \frac{\partial w}{\partial x} \frac{\partial \phi}{\partial x} d\Omega + T_y \int_{\Omega} \frac{\partial w}{\partial y} \frac{\partial \phi}{\partial y} d\Omega + \int_{\Omega} P_z \phi d\Omega = 0. \quad (10)
\]

B.1 Finite Element Approximation of Stretch Sensor Model

Recall equation 4.41, repeated here for convenience,

\[
S = \alpha \int_{\Omega_s} \left( \frac{\partial w}{\partial x} \right)^2 + \left( \frac{\partial w}{\partial y} \right)^2 d\Omega_s. \quad (11)
\]

Following a similar procedure as the finite element derivation in chapter 6, make the substitutions

\[
\frac{\partial w}{\partial x} \approx \sum_{i=1}^{N} w_i \frac{\partial \phi_i}{\partial x}, \\
\frac{\partial w}{\partial y} \approx \sum_{i=1}^{N} w_i \frac{\partial \phi_i}{\partial y},
\]

written instead as the vector inner products

\[
\frac{\partial w}{\partial x} \approx w^T \phi_x, \\
\frac{\partial w}{\partial y} \approx w^T \phi_y,
\]
where $\phi_x$ and $\phi_y$ denote the vector of basis function derivatives in $x$ or $y$, respectively, and $w$ represents the vector of basis function coefficients. Using this notation, substitute the above into equation 4.41 to yield

$$S = \frac{\alpha}{2A_0} \int_{\Omega_s} \left( w^T \phi_x \right)^2 + \left( w^T \phi_y \right)^2 \, d\Omega_s.$$
variables \( x, y \),

\[
S = \frac{\alpha}{2A_s} w^T (M_x + M_y) w
\]
\[
= \frac{\alpha}{2A_s} w^T (M_s) w,
\]

where \( M_x, M_y \) and \( M_s \) are defined as

\[
M_x = \begin{bmatrix}
\int \phi_1^1 \phi_1^1 & \int \phi_1^1 \phi_2^1 & \cdots & \int \phi_1^1 \phi_N^1 \\
\int \phi_2^1 \phi_1^1 & \int \phi_2^1 \phi_2^1 & & \\
& \ddots & & \\
& & \ddots & \\
\int \phi_N^1 \phi_1^1 & \cdots & \cdots & \int \phi_N^1 \phi_N^1
\end{bmatrix}
\]
\[
M_y = \begin{bmatrix}
\int \phi_1^1 \phi_1^1 & \int \phi_1^1 \phi_2^1 & \cdots & \int \phi_1^1 \phi_N^1 \\
\int \phi_2^1 \phi_1^1 & \int \phi_2^1 \phi_2^1 & & \\
& \ddots & & \\
& & \ddots & \\
\int \phi_N^1 \phi_1^1 & \cdots & \cdots & \int \phi_N^1 \phi_N^1
\end{bmatrix}
\]
\[
M_s = M_x + M_y.
\]

The above leads to the nonlinear measurement model for a finite element simulation containing a single stretch sensor:

\[
y = x^T C_s x,
\]

(12)

where the finite element coefficient state vector \( x \in R^N, y \in R^p \), and \( C_s \) is a \( N \times N \) observation matrix. For a single stretch measurement, \( C_s = 2A_s M_s \), where \( 1 \leq S \leq P \) for \( P \) sensors.
Generalizing the above to multiple sensors yields

\[
\begin{align*}
y_1 &= x^T C_1 x \\
y_2 &= x^T C_2 x \\
\vdots \\
y_P &= x^T C_P x,
\end{align*}
\]

where the \( C_1, C_2, \ldots, C_p \) individually take a form as in equation 12, each matrix constructed individually for each stretch sensor.

The model described above might seem prohibitively expensive in terms of computational power, especially if many sensors are used. In actuality, however, the matrices involved are sparse.

Because this model is nonlinear, one can expect the use of the gradient for computation. Computing the gradient for this sensor model is simplified if one solves for a single measurement, then generalizes the form of the result to \( P \) measurements. Consider a single measurement \( y_i = \frac{1}{2} x^T C_i x \). By elementary matrix calculus,

\[
\frac{\partial y}{\partial z} = \frac{1}{2} (C_i + C_i^T) x = C_i x,
\]

if \( C_i \) is symmetric, which is indeed the case here. The above calculation obviously results in a column vector of size \( N \times 1 \). Taking the transpose of the resulting vector, place the result in true gradient form for a multi-sensor model, i.e.

\[
(C_i x)^T = \begin{bmatrix} \frac{\partial y_1}{\partial x_1} & \frac{\partial y_1}{\partial x_2} & \cdots & \frac{\partial y_1}{\partial x_N} \end{bmatrix}.
\]
Generalizing to multiple sensors, i.e. multiple $y_i$s,

$$\left( C_i x \right)^T = \begin{bmatrix} \frac{\partial y_1}{\partial x_1} & \frac{\partial y_1}{\partial x_2} & \cdots & \frac{\partial y_1}{\partial x_N} \\ \frac{\partial y_2}{\partial x_1} & \frac{\partial y_2}{\partial x_2} & \cdots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial y_P}{\partial x_1} & \cdots & \cdots & \frac{\partial y_P}{\partial x_N} \end{bmatrix},$$

which is a $P \times N$ Jacobian matrix for $P$ measurements and $N$ system states. Thus, one can form the Jacobian by evaluating matrix-vector multiplications and assigning the result to an array. The gradient above becomes important for the JEEK utilized in chapter 6.