DATA-DRIVEN SYSTEM IDENTIFICATION OF NONLINEAR DYNAMICS FOR A 6-DOF AIRCRAFT MODEL USING SINDYC

A Thesis by

Thomas W. Thibodeaux IV

Bachelor of Science, University of Nebraska – Lincoln, 2012

Submitted to the Department of Aerospace Engineering and the faculty of the Graduate School of Wichita State University in partial fulfillment of the requirements for the degree of Master of Science

December 2019
The following faculty members have examined the final copy of this thesis for form and content, and recommend that it be accepted in partial fulfillment of the requirement for the degree of Master of Science, with a major in Aerospace Engineering.

James Steck, Committee Chair

Atri Dutta, Committee Member

Yimesker Yihun, Committee Member
DEDICATION

Mom and Dad,
Patrick and Megan,
dear friends,

I couldn’t have done it without you.
ACKNOWLEDGMENTS

I would like to thank my professor and committee chair, Dr. James Steck, whose invaluable instruction and guidance have made me a stronger researcher and engineer. I am sincerely grateful for the opportunity. I must also express my appreciation to the other members of my committee, Dr. Atri Dutta and Dr. Yimesker Yihun, for their time and input.

This research would not have been possible without the foundation built by Steven L. Brunton, Joshua L. Proctor, and Nathan Kutz, and extended by Eurika Kaiser. Thank you for developing a framework of such elegance, I hope I have done it justice.

I would like to thank my colleagues working in the General Aviation Flight Lab for their fellowship and counsel as I completed my coursework and conducted this research – I sincerely wish each of you success. In particular, I must thank Alok Menon, with whom I would often discuss facets of this research and contemporary challenges within flight control.

I want to thank my mother and father whose confidence that I could accomplish this goal was overshadowed only by their love and support along the way. I must give special thanks to my brother Patrick and sister-in-law Megan whose service to this country has, in no small part, inspired me to boldly pursue my own ambitions.

I cannot forget my dear friends, whose encouragement and camaraderie has kept me firmly grounded throughout my return to school. For each of you, I am most fortunate.
ABSTRACT

This work employs a data-driven, system identification technique to determine the governing dynamic equations of a nonlinear, 6-DOF aircraft model. A generalized aircraft model is developed, and nonlinearities are introduced in the gravity model, the aerodynamic force and moment models, and the coupled dynamic and kinematic models. As a case study, the constant parameters within the generalized model are chosen to closely match those of the SIAI Marchetti S-211, an Italian jet trainer. The model is trimmed and, to establish confidence, the dynamics are excited using both nonzero initial conditions and control inputs.

The nonlinear system identification technique, SINDYc (Sparse Identification of Nonlinear Dynamics with Control), is formally introduced, a thorough explanation is provided, and a simple example is conducted. SINDYc is a sparse regression technique that uses a library of candidate functions, composed of state and input variables, to determine the fewest number of terms required to represent the set of nonlinear differential equations which govern dynamic system behavior.

Using the three control inputs – aileron, elevator, and rudder – the S-211 model is aggressively excited as to sufficiently express its nonlinearities, and the resulting time histories of each state are recorded. This data is used with the SINDYc algorithm to reconstruct the nonlinear dynamic equations. Several iterations are performed with variations in the type of state noise and filtering, the method of numerical differentiation, and constraints imposed upon the library of candidate functions. In most cases, SINDYc is able to determine the terms present in the nonlinear dynamic equations with reasonable accuracy.

Finally, the identified systems are excited using simple control inputs, and their dynamic response is compared to that of the true system.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Chapter</th>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>1.1</td>
<td>Current Standards and Challenges of Aircraft Control</td>
<td>1</td>
</tr>
<tr>
<td>1.</td>
<td>1.2</td>
<td>Nonlinear Model Derivation and Nonlinear Control Methods</td>
<td>3</td>
</tr>
<tr>
<td>1.</td>
<td>1.3</td>
<td>Data-Driven System Identification and the SINDy Algorithm</td>
<td>4</td>
</tr>
<tr>
<td>2.</td>
<td>2.1</td>
<td>Development of Equations of Motion</td>
<td>8</td>
</tr>
<tr>
<td>2.</td>
<td>2.1.1</td>
<td>General Equations of Unsteady Aircraft Motion</td>
<td>8</td>
</tr>
<tr>
<td>2.</td>
<td>2.1.2</td>
<td>Dynamics in Nonlinear State-Space Form</td>
<td>9</td>
</tr>
<tr>
<td>2.</td>
<td>2.1.3</td>
<td>Transformation to Aerodynamic Coordinates</td>
<td>10</td>
</tr>
<tr>
<td>2.</td>
<td>2.2</td>
<td>Development of Aerodynamic Model</td>
<td>11</td>
</tr>
<tr>
<td>2.</td>
<td>2.2.1</td>
<td>Gravity Model</td>
<td>11</td>
</tr>
<tr>
<td>2.</td>
<td>2.2.2</td>
<td>Lift Model</td>
<td>12</td>
</tr>
<tr>
<td>2.</td>
<td>2.2.3</td>
<td>Downwash Model</td>
<td>13</td>
</tr>
<tr>
<td>2.</td>
<td>2.2.4</td>
<td>Drag Model</td>
<td>14</td>
</tr>
<tr>
<td>2.</td>
<td>2.2.5</td>
<td>Side Force Model</td>
<td>14</td>
</tr>
<tr>
<td>2.</td>
<td>2.3</td>
<td>Development of Force and Moment Equations</td>
<td>15</td>
</tr>
<tr>
<td>2.</td>
<td>2.3.1</td>
<td>Forces in X-Direction</td>
<td>15</td>
</tr>
<tr>
<td>2.</td>
<td>2.3.2</td>
<td>Forces in Y-Direction</td>
<td>16</td>
</tr>
<tr>
<td>2.</td>
<td>2.3.3</td>
<td>Forces in Z-Direction</td>
<td>16</td>
</tr>
<tr>
<td>2.</td>
<td>2.3.4</td>
<td>Moment About Roll Axis</td>
<td>17</td>
</tr>
<tr>
<td>2.</td>
<td>2.3.5</td>
<td>Moment About Pitch Axis</td>
<td>18</td>
</tr>
<tr>
<td>2.</td>
<td>2.3.6</td>
<td>Moment About Yaw Axis</td>
<td>19</td>
</tr>
<tr>
<td>2.</td>
<td>2.4</td>
<td>Aircraft Data – SIAI Marchetti S-211</td>
<td>19</td>
</tr>
<tr>
<td>2.</td>
<td>2.5</td>
<td>Collection of Nonlinear State Equations</td>
<td>19</td>
</tr>
<tr>
<td>2.</td>
<td>2.5.1</td>
<td>Establishing Trim</td>
<td>22</td>
</tr>
<tr>
<td>2.</td>
<td>2.6</td>
<td>Dynamic Model Excitation and Input Response</td>
<td>25</td>
</tr>
<tr>
<td>3.</td>
<td>3.1</td>
<td>Generalized Nonlinear Dynamics</td>
<td>31</td>
</tr>
<tr>
<td>3.</td>
<td>3.2</td>
<td>Data-Driven System Identification with SINDy</td>
<td>32</td>
</tr>
<tr>
<td>3.</td>
<td>3.2.1</td>
<td>Data Library Normalization</td>
<td>35</td>
</tr>
<tr>
<td>3.</td>
<td>3.2.2</td>
<td>The Sparsity-Promoting Parameter</td>
<td>36</td>
</tr>
<tr>
<td>3.</td>
<td>3.3</td>
<td>A Simple Example</td>
<td>37</td>
</tr>
</tbody>
</table>
# TABLE OF CONTENTS (continued)

<table>
<thead>
<tr>
<th>Chapter</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4. AIRCRAFT SYSTEM IDENTIFICATION USING SINDYC</td>
<td>42</td>
</tr>
<tr>
<td>4.1 System Inputs</td>
<td>43</td>
</tr>
<tr>
<td>4.2 System ID – No Noise, Exact Derivatives</td>
<td>45</td>
</tr>
<tr>
<td>4.3 System ID – No Noise, Central Differences</td>
<td>49</td>
</tr>
<tr>
<td>4.4 System ID – White Noise, Exact Derivatives</td>
<td>52</td>
</tr>
<tr>
<td>4.5 System ID – White Noise, Total Variation Regularized Differentiation</td>
<td>55</td>
</tr>
<tr>
<td>4.5.1 State De-Noising</td>
<td>55</td>
</tr>
<tr>
<td>4.5.2 Total Variation Regularized Differentiation</td>
<td>57</td>
</tr>
<tr>
<td>4.5.3 SINDYc Results – Full, Unconstrained Library</td>
<td>59</td>
</tr>
<tr>
<td>4.5.4 SINDYc Results – Unconstrained Library of Degree 2 Terms</td>
<td>59</td>
</tr>
<tr>
<td>4.5.5 SINDYc Results – Fully Constrained Library</td>
<td>63</td>
</tr>
<tr>
<td>4.6 Summary of System ID Results</td>
<td>66</td>
</tr>
<tr>
<td>4.7 Excitation of Identified Systems</td>
<td>66</td>
</tr>
<tr>
<td>5. DISCUSSION</td>
<td>71</td>
</tr>
<tr>
<td>5.1 Noise and State Precision</td>
<td>71</td>
</tr>
<tr>
<td>5.2 Applicability of a Fully Constrained Library</td>
<td>72</td>
</tr>
<tr>
<td>5.3 Significance of Identified Model Accuracy</td>
<td>73</td>
</tr>
<tr>
<td>6. FURTHER INVESTIGATION</td>
<td>74</td>
</tr>
<tr>
<td>6.1 Potential Model Improvements</td>
<td>74</td>
</tr>
<tr>
<td>6.2 Control Input Optimization</td>
<td>74</td>
</tr>
<tr>
<td>6.3 Nonlinear Control Framework</td>
<td>75</td>
</tr>
<tr>
<td>6.4 Implementation on a Flight Simulator</td>
<td>75</td>
</tr>
<tr>
<td>BIBLIOGRAPHY</td>
<td>76</td>
</tr>
<tr>
<td>APPENDICES</td>
<td>79</td>
</tr>
<tr>
<td>A. ERRORS ASSOCIATED WITH TRIGONOMETRIC APPROXIMATIONS</td>
<td>80</td>
</tr>
<tr>
<td>B. DERIVATION OF KINEMATIC EQUATIONS OF MOTION</td>
<td>81</td>
</tr>
<tr>
<td>C. TRANSFORMATION INTO AERODYNAMIC VARIABLES</td>
<td>82</td>
</tr>
<tr>
<td>D. DERIVATION OF AERODYNAMIC FORCE EQUATIONS</td>
<td>85</td>
</tr>
<tr>
<td>Chapter</td>
<td>Title</td>
</tr>
<tr>
<td>---------</td>
<td>-------</td>
</tr>
<tr>
<td>E.</td>
<td>DERIVATION OF AERODYNAMIC MOMENT EQUATIONS</td>
</tr>
<tr>
<td>F.</td>
<td>POLHAMUS FORMULA CALCULATIONS</td>
</tr>
<tr>
<td>G.</td>
<td>CALCULATION OF CERTAIN AERODYNAMIC COEFFICIENTS</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>SIAI-Marchetti S-211 In Flight</td>
<td>20</td>
</tr>
<tr>
<td>2.2</td>
<td>S-211, Initial Roll Rate Response, ( p(0) = 4.0 , \text{r/s} )</td>
<td>26</td>
</tr>
<tr>
<td>2.3</td>
<td>S-211, Initial Pitch Rate Response, ( q(0) = 4.0 , \text{r/s} )</td>
<td>26</td>
</tr>
<tr>
<td>2.4</td>
<td>S-211, Initial Yaw Rate Response, ( r(0) = 2.0 , \text{r/s} )</td>
<td>27</td>
</tr>
<tr>
<td>2.5</td>
<td>S-211, Response to Aileron Doublet</td>
<td>28</td>
</tr>
<tr>
<td>2.6</td>
<td>S-211, Response to Elevator Doublet</td>
<td>28</td>
</tr>
<tr>
<td>2.7</td>
<td>S-211, Response to Rudder Doublet</td>
<td>29</td>
</tr>
<tr>
<td>3.1</td>
<td>Dynamic Response of Simple System</td>
<td>38</td>
</tr>
<tr>
<td>4.1</td>
<td>Control Inputs for S-211 System Identification</td>
<td>44</td>
</tr>
<tr>
<td>4.2</td>
<td>S-211 Dynamic System Response, No Noise</td>
<td>45</td>
</tr>
<tr>
<td>4.3</td>
<td>S-211 Dynamic System Response, 20% ( \sigma ) Noise</td>
<td>52</td>
</tr>
<tr>
<td>4.4</td>
<td>State Error Following Low-Pass Filtering</td>
<td>56</td>
</tr>
<tr>
<td>4.5</td>
<td>State Derivatives, Computed Using Regularized Differentiation</td>
<td>57</td>
</tr>
<tr>
<td>4.6</td>
<td>State Derivative Error Following Regularized Differentiation</td>
<td>58</td>
</tr>
<tr>
<td>4.7</td>
<td>Identified Systems, Response to Aileron Doublet</td>
<td>67</td>
</tr>
<tr>
<td>4.8</td>
<td>Identified Systems, Response to Elevator Doublet</td>
<td>68</td>
</tr>
<tr>
<td>4.9</td>
<td>Identified Systems, Response to Rudder Doublet</td>
<td>69</td>
</tr>
<tr>
<td>A.1</td>
<td>Trigonometric Approximation Errors</td>
<td>80</td>
</tr>
<tr>
<td>G.1</td>
<td>Effect of Cubic Terms on Aerodynamic Coefficients</td>
<td>93</td>
</tr>
</tbody>
</table>
# LIST OF TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>S-211 Mass Properties</td>
<td>20</td>
</tr>
<tr>
<td>2.2</td>
<td>S-211 Geometric Parameters</td>
<td>21</td>
</tr>
<tr>
<td>2.3</td>
<td>S-211 Trim Conditions</td>
<td>21</td>
</tr>
<tr>
<td>2.4</td>
<td>S-211 Aerodynamic Coefficients</td>
<td>22</td>
</tr>
<tr>
<td>2.5</td>
<td>S-211 Nominal Dynamic Modes at Trim</td>
<td>30</td>
</tr>
<tr>
<td>4.1</td>
<td>Summary of System Identification Results</td>
<td>66</td>
</tr>
<tr>
<td>5.1</td>
<td>Restatement of System Identification Cases</td>
<td>71</td>
</tr>
<tr>
<td>F.1</td>
<td>S-211 Aerodynamic Surface Parameters</td>
<td>90</td>
</tr>
<tr>
<td>G.1</td>
<td>Collection of Cubic Term Coefficients</td>
<td>93</td>
</tr>
<tr>
<td>G.2</td>
<td>Parameters for Derivation of $C_{Y_{ub\beta}}$</td>
<td>94</td>
</tr>
</tbody>
</table>
CHAPTER 1
BACKGROUND AND MOTIVATION

1.1 Current Standards and Challenges of Aircraft Control

Modern aircraft are highly nonlinear dynamical systems that require sophisticated means to establish and maintain safe and controlled flight. For much of the 20th century, humans exclusively served the role of flight controller. Human pilots are able to learn the natural behavior of an aircraft and develop intuition for the dynamic response to flight control input. However, as aircraft become larger, more maneuverable, and designed with narrowing stability margins, they become more difficult to control. These aircraft require advance digital flight control frameworks, working in concert with the pilot, to ensure stable, reliable, and efficient flight characteristics. To accomplish this, a mathematical model of the natural behaviors, response to actuation, and method of controller synthesis is required. An aircraft’s fly-by-wire system represents the real world implementation of this control system.

Due to the availability of design and analysis tools and the deterministic nature of linear systems, virtually all fly-by-wire systems are designed and evaluated using the methods of linear control theory. This is often accomplished by linearizing the aircraft system about a multitude of trim points throughout its flight envelope and under varying aircraft configurations. A linear controller is synthesized at each equilibrium point, each controller applicable to a narrow region of flight conditions. This process, called gain scheduling, is a widely accepted method of nonlinear control and is used successfully across a range of industries and applications, including aircraft fly-by-wire systems. Leith et al. provides a comprehensive survey of current methods in gain scheduling for generalized nonlinear control systems [1].

The application of gain scheduling on aircraft control systems can be likened to identifying hundreds of dynamic systems and designing hundreds of associated controllers, each for a specific aircraft configuration at a specific condition – of course, in reality, there exists one nonlinear, time-varying system. The system identification of each linear model and the development of each controller must be accomplished prior to implementation, or ‘off-line.’ While this method is widely
accepted and used exclusively to certify commercial fly-by-wire systems in the United States, it may not capture all nonlinear behaviors – near stall, for example. Additionally, once designed, this method is static, incapable of adaptation to unfamiliar conditions.

This presents a potential problem. If the fly-by-wire system is designed to accommodate hundreds of narrow aircraft models and corresponding control schemes, how can the controller manage a dynamic system with model anomalies for which it has never been designed to control? This unfamiliar system may result from sudden failure or hazardous malfunction of its control systems. For example, a flight control surface may jam at a nonzero deflection, inducing a constant, uncommanded control input. Such an unfamiliar system may also result from major damage, the consequence of an uncontained engine failure or a combat encounter with anti-aircraft defenses. Even a dramatic, unexpected shift in aircraft weight distribution could result in a dynamic system not considered during design of the fly-by-wire system.

In order to overcome this potential problem, a solution needs to accomplish two things. First, a means to rapidly determine, in situ, a reliable dynamic model of the anomalous aircraft system is required. This model should also capture nonlinear dynamic behavior. Additionally, a method of nonlinear controller synthesis is necessary to condition the stability and control behaviors of the system. Control methodologies featuring these qualities are found in the toolbox of adaptive control theory, allowing control of systems of changing or initially unknown dynamics.

Unfortunately, adaptive control methods are not currently accepted for use on certified aircraft. While modern commercial aircraft do employ gain scheduling as a means to adapt controllers to a constantly changing system, Jacklin discusses the current obstacles standing between sophisticated adaptive controllers and their certification on commercial aircraft [2]. The benefits of an adaptive controller include allowing the aircraft to adjust its control gains to compensate for malfunction, damage, or some other unforeseen upset. However, quantifying and evaluating the performance of adaptive control systems is a significant challenge. Specifically, of the 25 criteria in FAA flight control certification guidelines, six are more difficult for adaptive control systems, and all of those six involve software requirement definitions and performance verification. In terms
of current gaps in research, Jacklin identifies four shortcomings. First, there exists a gap in controller requirements, as there is currently not well-accepted performance criteria. There is a gap in simulation models, as current models may not exhaustively capture aircraft behavior in all possible flight scenarios. Third, a gap exists in proving adaptive stability and convergence, as existing techniques, such as Lyapunov criteria, are less deterministic than stability criteria for linear control systems. Finally, there is a gap in online monitoring tools, as the real-time health monitoring of an adaptive control system is still in its infancy.

This thesis will present a method of nonlinear system identification using a 6-DOF aircraft model, nonlinear in both dynamics and aerodynamics. The method, called SINDYc (Sparse Identification of Nonlinear Dynamics with Control), uses time histories of the aircraft state variables and control actuation to develop a system of nonlinear, polynomial equations modeling the dynamic behavior. While outside the scope of this research, the identified model could be used in concert with a model predictive controller.

1.2 Nonlinear Model Derivation and Nonlinear Control Methods

Development of any dynamic aircraft model, linear or nonlinear, is a process requiring thoughtful assumptions and awareness of the conditions for which it is valid. Garrad et al. develops a longitudinal aircraft model, nonlinear in both dynamics and aerodynamics, and synthesizes a nonlinear feedback controller for high-AOA stall recovery [3]. The dynamic model includes trigonometric terms approximated with Taylor polynomials, and the aerodynamic model includes cubic functions to model stall behavior. This process finally results in a system of three nonlinear ODEs in airspeed, \( \dot{u} \); angle of attack, \( \dot{\alpha} \); and pitch, \( \dot{\theta} \). These terms are linearized about the trim condition, and an LQR controller is synthesized using the linear portion of the dynamic equations. Additionally, the author synthesizes two optimal nonlinear controllers satisfying the Hamilton-Jacobi equation – one of second order and another of third. The performance of each controller is evaluated by its ability to recover from a shallow and deep stall. The highest performance resulted from the cubic controller, followed by the quadratic controller, with the linear controller displaying the weakest ability to recover from stall.
There are a multitude of methods to control nonlinear systems. Beeler et al. provides a comprehensive survey of five prominent methods used to synthesize nonlinear optimal feedback controllers, evaluate their performance, and other specific considerations [4]. These five methods are 1) the Power Series Approximation 2) the State-Dependent Riccati Equation 3) Successive Galerkin Approximation 4) Interpolation of Two-Point Boundary Value (TPBV) Problem Solutions and 5) Interpolation of Iterative Solutions. Each method provides a different approach to solve the Hamilton-Jacobi-Bellman equations, either optimally or sub-optimally.

While not explicitly an optimal control framework, Model Predictive Control (MPC) is an advanced control framework that shows considerable promise in nonlinear applications [5]. MPC works by using a mathematical reference model of a nonlinear system and predicting the dynamic response, over some finite time horizon, as to optimize the response using some control policy. This policy is implemented in the next time step and process repeats indefinitely.

1.3 Data-Driven System Identification and the SINDy Algorithm

The generation of physical models from experimental data is a field of study that has risen to prominence as computers have gotten faster, more powerful, and more affordable. Published in Science Magazine, April 2009, Schmidt et al. propose a method of constructing natural laws from experimental data that accurately model dynamic systems [6]. The method was successfully able to determine the natural laws that describe simple pendulums, double pendulums, and single and double simple harmonic oscillators. The method works by first calculating all partial derivatives for every pair of state variables using the experimentally collected data. The algorithm generates candidate functions, initially at random, and determines how closely the predicted derivatives match the observed derivatives. Through an iterative process of varying candidate equations and combining pairs of equations to create chains of equations, the candidate functions tend toward the true natural laws. The method converges when the model is able to accurately predict behavior using a sufficiently compact set of equations. The equations which this process returns include equations of motion, laws of energy conservation, the Hamiltonian, the Lagrangian, and any constraint imposed upon the system. When choosing sufficient compactness for the set of
final equations, the authors noted a point in equation development where further complexity did not provide further accuracy, and further simplification resulted in significant error. This point of optimal parsimony is the target of this and other such methods.

One of the most promising frameworks used to create data-driven dynamic models is artificial neural networks. Yan et al. presents performance results of a model predictive controller acting on several systems of unknown dynamics [7]. The dynamics of each system are ‘learned’ offline using three methods: two neural networks and a least-squares support vector machine (LS-SVM) [8]. The two neural networks are of different architecture: one being a multilayer perceptron – backpropagation network (MLP-BP) and the other an extreme learning machine (ELM) [9]. Once trained, the predictor uses the learned model in the MPC framework. Generally, the ELM learned fastest by a significant margin – less than one second – but sacrificed testing accuracy. The MLP-BP and LS-SVM provided similar levels of accuracy, with the MLP-BP requiring tens of seconds to train and the LS-SVM requiring hundreds.

The method used in this investigation of nonlinear system identification was first published in 2016 by Brunton, Proctor, and Kutz [10]. The authors present SINDy (Sparse Identification of Nonlinear Dynamics), a powerful technique to generate nonlinear dynamic models in the form of systems of polynomial ODEs. The technique assumes the physical process can be modeled with only a few significant terms – a feature of many real-world dynamic systems. To apply SINDy, a time history of the system dynamics must be captured, and this history must sufficiently explore the state-space of the system, or at least the region of interest within that state-space. Additionally, a library of simple nonlinear candidate functions must be developed.

These simple functions include unity, the state variables themselves, monomial combinations of state variables, and trigonometric functions. It is critical that this library include terms that form the basis of the dynamics – a library consisting of only monomial terms will not accurately predict the dynamics of a system modeled by trigonometric functions. With the time history of the system and a library of candidate functions, the coefficients of the nonlinear dynamic equations are determined by performing a linear regression.
The regression determines the coefficients of the candidate functions that most closely map the system dynamics. The linear regression may result in considerable overfitting and assign nonzero coefficients to variable combinations that are not truly participants in the system dynamics. To combat this, a sparsity-promoting process is employed that neglects any terms with coefficients smaller than some threshold, and the regression is repeated with the remaining candidate terms. This process iterates until only the terms of meaningful contribution to the nonlinear dynamics remain. The authors assert this process is robust to Gaussian noise on the state variables. Additionally, if the state derivatives are not known directly, they may be estimated using finite differences. The authors apply SINDy to the nonlinear Lorenz system, a simple nonlinear model of chaotic convection, and successfully identify the dynamic terms with predicted coefficients within .03% of their true value. The authors further acknowledge that given a system of sufficiently high order or fast dynamics, like that of vortex shedding around a cylinder, the dynamics predicted by SINDy do not accurately predict transient behavior, but do capture the form of the dynamics and correctly identify the limit cycle behavior.

The same authors, Brunton et al, appended their SINDy algorithm so that it may apply to nonlinear dynamic systems that include a known control input, SINDYc [11]. The algorithm is modified such that the control input is included in the known time-history of the system, and the library of candidate functions additionally includes nonlinear terms of the control input. The library may also include nonlinear cross terms of the state and input variables. The authors note that the linear regression becomes ill conditioned if the control input is part of a feedback control framework – the control input must be an external forcing independent of the system state. Further, the algorithm will provide the best system identification if the control input perturbs the system sufficiently in state-directions that reveal the full richness of the system dynamics. The authors apply SINDYc to a predator-prey model, nonlinear in its two states with a single input, and successfully identify the nonlinear dynamics. Loiseau et al further appends the SINDy algorithm to enforce state constraints, such as energy balance and momentum conservation, in the regression [12].

The SINDy algorithm becomes especially powerful as Kaiser et al. extend SINDYc to
work in concert with a model-predictive control (MPC) framework [13]. The resulting algorithm, SINDY-MPC, when compared to neural networks of similar purpose, requires significantly less experimental data, is computationally more efficient, and is more robust to noise. These qualities make SINDY-MPC viable for on-line execution, especially when used with an on-line linear control framework as a stopgap until SINDY-MPC has collected sufficient data. The input is subject to constraints and the states subject to system dynamics.

Kaiser demonstrates this method using several nonlinear systems, including AOA tracking for an F-4 crusader. In this example, the SINDy algorithm outperforms a neural network and another data-driven system identification and control algorithm. The author concludes with a discussion of the performance of multiple nonlinear system identification methods based on criteria including noise robustness, training time, limited training data and others. The SINDY-MPC algorithm performs strongly in all categories with the exception of high dimensional systems, where it is computationally limited by increasing library size. Additionally, to account for very time-dependent systems, the author suggests actual implementation of the SINDY-MPC framework would likely involve alternating phases of system identification and system control. The author notes that a judicious choice of library functions is critical to the effectiveness of the SINDY algorithm, and the choice of library functions is largely based on expert knowledge of the system. This suggests future innovations in the field of data-driven system identification may call for "a shift from 'big data' to 'smart data.'"
CHAPTER 2
DERIVATION OF NONLINEAR DYNAMIC MODEL

2.1 Development of Equations of Motion

2.1.1 General Equations of Unsteady Aircraft Motion

Aircraft dynamics are modeled using a body-fixed coordinate system with the origin co-incident with the center of mass; the x-axis aligned longitudinally, positive forward; the y-axis aligned laterally, positive to the right; and the z-axis aligned vertically, positive downward. It is assumed the aircraft is symmetrical about the x-z plane. Additionally the model assumes angular momentum of the engines is unchanging in both magnitude and direction relative to the body-fixed coordinate system.

The components of velocity and angular velocity are expressed as \[u, v, w\] and \[p, q, r\], respectively. The aircraft attitude is described by \(\phi\), the roll angle, and \(\theta\), the pitch angle. The quantities \(I_x, I_y, I_z\) and \(I_{zx}\) represent the three moments of inertia and the only pertinent product of inertia, respectively. The general equations of unsteady aircraft motion are provided in three force equations, three moment equations, and two kinematic equations [14]. The forces and moments about each axis are given by \([X, Y, Z]\) and \([L, M, N]\), respectively. These are expanded in Section 2.3.

\[
m(\dot{u} + qw - rv) = X \tag{2.1}
\]
\[
m(\dot{v} + ru - pw) = Y \tag{2.2}
\]
\[
m(\dot{w} + pv - qu) = Z \tag{2.3}
\]
\[
I_x \dot{p} - I_{zx} \dot{r} + (I_z - I_y)qr - I_{zx}pq = L \tag{2.4}
\]
\[
I_y \dot{q} + (I_x - I_z)rp + I_{zx}(p^2 - r^2) = M \tag{2.5}
\]
\[
I_z \dot{r} - I_{zx} \dot{p} + (I_y - I_x)pq + I_{zx}qr = N \tag{2.6}
\]
\[
\dot{\phi} = p + (q \sin \phi + r \cos \phi) \tan \theta
\]  
(2.7)

\[
\dot{\theta} = q \cos \phi - r \sin \phi
\]  
(2.8)

### 2.1.2 Dynamics in Nonlinear State-Space Form

Solving for linear and angular acceleration terms, the dynamics can be expressed in non-
linear state-space form. A full derivation of the moment equations derivation is provided in Ap-
pendix E.

\[
\dot{u} = -qw + rv + \frac{1}{m}X
\]  
(2.9)

\[
\dot{v} = -ru + pw + \frac{1}{m}Y
\]  
(2.10)

\[
\dot{w} = -pv + qu + \frac{1}{m}Z
\]  
(2.11)

\[
\dot{p} = \left[\tilde{I}_{zx}(I_x - I_y) + \tilde{I}_{z,xx}\right] pq + \left[\tilde{I}_z(I_y - I_z) - \tilde{I}_{z,xx}^2\right] qr + \tilde{I}_z L + \tilde{I}_{z,xx} N
\]  
(2.12)

\[
\dot{q} = I_y^{-1}(I_z - I_x) rp + \tilde{I}_{zx} r^2 - I_y^{-1} I_{zx} p^2 + I_y^{-1} M
\]  
(2.13)

\[
\dot{r} = \left[\tilde{I}_{zx}(I_y - I_z) - \tilde{I}_{z,xx}\right] qr + \left[\tilde{I}_x(I_x - I_y) + \tilde{I}_{z,xx}^2\right] pq + \tilde{I}_x N + \tilde{I}_{z,xx} L
\]  
(2.14)

For compactness, recurring combinations of mass property terms are defined as follows.

\[
\tilde{I}_x \equiv \frac{I_x}{I_x I_z - I_{zx}^2}
\]

\[
\tilde{I}_{zx} \equiv \frac{I_{zx}}{I_x I_z - I_{zx}^2}
\]

\[
\tilde{I}_z \equiv \frac{I_z}{I_x I_z - I_{zx}^2}
\]

\[
\tilde{I}_{z,xx} \equiv \frac{I_{z,xx}}{I_x I_z - I_{zx}^2}
\]

\[
\tilde{I}_{zx} \equiv \frac{I_{z,xx} I_{zx}}{I_x I_z - I_{zx}^2}
\]

\[
\tilde{I}_{z,xx} \equiv \frac{I_{z,xx} I_z}{I_x I_z - I_{zx}^2}
\]

As the final dynamic model will be expressed as a system of nonlinear polynomial differ-
tential equations, the following trigonometric substitutions are made to the kinematic equations –
these substitutions appear throughout this paper. The error associated with them is discussed in
Appendix A. In short, at 45° the cosine, sine, and tangent approximations result in errors of 2.2%, 0.35%, and 5.3%, respectively.

\[
\cos \xi \approx 1 - \frac{\xi^2}{2} \quad \sin \xi \approx \xi - \frac{\xi^3}{6} \quad \tan \xi \approx \xi + \frac{\xi^3}{3} \quad (2.16)
\]

Substituting the trigonometric approximations into equations 2.7 and 2.8, the kinematics can be expressed in nonlinear state-space form. A complete derivation of these equations is provided in Appendix B.

\[
\dot{\phi} \approx p + q\phi \theta + r\theta \quad (2.17)
\]

\[
\dot{\theta} \approx q - \frac{q\phi^2}{2} - r\phi \quad (2.18)
\]

### 2.1.3 Transformation to Aerodynamic Coordinates

As the aerodynamic model is expressed in terms of \( \alpha \) and \( \beta \), not \( v \) and \( w \), the dynamic equations must be expressed in terms of aerodynamic variables. This is done using the following substitutions for \( v \) and \( w \) and their respective time derivatives. Since the terms \( \dot{u}\tan \beta \) and \( \dot{u}\tan \beta \) are products of small variables and therefore insignificant relative to the \( \sec^2 \) terms, they are neglected.

\[
v = u\tan \beta \quad w = u\tan \alpha \quad (2.19)
\]

\[
\dot{v} \approx \dot{u}\tan \beta + u\dot{\beta} \sec^2 \beta \quad \dot{w} \approx \dot{u}\tan \alpha + u\dot{\alpha} \sec^2 \alpha \quad (2.20)
\]

Applying the trigonometric substitutions of 2.16 and neglecting terms greater than degree three results in the following.

\[
v \approx u \left( \beta + \frac{\beta^3}{3} \right) \approx u\beta \quad w \approx u \left( \alpha + \frac{\alpha^3}{3} \right) \approx u\alpha \quad (2.21)
\]

\[
\dot{v} \approx \frac{u\dot{\beta}}{\left(1 - \frac{\beta^2}{2}\right)^2} \quad \frac{u\dot{\beta}}{1 - \beta^2} \quad \dot{w} \approx \frac{u\dot{\alpha}}{\left(1 - \frac{\alpha^2}{2}\right)^2} \quad \frac{u\dot{\alpha}}{1 - \alpha^2} \quad (2.22)
\]
Substituting these approximations into equations 2.9, 2.10, and 2.11 results in the following state-space equations. A detailed derivation is included in Appendix C.

\[ \dot{u} \approx -qu\alpha + ru\beta + \frac{1}{m}X \quad (2.23) \]
\[ \dot{\beta} \approx -r + p\alpha + \beta^2 r + \frac{1}{mU_1}Y - \frac{\beta^2}{mU_1}Y \quad (2.24) \]
\[ \dot{\alpha} \approx q - p\beta - \alpha^2 q + \frac{1}{mU_1}Z - \frac{\alpha^2}{mU_1}Z \quad (2.25) \]

Note that the variable \( u \) in the denominator of 2.24 and 2.25 has been replaced with \( U_1 \), the trim airspeed. While this limits the applicability of the model to dynamic behavior near \( u = U_1 \), it is necessary as the method of system identification will not include in its library variables of negative degree. However, it is possible to modify the method to include such terms.

The system identification to follow will be better conditioned if Equation 2.23 is expressed in terms of normalized velocity along the x-axis. Dividing both sides by \( U_1 \) results in the following state equation where \( \hat{u} = \frac{u}{U_1} \). As a result, when \( u = U_1 \), the trim speed, \( \hat{u} = 1 \).

\[ \hat{u} \approx -q\hat{u}\alpha + r\hat{u}\beta + \frac{1}{mU_1}X \quad (2.26) \]

### 2.2 Development of Aerodynamic Model

#### 2.2.1 Gravity Model

The gravity model is developed rather straightforwardly using a conventional 3-2-1 Euler rotation. The gravity vector is transformed from the Earth-fixed reference frame to the body-fixed reference frame as shown.

\[
W_{BF} = [R_\phi][R_\theta][R_\psi]W_{EF} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \phi & \sin \phi \\ 0 & -\sin \phi & \cos \phi \end{bmatrix} \begin{bmatrix} \cos \theta & 0 & -\sin \theta \\ 0 & 1 & 0 \\ \sin \theta & 0 & \cos \theta \end{bmatrix} \begin{bmatrix} \cos \psi & -\sin \psi & 0 \\ \sin \psi & \cos \psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix} \quad (2.27)
\]
The components of the weight vector may be approximated as polynomials by employing equations [2.16]. Note that all terms greater than degree three in equation [2.29] have been neglected. This is an assumption that is used repeatedly throughout this paper.

\[
\begin{align*}
\begin{cases}
W_x \\
W_y \\
W_z \end{cases}_{BF} & = \begin{cases}
-mg \sin \theta \\
mg \cos \theta \sin \phi \\
mg \cos \theta \cos \phi 
\end{cases} \\
\end{align*}
\] (2.28)

2.2.2 Lift Model

Lift is modeled as the sum of lift contributions from the wing and horizontal tail.

\[
L = L_w + L_t = C_{Lw} \bar{q} S + C_{Lt} \bar{q} S_t
\] (2.30)

The lift coefficient of the wing is modeled as a cubic function of \( \alpha \), the angle of attack of the wing. The lift coefficient of the horizontal tail is modeled as a cubic function of \( \alpha_t \), the angle of attack of the horizontal tail, and a linear function of \( \delta_e \), the elevator deflection, as follows.

\[
\begin{align*}
C_{Lw} &= C_{Lw0} + C_{Lwa} \alpha - C_{Lwa3} \alpha^3 \\
C_{Lt} &= C_{Lt0} + C_{Lt\alpha} \alpha_t - C_{Lt\alpha3} \alpha_t^3 + C_{Lt\delta_e} \delta_e
\end{align*}
\] (2.31, 2.32)

Performing these substitutions results in the complete lift model.

\[
L = \left( C_{Lw0} + C_{Lwa} \alpha - C_{Lwa3} \alpha^3 \right) \bar{q} S + \left( C_{Lt0} + C_{Lt\alpha} \alpha_t - C_{Lt\alpha3} \alpha_t^3 + C_{Lt\delta_e} \delta_e \right) \bar{q} S_t
\] (2.33)
Here $S$ is the surface area of the wing and $S_t$ is the surface area of the horizontal tail. The dynamic pressure is given as $\bar{q}$. The determination of $C_{Lw0}$ is detailed in Appendix G. The coefficients $C_{Lwa}$ and $C_{La}$ are computed using the Polhamus formula, detailed in Appendix F. The coefficients $C_{Lwa3}$ and $C_{La3}$ are determined by using a cubic fit that approximates the linear region at low angles of attack and approaches the maximum lift at high angles of attack, detailed in Appendix G. This model could be made more accurate by incorporating a $\alpha^2$ term, however to simplify model development, this is omitted – the method of system identification that follows would be capable of identifying such a term. The coefficient $C_{Lt_k}$ is taken from Napolitano [15].

2.2.3 Downwash Model

The downwash model is used to determine the effective angle of attack acting on the horizontal tail, $\alpha_t$. This is the result of both the angle of incidence, $i_t$, at which the tail is set and the air deflected downward off the wing. The angle of attack of the horizontal tail is expressed as follows.

$$\alpha_t = \alpha - i_t - \varepsilon$$

(2.34)

The downwash, $\varepsilon$, is modeled using a first-order Taylor approximation where $\varepsilon_0$ is the downwash at zero angle of attack and $\varepsilon_\alpha$ is the partial derivative of downwash with respect to alpha ($\partial \varepsilon / \partial \alpha$).

$$\varepsilon = \varepsilon_0 + \varepsilon_\alpha \alpha$$

(2.35)

Substituting appropriately results in the effective horizontal tail angle of attack. The parameters $i_t$ and $\varepsilon_\alpha$ are taken from Napolitano [15]; $\varepsilon_0$ is neglected.

$$\alpha_t = \alpha - i_t - \varepsilon_0 - \varepsilon_\alpha \alpha$$

$$= -i_t - \varepsilon_0 + (1 - \varepsilon_\alpha) \alpha$$

(2.36)
2.2.4 Drag Model

Drag is modeled using the drag polar as a function of the profile drag, $C_{D_0}$, and wing lift, $C_{L_w}$.

$$D = \left( C_{D_0} + k(C_{L_w})^2 \right) \bar{q}S \quad (2.37)$$

The constant $k$ is calculated by $k = 1/e\pi\mathcal{AR}$ where $e$ is the Oswald efficiency factor and $\mathcal{AR}$ is the aspect ratio of the wing. Substituting equation [2.31] into equation [2.37] produces the following drag model. Again, only terms of degree three and lower are retained. The quantity $C_{D_0}$ is taken from Napolitano [15].

$$D = \left[ C_{D_0} + k \left( C_{L_w0} + C_{L_w\alpha} \alpha - C_{L_w\alpha^3} \alpha^3 \right) \right]^2 \bar{q}S \quad (2.38)$$

$$\approx \left( C_{D_0} + kC_{L_w0}^2 + 2kC_{L_w0}C_{L_w\alpha} \alpha - 2kC_{L_w0}C_{L_w\alpha^3} \alpha^3 + kC_{L_w\alpha^2}^2 \right) \bar{q}S \quad (2.39)$$

2.2.5 Side Force Model

The side force model combines contributions from the wings, fuselage, vertical tail, and rudder. This is expressed as follows where $S_V$ is the surface area of the vertical tail.

$$F_Y = F_{Y_{wb}} + F_{Y_V} + F_{Y_{\delta_r}} \quad (2.40)$$

$$= C_{Y_{wb}} \bar{q}S + C_{Y_{V}} \bar{q}S_V + C_{Y_{\delta_r}} \bar{q}S \quad (2.41)$$

The side force coefficient of the vertical tail is modeled as a cubic function of $\beta$, the sideslip angle. Side force acting on the wings and fuselage due to sideslip is modeled as linear function of $\beta$. Similarly, the side force acting on the rudder is modeled as a linear function of $\delta_r$, the rudder deflection.

$$C_{Y_{wb}} = C_{Y_{wb}} \beta \quad C_{Y_V} = C_{Y_{V}} \beta - C_{Y_{V}} \beta^3 \quad C_{Y_{\delta_r}} = C_{Y_{\delta_r}} \delta_r \quad (2.42)$$
Performing these substitutions results in the complete side force model.

\[
F_Y = \left( C_{Y\beta} \beta - C_{Y\beta^3} \beta^3 \right) \bar{q}S + \left( C_{Y_{\alpha\delta}} \beta + C_{Y_{\delta\delta}} \delta \right) \bar{q}S
\] (2.43)

The coefficient \(C_{Y\beta}\) is computed using the Polhamus formula as detailed in Appendix \(F\). The coefficient \(C_{Y\beta^3}\) is determined by using a cubic fit that approximates the linear region at low sideslip and approaches the maximum lift at high sideslip, detailed in Appendix \(G\). In a similar way to the lift model, the side force model could be made more accurate by incorporating a \(\beta^2\) term, however to simplify model development, this is omitted. Derivation of \(C_{Y_{\alpha\delta}}\) is provided in Appendix \(G\). The coefficient \(C_{Y_{\delta\delta}}\) is taken from Napolitano [15].

2.3 Development of Force and Moment Equations

2.3.1 Forces in X-Direction

The net force in the x-direction is composed of weight, \(W_x\); thrust, \(F_T\); drag, \(D\); and lift, \(L\). As lift and drag act perpendicular and parallel, respectively, to the relative wind, these must be projected onto the body-fixed x-axis. Terms are assembled as follows.

\[
X = W_x + F_T - D \cos \alpha + L \sin \alpha
\] (2.44)

The thrust will be chosen as some constant value, \(T_{trim}\), as to be both reasonable and adequately trim the aircraft. Substituting the trigonometric approximations and equations 2.29, 2.33, 2.39 into 2.44 results in the following equation of force in the x-direction.

\[
X = -\frac{mg}{6} (6\theta - \theta^3) + T_{trim}
\]

\[
- \left( C_{D_0} + kC_{Lw_0}^2 + 2kC_{Lw_0}C_{Lwa} \alpha - 2kC_{Lw_0}C_{Lwa^3} \alpha^3 + kC_{Lw_a}^2 \alpha^2 \right) \bar{q}S \left( 1 - \frac{\alpha^2}{2} \right)
\]

\[
+ \left( C_{Lw_0} + C_{Lwa} \alpha - C_{Lwa^3} \alpha^3 \right) \bar{q}S + \left( C_{Lt_0} + C_{Lt_\alpha} \alpha - C_{Lt_a} \alpha^3 - C_{Lt_{\alpha_\delta}} \delta \right) \bar{q}S \bar{q} \left( \alpha - \frac{\alpha^3}{6} \right)
\] (2.45)
Multiplying through and eliminating all terms greater than degree three results in the complete x-direction equation. The full derivation can be found in Appendix D.

\[ X = -\frac{mg}{6} (6\theta - \theta^3) + T_{trim} \]

\[ \quad - \left[ C_{D_0} + kC_{Lw_0}^2 + 2kC_{Lw_0}C_{Lw_0}\alpha + \left( kC_{Lw_0}^2 - \frac{C_{D_0}}{2} - \frac{kC_{Lw_0}^2}{2} \right) \alpha^2 \right] \bar{q}S \]

\[ \quad + \left[ (2kC_{Lw_0}C_{Lw_0}^3 + kC_{Lw_0}C_{Lw_0}) \alpha^3 \right] \bar{q}S + \left( C_{Lw_0} \alpha + C_{Lw_0} \alpha^2 - \frac{C_{Lw_0}}{6} \alpha^3 \right) \bar{q}S \]

\[ \quad + \left( C_{L_\alpha} \alpha + C_{L_\alpha} \alpha \alpha + C_{L_\delta} \delta \alpha - \frac{C_{L_\delta}}{6} \alpha^3 \right) \bar{q}S_l \] (2.46)

### 2.3.2 Forces in Y-Direction

The net force in the y-direction is composed of weight, \( W_y \), and side force, \( F_y \). Terms are assembled as follows.

\[ Y = W_y + F_y \] (2.47)

Substituting the equations [2.29] and [2.43] into [2.47] results in the following equation of force in the y-direction.

\[ Y = \frac{mg}{6} (6\phi - \phi^3 - 3\theta^2 \phi) + \left( C_{Y_{\beta}} \beta - C_{Y_{\beta}} \beta^3 \right) \bar{q}S_v + \left( C_{Y_{\beta}} \beta + C_{Y_{\delta}} \delta \right) \bar{q}S \] (2.48)

### 2.3.3 Forces in Z-Direction

The net force in the z-direction is composed of weight, \( W_z \); drag, \( D \); and lift, \( L \). As lift and drag act perpendicular and parallel, respectively, to the relative wind, these must be projected onto the body-fixed z-axis. Terms are assembled as follows.

\[ Z = W_z - L \cos \alpha - D \sin \alpha \] (2.49)
Substituting the trigonometric approximations and equations 2.29, 2.33, 2.39 into 2.49 results in the following equation of force in the z-direction.

\[ Z = \frac{mg}{2} \left( 2 - \theta^2 - \phi^2 \right) \]

\[- \left[ \left( C_{Lw_0} + C_{L_\alpha} \alpha - C_{Lw_\alpha_3} \alpha^3 \right) \bar{q}S + \left( C_{L_\alpha} + C_{L\alpha_\alpha} \alpha + C_{L_{\delta_e}} \delta_e \right) \bar{q}S \right] \left( 1 - \frac{\alpha^2}{2} \right) \]

\[- \left[ C_{D_0} + k \left( C_{Lw_0} + C_{L_\alpha} \alpha - C_{Lw_\alpha_3} \alpha^3 \right)^2 \bar{q}S \left( \alpha - \frac{\alpha^3}{6} \right) \right] (2.50) \]

Multiplying through and eliminating all terms greater than degree three results in the complete z-direction equation. The full derivation can be found in Appendix D.

\[ Z = \frac{mg}{2} \left( 2 - \theta^2 - \phi^2 \right) \]

\[- \left[ C_{Lw_0} + C_{L_\alpha} \alpha - C_{Lw_\alpha_3} \alpha^3 \right] \bar{q}S \]

\[- \left( C_{L_\alpha} + C_{L\alpha_\alpha} \alpha + C_{L_{\delta_e}} \delta_e \right) \bar{q}S \]

\[- \left( C_{L_\alpha} + k C_{L_\alpha_\alpha} \alpha \right) \bar{q}S \]

\[- \left( k C_{L_\alpha_\alpha} \alpha \right) \bar{q}S (2.51) \]

### 2.3.4 Moment About Roll Axis

The net moment about the x-axis is composed of rolling moments due to sideslip, \( \dot{\beta} \); aileron deflection, \( \delta_\alpha \); rudder deflection, \( \delta_r \); and roll rate, \( \dot{p} \). Terms are assembled as follows.

\[ L = L_\beta + L_{\delta_\alpha} + L_{\delta_r} + L_p \]  

(2.52)

Each component may be expressed in terms of their respective stability derivatives. These terms are taken from Napolitano [15]. The following moment components introduce the aircraft wingspan, \( b \), and the non-dimensional roll rate, \( \frac{pb}{2U_1} \).

\[ L_\beta = C_{l_\beta} \dot{\beta} \bar{q}Sb \quad L_{\delta_\alpha} = C_{l_\delta_\alpha} \delta_\alpha \bar{q}Sb \quad L_{\delta_r} = C_{l_\delta_r} \delta_r \bar{q}Sb \quad L_p = C_{l_p} \frac{pb}{2U_1} \bar{q}Sb \]  

(2.53)
Substituting equations 2.53 into equation 2.52 results in the complete rolling moment equation.

\[ L = \left( C_{lq} \beta + C_{l_\delta_a} \delta_a + C_{l_\delta_r} \delta_r + C_{l_p} \frac{p b}{2U_1} \right) \bar{q} S b \]  

(2.54)

2.3.5 Moment About Pitch Axis

The net moment about the y-axis is composed of pitching moments due to angle of attack, \( \alpha \), and pitch rate, \( q \). Additionally included in the pitching moment are the lift on the wing and horizontal tail acting at distances of \( l_w \) and \( l_t \) from the y-axis, respectively. Terms are assembled and lift components expanded as follows.

\[ M = l_w L_w \cos \alpha - l_t L_t \cos \alpha + M_q \]
\[ = l_w C_{Lw} \bar{q} S \cos \alpha - l_t C_{Lt} \bar{q} S_t \cos \alpha + C_{Mq} \bar{q} q S b \]  

(2.55)

The component \( M_q \) may be expressed in terms of its respective stability derivatives as follows. This value is taken from the Napolitano [15]. The pitch rate is expressed here as the non-dimensional pitch rate, \( \frac{\bar{q} c}{2U_1} \).

\[ M_q = C_{Mq} \frac{q \bar{c}}{2U_1} \bar{q} S b \]  

(2.56)

Substituting equations 2.16, 2.31, 2.32 and 2.56 into equation 2.55 results in the following moment equation about the y-axis.

\[ M \approx l_w \left( C_{L_\alpha w} + C_{Lw, \alpha} \alpha - C_{Lw, \alpha^3} \alpha^3 \right) \bar{q} S \left( 1 - \frac{\alpha^2}{2} \right) \]
\[ - l_t \left( C_{L_\alpha t} + C_{L_\alpha t} \alpha - C_{L_\alpha t, \alpha^3} \alpha^3 + C_{L_\alpha t, \delta_r} \delta_r \right) \bar{q} S_t \left( 1 - \frac{\alpha^2}{2} \right) + C_{Mq} \frac{q \bar{c}}{2U_1} \bar{q} S b \]  

(2.57)

Multiplying through and eliminating all terms greater than degree three results in the complete
pitching moment equation.

\[ M \approx l_w \left[ C_{Lw0} + C_{Lw\alpha} \alpha - \frac{C_{Lw0}}{2} \alpha^2 - \left( C_{Lw_{\alpha\alpha}} + \frac{C_{Lw\alpha}}{2} \right) \alpha^3 \right] \bar{q}S \]

\[ - l_t \left[ C_{Lt0} + C_{Lt\alpha} \alpha + C_{Lt_{\alpha\alpha}} \alpha^2 - \frac{C_{Lt0} \alpha^2}{2} + \frac{C_{Lt_{\alpha\alpha}} \alpha^3}{2} - \frac{C_{Lt_{\alpha\beta}} \delta_e \alpha^2}{2} \right] \bar{q}S_t \]

\[ + C_M \frac{q^c}{2U_1} \bar{q}S_b \quad (2.58) \]

### 2.3.6 Moment About Yaw Axis

The net moment about the z-axis is composed of yawing moments due to rudder deflection, \( \delta_r \), aileron deflection, \( \delta_a \), and yaw rate, \( r \). Additionally included in the yawing moment is the side force on the vertical tail acting at a distance of \( l_V \) from the x-axis. Terms are assembled as follows.

\[ N = -l_V F_{Y_V} + N_{\delta_r} + N_{\delta_a} + N_r \quad (2.59) \]

The components \( N_{\delta_r} \), \( N_{\delta_a} \), and \( N_r \) may be expressed in terms of their respective stability and control derivatives as follows. These derivatives are taken from Napolitano [15]. The yaw rate is expressed here as the non-dimensional yaw rate, \( \frac{rb}{2U_1} \).

\[ N_{\delta_r} = C_{N_{\delta_r}} \delta_r \bar{q}Sb \quad N_{\delta_a} = C_{N_{\delta_a}} \delta_a \bar{q}Sb \quad N_r = C_{N_r} \frac{rb}{2U_1} \bar{q}Sb \quad (2.60) \]

Substituting equations 2.43 and 2.60 into equation 2.59 results in the complete yawing moment equation.

\[ N = -l_V \left( C_{Y_{\beta\beta}} \beta - C_{Y_{\beta\beta} \beta^3} \beta^3 \right) \bar{q}S_V + \left( C_{N_{\delta_r}} \delta_r + C_{N_{\delta_a}} \delta_a + \frac{C_{N_r}}{2U_1} r \right) \bar{q}S_b \quad (2.61) \]

### 2.4 Aircraft Data – SIAI Marchetti S-211

This section presents the mass properties, geometric parameters, trim condition, and aerodynamic coefficients of the SIAI Marchetti S-211. The S-211 is an Italian, turbofan-powered
military trainer and light attack aircraft in use from 1981 to the present. This aircraft, shown in Figure 2.1 [16], will be used as the case study for this paper.

![Figure 2.1: SIAI-Marchetti S-211 In Flight](image)

The S-211 was selected due to its high maneuverability, resulting in strong lateral-directional coupling, and, as an attack aircraft, it does realistically achieve high roll angle and angle of attack. This aircraft should display rich, nonlinear dynamics that will subsequently lend itself to model identification.

**TABLE 2.1**

<table>
<thead>
<tr>
<th>Quantity</th>
<th>Units</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mass</td>
<td>$m = 124.2$</td>
<td>slug</td>
</tr>
<tr>
<td>Gravitational Acceleration</td>
<td>$g = 32.2$</td>
<td>ft/s²</td>
</tr>
<tr>
<td>Moment of Inertia, x-axis</td>
<td>$I_{xx} = 800$</td>
<td>slug·ft²</td>
</tr>
<tr>
<td>Moment of Inertia, y-axis</td>
<td>$I_{yy} = 4800$</td>
<td>slug·ft²</td>
</tr>
<tr>
<td>Moment of Inertia, z-axis</td>
<td>$I_{zz} = 5200$</td>
<td>slug·ft²</td>
</tr>
<tr>
<td>Product of Inertia, xz-plane</td>
<td>$I_{xz} = 200$</td>
<td>slug·ft²</td>
</tr>
</tbody>
</table>
TABLE 2.2
S-211 GEOMETRIC PARAMETERS

<table>
<thead>
<tr>
<th>Quantity</th>
<th>Units</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean Aerodynamic Chord</td>
<td>( \bar{c} = 5.4 ) ft</td>
<td>[15]</td>
</tr>
<tr>
<td>Wing Surface</td>
<td>( S = 136 ) ft(^2)</td>
<td>[15]</td>
</tr>
<tr>
<td>Wingspan</td>
<td>( b = 26.3 ) ft</td>
<td>[15]</td>
</tr>
<tr>
<td>Wing AC Moment Arm</td>
<td>( l_w = 1.09 ) ft</td>
<td>[15]</td>
</tr>
<tr>
<td>Wing Incidence Angle</td>
<td>( i_w = 0.035 ) rad</td>
<td>[15]</td>
</tr>
<tr>
<td>H-Tail Surface</td>
<td>( S_t = 33.6 ) ft(^2)</td>
<td>[15]</td>
</tr>
<tr>
<td>H-Tail Span</td>
<td>( b_t = 13.3 ) ft</td>
<td>[15]</td>
</tr>
<tr>
<td>H-Tail AC Moment Arm</td>
<td>( l_t = 10.77 ) ft</td>
<td>[15]</td>
</tr>
<tr>
<td>H-Tail Incidence Angle</td>
<td>( i_t = 0.0 ) rad</td>
<td>[15]</td>
</tr>
<tr>
<td>V-Tail Surface</td>
<td>( S_V = 22.3 ) ft(^2)</td>
<td>[15]</td>
</tr>
<tr>
<td>V-Tail Span</td>
<td>( b_V = 5.8 ) ft</td>
<td>[15]</td>
</tr>
<tr>
<td>V-Tail AC Moment Arm</td>
<td>( l_V = 9.00 ) ft</td>
<td>[15]</td>
</tr>
<tr>
<td>Rudder Moment Arm</td>
<td>( l_r = 11.8 ) ft</td>
<td>[15]</td>
</tr>
</tbody>
</table>

TABLE 2.3
S-211 TRIM CONDITIONS

<table>
<thead>
<tr>
<th>Quantity</th>
<th>Units</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>Altitude</td>
<td>( h = 25000 ) ft</td>
<td>[15]</td>
</tr>
<tr>
<td>Thrust ( T_{trim} )</td>
<td>( 592 ) lbf</td>
<td>[15]</td>
</tr>
<tr>
<td>Airspeed</td>
<td>( U_1 = 610 ) ft/s</td>
<td>[15]</td>
</tr>
<tr>
<td>Dynamic Pressure</td>
<td>( \bar{q} = 198 ) lb/ft(^2)</td>
<td>[15]</td>
</tr>
</tbody>
</table>
TABLE 2.4
S-211 AERODYNAMIC COEFFICIENTS

<table>
<thead>
<tr>
<th>Source</th>
<th>Source</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{D_0} = 0.021 \quad [15]$</td>
<td>$C_{L_{V\beta}} = -0.0872 \quad \text{Appendix G}$</td>
<td></td>
</tr>
<tr>
<td>$C_{L_{W_0}} = 0.173 \quad \text{Appendix G}$</td>
<td>$C_{Y_{V\beta}} = -3.638 \quad \text{Appendix F}$</td>
<td></td>
</tr>
<tr>
<td>$C_{L_{W_\alpha}} = 4.957 \quad \text{Appendix F}$</td>
<td>$C_{Y_{V\beta}} = -7.00 \quad \text{Appendix G}$</td>
<td></td>
</tr>
<tr>
<td>$C_{L_{W_\beta}} = 11.00 \quad \text{Appendix G}$</td>
<td>$C_{Y_{\delta_e}} = 0.028 \quad [15]$</td>
<td></td>
</tr>
<tr>
<td>$C_{L_0} = 0.000 \quad \text{Appendix G}$</td>
<td>$C_{l_{\beta}} = -0.110 \quad [15]$</td>
<td></td>
</tr>
<tr>
<td>$C_{L_C} = 5.027 \quad \text{Appendix F}$</td>
<td>$C_{l_{\delta_e}} = 0.100 \quad [15]$</td>
<td></td>
</tr>
<tr>
<td>$C_{L_\alpha} = 13.00 \quad \text{Appendix G}$</td>
<td>$C_{l_{\delta_r}} = 0.050 \quad [15]$</td>
<td></td>
</tr>
<tr>
<td>$C_{L_{\delta_e}} = 0.380 \quad [15]$</td>
<td>$C_{l_p} = -0.390 \quad [15]$</td>
<td></td>
</tr>
<tr>
<td>$C_{M_{\delta}} = -0.240 \quad [15]$</td>
<td>$C_{N_{\delta_e}} = -0.120 \quad [15]$</td>
<td></td>
</tr>
<tr>
<td>$C_{M_{\delta}} = -17.7 \quad [15]$</td>
<td>$C_{N_{\delta_r}} = -0.003 \quad [15]$</td>
<td></td>
</tr>
<tr>
<td>$\epsilon_{\alpha} = 0.555 \quad [15]$</td>
<td>$C_{N_r} = -0.260 \quad [15]$</td>
<td></td>
</tr>
<tr>
<td>$e = 0.85 \quad \text{Estimate}$</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

2.5 Collection of Nonlinear State Equations

Incorporating all aircraft parameters into the dynamic equations results in the following model of nonlinear dynamics. Additionally all instances of $\alpha_t$ are replaced with the downwash model, $\alpha_t = -i_t - \epsilon_0 + (1 - \epsilon_{\alpha}) \alpha$. As before, all terms of degree greater than three are neglected. The model is described in eight state variables $\vec{x} = \{\hat{u}, \beta, \alpha, p, q, r, \phi, \theta\}$ and three input variables $\vec{u} = \{\delta_a, \delta_e, \delta_r\}$.

\[
\dot{u} = 0.01660\alpha - 0.05279\theta + 0.03337\alpha \delta_e + 1.319\alpha^2 + 0.1118\alpha^3 + 0.008798\theta^3 - \alpha q\hat{u} + \beta r\hat{u} - 0.0004333 \quad (2.62)
\]

\[
\dot{\beta} = -0.2429\beta + 0.009952\delta_r + 0.05279\phi - r + \alpha p - 0.009952\beta^2\delta_r - 0.05279\beta^2\phi + \beta^2 r - 0.02639\phi\theta^2 + 0.6509\beta^3 - 0.008798\phi^3 \quad (2.63)
\]
\[
\alpha = q - 0.03337 \delta_e - 1.966 \alpha - \beta p + 0.05005 \alpha^2 \delta_e - \alpha^2 q - 0.005439 \alpha^2 \\
+ 6.314 \alpha^3 - 0.02639 \phi^2 - 0.02639 \theta^2 - 0.008702 \quad (2.64)
\]

\[
\dot{p} = -13.50 \beta^3 - 91.31 \beta + 89.28 \delta_a + 40.57 \delta_r - 7.515 p - 0.1927 r + 0.05825 pq - 0.5146 qr \\
(2.65)
\]

\[
\dot{q} = 0.9167 pr - 5.672 \delta_e - 11.56 q - 3.083 \alpha + 2.836 \alpha^2 \delta_e - 0.5289 \alpha^2 \\
- 48.62 \alpha^3 - 0.04167 p^2 + 0.04167 r^2 + 1.058 \quad (2.66)
\]

\[
\dot{r} = 24.29 \beta - 54.01 \beta^3 + 3.025 \delta_a - 14.78 \delta_r - 0.2890 p - 0.7708 r - 0.7670 pq - 0.05825 qr \quad (2.67)
\]

\[
\dot{\phi} = p + r \theta + \phi q \phi \\
(2.68)
\]

\[
\dot{\theta} = -0.5000 q \phi^2 - r \phi + q \quad (2.69)
\]

### 2.5.1 Establishing Trim

Trimming the aircraft model is necessary for further analysis. At equilibrium, all time derivatives disappear as well as the following states and inputs: \( \beta = p = q = r = \phi = \delta_a = \delta_r = 0 \). All dynamic equations disappear with the exception of [2.62] [2.64] and [2.66] which reduce to the following in terms of the trim variables.

\[
0 = 0.01660 \alpha_{\text{trim}} - 0.05279 \theta_{\text{trim}} + 0.03337 \alpha_{\text{trim}} \delta_{e\text{trim}} + 1.319 \alpha_{\text{trim}}^2 + 0.1118 \alpha_{\text{trim}}^3 \\
+ 0.008798 \theta_{\text{trim}}^3 - 0.000433 \quad (2.70)
\]

\[
0 = -0.03337 \delta_{e\text{trim}} - 1.966 \alpha_{\text{trim}} + 0.05005 \alpha_{\text{trim}}^2 \delta_{e\text{trim}} - 0.005439 \alpha_{\text{trim}}^2 \\
+ 6.314 \alpha_{\text{trim}}^3 - 0.02639 \theta_{\text{trim}}^2 - 0.008702 \quad (2.71)
\]

\[
0 = -5.672 \delta_{e\text{trim}} - 3.083 \alpha_{\text{trim}} + 2.836 \alpha_{\text{trim}}^2 \delta_{e\text{trim}} - 0.5289 \alpha_{\text{trim}}^2 - 48.62 \alpha_{\text{trim}}^3 + 1.058 \quad (2.72)
\]
These are three nonlinear equations of three variables likely having at least one solution. Using a numerical solver starting from the point \( \{ \alpha, \delta_e, \theta \} = \{0, 0, 0\} \), the following trim condition results. It is comforting that these values are reasonable for a real aircraft.

\[
\alpha_{\text{trim}} = -0.007666 \text{ rad} \quad \delta_{e\text{trim}} = 0.190701 \text{ rad} \quad \theta_{\text{trim}} = -0.010076 \text{ rad} \quad (2.73)
\]

\[
= -0.439^\circ \quad = 10.926^\circ \quad = -0.577^\circ
\]

A negative angle of attack and pitch angle is a reasonable result since the S-211 incorporates a \(2^\circ\) wing incidence angle that contributes to positive lift at zero angle of attack \( (C_{Lw}) \). Incorporating these conditions into the equations of motion, the variables \( \alpha, \delta_e, \) and \( \theta \) may expressed as the sum of perturbed and trim states as follows.

\[
\alpha = \bar{\alpha} + \alpha_{\text{trim}} \quad \delta_e = \bar{\delta}_e + \delta_{e\text{trim}} \quad \theta = \bar{\theta} + \theta_{\text{trim}}
\]

\[
= \bar{\alpha} - 0.007666 \quad = \bar{\delta}_e + 0.190701 \quad = \bar{\theta} - 0.010076 \quad (2.74)
\]

Substituting conditions (2.74) back into (2.62), (2.63), (2.64), (2.66), and (2.68) results in the restate-ment of these equations about the trim point as follows. Note the elimination of constant terms, implying there are no net forces or moments therefore no acceleration along or about any axis.

\[
\dot{\alpha} = 0.002761 \bar{\alpha} - 0.0002558 \bar{\delta}_e - 0.05279 \bar{\theta} + 0.007666 q u + 0.03337 \bar{\alpha} \bar{\delta}_e \\
+ 1.316 \bar{\alpha}^2 + 0.1118 \bar{\alpha}^3 - 0.0002659 \bar{\theta}^2 + 0.008798 \bar{\theta}^3 + \beta r \dot{\alpha} - q \dot{\alpha} (2.75)
\]

\[
\dot{\beta} = -0.2429 \dot{\beta} + 0.009952 \bar{\delta}_e - 0.007666 \rho + 0.05279 \phi - r + p \bar{\alpha} + 0.0005318 \phi \bar{\theta} \\
- 0.009952 \beta^2 \bar{\delta}_e - 0.05279 \beta^2 \phi + \beta^2 r - 0.02639 \phi \bar{\theta}^2 - 0.6509 \beta^3 - 0.008798 \phi^3 \quad (2.76)
\]
\[
\dot{\alpha} = 0.9999q - 1.965\alpha - 0.03337\delta_e + 0.0005318\dot{\theta} - \beta p + 0.01533q\dot{\alpha} - 0.0007673\alpha\delta_e
- q\ddot{\alpha}^2 + 0.05005\alpha^2\ddot{\delta}_e - 0.02639\phi^2 - 0.1411\dot{\alpha}^2 + 6.314\dot{\alpha}^3 - 0.02639\ddot{\theta}^2
\] (2.77)

\[
\dot{q} = 0.9167pr - 3.092\alpha - 5.672\ddot{\delta}_e - 11.56q - 0.04348\dot{\alpha}\ddot{\delta}_e
+ 2.836\alpha^2\ddot{\delta}_e - 0.04167p^2 + 0.04167r^2 + 1.130\dot{\alpha}^2 - 48.62\alpha^3
\] (2.78)

\[
\dot{\phi} = p - 0.01008r - 0.01008\phi q + r\dot{\theta} + \phi q\dot{\theta}
\] (2.79)

This set of equations will be used in the nonlinear system identification to follow. Following incorporation of the trim conditions, the states \(\alpha, \delta_e, \text{ and } \theta\) transform to the perturbed states \(\tilde{\alpha}, \tilde{\delta}_e,\) and \(\tilde{\theta}\); however, the "\(\tilde{}\)" is omitted throughout the analysis.

2.6 Dynamic Model Excitation and Input Response

To build confidence in the system model and gain insight, the trimmed set of equations is excited using nonzero initial conditions in roll, pitch, and yaw rates. The dynamic response of the aircraft system under such initial conditions is provided as follows where all angles and angular rates are measured in radians or radians per second.

Figure 2.2 shows the dynamic behavior resulting from an initial roll rate of 4 rad/s. The sideslip, roll rate, and yaw rate all exhibit a natural frequency response of 5.16 rad/s (\(\omega_\beta = \omega_p = \omega_r = 5.16 \text{ rad/s}\)). Additionally the dampening ratios of the sideslip (\(\zeta_\beta\)), roll rate (\(\zeta_\rho\)), and yaw rate (\(\zeta_\tau\)) are 0.0468, 0.0557, and 0.0398, respectively. These results indicate strong lateral stability and dampening.
Figure 2.2: S-211, Initial Roll Rate Response, $p(0) = 4.0 \text{ r/s}$

Figure 2.3 shows the dynamic behavior resulting from an initial pitch rate of 4 rad/s. This pitch rate induces a strong angle-of-attack response that returns to zero within 2.5 seconds indicating the aircraft is strongly stiff and damped in pitch.

Figure 2.3: S-211, Initial Pitch Rate Response, $q(0) = 4.0 \text{ r/s}$
Figure 2.4 shows the dynamic behavior resulting from an initial yaw rate of 2 rad/s. The sideslip, roll rate, and yaw rate all exhibit a natural frequency response of 5.14 rad/s ($\omega_\beta = \omega_p = \omega_r = 5.14 \text{ rad/s}$). Additionally the dampening ratios of the sideslip ($\zeta_\beta$), roll rate ($\zeta_p$), and yaw rate ($\zeta_r$) are 0.0541, 0.0605, and 0.0544, respectively. These results indicate strong directional stability and dampening.

![Dynamic Behavior Diagram](image)

Figure 2.4: S-211, Initial Yaw Rate Response, $r(0) = 2.0$ rad/s

Additionally, the system may be excited using doublet inputs at the three control actuators – ailerons, elevator, and rudder. The dynamic responses to each of these inputs are provided as follows where all angles and angular rates are measured in radians and radians per second, respectively.

As shown in Figure 2.5 an aileron input induces a strong response in roll rate and roll angle with weaker responses in sideslip and yaw rate.
As shown in Figure 2.6, an elevator input induces strong responses in angle-of-attack, pitch rate, and pitch angle.
As showing in Figure 2.7, a rudder input induces a strong response in sideslip, yaw rate, roll rate, and roll angle. Notably, rudder input produces significant lateral behavior, generating a stronger roll response than yaw response.

Figure 2.7: S-211, Response to Rudder Doublet

The nominal dynamic modes of the S-211 are provided in table 2.5 as follows [15]. This aircraft is stable in all modes – strongly stable in pitch – with the exception of its spiral mode. Note that these characteristic dynamics differ from the natural behavior that results from the S-211 model. These differences are certainly the result of assumptions specific to this nonlinear model and the omission of certain aerodynamic effect such as the fuselage contribution to lift and sideslip.
TABLE 2.5
S-211 NOMINAL DYNAMIC MODES AT TRIM

<table>
<thead>
<tr>
<th>Mode</th>
<th>Time Constant</th>
<th>Natural Frequency</th>
<th>Dampening Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>Short Period</td>
<td>–</td>
<td>1.645</td>
<td>0.742</td>
</tr>
<tr>
<td>Phugoid</td>
<td>–</td>
<td>0.293</td>
<td>0.019</td>
</tr>
<tr>
<td>Dutch Roll</td>
<td>–</td>
<td>1.798</td>
<td>0.212</td>
</tr>
<tr>
<td>Roll</td>
<td>0.276</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>Spiral</td>
<td>-8.09</td>
<td>–</td>
<td>–</td>
</tr>
</tbody>
</table>
GENERALIZED NONLINEAR DYNAMICS

Generalized nonlinear dynamics may be expressed using the following ordinary differential equation. The vector function \( \bar{g} \) describes the dynamics as nonlinear functions of \( \bar{x} \) and \( \bar{u} \), the vectors of states and inputs, respectively.

\[
\dot{\bar{x}} = \bar{g}(\bar{x}, \bar{u}) \tag{3.1}
\]

While an analytical solution to a nonlinear system of sufficient complexity may be impossible or prohibitively challenging, numerical methods provide an adequate approximate solution. Consider a nonlinear dynamic system of \( n \) states and \( m \) inputs where time is discretized into sufficiently small, finite steps of size \( \Delta t \) such that \( t_k = k\Delta t \). Provided a set of initial state conditions, \( \bar{x}(0) \), and perfect knowledge of the inputs as functions of discrete time, \( \bar{u}(k) \), the system of ODEs may be integrated numerically over the interval \( t \in [0, k_f\Delta t] \), resulting in a discrete time-history of the system states as follows.

\[
\begin{cases}
\text{Initial Conditions} \Rightarrow & \bar{g}\{x_1(0), x_2(0), \ldots, x_n(0), u_1(0), u_2(0), \ldots, u_m(0)\} \\
\{\dot{x}_1(1), \dot{x}_2(1), \ldots, \dot{x}_n(1)\} = & \bar{g}\{x_1(1), x_2(1), \ldots, x_n(1), u_1(1), u_2(1), \ldots, u_m(1)\} \\
\{\dot{x}_1(2), \dot{x}_2(2), \ldots, \dot{x}_n(2)\} = & \bar{g}\{x_1(2), x_2(2), \ldots, x_n(2), u_1(2), u_2(2), \ldots, u_m(2)\} \\
& \vdots \\
\{\dot{x}_1(k), \dot{x}_2(k), \ldots, \dot{x}_n(k)\} = & \bar{g}\{x_1(k), x_2(k), \ldots, x_n(k), u_1(k), u_2(k), \ldots, u_m(k)\} \\
& \vdots \\
\{\dot{x}_1(k_f), \dot{x}_2(k_f), \ldots, \dot{x}_n(k_f)\} = & \bar{g}\{x_1(k_f), x_2(k_f), \ldots, x_n(k_f), u_1(k_f), u_2(k_f), \ldots, u_m(k_f)\} 
\end{cases}
\tag{3.2}
\]
In many real-world systems the function $\vec{g}$ may be unknown or unknowable. However, by exciting the system using known inputs, a significant amount of state and state-derivative data may be collected through direct measurement or estimation. Could this time-history data be used to approximate $\vec{g}$? How accurate could this approximation be made?

### 3.2 Data-Driven System Identification with SINDYc

Nonlinear system identification using time history data is precisely the aim of SINDYc (Sparse Identification of Nonlinear Dynamics with Control), a method proposed by Brunton, Proctor, and Kutz in their 2016 papers [10][11]. In order to employ this method some rearrangement of the data is required.

$$\dot{X} \equiv \begin{bmatrix} \dot{x}_1(1) & \dot{x}_2(1) & \ldots & \dot{x}_n(1) \\ \dot{x}_1(2) & \dot{x}_2(2) & \ldots & \dot{x}_n(2) \\ \vdots & \vdots & \ddots & \vdots \\ \dot{x}_1(k_f) & \dot{x}_2(k_f) & \ldots & \dot{x}_n(k_f) \end{bmatrix}$$

$$X \equiv \begin{bmatrix} x_1(1) & x_2(1) & \ldots & x_n(1) \\ x_1(2) & x_2(2) & \ldots & x_n(2) \\ \vdots & \vdots & \ddots & \vdots \\ x_1(k_f) & x_2(k_f) & \ldots & x_n(k_f) \end{bmatrix}$$

Each column of these data matrices corresponds to a specific state and each row represents a moment in discrete time, increasing downwards – resulting in $k_f$ rows and $n$ columns in each. A data matrix of $m$ inputs may be similarly assembled.

$$U \equiv \begin{bmatrix} u_1(1) & u_2(1) & \ldots & u_m(1) \\ u_1(2) & u_2(2) & \ldots & u_m(2) \\ \vdots & \vdots & \ddots & \vdots \\ u_1(k_f) & u_2(k_f) & \ldots & u_m(k_f) \end{bmatrix}$$

The key feature to recognize here is that every row in $[\dot{X}]$ is related to every corresponding row in $[X]$ and $[U]$ through the nonlinear dynamics described by $\vec{g}$. This is expressed explicitly as
follows.

\[ \{ \dot{x}_1(k), \dot{x}_2(k), \ldots, \dot{x}_n(k) \} = \vec{g} \{ x_1(k), x_2(k), \ldots, x_n(k), u_1(k), u_2(k), \ldots, u_m(k) \} \]  (3.3)

for all \( k \in [0, k_f] \)

In other words, if a vector function could be constructed that satisfies Equation 3.3 at every \( k \), within a margin of error, that vector function may represent a strong approximation of the nonlinear dynamics. However, this model may only be reliable in the regions of state-space explored by the time-history data. The SINDYc algorithm provides a methodology to assemble such a candidate function.

To do this, a library of simple candidate functions, \( [\Theta] \), must be assembled from which the algorithm will activate the ones that most closely satisfy Equation 3.3. This library includes constant terms, nonlinear monomial combinations of the state and input variables, and the state and input variables themselves. As these values are known from the time-history data, one set of candidate functions is evaluated for each time step, \( k \).

\[ \Theta(x, u) = \begin{bmatrix} 1 & \vec{x} & \vec{u} & x_1^2 & x_2^2 & x_3^2 & \cdots & x_1x_2 & x_2x_3 & \cdots & x_2^3 & x_3^3 & \cdots & x_3^2u_1 & x_3^2u_2 & \cdots \end{bmatrix} \]  (3.4)

\[ [\Theta(X, U)] = \begin{bmatrix} 1 & \vec{x} & \vec{u} & x_1^2 & x_2^2 & x_3^2 & \cdots & x_1x_2 & x_2x_3 & \cdots & x_2^3 & x_3^3 & \cdots & x_3^2u_1 & x_3^2u_2 & \cdots \\ 1 & \vec{x} & \vec{u} & x_1^2 & x_2^2 & x_3^2 & \cdots & x_1x_2 & x_2x_3 & \cdots & x_2^3 & x_3^3 & \cdots & x_3^2u_1 & x_3^2u_2 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & \vec{x} & \vec{u} & x_1^2 & x_2^2 & x_3^2 & \cdots & x_1x_2 & x_2x_3 & \cdots & x_2^3 & x_3^3 & \cdots & x_3^2u_1 & x_3^2u_2 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix} \]  (3.5)

While \([\Theta]\) has a finite number of rows, \( k_f \), the number and nature of the columns are only limited by one’s imagination (and RAM); therefore, selection of the monomials must somehow be constrained. In selecting these candidate functions, knowledge of the true system behavior will
prove especially useful. In fact, if no combination of terms form a basis of the true nonlinear
dynamics, this method will produce an inadequate model approximation [10].

To determine which terms contribute to the dynamics, a system of linear equations is as-
sembled using \([\dot{X}]\) and \([\Theta]\).

\[
\begin{bmatrix}
\dot{X}
\end{bmatrix}_{k_f \times n} =
\begin{bmatrix}
\Theta(X, U)
\end{bmatrix}_{k_f \times p}
\begin{bmatrix}
\xi_{1, x_1} & \xi_{1, x_2} & \cdots & \xi_{1, x_n} \\
\xi_{2, x_1} & \xi_{2, x_2} & \cdots & \xi_{2, x_n} \\
\vdots & \vdots & \ddots & \vdots \\
\xi_{x_1, x_1} & \xi_{x_1, x_2} & \cdots & \xi_{x_1, x_n} \\
\vdots & \vdots & \cdots & \vdots \\
\xi_{x_2, x_1} & \xi_{x_2, x_2} & \cdots & \xi_{x_2, x_n} \\
\vdots & \vdots & \cdots & \vdots \\
\xi_{x_n, x_1} & \xi_{x_n, x_2} & \cdots & \xi_{x_n, x_n}
\end{bmatrix}_{p \times n}
\]

or compactly

\[
\begin{bmatrix}
\dot{X}
\end{bmatrix} = \begin{bmatrix}
\Theta(X, U)
\end{bmatrix} \begin{bmatrix}
\Xi
\end{bmatrix} \tag{3.6}
\]

Here \([\Xi]\) is an array of potential coefficients to the monomials in \([\Theta]\) and \(p\) is the number
of candidate monomials. Recall that \([\dot{X}]\) and \([\Theta]\) are not variables here, \textit{they are arrays of known
values}. The subscripts of each \(\xi\) term corresponds to a candidate function and a state equation,
respectively. To solve for the columns of \([\Xi]\), a system of equations is created for each state variable
using the least squares method.

\[
\begin{bmatrix}
\Theta(X, U)
\end{bmatrix}^T \begin{bmatrix}
\dot{x}_1
\end{bmatrix} = \begin{bmatrix}
\Theta(X, U)
\end{bmatrix}^T \begin{bmatrix}
\Theta(X, U)
\end{bmatrix} \begin{bmatrix}
\xi_1
\end{bmatrix}
\]

\[
\begin{bmatrix}
\Theta(X, U)
\end{bmatrix}^T \begin{bmatrix}
\dot{x}_2
\end{bmatrix} = \begin{bmatrix}
\Theta(X, U)
\end{bmatrix}^T \begin{bmatrix}
\Theta(X, U)
\end{bmatrix} \begin{bmatrix}
\xi_2
\end{bmatrix}
\]

\[
\vdots
\]

\[
\begin{bmatrix}
\Theta(X, U)
\end{bmatrix}^T \begin{bmatrix}
\dot{x}_n
\end{bmatrix} = \begin{bmatrix}
\Theta(X, U)
\end{bmatrix}^T \begin{bmatrix}
\Theta(X, U)
\end{bmatrix} \begin{bmatrix}
\xi_n
\end{bmatrix} \tag{3.7}
\]
The subscripts on $\dot{x}$ and $\tilde{\xi}$ represent the corresponding columns of $[\bar{X}]$ and $[\Xi]$. For example $\dot{x}_1$ and $\tilde{\xi}_1$ represent the first columns of $[\bar{X}]$ and $[\Xi]$, respectively. Since the systems of 3.6 contain $p$ equations of $p$ variables, each can be solved for a column of $[\Xi]$ using a numerical linear system solver.

Almost every element of $[\Xi]$ will be some nonzero value, even though only a relatively small number of terms in $[\Theta]$ contribute meaningfully to the system dynamics. Therefore every element in $[\Xi]$ not greater than some threshold is eliminated, as described in section 3.2.2. Equation 3.6 is solved again with the columns of $[\Theta]$ and the rows of $[\Xi]$ removed as required. This process is repeated until all remaining terms in $[\Xi]$ meet or exceed the minimum threshold. The resulting matrix, $[\Xi^*]$, contains the coefficients of terms in $[\Theta]$ that together generate a sparse, data-driven model of the nonlinear dynamics of $\vec{g}$. Explicitly, this is expressed as follows.

$$
\dot{x}_1 = g_1(\vec{x}, \vec{u}) \approx \Theta(\vec{x}, \vec{u})\tilde{\xi}_1^*
\dot{x}_2 = g_2(\vec{x}, \vec{u}) \approx \Theta(\vec{x}, \vec{u})\tilde{\xi}_2^*
\vdots
\dot{x}_n = g_n(\vec{x}, \vec{u}) \approx \Theta(\vec{x}, \vec{u})\tilde{\xi}_n^*
$$

(3.8)

Note that the terms $\dot{x}$, $\vec{x}$, $\vec{u}$ are symbolic variables and $n$ is the number of states. Additionally $\Theta$ is a symbolic row vector of monomial terms, that, when multiplied by the column vector $\tilde{\xi}^*$, results in the nonlinear approximations of $\vec{g}$.

3.2.1 Data Library Normalization

Certain systems may possess states that, even under strong input, are always very small in magnitude. As a result some elements of the library matrix $[\Theta]$, specifically products of small variables, could be so minute as to be problematic in the linear regression. In these cases it may be beneficial to normalize the columns of $[\Theta]$ using the $L^2$ norm as follows, where $\bar{\Theta}_1$, $\bar{\Theta}_2$, and $\bar{\Theta}_p$
are the first, second, and last columns of $[\Theta]$, respectively.

$$
[\Theta(X, U)]_{\text{norm}} = \left[ \frac{1}{\|\Theta_1\|_2} \Theta_1 \quad \frac{1}{\|\Theta_2\|_2} \Theta_2 \quad \ldots \quad \frac{1}{\|\Theta_p\|_2} \Theta_p \right]
$$

(3.9)

Following this normalization, the system of equations that SINDYc explicitly solves is modified as follows.

$$
\dot{X} = [\Theta(X, U)]_{\text{norm}} \Xi_{\text{norm}}
$$

(3.10)

Notice the matrix $[\Xi]$ also requires a subscript 'norm' because as each column of $[\Theta]$ was scaled down by a factor of $\|\Theta_i\|_2$, each resulting row of $[\Xi]$ is conversely scaled up by the same factor. Therefore to recover the proper elements of $[\Xi]$, each row must be rescaled by its corresponding normalization factor as follows.

$$
[\Xi] = \left[ \frac{1}{\|\Theta_1\|_2} \xi_{R1} \frac{1}{\|\Theta_2\|_2} \xi_{R2} \ldots \frac{1}{\|\Theta_p\|_2} \xi_{Rp} \right]
$$

(3.11)

### 3.2.2 The Sparsity-Promoting Parameter

As part of the SINDYc algorithm, the solutions to Equations 3.6 often return many small or insignificant terms of $[\Xi]$ that are not present in the true dynamics. To resolve this, a method is required to eliminate these unnecessary terms, leaving only the sparse subset of terms that truly contributes to the nonlinear dynamics. The vector $\tilde{\lambda}$ defines the coefficient threshold for each equation of state, below which coefficients of $[\Xi]$ are eliminated and the linear regression repeated.

$$
\tilde{\lambda} \equiv \left\{ \lambda_{x_1}, \lambda_{x_2}, \lambda_{x_3}, \ldots, \lambda_{x_n} \right\}
$$

(3.12)

If the data library $[\Theta]$ does not require normalization, the terms in $\tilde{\lambda}$ are applied directly to the resulting elements of the coefficient matrix $[\Xi]$ to promote sparsity. However, if normalization is performed as in Equation 3.9, the terms in $\tilde{\lambda}$ are applied to $[\Xi]_{\text{norm}}$, and consequently loose their
intuitive meaning. Whatever the case, the selection of $\mathbf{\lambda}$ is often an iterative process, requiring reasonable judgment in determining whether a resultant matrix $[\mathbf{\Xi}]$ is optimally sparse.

In practice, the initial resulting $[\mathbf{\Xi}]$ matrix, as a solution to Equation 3.6, contains nonzero elements throughout. However, terms that characterize the true dynamics of the system are generally orders of magnitude greater than insignificant terms. Therefore elements of the sparsity-promoting vector $\mathbf{\lambda}$ do not need to be large in order to eliminate small terms of $[\mathbf{\Xi}]$. However, if $\mathbf{\lambda}$ is chosen too small, the resulting $[\mathbf{\Xi}]$ will contain insignificant terms, and, if too large, the resulting $[\mathbf{\Xi}]$ will omit terms contributing to the dynamics. Provided the nonlinear system is truly sparse in the space of possible functions, there should exist considerable margin between a $\mathbf{\lambda}$ that is too small and one that is too large. This feature may be exploited in order to judge the resulting $[\mathbf{\Xi}]$ matrix for optimal sparsity. However, if true terms of $[\mathbf{\Xi}]$ are sufficiently small, they may not be distinguishable from insignificant terms, resulting in their omission from the identified model.

3.3 A Simple Example

Consider the following nonlinear system of ODEs in two variables, $x_1$ and $x_2$, and a single input, $u$.

\[
\begin{align*}
\dot{x}_1 &= -2x_2 + 1.4x_1^2 - 2.1u \\
\dot{x}_2 &= 4.5x_1 - 0.4x_2 - 2x_1x_2
\end{align*}
\] (3.13)

To apply the SINDYc algorithm, the system must be excited using some time-varying control input. The following function describes a amplitude-modulated control input and is plotted in yellow in Figure 3.13.

\[ u(t) = 2\cos(t)\sin(8t) \] (3.14)

The choice of input function is critical to proper system identification. The input must drive the dynamics in a way that provokes the nonlinear behavior so their effect is captured within the time-history data. Starting from rest with $\{x_1, x_2\} = \{0, 0\}$ and applying this control input to the
system for ten seconds and a time step of 0.01 seconds results in the response shown in Figure 3.1. This response is collected into the appropriate data matrices, the first few rows of which are provided as follows. Note that the initial conditions, $\vec{x}(0) = \vec{0}$, are omitted from the data matrices as they do not reflect the dynamics.

$$
\dot{X} = \begin{bmatrix}
-0.3356 & -0.075 \\
-0.6685 & -0.0301 \\
-0.9962 & -0.0674 \\
\vdots & \vdots 
\end{bmatrix}
\quad
X = \begin{bmatrix}
-0.0017 & 0.0000 \\
-0.0067 & -0.0002 \\
-0.0150 & -0.0007 \\
\vdots & \vdots 
\end{bmatrix}
\quad
U = \begin{bmatrix}
0.1598 \\
0.3186 \\
0.4752 \\
\vdots 
\end{bmatrix}
$$

(3.15)

Next the matrix library of monomial functions, $[\Theta]$, must be constructed. The library will be limited to terms of degree two and lower. Each row of $[\Theta]$ will be evaluated as follows and the first several rows of the library matrix are provided. For this example $[\Theta]$ will not require normalization.

$$
\Theta(x_1, x_2, u) = \begin{bmatrix}
1 & x_1 & x_2 & u & x_1^2 & x_2^2 & u^2 & x_1x_2 & x_1u & x_2u 
\end{bmatrix}
$$

(3.16)
\[
\begin{bmatrix}
\Theta(X, U)
\end{bmatrix} = \begin{bmatrix}
100 & -0.17 & 0.0 & 16.0 & 0.00 & 0.0 & 2.6 & 0.0 & -0.03 & 0.00 \\
100 & -0.67 & -0.02 & 31.9 & 0.00 & 0.0 & 10.2 & 0.0 & -0.21 & -0.01 \\
100 & -1.50 & -0.07 & 47.5 & 0.02 & 0.0 & 22.6 & 0.0 & -0.74 & -0.03 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
\end{bmatrix} \times 10^{-2}
\]

Assembling these matrices as prescribed by Equation 3.6 will result in two systems of equations – one for \(x_1\) and another for \(x_2\). The combined system is given as follows.

\[
\dot{X} = \begin{bmatrix}
\Theta(X, U)
\end{bmatrix} \Xi
\]

\[
\begin{bmatrix}
-0.3356 & -0.0075 \\
-0.6685 & -0.0301 \\
-0.9962 & -0.0674 \\
\vdots & \vdots \\
\end{bmatrix} = \begin{bmatrix}
100 & -0.17 & 0.0 & 16.0 & 0.00 & 0.0 & 2.6 & 0.0 & -0.03 & 0.00 \\
100 & -0.67 & -0.02 & 31.9 & 0.00 & 0.0 & 10.2 & 0.0 & -0.21 & -0.01 \\
100 & -1.50 & -0.07 & 47.5 & 0.02 & 0.0 & 22.6 & 0.0 & -0.74 & -0.03 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
\end{bmatrix} \times 10^{-2}
\]

\[
\begin{bmatrix}
\xi_{1,\dot{x}_1} & \xi_{1,\dot{x}_2} \\
\xi_{x_1,\dot{x}_1} & \xi_{x_1,\dot{x}_2} \\
\xi_{x_2,\dot{x}_1} & \xi_{x_2,\dot{x}_2} \\
\xi_{u,\dot{x}_1} & \xi_{u,\dot{x}_2} \\
\vdots & \vdots \\
\xi_{x_2,\dot{x}_1} & \xi_{x_2,\dot{x}_2} \\
\end{bmatrix}
\times \begin{bmatrix}
\xi_{1,\dot{X}} \\
\xi_{x_1,\dot{X}} \\
\xi_{x_2,\dot{X}} \\
\xi_{u,\dot{X}} \\
\vdots \\
\xi_{x_2,\dot{X}} \\
\end{bmatrix}
\]

The subscripts on each \(\xi\) term correspond to the column of \([\Theta]\) and the associated state variable, respectively. Evaluating both sides, the result is a system of ten equations and ten unknowns.
The system above only applies to $x_1$, a similar system is associated with $x_2$. Using a numerical solver to compute $\vec{\xi}_{\dot{x}_1}$ and $\vec{\xi}_{\dot{x}_2}$, the following results.

$$
\vec{\xi}_{\dot{x}_1} = \begin{bmatrix}
\xi_{x_1 \dot{x}_1} \\
\xi_{x_1 \dot{x}_1} \\
\xi_{x_2 \dot{x}_1} \\
\xi_{u \dot{x}_1} \\
\xi_{x_1^2 \dot{x}_1} \\
\xi_{x_2^2 \dot{x}_1} \\
\xi_{x_1 x_2 \dot{x}_1} \\
\xi_{x_1 u \dot{x}_1} \\
\xi_{x_2 u \dot{x}_1}
\end{bmatrix} = \begin{bmatrix}
0.00 \\
0.00 \\
-2.00 \\
-2.10 \\
1.40 \\
0.00 \\
0.00 \\
0.00 \\
0.00
\end{bmatrix}
\begin{bmatrix}
1 \\
x_1 \\
x_2 \\
u \\
x_1^2 \\
x_2^2 \\
x_1 x_2 \\
x_1 u \\
x_2 u
\end{bmatrix}
= \begin{bmatrix}
0.00 \\
4.50 \\
-0.40 \\
0.00 \\
0.00 \\
0.00 \\
-2.00 \\
0.00 \\
0.00
\end{bmatrix}
\begin{bmatrix}
\xi_{1, x_1} \\
\xi_{x_1, x_1} \\
\xi_{x_2, x_1} \\
\xi_{u, x_1} \\
\xi_{x_1^2, x_1} \\
\xi_{x_2^2, x_1} \\
\xi_{x_1 x_2, x_1} \\
\xi_{x_1 u, x_1} \\
\xi_{x_2 u, x_1}
\end{bmatrix}
= \begin{bmatrix}
\xi_{1, x_2} \\
\xi_{x_1, x_2} \\
\xi_{x_2, x_2} \\
\xi_{u, x_2} \\
\xi_{x_1^2, x_2} \\
\xi_{x_2^2, x_2} \\
\xi_{x_1 x_2, x_2} \\
\xi_{x_1 u, x_2} \\
\xi_{x_2 u, x_2}
\end{bmatrix} = \begin{bmatrix}
0.00 \\
4.50 \\
-0.40 \\
0.00 \\
0.00 \\
0.00 \\
-2.00 \\
0.00 \\
0.00
\end{bmatrix} \begin{bmatrix}
1 \\
x_1 \\
x_2 \\
u \\
x_1^2 \\
x_2^2 \\
x_1 x_2 \\
x_1 u \\
x_2 u
\end{bmatrix}
$$

A comparison of $\vec{\xi}_{\dot{x}_1}$ and $\vec{\xi}_{\dot{x}_2}$ with Equations 3.13 reveals the SINDYc algorithm deter-
mined the active terms of $[\Theta]$ and their associated coefficients after a single iteration – no iterative sparsification is required.
CHAPTER 4
AIRCRAFT SYSTEM IDENTIFICATION USING SINDYC

The SINDYc algorithm is used to identify the nonlinear system of ODEs that model the
dynamic behavior of the SIAI-Marchetti S-211. This set of equations is restated as an array – the form
of the SINDYc result – allowing direct comparison of the true system to the algorithmic results.
Each column of $\Xi_{\text{true}}$ contains the coefficients associated with one state differential equation.

$$\begin{align*}
\Xi_{\text{true}, \text{reduced}} =
\begin{bmatrix}
\begin{array}{cccccccccc}
\dot{u} & \dot{\beta} & \dot{\alpha} & \dot{p} & \dot{q} & \dot{r} & \dot{\phi} & \dot{\theta} \\
\beta & 0 & -0.2429 & 0 & -91.31 & 0 & 24.29 & 0 & 0 \\
\alpha & 2.761e-3 & 0 & -1.965 & 0 & -3.092 & 0 & 0 & 0 \\
p & 0 & -7.666e-3 & 0 & -7.515 & 0 & -0.2890 & 1.000 & 0 \\
q & 0 & 0 & 0.9999 & 0 & -11.56 & 0 & 0 & 1.000 \\
r & 0 & -1.000 & 0 & -0.1927 & 0 & -0.7708 & -0.01008 & 0 \\
\phi & 0 & 0.05279 & 0 & 0 & 0 & 0 & 0 & 0 \\
\theta & -0.05279 & 0 & 5.318e-4 & 0 & 0 & 0 & 0 & 0 \\
\delta_{\beta} & 0 & 0 & 89.28 & 0 & 3.025 & 0 & 0 & 0 \\
\delta_{\alpha} & -2.558e-4 & 0 & -0.03337 & 0 & -5.672 & 0 & 0 & 0 \\
\delta_{\alpha} & 0 & 9.952e-3 & 0 & 40.57 & 0 & -14.78 & 0 & 0 \\
uq & 7.666e-3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\beta p & 0 & 0 & -1.000 & 0 & 0 & 0 & 0 & 0 \\
\alpha^2 & 1.316 & 0 & -0.1411 & 0 & 1.130 & 0 & 0 & 0 \\
\alpha p & 0 & 1.000 & 0 & 0 & 0 & 0 & 0 & 0 \\
aq & 0 & 0 & 0.01533 & 0 & 0 & 0 & 0 & 0 \\
\alpha^2 q & 0.03337 & 0 & -7.673e-4 & 0 & -0.04348 & 0 & 0 & 0 \\
p^2 & 0 & 0 & 0 & 0 & -0.04167 & 0 & 0 & 0 \\
pq & 0 & 0 & 0 & 0.05825 & 0 & -0.7670 & 0 & 0 \\
pr & 0 & 0 & 0 & 0 & 0.9167 & 0 & 0 & 0 \\
qr & 0 & 0 & 0 & -0.5146 & 0 & -0.05825 & 0 & 0 \\
q\phi & 0 & 0 & 0 & 0 & 0 & -0.01008 & 0 & 0 \\
r^2 & 0 & 0 & 0 & 0.04167 & 0 & 0 & 0 & 0 \\
r\phi & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1.000 \\
r\theta & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\phi^2 & 0 & 0 & 0 & 0 & -0.02639 & 0 & 0 & 0 \\
\phi \theta & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\theta^2 & -2.659e-4 & 0 & -0.02639 & 0 & 0 & 0 & 0 & 0 \\
uq r & 1.000 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
uq \alpha q & -1.000 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\beta^3 & 0 & -0.6509 & 0 & -13.50 & 0 & -54.01 & 0 & 0 \\
\beta^2 r & 0 & 1.000 & 0 & 0 & 0 & 0 & 0 & 0 \\
\beta^2 \phi & 0 & -0.05279 & 0 & 0 & 0 & 0 & 0 & 0 \\
\beta^2 \delta_{\alpha} & 0 & -9.952e-3 & 0 & 0 & 0 & 0 & 0 & 0 \\
\alpha^3 & 0.1118 & 0 & 6.314 & 0 & -48.62 & 0 & 0 & 0 \\
\alpha^2 q & 0 & 0 & -1.000 & 0 & 0 & 0 & 0 & 0 \\
\alpha^2 \delta_{\alpha} & 0 & 0 & 0.05005 & 0 & 2.836 & 0 & 0 & 0 \\
q \phi^2 & 0 & 0 & 0 & 0 & 0 & 0 & -0.5000 & 0 \\
q \phi \theta & 0 & 0 & 0 & 0 & 0 & 0 & 1.000 & 0 \\
\phi^3 & 0 & -8.798e-3 & 0 & 0 & 0 & 0 & 0 & 0 \\
\phi \theta^2 & 0 & -0.02639 & 0 & 0 & 0 & 0 & 0 & 0 \\
\theta^3 & 8.798e-3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 
\end{array}
\end{bmatrix}
\end{align*}$$

(4.1)
The matrix \( \Xi_{\text{true}} \) expresses the true dynamics in terms of a matrix of coefficients expressed to four significant figures. While the actual coefficient matrix contains 264 rows, corresponding to all combinations of state and input variables, the rows are composed of all zeros and have been omitted for compactness, hence the 'reduced' subscript.

4.1 System Inputs

Selection of control inputs is a critical step for proper system identification using the SINDYc algorithm. In order for this method to accurately recognize the nonlinearities present, a control input must sufficiently excite the nonlinear dynamics – these behaviors must exist in the state response data. This is most readily accomplished by using a control input that is aggressive, oscillatory, and modulating. However, this manner of input does present significant practical challenges regarding S-211 model.

First, the trigonometric approximations of Equations 2.16 impose limitations on \( \beta, \alpha, \phi, \) and \( \theta \) that, if exceeded, compromise model validity. The errors associated with these approximations are discussed in Appendix A. Second, the model becomes unstable in airspeed and pitch at high angles of roll, a problem compounded by the S-211’s sensitivity to aileron and rudder deflections. Finally, as the nonlinear model assumes constant dynamic pressure, its integrity declines as the model airspeed deviates from the trimmed airspeed. Therefore the control inputs must maintain airspeed within a neighborhood around trim.

Each of these input policies contain several important features. Obviously they are oscillatory as their purpose is to excite dynamic modes within the system. Their magnitude is chosen to be sufficiently strong as to excite the state variables well into their nonlinear regions. As discussed in Appendix G, the lift and side force models approximate linear behavior at low angles, therefore the magnitude of the control input is chosen to produce a state response in excess of this linear region. Second, each input occurs at a different frequency and they are mutually phase shifted by \( \frac{1}{6} \) cycle. This is to ensure the effect of each input is distinct from the others. Consider the extreme case where all inputs are identical. This would poorly condition the SINDYc algorithm because the individual effect of each input is indistinguishable. Finally, to enhance the variability and rich-
ness of the control inputs, each is amplitude modulated. This was done to ensure the dynamics are not subject to a pure frequency input, further fostering richness and variability in the dynamic response.

This final set of control inputs is developed by incrementally increasing the intensity of sinusoidal, amplitude-modulated, phase-shifted surface deflections until dynamic behavior results that is both stable and within the limitations of the dynamic model. The control policy is described as follows, and its form is shown in Figure 4.1.

\[
\begin{align*}
\delta_a(t) &= 0.5 \cos(t) \cdot 0.4 \sin \left( 6 \left( t + \frac{\pi}{3} \right) \right) \\
\delta_e(t) &= 0.4 \cos(t) \cdot 0.8 \sin (4t) + 0.11 \\
\delta_r(t) &= -0.4 \cos(t) \cdot 0.8 \sin \left( 9 \left( t + \frac{2\pi}{3} \right) \right)
\end{align*}
\]

Figure 4.1: Control Inputs for S-211 System Identification

These input functions provide adequate excitation for identification and are used for each variation of the SINDYc algorithm to follow. This way, the algorithm’s performance may be
meaningfully compared and evaluated without attributing variation to differences in control inputs. For use in the SINDYc algorithm, this input data is assembled into a data matrix, \[ U \].

### 4.2 System ID – No Noise, Exact Derivatives

Application of control input functions produces the dynamic response as shown in Figure 4.2. The dynamic equations were integrated over 30 seconds using a step of 0.001 second, resulting in 30,000 discrete time steps. The model was initialized from the condition \( \{ \hat{u}, \beta, \alpha, p, q, r, \phi, \theta \} = \{1, 0, 0, 0, 0, 0, 0, 0\} \). Recall \( \hat{u} = 1 \) corresponds to an airspeed equal to the trim airspeed, \( u = U_1 \).

This data is assembled into a state data matrix \[ X \]. In this application of the SINDYc algorithm, the associated state derivative data matrix, \[ \dot{X} \], is constructed using the exact derivatives provided by the nonlinear state equations described in Equation 4.1. No noise is superimposed upon the state data.

The library of candidate functions, \[ \Theta(X, U) \], will be constructed using combinations of state and input variables up to and including cubic monomials. Given the eight state variables and three inputs, the number of resulting candidate functions is 364. The first and last terms of \( \Theta(\vec{x}, \vec{u}) \) are provided as follows, where \( \vec{x} = \{ \hat{u}, \alpha, \beta, p, q, r, \phi, \theta \} \) and \( \vec{u} = \{ \delta_a, \delta_e, \delta_r \} \). Likely a result of

![Figure 4.2: S-211 Dynamic System Response, No Noise](image-url)
cubic combinations of small state variables, the linear regression within SINDYc is ill conditioned unless $[\Theta]$ is normalized as described by Equation 3.9.

$$\Theta(\vec{x}, \vec{u}) = \begin{bmatrix}
1 & \vec{x} & \vec{u} & \vec{u}^2 & \hat{u} & \beta & \hat{u} & \alpha & \hat{u} & \beta & \hat{u} & \alpha & \hat{u} & \beta & \ldots & \delta_a \delta_e \delta_r & \delta_a \delta_r^2 & \delta_e^3 & \delta_r^2 \delta_e & \delta_e \delta_r^2 & \delta_r^3
\end{bmatrix}_{1 \times 364}$$ (4.3)

Using these data matrices, the SINDYc algorithm was applied and achieved successful system identification after two sparsification iterations. SINDYc was able to identify all active terms and their coefficients while returning no false positives for inactive terms. The resulting coefficient matrix, $[\Xi]_{\text{result}}$; the matrix of errors, $[E]$; and its 2-norm, $\|E\|_2$, are provided as follows. Rows of all zeros have been removed.
\[ \begin{array}{cccccccccc}
\dot{\beta} & \dot{\alpha} & \dot{p} & \dot{q} & \dot{r} & \phi & \theta \\
2.761e-3 & -91.31 & 0 & 24.29 & 0 & 0 & 0 \\
-7.666e-3 & 0 & -3.092 & 0 & 0 & 0 & 0 \\
0 & 9.952e-3 & 0 & 40.57 & 0 & -14.78 & 0 & 0 \\
7.666e-3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-1.000 & 0 & -0.927 & 0 & -0.7708 & -0.01008 & 0 & 0 \\
0.05279 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-0.05279 & 0 & 5.316e-4 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 89.28 & 0 & 3.025 & 0 & 0 & 0 \\
-2.558e-4 & 0 & -0.03337 & 0 & -5.672 & 0 & 0 & 0 \\
-9.952e-3 & 0 & 40.57 & 0 & -14.78 & 0 & 0 & 0 \\
7.666e-3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & -1.000 & 0 & 0 & 0 & 0 & 0 \\
1.316 & 0 & -0.1411 & 0 & 1.130 & 0 & 0 & 0 \\
1.000 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0.01533 & 0 & -7.673e-4 & 0 & -0.04348 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & -0.04167 & 0 & 0 & 0 \\
0.05825 & 0 & 0.9167 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & -0.5146 & 0 & -0.05825 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & -0.01008 & 0 & 0 \\
0.04167 & 0 & 0 & 0 & 0 & 0 & -1.000 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-0.02639 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
5.316e-4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
2.559e-4 & 0 & -0.02639 & 0 & 0 & 0 & 0 & 0 \\
1.000 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-1.000 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-0.6509 & 0 & -13.50 & 0 & -54.01 & 0 & 0 & 0 \\
1.000 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-0.05279 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-9.952e-3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0.1118 & 0 & 6.314 & 0 & -48.62 & 0 & 0 & 0 \\
0 & 0 & -1.000 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0.05005 & 0 & 2.836 & 0 & 0 & 0 \\
-0.02639 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-8.798e-3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-0.02639 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
8.798e-3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\end{array} \]
The SINDYc algorithm could only provide accurate results if sufficient state and input data had been collected. Specifically, if the system is excited for too short a time, the resulting linear systems, as described by Equations 3.6, are rank degenerate and a solution impossible. This feature of the algorithm may provide a criteria for sufficient data collection whereby the point at which the system achieves full rank, SINDYc may be successfully implemented.
4.3 System ID – No Noise, Central Differences

This identification is repeated using central differences in place of exact derivatives within the state derivative data matrix, \([\dot{X}]\). This method of approximating the derivative is more reflective of real-world applications where the explicit dynamics are unknown or the state derivatives are not directly measured. The four-point central-difference formula is provided as follows.

\[
\dot{x}(k) \approx -\frac{x(k+2) + 8x(k+1) - 8x(k-1) + x(k-2)}{12\Delta t}
\]  
(4.7)

To use this approximation, the first and last two entries of the time history were omitted from the data matrices as their central difference is undefined.

Using approximate derivatives, the SINDYc algorithm is applied and achieves successful system identification after three sparsification iterations. SINDYc is able to identify all active terms and their coefficients. However the algorithm does return four false positives – terms not present in the dynamics, but retained after the sparsification. The resulting coefficient matrix, \(\Xi_{\text{result}}\); the matrix of errors, \(E\); and its 2-norm, \(\|E\|_2\), are provided as follows. Rows of all zeros have been removed.
\[
\begin{array}{cccccccccc}
\text{result, reduced} & & & & & & & & & \\
\beta & 0 & -0.3429 & 0 & -91.31 & 0 & 24.29 & 0 & 0 \\
\alpha & 2.761e-3 & 0 & -1.965 & 0 & -3.092 & 0 & 0 & 0 \\
p & 0 & -7.666e-3 & 0 & -7.515 & 0 & -0.289 & 1 & 0 \\
q & 0 & 0 & 0.9999 & 0 & -11.56 & 0 & 0 & 1 \\
r & 0 & -1 & 0 & -0.1927 & 0 & -0.7708 & -0.01008 & 0 \\
\phi & 0 & 0.05279 & 0 & 0 & 0 & 0 & 0 & 0 \\
\theta & -0.05279 & 0 & 5.320e-4 & 0 & 0 & 0 & 0 & 0 \\
\delta_r & 0 & 0 & 0 & 89.28 & 0 & 3.025 & 0 & 0 \\
\delta_\phi & -2.558e-4 & 0 & -0.03337 & 0 & -5.672 & 0 & 0 & 0 \\
\delta_\theta & 0 & 9.952e-3 & 0 & 40.57 & 0 & -14.78 & 0 & 0 \\
\eta_{\mu} & 7.666e-3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\beta_\rho & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\
\alpha^2 & 1.316 & 0 & -0.1411 & 0 & 1.13 & 0 & 0 & 0 \\
\alpha_\rho & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
\alpha_q & 0 & 0 & 0.01533 & 0 & 0 & 0 & 0 & 0 \\
\alpha_\delta_r & 0.03337 & 0 & -7.673e-4 & 0 & -0.04348 & 0 & 0 & 0 \\
\rho_\gamma & 0 & 0 & 0 & 0 & -0.04167 & 0 & 0 & 0 \\
\rho_\eta & 0 & 0 & 0 & 0.05825 & 0 & -0.767 & 0 & 0 \\
\rho_r & 0 & 0 & 0 & 0.9167 & 0 & 0 & 0 & 0 \\
\eta_{\rho_\gamma} & 0 & 0 & 0 & -0.5146 & 0 & -0.05825 & 0 & 0 \\
\eta_{\rho_\theta} & 0 & 0 & 0 & 0.04167 & 0 & 0 & 0 & 0 \\
\eta_{\rho_\phi} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 \\
\rho_\phi & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\rho_\theta & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
\phi_\phi & 0 & 0 & -0.02639 & 0 & 0 & 0 & 0 & 0 \\
\phi_\theta & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\phi^2 & -2.659e-4 & 0 & -0.02639 & 0 & 0 & 0 & 0 & 0 \\
\eta_{\mu_\phi} & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
\eta_{\mu_\theta} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\beta^2_\beta & 0 & -0.6509 & 0 & -13.5 & 0 & -54.01 & 0 & 0 \\
\beta^2_\alpha & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
\beta^2_p & 0 & -0.05279 & 0 & 0 & 0 & 0 & 0 & 0 \\
\beta^2_{\delta_r} & 0 & -9.952e-3 & 0 & 0 & 0 & 0 & 0 & 0 \\
\alpha^2 & 0.1118 & 0 & 6.314 & 0 & -48.62 & 0 & 0 & 0 \\
\alpha_\delta_r & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\
\alpha_\rho & 0 & 0 & 0.05005 & 0 & 2.836 & 0 & 0 & 0 \\
\alpha_\theta & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -0.5 \\
\phi_\delta & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
\phi_\phi & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\phi_\theta & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\phi^2 & -8.798e-3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\theta^2 & 8.798e-3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\end{array}
\]
\[ [E] = [\Xi]_{\text{true}} - [\Xi]_{\text{result}} = \]

\[
\begin{array}{cccccccc}
\dot{u} & \dot{\beta} & \dot{\alpha} & \dot{\rho} & \dot{\dot{q}} & \dot{\dot{r}} & \dot{\phi} & \dot{\theta} \\
\beta & 0 & 8.8e-10 & 0 & 1.0e-8 & 0 & -7.8e-9 & 0 & 0 \\
\alpha & -3.3e-10 & 0 & -3.0e-9 & 0 & 7.9e-10 & 0 & 0 & 0 \\
\rho & 0 & 6.1e-11 & 0 & 2.0e-10 & 0 & -3.8e-10 & 6.8e-11 & 0 \\
q & 0 & 0 & 1.7e-9 & 0 & -1.8e-8 & 0 & 0 & 0 \\
r & 0 & -2.4e-11 & 0 & 1.5e-9 & 0 & -2.0e-10 & -1.6e-10 & 0 \\
\phi & 0 & 2.7e-11 & 0 & 0 & 0 & 0 & 0 & 0 \\
\theta & 1.7e-11 & 0 & -2.0e-7 & 0 & 0 & 0 & 0 & 0 \\
\delta_r & 1.1e-10 & 0 & 5.6e-10 & 0 & -6.9e-9 & 0 & 0 & 0 \\
\delta_r & 0 & 3.1e-10 & 0 & 1.0e-8 & 0 & -5.7e-9 & 0 & 0 \\
uq & 1.3e-10 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\beta & 0 & 0 & -1.2e-9 & 0 & 0 & 0 & 0 & 0 \\
\alpha^2 & -8.6e-9 & 0 & 3.3e-8 & 0 & 2.0e-8 & 0 & 0 & 0 \\
\alpha^2 & 0 & 1.2e-9 & 0 & 0 & 0 & 0 & 0 & 0 \\
\alpha^2 & 0 & 0 & 2.4e-8 & 0 & 0 & 0 & 0 & 0 \\
\alpha^2 & -8.2e-11 & 0 & 2.7e-8 & 0 & -5.1e-8 & 0 & 0 & 0 \\
p^2 & 0 & 0 & 0 & 0 & -9.6e-11 & 0 & 0 & 0 \\
\phi & 0 & 0 & 0 & 0 & -9.6e-11 & 0 & 0 & 0 \\
\phi & 0 & 0 & 2.0e-10 & 0 & 0 & 0 & 0 & 0 \\
a & 1.8e-9 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
a & -2.7e-9 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
b^3 & 0 & -7.1e-8 & 0 & -3.5e-7 & 0 & 2.4e-7 & 0 & 0 \\
b^7 & 0 & -1.5e-8 & 0 & 0 & 0 & 0 & 0 & 0 \\
b^7 & 0 & 1.0e-8 & 0 & 0 & 0 & 0 & 0 & 0 \\
b^7 & 0 & -9.9e-8 & 0 & 0 & 0 & 0 & 0 & 0 \\
a^8 & -1.8e-7 & 0 & 4.6e-7 & 0 & 3.9e-6 & 0 & 0 & 0 \\
a^8 & 0 & 0 & 2.0e-7 & 0 & 0 & 0 & 0 & 0 \\
a^8 & 0 & 1.3e-8 & 0 & -4.2e-7 & 0 & 0 & 0 & 0 \\
a^8 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\phi & 0 & 0 & 0 & 0 & 3.4e-9 & 0 & 0 & 0 \\
\phi & 0 & -9.3e-11 & 0 & 0 & 0 & 0 & 0 & 0 \\
\phi & 0 & -6.6e-10 & 0 & 0 & 0 & 0 & 0 & 0 \\
\phi & 0 & 1.3e-9 & 0 & 0 & 0 & 0 & 0 & 0 \\
\end{array}
\]

\[(4.9)\]

\[\|E\|_2 = 3.90e - 6 \quad (4.10)\]

Even though the derivatives is this case are approximate, the states themselves are known exactly. The fact that SINDYc provides accurate results under these conditions may indicate the algorithm is somewhat insensitive to small errors in state derivative data.
4.4 System ID – White Noise, Exact Derivatives

This process is repeated using exact derivatives, resulting from the state equations as described by Equation [4.1] but with noise superimposed upon the state data. A normally distributed random number generator is used to create a separate noise signal for each state with an intensity 20% of the state data standard deviation. The system response with noise is shown in Figure 4.3.

![Figure 4.3: S-211 Dynamic System Response, 20% σ Noise](image)

Using this noisy state data, the SINDYc algorithm is applied and achieves successful system identification after two sparsification iterations. SINDYc is able to identify all active terms and their coefficients while returning no false positives for inactive terms. The resulting coefficient matrix, $\Xi_{\text{result}}$; the matrix of errors, $[E]$; and its 2-norm, $\|E\|_2$, are provided as follows. Rows of all zeros have been removed.
\[\begin{array}{cccccccc}
\dot{u} & \dot{\beta} & \dot{\alpha} & \dot{p} & \dot{q} & \dot{r} & \phi & \dot{\theta} \\
\beta & 0 & -0.2429 & 0 & -91.31 & 0 & 24.29 & 0 & 0 \\
\alpha & 2.761e-3 & 0 & -1.965 & 0 & -3.092 & 0 & 0 & 0 \\
p & 0 & -7.666e-3 & 0 & -7.515 & 0 & -0.2890 & 1.000 & 0 \\
q & 0 & 0 & 0.9999 & 0 & -11.56 & 0 & 0 & 1.000 \\
r & 0 & -1.000 & 0 & -0.1927 & 0 & -0.7708 & -0.01008 & 0 \\
\phi & 0 & 0.05279 & 0 & 0 & 0 & 0 & 0 & 0 \\
\theta & -0.05279 & 0 & 5.316e-4 & 0 & 0 & 0 & 0 & 0 \\
\delta_r & 0 & 0 & 0 & 89.28 & 0 & 3.025 & 0 & 0 \\
\delta_p & -2.558e-4 & 0 & -0.03337 & 0 & -5.672 & 0 & 0 & 0 \\
\delta_q & 0 & 9.952e-3 & 0 & 40.57 & 0 & -14.78 & 0 & 0 \\
uq & 7.666e-3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\beta p & 0 & 0 & -1.000 & 0 & 0 & 0 & 0 & 0 \\
\alpha^2 & 1.316 & 0 & -0.1411 & 0 & 1.130 & 0 & 0 & 0 \\
\alpha p & 0 & 1.000 & 0 & 0 & 0 & 0 & 0 & 0 \\
\alpha q & 0 & 0 & 0.01533 & 0 & 0 & 0 & 0 & 0 \\
\alpha \delta e & 0.03337 & 0 & -7.673e-4 & 0 & -0.04348 & 0 & 0 & 0 \\
p^2 & 0 & 0 & 0 & 0 & -0.04167 & 0 & 0 & 0 \\
pq & 0 & 0 & 0 & 0.05825 & 0 & -0.7670 & 0 & 0 \\
pr & 0 & 0 & 0 & 0 & 0.9167 & 0 & 0 & 0 \\
qr & 0 & 0 & 0 & -0.5146 & 0 & -0.05825 & 0 & 0 \\
q\phi & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
r^2 & 0 & 0 & 0 & 0 & 0.04167 & 0 & 0 & 0 \\
r\phi & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1.000 \\
r\theta & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\phi^2 & 0 & 0 & 0 & 0 & 0.02639 & 0 & 0 & 0 \\
\phi \theta & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\theta^2 & -2.659e-4 & 0 & -0.02639 & 0 & 0 & 0 & 0 & 0 \\
u \beta r & 1.000 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
u \alpha q & -1.000 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\beta^3 & 0 & -0.6509 & 0 & -13.50 & 0 & -54.01 & 0 & 0 \\
\beta^2 r & 0 & 1.000 & 0 & 0 & 0 & 0 & 0 & 0 \\
\beta^2 \phi & 0 & -0.05279 & 0 & 0 & 0 & 0 & 0 & 0 \\
\beta^2 \delta e & 0 & -9.952e-3 & 0 & 0 & 0 & 0 & 0 & 0 \\
\alpha^3 & 0.1118 & 0 & 6.314 & 0 & -48.62 & 0 & 0 & 0 \\
\alpha^2 q & 0 & 0 & -1.000 & 0 & 0 & 0 & 0 & 0 \\
\alpha^2 \delta e & 0 & 0 & 0.05005 & 0 & 2.836 & 0 & 0 & 0 \\
\phi^2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -0.5000 \\
\phi \theta & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\phi^3 & 0 & -8.798e-3 & 0 & 0 & 0 & 0 & 0 & 0 \\
\phi \theta^2 & 0 & -0.02639 & 0 & 0 & 0 & 0 & 0 & 0 \\
\theta^3 & 8.798e-3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\end{array}\]
\[ \|E\|_2 = 2.0e - 7 \]  

At first glance, this result may indicate the SINDYc algorithm possesses a robustness to noise. And that might be true, but it should be duly noted that, while noise is superimposed upon the state data, the resulting state derivatives are computed exactly using the nonlinear state equations. Therefore the state equations are being faithfully enforced everywhere throughout the time history even though the state data is noisy.
4.5 System ID – White Noise, Total Variation Regularized Differentiation

For this case, white noise is superimposed upon the state data and the state derivatives are computed numerically. A normally distributed random number generator is used to create a separate noise signal for each state with an intensity 2% of the state data standard deviation. In order to use the SINDYc algorithm under these conditions, the state data must be de-noised and the state derivatives must be numerically computed. A low-pass filter is used to de-noise the state data and a regularized differentiation algorithm is used to compute the state derivatives.

4.5.1 State De-Noising

A low-pass filter is used to de-noise the state data. Recall the noisy state data is shown in Figure 4.3. The filter assumes a sampling frequency of 1000 Hz, matching the time step of 0.001 s, and a passband frequency of 15 Hz. The passband frequency was chosen through an iterative process with the goal of minimizing error between the filtered and clean state data. As a result, signals of frequency greater than 15 Hz are rejected. The resulting error following low-pass filtering is shown in Figure 4.4 with error defined as the difference between the clean state and the de-noised state (following application of the low-pass filter).
Figure 4.4: State Error Following Low-Pass Filtering
4.5.2 Total Variation Regularized Differentiation

Central differences may not be employed to compute the derivative of noisy data since the finite differences used significantly amplify noise. Additionally, traditional de-noising of the state data does not provide sufficient regularization to employ finite difference methods. However, the derivative may be determined using total-variation regularization, a technique within the framework of Tikhonov regularization \cite{17}. This method poses the derivative as the function that minimizes the functional provided as follows.

\[
F(u) = \alpha \int_0^T |u'| + \frac{1}{2} \int_0^T |Au - f|^2
\]  

(4.14)

Within this framework, the parameter $\alpha$ is used to tune the regularization. Larger values of $\alpha$ result in a smoother approximation of the derivative but sacrifice accuracy if too large. The regularized state derivatives and associated errors are shown in Figures 4.5 and 4.6, respectively. Error is defined here as the difference between the exact derivative, computed using the clean state data and the dynamic equations, and the resulting regularized derivative.

![Figure 4.5: State Derivatives, Computed Using Regularized Differentiation](image-url)
Figure 4.6: State Derivative Error Following Regularized Differentiation
4.5.3 SINDYc Results – Full, Unconstrained Library

Under these conditions – 2\% noise and regularized differentiation – the SINDYc algorithm completely fails to identify the nonlinear dynamics of the S-211 aircraft. Specifically, while the algorithm does converge after several iterations, the resulting $\Xi$ matrix contains no nonzero elements – all candidate functions are active for all states equations. Additionally, the sparsity-promoting parameters must be chosen to be extremely high – into the hundreds or thousands – before any sparsification results. Even then, the terms that remain bear no reflection to the true dynamics, often assuming values with very high orders of magnitude.

The algorithm’s failure under these conditions has been determined only following exhaustive strategies in filtering, numerical derivation, and regularization with respect to both the state and state derivative data. This was also attempted using minute noise signals – as low as 0.0001\% – to no avail.

4.5.4 SINDYc Results – Unconstrained Library of Degree 2 Terms

While the SINDYc algorithm is unable to identify the nonlinear dynamics from the full unconstrained library of candidate functions, it is able to identify with some accuracy the active terms using a reduced library of monomials up to and including squared combinations of state and input variables. This reduced library includes 78 candidate functions, down from 364 contained within the full library. Considering only squared terms, eliminating all cubic candidate functions, the target $\Xi$ matrix is provided as follows.
\[\Xi\] degree \leq 2, true, reduced =

<table>
<thead>
<tr>
<th>( \xi )</th>
<th>( \eta )</th>
<th>( \zeta )</th>
<th>( \eta )</th>
<th>( \xi )</th>
<th>( \eta )</th>
<th>( \zeta )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \beta )</td>
<td>0</td>
<td>-0.2429</td>
<td>0</td>
<td>-91.31</td>
<td>0</td>
<td>24.29</td>
</tr>
<tr>
<td>( \alpha )</td>
<td>2.761e-3</td>
<td>0</td>
<td>-1.965</td>
<td>0</td>
<td>-3.092</td>
<td>0</td>
</tr>
<tr>
<td>( \phi )</td>
<td>0</td>
<td>-7.666e-3</td>
<td>0</td>
<td>-7.515</td>
<td>0</td>
<td>-0.289</td>
</tr>
<tr>
<td>( \theta )</td>
<td>0</td>
<td>0.9999</td>
<td>0</td>
<td>11.56</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>( \delta_\eta )</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>5.318e-4</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>( \delta_\phi )</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>89.28</td>
<td>0</td>
<td>3.025</td>
</tr>
<tr>
<td>( \delta_\theta )</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>( \delta_\phi )</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>( \theta^2 )</td>
<td>7.666e-3</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>( \beta )</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>-0.1000</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>( \alpha )</td>
<td>1.316</td>
<td>0</td>
<td>-0.1411</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>( \phi )</td>
<td>0</td>
<td>1.000</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>( \theta )</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>( \delta_\pi )</td>
<td>0.03337</td>
<td>0</td>
<td>-7.673e-4</td>
<td>0</td>
<td>-0.40434</td>
<td>0</td>
</tr>
<tr>
<td>( \delta_\phi )</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>( \delta_\theta )</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>( \delta_\phi )</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>( \delta_\theta )</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>( \beta \beta )</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>( \alpha \alpha )</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>( \phi \phi )</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>( \theta \theta )</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Furthermore, to achieve reasonable results the algorithm was also configured to neglect constant coefficient terms. Applying the SINDYc algorithm under these conditions results in the following system identification.

\[\Xi\] degree \leq 2, result, reduced =

<table>
<thead>
<tr>
<th>( \xi )</th>
<th>( \eta )</th>
<th>( \zeta )</th>
<th>( \eta )</th>
<th>( \xi )</th>
<th>( \eta )</th>
<th>( \zeta )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \beta )</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>-91.17</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>( \alpha )</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>( \phi )</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>-7.504</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>( \theta )</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>( \delta_\alpha )</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>89.03</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>( \delta_\phi )</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>( \delta_\theta )</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>( \delta_\phi )</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>( \theta^2 )</td>
<td>0.03340</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>( \beta )</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>( \beta )</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>( \alpha )</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>( \phi )</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>( \theta )</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

60
Elements of the resulting matrix which closely approximate the true coefficients have been underlined. After five iterations, SINDYc is able to successfully identify the largest coefficients within most of the dynamic equations. However, the algorithm fails to recognize many smaller terms and was unable to accurately identify any terms in the $\dot{u}$ equation. The associated error matrix, $[E]$, and its norm, $\|E\|_2$, are provided as follows.

$$[E] = [\Xi]_{\text{true}} - [\Xi]_{\text{result}} =$$

\[
\begin{array}{cccccccc}
\beta & u & \beta & \alpha & \dot{p} & \dot{q} & \dot{r} & \dot{\phi} \\
0 & 0 & -0.2429 & 0 & -0.1451 & 0 & -0.9282 & 0 \\
\alpha & 0.002761 & -0.3305 & 0 & -3.092 & 0 & 0 & 0 \\
p & 0 & -0.01078 & 0 & 0 & 0 & -0.289 & -0.01218 \\
q & 0 & 0 & -0.07848 & 0 & 0 & 0.2636 & 0 \\
r & 0 & -0.01429 & 0 & -0.1927 & 0 & -0.7708 & -0.01008 \\
\phi & 0.1030 & 0.009391 & 0 & 0 & 0 & -0.3889 & 0 \\
\theta & -0.05279 & 0 & 0.0005318 & 0 & 0 & 0 & 0 \\
\delta_\alpha & 0 & 0 & 0 & 0.2538 & 0 & 3.025 & 0 \\
\delta_\beta & -0.0002558 & 0 & -0.33337 & 0 & 0 & 0 & 0 \\
\delta_p & 0 & 0.009952 & 0 & 0.08725 & 0 & -0.02529 & 0 \\
\delta_q & 0 & 0 & 0 & 0 & 0 & 0.3369 & 0 \\
\delta_r & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\delta_\phi & -0.02685 & 0 & 0 & 0 & 0 & 0 & 0 \\
\delta_\theta & -1.013 & 0 & 0 & 0 & 0 & 0 & 0 \\
\delta_{\alpha^2} & 1.316 & 0 & -0.1411 & 0 & 1.13 & 0 & 0 \\
\delta_{\alpha\beta} & 0.04753 & 0 & 0 & 0 & 0 & 0 & 0 \\
\alpha_\alpha & 0 & 0 & 0.01533 & 0 & 0 & 0 & 0 \\
\alpha_\beta & 0.03337 & 0 & -0.0007673 & 0 & -0.04348 & 0 & 0 \\
\beta_\alpha & 0 & 0 & 0 & 0 & -0.04167 & 0 & 0 \\
\beta_\beta & 0 & 0 & 0.05825 & 0 & -0.767 & 0 & 0 \\
\beta_p & 0 & 0 & 0 & 0 & -0.1660 & 0 & 0 \\
\beta_q & 0 & 0 & 0 & -0.5146 & 0 & -0.05825 & 0 \\
\beta_r & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\beta_\phi & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\beta_\theta & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\theta_\alpha & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\theta_\beta & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\theta_\phi & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\theta_\theta & -0.0002659 & 0 & -0.02639 & 0 & 0.005318 & 0 & 0 \\
\end{array}
\]

$$\|E\|_2 = 3.41$$

Since these differences are significantly larger than previous applications of SINDYc, the complete error matrix is expressed in terms of percent error as follows. Each element of matrix $[4.20]$ is computed using Equation $4.19$. 

\[
\frac{\Delta}{\text{true}} \times 100
dataref{4.19}
\[(e_{i,j})_{\%} = \frac{(\xi_{i,j})_{\text{true}} - (\xi_{i,j})_{\text{result}}}{(\xi_{i,j})_{\text{true}}} \times 100 \] (4.19)

\[\begin{array}{ccccccccccc}
\hat{u} & \hat{\beta} & \alpha & \hat{p} & \hat{q} & \hat{r} & \hat{\phi} & \hat{\theta} \\
\beta & 0 & 100 & 0 & 0.16 & 0 & -3.8 & 0 & 0 \\
\alpha & 100 & 0 & 16.8 & 0 & 100 & 0 & 0 & 0 \\
\hat{p} & 0 & 100 & 0 & 0.14 & 0 & 100 & -1.2 & 0 \\
\hat{q} & 0 & 0 & -7.8 & 0 & -2.3 & 0 & 0 & 21.7 \\
\hat{r} & 0 & 1.4 & 0 & 100 & 0 & 100 & 0 & 0 \\
\phi & \infty & 17.8 & 0 & 0 & 0 & \infty & 0 & 0 \\
\hat{\theta} & 100 & 0 & 100 & 0 & 0 & 0 & 0 & 0 \\
\delta_\alpha & 0 & 0 & 0 & 0.28 & 0 & 100 & 0 & 0 \\
\delta_\beta & 100 & 0 & 100 & 0 & -2.8 & 0 & 0 & 0 \\
\delta_\gamma & 0 & 100 & 0 & 0.22 & 0 & 0.17 & 0 & 0 \\
uq & 100 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
u\phi & \infty & 0 & 0 & 0 & 0 & \infty & 0 & 0 \\
\beta\hat{p} & 0 & 0 & 2.7 & 0 & 0 & 0 & 0 & 0 \\
\beta\hat{r} & \infty & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\alpha^2 & 100 & 0 & 100 & 0 & 100 & 0 & 0 & 0 \\
\alpha\hat{p} & 0 & 4.8 & 0 & 0 & 0 & 0 & 0 & 0 \\
\alpha\hat{q} & 0 & 0 & 100 & 0 & 0 & 0 & 0 & 0 \\
\alpha\delta_\alpha & 100 & 0 & 100 & 0 & 100 & 0 & 0 & 0 \\
\hat{p}^2 & 0 & 0 & 0 & 0 & 100 & 0 & 0 & 0 \\
\hat{p}q & 0 & 0 & 0 & 100 & 0 & 100 & 0 & 0 \\
\hat{p}r & 0 & 0 & 0 & 0 & -18.1 & 0 & 0 & 0 \\
\hat{q}r & 0 & 0 & 0 & 100 & 0 & 100 & 0 & 0 \\
\hat{q}\phi & 0 & 0 & 0 & 0 & 0 & 0 & 100 & 0 \\
\hat{r}^2 & 0 & 0 & 0 & 0 & 100 & 0 & 0 & 0 \\
\hat{r}\phi & 0 & 0 & 0 & 0 & 0 & 0 & -8.0 & 0 \\
\hat{r}\hat{\theta} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 100 \\
\hat{\phi}\hat{\theta} & 0 & 100 & 0 & 0 & 0 & 0 & 0 & 0 \\
\hat{\theta}^2 & 100 & 0 & 100 & 0 & 0 & 0 & 0 & 0 \\
\end{array} \] (4.20)

This error matrix contains elements that show a 100% error and others with infinite error ($\infty$). Elements with 100% error correspond to coefficients present in the nonlinear dynamics that SINDYc failed to identify. Elements with infinite error correspond to coefficients not present in the nonlinear dynamics which SINDYc incorrectly identified. Generally, SINDYc is able to more accurately identify coefficients of greater magnitude and lower degree as it struggled to identify small coefficients and those of 2nd degree.

While it may seem that more terms might be identified by decreasing the sparsity-promoting parameters, in practice this results in many erroneous coefficient identifications. The $[\Xi]$ matrix provided in Equation 4.16 is very near optimal sparsity under these conditions.
4.5.5 SINDYc Results – Fully Constrained Library

Finally SINDYc is applied in a way that significantly constrains the possible candidate functions available for identification. Specifically, the columns of \( \Xi \) are computed using only monomials known to be present in the nonlinear dynamics of each state equation. This represents a best-case scenario where the active monomials are known, but their corresponding coefficients are not.

These constraints imposed, following four iterations, the resulting coefficient matrix, \( \Xi_{\text{result}} \); the matrix of errors, \( [E] \); and its 2-norm, \( ||E||_2 \), are provided as follows. Rows of all zeros have been removed. Comparison of this result to the true dynamics provided in Equation 4.1 reveals SINDYc successfully identifies a significant fraction of the true dynamics including those small in magnitude and large in degree.

\[
\Xi_{\text{result, reduced}} =
\begin{bmatrix}
\beta & \dot{\beta} & \alpha & \dot{\alpha} & \rho & \dot{\rho} & \phi & \dot{\phi} & \theta & \dot{\theta} \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\alpha & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
p & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
q & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
r & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\phi & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\dot{\theta} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\delta_x & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\delta_r & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
uq & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\beta p & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\alpha^2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\alpha p & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\alpha \delta_x & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
p^2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
pq & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
pr & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
qr & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
r^2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
r \phi & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
r \theta & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\beta^3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\beta^2 p & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\alpha^3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\alpha^2 \delta_x & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
q \phi & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
q \phi \theta & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\phi^3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\end{bmatrix}
\]

(4.21)
\[ \|E\|_2 = 26.6 \]  

Since these differences are significantly larger than previous applications of SINDYc, the complete error matrix is expressed as follows in terms of percent error. Each element of matrix \[4.24\] is computed using Equation \[4.19\].
This error matrix contains elements showing 100% error, indicating that SINDYe was completely unable to identify these coefficients even though they were part of a perfectly constrained set of known dynamics. Note there are no terms of infinite error since, given the fully constrained library, it was not possible for SINDYe to erroneously identify coefficients of inactive monomials.
4.6 Summary of System ID Results

TABLE 4.1

| Case | System ID Conditions | Error $||E||_2$ | Sparsity-Promoting Parameter $\lambda$ | Iterations to Convergence |
|------|----------------------|----------------|----------------------------------------|--------------------------|
| 1    | No Noise Exact Derivatives Unconstrained Library | 2.01e-7 | $\{5e^{-4}, 0.001, 5e^{-4}, 0.1, 0.01, 0.1, 0.1, 0.1\}$ | 2 |
| 2    | No Noise Central Differences Unconstrained Library | 3.90e-6 | $\{5e^{-4}, 0.001, 5e^{-4}, 0.1, 0.01, 0.1, 0.1, 0.1\}$ | 3 |
| 3    | 20% $\sigma$ Noise Exact Derivatives Unconstrained Library | 2.01e-7 | $\{5e^{-4}, 0.001, 5e^{-4}, 0.1, 0.01, 0.1, 0.1, 0.1\}$ | 2 |
| 4    | 2% $\sigma$ Noise Regularized Differentiation Library of Degree 2 Terms | 3.41[$\dagger$] | $\{2.75, 3, 5, 52, 64, 35, 26.5, 14.5\}$ | 5 |
| 5    | 2% $\sigma$ Noise Regularized Differentiation Fully Constrained Library | 26.6 | $\{.05, 0.3, 0.2, 2.0, 0.2, 1.4, 0.8, 1.0\}$ | 4 |

4.7 Excitation of Identified Systems

The errors given in Table 4.1 do provide some measure of accuracy for each case of system identification, but they cannot capture how closely the identified models match the behavior of the true dynamics. However, this can be accomplished by exciting the identified models and comparing their response to the true system.

Each system is excited using the same control input doublets used in Section 2.6. The systems identified in cases 1, 2, and 3 respond virtually identically to the true system model and are therefore omitted from the following excitations.

As shown in Figure 4.7, an aileron doublet produces strong lateral response. The system identified in Case 5 tracks the dynamic response of the true system nearly perfectly. During control input application the Case 4 system does seem to accurately model roll rate and roll angle.

$\dagger$Error represents difference between identified and true coefficients up to and including degree 2.
However, while the system identified in Case 4 appears to share its natural frequency with the true dynamics, once free from control input, its response in sideslip, roll rate, and yaw rate is somewhat unstable, oscillating with increasing amplitude. Along this axis, the Case 4 system poorly models the dynamic response to this input. Specifically it seems to lack the proper dampening qualities.

Figure 4.7: Identified Systems, Response to Aileron Doublet

As shown in Figure 4.8 an elevator doublet produces strong longitudinal response. Again
the Case 5 system is a near perfect analog for the true system. Additionally, the Case 4 system performs quite well, modeling with reasonable accuracy the dynamic behavior both during and following control application. Both systems, Case 4 and 5, retain the pitch stability of the true system. Along this axis, both Case 4 and 5 may provide sufficiently accurate modeling, although Case 5 is a remarkably strong model.

Figure 4.8: Identified Systems, Response to Elevator Doublet
As shown in Figure 4.9, a rudder doublet produces strong lateral and directional response. As before, the Case 5 system is again a near perfect match to the true system. During control application, the Case 4 system reasonably approximate the behavior of the true system. However, it again seems to lack the dampening of the true system as it models persistent conciliations of significantly greater magnitude that also appear to be only marginally stable. For this type of input the Case 4 system poorly models the dynamic response.

Figure 4.9: Identified Systems, Response to Rudder Doublet
This analysis indicates that the fully constrained library used in Case 5 may provide a method to confidently model nonlinear dynamics when the active terms are known, but their coefficients are not. Even with a significant fraction of the terms missing from the Case 5 model, its response is very close to the true behavior given these simple inputs. Conversely, the Case 4 model cannot be used to confidently model the system.
CHAPTER 5
DISCUSSION

Under certain conditions, the SINDYc algorithm displays remarkable ability to identify the nonlinear dynamic model of the S-211 aircraft. While it is most effective when the state measurements are free of noise and the derivatives are known exactly, accuracy is still very high when either state measurements are noisy or derivatives are numerically computed, but the method fails when both are true, unless further constraints are imposed.

The goal of this research was to present a method which produces a nonlinear 6-DOF aircraft model given only time-history data of each state and input variable. To that end SINDYc provides an excellent framework when properly applied. Future adaptive aircraft control systems will require a method to manage unexpected changes to aircraft dynamics, this may come in the form of malfunction or damage. While SINDYc was created, in part, to identify nonlinear systems of unknown dynamics, its performance is significantly enhanced when some knowledge of the system is available. SINDYc and methods like it could allow for rapid identification of aircraft dynamics and subsequent adaptive flight control.

### TABLE 5.1

<table>
<thead>
<tr>
<th>Case</th>
<th>Noise</th>
<th>Derivatives</th>
<th>Library</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>No Noise</td>
<td>Exact Derivatives</td>
<td>Unconstrained Library</td>
</tr>
<tr>
<td>2</td>
<td>No Noise</td>
<td>Central Differences</td>
<td>Unconstrained Library</td>
</tr>
<tr>
<td>3</td>
<td>20% $\sigma$ Noise</td>
<td>Exact Derivatives</td>
<td>Unconstrained Library</td>
</tr>
<tr>
<td>4</td>
<td>2% $\sigma$ Noise</td>
<td>Regularized Differentiation</td>
<td>Library of Degree 2 Terms</td>
</tr>
<tr>
<td>5</td>
<td>2% $\sigma$ Noise</td>
<td>Regularized Differentiation</td>
<td>Fully Constrained Library</td>
</tr>
</tbody>
</table>

#### 5.1 Noise and State Precision

The Case 2 system identification uses central differences to compute the state derivatives at all points in the time history. Inherent in this method are errors associate with derivative calculation. Despite this, the resulting SINDYc system identification is extremely close to the true system.
However, when any amount of noise is imposed upon the states themselves, as in Section 4.5.5, the algorithm completely fails. This may indicate that SINDYc is significantly more sensitive to errors in state data than errors in state derivative data. Since the state data pervades nearly every element of the data library, $\Theta(X,U)$, especially in the form of triplet products and higher order terms, it is reasonable to conclude that errors in this data are compounded to the point of rendering SINDYc completely ineffective, when unconstrained.

5.2 Applicability of a Fully Constrained Library

Section 4.5.5 employed SINDYc using noisy state data and numerical derivatives computed using regularized differentiation. However, this produces reasonable results only when the library of candidate functions is constrained exclusively to terms known to be active. Obviously if the true candidate functions are unknown, this constraint is impossible. Clearly SINDYc would be remarkably useful if this constraint can be imposed, provided the resulting system identification is accurate.

In the case of an aircraft, these component functions are knowable. While he does not develop the full nonlinear equations of motion, Morelli does present the nonlinear parametrization of each aerodynamic coefficient in his modeling of an F-16 fighter aircraft [18]. For example, the coefficient of rolling moment due to aileron deflection, $C_{l_{\delta_a}}$, may be expressed as follows where each $k$ term is a constant coefficient.

$$C_{l_{\delta_a}} = k_0 + k_1 \alpha + k_2 \beta + k_3 \alpha^2 + k_4 \alpha \beta + k_5 \alpha^2 \beta + k_6 \alpha^3$$ (5.1)

Such expressions are provided for each aerodynamic coefficient and these could be used in a dynamic framework to develop a full nonlinear model. Using such a model and a constrained function library, SINDYc may be able to identify these equations of motion provided sufficient state excitation through appropriate control input.
5.3 Significance of Identified Model Accuracy

The system identification performed in Case 5, using a fully constrained data library, does possess significantly more error, as measured by the norm of the error matrix, $\|E\|_2$, than Cases 1, 2, and 3. However, the Case 5 result is not a poor model as demonstrated by the analysis of Section 4.7.

This illustrates a critical point. While accurate system identification is obviously desirable, the ultimate purpose of such an identification is its use in a nonlinear control framework. To this end, the identified model does not have to be perfect, it only needs to express a strong enough description of the dynamics as to model behavior with reasonably accuracy. The conclusive evaluation of the Case 5 system requires it to be implemented in a way that attempts to control the true system dynamics, as described by Equation 4.1 and comparing that performance to the performance of well-established control techniques.
CHAPTER 6
FURTHER INVESTIGATION

6.1 Potential Model Improvements

There may be several ways to improve the dynamic model derived in the this paper by re-vising certain assumptions. When transforming the force equations into the aerodynamic variables in Section 2.1.3 the trim speed is assumed to be constant. Additionally, all aerodynamic force and moment terms use a constant dynamic pressure, $\bar{q}$. Of course dynamic pressure is a function of velocity, $\bar{q} = \frac{1}{2} \rho u^2$. These two assumptions reduce the model accuracy as the difference between model airspeed and trim airspeed grows. By appropriately incorporating the state variable $u$ into these terms, the model may be applicable over a greater region of airspeeds.

The aerodynamic models of lift and side force have been constructed using cubic functions as in Equation 6.1. While this does introduce nonlinearity that represents stall-like behavior, it does not accurately reflect real stall behavior. The inclusion of a squared term would allow these models to more precisely fit the true aerodynamic behavior of an actual aircraft, resulting in Equation 6.2.

$$C_{Lw} = C_{L_{w0}} + C_{L_{w\alpha}} \alpha - C_{L_{w\alpha^2}} \alpha^2 - C_{L_{w\alpha^3}} \alpha^3$$ (6.1)

$$C_{Lw} = C_{L_{w0}} + C_{L_{w\alpha}} \alpha - C_{L_{w\alpha^2}} \alpha^2 - C_{L_{w\alpha^3}} \alpha^3$$ (6.2)

6.2 Control Input Optimization

The control input policy used to excite the dynamic model of the S-211 is developed in Section 4.1. Whereas the magnitude of the control inputs was made to be as great as possible without resulting in system instability, the frequencies and amplitude modulations present in the input policy were chosen largely by well-informed trial and error. While the input policy may result in accurate system identification, it is by no means exhaustively optimized.

The three control inputs, $\delta_a$, $\delta_e$, $\delta_r$, are driven sinusoidally with a phase shift of $\frac{1}{6}$ cycles between them, as shown by Equations 4.2. Perhaps a different choice of phase shift would result in more accurate identification under all conditions. Additionally, the control inputs contain
amplitude modulation, but the addition of frequency modulation could serve to generate a richer, more irregular state response, possibly promoting system identification. Finally, the control input frequencies were chosen to generate an aggressive state response, but perhaps more gentle inputs exercised over a longer time period may result stronger system identification. A comprehensive exploration of such variations to the control input policy may quantify the effect of each.

6.3 Nonlinear Control Framework

The next logical step of this research is to incorporate an identified system, resulting from SINDYc, into a nonlinear control framework and compare its performance to that of traditional control methods. Kaiser et al. employs a model predictive control (MPC) architecture to produce desired behavior in several nonlinear systems with models resulting from SINDYc [13].

Such a control strategy, when implemented on an aircraft model, should quantify several aspects of a SINDYc-MPC framework. This includes the amount of time and data required to train such a controller, the computational rigor and time required to execute control policies, and its robustness to noise and disturbances. Additionally its performance should be evaluated by the accuracy of predicted dynamic behavior, the effectiveness of the control policy to produce desired outcomes, and its handling of strong nonlinearities.

6.4 Implementation on a Flight Simulator

Prior to the application of the SINDYc algorithm within this paper, a nonlinear aircraft model was developed in Chapter 2. This allows for very direct, concise comparison of the SINDYc results to a model that is known perfectly. If SINDYc could eventually be used with confidence in the manner described by this paper, it may then be employable on a flight simulator provided the simulation includes the nonlinear dynamics. This could be accomplished by prescribing an appropriate, automated control policy and recording the relevant state and input data. Subsequently some tuning of the regularization parameter described in Section 4.5.2 and the sparsity-promoting parameters described in Section 3.2.2 would be required. However, if properly implemented, SINDYc may be able to generate a nonlinear model for an aircraft for which the dynamics are not explicitly known.
BIBLIOGRAPHY


APPENDICES
APPENDIX A

ERRORS ASSOCIATED WITH TRIGONOMETRIC APPROXIMATIONS

The trigonometric approximations described by Equations 2.16 provide strong accuracy within certain limits. The errors, $E$, associated with these approximations are computed as follows. The resulting percent errors are shown in Figure A.1 over the range of inputs $0 \leq \xi \leq \frac{\pi}{4}$. While the error formulas are evaluated using radians, these are converted to degrees in the plot.

$$
E_{\cos} = \frac{|(1 - \frac{\xi^2}{2}) - \cos \xi|}{\cos \xi} \quad E_{\sin} = \frac{|(\theta - \frac{\xi^3}{6}) - \sin \xi|}{\sin \xi} \quad E_{\tan} = \frac{|(\theta + \frac{\xi^3}{3}) - \tan \xi|}{\tan \xi}
$$  \hspace{1cm} (A.1)

This analysis reinforces the fidelity of these approximations. The states described by angular displacements: sideslip, $\beta$; angle of attack, $\alpha$, roll angle, $\phi$; and pitch angle, $\theta$ are excited in such a way as to remain within this range. In doing so, the error associated with these states is guaranteed to be less than 5%.

![Figure A.1: Trigonometric Approximation Errors](image-url)
APPENDIX B
DERIVATION OF KINEMATIC EQUATIONS OF MOTION

Derivation of the Roll Angle Equation, \( \dot{\phi} \)

\[
\dot{\phi} = p + (q \sin \phi + r \cos \phi) \tan \theta \approx p + \left[ q \left( \phi - \frac{\phi^3}{6} \right) + r \left( 1 - \frac{\phi^2}{2} \right) \right] \left( \theta + \frac{\theta^3}{3} \right) \tag{B.1}
\]

\[
\approx p + \left( q\phi + r - r\phi^2 \right) \left( \theta + \frac{\theta^3}{3} \right) \tag{B.2}
\]

\[
\approx p + q\phi \theta + r\theta \tag{B.3}
\]

Derivation of the Pitch Angle Equation, \( \dot{\theta} \)

\[
\dot{\theta} = q \cos \phi - r \sin \phi \approx q \left( 1 - \frac{\phi^2}{2} \right) - r \left( \phi - \frac{\phi^3}{6} \right) \tag{B.4}
\]

\[
\approx q - \frac{q\phi^2}{2} - r\phi \tag{B.5}
\]
APPENDIX C
TRANSFORMATION INTO AERODYNAMIC VARIABLES

Derivation of the Airspeed Equation, $\dot{u}$

Transform $v$ and $w$.

\[ v = u \tan \beta \quad w = u \tan \alpha \]  \hspace{1cm} (C.1)

\[ \approx u \left( \beta + \frac{\beta^3}{3} \right) \quad \approx u \left( \alpha + \frac{\alpha^3}{3} \right) \]  \hspace{1cm} (C.2)

\[ \approx u\beta \quad \approx u\alpha \]  \hspace{1cm} (C.3)

Substitute $v$ and $w$ into (2.9)

\[ \dot{u} = -qw + rv + \frac{1}{m}X \]  \hspace{1cm} (C.4)

\approx -qu\alpha + ru\beta + \frac{1}{m}X \]  \hspace{1cm} (C.5)

Derivation of the Sideslip Equation, $\dot{\beta}$

Transform $\dot{v}$ to $\dot{\beta}$

\[ \dot{v} = \frac{u\dot{\beta}}{\cos^2 \beta} = \frac{u\dot{\beta}}{(1 - \beta^2)^2} = \frac{u\dot{\beta}}{1 - \beta^2 + \frac{\beta^4}{4}} \]  \hspace{1cm} (C.7)

\[ \dot{\beta} = \frac{u\dot{\beta}}{1 - \beta^2} \]  \hspace{1cm} (C.8)
APPENDIX C (continued)

Substitute $\dot{v}$ expression into 2.10 and use $U_1$ in place of $u$ in the final formulation.

$$
\dot{v} = -ru + pw + \frac{1}{m}Y
$$  \hspace{1cm} (C.9)

$$\frac{u\dot{\beta}}{1-\beta^2} = -ru + pw + \frac{1}{m}Y \hspace{1cm} (C.10)$$

$$
\dot{\beta} = -r + p\frac{w}{u} + \frac{1}{mu}Y + \beta^2 r - \beta^2 p\frac{w}{u} - \frac{\beta^2 Y}{mu}
$$  \hspace{1cm} (C.11)

$$
\dot{\beta} = -r + p\tan \alpha + \frac{1}{mu}Y + \beta^2 r - \beta^2 p\tan \alpha - \frac{\beta^2 Y}{mu}
$$  \hspace{1cm} (C.12)

$$
\dot{\beta} = -r + p\left(\alpha + \frac{\alpha^3}{3}\right) + \frac{1}{mu}Y + \beta^2 r - \beta^2 p\left(\alpha + \frac{\alpha^3}{3}\right) - \frac{\beta^2 Y}{mu}
$$  \hspace{1cm} (C.13)

$$
\dot{\beta} = -r + p\alpha + \beta^2 r + \frac{1}{mU_1}Y - \frac{\beta^2 Y}{mU_1}
$$  \hspace{1cm} (C.14)

**Derivation of the Angle of Attack Equation, $\alpha$**

Transform $\dot{w}$ to $\alpha$

$$
w = u\tan \alpha \hspace{1cm} (C.15)$$

$$
\dot{w} = \dot{u}\tan \alpha + u\dot{\alpha}\sec^2 \alpha = \frac{u\dot{\alpha}}{\cos^2 \alpha} = \frac{u\dot{\alpha}}{\left(1 - \frac{\alpha^2}{2}\right)^2} = \frac{u\dot{\alpha}}{1 - \alpha^2 + \frac{\alpha^4}{4}} \hspace{1cm} (C.16)
$$

$$
\dot{\alpha} = \frac{u\dot{\alpha}}{1 - \alpha^2} \hspace{1cm} (C.17)
$$
Substitute $\dot{w}$ expression into (2.11) and use $U_1$ in place of $u$ in the final formulation.

\[
\dot{w} = -pv + qu + \frac{1}{m}Z \tag{C.18}
\]

\[
\frac{u\dot{\alpha}}{1 - \alpha^2} = -pv + qu + \frac{1}{m}Z \tag{C.19}
\]

\[
\dot{\alpha} = -p \frac{v}{u} + q + \frac{1}{mu}Z + \alpha^2 \frac{v}{u} - \alpha^2 q - \frac{\alpha^2}{mu}Z \tag{C.20}
\]

\[
\dot{\alpha} = -p \tan \beta + q + \frac{1}{mu}Z + \alpha^2 p \tan \beta - \alpha^2 q - \frac{\alpha^2}{mu}Z \tag{C.21}
\]

\[
\ddot{\alpha} = -p \left( \beta + \frac{\beta^3}{3} \right) + q + \frac{1}{mu}Z + \alpha^2 p \left( \beta + \frac{\beta^3}{3} \right) - \alpha^2 q - \frac{\alpha^2}{mu}Z \tag{C.22}
\]

\[
\dot{\alpha} = -p\beta + q - \alpha^2 q + \frac{1}{mU_1}Z - \frac{\alpha^2}{mU_1}Z \tag{C.23}
\]
APPENDIX D
DERIVATION OF AERODYNAMIC FORCE EQUATIONS

Derivation of Forces in the X-direction

\[ D \cos \alpha = (C_{D_0} + k(C_{Lw})^2) \bar{q}S \cos \alpha \]
\[ \approx \left[ C_{D_0} + k \left( C_{Lw_0} + C_{Lw_a} \alpha - C_{Lw_{a_3}} \alpha^3 \right)^2 \right] \bar{q}S \left( 1 - \frac{\alpha^2}{2} \right) \]
\[ \approx \left( C_{D_0} + kC_{Lw_0}^2 + 2kC_{Lw_0}C_{Lw_a} \alpha - 2kC_{Lw_0}C_{Lw_{a_3}} \alpha^3 + kC_{Lw_a}^2 \alpha^2 \right) \bar{q}S \left( 1 - \frac{\alpha^2}{2} \right) \]
\[ + \left( - \frac{C_{D_0}}{2} \alpha^2 - \frac{kC_{Lw_0}^2}{2} \alpha^2 - \frac{2kC_{Lw_0}C_{Lw_a}}{2} \alpha^3 \right) \bar{q}S \]
\[ \approx \left[ C_{D_0} + kC_{Lw_0}^2 + 2kC_{Lw_0}C_{Lw_a} \alpha + \left( kC_{Lw_a}^2 - \frac{C_{D_0}}{2} - \frac{kC_{Lw_0}^2}{2} \right) \alpha^2 \right] \bar{q}S \]
\[ - \left[ \left( 2kC_{Lw_0}C_{Lw_{a_3}} + kC_{Lw_0}C_{Lw_a} \right) \alpha^3 \right] \bar{q}S \]

\[ (D.1) \]

\[ L \sin \alpha = (C_{Lw} \bar{q}S + C_{Lt} \bar{q}S_t) \sin \alpha \]
\[ \approx \left[ \left( C_{Lw_0} + C_{Lw_a} \alpha - C_{Lw_{a_3}} \alpha^3 \right) \bar{q}S + \left( C_{Ll_0} + C_{Ll_a} \alpha_t - C_{Ll_{a_3}} \alpha_t^3 + C_{Ll_e} \delta_e \right) \bar{q}S_t \right] \left( \alpha - \frac{\alpha^3}{6} \right) \]
\[ \approx \left[ \left( C_{Lw_0} \alpha + C_{Lw_a} \alpha^2 \right) \bar{q}S + \left( C_{Ll_0} \alpha + C_{Ll_a} \alpha_t \alpha + C_{Ll_e} \delta_e \alpha \right) \bar{q}S_t \right] \]
\[ - \frac{1}{6} \left[ \left( C_{Lw_0} \alpha^3 \right) \bar{q}S + \left( C_{Ll_0} \alpha^3 \right) \bar{q}S_t \right] \]
\[ \approx \left( C_{Lw_0} \alpha + C_{Lw_a} \alpha^2 - \frac{C_{Lw_0}}{6} \alpha^3 \right) \bar{q}S + \left( C_{Ll_0} \alpha + C_{Ll_a} \alpha_t \alpha + C_{Ll_e} \delta_e \alpha - \frac{C_{Ll_0}}{6} \alpha^3 \right) \bar{q}S_t \]

\[ (D.2) \]
Derivation of Forces in the Z-direction

\[ L \cos \alpha = (C_{Lw} \tilde{q}S + C_{Lt} \tilde{q}S_t) \cos \alpha \]

\[
\approx \left[ \left( C_{Lwo} + C_{Lwa} \alpha - C_{Lwa3} \alpha^3 \right) \tilde{q}S + \left( C_{Lt0} + C_{Lta} \alpha_t - C_{Lta3} \alpha_t^3 + C_{Lte} \delta_e \right) \tilde{q}S_t \right] \left( 1 - \frac{\alpha^2}{2} \right) \\
- \left[ \left( \frac{C_{Lwo}}{2} \alpha^2 + \frac{C_{Lwa}}{2} \alpha^3 \right) \tilde{q}S + \left( \frac{C_{Lt0}}{2} \alpha^2 + \frac{C_{Lta}}{2} \alpha_t^2 + \frac{C_{Lte}}{2} \delta_e \alpha^2 \right) \tilde{q}S_t \right] \\
\approx \left[ C_{Lwo} + C_{Lwa} \alpha - \frac{C_{Lwo}}{2} \alpha^2 - \left( C_{Lwa3} + \frac{C_{Lwa}}{2} \right) \alpha^3 \right] \tilde{q}S \\
+ \left( C_{Lto} + C_{Lta} \alpha_t - C_{Lta3} \alpha_t^3 + C_{Lte} \delta_e - \frac{C_{Lto}}{2} \alpha^2 - \frac{C_{Lta}}{2} \alpha_t^2 - \frac{C_{Lte}}{2} \delta_e \alpha^2 \right) \tilde{q}S_t \\
(D.3) \]

\[ D \sin \alpha = \left( C_{D0} + k(C_{Lw})^2 \right) \tilde{q}S \sin \alpha \]

\[
\approx \left[ C_{D0} + \frac{k}{6} \left( C_{Lwo} + C_{Lwa} \alpha - C_{Lwa3} \alpha^3 \right)^2 \right] \tilde{q}S \left( \alpha - \frac{\alpha^3}{6} \right) \\
\approx \left( C_{D0} + kC_{Lwo}^2 + 2kC_{Lwa} \alpha - 2kC_{Lwo}C_{Lwa} \alpha^3 + kC_{Lwa}^2 \alpha^2 \right) \tilde{q}S \left( \alpha - \frac{\alpha^3}{6} \right) \\
(D.4) \\
\approx \left[ \left( C_{D0} + kC_{Lwo}^2 \right) \alpha + 2kC_{Lwo}C_{Lwa} \alpha^2 + \left( kC_{Lwa} - \frac{C_{D0}}{6} - \frac{kC_{Lwo}^2}{6} \right) \alpha^3 \right] \tilde{q}S \]
APPENDIX E
DERIVATION OF AERODYNAMIC MOMENT EQUATIONS

Starting with general equations of unsteady aircraft motion, solve for angular accelerations.

\[ I_x \ddot{\dot{p}} - I_{zx} \dot{\phi} + (I_z - I_y)qr - I_{zx} pq = L \]  \hspace{1cm} (E.1)

\[ I_x \ddot{\dot{p}} - I_{zx} \dot{\phi} = (I_y - I_z) qr + I_{zx} pq + L \]  \hspace{1cm} (E.2)

\[ \Rightarrow \dot{\dot{p}} = I_x^{-1} I_{zx} \dot{\phi} + I_x^{-1} (I_y - I_z) qr + I_x^{-1} I_{zx} pq + I_x^{-1} L \]  \hspace{1cm} (E.3)

\[ I_y \dot{\dot{q}} + (I_x - I_z) rp + I_{zx} (p^2 - r^2) = M \]  \hspace{1cm} (E.4)

\[ \dot{\dot{q}} = I_y^{-1} (I_z - I_x) rp + I_y^{-1} I_{zx} (r^2 - p^2) + I_y^{-1} M \]  \hspace{1cm} (E.5)

\[ \Rightarrow \dot{\dot{q}} = I_y^{-1} (I_z - I_x) rp + I_y^{-1} I_{zx} r^2 - I_y^{-1} I_{zx} p^2 + I_y^{-1} M \]  \hspace{1cm} (E.6)

\[ I_z \dot{\dot{r}} - I_{zx} \dot{\phi} + (I_y - I_x) pq + I_{zx} qr = N \]  \hspace{1cm} (E.7)

\[ I_z \dot{\dot{r}} - I_{zx} \dot{\phi} = (I_x - I_y) pq - I_{zx} qr + N \]  \hspace{1cm} (E.8)

\[ \Rightarrow \dot{\dot{r}} = I_z^{-1} I_{zx} \dot{\phi} + I_z^{-1} (I_x - I_y) pq - I_z^{-1} I_{zx} qr + I_z^{-1} N \]  \hspace{1cm} (E.9)

**Derivation of Moment about the X-axis**

Perform \( \dot{\dot{r}} \) substitution using the underlined portion of Equation E.1

\[ I_x \dot{\ddot{r}} - I_{zx} \dot{\dot{p}} \]  \hspace{1cm} (E.10)

\[ I_x \dot{\ddot{r}} - I_{zx} \dot{\dot{p}} - I_{zx} I_z^{-1} (I_x - I_y) pq + I_z^{-1} I_{zx}^2 qr - I_{zx} I_z^{-1} N \]  \hspace{1cm} (E.11)

\[ \dot{\dot{r}} (I_x - I_z^{-1} I_{zx}^2) - I_{zx} I_z^{-1} (I_x - I_y) pq + I_z^{-1} I_{zx}^2 qr - I_{zx} I_z^{-1} N \]  \hspace{1cm} (E.12)
Reintroduce and simplify.

\[ I_x \dot{p} - I_{zx} \dot{r} + (I_z - I_y)qr - I_{zx} pq = L \]  
\( \text{(E.13)} \)

\[ I_x \dot{p} - I_{zx} \dot{r} = (I_y - I_z)qr + I_{zx} pq + L \]  
\( \text{(E.14)} \)

\[ \dot{p} (I_x - I^{-1}_z I^2_{zx}) - I_{zx} I^{-1}_z (I_y - I_z) pq + I^{-1}_x I^2_z qr - I_{zx} I^{-1}_z N = (I_y - I_z)qr + I_{zx} pq + L \]  
\( \text{(E.15)} \)

\[ \dot{p} (I_x I_z - I^2_{zx}) = I_{zx} (I_x - I_y) pq - I^2_{zx} qr + I_{zx} N + I_z (I_y - I_z) qr + I_{zx} pq + I_z L \]  
\( \text{(E.16)} \)

\[ \dot{p} = \tilde{I}_{zx} (I_x - I_y) pq - \tilde{I}^2_{zx} qr + \tilde{I}_{zx} N + \tilde{I}_z (I_y - I_z) qr + \tilde{I}_{zx,zx} pq + \tilde{I}_z L \]  
\( \text{(E.17)} \)

resulting in the complete \( \dot{p} \) equation

\[ \dot{p} = \left[ \tilde{I}_{zx} (I_x - I_y) + \tilde{I}_{z,zx} \right] pq + \left[ \tilde{I}_z (I_y - I_z) - \tilde{I}^2_{zx} \right] qr + \tilde{I}_z L + \tilde{I}_{zx} N \]  
\( \text{(E.18)} \)

**Derivation of Moment about the Z-axis**

Perform \( \dot{p} \) substitution using the underlined portion of Equation [E.7].

\[ I_z \ddot{r} - I_{zx} [\dot{p}] \]  
\( \text{(E.19)} \)

\[ I_z \ddot{r} - I_x^{-1} I^2_{zx} \dot{r} - I_{zx} I^{-1}_x (I_y - I_z) qr - I_x^{-1} I^2_{zx} pq - I_{zx} I^{-1}_x L \]  
\( \text{(E.20)} \)

\[ \ddot{r} (I_z - I_x^{-1} I^2_{zx}) - I_{zx} I^{-1}_x (I_y - I_z) qr - I_x^{-1} I^2_{zx} pq - I_{zx} I^{-1}_x L \]  
\( \text{(E.21)} \)
Reintroduce and simplify.

\[ I_z \ddot{r} - I_{z\chi} \dot{p} + (I_y - I_x)pq + I_{z\chi}qr = N \]  \hspace{1cm} (E.22)

\[ I_z \ddot{r} - I_{z\chi} \dot{p} = (I_x - I_y)pq - I_{z\chi}qr + N \]  \hspace{1cm} (E.23)

\[ \dot{r} (I_z - I_x^{-1}I_{z\chi}^2) - I_{z\chi}I_x^{-1}(I_y - I_z)qr - I_x^{-1}I_{z\chi}^2pq - I_{z\chi}I_x^{-1}L = (I_x - I_y)pq - I_{z\chi}qr + N \]  \hspace{1cm} (E.24)

\[ \dot{r} (I_xI_z - I_{x\chi}^2) = I_{z\chi}(I_y - I_z)qr + I_{z\chi}^2pq + I_{z\chi}L + I_x(I_x - I_y)pq - I_xI_{z\chi}qr + I_xN \]  \hspace{1cm} (E.25)

\[ \dot{r} = \tilde{I}_{z\chi}(I_y - I_z)qr + \tilde{I}_{z\chi}^2pq + \tilde{I}_{z\chi}L + \tilde{I}_x(I_x - I_y)pq - \tilde{I}_{x\chi}qr + \tilde{I}_xN \]  \hspace{1cm} (E.26)

resulting in the complete \( \dot{r} \) equation

\[ \dot{r} = \left[ \tilde{I}_{z\chi}(I_y - I_z) - \tilde{I}_{x\chi} \right] qr + \left[ \tilde{I}_x(I_x - I_y) + \tilde{I}_{z\chi}^2 \right] pq + \tilde{I}_xN + \tilde{I}_{z\chi}L \]  \hspace{1cm} (E.27)
The Polhamus Formula:

\[
C = \frac{2\pi A}{2 + \sqrt{\left[\frac{A^2 (1-M^2)}{k^2} \left(1 + \frac{\tan^2 \Lambda_{1/2}}{1-M^2}\right)\right]} + 4}
\]

where

\[
k = 1 + \frac{AR (1.87 - 0.0002\Lambda_{LE})}{100} \quad \text{for } AR < 4 \quad \text{(F.2)}
\]

\[
k = 1 + \frac{(8.2 - 2.3\Lambda_{LE}) - AR (0.22 - 0.153\Lambda_{LE})}{100} \quad \text{for } AR \geq 4 \quad \text{(F.3)}
\]

For all surfaces, \( M = 0.6 \). Other parameters are provided as follows.

**TABLE F.1**

<table>
<thead>
<tr>
<th>Surface</th>
<th>( A )</th>
<th>( \Lambda_{LE} )</th>
<th>( \Lambda_{1/2} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wing</td>
<td>5.09</td>
<td>19.5° = 0.3403 rad</td>
<td>9.75° = 0.2251 rad</td>
</tr>
<tr>
<td>H-Tail</td>
<td>5.26</td>
<td>18.5° = 0.3229 rad</td>
<td>9.25° = 0.2313 rad</td>
</tr>
<tr>
<td>V-Tail</td>
<td>3.02</td>
<td>40.2° = 0.7016 rad</td>
<td>20.1° = 0.3508 rad</td>
</tr>
</tbody>
</table>

**Calculation of Wing Lift Coefficient, \( C_{Lw} \)**

\[
k = 1 + \frac{[8.2 - 2.3(0.3403)] - 5.09[0.22 - 0.153(0.3403)]}{100} = 1.0656 \quad \text{(F.4)}
\]

\[
C_{Lw} = \frac{2\pi (5.09)}{2 + \sqrt{\left[\frac{5.09^2 (1-0.6^2)}{1.0656^2} \left(1 + \frac{\tan^2 0.1702\pi}{1-0.6^2}\right)\right]} + 4}
\]

\[
= 4.9569 \quad \text{(F.5)}
\]
APPENDIX F (continued)

Calculation of H-Tail Lift Coefficient, $C_{L_{\alpha}}$

\[
k = 1 + \frac{[8.2 - 2.3(0.3229r)] - 5.26[0.22 - 0.153(0.3229r)]}{100} = 1.0656 \quad \text{(F.6)}
\]

\[
C_{L_{\alpha}} = \frac{2\pi (5.26)}{2 + \sqrt{\left[\frac{5.26^2(1-0.6^2)}{1.0656^2} \left(1 + \frac{\tan^2 0.2313 r}{1-0.6^2}\right)\right] + 4}} = 5.0272 \quad \text{(F.7)}
\]

Calculation of V-Tail Side Force Coefficient, $C_{Y_{\beta}}$

\[
k = 1 + \frac{3.02[1.87 - 0.0002(0.7016r)]}{100} = 1.0564 \quad \text{(F.8)}
\]

\[
C_{Y_{\beta}} = \frac{2\pi (3.02)}{2 + \sqrt{\left[\frac{3.02^2(1-0.6^2)}{1.0564^2} \left(1 + \frac{\tan^2 0.3508 r}{1-0.6^2}\right)\right] + 4}} = 3.6376 \quad \text{(F.9)}
\]
APPENDIX G

CALCULATION OF CERTAIN AERODYNAMIC COEFFICIENTS

Calculation of $C_{Lw0}$ and $C_{Lt0}$

The lift coefficients at zero angle of attack for the wing and tail, $C_{Lw0}$ and $C_{Lt0}$, are calculated using the lift coefficients and incidence angles provided in Tables 2.2 and 2.4. When the aircraft is at zero angle of attack, the wing and tail have nonzero effective angle of attack, $\alpha_{\text{eff}}$, as follows.

\[
\alpha_{w,\text{eff}} = \alpha + i_w \quad \quad \alpha_{t,\text{eff}} = -i_t - \epsilon_0 + (1 - \epsilon_\alpha)\alpha \quad \quad (G.1)
\]

Using these effective angles of attack with their associated lift coefficients results in the following.

\[
C_{Lw0} = C_{Lw_\alpha} \alpha_{w,\text{eff}} = C_{Lw_\alpha} i_w = (4.957) (0.035 \text{ rad}) = 0.173
\]

\[
C_{Lt0} = C_{Lt_\alpha} \alpha_{t,\text{eff}} = C_{Lt_\alpha} (-i_t - \epsilon_0) = (5.027) (0 + 0) = 0
\]

Calculation of $C_{Lw_\alpha^3}$, $C_{Lt_\alpha^3}$, and $C_{Y_\beta^3}$

The coefficients of the cubic terms, $\alpha^3$, $\alpha_t^3$, and $\beta^3$, are chosen to produce reasonable values of $C_{Lw_{\text{max}}}$, $C_{Lt_{\text{max}}}$, and $C_{Y_{\text{vmax}}}$. A summary of the resulting maximum angles and coefficients is provided in Table G.1. While this model is not a perfect analog, it does produce nonlinear behavior similar to that of actual lift curves and lends itself to identification using the SINDYc algorithm.
APPENDIX G (continued)

TABLE G.1

COLLECTION OF CUBIC TERM COEFFICIENTS

<table>
<thead>
<tr>
<th>C_{Lw_{\alpha^3}}</th>
<th>Value</th>
<th>Resulting</th>
<th>Resulting</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>\alpha_{\text{max}}, \alpha_t, \beta_{\text{max}}</td>
<td>\alpha_{\text{max}}, \alpha_t, \beta_{\text{max}}</td>
</tr>
<tr>
<td>\alpha_{Lw_{\alpha^3}}</td>
<td>11.00</td>
<td>22.2 deg</td>
<td>1.2808</td>
</tr>
<tr>
<td>\alpha_{Lt_{\alpha^3}}</td>
<td>13.00</td>
<td>20.6 deg</td>
<td>1.2032</td>
</tr>
<tr>
<td>\alpha_{Yv_{\beta^3}}</td>
<td>7.00</td>
<td>23.8 deg</td>
<td>1.0095</td>
</tr>
</tbody>
</table>

Incorporation of these cubic terms produces lift-curves as shown in Figure G.1. The linear portion is also shown for reference and it approximates the cubic function at low angles, but the nonlinear portion dominates at angles greater than 10\(^\circ\). Note that \(C_{Yv_{\beta^3}}\) and \(C_{Yv_{\beta^3}}\) are actually negative, but are shown positive for direct comparison. Additionally the location of coefficient maximums is also identified.

Figure G.1: Effect of Cubic Terms on Aerodynamic Coefficients
APPENDIX G (continued)

Calculation of $C_{Y_{wb}\beta}$

This coefficient $C_{Y_{wb}\beta}$ is calculated by following the process described in Roskam’s *Airplane Design* Part VI, Section 10.2.4 [19]. The following values are required.

<table>
<thead>
<tr>
<th>TABLE G.2</th>
</tr>
</thead>
<tbody>
<tr>
<td>PARAMETERS FOR DERIVATION OF $C_{Y_{wb}\beta}$</td>
</tr>
<tr>
<td>Quantity</td>
</tr>
<tr>
<td>Dihedral Angle</td>
</tr>
<tr>
<td>Wing Vertical Offset</td>
</tr>
<tr>
<td>Fuselage Radius</td>
</tr>
<tr>
<td>Fuselage CS Area Where Flow Turns Viscous</td>
</tr>
<tr>
<td>Wing Area</td>
</tr>
</tbody>
</table>

The coefficient $C_{Y_{wb}\beta}$ is the sum of $C_{Y_w\beta}$ and $C_{Y_b\beta}$, that is the contribution of the wing plus the contribution of the body.

$$C_{Y_{wb}\beta} = C_{Y_w\beta} + C_{Y_b\beta} \quad (G.3)$$

Each term is computed separately then summed, as follows.

$$C_{Y_w\beta} = -0.00573|\Gamma|$$

$$= -0.00573|−1.9|$$

$$= -0.010887$$

$$C_{Y_b\beta} = -2K_i \left(\frac{S_0}{S}\right)$$

$$= -2 \left(\frac{z_w}{r_f}\right) = 1.375 \text{ (Roskam Fig. 10.8)}$$

$$= -0.09807$$

$$C_{Y_{wb}\beta} = C_{Y_w\beta} + C_{Y_b\beta} = 0.010887 - 0.09807 \quad (G.5)$$

$$= -0.08718$$