Modeling and Control of Flapping Wing Robots

Ian P. Murphy

Thesis submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfilment of the requirements for the degree of

Masters of Science
in
Mechanical Engineering

Andrew J. Kurdila, Co-Chair
Javid Bayandor, Co-Chair
Ricardo A. Burdisso

February 8, 2013
Blacksburg, Virginia

Keywords: Flapping Flight, Robotic Modeling, Nonlinear Control, Adaptive Control

Modeling and Control of Flapping Wing Robots

Ian P. Murphy

(ABSTRACT)

The study of fixed wing aeronautical engineering has matured to the point where years of research result in small performance improvements. In the past decade, micro air vehicles, or MAVs, have gained attention of the aerospace and robotics communities. Many researchers have begun investigating aircraft schemes such as ones which use rotary or flapping wings for propulsion. While the engineering of rotary wing aircraft has seen significant advancement, the complex physics behind flapping wing aircraft remains to be fully understood. Some studies suggest flapping wing aircraft can be more efficient when the aircraft operates in low Reynolds regimes or requires hovering. Because of this inherent complexity, the derivation of flapping wing control methodologies remains an area with many open research problems. This thesis investigates flapping wing vehicles whose design is inspired by avian flight. The flapping wing system is examined in the cases where the core body is fixed or free in the ground frame. When the core body is fixed, the Denavit Hartenberg representation is used for the kinematic variables. An alternative approach is introduced for a free base body case. The equations of motion are developed using Lagranges equations and a process is developed to derive the aerodynamic contributions using a virtual work principle. The aerodynamics are modeled using a quasi-steady state formulation where the lift and drag coefficients are treated as unknowns. A collection of nonlinear controllers are studied, specifically an ideal dynamic inversion controller and two switching dynamic inversion controllers. A dynamic inversion controller is modified with an adaptive term that learns the aerodynamic effects on the equation of motion. The dissipative controller with adaptation is developed to improve performance. A Lyapunov analysis of the two adaptive controllers guarantees boundedness for all error terms. Asymptotic stability is guaranteed for the derivative error in the dynamic inversion controller and for both the position and derivative error in the dissipative controller. The controllers are simulated using two dynamic models based on flapping wing prototypes designed at Virginia Tech. The numerical experiments validate the Lyapunov analysis and illustrate that unknown parameters can be learned if persistently excited.

This work received support through contract JFC-11-139 as part of the Institute for Critical Technology and Applied Science (ICTAS).

Contents

1 Introduction and Motivation **1**

1.1 Literature Review 2

1.2 Nomenclature List 7

2 Dynamic Models of Flapping Wing Robots **8**

2.1 Review of Robot Kinematics 8

2.1.1 Rotation Matrices 8

2.1.2 Homogeneous Transforms 9

2.1.3 Denavit-Hartenberg Convention 11

2.1.4 Angular Velocity and Translational Velocity 12

2.2 Lagrange's Equations 14

2.2.1 Fixed Base Body 15

2.2.2 Moving Base Body 16

2.3 Aerodynamic Model 21

2.4 Examples of Dynamic and Kinematic Models 25

3 Control of Flapping Wing Robots **31**

3.1 Approximate Dynamic Inversion 31

3.1.1 Hard Switching Controller 32

3.1.2 Soft Switching Controller 34

3.2 Adaptive Control 34

3.2.1 Approximate Dynamic Inversion with Adaptation 34

3.2.2 Dissipative Controller with Adaptation 36
3.3 Numerical Simulation 37
3.3.1 Software Architecture 37
3.3.2 Dynamic Inversion Controller 40
3.3.3 Dissipative Controller 51

4 Conclusion **58**
4.1 Future Work 59

A Dynamic Model Examples **63**
A.1 Daedalus Mass Matrix 63
A.2 Daedalus Nonlinear Coriolis Matrix 64
A.3 Daedalus Potential Energy Gradient 66
A.4 Larus Mass Matrix 66
A.5 Larus Nonlinear Coriolis Matrix 68
A.6 Larus Potential Energy Gradient 72

List of Figures

2.1 A number of reference frames, typical of a kinematic chain, with point p and connecting vectors . . . 9

2.2 Homogeneous transformation from the 0 to 1 frame . . . 10

2.3 Reference frames in the DH convention with DH parameters labeled . . . 12

2.4 Ground (zero) frame, and i frame with point p with connecting vectors labeled . . . 13

2.5 Body with frame at the center of mass about which the inertia tensor is calculated . 16

2.6 Illustration of free base body (0,0) with three serial chains, c, $c = 1$ and $c = 2$ 17

2.7 Illustration of the body fixed frame on a wing section with α, the angle of attack, and β, the sideslip . . . 22

2.8 Typical graph of the coefficients of drag and lift versus angle of attack, α_i [1] . . . 23

2.9 Daedalus CAD model with Denavit-Hartenberg axes attached . . . 25

2.10 Daedalus CAD model with skin covering . . . 25

2.11 DH frames of the Larus prototype . . . 29

2.12 Left wing DH frames of the Icarus prototype pictured with $\theta_{(L,0)}$ through $\theta_{(L,4)}$ equal to zero . . . 30

2.13 Right wing DH frames of the Icarus prototype pictured with $\theta_{(R,1)}$ through $\theta_{(R,4)}$ equal to zero . . . 30

3.1 Coriolis4DOF code hierarchy . . . 38

3.2 Fundx Simulink diagram, with signal dimension displayed . . . 39

3.3 Fundx code hierarchy . . . 40

3.4 True c_L curve used in the numerical studies . . . 41

3.5 True c_L curve decomposed into the nine separate basis functions . . . 42

3.6 Path of the linkages through its desired joint trajectories during the downstroke (left), and upstroke (right) 43

3.7 Errors in joint positions with lift acting on body 2 44

3.8 Errors in joint velocities 45

3.9 Lift on body 2 and estimated lift on body 2 46

3.10 Parameter error on body 2 lift 47

3.11 Angle of attack of body 2 48

3.12 True c_D curve using nine unknown functions 48

3.13 Errors in joint positions with drag and lift acting on bodies $2-4$ 49

3.14 Errors in joint velocities with drag and lift acting on bodies $2-4$ 49

3.15 Parameter errors on bodies $1-4$ 50

3.16 Angle of attack for wing sections $1-4$ 50

3.17 Errors in joint positions with drag and lift acting on bodies $2-4$ 52

3.18 Errors in joint velocities with drag and lift acting on bodies $2-4$ 53

3.19 Dissipative controller effort on bodies $2-4$ 53

3.20 Parameter error on bodies $1-4$ in the dissipative controller simulation 54

3.21 Error of the third lift parameter of body 2 54

3.22 Error in the fourth lift parameter of body 2 55

3.23 Angle of attack response for all bodies in the dissipative controller simulation . . . 55

3.24 Total lift acting on the Daedalus wing model in the dissipative controller simulation 56

3.25 Errors in Larus joint positions with drag and lift acting on bodies $2-4$ 56

3.26 Errors in Larus joint velocities with drag and lift acting on bodies $2-4$ 57

3.27 Dissipative controller effort on bodies $2-4$ using the Larus dynamic model 57

List of Tables

2.1 Notation for the discussion of moving base body models 18

2.2 Daedalus DH parameters 26

2.3 Daedalus sample DH parameters 26

2.4 Daedalus sample mass parameters 26

2.5 Daedalus moment of inertia parameters, $kg \cdot m^2 \times 10^{-5}$ 27

2.6 Daedalus sample relative center of mass positions 27

2.7 DH Parameters for the Larus model 27

2.8 Larus sample DH parameters 28

2.9 Larus sample mass parameters 28

2.10 Larus moment of inertia parameters, $kg \cdot m^2 \times 10^{-5}$ 28

2.11 Larus sample relative center of mass positions 28

2.12 Icarus right wing DH parameters 29

2.13 Icarus left wing DH parameters 30

3.1 Fundx input and output for dissipative controller simulation 39

3.2 Daedalus DH parameters used in dynamic inversion simulation 40

3.3 Daedalus relative aerodynamic centers, $\boldsymbol{r}^i_{a_i}$ 41

Chapter 1

Introduction and Motivation

A wide range of problems are encountered in the analysis, design and fabrication of robots that emulate bird flight. In analysing these types of systems, the unsteady, nonlinear aerodynamics present significant difficulties. The low Reynolds numbers experienced by small scale aircraft also affect the design of wings that improve propulsion. In addition to the importance in understanding the complex unsteady aerodynamics, these systems require advanced flight control development to achieve stable flapping flight.

Construction of such vehicles also remains formidable. Vehicles based on bird flight must be structurally sound. Designs must account not only for the aerodynamic forces, but also for the dynamic loads induced in the flapping wings. The mechanical construction must rely on strong but lightweight structures and materials to give the vehicle a lift/weight ratio over one. Successful prototypes therefore require specialized fabrication and manufacturing techniques to create parts from materials such as carbon fiber. Electrical components, such as micro-controllers, batteries, signal processors, or sensors must be sufficiently light and compact to be a payload for a small scale flying vehicle. Further development of light, efficient motors and electromechanical power sources is even more crucial for success of flapping wing robots. Since the mass of the motors is often a significant proportion of weight, any increased motor mass will require stronger structures and improved lift. For this reason, light and compact power sources remain a limiting factor in the efficiency of an electromechanical system.

This thesis studies the kinematic and dynamic problems that arise in the study of articulated, flapping wing aircraft and the application of various control techniques to these problems. The articulated wing is modeled as a robotic arm, and the kinematics are analyzed using common techniques used in the study of serial chain robots. This study also develops a method for treating the kinematics and dynamics of a moving base body, or free flying system. The dynamic models are combined with a quasi-steady state aerodynamic model to generate the full system equations of motion. The equations of motion are considered in the development of two nonlinear controllers which utilize adaptation to account for uncertainties in the aerodynamic loads. The controllers are based on dynamic inversion and dissipative control techniques found in standard robotics texts, but have been modified to remain effective under the unique system equations. Lyapunov analysis is used to study the stability of these nonlinear systems and is integral for the generation of update laws. The update laws allow for adaptation in the system model and as a result, increase stability and performance.

Numerical studies are carried out for the closed loop system to verify the control designs. Denavit Hartenberg kinematic models are derived for various flapping wing vehicle designs developed by senior design teams at Virginia Polytechnic Institute and State University (Virginia Tech). The system parameters, such as masses and link lengths, are chosen to represent the physical parameters of these prototypes. The desired wing trajectories are composed of joint angle functions, which are designed for motions which the prototypes can achieve kinematically. These motions include two flapping modes observed in birds [2, 3], flapping and lead/lag. The amplitude and frequency vary in bird flight depending on the level of thrust and lift required. For purposes of verifying the control design, the amplitudes and frequencies were held constant. Values for the simulations were chosen considering several sources of data for level flying birds in the correct scale [2–4].

1.1 Literature Review

With the growing demand for unmanned air vehicles (UAVs), research that studies micro air vehicles (MAVs) has also experienced increased attention from researchers. One current trend in these research endeavours is a focus on bio-inspired design. Researchers look for inspiration by studying creatures such as birds [2, 3, 5–7], insects [8–11], bats [12], and even some less common animals such as those found in Chrysopelea, the genus of flying snakes [13–15]. Bungent and Seelecke [12] rely on principles of bioinspired design to define three major categories of MAVs. The first type includes traditional fixed wing aircraft. These are generally scaled down versions of common aircraft with propeller driven propulsion systems. The issue that drives research for this class is that characteristic length greatly decreases as these vehicles are scaled down, and this inevitably implies they operate in a low Reynolds number regime. As Bungent and Seelecke describe, this lower Reynolds number increases viscous effects and has motivated research in other MAV flight methods. Rotary wing MAVs constitute their second category of MAVs. These vehicles make up a significant share of the field with quadrotor vehicles being a prominent example. Lastly, flapping wing vehicles make up the third category. There has been a large increase in their number in the past decade. Flapping wings can execute motions that some claim [6, 10, 16] are more well suited for low Reynolds number flight. Bungent and Seelecke detail their success using shape memory alloy (SMA) wires in an experiment to actuate wings to match the range of motion seen in bats. Their prototype has two degrees of freedom and uses resistance as feedback in the control of the SMA actuators.

Flying insects are often studied as subjects of inspiration for flapping wing flight. Singh and Chopra [11] point out some benefits to flapping wing aircraft such as advanced maneuverability and low noise production. They investigate some aspects of hovering flight including thrust versus beat frequency, and unsteady aerodynamics. They discuss the fact that quasi-steady aerodynamics do not agree with phenomenon seen in insect flight, and propose that flying insects take advantage of unsteady aerodynamics in order to achieve the performance necessary for hovering flight. They develop a flapping mechanism for experimental studies. It consists of two degrees of freedom in each wing. For their scale, for vehicles that have a 148mm wing length, they find an optimal beat frequency between 10-11 Hz. There is decreased thrust when frequency was increased beyond this point. They study several wing shapes, and wings are constructed using an aluminum frame with Mylar film. They note that RC Microlite material is lighter but provides less lift than Mylar. In closing they compare the elastic and rigid wing analysis with the results obtained from the flapping apparatus.

Deng et al. have published several articles that study system modeling and control design of insect-based flapping wing vehicles. The authors classify their project as research in “micromechanical flying insects (MFIs)”. Deng et al. [9] synthesize results in their work from multiple areas of study including insect biology, including sensory feedback, wing aerodynamics, body dynamics, actuator dynamics, sensors, and flight control. The authors note one advantage flapping wing insects have over rotary aircraft is that they are less susceptible to environmental disturbances. The authors use standard quasi-steady aerodynamics to model lift and drag, and blade element theory is used for rotating wings. Blade element theory was developed by the rotary wing industry and is performed by dividing the wing length into infinitesimal divisions and integrating along the length of the wing. The authors compare this aerodynamic analysis with the experimental results found with their prototype ‘robofly’. They discuss reference frames, kinematics and dynamics of the base body, and compare the use of Euler angles and quaternions. The authors relate the use of the ‘ocelli’ in flying insects to horizon detectors in aircraft. These use light intensity measurements from the sun at multiple locations to provide support in attitude stabilization. They propose two more methods for feedback, a magnetic compass for heading information and structures analogous to the ‘halteres’ on these insects to measure body rotations.

In the second article of the series by Deng et al. [8], a high frequency periodic control problem is formulated for the underactuated system. The controller is designed for implementation with limited on-board computational resources. In their control design, the authors use an LQR method for weighing performance and control effort. The authors observe that the use of high-frequency periodic control with geometric control theory would be prohibitively computationally expensive for on-board implementation. The authors instead apply averaging theory to the problem which more closely matches standard controller design employed in helicopter flight control. Their approach does not guarantee asymptotic stability, but holds that the error is bounded and deemed negligible in size from a practical sense.

Taylor and Thomas in [17] look at various aspects of flapping flight stability. They first review two main approaches to achieve stability, inherent stability and active control. Inherent stability is achieved when disturbances are passively damped out. In the case of helicopter stability, increase in velocity creates a moment when the retreating blade has decreased lift and the advancing blade has increased lift. Taylor and Thomas describe this as mildly unstable longitudinal dynamics. Similarly, flapping wings exhibit a longitudinal instability when increased velocity flow over the wings causes bending about the chord. On the other hand, birds whose wings have sufficient stiffness may not be greatly affected by this behavior. The article then analyzes the flapping wing problem using blade element theory and quasi-steady aerodynamics. Using this analysis, a flapping flight vehicle model is derived that has first order stability in hover. During forward flight, stability is dependent on three terms: the coordinates of the aerodynamic center, forward inclination of the flight force vector, and sign of the pitching moment coefficient about the aerodynamic center.

Leonard [18] studies the dynamics and stability of hover capable MAVs modeled after flying insects. Trim solutions are examined in hover, climb, and forward flight cases. A set of quasi-coordinates are used in conjunction with Lagrange's equations to derive the state space equations of motion for a three frame wing model. Leonard then discusses an aerodynamic model that is based on quasi-steady potential flow. Reversed flow theory serves as a foundation for this model which allows for control by adjusting the point of flow reversal across the wing. The reversed flow occurs when the leading edge becomes the trailing edge and vice versa. With precise control of the wing rotation, a switch in flow direction before the end of one half stroke will result in an upward force while a late switch in flow direction results causes a downward force. Two airload models are validated and agree very closely with each other in predicting thrust and lift in the body frame. Lastly, Leonard explores open loop dynamic stability using a Stroke-Averaged system and performs the analysis using a periodic trim solution technique based on Floquent analysis. The simulations show that the system is naturally unstable and that wing hinge location and potentially trim do not seem to affect the stability. The author assumes all the bodies are rigid, and assumes a constant inflow and viscosity model.

Ansari et al. in [10] study insect flight from an aerodynamics point of view. As motivation for their work, they claim that insect based bioinspired flight leads to improved efficiency and manoeuvrability in comparison to its fixed and rotary wing counterparts. Ansari et al. [10] expand upon the basic aerodynamic models such as in Deng et al. [8] that are based on quasi-steady formulations. The authors claim their aerodynamic models are some of the 'most satisfactory to date'(2006) while they review a variety of approaches such as steady-state, quasi-steady, semi-empirical, and fully unsteady methods. The large lift force not accounted for in the typical quasi-steady state model is due to the influence of the leading edge vortices (LEV).

Another study on flapping flight aerodynamics appears in [7]. The authors von Ellenrieder et al. develop a general study of flapping wing aerodynamics as applied to both insects and birds as well as some discussion of fish and marine animals. This study contains a 2D analysis of steady cruising and propulsion in these 'natural flyers'. It also states that observed wing pitching in several flying insects is assisted by wing inertia and aerodynamic effects. In hovering flight, these contributions may be used for a completely passive control of the pitch. The authors consider whether flying is a limit cycle process, where there is a coupling between internal power and passive energy dissipation. They suggest that a limit cycle may occur in flapping flight but conclude that further testing is necessary before it is confirmed.

Gopalakrishnan et al. [19] study the formation and effects of the leading edge vortex on low Reynolds number flight (for Re$= 10,000$). They use an single rigid insect type wing for their simulations. They use a delayed-stall, unsteady aerodynamic formulation with varying angle of attack and show that a moderate angle of attack will result in high propulsive efficiency.

Mazaheri and Ebrahimi in [20] perform an aerodynamic study based on an experiment featuring flexible membrane flapping. They follow up on prior research [21] showing that an increase in thrust and lift results when the torsional stiffness in a flapping wing is decreased. The authors design and test a flapping wing prototype constructed with a flexible membrane wing. They investigate the relationship of thrust and lift as a function of flapping frequency, angle of incidence, and flow velocity. The mechanism is designed to oscillate between +30° and −19°. For higher flapping rates, from $3Hz$ to $10Hz$ in their case, the thrust always increases. For lower flapping rates, lift force is nearly independent of the frequency. For lower angles of incidence, a large increase in lift is observed for frequencies over 5 Hz. At high velocities, flapping rate has a small effect on lift. Angle of incidence is proportional to lift. They also noted that angle of incidence and thrust are inversely proportional.

Aditya and Malolan in [5] investigate the effect of the Strouhal number on the propulsion of a flapping wing. In this article, the Strouhal number is calculated as:

$$S_t = \frac{fA}{U_\infty} = \frac{2 \cdot f \cdot b_{semi-eff} \cdot \sin\left(\frac{\phi}{2}\right)}{U_\infty}$$

where A is amplitude, the semi-span of the wing is $b_{semi-eff}$, the flapping angle is ϕ, the flow velocity or forward speed is U_∞, and flapping frequency is f. The authors recommend a range of values for the Strouhal number in wing design. They observe that a range of 0.2 to 0.4 is typical for birds. Fast flying birds exhibit Strouhal numbers near $S_t = 0.2$ and slower flying birds have values around $S_t = 0.4$. They test a 15cm span flapping mechanism and find their highest peak propulsive forces result from a Strouhal number between $0.1 - 0.2$.

Taylor et al. [22] agrees that Strouhal number is also crucial for efficiency in flapping flight. The same range of Strouhal number $S_t = 0.2$ to 0.4 is observed for both flying animals and marine animals such as fish, dolphins and sharks. The investigators perform a statistical analysis and find intermittent flying birds to have a mean Strouhal number $S_t = 0.34$ and for direct flyers to have a Strouhal number $S_t = 0.20$. In summary Taylor et al. in [22] create a rule of thumb that the velocity will be three times the product of the flapping frequency and amplitude.

Han [3] investigates unsteady aerodynamics in stable flight of a seagull. He notes that combining three "flapping, folding, and lead/lag" modes generates an optimal lift scenario. He uses a boundary element method to perform the unsteady aerodynamic analysis. The kinematic model used in this study selects the three DOF system introduced by Liu et al. [2]. This choice includes a flapping angle ψ_1, a folding angle ψ_2 and lead/lag angle ϕ_2. Using only one of these flapping modes decreases drag for flapping, folding, and lead/lag. Using a flapping mode alone increases lift coefficient the rest of the wing fixed. Of the combined tests, flapping and folding generates the greatest thrust generation of all the tests, and flapping with lead/lag produced the second best lift coefficient. The overall highest lift coefficient is achieved via the combination of the three, although thrust coefficient is slightly decreased from the flapping and folding combination (0.2 to 0.1704).

Other than the data contained in engineering articles, there are few biological studies on birds which include quantitative research. Tobalske at the University of Montana Flight Laboratory performs many quantitative empirical studies on skeletal mechanics, aerodynamics and mechanical power of birds. Many of the experiments provide useful tests on live birds, which document precise kinematic and aerodynamic data of bird flight. In a study of tip-reversal by Crandell and Tobalske [23], an experiment is performed using four synchronized high-speed cameras to obtain in vivo angle of attack and coefficient of lift for an rock dove during upstroke and downstroke. The downstroke produced an estimate of 115% of the bodyweight, at an effective wing length of 0.314m.

Liu [2] also performs quantitative in vivo tests on birds and geometric study of wings from bird specimens. Liu uses footage of level-flying, or steady flying seagulls, cranes and geese. He develops a wing model with three degrees of freedom that is able to represent the motion of the three birds. The seagull wing used in this investigation can be approximated from illustrations and is near to 0.6m. The investigation also documents wing geometry of several birds including the Seagull, Merganser, Teal and Owl. Specifically, the author studies airfoil parameters such as thickness, camber, and chord length.

Some in vivo flight testing appears as early as 1982 in the Ph.D thesis by Scoley [4]. Although the video photography and processing is primitive compared to today's standard, some morphological data is still relevant and can be used as a reference for relative scaling. Articles such as these contain extensive collections of data for different species. Two species of the genus Larus are examined, the Black Headed Gull (Larus Ridibundus), and Common Gull (Larus Canus). The Common Gull has average mass of 0.388kg, average wingspan of 1.13m and an average stroke frequency of 3.76 Hz. The smaller Black Headed Gull has an average mass of 0.251kg, average wingspan of 0.94m and average stroke frequency of 3.84 Hz. The flighted bird with the largest wingspan in this study is the Wandering Albatross, Diomedea Exulans. It has a mass of 8.5kg, wingspan of 3.42m, and flapping frequency of 2.61 Hz.

In summary, although there have been numerous studies that model flapping flight, only a few effort that study the control of flapping wing have appeared in the literature. This thesis constructs dynamic models of articulated flapping wing robots and synthesizes controllers that guarantee tracking of flapping motions. Chapter 2 begins with a derivation of the equations of motion for some classes of flapping wing robotic systems. Chapter 3 discusses the adaptive controller derived in this thesis. The thesis concludes in Chapter 4, where a summary of accomplishments is given and future work is outlined.

1.2 Nomenclature List

α_i	Angle of attack for i^{th} wing section
β_i	Side slip for i^{th} wing section
γ	Learning gain
ρ	Density of surrounding fluid
$\boldsymbol{\sigma}$	Error tracking term for dissipative controller
$\boldsymbol{\tau}$	Control input vector
Φ	Vector of basis functions
$\boldsymbol{\omega}$	Angular velocity
$\boldsymbol{\Omega}$	Vector of unknown parameters
$\hat{\boldsymbol{\Omega}}$	Vector of unknown parameter estimates
$A_{w,i}$	2D wing area of body i
$\hat{b}_1$, $\hat{b}_2$, $\hat{b}_3$	Body fixed frame unit vectors
$\mathbf{B}$	Control influence matrix
c_D	Drag coefficient
c_L	Lift coefficient
$\mathbf{C}$	Nonlinear Coriolis matrix
D_i	Drag force on i^{th} wing section
$\boldsymbol{G}$	Control Gain Matrix
$\mathbf{H}_i^j$	DH transformation matrix from frame i to frame j
$\boldsymbol{I}$	Inertia tensor
$\mathbf{J}^v$	Velocity Jacobian matrix
$\mathbf{J}^\omega$	Angular velocity Jacobian matrix
L_i	Lift force on i^{th} wing section
$\mathbf{M}$	Generalized inertial matrix
$\mathbf{n}$	Vector of nonlinear Coriolis and centripetal acceleration terms
$\mathbf{n}_a$	Vector of unknown aerodynamic loads
$\mathbf{q}$	Vector of generalized coordinates
$\mathbf{Q}_i$	Generalized force vector
$\delta\boldsymbol{r}$	Virtual displacement vector
$\hat{s}_1$, $\hat{s}_2$, $\hat{s}_3$	Stability axis unit vectors
u, v, w	Velocity components for i^{th} wing sec.
$\mathbf{v}$	The velocity of aerodynamic center w.r.t. fluid
$\mathcal{V}$	Lyapunov candidate
δW	Virtual work
$\mathbf{X}$	System states

Chapter 2

Dynamic Models of Flapping Wing Robots

2.1 Review of Robot Kinematics

Kinematics of a robotic systems is complex since a typical system is composed of numerous rigid bodies. Descriptions of such systems often introduce several coordinate frames, or frames of reference, to define kinematic variables. In this thesis, a frame is defined by a right hand, orthonormal set of unit vectors. Figure (2.1) illustrates a typical collection of frames. We will write $\left\{\mathbf{x}_0 \quad \mathbf{y}_0 \quad \mathbf{z}_0\right\}$ as the basis for the 0 frame, and $\left\{\mathbf{x}_1 \quad \mathbf{y}_1 \quad \mathbf{z}_1\right\}$ as the basis for the 1 frame and so forth. An arbitrary vector $\boldsymbol{a}$ can be expressed in terms of its components or coordinates relative to any other bases defined in the problem. In this thesis we choose to represent the coordinates of the vector $\boldsymbol{a}$ relative to frame i as $\boldsymbol{a}^i$.

$$\boldsymbol{a}^i = \begin{Bmatrix} a_1^i \\ a_2^i \\ a_3^i \end{Bmatrix} \qquad \boldsymbol{a} = a_1^i \mathbf{x}_i + a_2^i \mathbf{y}_i + a_3^i \mathbf{z}_i$$

We define $\boldsymbol{r}_p^i$ to be the coordinates relative to the frame i of the position vector that connects the origin of frame i to point p. Likewise, the vector connecting two frames will be denoted as $\boldsymbol{r}_{i,j}$ for the vector from the origin of frame i to the origin of frame j. Any frame position relative to the ground frame will appear as $\boldsymbol{r}_i := \boldsymbol{r}_{0,i}$.

2.1.1 Rotation Matrices

In the study of robot kinematics, orientation between two different, right handed, orthogonal frames is represented by rotation matrices. The 3×3 rotation matrix that relates the frames 0 and 1 is

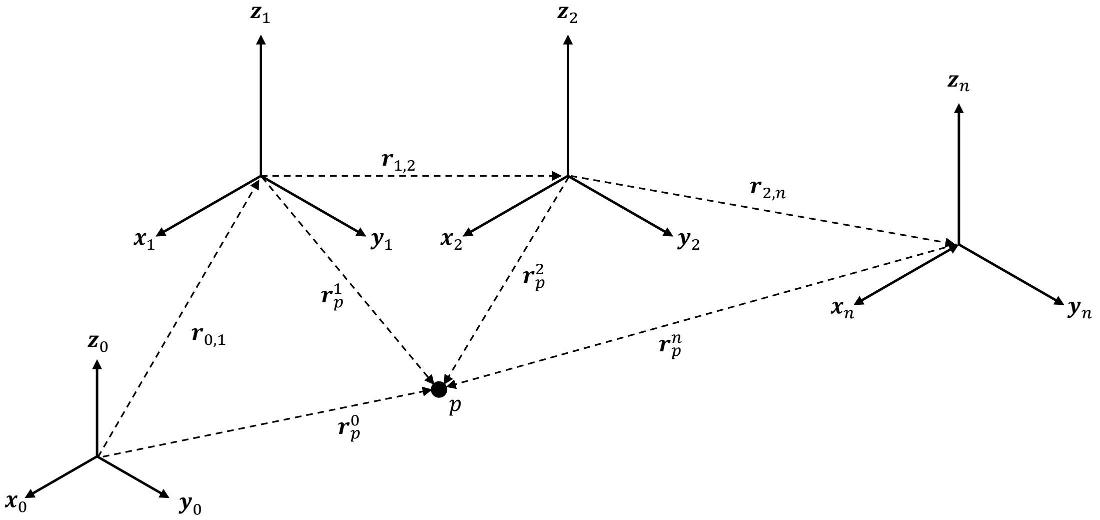

Figure 2.1: A number of reference frames, typical of a kinematic chain, with point p and connecting vectors

written as $\mathbf{R}_0^1$ and is defined as:

$$\mathbf{R}_0^1 = \begin{bmatrix} \mathbf{x}_0 \cdot \mathbf{x}_1 & \mathbf{y}_0 \cdot \mathbf{x}_1 & \mathbf{z}_0 \cdot \mathbf{x}_1 \\ \mathbf{x}_0 \cdot \mathbf{y}_1 & \mathbf{y}_0 \cdot \mathbf{y}_1 & \mathbf{z}_0 \cdot \mathbf{y}_1 \\ \mathbf{x}_0 \cdot \mathbf{z}_1 & \mathbf{y}_0 \cdot \mathbf{z}_1 & \mathbf{z}_0 \cdot \mathbf{z}_1 \end{bmatrix}$$

where $\mathbf{x}_0, \mathbf{y}_0, \mathbf{z}_0$ and $\mathbf{x}_1, \mathbf{y}_1, \mathbf{z}_1$ denote the right handed orthogonal bases of the 0 and 1 frame, respectively. The defining property of a rotation matrix is that its inverse is equal to its transpose, and therefore we can write

$$\left(\mathbf{R}_0^1\right)^{-1} = \left(\mathbf{R}_0^1\right)^{\mathrm{T}}$$

From this definition, it is clear that

$$\left(\mathbf{R}_0^1\right)^{\mathrm{T}} = \mathbf{R}_1^0$$

The rotation matrix $\mathbf{R}_0^1$ maps the components of a vector relative to the 0 frame into components relative to the 1 frame.

$$\boldsymbol{r}^1 = \mathbf{R}_0^1 \boldsymbol{r}^0$$

2.1.2 Homogeneous Transforms

While relative rotation is represented in terms of rotation matrices, rigid body motion is specified in terms of homogeneous transformations. A homogeneous transformation describes both rotation and displacement between reference frames. This topic is covered in many standard sources on robot kinematics [24–26]. The relative translation $\boldsymbol{r}_{0,1}^{1}$ contains the coordinates relative to the 1 frame the vector that defines the position of the 1 frame relative to the 0 frame. The homogeneous transform that describes the rigid body motion between the 0 and 1 frames is written as $\mathbf{H}_0^1$, defined below.

$$\mathbf{H}_0^1 = \begin{bmatrix} \mathbf{R}_0^1 & \boldsymbol{r}_{1,0}^1 \\ \mathbf{0} & 1 \end{bmatrix}$$

The homogeneous transform can then be used to relate the homogeneous coordinates of any point in reference frame 0 to reference frame 1. The homogeneous coordinates in frame 0 are denoted as $\mathbb{r}^0$ and the homogeneous coordinates for frame 1 are $\mathbb{r}^1$.

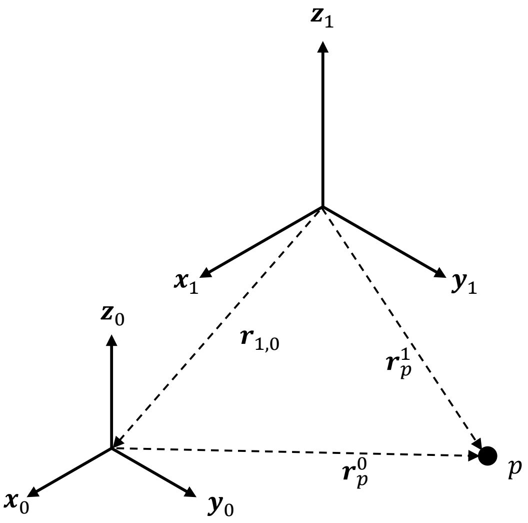

Figure 2.2: Homogeneous transformation from the 0 to 1 frame

$$\begin{bmatrix} \boldsymbol{r}_p^1 \\ 1 \end{bmatrix} = \mathbf{H}_0^1 \begin{bmatrix} \boldsymbol{r}_p^0 \\ 1 \end{bmatrix}$$

$$\mathbb{r}_p^1 = \mathbf{H}_0^1 \mathbb{r}_p^0$$

Both the homogeneous transform and rotation matrix can be combined to describe the kinematics of a chain. For example, the homogeneous transform between the third and the ground reference frame, in terms of the homogeneous coordinates, is obtained as

$$\mathbb{r}^3 = \mathbf{H}_2^3 \mathbf{H}_1^2 \mathbf{H}_0^1 \mathbb{r}^0 = \mathbf{H}_0^3 \mathbb{r}^0$$

2.1.3 Denavit-Hartenberg Convention

One of the most popular techniques used to represent robot kinematics is the Denavit-Hartenberg (DH) Convention. Typically, six parameters, three translations and three rotations, are necessary to define the relative location and pose of two reference frames. By imposing some constraints on how the adjacent frames are defined, the DH convention introduces a set of four parameters to define relative pose and orientation, two angles and two scalar translation values. The main assumption in the DH Convention is that, for adjacent frames i and $i-1$ in a serial chain, the $\mathbf{x}_i$ axis is perpendicular to and intersects the $\mathbf{z}_{i-1}$ axis. The DH parameters then orient the $i-1$ frame relative to the i frame in terms of the following four parameters:

1. The displacement d_i translates the i frame in the direction of $\mathbf{z}_{i-1}$.
2. The offset a_i translates the i frame along the $\mathbf{x}_{i-1}$ axis.
3. The rotation θ_i rotates the i frame about $\mathbf{z}_{i-1}$.
4. The twist α_i lastly rotates the i frame about its $\mathbf{x}_i$ axis.

The parameters introduced above are discussed further in [25], and are illustrated in Figure (2.3). Using this standard notation, a rotation matrix and homogeneous transform that define the rigid body motion can be derived for any two successive frames in a kinematic chain. The rotation matrix contained in the DH transformation can be expressed in terms of the two rotation matrices that are defined via an intermediate frame $\mathbb{B}$ [25]:

$$\mathbf{R}^i_{i-1} = \mathbf{R}^i_{\mathbb{B}}\mathbf{R}^{\mathbb{B}}_{i-1}$$

$$\mathbf{R}^i_{i-1} = \begin{bmatrix} \cos\theta_i & \sin\theta_i & 0 \\ -\sin\theta_i & \cos\theta_i & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha_i & \sin\alpha_i \\ 0 & -\sin\alpha_i & \cos\alpha_i \end{bmatrix}$$

$$\mathbf{R}^i_{i-1} = \begin{bmatrix} \cos\theta_i & \sin\theta_i & 0 \\ -\cos\alpha_i\sin\theta_i & \cos\alpha_i\cos\theta_i & \sin\alpha_i \\ \sin\alpha_i\sin\theta_i & -\sin\alpha_i\cos\theta_i & \cos\alpha_i \end{bmatrix}$$

The homogeneous transform can be constructed from the DH parameters using the rotation matrices above and the two translational parameters a_i and d_i. The translation $\boldsymbol{r}^i_{i-1}$ can be constructed by inspection from Figure (2.3) and is a function of θ_i, d_i, and a_i.

$$\mathbf{H}^i_{i-1} = \begin{bmatrix} \mathbf{R}^i_{i-1} & \boldsymbol{r}^i_{i-1,i} \\ \mathbf{0} & 1 \end{bmatrix} \quad \boldsymbol{r}^i_{i-1,i} = \begin{bmatrix} a_i\cos\theta_i \\ a_i\sin\theta_i \\ d_i \end{bmatrix}$$

$$\mathbf{H}^i_{i-1} = \begin{bmatrix} \cos\theta_i & \sin\theta_i & 0 & a_i\cos\theta_i \\ -\cos\alpha_i\sin\theta_i & \cos\alpha_i\cos\theta_i & \sin\alpha_i & a_i\sin\theta_i \\ \sin\alpha_i\sin\theta_i & -\sin\alpha_i\cos\theta_i & \cos\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

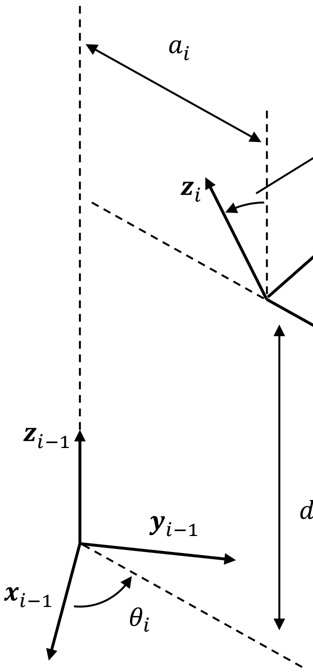

Figure 2.3: Reference frames in the DH convention with DH parameters labeled

2.1.4 Angular Velocity and Translational Velocity

Many robotic systems are constructed from chains of rigid bodies. Standard methods have been developed to derive the velocities or angular velocities in a kinematic chain. We consider a case where the point p is fixed in the i^{th} frame and is denoted by $\boldsymbol{r}^i_p$. The mapping between the frame i frame and frame $i-1$ is determined by the homogeneous transformation matrix $\mathbf{H}^{i-1}_i$. The position vector $\boldsymbol{r}^{i-1}_p$ can be decomposed as

$$\boldsymbol{r}^{i-1}_p = \mathbf{R}^{i-1}_i \boldsymbol{r}^i_p + \boldsymbol{r}^{i-1}_{i-1,i}$$

This formula is differentiated with respect to time to find the velocity.

$$\mathbf{v}^i_p = \frac{d}{dt}\boldsymbol{r}^i_p$$

The frame number is omitted in both the velocity and position for the zero frame:

$$\mathbf{v}_p := \mathbf{v}^0_p \quad \boldsymbol{r}_p := \boldsymbol{r}^0_p$$

For a DH kinematic chain, the velocity of a point p in the ground frame is found using the velocity of frame i in the ground frame, and a rotation rate about its $\mathbf{z}_i$ axis, $\dot{\theta}_i$. The frames and point p are illustrated Figure (2.4)

$$\mathbf{v}_p = \dot{\theta}_i \mathbf{z}_i \times \boldsymbol{r}^i_p + \mathbf{v}_i \tag{2.1}$$

If the frame velocity, $\mathbf{v}_i$, is calculated with Equation (2.1), recursion can be used to find velocities for an entire series of frames. This idea is fundamental in joint-space kinematics for robotic linkages that form kinematic chains.

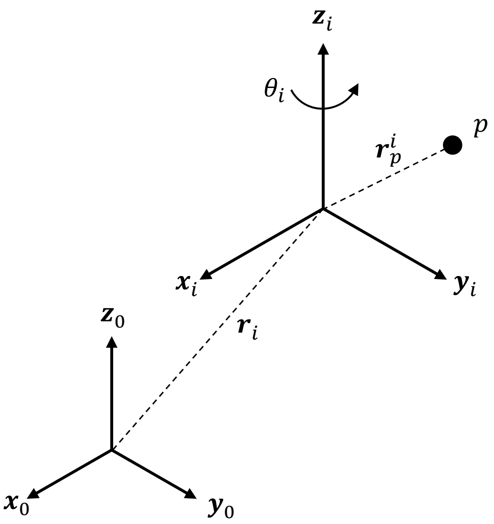

Figure 2.4: Ground (zero) frame, and *i* frame with point *p* with connecting vectors labeled

The velocity Jacobian $\mathbf{J}^{v_i}$ is defined to yield the components of velocity relative to the i^{th} frame. We suppress the frame number when the Jacobian yields components relative to the basis for the frame 0 so that $\mathbf{J}^v := \mathbf{J}^{v_0}$. The components of the velocity of point p relative to the ground frame is then

$$\mathbf{v}_p = \mathbf{J}^v \dot{\mathbf{q}} \tag{2.2}$$

where the vector $\dot{\mathbf{q}} = \dot{\mathbf{q}}_1 \cdots \dot{\mathbf{q}}_N{}^{\mathrm{T}}$ is an N vector of derivatives of the joint variables. The angular velocity Jacobian $\mathbf{J}^\omega$ is similarly defined as

$$\omega_p = \mathbf{J}^\omega \dot{\mathbf{q}} \tag{2.3}$$

By using Theorem 2.15 in Kurdila, et al. [25], finding the angular velocity Jacobian is a trivial process. When joint variables are selected to be either relative rotations or displacements, angular velocity vectors can be summed by adding the angular velocities of adjacent frames. Specifically, the angular velocity Jacobian can be generated using

$$\mathbf{J}^\omega = \begin{bmatrix} \rho_1 \mathbf{z}_0^0 & \rho_2 \mathbf{R}_1^0 \mathbf{z}_1^1 & \cdots & \rho_N \mathbf{R}_{N-1}^0 \mathbf{z}_{N-1}^{N-1} \end{bmatrix}$$

In this equation, ρ_i is 1 for a revolute joint and 0 for a prismatic joint [24–26]. Similarly, assuming that each joint is revolute, the velocity Jacobian is found to be

$$\mathbf{J}^v = \begin{bmatrix} \mathbf{z}_0^0 \times (\boldsymbol{r}_1^0 - \boldsymbol{r}_0^0) & \mathbf{R}_1^0 \mathbf{z}_1^1 \times (\boldsymbol{r}_2^0 - \boldsymbol{r}_1^0) & \cdots & \mathbf{R}_{N-1}^0 \mathbf{z}_{N-1}^{N-1} \times (\boldsymbol{r}_N^0 - \boldsymbol{r}_N^0) \end{bmatrix}$$

The position vectors $\boldsymbol{r}_1 \cdots \boldsymbol{r}_N$ represent the position vectors from the zero frame to the i^{th} frame. For prismatic joints, we have instead

$$\mathbf{J}^v = \begin{bmatrix} \mathbf{z}_0^0 & \mathbf{z}_1^0 & \cdots & \mathbf{z}_n^0 \end{bmatrix}$$

2.2 Lagrange's Equations

Once a kinematic description of the robotic system has been constructed, it remains to derive the equations governing the motion of the robot. Numerous approaches to this problem have been developed in the literature. These methods can be based on techniques of analytical mechanics, such as Lagrange's equations, on Newton-Euler formulations, or other principles such as Kane's equations. All techniques have relative advantages and disadvantages. As an extension of Hamilton's Principle, Lagrange's equations of motion serve as a convenient and popular way of generating the equation of motion for mechanical systems. In general, these equations can be written

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{\mathbf{q}}}\right) - \frac{\partial T}{\partial \mathbf{q}} + \frac{\partial V}{\partial \mathbf{q}} = \mathbf{Q}$$

In the above expression, the potential energy of the system is V, T is the kinetic energy, $\mathbf{q}$ is the vector of generalized coordinates, and $\mathbf{Q}$ is the vector of generalized forces. The generalized forces arise from the formulation of the virtual work done by external nonconservative forces acting on the system. If external force $\boldsymbol{F}_p$ acts at the physical point p, the virtual work is defined as the sum [25]

$$\delta W_{nc} = \sum_p \boldsymbol{F}_p \cdot \delta \boldsymbol{r}_p$$

Virtual work of the nonconservative external forces is then calculated in terms of the generalized forces.

$$\delta W_{nc} = \sum_{i=1} Q_i \cdot \delta q_i = \mathbf{Q}^{\mathrm{T}} \delta \mathbf{q} = \delta \mathbf{q}^{\mathrm{T}} \mathbf{Q}$$

It is often assumed that kinetic energy is a quadratic function of the generalized coordinates. The kinetic energy then has the form

$$T = \frac{1}{2} \dot{\mathbf{q}}^{\mathrm{T}} \mathbf{M}(\mathbf{q}) \dot{\mathbf{q}} \tag{2.4}$$

Such systems are known as T_2 or natural systems [27]. It is also commonly assumed that potential energy is a function of only the generalized coordinates, that is,

$$V = V(\mathbf{q})$$

If both of these conditions hold true, the equations of motion for the system can be written in a general form

$$\mathbf{M}(\mathbf{q}(t))\ddot{\mathbf{q}}(t) = \mathbf{n}(\mathbf{q}(t), \dot{\mathbf{q}}(t), t) + \mathbf{B}(\mathbf{q}(t))\boldsymbol{\tau}(t) \tag{2.5}$$

The vector $\mathbf{n}$ is a collection of nonlinear terms due to centripetal or Coriolis effects and is a function of the generalized coordinates and their derivatives. The vector $\boldsymbol{\tau}$ is the vector of control inputs. In this thesis all of the joints are assumed to be driven by a rotational actuator. Because all of the joints contain a rotary actuator, the control influence matrix is the $\mathbf{B}$ identity matrix [25]. When the actuation scheme changes, $\mathbf{B}$ is determined via calculation of virtual work. This will be discussed in more detail in Section (2.3).

2.2.1 Fixed Base Body

Two classes of robotic systems are studied in this thesis, those that have a base body fixed in the inertial frame, and those that do not. We first discuss the case where the base body is fixed in the inertial frame since the presentation is simpler. This discussion motivates the approach for the case where the base body is in motion. Robotic systems that have a base body fixed in the inertial frame are appropriate for robots fixed in laboratory or wind tunnel experiments. We assume that the robot is fully actuated. The kinematics for this case can be cast via standard approaches for robotics. The set of equations of motion for the flapping wing robot will have the form of Equation (2.5). However, it is important to note that in the study of flapping wing robots not all the terms that appear in the robot equations of motion are measurable or known. The aerodynamic contributions to the virtual work are not usually available. The equations of motion include a second nonlinear term $\mathbf{n}_a$ that represent the aerodynamic effects on the system.

$$\mathbf{M}(\mathbf{q}(t))\ddot{\mathbf{q}}(t) = \mathbf{n}(\mathbf{q}(t), \dot{\mathbf{q}}(t), t) + \mathbf{n}_a(\mathbf{q}(t), \dot{\mathbf{q}}(t), t) + \mathbf{B}(\mathbf{q}(t))\boldsymbol{\tau}(t) \tag{2.6}$$

The functional form of the aerodynamic contributions is studied in Section (2.3). As in many robotics formulations, the mass matrix is generated in our model through the use of Jacobian matrices. This method provides explicit expansions of the translational and rotational kinetic energy. First we recall the standard Lagrangian formulation for kinetic energy of a single body i using the velocity at the center of mass $\boldsymbol{v}_i$, angular velocity $\boldsymbol{\omega}_i$, and inertia tensor about the center of mass $\boldsymbol{I}_i$. The frame used for calculating the inertia tensor is depicted in Figure (2.5).

$$T_i = \frac{1}{2} m_i \mathbf{v}_i^{\mathrm{T}} \mathbf{v}_i + \frac{1}{2} \boldsymbol{\omega}_i^{\mathrm{T}} \boldsymbol{I}_i \boldsymbol{\omega}_i \tag{2.7}$$

From Section (2.1.4), we recall the Jacobian definitions in Equation (2.2) and Equation (2.3).

$$\mathbf{v}_p = \mathbf{J}^v \dot{\mathbf{q}} \quad \omega_p = \mathbf{J}^\omega \dot{\mathbf{q}}$$

The velocity is calculated at the center of mass for each link, denoted as point $g(i)$ for the center of mass for link i. The velocity Jacobian used in calculation of $\mathbf{v}_i$ is introduced as $\mathbf{J}^v_{g(i)}$. The angular velocity Jacobian introduced for calculating $\boldsymbol{\omega}_i$, the angular velocity of link i, is designated as $\mathbf{J}^\omega_i$. The velocity equations are substituted into Equation (2.7), then the generalized coordinate vectors are factored.

$$T_i = \frac{1}{2} m_i \left(\mathbf{J}^v_{g(i)} \dot{\mathbf{q}}\right)^{\mathrm{T}} \left(\mathbf{J}^v_{g(i)} \dot{\mathbf{q}}\right) + \frac{1}{2} \left(\mathbf{J}^\omega_i \dot{\mathbf{q}}\right)^{\mathrm{T}} \boldsymbol{I}_i \left(\mathbf{J}^\omega_i \dot{\mathbf{q}}\right)$$

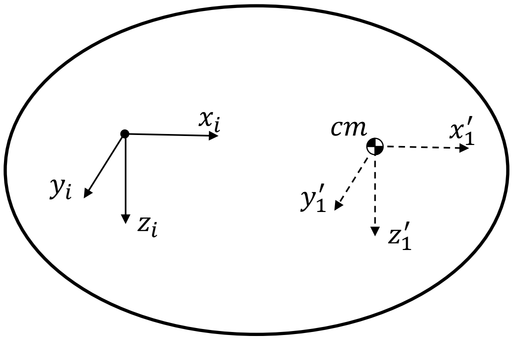

Figure 2.5: Body with frame at the center of mass about which the inertia tensor is calculated

$$T_i = \frac{1}{2}\dot{\mathbf{q}}^{\mathrm{T}}\left(m_i\left(\mathbf{J}^{v}_{g(i)}\right)^{\mathrm{T}}\mathbf{J}^{v}_{g(i)} + \left(\mathbf{J}^{\omega}_{i}\right)^{\mathrm{T}}\boldsymbol{I}_i\mathbf{J}^{\omega}_{i}\right)\dot{\mathbf{q}}$$

This formula is then simplified by introducing two matrices, the translational generalized inertia component $\mathbf{M}^{\mathbf{v}}_{i}$, and rotational generalized inertia component $\mathbf{M}^{\omega}_{i}$. The contribution due to translational motion to the mass matrix is calculated via the identity

$$\mathbf{M}^{\mathbf{v}}_{i} = m_i\left(\mathbf{J}^{v}_{g(i)}\right)^{\mathrm{T}}\mathbf{J}^{v}_{g(i)},$$

while the contribution due to rotational motion is

$$\mathbf{M}^{\omega}_{i} = \left(\mathbf{J}^{\omega}_{i}\right)^{\mathrm{T}}\boldsymbol{I}_i\mathbf{J}^{\omega}_{i}$$

In this equation $\boldsymbol{I}_i$ is the inertia matrix relative to the same basis used in the definition of the Jacobian matrix $\mathbf{J}^{\omega}_{i}$. After $\mathbf{M}^{\mathbf{v}}_{i}$ and $\mathbf{M}^{\omega}_{i}$ are generated, the mass matrix for body i is compiled by summing these two matrices [25].

$$\mathbf{M}_i = \mathbf{M}^{\mathbf{v}}_{i} + \mathbf{M}^{\omega}_{i}$$

The total kinetic energy for a single body i can then be assembled in the form similar to Equation (2.4)

$$T_i = \frac{1}{2}\dot{\mathbf{q}}^{\mathrm{T}}\mathbf{M}_i(\mathbf{q})\dot{\mathbf{q}}$$

The total energy for a fixed base robotic system is found by summing the kinetic and potential energies for all bodies.

2.2.2 Moving Base Body

To construct a kinematic and dynamic model for an ornithopter in flight, the fixed base constraint is not applicable. In this thesis we assume that there is one free base body having no directly actuated degrees of freedom, with one or more attached serial chains. We assume there are no closed loops in the system. The kinematic chains will be modeled using the DH convention. This model is illustrated in Figure (2.6). This figure shows a base body with two kinematic chains and illustrates the numbering convention used. The base body is labeled with a body fixed reference frame $(0,0)$, and ground frame is labeled as G. Links in the chains are labeled as (c,k), where c is the chain number and k is the element in chain c. A point on body k in chain 1 is labeled as $p_{(1,k)}$. A standard notation is developed for the discussion of the kinematics and dynamics of a moving base body system and summarized in Table (2.1).

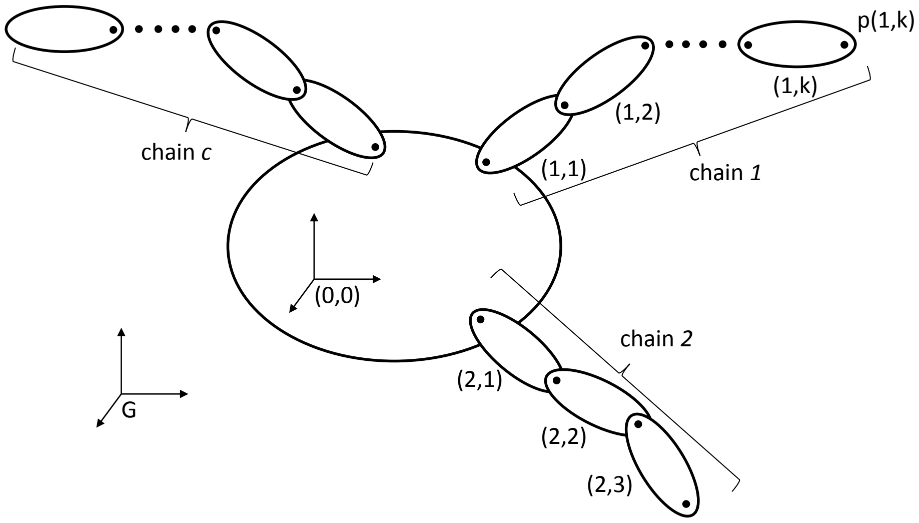

Figure 2.6: Illustration of free base body (0,0) with three serial chains, c, c = 1 and c = 2

Table 2.1: Notation for the discussion of moving base body models

Name	Symbol	Description
Chain	c	Chain number
Frame, body	(c,k)	Chain c, link k
Point	$p_{(c,k)}$	Point p on body c,k
Generalized coordinate	$\mathbf{q}_{(c,k)}$	Joint coordinate at chain c, link k
Vector of joint coordinates	$\mathbf{q}_c$	All coordinates in chain c
Velocity of body fixed point	$\boldsymbol{v}^{(c,k)}_{p(c,k)}$	Velocity of point p in body (c,k) relative to frame (c,k)
Angular velocity of frame	$\boldsymbol{\omega}_{(c_1,k_1),(c_2,k_2)}$	Angular velocity of link (c_1,k_1) relative to (c_2,k_2)
Velocity Jacobian	$\mathbf{J}^{v}_{0(c,k)}$	Jacobian at body origin (c,k) relative to base frame $(c,0)$
	$\mathbf{J}^{v}_{p(c,k)}$	Jacobian at point p on body (c,k) relative to base frame $(c,0)$
	$\mathbf{J}^{v}_{0(c,0)}$	Jacobian at base of chain c, relative to the base body
Angular velocity Jacobian	$\mathbf{J}^{\omega}_{(c,k)}$	Jacobian at body (c,k) relative to base frame $(c,0)$
	$\mathbf{J}^{\omega}_{(0,0)}$	Jacobian at base body relative to the ground frame G.
Mass of link	$m_{(c,k)}$	Mass of link (c,k)
Rotational inertia matrix	$\boldsymbol{I}_{(c,k)}$	-
Total mass matrix	$\mathbf{M}_{sys}$	-
Link mass contribution	$\mathbf{M}_{(c,k)}$	Contribution for link (c,k)
Translation link contribution	$\mathbf{M}^{v}_{(c,k)}$	-
Rotational link contribution	$\mathbf{M}^{\omega}_{(c,k)}$	-
Number of chains	n_c	-
Number of links (coordinates) in chain i	n_{ci}	n_{c1} for chain 1 or n_{c3} for chain 3
Number of coordinates in chain and body	n_{Tki}	Sum of $n_{ci} + n_{c0}$
Total number of coordinates in system	n_T	-

The first step in the analysis of the moving base body dynamics calculates the velocity for a point p fixed in the k^{th} link of the chain c. This is found by adding the velocity of the base of chain c and the relative velocity of p.

$$\mathbf{v}_{p(c,k)} = \mathbf{v}_{c,0} + \mathbf{v}^{(c,0)}_{p(c,k)}$$

The relative velocity of the end of the chain can be found using the velocity Jacobian found in conventional robotic kinematics [24–26]. The relative chain Jacobian in full terms is written as $\mathbf{J}^{vG}_{(c,0),p(c,k)}$ to specify that the velocity is calculated relative to the chain base $(c,0)$, and in the ground frame basis G. The simplified form, to which it is referred to in the following text is $\mathbf{J}^{v}_{p(c,k)}$. We must introduce another Jacobian $\mathbf{J}^{vG}_{G,0(c,0)}$ to find the velocity of the base of the chain relative to the ground frame and in terms of the body coordinates $\dot{\mathbf{q}}_0$. The ground frame G of the chain base Jacobian is suppressed in the subscript and superscript and the Jacobian appears as $\mathbf{J}^{v}_{0,(c,0)}$.

$$\mathbf{v}_{p(c,k)} = \mathbf{J}^{v}_{0(c,0)}\dot{\mathbf{q}}_0 + \mathbf{J}^{v}_{p(c,k)}\dot{\mathbf{q}}_c = \begin{bmatrix} \mathbf{J}^{v}_{0,(c,0)} & \mathbf{J}^{v}_{p(c,k)} \end{bmatrix} \begin{bmatrix} \dot{\mathbf{q}}_0 \\ \dot{\mathbf{q}}_c \end{bmatrix}$$

The angular velocity can be calculated in a similar fashion using two angular velocity Jacobian matrices. The first Jacobian in full form is $\mathbf{J}^{\omega G}_{G,(0,0)}$, indicating the frame $(0,0)$ or free body frame rotates relative to the ground frame G, and in the ground frame basis. The first angular velocity Jacobian is abbreviated as $\mathbf{J}^{\omega}_{(0,0)}$. The second angular velocity Jacobian rotates the body (c,k) fixed in the chain base frame $(c,0)$, and without simplification it is denoted as $\mathbf{J}^{\omega G}_{(c,0),p(c,k)}$. The simplified form used in the text is $\mathbf{J}^{\omega}_{p(c,k)}$.

$$\boldsymbol{\omega}_{(c,k)} = \mathbf{J}^{\omega}_{(0,0)}\dot{\mathbf{q}}_0 + \mathbf{J}^{\omega}_{(c,k)}\dot{\mathbf{q}}_c = \begin{bmatrix} \mathbf{J}^{\omega}_{0(c,0)} & \mathbf{J}^{\omega}_{p(c,k)} \end{bmatrix} \begin{bmatrix} \dot{\mathbf{q}}_0 \\ \dot{\mathbf{q}}_c \end{bmatrix}$$

The kinetic energy of a body k in a chain c is found by calculating the contributions due to translation and rotation. The translational component is found using the mass and the velocities at the center of mass of an link.

$$\begin{aligned}
T^{v}_{(c,k)} &= \frac{1}{2} m_{(c,k)} \mathbf{v}^{\mathrm{T}}_{g(c,k)} \mathbf{v}_{g(c,k)} \\
&= \frac{1}{2} m_{(c,k)} \begin{bmatrix} \dot{\mathbf{q}}_0^{\mathrm{T}} & \dot{\mathbf{q}}_c^{\mathrm{T}} \end{bmatrix} \begin{bmatrix} \left(\mathbf{J}^{v}_{0(c,0)}\right)^{\mathrm{T}} \\ \left(\mathbf{J}^{v}_{g(c,k)}\right)^{\mathrm{T}} \end{bmatrix} \begin{bmatrix} \mathbf{J}^{v}_{0(c,0)} & \mathbf{J}^{v}_{g(c,k)} \end{bmatrix} \begin{bmatrix} \dot{\mathbf{q}}_0 \\ \dot{\mathbf{q}}_c \end{bmatrix} \\
&= \frac{1}{2} m_{(c,k)} \begin{bmatrix} \dot{\mathbf{q}}_0^{\mathrm{T}} & \dot{\mathbf{q}}_c^{\mathrm{T}} \end{bmatrix} \begin{bmatrix} \left(\mathbf{J}^{v}_{0(c,0)}\right)^{\mathrm{T}} \mathbf{J}^{v}_{0(c,0)} & \left(\mathbf{J}^{v}_{0(c,0)}\right)^{\mathrm{T}} \mathbf{J}^{v}_{g(c,k)} \\ \left(\mathbf{J}^{v}_{g(c,k)}\right)^{\mathrm{T}} \mathbf{J}^{v}_{0(c,0)} & \left(\mathbf{J}^{v}_{g(c,k)}\right)^{\mathrm{T}} \mathbf{J}^{v}_{g(c,k)} \end{bmatrix} \begin{bmatrix} \dot{\mathbf{q}}_0 \\ \dot{\mathbf{q}}_c \end{bmatrix}
\end{aligned}$$

The mass matrix contribution from this equation can be extracted from the terms listed below.

$$\begin{bmatrix} \mathbf{M}^{\mathbf{v}}_{0,0} & \mathbf{M}^{\mathbf{v}}_{0,(c,k)} \\ \left(\mathbf{M}^{\mathbf{v}}_{0,(c,k)}\right)^{\mathrm{T}} & \mathbf{M}^{\mathbf{v}}_{(c,k),(c,k)} \end{bmatrix} = m_{(c,k)} \begin{bmatrix} \left(\mathbf{J}^{v}_{(c,0)}\right)^{\mathrm{T}} \mathbf{J}^{v}_{(c,0)} & \left(\mathbf{J}^{v}_{(c,0)}\right)^{\mathrm{T}} \mathbf{J}^{v}_{g(c,k)} \\ \left(\mathbf{J}^{v}_{g(c,k)}\right)^{\mathrm{T}} \mathbf{J}^{v}_{(c,0)} & \left(\mathbf{J}^{v}_{g(c,k)}\right)^{\mathrm{T}} \mathbf{J}^{v}_{g(c,k)} \end{bmatrix}$$

The rotational kinetic energy is found using the inertia matrix $\boldsymbol{I}_{(c,k)}$ and angular velocity Jacobians. The contribution of rotational motion to the generalized mass matrix is calculated in terms of the inertia about the center of mass of each link (c,k).

$$\begin{aligned}
T^{\omega}_{(c,k)} &= \frac{1}{2} \boldsymbol{\omega}^{\mathrm{T}}_{(c,k)} \boldsymbol{I}_{(c,k)} \boldsymbol{\omega}_{(c,k)} \\
&= \frac{1}{2} \begin{bmatrix} \dot{\mathbf{q}}_0^{\mathrm{T}} & \dot{\mathbf{q}}_c^{\mathrm{T}} \end{bmatrix} \begin{bmatrix} \mathbf{J}^{\omega}_{(0,0)} & \mathbf{J}^{\omega}_{(c,k)} \end{bmatrix}^{\mathrm{T}} \boldsymbol{I}_{(c,k)} \begin{bmatrix} \mathbf{J}^{\omega}_{(0,0)} & \mathbf{J}^{\omega}_{(c,k)} \end{bmatrix} \begin{bmatrix} \dot{\mathbf{q}}_0 \\ \dot{\mathbf{q}}_c \end{bmatrix}
\end{aligned}$$

In the above expression, the Jacobian matrices and the inertia matrix must be expressed in terms of the same basis. In this case, the ground frame is implied. The mass matrix contributions from the rotational kinetic energy are computed below.

$$\begin{bmatrix} \mathbf{M}^{\omega}_{0,0} & \mathbf{M}^{\omega}_{0,(c,k)} \\ \left(\mathbf{M}^{\omega}_{0,(c,k)}\right)^{\mathrm{T}} & \mathbf{M}^{\omega}_{(c,k),(c,k)} \end{bmatrix} = \begin{bmatrix} \mathbf{J}^{\omega}_{(0,0)} & \mathbf{J}^{\omega}_{(c,k)} \end{bmatrix}^{\mathrm{T}} \boldsymbol{I}_{(c,k)} \begin{bmatrix} \mathbf{J}^{\omega}_{(0,0)} & \mathbf{J}^{\omega}_{(c,k)} \end{bmatrix}$$

To complete the mass matrix for each chain, the rotational and translational components are summed for each link in the chain.

$$\begin{bmatrix} \mathbf{M}_{0,0} & \mathbf{M}_{0,c} \\ (\mathbf{M}_{0,c})^{\mathrm{T}} & \mathbf{M}_{c,c} \end{bmatrix} = \sum_{k=1}^{n_{c_i}} \begin{bmatrix} \mathbf{M}^{v}_{0,0} & \mathbf{M}^{v}_{0,(c,k)} \\ \left(\mathbf{M}^{v}_{0,(c,k)}\right)^{\mathrm{T}} & \mathbf{M}^{v}_{(c,k),(c,k)} \end{bmatrix} + \begin{bmatrix} \mathbf{M}^{\omega}_{0,0} & \mathbf{M}^{\omega}_{0,(c,k)} \\ \left(\mathbf{M}^{\omega}_{0,(c,k)}\right)^{\mathrm{T}} & \mathbf{M}^{\omega}_{(c,k),(c,k)} \end{bmatrix}$$

The final step in generating the mass matrix for a system in which the base body is in motion is the calculation of the contribution from the base body. The procedure for calculating this contribution is trivial once the chain kinetic energy formulations are completed.

$$\begin{aligned} T^{v}_{(0,0)} &= \frac{1}{2} m_{(0,0)} \mathbf{v}^{\mathrm{T}}_{g(0,0)} \mathbf{v}_{g(0,0)} \\ &= \frac{1}{2} m_{(0,0)} \dot{\mathbf{q}}^{\mathrm{T}}_{0} \left(\mathbf{J}^{\mathbf{v}}_{g(0,0)}\right)^{\mathrm{T}} \mathbf{J}^{\mathbf{v}}_{g(0,0)} \dot{\mathbf{q}}_{0} \\ T^{\omega}_{(0,0)} &= \frac{1}{2} \boldsymbol{\omega}^{\mathrm{T}}_{(0,0)} \boldsymbol{I}_{(0,0)} \boldsymbol{\omega}_{(0,0)} \\ &= \frac{1}{2} \dot{\mathbf{q}}^{\mathrm{T}}_{0} \left(\mathbf{J}^{\omega}_{g(0,0)}\right)^{\mathrm{T}} \boldsymbol{I}_{(0,0)} \mathbf{J}^{\omega}_{g(0,0)} \dot{\mathbf{q}}_{0} \end{aligned}$$

The total mass matrix of the system $\mathbf{M}_{sys}$ can then be compiled from the constituent mass matrices. It will have dimensions $n_T \times n_T$ where $n_T = \sum_{i=0}^{n_c} n_{c_i}$, the sum of the number of links in each chain, for every chain. The base body mass matrix and $\mathbf{M}_{0,0}$ values from each element are summed together. Within each chain the terms $\mathbf{M}_{0,c}$ and $\mathbf{M}_{c,0}$ will compose the first horizontal row and first vertical column, respectively. The diagonal, not including the first element, will be composed of $\mathbf{M}_{c,c}$ from each chain. The matrix $\mathbf{M}_{sys}$ will appear in the form:

$$\mathbf{M}_{sys} = \begin{bmatrix} \mathbf{M}_{0,0} & \mathbf{M}_{0,1} & \mathbf{M}_{0,2} & \cdots & \mathbf{M}_{0,c} \\ \mathbf{M}_{1,0} & \mathbf{M}_{1,1} & 0 & \cdots & 0 \\ \mathbf{M}_{2,0} & 0 & \mathbf{M}_{2,2} & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{M}_{c,0} & 0 & 0 & \cdots & \mathbf{M}_{c,c} \end{bmatrix}$$

The total kinetic energy for the system then appears as:

$$T_{sys} = \frac{1}{2} \begin{bmatrix} \dot{\mathbf{q}}^{\mathrm{T}}_{0} & \dot{\mathbf{q}}^{\mathrm{T}}_{1} & \cdots & \dot{\mathbf{q}}^{\mathrm{T}}_{n_c} \end{bmatrix} \mathbf{M}_{sys} \begin{bmatrix} \dot{\mathbf{q}}_{0} \\ \dot{\mathbf{q}}_{1} \\ \vdots \\ \dot{\mathbf{q}}_{n_c} \end{bmatrix}$$

2.3 Aerodynamic Model

In this section, we discuss the calculation of the contribution of the aerodynamic loads to the equations of motion from a wing composed of a single kinematic chain. We simulate the free base body case by introducing a velocity of the surrounding fluid. The unknown aerodynamic contributions to the governing equations arise from the virtual work performed by the aerodynamic forces. We let $\boldsymbol{r}_{a_i}$ be the position vector of the aerodynamic center of the i^{th} wing section in the inertial frame. The total virtual work is the sum of the external aerodynamic force(s) $\boldsymbol{F}_{a_i}$ for N bodies in a chain dotted with the virtual displacement $\delta\boldsymbol{r}_{a_i}$ of the point of application of $\boldsymbol{F}_{a_i}$.

$$\delta W = \sum_{i=1}^{N} \boldsymbol{F}_{a_i} \cdot \delta\boldsymbol{r}_{a_i} = \sum_{i=1}^{N} Q_{i,a} \cdot \delta q_i$$

By convention there are two rotations, the sideslip β_i and the angle of attack α_i, that are used to define the lift and drag forces on a wing. For a typical calculation on wing section i, we suppose first that the body fixed frame B_i is defined in terms of the unit vectors $\hat{b}_{1,i}$, $\hat{b}_{2,i}$, and $\hat{b}_{3,i}$ at the aerodynamic center of the body. The rotation matrix that maps body fixed B_i frame to the wind frame S_i is constructed by concatenating two single axis rotation matrices. First, an intermediate frame C_i is defined in terms of the unit vectors $\hat{c}_{1,i}$, $\hat{c}_{2,i}$, and $\hat{c}_{3,i}$ that are obtained by rotating the basis vectors of the B_i frame through the angle of attack α_i about the $\hat{c}_{2,i} = \hat{b}_{2,i}$ axis. Figure (2.7) depicts a wing body with body fixed axis $\hat{b}_1$, $\hat{b}_2$, and $\hat{b}_3$ with the velocity $\boldsymbol{v}_{a_i}$ aligned with the wind axis. The wind frame S_i is subsequently defined in terms of the unit vectors $\hat{s}_{1,i}$, $\hat{s}_{2,i}$, and $\hat{s}_{3,i}$ that result when the basis vectors for the intermediate C_i frame are rotated through the sideslip β_i about the $\hat{s}_{3,i} = \hat{c}_{3,i}$ axis. The product of these single axis rotations define the rotation matrix $\mathbf{R}_{B_i}^{S_i}(\alpha_i,\beta_i)$ which can be used to transform the coordinates of any vector in the B_i frame to the S_i frame.

$$\begin{bmatrix} \hat{s}_{1,i} \\ \hat{s}_{2,i} \\ \hat{s}_{3,i} \end{bmatrix} = \begin{bmatrix} \cos\beta_i & \sin\beta_i & 0 \\ -\sin\beta_i & \cos\beta_i & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \cos\alpha_i & 0 & \sin\alpha_i \\ 0 & 1 & 0 \\ -\sin\alpha_i & 0 & \cos\alpha_i \end{bmatrix} \begin{bmatrix} \hat{b}_{1,i} \\ \hat{b}_{2,i} \\ \hat{b}_{3,i} \end{bmatrix}$$

By definition, the lift and drag forces on any wing section constitute the aerodynamic force vector acting on the i^{th} wing section via the expression

$$\boldsymbol{F}_{a,i} = -D_i(\hat{s}_{1,i}) - L_i(\hat{s}_{3,i})$$

Lift and drag are approximated using a standard two-dimensional, quasi-steady aerodynamic model in this thesis. Furthermore, the lift and drag forces are assumed to act at the aerodynamic center and therefore the moment is ignored. We suppose ρ is the density of fluid surrounding the wing, $\mathbf{v}_{a_i}$ is the velocity of the aerodynamic center with respect to the ground frame, $\mathbf{v}_w$ is the velocity of the fluid with respect to the ground frame, and $A_{w,i}$ is the two dimensional wing area enclosed between the leading edge and the trailing edge of the i^{th} wing section. The functions $c_{Di}(\alpha_i)$ and $c_{Li}(\alpha_i)$ are the drag and lift coefficients, respectively, for wing section i as a function of the angle of attack. These are assumed unknown in this paper. See Figure 2.8 for example plots of typical

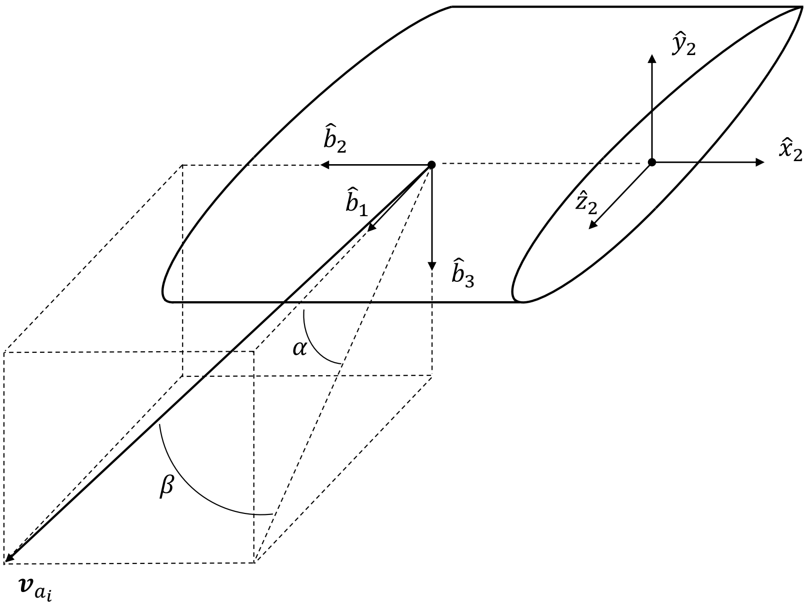

Figure 2.7: Illustration of the body fixed frame on a wing section with α, the angle of attack, and β, the sideslip

functions $c_{Li}(\alpha_i)$ as the angle of attack α_i varies. Pictured is the lift coefficient for an S1223 airfoil at low Reynolds number $Re = (2x10^2)$ measured in wind-tunnel experiments [1]. The lift and drag on the i^{th} section are computed as

$$L_i = \frac{1}{2}\rho||\mathbf{v}_w - \mathbf{v}_{a_i}||^2 c_L(\alpha_i) A_{w,i}$$
$$D_i = \frac{1}{2}\rho||\mathbf{v}_w - \mathbf{v}_{a_i}||^2 c_D(\alpha_i) A_{w,i}$$

The strategy for calculating the virtual work due to aerodynamic forces proceeds by finding the velocities of each aerodynamic center. These velocities are a function of the generalized coordinates.

$$\mathbf{v}_{a_i} = \mathbf{v}_{a_i}(q_1, \ldots q_N)$$

We write the total equivalent velocity of the aerodynamic center of wing section i in the form $\mathbf{v}_{T_i} = \mathbf{v}_w - \mathbf{v}_{a_i} = u_{a_i}\hat{b}_{i,1} + v_{a_i}\hat{b}_{i,2} + w_{a_i}\hat{b}_{i,3}$. The angle of attack α_i and sideslip β_i is defined by this

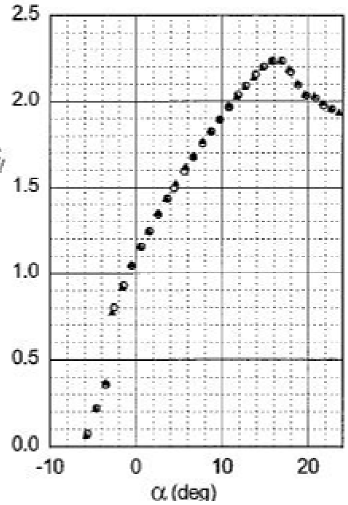

Figure 2.8: Typical graph of the coefficients of drag and lift versus angle of attack, α_i [1]

velocity and can therefore be found using the following formulae:

$$\alpha_i = \tan^{-1}\left(\frac{w_{a_i}(q_1,\ldots q_N,\dot{q}_1,\ldots \dot{q}_N)}{u_{a_i}(q_1,\ldots q_N,\dot{q}_1,\ldots \dot{q}_N)}\right)$$
$$\beta_i = \sin^{-1}\left(\frac{v_{a_i}(q_1,\ldots q_N,\dot{q}_1,\ldots \dot{q}_N)}{\|\mathbf{v}_{T_i}(q_1,\ldots q_N,\dot{q}_1,\ldots \dot{q}_N)\|}\right)$$

The $3 \times N$ velocity Jacobian matrix can be formed to find the vector of aerodynamic center velocities.

$$\mathbf{v}_{a_i} = \mathbf{J}^{v}_{a_i}\dot{\mathbf{q}}$$

This standard calculation for kinematic chains is discussed in numerous references, see [24,25,28]. The rotation matrix $\mathbf{R}^{B_i}_0$ maps the velocity Jacobian from the zero frame to the body fixed frame. As a consequence, the velocity is written

$$\mathbf{v}^{B_i}_{a_i} = \begin{bmatrix} u_{a_i} \\ v_{a_i} \\ w_{a_i} \end{bmatrix} = \mathbf{R}^{B_i}_0 \mathbf{J}^{v_{B_i}}_{a_i}\dot{\mathbf{q}} = \mathbf{J}^{v}_{a_i,B_i}\dot{\mathbf{q}}$$

The virtual displacements of the aerodynamic centers can also be found using the velocity Jacobian in either the zero frame or body fixed frame.

$$\delta \boldsymbol{r}_{a_i} = \mathbf{J}^{v}_{a_i}\delta\mathbf{q}$$
$$\delta \boldsymbol{r}^{B_i}_{a_i} = \mathbf{J}^{v_{B_i}}_{a_i}\delta\mathbf{q} = \mathbf{R}^{B_i}_0 \mathbf{J}^{v}_{a_i}\delta\mathbf{q}$$

With the expressions above, we have enough information to calculate the virtual work due to aerodynamic forces. The drag and lift for each wing section can be transformed to the body fixed frame using the proper rotation matrix.

$$\delta W_i = \boldsymbol{F}_{a,i} \cdot \delta \boldsymbol{r}_a$$

$$\delta W_i = \{-D_i, 0, -L_i\}\mathbf{R}_{B_i}^{S_i}\mathbf{J}_{a_i}^{v_{B_i}}\delta\mathbf{q}$$

It follows that the generalized forces due to aerodynamic loads on each wing section i can be written as

$$\mathbf{Q}_a = \sum_{i=1}^{N} -\left(\mathbf{J}_{a_i}^{v_{B_i}}\right)^{\mathrm{T}}\left(\mathbf{R}_{B_i}^{S_i}\right)^{\mathrm{T}}\begin{Bmatrix} D_i \\ 0 \\ L_i \end{Bmatrix}$$

We introduce the matrix $\mathbf{\Psi}$,

$$\mathbf{\Psi} = -\left[\left(\mathbf{J}_{a_1}^{v_{B_1}}\right)^{\mathrm{T}}\left(\mathbf{R}_{B_1}^{S_1}\right)^{\mathrm{T}} \quad \dots \quad \left(\mathbf{J}_{a_N}^{v_{B_N}}\right)^{\mathrm{T}}\left(\mathbf{R}_{B_N}^{S_N}\right)^{\mathrm{T}}\right]\left[\begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} \cdots \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix}\right]$$

with the right hand component of size $3 \times 2N$. The generalized forces due to the aerodynamics can be written as

$$\mathbf{Q}_a = \mathbf{\Psi}\begin{Bmatrix} D_1 \\ L_1 \\ \vdots \\ D_N \\ L_N \end{Bmatrix} = \mathbf{\Psi}\mathcal{F}$$

Control synthesis will be based on the state space form of the governing equations

$$\begin{bmatrix} \dot{\mathbf{X}}_1 \\ \dot{\mathbf{X}}_2 \end{bmatrix} = \begin{bmatrix} \dot{\mathbf{q}} \\ \ddot{\mathbf{q}} \end{bmatrix} = \begin{bmatrix} \mathbf{X}_2 \\ \mathbf{M}^{-1}\mathbf{n} \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ \mathbf{M}^{-1}\mathbf{\Psi} \end{bmatrix}\mathcal{F} + \begin{bmatrix} \mathbf{0} \\ \mathbf{M}^{-1}\mathbf{B} \end{bmatrix}\boldsymbol{\tau}$$

where $\mathbf{X}_1 = \mathbf{q}$ and $\mathbf{X}_2 = \dot{\mathbf{q}}$. The final step in preparing the governing equations for control synthesis introduces representations of the unknown drag and lift coefficients $c_{Di}(\alpha_i)$ and $c_{Li}(\alpha_i)$ for each wing section i

$$\begin{aligned} \begin{Bmatrix} D_i \\ L_i \end{Bmatrix} &= \frac{1}{2}\rho||\mathbf{v}_w - \mathbf{v}_{a_i}||^2 A_{w,i}\begin{Bmatrix} \sum_i \Omega_{D_i,k}\tilde{\phi}_{D_i,k}(\alpha_i) \\ \sum_k \Omega_{L_i,k}\tilde{\phi}_{L_i,k}(\alpha_i) \end{Bmatrix} \\ &= \begin{bmatrix} \mathbf{\Omega}_{D_i}^T & 0 \\ 0 & \mathbf{\Omega}_{L_i}^T \end{bmatrix}\begin{Bmatrix} \frac{1}{2}\rho||\mathbf{v}_w - \mathbf{v}_{a_i}||^2 A_{w,i}\tilde{\mathbf{\Phi}}_{D_i} \\ \frac{1}{2}\rho||\mathbf{v}_w - \mathbf{v}_{a_i}||^2 A_{w,i}\tilde{\mathbf{\Phi}}_{L_i} \end{Bmatrix} \\ &= \mathbf{\Omega}_i^T\mathbf{\Phi}_i \end{aligned} \tag{2.8}$$

We assemble all the aerodynamic forces and construct

$$\begin{Bmatrix} D_1 \\ L_1 \\ \vdots \\ D_n \\ L_n \end{Bmatrix} = \begin{bmatrix} \mathbf{\Phi}_{D_1}^{\mathrm{T}} & 0 & \cdots & 0 & 0 \\ 0 & \mathbf{\Phi}_{L_1}^{\mathrm{T}} & & & 0 \\ \vdots & & \ddots & & \\ 0 & & & \mathbf{\Phi}_{D_N}^{\mathrm{T}} & 0 \\ 0 & 0 & & 0 & \mathbf{\Phi}_{L_N}^{\mathrm{T}} \end{bmatrix}\begin{bmatrix} \mathbf{\Omega}_{D_1} \\ \mathbf{\Omega}_{L_1} \\ \vdots \\ \mathbf{\Omega}_{D_N} \\ \mathbf{\Omega}_{L_N} \end{bmatrix} = \mathbf{\Phi}^{\mathrm{T}}\mathbf{\Omega}$$

2.4 Examples of Dynamic and Kinematic Models

Three ornithopter prototype models are evaluated in this thesis. These prototypes are products of the flapping flight project undertaken by Dr. Andrew Kurdila and Dr. Javid Bayandor in the Department of Mechanical Engineering at Virginia Tech. This section will focus on kinematic descriptions of these prototypes.

The first prototype was named Daedalus and was the product of senior Mechanical Engineering students [29]. The Daedalus wing was designed as a serial chain robotic mechanism with four rotational degrees of freedom and three parallel degrees of freedom.

The Daedalus model is the simplest of the three designs. Its structure is directly amenable to the standard Denavit-Hartenberg convention for kinematic chains. Figure (2.9) depicts this model with a set of DH frames attached at its corresponding joints. The joints are labeled o, p, q, r, s in Figure (2.9). Figure (2.10) is an image of the Daedalus CAD model with wing covering. The Daedalus model is a four degree of freedom chain and the DH parameters are listed in Table (2.2). As we recall from Section (2.1.3), the parameters are the displacement d_i, offset a_i, rotation θ_i, and twist α_i for each link $i = 1, \cdots, 4$. The mass matrix **M** was generated for the Daedalus model using numerical values for the masses, relative mass locations, and DH parameters other than the generalized coordinates. MATLAB (MathWorks, Natick, Massachusetts) was used for symbolic computation and simplification. Reasonable estimates of these variables were provided in order to further simplify the mass matrix. The length parameters are listed in meters, masses are in kilograms, and angles are in radians. Tables (2.3), (2.4), and (2.6) lists these values. Moments

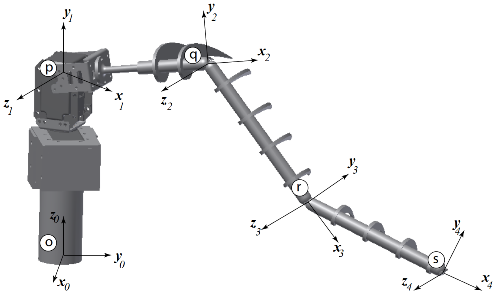

Figure 2.9: Daedalus CAD model with Denavit-Hartenberg axes attached

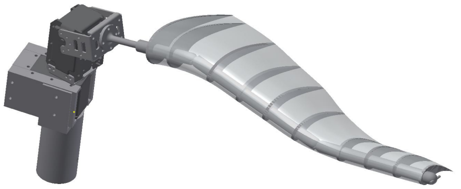

Figure 2.10: Daedalus CAD model with skin covering

Table 2.2: Daedalus DH parameters

Link	d_i	a_i	θ_i	α_i
1	$r_{o,p}$	0	$\theta_1(t)$	$\frac{\pi}{2}$
2	0	$r_{p,q}$	$\theta_2(t)$	0
3	0	$r_{q,r}$	$\theta_3(t)$	0
4	0	$r_{r,s}$	$\theta_4(t)$	0

Table 2.3: Daedalus sample DH parameters

Link	$d_i(m)$	$a_i(m)$	$\theta_i(rad)$	$\alpha_i(rad)$
1	0.1	0	$\theta_1(t)$	$\frac{\pi}{2}$
2	0	0.1	$\theta_2(t)$	0
3	0	0.1	$\theta_3(t)$	0
4	0	0.1	$\theta_4(t)$	0

of inertia are denoted as $I_{i,xx}, I_{i,xy}, I_{i,xz}, I_{i,yx}, I_{i,yy}, \cdots I_{i,zz}$ where i is the link number for $i = 1, \cdots, 4$. They are established with respect to a coordinate system parallel to the i frame but having an origin at the i^{th} center of mass. The diagonal terms $I_{i,xx}, I_{i,yy}$, and $I_{i,zz}$ are calculated using a simplified thin, rigid rod assumption. The two axes perpendicular to the length of the appropriate link are estimated using $\frac{1}{12}ml^2$, with the mass and length of the link, while the axis parallel to the link as well as the other non-diagonal moment of inertia terms are deemed negligible. The mass matrix is listed as elements $m_{11} \cdots m_{44}$. See Appendix (A) for the full listing of the Daedalus mass matrix and nonlinear right hand side.

The Daedalus joint structure was revised to improve performance and increase correlation with biological kinematics. Rearrangement of the joints allows for the wing to create an optimal angle of attack, while reducing detrimental drag and negative lift forces on the upstroke. Additional degrees of freedom allow for the different flapping modes analyzed in Han [3], which he titles flapping, folding and Lead/Lag. The combination of all three produces the highest lift coefficient and the combination of flapping and folding lead to the highest thrust production.

The second model that we study is based on the Larus prototype. The prototype for this project was developed between Fall 2011 and Spring 2012. There were nine senior Mechanical Engineering students involved [30]. This model contains four degrees of freedom, but three act at the shoulder area of the wing, with only one degree of freedom embedded in the wing itself. See Figure (2.11)

Table 2.4: Daedalus sample mass parameters

Link	$m_i(kg)$
1	0.05
2	0.05
3	0.05
4	0.025

Table 2.5: Daedalus moment of inertia parameters, $kg \cdot m^2 \times 10^{-5}$

Link	$I_{i,xx}$	$I_{i,xy}$	$I_{i,xz}$	$I_{i,yy}$	$I_{i,yz}$	$I_{i,zy}$	$I_{i,zz}$
1	4.1667	0	0	0	0	0	4.1667
2	0	0	0	4.1667	0	0	4.1667
3	0	0	0	4.1667	0	0	4.1667
4	0	0	0	2.0833	0	0	2.0833

Table 2.6: Daedalus sample relative center of mass positions

Link	x coordinate(m)	y coordinate(m)	z coordinate(m)
1	0	0.0333	0
2	0.05	0	0
3	0.05	0	0
4	0.05	0	0

for an illustration of the DH frames. One unique feature is the linkage system between frame 3 and frame 4, r and s in the figure. A symmetric four-bar linkage is incorporated in this section of the wing which couples the rotation about θ_4 and an additional rotation θ_5. The constraint is modeled in the dynamics by adding the contribution of the fifth body, T_5 to the existing terms $T_{1,4}$.

$$T = T_{1,4} + T_5 = T_{1,4} + \frac{1}{2}m_5\mathbf{v}_5^{\mathrm{T}}\mathbf{v}_5 + \frac{1}{2}\boldsymbol{\omega}_5^{\mathrm{T}}\boldsymbol{I}_5\boldsymbol{\omega}_5$$

The revised joint structure is modeled using a set of frames generated by the DH Convention in Figure (2.11). The origin of the frames are labeled o, p, q, r, s, and the generalized coordinates are denoted $\theta_1(t) \cdots \theta_4(t)$. We again employ the DH parameters for links $1 \cdots 4$ and amend our kinetic energy calculation to include the contribution from the last mass. As with the Daedalus model, a symbolic mass matrix was computed in terms of the generalized coordinates and inertia properties. See Appendix (A) for full listing of the Larus mass matrix. Tables (2.8 - 2.11) list the parameters for the Larus model.

The third robotic bird prototype currently in development is named Icarus. The design team is composed of senior mechanical engineering students [31]. The Icarus kinematic model is based on a free base body structure with two articulated wings. The DH frames of the left wing and a

Table 2.7: DH Parameters for the Larus model

Link	d_i	a_i	θ_i	α_i
1	$d_{o,p}$	$a_{o,p}$	$\theta_1(t)$	$\frac{\pi}{2}$
2	0	$a_{p,q}$	$\theta_2(t)$	$\frac{\pi}{2}$
3	$d_{q,r}$	0	$\theta_3(t)$	0
4	0	$a_{r,s}$	$\theta_4(t)$	0
5	0	0	$\theta_5(\theta_3, t)$	0

Table 2.8: Larus sample DH parameters

Link	$d_i(m)$	$a_i(m)$	$\theta_i(rad)$	$\alpha_i(rad)$
1	0.02	0.01	$\theta_1(t)$	$\frac{\pi}{2}$
2	0	0.01	$\theta_2(t)$	0
3	0.02	0	$\theta_3(t)$	0
4	0	0.2	$\theta_4(t)$	0
5	0	0	$\theta_5(\theta_3, t)$	0

Table 2.9: Larus sample mass parameters

Link	$m_i(kg)$
1	0.05
2	0.05
3	0.05
4	0.025
5	0.01

Table 2.10: Larus moment of inertia parameters, $kg \cdot m^2 \times 10^{-5}$

Link	$I_{i,xx}$	$I_{i,xy}$	$I_{i,xz}$	$I_{i,yy}$	$I_{i,yz}$	$I_{i,zy}$	$I_{i,zz}$
1	4.1667	0	0	0	0	0	4.1667
2	0	0	0	4.1667	0	0	4.1667
3	0	0	0	4.1667	0	0	4.1667
4	0	0	0	4.1667	0	0	4.1667
5	0	0	0	2.0833	0	0	2.0833

Table 2.11: Larus sample relative center of mass positions

Link	x coordinate(m)	y coordinate(m)	z coordinate(m)
1	−0.005	−0.005	0
2	−0.0025	0	0.005
3	0	−0.05	0.1
4	−0.05	0.1	0
5	0.05	0	0

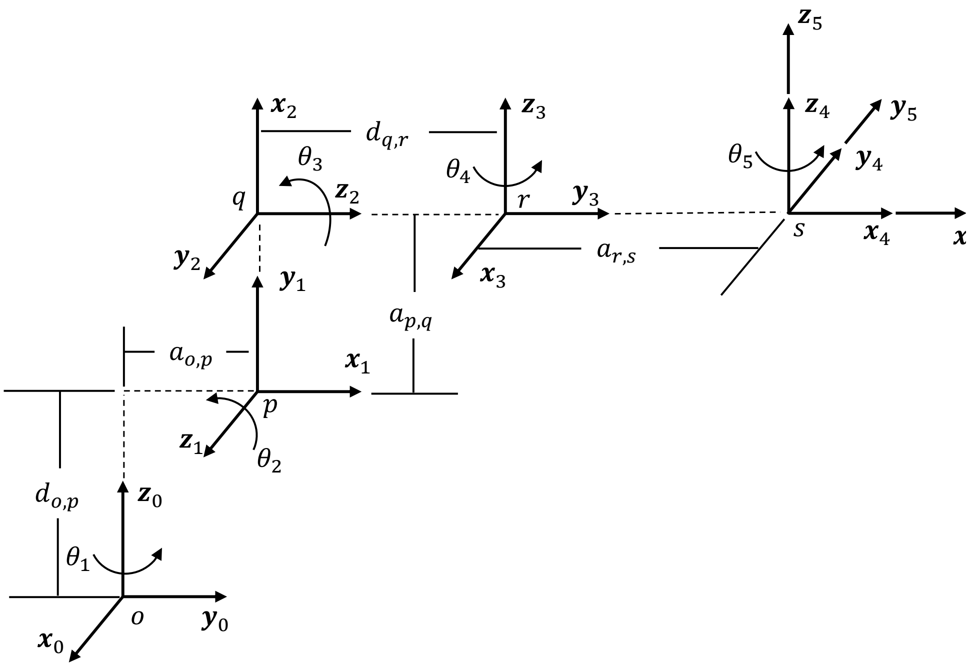

Figure 2.11: DH frames of the Larus prototype

Table 2.12: Icarus right wing DH parameters

Link	$d_i(m)$	$a_i(m)$	$\theta_i(rad)$	$\alpha_i(rad)$
(R,1)	0	$a_{o,p}$	$\theta_1(t)$	0
(R,2)	0	$a_{p,q}$	$\theta_2(\theta_1,t)$	$\frac{\pi}{2}$
(R,3)	0	0	$\theta_3(t)$	$\frac{\pi}{2}$
(R,4)	$d_{q,r}$	0	$\theta_4(t)$	0

base body frame are presented in Figure (2.12), while the right wing frames and base body frame are pictured in Figure (2.13). In this figure, the base body frame is notated as $(0,0)$, while the left wing chain base is denoted $(L,0)$. The left wing is labeled as chain L, while the right wing is labeled chain R. The notation used in this model follows the convention from Section (2.2.2). There are five degrees of freedom, θ_1, θ_3, θ_4, θ_5, and θ_6. The first degree of freedom θ_1 drives both wings. The second body on both wings is rotated in frame 1 by θ_2. The rotation θ_2 is a function of θ_1 due to a mechanical constraint. The last two degrees of freedom on both wings are driven independently by variables θ_3 through θ_6. The DH parameters for the two wings are tabulated in Tables (2.13) and (2.12). In the left wing, the rotations for all of the links are inverted so that the first and second angles are symmetric about the $x-z$ base body plane. The last two rotations on the left wing are also negated so that if $\theta_3 = \theta_5$ and $\theta_4 = \theta_6$, the rotations will also be symmetric about the base body.

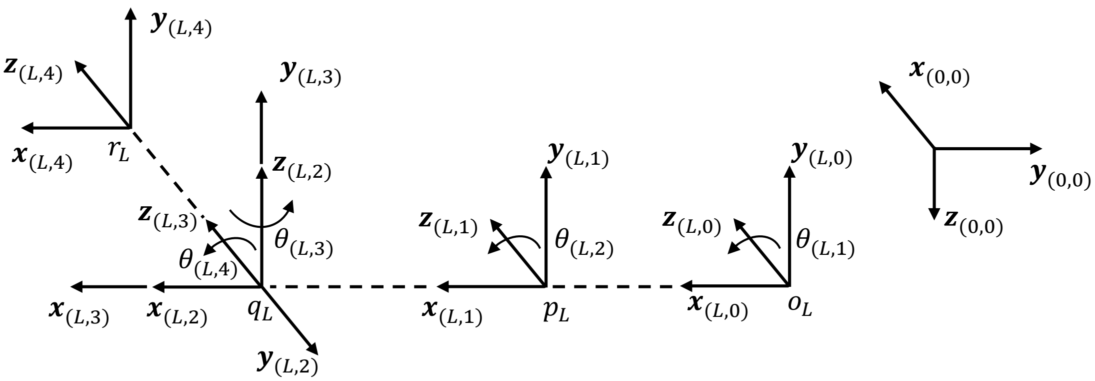

Figure 2.12: Left wing DH frames of the Icarus prototype pictured with $\theta_{(L,0)}$ through $\theta_{(L,4)}$ equal to zero

Table 2.13: Icarus left wing DH parameters

Link	$d_i(m)$	$a_i(m)$	$\theta_i(rad)$	$\alpha_i(rad)$
(L,1)	0	$a_{o,p}$	$-\theta_1(t)$	0
(L,2)	0	$a_{p,q}$	$-\theta_2(\theta_1,t)$	$\frac{3\pi}{2}$
(L,3)	0	0	$-\theta_5(t)$	$\frac{\pi}{2}$
(L,4)	$d_{q,r}$	0	$-\theta_6(t)$	0

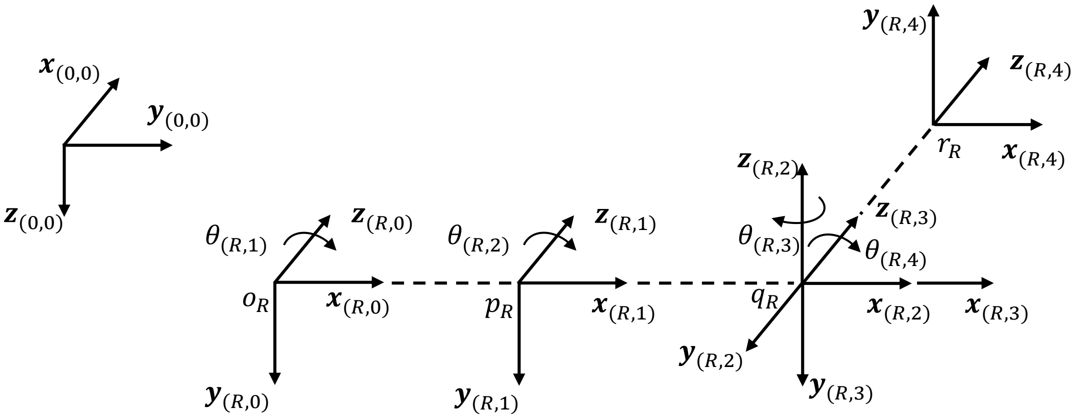

Figure 2.13: Right wing DH frames of the Icarus prototype pictured with $\theta_{(R,1)}$ through $\theta_{(R,4)}$ equal to zero

Chapter 3

Control of Flapping Wing Robots

This section investigates some methods of closed loop control for flapping wing robots that have a fixed root or core body. From the generalized equations of motion, there are two terms that present difficulties in developing a controller. The first is the non-linearities that arise from the coupled equations of motion. These are represented as $\mathbf{n}$ in Equation (2.6). This problem has been addressed in the robotics literature [24–26, 32] via a number of techniques including approximate dynamic inversion or computed torque control. When parameters such as dimensions, masses, and other inertial properties are known exactly, this method is applicable. For many robotic systems, such as the control of pick and place tasks for manipulator arms or control of limbs in a humanoid robot, this method is sufficient to guarantee stability. Even if the parameters are unknown, but the uncertainty is relatively small, robustness analysis can be used to achieve some measure of stability. In flapping wing vehicles, a new challenge arises. The second term that presents difficulty in control design for flapping wing vehicles is $\mathbf{n}_a$ in Equation (2.6). This term is a function of the states $\mathbf{q}$ and their derivatives $\dot{\mathbf{q}}$. Since this function is based on complex, unsteady, nonlinear aerodynamic phenomenon, a model with high accuracy is not usually available. This thesis explores the use of some adaptive control techniques which are able to generate an adaptive approximation of the unknown aerodynamics terms and control the resulting motion.

3.1 Approximate Dynamic Inversion

Nonlinear systems, as they are generated by nonlinear ordinary differential equations (ODE's), can pose challenging control synthesis problems. A substantial effort in literature has been made that seeks to transform a system of nonlinear ODE's into a system of linear ODE's in certain cases. This method is known as feedback linearization, computed torque control, or dynamic inversion [24,28]. From Section 2.2 the equation of motion for natural systems that arise when we model a robotic linkage are in the form

$$\mathbf{M}(\mathbf{q}(t))\ddot{\mathbf{q}}(t) = \mathbf{n}(\mathbf{q}(t), \dot{\mathbf{q}}(t), t) + \mathbf{B}(\mathbf{q}(t))\boldsymbol{\tau}(t)$$

The computed torque control strategy makes a careful selection of the control law to yield a linear system of ODE's [25].

$$\boldsymbol{\tau} = \mathbf{M}\boldsymbol{v} - \mathbf{n}$$

The new outer loop control input $\boldsymbol{v}$ is then chosen based on well developed linear control techniques. One popular feedback law used in dynamic inversion is the PD, or proportional derivative, control law [24]. The feedforward control $\ddot{\mathbf{q}}_d$ is generally included but can be set to 0 for setpoint control. The desired states $\mathbf{q}_d$ and the derivative of the desired states $\dot{\mathbf{q}}_d$ can be set to a constant for setpoint control or an input function for tracking control.

$$\boldsymbol{v} = \ddot{\mathbf{q}}_d - \boldsymbol{G}_0(\mathbf{q} - \mathbf{q}_d) - \boldsymbol{G}_1(\dot{\mathbf{q}} - \dot{\mathbf{q}}_d)$$

This controller is effective in theory, but is not fully realizable in practice. For feedback linearization to be effective upon implementation, the system parameters **M** and **n** must be known exactly. This is not possible in many real life applications, where the mass, inertia, dimensions, input force and other parameters are known approximately, and some degree of uncertainty always exists. Approximate dynamic inversion or robust dynamic inversion is one method that has been developed to handle uncertainty in the system dynamics. In this framework it is assumed that the actual mass matrix **M** is approximated by the predicted mass matrix $\mathbf{M}_p$. The predicted mass matrix $\mathbf{M}_p$ depends on the estimates of the unknown parameters. The approximation error or discrepancy between the predicted and actual value is designated with the tilde symbol and is calculated as

$$\tilde{\mathbf{M}} = \mathbf{M}_p - \mathbf{M}$$

Likewise, we introduce a predicted nonlinear term denoted $\mathbf{n}_p$, and compute the approximation error $\tilde{\mathbf{n}} = \mathbf{n}_p - \mathbf{n}$. External disturbance inputs are considered in $\boldsymbol{\tau}_d$. The approximate inversion controller takes the form

$$\boldsymbol{\tau} = \mathbf{M}_p\boldsymbol{v} - \mathbf{n}_p$$

Several standard controllers have been based on this structure. The system equation after approximate dynamic inversion now becomes

$$\ddot{\mathbf{q}} = \boldsymbol{v} + \boldsymbol{d}$$

The discrepancy $\boldsymbol{d}$ is found to be

$$\boldsymbol{d} = \mathbf{M}^{-1}\tilde{\mathbf{M}}\boldsymbol{v} - \mathbf{M}^{-1}\tilde{\mathbf{n}} + \mathbf{M}^{-1}\boldsymbol{\tau}_d$$

Once the approximate dynamic inversion control is used, the choice of the outer loop controller remains. We discuss two common choices for the outer loop controller, a switching controller and a smoothed switching controller.

3.1.1 Hard Switching Controller

By introducing a discontinuous control input, namely a 'hard switching' controller, the output can remain asymptotically stable despite the mismatch in dynamic properties. This method for handling uncertainties has appeared as early as 1991 in [32]. This control law defines several new matrices and constants. The matrix **A** is a $2N \times 2N$ gain matrix.

$$\mathbf{A} = \begin{bmatrix} 0 & \boldsymbol{I} \\ -\boldsymbol{G}_0 & -\boldsymbol{G}_1 \end{bmatrix}$$

This controller is derived from a Lyapunov equation. We choose a symmetric positive definite matrix **Q**. For simplicity it can be selected as $\mathbf{Q} = \boldsymbol{I}$. Then, the symmetric positive definite matrix **P**, with dimensions $N \times N$, is defined to be the solution of the Lyapunov equation

$$\mathbf{A}^{\mathrm{T}}\mathbf{P} + \mathbf{P}\mathbf{A} = -\mathbf{Q}$$

The $2N \times N$ matrix **D** is defined as

$$\mathbf{D} = \begin{bmatrix} 0 \\ \boldsymbol{I} \end{bmatrix}$$

Lastly, **X** is the vector of state error and state error derivative, and a positive constant k is selected to be greater than $||\boldsymbol{d}||$. The switching control input **u** is added to the outer loop control $\boldsymbol{v}$

$$\mathbf{u} = \begin{cases} -k\frac{\mathbf{D}^{\mathrm{T}}\mathbf{P}\mathbf{X}}{||\mathbf{D}^{\mathrm{T}}\mathbf{P}\mathbf{X}||} & if \quad \mathbf{D}^{\mathrm{T}}\mathbf{P}\mathbf{X} \neq 0 \\ 0 & if \quad \mathbf{D}^{\mathrm{T}}\mathbf{P}\mathbf{X} = 0 \end{cases}$$

The control input τ with the new hard switching input **u** becomes

$$\tau = \mathbf{M}_p(\boldsymbol{v} + \mathbf{u}) - \mathbf{n}_p$$

The system equations then become

$$\ddot{\mathbf{q}} = \boldsymbol{v} + \boldsymbol{d} + \mathbf{u}$$

To investigate the stability, a Lyapunov function is chosen as $\mathcal{V} = \frac{1}{2}\mathbf{X}^{\mathrm{T}}\mathbf{P}\mathbf{X}$. The Lyapunov derivative is

$$\begin{aligned} \dot{\mathcal{V}} =& \frac{1}{2}(\mathbf{X}^{\mathrm{T}}\mathbf{P}\dot{\mathbf{X}} + \dot{\mathbf{X}}^{\mathrm{T}}\mathbf{P}\mathbf{X}) \\ =& \mathbf{X}^{\mathrm{T}}(\mathbf{P}\mathbf{A} + \mathbf{A}^{\mathrm{T}}\mathbf{P})\mathbf{X} + \mathbf{X}^{\mathrm{T}}\mathbf{P}\mathbf{D}(\mathbf{u} + \boldsymbol{d}) \end{aligned}$$

The term $\mathbf{P}\mathbf{A} + \mathbf{A}^{\mathrm{T}}\mathbf{P}$ is the left side of the Lyapunov equation and equates to $-\mathbf{Q}$. If looking at the second controller mode, $\mathbf{D}^{\mathrm{T}}\mathbf{P}\mathbf{X} = 0$, the Lyapunov derivative simplifies to the negative definite formulation:

$$\dot{\mathcal{V}} = -\mathbf{X}^{\mathrm{T}}\mathbf{Q}\mathbf{X}$$

If instead, the other controller mode is used, the Lyapunov derivative is equal to

$$\dot{\mathcal{V}} = -\mathbf{X}^{\mathrm{T}}\mathbf{Q}\mathbf{X} + \mathbf{X}^{\mathrm{T}}\mathbf{P}\mathbf{D}\boldsymbol{d} - k||\mathbf{D}^{\mathrm{T}}\mathbf{P}\mathbf{X}||$$

If we assume that $||\boldsymbol{d}|| \leq k$, the second and third terms in $\dot{\mathcal{V}}$ can be written as

$$||\mathbf{D}^{\mathrm{T}}\mathbf{P}\mathbf{X}||(||\boldsymbol{d}||-k) < 0$$

The Lyapunov function then becomes negative definite in the form:

$$\mathcal{V} \leq -\mathbf{X}^{\mathrm{T}}\mathbf{Q}\mathbf{X}$$

3.1.2 Soft Switching Controller

One issue that commonly results in implementation of the hard switching controller is chattering type effects. To lessen the effects of chattering, a soft switching control can be implemented [25]. The control input is smoothed using a parameter ϵ. Instead of completely switching off, the input will be reduced when $||\mathbf{D}^{\mathrm{T}}\mathbf{PX}||< \epsilon$.

$$\mathbf{u} = \begin{cases} -k\frac{\mathbf{D}^{\mathrm{T}}\mathbf{PX}}{||\mathbf{D}^{\mathrm{T}}\mathbf{PX}||} & if \quad ||\mathbf{D}^{\mathrm{T}}\mathbf{PX}||\geq \epsilon \\ -\frac{k}{\epsilon}\mathbf{D}^{\mathrm{T}}\mathbf{PX} & if \quad ||\mathbf{D}^{\mathrm{T}}\mathbf{PX}||< \epsilon \end{cases}$$

The Lyapunov function $\mathcal{V} = \frac{1}{2}\mathbf{X}^{\mathrm{T}}\mathbf{PX}$ is used again to investigate the revised input. For the first case of the soft switching controller, the Lyapunov derivative will be less than zero if $||\boldsymbol{d}||\leq k$. When $||\mathbf{D}^{\mathrm{T}}\mathbf{PX}||< \epsilon$ the Lyapunov derivative is

$$\begin{aligned} \dot{\mathcal{V}} &= -\frac{1}{2}\mathbf{X}^{\mathrm{T}}\mathbf{QX} - \frac{k}{\epsilon}||\mathbf{D}^{\mathrm{T}}\mathbf{PX}||^2 + \mathbf{X}^{\mathrm{T}}\mathbf{PD}\boldsymbol{d} \\ &\leq -\frac{1}{2}\mathbf{X}^{\mathrm{T}}\mathbf{QX} + ||\mathbf{D}^{\mathrm{T}}\mathbf{PX}||\left(||\boldsymbol{d}|| - \frac{k}{\epsilon}||\mathbf{D}^{\mathrm{T}}\mathbf{PX}||\right) \\ &\leq -\frac{1}{2}\mathbf{X}^{\mathrm{T}}\mathbf{QX} + 2\epsilon k \end{aligned}$$

Using LaSalle's Invariance Principle, the error is bounded and stable and depends on the term $2\epsilon k$. The subset $\mathcal{M}$ is defined as the largest positive invariant subset where

$$\mathcal{M} := \left\{\mathbf{X} : \frac{1}{2}\mathbf{X}^{\mathrm{T}}\mathbf{QX} \leq 2\epsilon k\right\}$$

3.2 Adaptive Control

3.2.1 Approximate Dynamic Inversion with Adaptation

The controllers derived via approximate dynamic inversion in the last section are recommended provided that the uncertainty is not too great. A detailed discussion of the size of the uncertainty and its relation to closed loop stability can be found in [28], for example. For flapping wing robots, the unknown aerodynamic terms can have the same order of magnitude as the inertial terms. It is doubtful that robustness methods can yield agile and accurate tracking response for such systems. In this section, we present two adaptive control methods that are appropriate for the flapping wing robotic systems. The equations of motion derived from Lagranges Equation with the aerodynamic term takes the form

$$\mathbf{M}(\mathbf{q}(t))\ddot{\mathbf{q}}(t) = \mathbf{n}(\mathbf{q}(t), \dot{\mathbf{q}}(t), t) + \mathbf{n}_a(\mathbf{q}(t), \dot{\mathbf{q}}(t), t) + \mathbf{B}(\mathbf{q}(t))\boldsymbol{\tau}(t)$$

The dynamic inversion inner loop control is defined in terms of the outer loop control $\boldsymbol{\nu}$.

$$\boldsymbol{\tau} = \mathbf{M}\boldsymbol{\nu} - \mathbf{n}$$

After manipulation, the closed loop equations of motion becomes

$$\ddot{\mathbf{q}} = \mathbf{M}^{-1}\mathbf{n}_a + \boldsymbol{\nu} \tag{3.1}$$

The outer loop is chosen with a simple PD control but the unknown nonlinear term still remains. The state error is defined as $\tilde{\mathbf{q}} = \mathbf{q} - \mathbf{q}_d$. As we recall from Section 2.3, $\mathbf{n}_a = \boldsymbol{\Psi}\boldsymbol{\Omega}^{\mathrm{T}}\boldsymbol{\Phi}$ is comprised of the matrix $\boldsymbol{\Psi}$, the vector of unknown parameters $\boldsymbol{\Omega}$, and the basis vector $\boldsymbol{\Phi}$. The outer loop control is selected to be

$$\boldsymbol{\nu} = \ddot{\mathbf{q}}_d - \boldsymbol{G}_0\tilde{\mathbf{q}} - \boldsymbol{G}_1\dot{\tilde{\mathbf{q}}} - \mathbf{M}^{-1}\boldsymbol{\Psi}\hat{\boldsymbol{\Omega}}^{\mathrm{T}}\boldsymbol{\Phi} \tag{3.2}$$

where $\hat{\boldsymbol{\Omega}}$ is a vector of estimates of the unknown parameters $\boldsymbol{\Omega}$. By inserting the outer loop control law from Equation (3.2) into the closed loop equation of motion in Equation (3.1), the new closed loop equation of motion now becomes:

$$\ddot{\mathbf{q}} = \mathbf{M}^{-1}\mathbf{n}_a + \ddot{\mathbf{q}}_d - \boldsymbol{G}_0\tilde{\mathbf{q}} - \boldsymbol{G}_1\dot{\tilde{\mathbf{q}}} - \mathbf{M}^{-1}\boldsymbol{\Psi}\hat{\boldsymbol{\Omega}}^{\mathrm{T}}\boldsymbol{\Phi}$$

By decomposing $\mathbf{n}_a$ into its terms $\boldsymbol{\Psi}\boldsymbol{\Omega}^{\mathrm{T}}\boldsymbol{\Phi}$, and introducing an unknown parameter error term $\tilde{\boldsymbol{\Omega}} = \boldsymbol{\Omega} - \hat{\boldsymbol{\Omega}}$ this equation can be simplified as:

$$\ddot{\mathbf{q}} = \ddot{\mathbf{q}}_d - \boldsymbol{G}_0\tilde{\mathbf{q}} - \boldsymbol{G}_1\dot{\tilde{\mathbf{q}}} + \mathbf{M}^{-1}\boldsymbol{\Psi}\tilde{\boldsymbol{\Omega}}^{\mathrm{T}}\boldsymbol{\Phi} \tag{3.3}$$

The Lyapunov function candidate is chosen as:

$$\mathcal{V} = \frac{1}{2}\dot{\tilde{\mathbf{q}}}^{\mathrm{T}}\dot{\tilde{\mathbf{q}}} + \frac{1}{2}\gamma\tilde{\boldsymbol{\Omega}}^{\mathrm{T}}\tilde{\boldsymbol{\Omega}} + \frac{1}{2}\tilde{\mathbf{q}}^{\mathrm{T}}\boldsymbol{G}_0\tilde{\mathbf{q}}$$

where γ is the learning gain. The derivative of the Lyapunov Function along the trajectories of the system can be written as

$$\dot{\mathcal{V}} = \dot{\tilde{\mathbf{q}}}^{\mathrm{T}}\ddot{\tilde{\mathbf{q}}} + \dot{\tilde{\mathbf{q}}}^{\mathrm{T}}\boldsymbol{G}_0\tilde{\mathbf{q}} + \gamma\dot{\tilde{\boldsymbol{\Omega}}}^{\mathrm{T}}\tilde{\boldsymbol{\Omega}}$$

By collecting like terms, the derivative of the Lyapunov Candidate is rewritten.

$$\dot{\mathcal{V}} = \dot{\tilde{\mathbf{q}}}^{\mathrm{T}}(\ddot{\tilde{\mathbf{q}}} + \boldsymbol{G}_0\tilde{\mathbf{q}}) + \gamma\dot{\tilde{\boldsymbol{\Omega}}}^{\mathrm{T}}\tilde{\boldsymbol{\Omega}}$$

Equation (3.3) can be rearranged to solve for $\ddot{\tilde{\mathbf{q}}} + \boldsymbol{G}_0\tilde{\mathbf{q}}$, then substituted into the Lyapunov function derivative.

$$\ddot{\tilde{\mathbf{q}}} + \boldsymbol{G}_0\tilde{\mathbf{q}} = -\boldsymbol{G}_1\dot{\tilde{\mathbf{q}}} + \mathbf{M}^{-1}\boldsymbol{\Psi}\tilde{\boldsymbol{\Omega}}^{\mathrm{T}}\boldsymbol{\Phi}$$

$$\dot{\mathcal{V}} = -\dot{\tilde{\mathbf{q}}}^{\mathrm{T}}\boldsymbol{G}_1\dot{\tilde{\mathbf{q}}} + \dot{\tilde{\mathbf{q}}}^{\mathrm{T}}\mathbf{M}^{-1}\boldsymbol{\Psi}\tilde{\boldsymbol{\Omega}}^{\mathrm{T}}\boldsymbol{\Phi} + \gamma\dot{\tilde{\boldsymbol{\Omega}}}^{\mathrm{T}}\tilde{\boldsymbol{\Omega}}$$

For the system to be stable, the Lyapunov Function must be positive definite and the derivative of the Lyapunov Function must be negative definite [32]. The Lyapunov Function is positive definite. The terms in $\dot{\mathcal{V}}$ are not all negative definite. For this reason, we would like to factor out $\tilde{\boldsymbol{\Omega}}$ and cancel the other terms. We consider a case where the aerodynamic forces act only on a single body, i.e. $\mathbf{\Omega}$ and $\mathbf{\Phi}$ are column and row vectors. With this assumption, $\tilde{\mathbf{\Omega}}^{\mathrm{T}}\mathbf{\Phi} \equiv \mathbf{\Phi}^{\mathrm{T}}\tilde{\mathbf{\Omega}}$ and we can write

$$\dot{\mathcal{V}} = -\dot{\tilde{\mathbf{q}}}^{\mathrm{T}}\boldsymbol{G}_1\dot{\tilde{\mathbf{q}}} + (\dot{\tilde{\mathbf{q}}}^{\mathrm{T}}\mathbf{M}^{-1}\mathbf{\Psi}\mathbf{\Phi}^{\mathrm{T}} + \gamma\dot{\tilde{\mathbf{\Omega}}}^{\mathrm{T}})\tilde{\mathbf{\Omega}}$$

The derivative of the unknown parameter error $\dot{\tilde{\mathbf{\Omega}}} = -\dot{\hat{\mathbf{\Omega}}}$ since $\mathbf{\Omega}$ is constant. By solving for $\dot{\hat{\mathbf{\Omega}}}$ the update law is defined.

$$\dot{\tilde{\mathbf{q}}}^{\mathrm{T}}\mathbf{M}^{-1}\mathbf{\Psi}\mathbf{\Phi}^{\mathrm{T}} + \gamma\dot{\tilde{\mathbf{\Omega}}}^{\mathrm{T}} = 0$$

$$\dot{\hat{\mathbf{\Omega}}} = \gamma^{-1}\mathbf{\Phi}\mathbf{\Psi}^{\mathrm{T}}\mathbf{M}^{-1}\dot{\tilde{\mathbf{q}}}$$

A positive definite Lyapunov function $\mathcal{V}$ and negative semi-definite Lyapunov derivative $\dot{\mathcal{V}}$ guarantees boundedness for $\tilde{\mathbf{q}}$,$\dot{\tilde{\mathbf{q}}}$, and $\tilde{\mathbf{\Omega}}$. Since $\dot{\tilde{\mathbf{q}}}$ is negative definite in $\dot{\mathcal{V}}$ and positive definite in $\mathcal{V}$, it is guaranteed to converge.

3.2.2 Dissipative Controller with Adaptation

To improve performance and guarantee asymptotic stability in both the position and derivative error, a dissipative controller is investigated. The equation of motion is rewritten with the nonlinear term $\mathbf{n}$ broken down into the Coriolis matrix $\mathbf{C}$ and the gradient of the potential energy $\frac{\partial V}{\partial \mathbf{q}}$

$$\mathbf{M}(\mathbf{q}(t))\ddot{\mathbf{q}}(t) + \mathbf{C}(\mathbf{q}(t), \mathbf{q}\dot{(t)}) + \frac{\partial V}{\partial \mathbf{q}(t)} - \mathbf{n}_a(\mathbf{q}(t), \dot{\mathbf{q}}(t), t) = \mathbf{B}(\mathbf{q}(t))\boldsymbol{\tau}(t) \tag{3.4}$$

A new measure of the tracking error $\boldsymbol{\sigma}$ is introduced with diagonal, positive gain matrix $\mathbf{\Lambda}$:

$$\boldsymbol{\sigma} = \dot{\tilde{\mathbf{q}}} + \mathbf{\Lambda}\tilde{\mathbf{q}}$$

The control input is selected as

$$\tau = \mathbf{M}(\mathbf{q})(\ddot{\mathbf{q}} + \dot{\sigma}) + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})(\dot{\mathbf{q}} + \sigma) + \frac{\partial V}{\partial \mathbf{q}} + \boldsymbol{G}_D\boldsymbol{\sigma} - \Psi\mathbf{\Phi}^{\mathrm{T}}\hat{\mathbf{\Omega}}$$

It is assumed that the actuators are revolute and $\mathbf{B}$ is the identity matrix. The control input is substituted into Equation 3.4 to obtain:

$$\mathbf{M}(\mathbf{q})(-\dot{\sigma}) + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})(-\sigma) - \boldsymbol{G}_D\boldsymbol{\sigma} - \Psi\mathbf{\Phi}^{\mathrm{T}}(\mathbf{\Omega} - \hat{\mathbf{\Omega}}) = 0$$

$$\mathbf{M}(\mathbf{q})\dot{\sigma} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\sigma + \boldsymbol{G}_D\boldsymbol{\sigma} + \Psi\mathbf{\Phi}^{\mathrm{T}}\tilde{\mathbf{\Omega}} = 0 \tag{3.5}$$

A Lyapunov function candidate is chosen to examine stability. The Lyapunov function, and its derivative along the system trajectories, are expressed in the form

$$\mathcal{V} = \frac{1}{2}\boldsymbol{\sigma}^{\mathrm{T}}\mathbf{M}(\mathbf{q})\boldsymbol{\sigma} + \tilde{\mathbf{q}}^{\mathrm{T}}\mathbf{\Lambda}\boldsymbol{G}_D\tilde{\mathbf{q}} + \frac{1}{2}\tilde{\mathbf{\Omega}}^{\mathrm{T}}\boldsymbol{G}_{\Omega}\tilde{\mathbf{\Omega}}$$

$$\dot{\mathcal{V}} = \boldsymbol{\sigma}^{\mathrm{T}}\mathbf{M}(\mathbf{q})\dot{\boldsymbol{\sigma}} + 2\tilde{\mathbf{q}}^{\mathrm{T}}\mathbf{\Lambda}\boldsymbol{G}_D\dot{\tilde{\mathbf{q}}} + \dot{\tilde{\mathbf{\Omega}}}^{\mathrm{T}}\boldsymbol{G}_{\Omega}\tilde{\mathbf{\Omega}} + \frac{1}{2}\boldsymbol{\sigma}^{\mathrm{T}}\dot{\mathbf{M}}\boldsymbol{\sigma} \tag{3.6}$$

Equation (3.5) is solved for $\mathbf{M}(\mathbf{q})\dot{\boldsymbol{\sigma}}$ and substituted into the Lyapunov derivative in Equation (3.6).

$$\begin{aligned}\dot{\mathcal{V}} =&\boldsymbol{\sigma}^{\mathrm{T}}(-\mathbf{C}(\mathbf{q},\dot{\mathbf{q}})-\boldsymbol{G}_D\boldsymbol{\sigma}-\boldsymbol{\Psi}\boldsymbol{\Phi}^{\mathrm{T}}\tilde{\boldsymbol{\Omega}})+2\tilde{\mathbf{q}}^{\mathrm{T}}\boldsymbol{\Lambda}\boldsymbol{G}_D\dot{\tilde{\mathbf{q}}}+\dot{\tilde{\boldsymbol{\Omega}}}^{\mathrm{T}}\boldsymbol{G}_\Omega\tilde{\boldsymbol{\Omega}}+\frac{1}{2}\boldsymbol{\sigma}^{\mathrm{T}}\dot{\mathbf{M}}\boldsymbol{\sigma}\\ =&\frac{1}{2}\boldsymbol{\sigma}^{\mathrm{T}}(\dot{\mathbf{M}}(\mathbf{q})-2\mathbf{C}(\mathbf{q},\dot{\mathbf{q}}))\boldsymbol{\sigma}-\boldsymbol{\sigma}^{\mathrm{T}}\boldsymbol{G}_D\boldsymbol{\sigma}-\boldsymbol{\sigma}^{\mathrm{T}}\boldsymbol{\Psi}\boldsymbol{\Phi}^{\mathrm{T}}\tilde{\boldsymbol{\Omega}}+2\tilde{\mathbf{q}}^{\mathrm{T}}\boldsymbol{\Lambda}\boldsymbol{G}_D\dot{\tilde{\mathbf{q}}}+\dot{\tilde{\boldsymbol{\Omega}}}^{\mathrm{T}}\boldsymbol{G}_\Omega\tilde{\boldsymbol{\Omega}}\end{aligned}$$

The first term is equal to zero since the matrix $\dot{\mathbf{M}}(\mathbf{q})-2\mathbf{C}(\mathbf{q},\dot{\mathbf{q}})$ is skew symmetric. The derivative of the Lyapunov function can now be written in the form

$$\begin{aligned}\dot{\mathcal{V}} =&-\dot{\tilde{\mathbf{q}}}^{\mathrm{T}}\boldsymbol{G}_D\dot{\tilde{\mathbf{q}}}-2\tilde{\mathbf{q}}^{\mathrm{T}}\Lambda\boldsymbol{G}_D\dot{\tilde{\mathbf{q}}}-\tilde{\mathbf{q}}^{\mathrm{T}}\Lambda\boldsymbol{G}_D\Lambda\tilde{\mathbf{q}}+(\dot{\tilde{\boldsymbol{\Omega}}}^{\mathrm{T}}\boldsymbol{G}_\Omega-\boldsymbol{\sigma}^{\mathrm{T}}\boldsymbol{\Psi}\boldsymbol{\Phi}^{\mathrm{T}})\tilde{\boldsymbol{\Omega}}+2\tilde{\mathbf{q}}^{\mathrm{T}}\Lambda\boldsymbol{G}_D\dot{\tilde{\mathbf{q}}}\\ =&-\dot{\tilde{\mathbf{q}}}^{\mathrm{T}}\boldsymbol{G}_D\dot{\tilde{\mathbf{q}}}-\tilde{\mathbf{q}}^{\mathrm{T}}\Lambda\boldsymbol{G}_D\Lambda\tilde{\mathbf{q}}+(\dot{\tilde{\boldsymbol{\Omega}}}^{\mathrm{T}}\boldsymbol{G}_\Omega-\boldsymbol{\sigma}^{\mathrm{T}}\boldsymbol{\Psi}\boldsymbol{\Phi}^{\mathrm{T}})\tilde{\boldsymbol{\Omega}}\end{aligned}$$

When we choose the learning update law

$$\begin{aligned}\dot{\tilde{\boldsymbol{\Omega}}}^{\mathrm{T}}\boldsymbol{G}_\Omega-&\boldsymbol{\sigma}^{\mathrm{T}}\boldsymbol{\Psi}\boldsymbol{\Phi}^{\mathrm{T}}=0\\ \dot{\tilde{\boldsymbol{\Omega}}}=&\boldsymbol{G}_\Omega^{-1}\boldsymbol{\Phi}\boldsymbol{\Psi}^{\mathrm{T}}\boldsymbol{\sigma}\\ \dot{\hat{\boldsymbol{\Omega}}}=&-\boldsymbol{G}_\Omega^{-1}\boldsymbol{\Phi}\boldsymbol{\Psi}^{\mathrm{T}}\boldsymbol{\sigma}\end{aligned}$$

The gain matrix $\boldsymbol{G}_\Omega$ is square and symmetric so it is equal to its transpose $\boldsymbol{G}_\Omega^{\mathrm{T}}=\boldsymbol{G}_\Omega$. The terms imply the Lyapunov derivative is negative semi-definite, which means that $\boldsymbol{\sigma}$, $\tilde{\boldsymbol{\Omega}}$, and $\tilde{\mathbf{q}}$ are bounded. We also infer that both $\dot{\tilde{\mathbf{q}}}$ and $\tilde{\mathbf{q}}$ approach zero asymptotically because their terms in $\dot{\mathcal{V}}$ are negative definite.

3.3 Numerical Simulation

3.3.1 Software Architecture

The closed loop simulations on the Adaptive Dynamic Inversion Controller and Adaptive Dissipative Controller were run using MATLAB and Simulink. The software architecture is partitioned into two main sections, Coriolis4DOF_Daedalus and Fundx. Coriolis4DOF_Daedalus is a top level MATLAB script responsible for creating the linear in parameter matrices used as the precursor for the next section. This includes the mass matrix, nonlinear Coriolis matrix, and potential energy gradient. Computing these matrices every iteration in the main simulation would increase run time significantly.

Figure (3.1) shows the entire user defined file hierarchy for the Coriolis4DOF_Daedalus file. The output of this code is the three elements connected to the external data symbol. The first level of functions are DHKineticEnergy, DHPotentialEnergy, and LagrangesCoriolis. DHKineticEnergy inputs the DH parameters, symbolic velocities names, link types, center of mass vectors, masses, and inertia values. The symbolic, scalar kinetic energy value and symbolic mass matrix are the outputs. The output is similar as those listed in Appendix (A.1) but with masses m_1 through m_4 and all the inertia terms are numerically evaluated. The inputs for DHPotentialEnergy are the DH parameters, local mass locations, and masses, and the output is the Potential Energy term V. Lastly, the inputs for LagrangesCoriolis are the kinetic and potential energies and the outputs are the Coriolis matrix and potential energy gradient.

DHKineticEnergy calls three children, DHJacobianVelocity, DHInertiaMatrix, and DHJacobianOmega. The translational velocity Jacobian $\mathbf{J}^v$ and angular velocity Jacobian $\mathbf{J}^\omega$ are generated with the DHJacobianVelocity and DHJacobianOmega functions, respectively. DHInertiaMatrix generates the Inertia Matrix $\boldsymbol{I}$ used in the calculation of the rotational mass matrix component: $\mathbf{M}_i^\omega = \left(\mathbf{J}_i^\omega\right)^{\mathrm{T}} \boldsymbol{I}_i \mathbf{J}_i^\omega$. These three children and DHPotentialEnergy call DHTransforms, which generates the homogeneous transform $\mathbf{H}$ for each link.

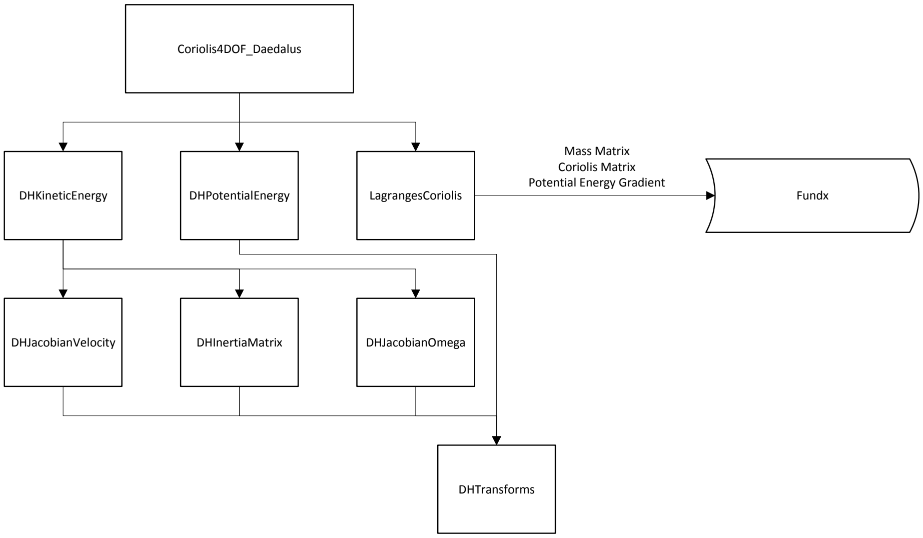

Figure 3.1: Coriolis4DOF code hierarchy

The main execution of the simulation is titled Fundx. The data extracted from Coriolis4DOF_Daedalus is copied as text into Fundx before runtime. Fundx is run as an Interpreted MATLAB Function in Simulink where it is given initial conditions and integrated. The Simulink solver ODE 113 was selected for all cases. See Figure (3.2) for the Fundx Simulink block diagram. The inputs and outputs to Fundx are tabulated in Table (3.1). The inputs are denoted as the column vector x and the output is the column vector derivative dx. The example in the table uses the closed loop equation of motion and update law from the Dissipative controller simulation. In this example, the height of x would be 81 units, where there is one for time, four for $\mathbf{q}$, four for $\dot{\mathbf{q}}$, and 72 for the entire parameter estimation vector $\hat{\boldsymbol{\Omega}}$.

The user defined hierarchy for Fundx is illustrated in Figure (3.3). Fundx has seven immediate children in this hierarchy. The first child is the function DHCalcAlphaBetaWind. This function inputs

Table 3.1: Fundx input and output for dissipative controller simulation

Input, x	Output, dx	Description
t	1	Time counter
$\mathbf{q}$	$\dot{\mathbf{q}}$	Velocity obtained from input
$\dot{\mathbf{q}}$	$\ddot{\mathbf{q}} = \ddot{\mathbf{q}}_d - \boldsymbol{\Lambda}\dot{\tilde{\mathbf{q}}} + \mathbf{M}^{-1}\left(\mathbf{C}\boldsymbol{\sigma} + \boldsymbol{G}_D\boldsymbol{\sigma} + \boldsymbol{\Psi}\boldsymbol{\Phi}^{\mathrm{T}}\tilde{\boldsymbol{\Omega}}\right)$	Closed Loop Equation of Motion
$\hat{\boldsymbol{\Omega}}$	$\dot{\hat{\boldsymbol{\Omega}}} = \boldsymbol{G}_\Omega\boldsymbol{\Phi}\boldsymbol{\Psi}^{\mathrm{T}}\boldsymbol{\sigma}$	Update Law

the generalized coordinates $\mathbf{q}$, constant DH parameters, wind vector v_w, link types, aerodynamic centers, and aerodynamic rotation matrices $\mathbf{R}_0^B$. DHCalcAlphaBetaWind calculates the angle of attack α_i, sideslip β_i, and velocity at aerodynamic center $\mathbf{v}_{a_i}$. The next child is SBRotation, which generates the Rotation Matrix $\mathbf{R}_{B_i}^{S_i}$ using angle of attack and sideslip. PsiRestructure is a child of Fundx which generates the $3\times 2N$ matrix to structure $\boldsymbol{\Psi}$ with the correct dimensions with input N. The child function Hat generates a linear regression basis function at the specific angle of attack, location of the hat function, and mesh spacing. This is used in the construction of $\boldsymbol{\Phi}$. The last three direct subfunctions of Fundx are qdesired, dqdesired, and ddqdesired. These three functions call a second level subfunction qdqddq_desired which returns the desired functions, desired derivatives, and desired double derivatives. Fundx lastly calculates the variables in dx listed in Table (3.1).

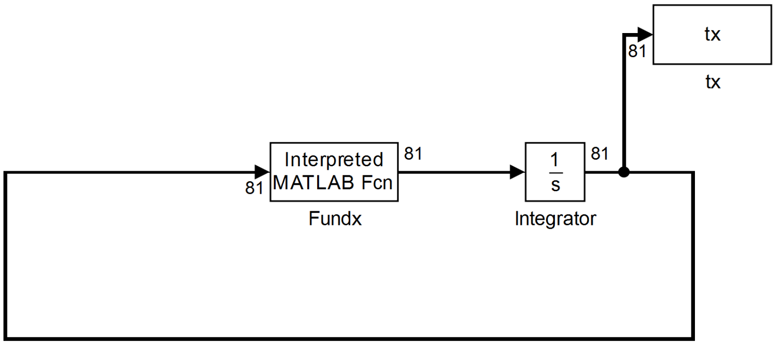

Figure 3.2: Fundx Simulink diagram, with signal dimension displayed

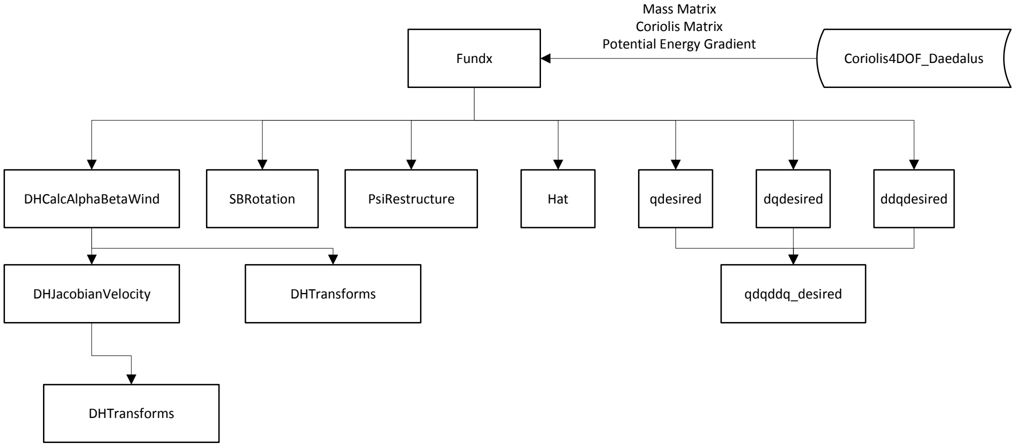

Figure 3.3: Fundx code hierarchy

3.3.2 Dynamic Inversion Controller

Several simulations were performed in Simulink to verify the various control designs. To begin, a simple model was run using the dynamic inversion controller with adaptation. The joint structure is modeled after the Daedalus wing design. The mass matrix was used as described in Section 2.4 and Appendix A.1. The DH parameters are re-listed below. For this simulation, only the second body was assumed to generate lift. All other aerodynamic forces on the robotic system were neglected. The aerodynamic centers for all the bodies are listed along with the aerodynamic rotation matrix $\mathbf{R}_2^{B_2}$.

$$\mathbf{R}_2^{B_2} : \begin{bmatrix} 0 & -1 & 0 \\ 0 & 0 & -1 \\ 1 & 0 & 0 \end{bmatrix}$$

The basis functions used in the adaptive controller are a set of piecewise linear functions. This first simulation used nine functions to approximate $c_L(\alpha_i)$. The range, step size, and true parameters are

Table 3.2: Daedalus DH parameters used in dynamic inversion simulation

Link	$d_i(m)$	$a_i(m)$	$\theta_i(rad)$	$\alpha_i(rad)$
1	0.1	0	θ_1	$\frac{\pi}{2}$
2	0	0.1	θ_2	0
3	0	0.1	θ_3	0
4	0	0.1	θ_4	0

Table 3.3: Daedalus relative aerodynamic centers, $\boldsymbol{r}^i_{a_i}$

Link	x coordinate (m)	y coordinate (m)	z coordinate (m)
1	0	0	0
2	-0.05	0	-0.05
3	0	-0.05	0
4	-0.05	0	0

listed below. The true parameters used in this simulation are based on experimental results from an $S1223$ airfoil [1,33].

$$\text{Angle of Attack Range (degrees):}\begin{bmatrix}-22.5 & 22.5\end{bmatrix}$$

$$\text{Angle of Attack Step Size (degrees):}\begin{bmatrix}5.625\end{bmatrix}$$

$$\boldsymbol{\Omega} = \begin{bmatrix}-0.5 & -0.3 & -0.2 & 0.05 & 1.1 & 1.6 & 1.95 & 2.22 & 1.9\end{bmatrix}$$

Using these true parameters and the basis functions, the true c_L graph is generated and depicted in Figure 3.4. The separate basis functions that compose the c_L curve are plotted in Figure (3.5). The

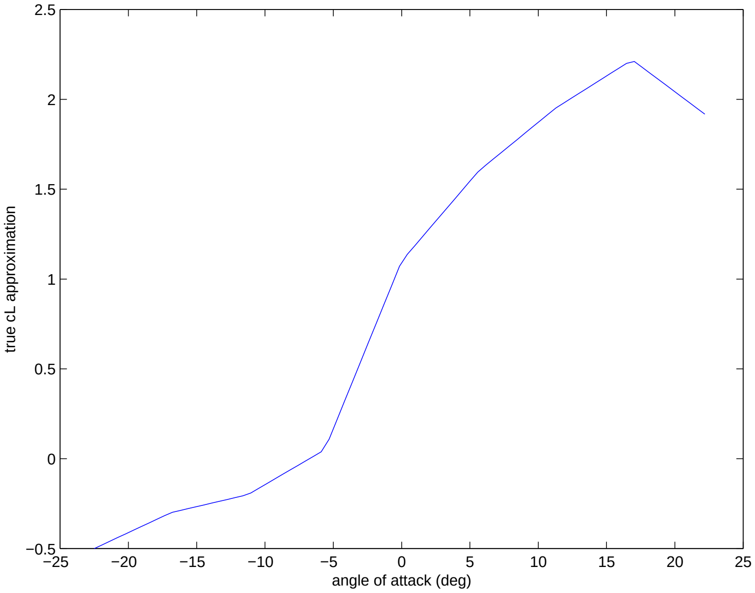

Figure 3.4: True c_L curve used in the numerical studies

wing area for body 2 is selected to be 1m^2 constant wind vector $\mathbf{v}_w = -5\hat{y}_0$ m/s. The gravitational constant is chosen as $g = 9.81$ m/s and the air density is chosen as $\rho = 1.2$ kg/m^3. The controller

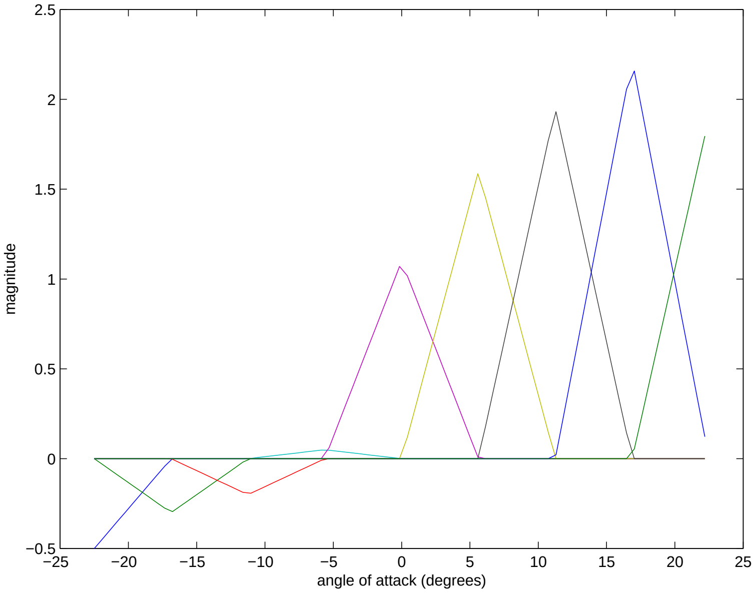

Figure 3.5: True c_L curve decomposed into the nine separate basis functions

gains are set as

$$\boldsymbol{G}_0 : \begin{bmatrix} 50 & 0 & 0 & 0 \\ 0 & 50 & 0 & 0 \\ 0 & 0 & 50 & 0 \\ 0 & 0 & 0 & 50 \end{bmatrix}$$

$$\boldsymbol{G}_1 : \begin{bmatrix} 50 & 0 & 0 & 0 \\ 0 & 50 & 0 & 0 \\ 0 & 0 & 50 & 0 \\ 0 & 0 & 0 & 50 \end{bmatrix}$$

$$\gamma : \begin{bmatrix} 1000 \end{bmatrix}$$

Periodic functions are used to drive θ_2 and θ_3 which represent the flapping motion observed in birds. The desired joint angle histories are chosen as follows:

$$\begin{aligned} \theta_{2,d} &= A\sin(\omega t) \\ \theta_{3,d} &= -A\sin\left(\omega t + \frac{\pi}{2}\right) - A \end{aligned}$$

where A is the amplitude and ω is the flapping frequency. A frequency of $\omega = 3$ Hz and amplitude of $A = 15°$ is reasonable for mimicking birds on the larger end of the range in the Larus genus [3, 4]. An illustrative example of the desired flapping motion is pictured in Figure (3.6). In the two images, the spheres are the spatial locations of the joints through the flapping trajectory. Although the illustration of the upstroke and downstroke appear similar in static images, the angle of the third joint θ_3 is minimal for a high percentage of the downstroke while it is significant for a large percentage of the upstroke. This motion is referred to lead and lag motion in the literature [3].

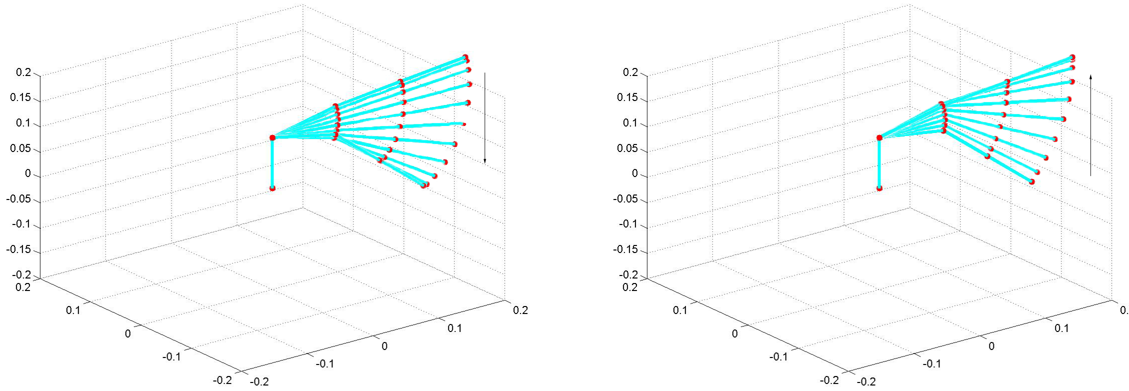

Figure 3.6: Path of the linkages through its desired joint trajectories during the downstroke (left), and upstroke (right)

The initial conditions were chosen with 10% error in the unknown lift parameters. The states and state derivatives were chosen as zero.

$$\hat{\mathbf{\Omega}}_L(0) = \begin{bmatrix} -0.4500 & -0.2700 & -0.1800 & 0.0450 & 0.9900 & 1.4400 & 1.7550 & 1.9980 & 1.7100 \end{bmatrix}$$

$$\mathbf{q}(0) = \begin{bmatrix} 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\dot{\mathbf{q}}(0) = \begin{bmatrix} 0 & 0 & 0 & 0 \end{bmatrix}$$

The results of the numerical experiments are depicted in Figures (3.7) through (3.27). As predicted in the Lyapunov analysis, the adaptive control only guarantees boundedness in the position error $\tilde{\mathbf{q}}$, while the derivative error $\dot{\tilde{\mathbf{q}}}$ is guaranteed to converge towards zero. Figure (3.7) illustrates that the joint positions are indeed bounded during the simulation. As expected, the tracking error rate converges to zero in Figure (3.8), but there is still considerable steady state error oscillating about zero.

One of the most important conclusions from this set of numerical experiments is that the identification of aerodynamic loads is feasible, as shown in Figure (3.9). The estimated lift converges closely to the true value. Lift and estimated lift on body 2 is plotted in Figure (3.9). The error in parameter prediction ($\tilde{\mathbf{\Omega}}$) is depicted in Figure (3.10). It is important to note that some of the parameters converge towards their true values but others are not updated. This phenomenon is well known and documented in the adaptive control literature. For convergence it is required that the parameter is persistently excited. If the θ_2 joint trajectory passes through the entire range listed modeled in the c_L approximation, all of the parameters would be updated. Since only three basis

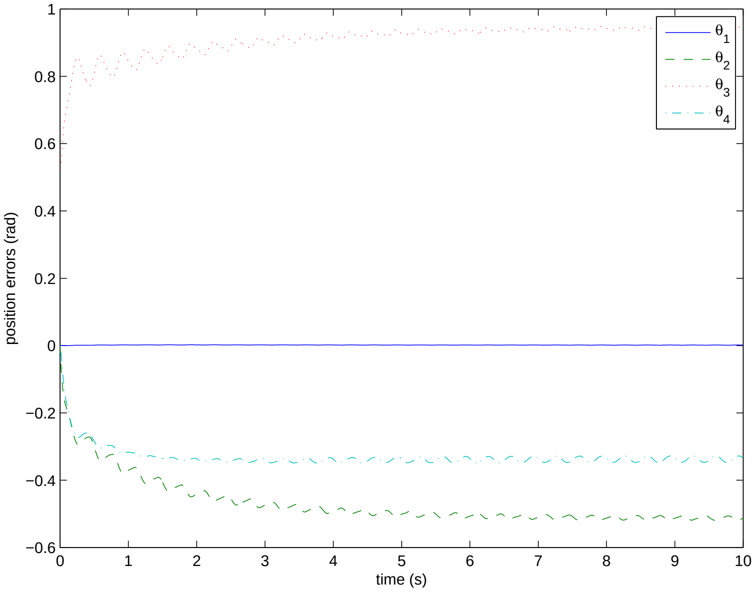

Figure 3.7: Errors in joint positions with lift acting on body 2

function ranges are visited, and for differing amounts of time, the error decreases most rapidly in the range from -5.625° to $+5.625^{\circ}$. Less change is made to parameters ranging -11.25° to 0° and 0° to 11.25°. The angle of attack of body 2 is plotted in Figure (3.11). This simulation provides a good indication of which parameters update and converge during the simulation.

The next numerical experiment considers both drag and lift on bodies 2 through 4. The true values of lift remain unchained from the previous experiment for all of the wing bodies. A separate set of nine parameters were chosen to model the true $c_D(\alpha_i)$ function. The DH parameters, mass variables, aerodynamic center locations, angle of attack step size and range, wind vector, air density and gravitational constant were not changed. The aerodynamic rotation matrices were updated to include all of the bodies.

$$\mathbf{R}_1^{B_1} : \begin{bmatrix} 0 & -1 & 0 \\ 0 & 0 & -1 \\ 1 & 0 & 0 \end{bmatrix} \quad \mathbf{R}_2^{B_2} : \begin{bmatrix} 0 & -1 & 0 \\ 0 & 0 & -1 \\ 1 & 0 & 0 \end{bmatrix}$$

$$\mathbf{R}_3^{B_3} : \begin{bmatrix} 0 & -1 & 0 \\ 0 & 0 & -1 \\ 1 & 0 & 0 \end{bmatrix} \quad \mathbf{R}_4^{B_4} : \begin{bmatrix} 0 & -1 & 0 \\ 0 & 0 & -1 \\ 1 & 0 & 0 \end{bmatrix}$$

The wing areas were updated to account for the wing areas among the last three bodies. The first body was omitted by choosing the area to be zero. The result is a simulation in which lift and drag are generated only the wing sections, where the first body has minimal aerodynamic effects on the system by comparison. Shifting the area towards the root, such as on link 2, will increase lift while adding less significantly to the control effort than on body 3 or 4. The wing areas were reselected

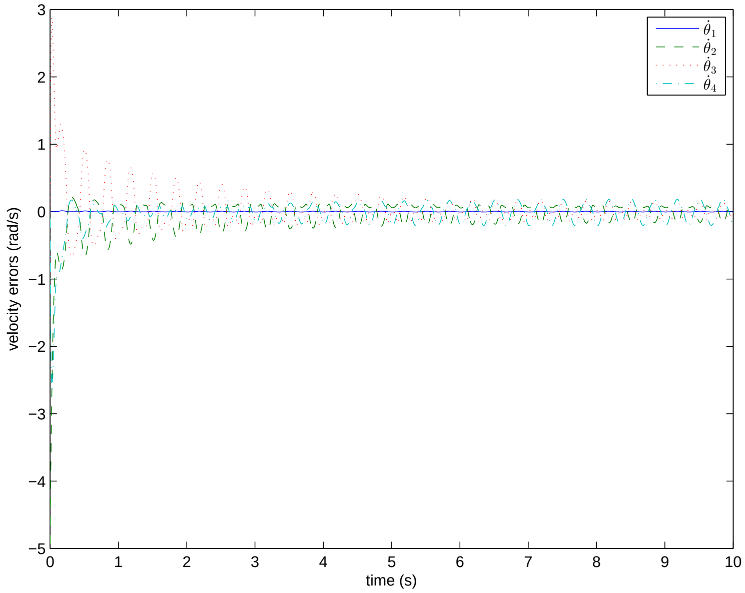

Figure 3.8: Errors in joint velocities

to generate similar magnitudes of lift from the previous experiment.

$$\text{Wing areas}(m^2) : \begin{bmatrix} 0 & 0.25 & 0.1 & 0.05 \end{bmatrix}$$

The true unknown parameters are the coefficients that multiply the nine functions used to represent the drag and lift on each body. The true drag parameter, as in the case of the true lift coefficients, were selected to model experimental data for an S1223 airfoil. Some values for large negative angles of attack were interpolated from the data to choose values for the true unknown parameters and graph which exceeded −6°. The drag function c_D generated by the true parameters appears in Figure (3.12).

$$\mathbf{\Omega} = \begin{bmatrix} \mathbf{\Omega}_{D_1} \\ \mathbf{\Omega}_{L_1} \\ \vdots \\ \mathbf{\Omega}_{D_4} \\ \mathbf{\Omega}_{L_4} \end{bmatrix} = \begin{bmatrix} 0.5 & 0.3 & 0.13 & 0.08 & 0.04 & 0.05 & 0.1 & 0.2 & 0.4 \\ -0.5 & -0.3 & -0.2 & 0.05 & 1.1 & 1.6 & 1.95 & 2.22 & 1.9 \\ 0.5 & 0.3 & 0.13 & 0.08 & 0.04 & 0.05 & 0.1 & 0.2 & 0.4 \\ -0.5 & -0.3 & -0.2 & 0.05 & 1.1 & 1.6 & 1.95 & 2.22 & 1.9 \\ 0.5 & 0.3 & 0.13 & 0.08 & 0.04 & 0.05 & 0.1 & 0.2 & 0.4 \\ -0.5 & -0.3 & -0.2 & 0.05 & 1.1 & 1.6 & 1.95 & 2.22 & 1.9 \\ 0.5 & 0.3 & 0.13 & 0.08 & 0.04 & 0.05 & 0.1 & 0.2 & 0.4 \\ -0.5 & -0.3 & -0.2 & 0.05 & 1.1 & 1.6 & 1.95 & 2.22 & 1.9 \end{bmatrix}$$

The initial conditions were repeated from the first experiment with 10% error in each body for both lift and drag.

$$\hat{\mathbf{\Omega}}_L(0) = \begin{bmatrix} -0.4500 & -0.2700 & -0.1800 & 0.0450 & 0.9900 & 1.4400 & 1.7550 & 1.9980 & 1.7100 \end{bmatrix}$$

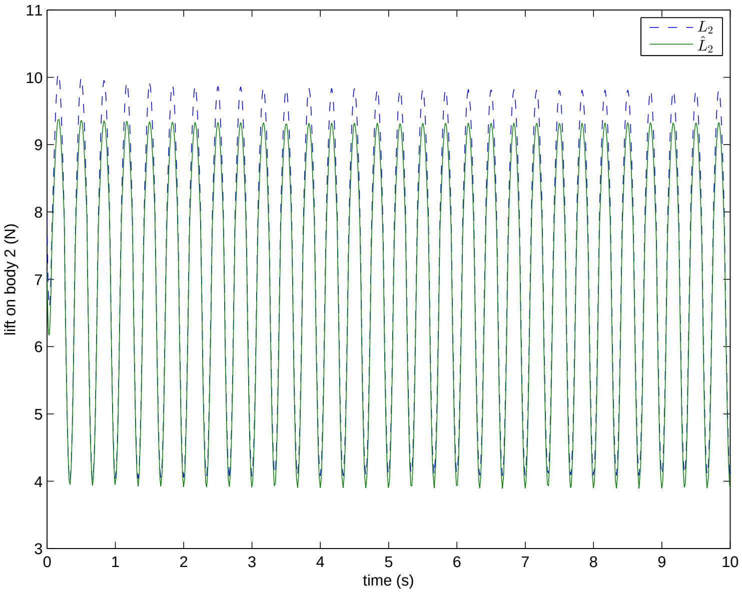

Figure 3.9: Lift on body 2 and estimated lift on body 2

$$\hat{\boldsymbol{\Omega}}_D(0) = \begin{bmatrix} 0.4500 & 0.2700 & 0.1170 & 0.0720 & 0.0360 & 0.0450 & 0.0900 & 0.1800 & 0.3600 \end{bmatrix}$$

The learning gain was tuned lower than the single body simulation, which effectively raises the learning rate. The other control gains remained unchanged.

$$\boldsymbol{G}_0 : \begin{bmatrix} \ddots & & \\ & 50 & \\ & & \ddots \end{bmatrix}$$

$$\boldsymbol{G}_1 : \begin{bmatrix} \ddots & & \\ & 50 & \\ & & \ddots \end{bmatrix}$$

$$\gamma : \begin{bmatrix} 100 \end{bmatrix}$$

After modifying the unknown aerodynamic model to include multiple wing bodies, the results of the numerical experiment are depicted in Figures (3.13) through (3.16). The joint position errors illustrated in Figure (3.13) are qualitatively similar to that in the single body case. The magnitude of errors in joint velocities, shown in Figure (3.14) were also in the same range as in the single body simulation. The new drag forces on the system increased the magnitude of the response of θ_1 and $\dot{\theta}_1$ since these forces caused a torque about the θ_1 axis due to the direction of the wind.

The parameter errors vary between each wing segment as seen in Figure (3.15). Body 1 has no learning during the simulation since its area is set to zero. As expected, the parameters associated

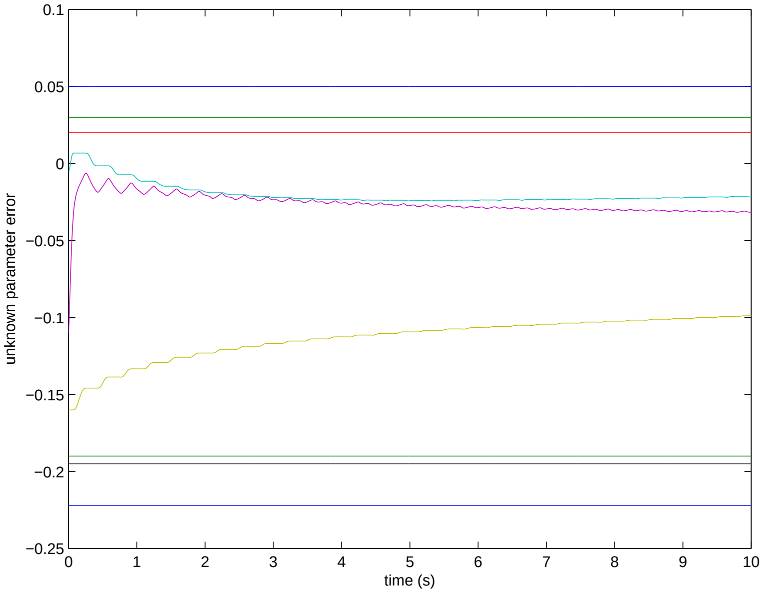

Figure 3.10: Parameter error on body 2 lift

with body 4 exhibit the most learning throughout the trial since its angle of attack varies the most compared to the other links. See Figure (3.16) for the graphs of angle of attack in this simulation. Body 2 undergoes a smaller fluctuation in angle of attack, resulting in smaller aerodynamic forces, and consequently its associated parameters do not update as frequently. Since convergence of the parameter error is not guaranteed by the controller, it is expected that not all parameters will converge towards zero. Some of the parameters affiliated with the highest and lowest angle of attack are not modified by the update law. Specifically, 14 parameters are affected by the update law in the fourth body, 10 parameters in the third body, and six parameters in the second body.

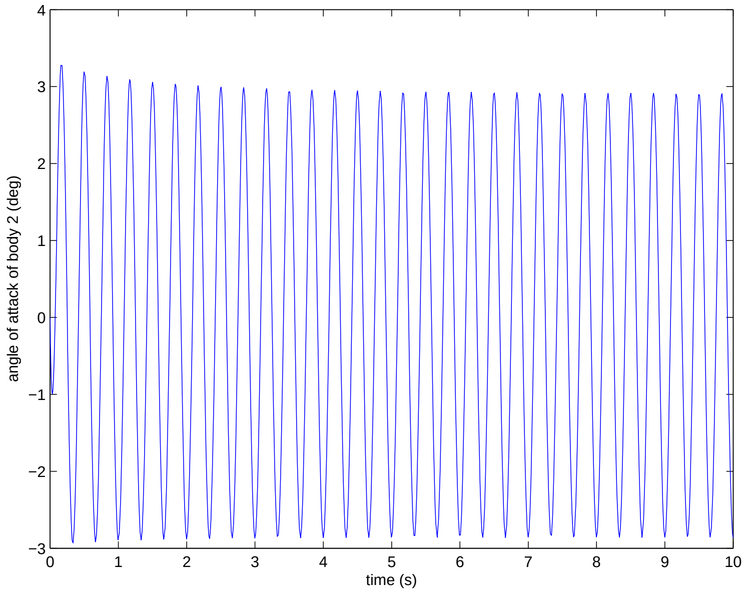

Figure 3.11: Angle of attack of body 2

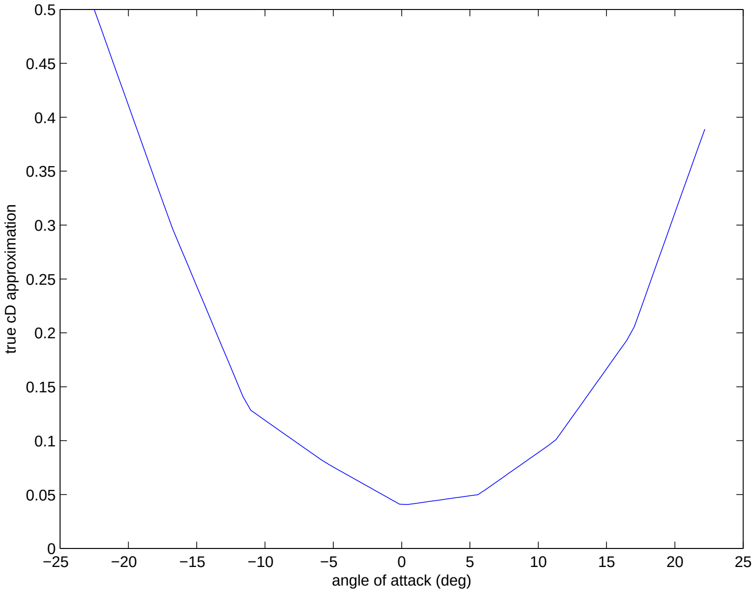

Figure 3.12: True c_D curve using nine unknown functions

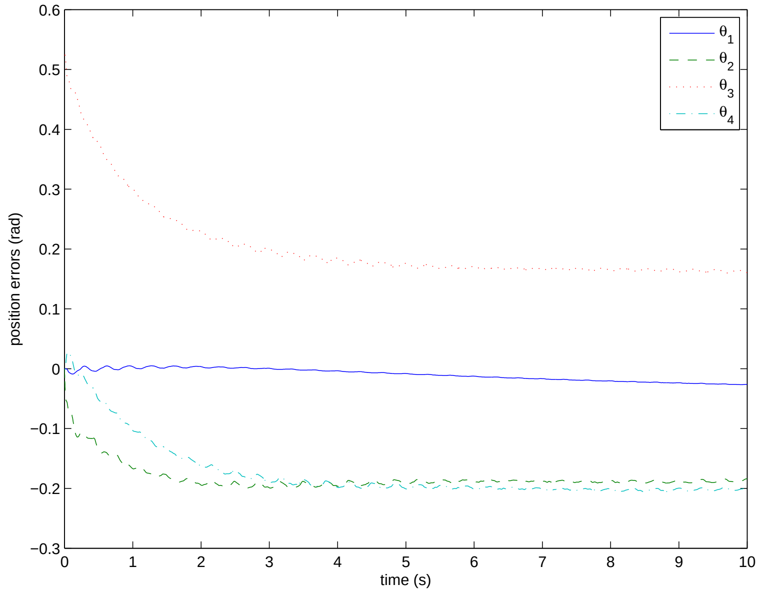

Figure 3.13: Errors in joint positions with drag and lift acting on bodies $2-4$

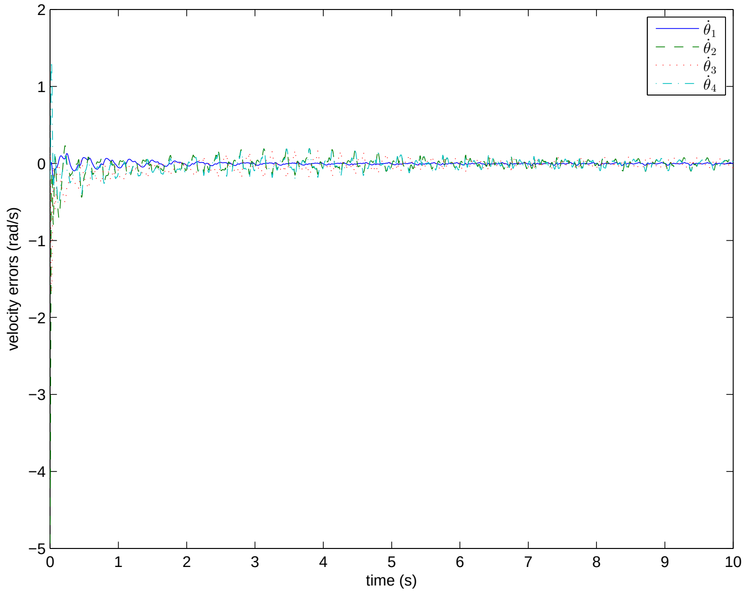

Figure 3.14: Errors in joint velocities with drag and lift acting on bodies $2-4$

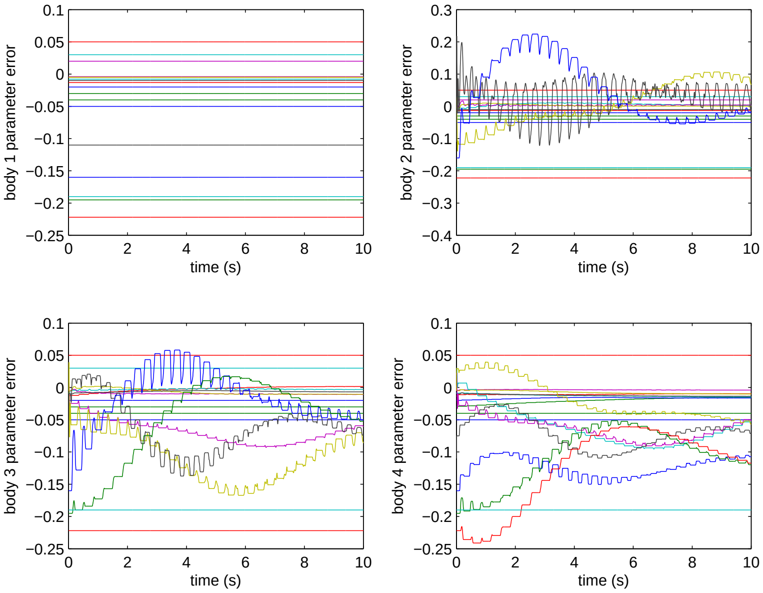

Figure 3.15: Parameter errors on bodies $1-4$

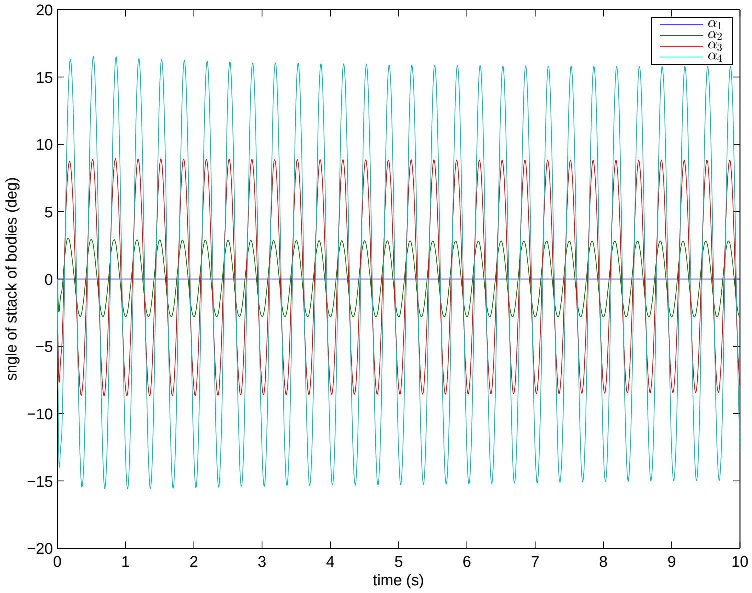

Figure 3.16: Angle of attack for wing sections $1-4$

3.3.3 Dissipative Controller

The dissipative controller discussed in Section (3.2.2) was compared in this section with the adaptive dynamic inversion controller. The numerical experiment is based on the mass and DH parameters for the Daedalus model as listed in the previous section. The aerodynamic rotation matrices, wing areas, and true unknown parameters were not changed from the second dynamic inversion controller. New gains were selected for the controller, including the gains $\boldsymbol{G}_\Omega$, Λ, and $\boldsymbol{G}_D$. The gains were chosen on basis for low settling error with less weight on settling time.

$$\boldsymbol{G}_\Omega : \begin{bmatrix} \ddots & & \\ & 50 & \\ & & \ddots \end{bmatrix}$$

$$\Lambda : \begin{bmatrix} \ddots & & \\ & 1 & \\ & & \ddots \end{bmatrix}$$

$$\boldsymbol{G}_D : \begin{bmatrix} \ddots & & \\ & 1 & \\ & & \ddots \end{bmatrix}$$

The controller shows improved performance in comparison to the dynamic inversion controller. Both the joint position errors, in Figure (3.17), and joint velocity errors, in Figure (3.18), converge quickly. The 5% settling time of all the transients depicted in Figures (3.17) and (3.18) is less than 2.985 seconds. The control input torques associated with these gains is pictured in Figure (3.19). There is a high initial transient in two of the input torques, but soon after there is a steady periodic response. The peak input torques for joints 2 and 3 were higher than the other two input torques, but were not excessive. The maximum control input for τ_3 is 1.5 N·m and τ_2 has a maximum magnitude of 3.852 N·m. For comparison, the Dynamixel RX-24F servomotor used in the Larus prototype has a stall torque of 2.6 N·m. Gearing can be used to increase the torque at the sacrifice of angular velocity.

The update law in the dissipative controller appears to work more smoothly than the previous controller. The unknown parameter errors plotted in Figure (3.20) decrease exponentially toward zero for most of the parameters. Some examples of excited parameters are the third and fourth lift parameter of body 2, in Figures (3.21) and (3.22), respectively. Again, non-convergence of some of the parameters can be understood in terms of persistency of excitation of the angle of attack. The number of parameters updated effectively in the simulation remains unchanged from that of the previous controller, with six parameters in Body 2, 10 parameters in Body 3, and 14 parameters in Body 4. Figure (3.23) shows the similarity of the angle of attack responses to the angle of attack response in the dynamic inversion controller.

The lift function is pictured in Figure (3.24). With the chosen wing sizes and wind velocity the lift

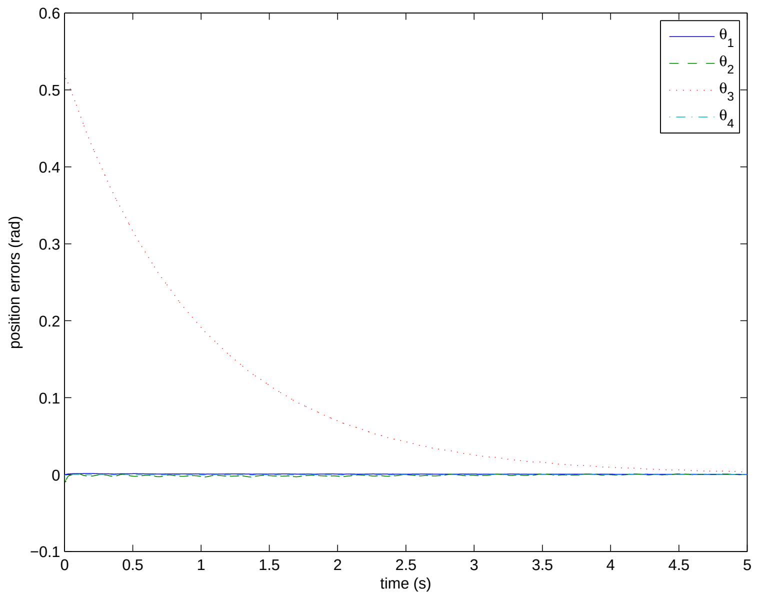

Figure 3.17: Errors in joint positions with drag and lift acting on bodies $2-4$

oscillates between 1.874 and 9.548N in steady state.

The dissipative controller was tested using the Larus model to further verify the controller. None of the experimental parameters, such as wing areas, controller gains, or initial conditions were modified from the Daedalus simulation. The dynamic terms, including the mass matrix, Coriolis matrix, and potential energy gradient were generated as outlined in Section (2.4) and as listed in Appendix Sections (A.4) through (A.6). The position and velocity error histories are graphed in Figures (3.25) and (3.26). The states and their derivatives converged to zero using the Larus model. The addition of the constrained body did not appear to affect the stability or performance of the controller. The control effort as plotted in Figure (3.27) was smaller in magnitude during the steady tracking regime compared to the Daedalus model simulation. The transient peak input torque was larger by comparison.

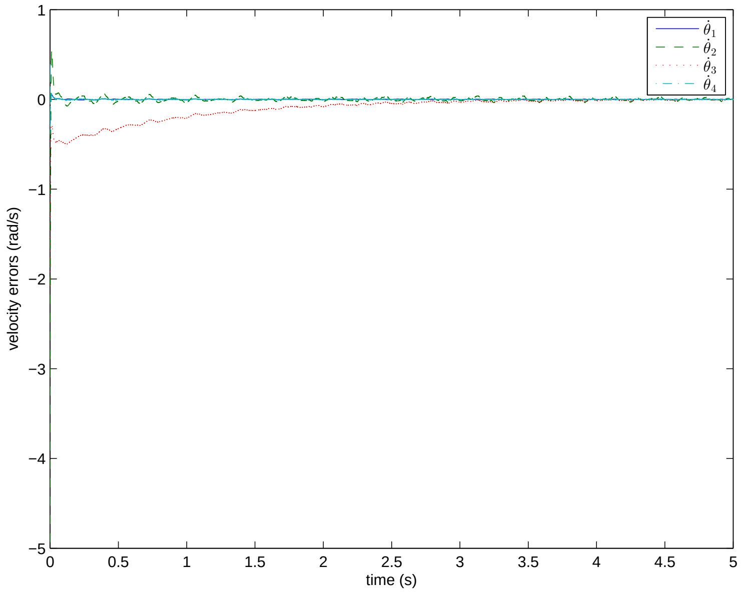

Figure 3.18: Errors in joint velocities with drag and lift acting on bodies $2-4$

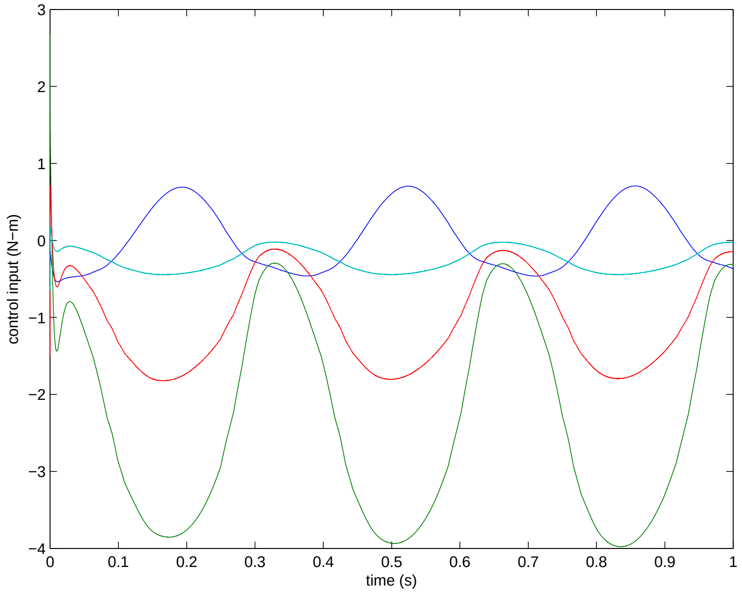

Figure 3.19: Dissipative controller effort on bodies $2-4$

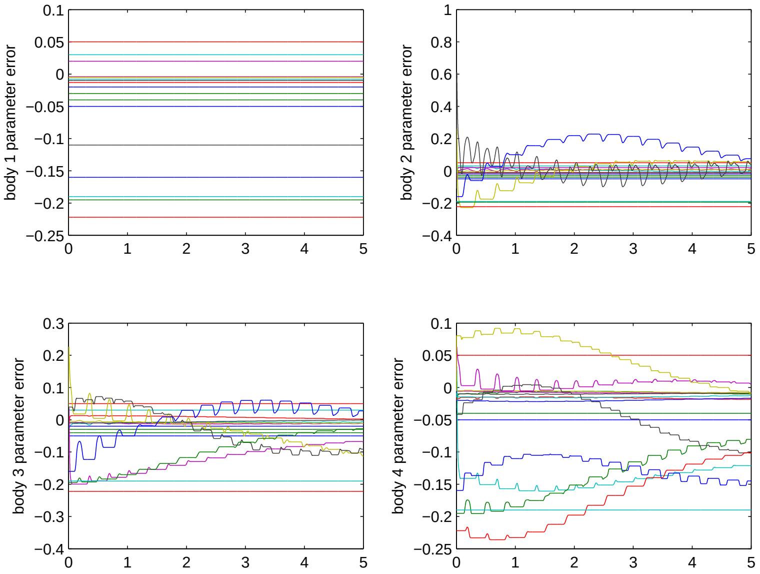

Figure 3.20: Parameter error on bodies $1-4$ in the dissipative controller simulation

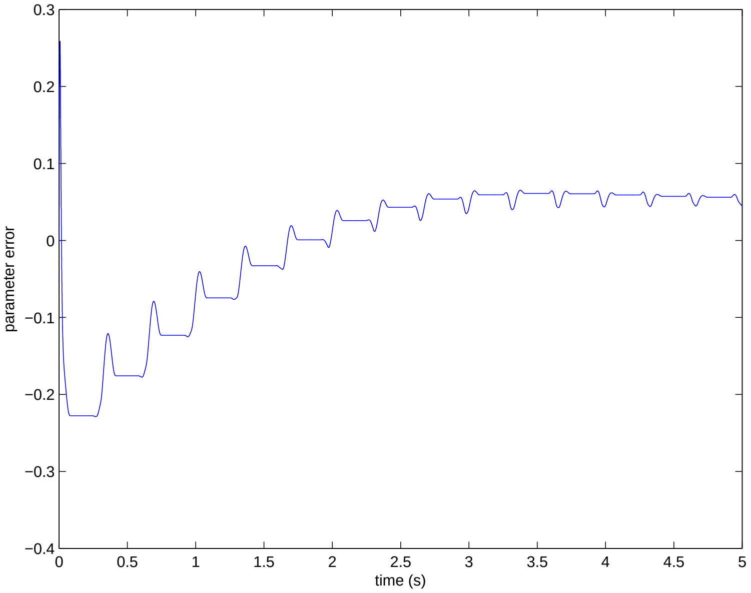

Figure 3.21: Error of the third lift parameter of body 2

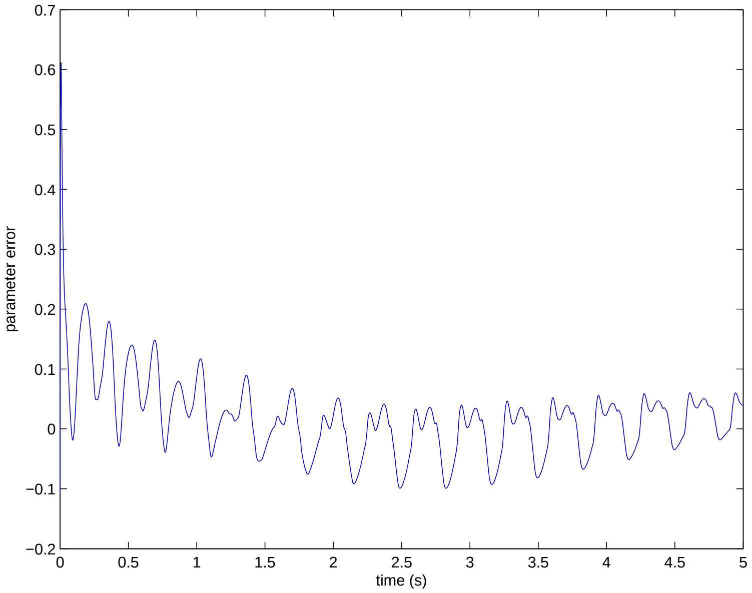

Figure 3.22: Error in the fourth lift parameter of body 2

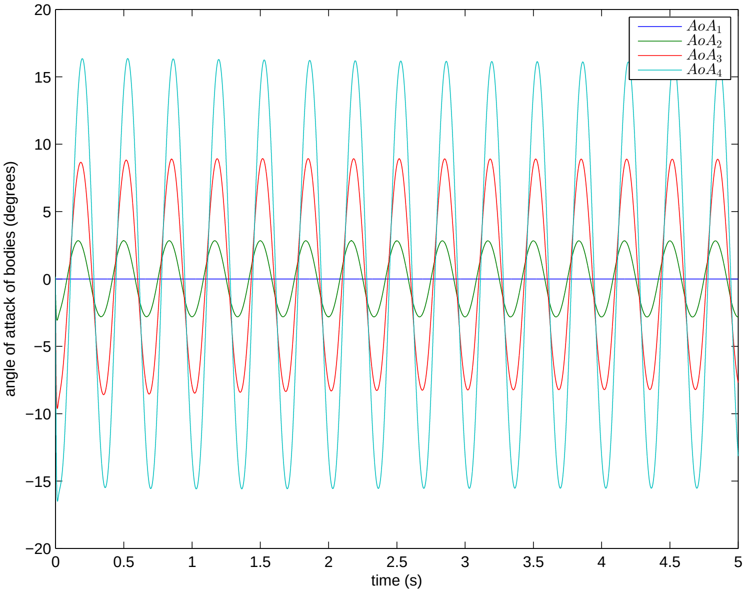

Figure 3.23: Angle of attack response for all bodies in the dissipative controller simulation

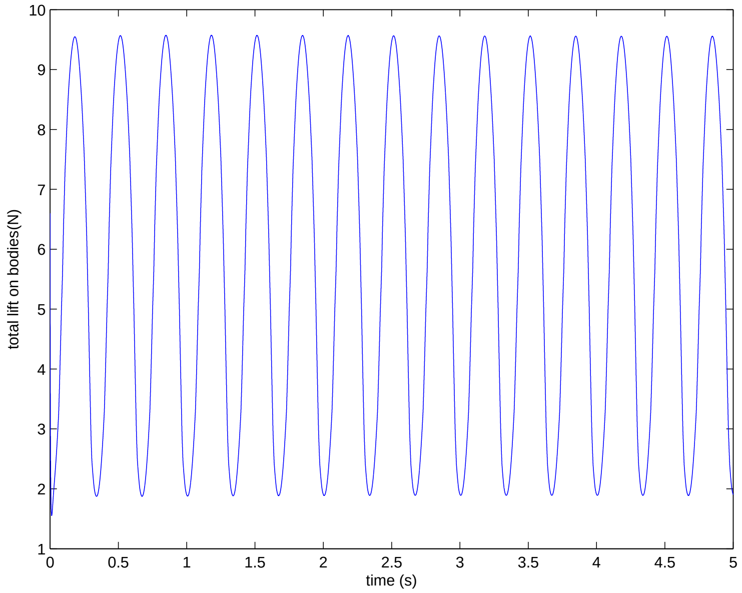

Figure 3.24: Total lift acting on the Daedalus wing model in the dissipative controller simulation

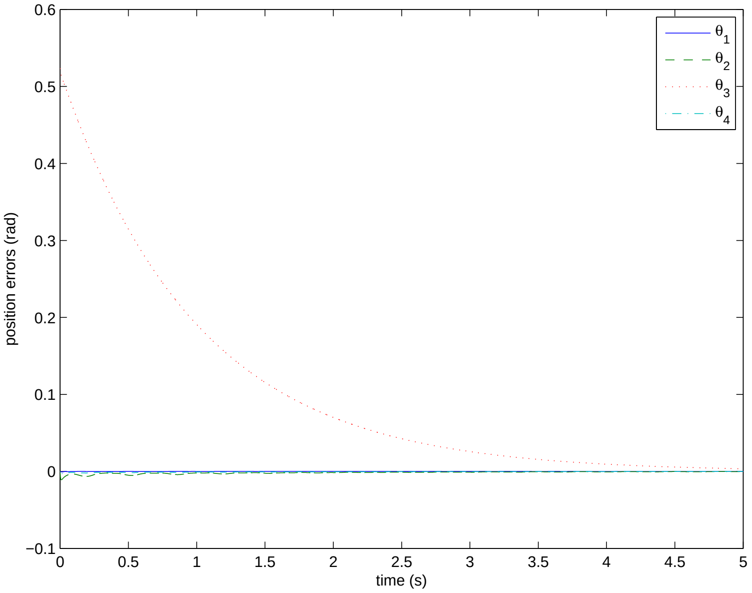

Figure 3.25: Errors in Larus joint positions with drag and lift acting on bodies $2-4$

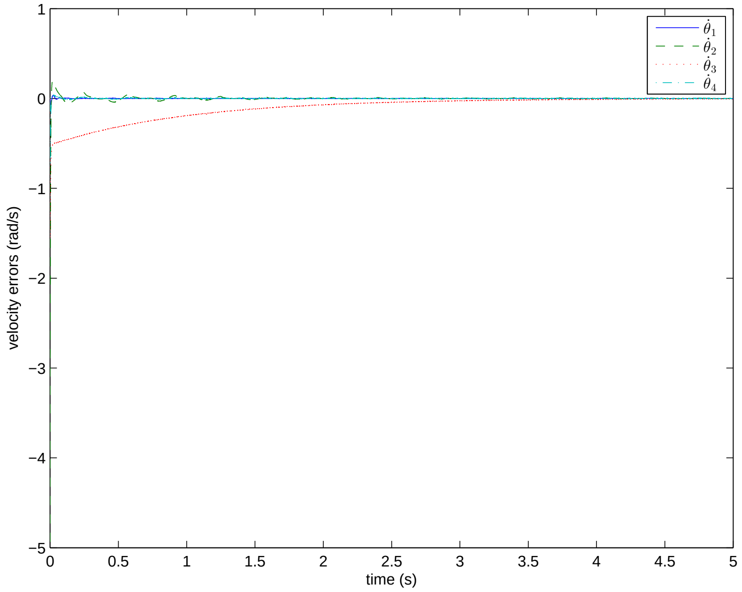

Figure 3.26: Errors in Larus joint velocities with drag and lift acting on bodies $2-4$

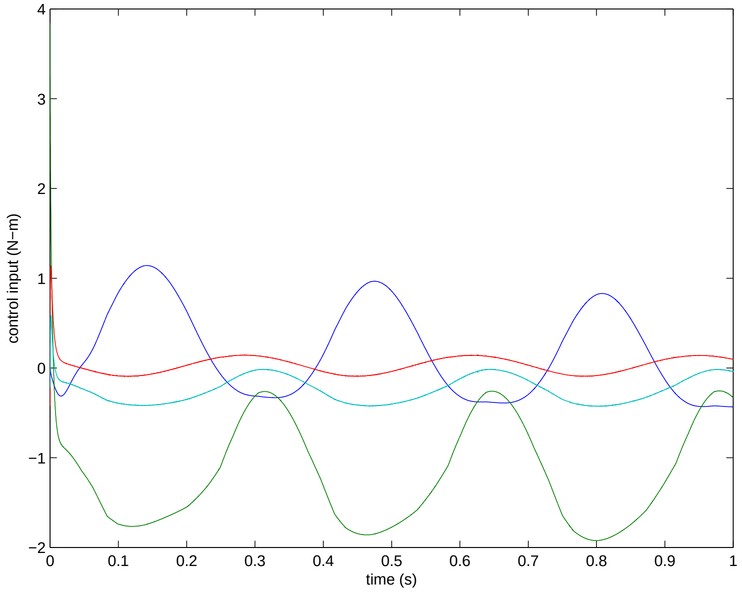

Figure 3.27: Dissipative controller effort on bodies $2-4$ using the Larus dynamic model

Chapter 4

Conclusion

This thesis has studied two classes of flapping wing robot models. One is a kinematic and dynamic model of systems with a fixed base body. The kinematic and dynamic models for the fixed base case are constructed using techniques applicable to kinematic chains. The kinematic description employed for these type of robotic systems are available in standard robotics texts [24–26]. Using the Denavit-Hartenberg convention and analytical mechanics, the mass matrix, Coriolis matrix, and potential energy gradient are generated for the Daedalus and Larus models. These terms are used to build the open loop equations of motion. This thesis introduces a method for calculating the contributions of the aerodynamic lift and drag forces to the robotic equations of motion. The generalized equations of motion are then written in the form

$$\mathbf{M}(\mathbf{q}(t))\ddot{\mathbf{q}}(t) = \mathbf{n}(\mathbf{q}(t), \dot{\mathbf{q}}(t), t) + \mathbf{n}_a(\mathbf{q}(t), \dot{\mathbf{q}}(t), t) + \mathbf{B}(\mathbf{q}(t))\boldsymbol{\tau}(t) \tag{4.1}$$

where $\mathbf{M}$ is the mass matrix, $\mathbf{n}$ is the nonlinear term that arise in the robot dynamics, $\mathbf{n}_a$ is the nonlinear term that appear when the aerodynamics are considered, and $\mathbf{B}$ is the control influence matrix. These equations differ from standard textbook models of robotic systems. Two types of adaptive nonlinear controllers are developed for creating the closed loop equations of motion for the fixed body case. The first is a dynamic inversion controller with adaptation. The second is a dissipative controller with adaptation. A Lyapunov analysis demonstrates that the dynamic inversion controller will ensure that the joint tracking error rates converge towards zero, but the joint position errors are only guaranteed to be bounded. Numerical experiments based on Mathworks Simulink have been carried out to verify the Lyapunov analysis of stability and convergence. The simulation also demonstrates that the adaptive control is able to learn parameters that determine the aerodynamic loads when they are persistently excited.

The dissipative controller was studied to improve performance in comparison to the first controller. Both the joint position and velocity errors are guaranteed to diminish to zero from the Lyapunov analysis. The simulation for this model again verified the Lyapunov analysis. Within the first five seconds of the numerical experiment for this controller, the maximum velocity and position steady state error was below 0.7% of the initial position error and 0.1% of the initial velocity error, respectively. With the gains set to yield a maximum control effort of 3.85 N·m, the position settled in 2.98 seconds to within 5% of the initial error. The development of a new robotic bird prototype will aid in verification of the numerical simulation.

4.1 Future Work

Several novel directions of study are suggested by the research carried out in this thesis. The numerical experiments in the thesis do not include tests of robustness to external disturbances, such as a time varying wind vector. Robustness of the control methodology to such external effects may be an important factor in the performance of the adaptive controller and should be studied in detail. The numerical experiments also did not model any uncertainty in parameters such as masses, lengths, center of mass locations, etc. Only uncertainty in the aerodynamic loads has been studied. The combined effect of uncertainty in model parameters (such as link length, mass, mass center location) and in aerodynamic load should be investigated. Another important and fundamental limitation of the strategy outlined in this paper is the selection of a quasi steady aerodynamic model in the controller. It is well-known that the flow surrounding flapping wings is unsteady, and nonlinear. More realistic flow models should be studied.

The third flapping wing prototype Icarus is proposed to finish completion before May 2013. To combine the efforts of the control design and aircraft design, a new kinematic model needs to be developed based on the new structure featured in Icarus. More importantly, the moving base body dynamic model and control design need to be extended so that they are applicable to a flying vehicle such as Icarus. The nonlinear aerodynamic term needs to be re-evaluated to model the effects of the aerodynamics on the whole flapping flight system, including the base body. This thesis does not include the steps for generating the various Jacobian matrices used to relate the base body to different chain bases or the ground frame.

The control design for the moving base body dynamic model will be a complex problem, one that has not been directly addressed in literature. The adaptive control and learning theory should be developed further to better model the aerodynamics and increase robustness. In the immediate future, the development of an L_1 controller with an “adaptive basis” will be investigated and simulated using the existing dynamic model. Furthermore, the aerodynamic model could be advanced simultaneously with the adaptive controller to result in a highly robust controller. As lightweight and compact on board hardware processing evolves over the next decade or more, complex unsteady aerodynamics models could perhaps be implemented in a real time algorithm using a microcontroller or microprocessing unit.

Bibliography

[1] M.S. Selig and J.J. Guglielmo. High-Lift Low Reynolds Number Airfoil Design. *Journal of Aircraft*, 34, 1997.

[2] T. Liu, K. Kuykendoll, R. Rhew, and S. Jones. Avian Wing Geometry and Kinematics. *AIAA Journal*, 47, 2006.

[3] Cheolheui Han. Investigation of Unsteady Aerodynamic Characteristics of a Seagull Wing in Level Flight. *Journal of Bionic Engineering*, 6, 2009.

[4] K. Scoley. *Developments in Vertibrate Flight, Climbing and Gliding of Mammals and Replites, and The Flapping Flight of Birds*. PhD thesis, University of Bristol, 1982.

[5] K. Aditya and V. Malolan. Investigation of Strouhal Number effect on flapping wing micro air vehicle. *AIAA Paper*, 2007.

[6] D. Mueller, H.A. Bruck, and S.K. Gupta. Measurement of thrust and lift forces associated with drag of compliant flapping wing for micro air vehicles using a new test stand design. *Experimental Mechanics*, 50:725–735, 2010.

[7] K.D. Ellenrieder, K. Parker, and J. Soria. Fluid Mechanics of Flapping Wings. *Experimental Thermal and Fluid Science*, 32, 2008.

[8] X. Deng, L. Schenato, and S.S. Sastry. Flapping Flight for Biomimetic Robotic Insects: Part II-Flight Control Design. *IEEE Transactions on Robotics*, 22:789–802, 2006.

[9] X. Deng, L. Schenato, W. Wu, and S.S. Sastry. Flapping Flight for Biomimetic Robotic Insects: Part I-System Modeling. *IEEE Transactions on Robotics*, 22:789–802, 2006.

[10] S.A. Ansari, R. Zbikowski, and K. Knowles. Aerodynamics Modelling of Insect-Like Flapping Flight for Micro Air Vehicles. *Progress in Aerospace Sciences*, 42:129–172, 2006.

[11] B. Singh and I. Chopra. Insect Based Hover-Capable Flapping Wings for Micro Air Vehicles: Experiments and Analysis. *AIAA Journal*, 46:2115–2135, 2008.

[12] G. Bunget and S. Seelecke. BATMAV: a 2-DOF Bio-Inspired Flapping Flight Platform. *Active and Passive Smart Structures and Integrated Systems*, 7643:1–10, 2010.

[13] K. Miklasz, M. LaBarbera, X. Chen, and J.J. Socha. Effects of Body Cross-sectional Shape on Flying Snake Aerodynamics. *Experimental Mechanics*, 2010.

[14] J.J. Socha. Gliding Flight in Chrysopelea: Turning a Snake into a Wing. *Integrative and Comparative Biology*, 51:969–982, 2011.

[15] J.J. Socha, T. O'Dempsey, and M. LaBarbera. A 3-d kinematic analysis of gliding in flying snake, chrysopelea paradisi. *Journal of Experimental Biology*, 208, 2005.

[16] K. Sibilski, J. Pietrucha, and M. Zlocka. The comparative evaluation of power requirements for fixed, rotary, and flapping wings micro air vehicles. *AIAA Paper*, 2007.

[17] G.K. Taylor and A.L.R. Thomas. Animal Flight Dynamics II. Longitudal Stability in Flapping Flight. *Journal of Theoretical Biology*, 214:351–370, 2002.

[18] B.Y. Leonard. Flapping wing flight dynamic modeling. Master's thesis, Virginia Polytechnic Institute and State University, 2011.

[19] P. Gopalakrishnan and D.K. Tafti. Effect of Rotation Kinematics and Angle of Attack on Flapping Flight. *AIAA Journal*, 47:2505–2519, 2009.

[20] K. Mazaheri and A Ebrahimi. Experiment Investigation on Aerodynamic Performance of a Flapping Wing Vehicle in Forward Flight. *Journal of Fluids and Structures*, 27:586–595, 2011.

[21] A. Muniappa, V. Baskar, and V. Duriyanandhan. Lift and thrust characteristics of flapping wing micro air vehicle (MAV). *AIAA Paper*, 2011.

[22] G.K. Taylor, R.L. Nudds, and A.L.R. Thomas. Flying and swimming animals cruise at a strouhal number tuned for high power efficiency. *Nature*, pages 707–711, 2003.

[23] K. Crandell and B. Tobalske. Aerodynamics of tip-reversal upstroke in a revolving pigeon wing. *The Journal of Experimental Biology*, 214, 2011.

[24] M.W. Spong, S. Hutchinson, and M. Vidyasagar. *Robot Modeling and Control*. John Wiley & Sons, Inc., Hoboken, New Jersey, 2006.

[25] A. J. Kurdila, A. Leonessa, and J. Vignola. *Dynamics and Control of Robotic Systems*. Springer, New York City, New York, 2012.

[26] B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo. *Robotics Modeling, Planning and Control*. Springer-Verlag London Limited, London, UK, 2009.

[27] L. Meirovitch. *Dynamics and Control of Structures*. John Wiley & Sons, Inc., Hoboken, New Jersey, 1990.

[28] F.L. Lewis, C.T. Abdallah, and D.M. Dawson. *Control Of Robot Manipulators*. Maxwell Macmillan Publishing Co., Oxford, UK, 1993.

[29] S. Beutel, A. Dupont, G. Hofmeister, I. Hyun, J. Bayandor, and A. Kurdila. Daedalus: A Fresh Bio-Inspired Aircraft Design Concept. In *50th AIAA Aerospace Sciences Meeting*, Nashville, TN, January 2012.

[30] E. Fisher, A. Bauhofer, C. Beauchene, B. Dress, S. Marshall, C. McCraw, C. Mehrvarzi, C. Ng, J. Simcic, J. Bayandor, and A. Kurdila. A Bio-Inspired Aircraft Design Concept. In *Proceedings of the ASME 2012 Fluids Engineering Division Summer Meeting*, Rio Grande, Puerto Rico, July 2012.

[31] J. Feaster, A. Golding, E. Guerra, K. Lukesch, A. Matta, T. Suh, D. Volk, J. Witte, J. Bayandor, and A. Kurdila. On Design and Analysis of Larus Genus Inspired Biomimetic Aircraft. Blacksburg, VA, May 2013.

[32] J.E. Slotine and W. Li. *Applied Nonlinear Control*. Prentice Hall, Englewood Cliffs, New Jersey, 1991.

[33] R. Ma and P. Liu. Numerical Simulation of Low-Reynolds-Number and High-Lift Airfoil s1223. *Proceedings of the World Congress on Engineering*, 2, 2009.

Appendix A

Dynamic Model Examples

A.1 Daedalus Mass Matrix

$m(1,1) = [I_{1,yy} + I_{3,xy}\sin(2\theta_2 + 2\theta_3) + I_{4,yy}\cos(\theta_2 + \theta_3 + \theta_4)^2 + I_{4,xx}\sin(\theta_2 + \theta_3 + \theta_4)^2 + (m_3(\cos(\theta_2 + \theta_3) + 2\cos(\theta_2))^2)/400 + I_{3,yy}\cos(\theta_2 + \theta_3)^2 + m_2(\cos(2\theta_2)/800 + 1/800) + I_{3,xx}\sin(\theta_2 + \theta_3)^2 + I_{2,yy}\cos(\theta_2)^2 + (m_4(\cos(\theta_2 + \theta_3 + \theta_4) + 2\cos(\theta_2 + \theta_3) + 2\cos(\theta_2))^2)/400 + I_{2,xy}\sin(2\theta_2) + I_{2,xx}\sin(\theta_2)^2 + I_{4,xy}\sin(2\theta_2 + 2\theta_3 + 2\theta_4)]$

$m(1,2) = [I_{3,yz}\cos(\theta_2 + \theta_3) + I_{3,xz}\sin(\theta_2 + \theta_3) + I_{2,yz}\cos(\theta_2) + I_{2,xz}\sin(\theta_2) + I_{4,yz}\cos(\theta_2 + \theta_3 + \theta_4) + I_{4,xz}\sin(\theta_2 + \theta_3 + \theta_4)]$

$m(1,3) = [I_{3,yz}\cos(\theta_2 + \theta_3) + I_{3,xz}\sin(\theta_2 + \theta_3) + I_{4,yz}\cos(\theta_2 + \theta_3 + \theta_4) + I_{4,xz}\sin(\theta_2 + \theta_3 + \theta_4)]$

$m(1,4) = [I_{4,yz}\cos(\theta_2 + \theta_3 + \theta_4) + I_{4,xz}\sin(\theta_2 + \theta_3 + \theta_4)]$

$m(2,1) = [I_{3,yz}\cos(\theta_2 + \theta_3) + I_{3,xz}\sin(\theta_2 + \theta_3) + I_{2,yz}\cos(\theta_2) + I_{2,xz}\sin(\theta_2) + I_{4,yz}\cos(\theta_2 + \theta_3 + \theta_4) + I_{4,xz}\sin(\theta_2 + \theta_3 + \theta_4)]$

$m(2,2) = [I_{1,zz} + I_{3,zz} + I_{4,zz} + m_2/400 + m_3/80 + (9m_4)/400 + (m_4\cos(\theta_3 + \theta_4))/100 + (m_3\cos(\theta_3))/100 + (m_4\cos(\theta_3))/50 + (m_4\cos(\theta_4))/100]$

$m(2,3) = [I_{3,zz} + I_{4,zz} + m_3/400 + m_4/80 + (m_4\cos(\theta_3 + \theta_4))/200 + (m_3\cos(\theta_3))/200 + (m_4\cos(\theta_3))/100 + (m_4\cos(\theta_4))/100]$

$m(2,4) = [I_{4,zz} + m_4/400 + (m_4\cos(\theta_3 + \theta_4))/200 + (m_4\cos(\theta_4))/200]$

$m(3,1) = [I_{3,yz}\cos(\theta_2 + \theta_3) + I_{3,xz}\sin(\theta_2 + \theta_3) + I_{4,yz}\cos(\theta_2 + \theta_3 + \theta_4) + I_{4,xz}\sin(\theta_2 + \theta_3 + \theta_4)]$

$$m(3,2) = [I_{3,zz} + I_{4,zz} + m_3/400 + m_4/80 + (m_4 \cos(\theta_3 + \theta_4))/200 + (m_3 \cos(\theta_3))/200 + (m_4 \cos(\theta_3))/100 + (m_4 \cos(\theta_4))/100]$$

$$m(3,3) = [I_{3,zz} + I_{4,zz} + m_3/400 + m_4/80 + (m_4 \cos(\theta_4))/100]$$

$$m(3,4) = [I_{4,zz} + m_4/400 + (m_4 \cos(\theta_4))/200]$$

$$m(4,1) = [I_{4,yz} \cos(\theta_2 + \theta_3 + \theta_4) + I_{4,xz} \sin(\theta_2 + \theta_3 + \theta_4)]$$

$$m(4,2) = [I_{4,zz} + m_4/400 + (m_4 \cos(\theta_3 + \theta_4))/200 + (m_4 \cos(\theta_4))/200]$$

$$m(4,3) = [I_{4,zz} + m_4/400 + (m_4 \cos(\theta_4))/200]$$

$$m(4,4) = [I_{4,zz} + m_4/400]$$

A.2 Daedalus Nonlinear Coriolis Matrix

$$C(1,1) = [\dot{\theta}_2(I_{3,xy} \cos(2\theta_2 + 2\theta_3) - (m_2 \sin(2\theta_2))/800 + (I_{3,xx} \sin(2\theta_2 + 2\theta_3))/2 - (I_{3,yy} \sin(2\theta_2 + 2\theta_3))/2 + I_{2,xy} \cos(2\theta_2) + I_{4,xy} \cos(2\theta_2 + 2\theta_3 + 2\theta_4) + (I_{2,xx} \sin(2\theta_2))/2 - (I_{2,yy} \sin(2\theta_2))/2 + (I_{4,xx} \sin(2\theta_2 + 2\theta_3 + 2\theta_4))/2 - (I_{4,yy} \sin(2\theta_2 + 2\theta_3 + 2\theta_4))/2 - (m_3(\cos(\theta_2 + \theta_3) + 2\cos(\theta_2))(\sin(\theta_2 + \theta_3) + 2\sin(\theta_2)))/400 - (m_4(\cos(\theta_2 + \theta_3 + \theta_4) + 2\cos(\theta_2 + \theta_3) + 2\cos(\theta_2))(\sin(\theta_2 + \theta_3 + \theta_4) + 2\sin(\theta_2 + \theta_3) + 2\sin(\theta_2)))/400) + \dot{\theta}_3(I_{3,xy} \cos(2\theta_2 + 2\theta_3) + (I_{3,xx} \sin(2\theta_2 + 2\theta_3))/2 - (I_{3,yy} \sin(2\theta_2 + 2\theta_3))/2 + I_{4,xy} \cos(2\theta_2 + 2\theta_3 + 2\theta_4) + (I_{4,xx} \sin(2\theta_2 + 2\theta_3 + 2\theta_4))/2 - (I_{4,yy} \sin(2\theta_2 + 2\theta_3 + 2\theta_4))/2 - (m_4(\sin(\theta_2 + \theta_3 + \theta_4) + 2\sin(\theta_2 + \theta_3))(\cos(\theta_2 + \theta_3 + \theta_4) + 2\cos(\theta_2 + \theta_3) + 2\cos(\theta_2)))/400 - (m_3 \sin(\theta_2 + \theta_3)(\cos(\theta_2 + \theta_3) + 2\cos(\theta_2)))/400) + \dot{\theta}_4(I_{4,xy} \cos(2\theta_2 + 2\theta_3 + 2\theta_4) + (I_{4,xx} \sin(2\theta_2 + 2\theta_3 + 2\theta_4))/2 - (I_{4,yy} \sin(2\theta_2 + 2\theta_3 + 2\theta_4))/2 - (m_4 \sin(\theta_2 + \theta_3 + \theta_4)(\cos(\theta_2 + \theta_3 + \theta_4) + 2\cos(\theta_2 + \theta_3) + 2\cos(\theta_2)))/400)]$$

$$C(1,2) = [\dot{\theta}_1(I_{3,xy} \cos(2\theta_2 + 2\theta_3) - (m_2 \sin(2\theta_2))/800 + (I_{3,xx} \sin(2\theta_2 + 2\theta_3))/2 - (I_{3,yy} \sin(2\theta_2 + 2\theta_3))/2 + I_{2,xy} \cos(2\theta_2) + I_{4,xy} \cos(2\theta_2 + 2\theta_3 + 2\theta_4) + (I_{2,xx} \sin(2\theta_2))/2 - (I_{2,yy} \sin(2\theta_2))/2 + (I_{4,xx} \sin(2\theta_2 + 2\theta_3 + 2\theta_4))/2 - (I_{4,yy} \sin(2\theta_2 + 2\theta_3 + 2\theta_4))/2 - (m_3(\cos(\theta_2 + \theta_3) + 2\cos(\theta_2))(\sin(\theta_2 + \theta_3) + 2\sin(\theta_2)))/400 - (m_4(\cos(\theta_2 + \theta_3 + \theta_4) + 2\cos(\theta_2 + \theta_3) + 2\cos(\theta_2))(\sin(\theta_2 + \theta_3 + \theta_4) + 2\sin(\theta_2 + \theta_3) + 2\sin(\theta_2)))/400) + \dot{\theta}_3(I_{3,xz} \cos(\theta_2 + \theta_3) - I_{3,yz} \sin(\theta_2 + \theta_3) + I_{4,xz} \cos(\theta_2 + \theta_3 + \theta_4) - I_{4,yz} \sin(\theta_2 + \theta_3 + \theta_4)) + \dot{\theta}_2(I_{3,xz} \cos(\theta_2 + \theta_3) - I_{3,yz} \sin(\theta_2 + \theta_3) + I_{2,xz} \cos(\theta_2) - I_{2,yz} \sin(\theta_2) + I_{4,xz} \cos(\theta_2 + \theta_3 + \theta_4) - I_{4,yz} \sin(\theta_2 + \theta_3 + \theta_4)) + \dot{\theta}_4(I_{4,xz} \cos(\theta_2 + \theta_3 + \theta_4) - I_{4,yz} \sin(\theta_2 + \theta_3 + \theta_4))]$$

$$C(1,3) = [\dot{\theta}_1(I_{3,xy} \cos(2\theta_2 + 2\theta_3) + (I_{3,xx} \sin(2\theta_2 + 2\theta_3))/2 - (I_{3,yy} \sin(2\theta_2 + 2\theta_3))/2 + I_{4,xy} \cos(2\theta_2 + 2\theta_3 + 2\theta_4) + (I_{4,xx} \sin(2\theta_2 + 2\theta_3 + 2\theta_4))/2 - (I_{4,yy} \sin(2\theta_2 + 2\theta_3 + 2\theta_4))/2 - (m_4(\sin(\theta_2 + \theta_3 + \theta_4) + 2\sin(\theta_2 + \theta_3))(\cos(\theta_2 + \theta_3 + \theta_4) + 2\cos(\theta_2 + \theta_3) + 2\cos(\theta_2)))/400 - (m_3 \sin(\theta_2 + \theta_3)(\cos(\theta_2 + \theta_3) +$$

$2\cos(\theta_2)))/400) + \dot{\theta}_2(I_{3,xz}\cos(\theta_2+\theta_3) - I_{3,yz}\sin(\theta_2+\theta_3) + I_{4,xz}\cos(\theta_2+\theta_3+\theta_4) - I_{4,yz}\sin(\theta_2+\theta_3+\theta_4)) + \dot{\theta}_3(I_{3,xz}\cos(\theta_2+\theta_3) - I_{3,yz}\sin(\theta_2+\theta_3) + I_{4,xz}\cos(\theta_2+\theta_3+\theta_4) - I_{4,yz}\sin(\theta_2+\theta_3+\theta_4)) + \dot{\theta}_4(I_{4,xz}\cos(\theta_2+\theta_3+\theta_4) - I_{4,yz}\sin(\theta_2+\theta_3+\theta_4))]$

$C(1,4) = [\dot{\theta}_1(I_{4,xy}\cos(2\theta_2+2\theta_3+2\theta_4) + (I_{4,xx}\sin(2\theta_2+2\theta_3+2\theta_4))/2 - (I_{4,yy}\sin(2\theta_2+2\theta_3+2\theta_4))/2 - (m_4\sin(\theta_2+\theta_3+\theta_4)(\cos(\theta_2+\theta_3+\theta_4) + 2\cos(\theta_2+\theta_3) + 2\cos(\theta_2)))/400) + \dot{\theta}_2(I_{4,xz}\cos(\theta_2+\theta_3+\theta_4) - I_{4,yz}\sin(\theta_2+\theta_3+\theta_4)) + \dot{\theta}_3(I_{4,xz}\cos(\theta_2+\theta_3+\theta_4) - I_{4,yz}\sin(\theta_2+\theta_3+\theta_4)) + \dot{\theta}_4(I_{4,xz}\cos(\theta_2+\theta_3+\theta_4) - I_{4,yz}\sin(\theta_2+\theta_3+\theta_4))]$

$C(2,1) = [(I_{2,yy}\dot{\theta}_1\sin(2\theta_2))/2 - I_{4,xy}\dot{\theta}_1\cos(2\theta_2+2\theta_3+2\theta_4) - (I_{2,xx}\dot{\theta}_1\sin(2\theta_2))/2 - I_{2,xy}\dot{\theta}_1\cos(2\theta_2) - (I_{4,xx}\dot{\theta}_1\sin(2\theta_2+2\theta_3+2\theta_4))/2 + (I_{4,yy}\dot{\theta}_1\sin(2\theta_2+2\theta_3+2\theta_4))/2 - I_{3,xy}\dot{\theta}_1\cos(2\theta_2+2\theta_3) + (\dot{\theta}_1 m_2\sin(2\theta_2))/800 - (I_{3,xx}\dot{\theta}_1\sin(2\theta_2+2\theta_3))/2 + (I_{3,yy}\dot{\theta}_1\sin(2\theta_2+2\theta_3))/2 + (\dot{\theta}_1 m_3(\cos(\theta_2+\theta_3) + s2\cos(\theta_2))(\sin(\theta_2+\theta_3) + 2\sin(\theta_2)))/400 + (\dot{\theta}_1 m_4(\cos(\theta_2+\theta_3+\theta_4) + 2\cos(\theta_2+\theta_3) + 2\cos(\theta_2))(\sin(\theta_2+\theta_3+\theta_4) + 2\sin(\theta_2+\theta_3) + 2\sin(\theta_2)))/400]$

$C(2,2) = [-\dot{\theta}_3((m_4\sin(\theta_3+\theta_4))/200 + (m_3\sin(\theta_3))/200 + (m_4\sin(\theta_3))/100) - \dot{\theta}_4((m_4\sin(\theta_3+\theta_4))/200 + (m_4\sin(\theta_4))/200)]$

$C(2,3) = [-\dot{\theta}_2((m_4\sin(\theta_3+\theta_4))/200 + (m_3\sin(\theta_3))/200 + (m_4\sin(\theta_3))/100) - \dot{\theta}_3((m_4\sin(\theta_3+\theta_4))/200 + (m_3\sin(\theta_3))/200 + (m_4\sin(\theta_3))/100) - \dot{\theta}_4((m_4\sin(\theta_3+\theta_4))/200 + (m_4\sin(\theta_4))/200)]$

$C(2,4) = [-(m_4(\sin(\theta_3+\theta_4) + \sin(\theta_4))(\dot{\theta}_2 + \dot{\theta}_3 + \dot{\theta}_4))/200]$

$C(3,1) = [(I_{4,yy}\dot{\theta}_1\sin(2\theta_2+2\theta_3+2\theta_4))/2 - (I_{4,xx}\dot{\theta}_1\sin(2\theta_2+2\theta_3+2\theta_4))/2 - I_{4,xy}\dot{\theta}_1\cos(2\theta_2+2\theta_3+2\theta_4) - I_{3,xy}\dot{\theta}_1\cos(2\theta_2+2\theta_3) - (I_{3,xx}\dot{\theta}_1\sin(2\theta_2+2\theta_3))/2 + (I_{3,yy}\dot{\theta}_1\sin(2\theta_2+2\theta_3))/2 + (\dot{\theta}_1 m_3\sin(\theta_2+\theta_3)(\cos(\theta_2+\theta_3) + 2\cos(\theta_2)))/400 + (\dot{\theta}_1 m_4(\sin(\theta_2+\theta_3+\theta_4) + 2\sin(\theta_2+\theta_3))(\cos(\theta_2+\theta_3+\theta_4) + 2\cos(\theta_2+\theta_3) + 2\cos(\theta_2)))/400]$

$C(3,2) = [\dot{\theta}_2((m_4\sin(\theta_3+\theta_4))/200 + (m_3\sin(\theta_3))/200 + (m_4\sin(\theta_3))/100) - (\dot{\theta}_4 m_4\sin(\theta_4))/200]$

$C(3,3) = [-(\dot{\theta}_4 m_4\sin(\theta_4))/200]$

$C(3,4) = [-(m_4\sin(\theta_4)(\dot{\theta}_2 + \dot{\theta}_3 + \dot{\theta}_4))/200]$

$C(4,1) = [(I_{4,yy}\dot{\theta}_1\sin(2\theta_2+2\theta_3+2\theta_4))/2 - (I_{4,xx}\dot{\theta}_1\sin(2\theta_2+2\theta_3+2\theta_4))/2 - I_{4,xy}\dot{\theta}_1\cos(2\theta_2+2\theta_3+2\theta_4) + (\dot{\theta}_1 m_4\sin(\theta_2+\theta_3+\theta_4)(\cos(\theta_2+\theta_3+\theta_4) + 2\cos(\theta_2+\theta_3) + 2\cos(\theta_2)))/400]$

$C(4,2) = [\dot{\theta}_2((m_4\sin(\theta_3+\theta_4))/200 + (m_4\sin(\theta_4))/200) + (\dot{\theta}_3 m_4\sin(\theta_4))/200]$

$C(4,3) = [(m_4\sin(\theta_4)(\dot{\theta}_2 + \dot{\theta}_3))/200]$

$C(4,4) = [0]$

A.3 Daedalus Potential Energy Gradient

$$\frac{\partial V}{\partial \mathbf{q}}(1) = [0]$$

$$\frac{\partial V}{\partial \mathbf{q}}(2) = [g \cdot m_4(\cos(\theta_2)^2/5 - \sin(\theta_2)^2/5 + (\cos(\theta_4)(\cos(\theta_4)(\cos(\theta_2)\cos(\theta_3) - \sin(\theta_2)\sin(\theta_3)) - \sin(\theta_4)(\cos(\theta_2)\sin(\theta_3) + \cos(\theta_3)\sin(\theta_2))))/10 - (\sin(\theta_4)(\cos(\theta_4)(\cos(\theta_2)\sin(\theta_3) + \cos(\theta_3)\sin(\theta_2)) + \sin(\theta_4)(\cos(\theta_2)\cos(\theta_3) - \sin(\theta_2)\sin(\theta_3))))/10 + (\cos(\theta_3)(\cos(\theta_2)\cos(\theta_3) - \sin(\theta_2)\sin(\theta_3)))/10 - (\cos(\theta_4)(\cos(\theta_2)\cos(\theta_3) - \sin(\theta_2)\sin(\theta_3)))/20 - (\sin(\theta_3)(\cos(\theta_2)\sin(\theta_3) + \cos(\theta_3)\sin(\theta_2)))/10 + (\sin(\theta_4)(\cos(\theta_2)\sin(\theta_3) + \cos(\theta_3)\sin(\theta_2)))/20) - g \cdot m_3((\cos(\theta_2)\cos(\theta_3))/20 - (\sin(\theta_2)\sin(\theta_3))/20 - \cos(\theta_2)^2/5 + \sin(\theta_2)^2/5 - (\cos(\theta_3)(\cos(\theta_2)\cos(\theta_3) - \sin(\theta_2)\sin(\theta_3)))/10 + (\sin(\theta_3)(\cos(\theta_2)\sin(\theta_3) + \cos(\theta_3)\sin(\theta_2)))/10) - g \cdot m_2(\cos(\theta_2)/20 - \cos(\theta_2)^2/5 + \sin(\theta_2)^2/5)]$$

$$\frac{\partial V}{\partial \mathbf{q}}(3) = [g \cdot m_4((\cos(\theta_4)(\cos(\theta_4)(\cos(\theta_2)\cos(\theta_3) - \sin(\theta_2)\sin(\theta_3)) - \sin(\theta_4)(\cos(\theta_2)\sin(\theta_3) + \cos(\theta_3)\sin(\theta_2))))/10 - (\sin(\theta_4)(\cos(\theta_4)(\cos(\theta_2)\sin(\theta_3) + \cos(\theta_3)\sin(\theta_2)) + \sin(\theta_4)(\cos(\theta_2)\cos(\theta_3) - \sin(\theta_2)\sin(\theta_3))))/10 + (\cos(\theta_3)(\cos(\theta_2)\cos(\theta_3) - \sin(\theta_2)\sin(\theta_3)))/5 - (\cos(\theta_4)(\cos(\theta_2)\cos(\theta_3) - \sin(\theta_2)\sin(\theta_3)))/20 - (\sin(\theta_3)(\cos(\theta_2)\sin(\theta_3) + \cos(\theta_3)\sin(\theta_2)))/5 + (\sin(\theta_4)(\cos(\theta_2)\sin(\theta_3) + \cos(\theta_3)\sin(\theta_2)))/20) - g \cdot m_3((\cos(\theta_2)\cos(\theta_3))/20 - (\sin(\theta_2)\sin(\theta_3))/20 - (\cos(\theta_3)(\cos(\theta_2)\cos(\theta_3) - \sin(\theta_2)\sin(\theta_3)))/5 + (\sin(\theta_3)(\cos(\theta_2)\sin(\theta_3) + \cos(\theta_3)\sin(\theta_2)))/5)]$$

$$\frac{\partial V}{\partial \mathbf{q}}(4) = [g \cdot m_4((\cos(\theta_4)(\cos(\theta_4)(\cos(\theta_2)\cos(\theta_3) - \sin(\theta_2)\sin(\theta_3)) - \sin(\theta_4)(\cos(\theta_2)\sin(\theta_3) + \cos(\theta_3)\sin(\theta_2))))/5 - (\sin(\theta_4)(\cos(\theta_4)(\cos(\theta_2)\sin(\theta_3) + \cos(\theta_3)\sin(\theta_2)) + \sin(\theta_4)(\cos(\theta_2)\cos(\theta_3) - \sin(\theta_2)\sin(\theta_3))))/5 - (\cos(\theta_4)(\cos(\theta_2)\cos(\theta_3) - \sin(\theta_2)\sin(\theta_3)))/20 + (\sin(\theta_4)(\cos(\theta_2)\sin(\theta_3) + \cos(\theta_3)\sin(\theta_2)))/20)]$$

A.4 Larus Mass Matrix

$$m(1,1) = [I_{2,xx} + I_{3,xx}/2 + I_{4,xx}/2 + I_{1,yy} + I_{3,yy}/2 + I_{4,yy}/2 + I_{5,zz} + (361m_1)/40000 + (401m_2)/40000 + (41m_3)/800 + (21m_4)/800 + (11m_5)/400 + (1521m_2cos(\theta_2)^2)/160000 + (m_3cos(\theta_2)^2)/100 + (m_4cos(\theta_2)^2)/100 + (m_5cos(\theta_2)^2)/100 - (3m_4cos(2\theta_2 + 2\theta_3 + 2\theta_4))/800 + (m_5cos(2\theta_2 + 2\theta_3 + 2\theta_4))/200 - (I_{3,xx}cos(2\theta_2 + 2\theta_3))/2 + (I_{3,yy}cos(2\theta_2 + 2\theta_3))/2 - (m_4sin(2\theta_2 + 2\theta_3 + 2\theta_4))/200 + I_{3,xy}sin(2\theta_2 + 2\theta_3) - (m_3cos(2\theta_2 + 2\theta_3))/800 + (m_4cos(2\theta_2 + \theta_3 + \theta_4))/200 + (m_5cos(2\theta_2 + \theta_3 + \theta_4))/100 - (m_4sin(2\theta_2 + \theta_3 + \theta_4))/100 + (m_5cos(\theta_1 + \theta_2 + \theta_3 + \theta_4))/200 + (m_5cos(\theta_1 + \theta_2))/200 + (m_4cos(\theta_3 + \theta_4))/200 + (m_5cos(\theta_3 + \theta_4))/100 + (m_5cos(\theta_1 - \theta_2 - \theta_3 - \theta_4))/200 + (m_3sin(\theta_2 + \theta_3))/100 - (m_4sin(\theta_3 + \theta_4))/100 + (39m_2cos(\theta_2))/2000 + (m_3cos(\theta_2))/50 + (m_4cos(\theta_2))/50 + (m_5cos(\theta_1))/100 + (m_5cos(\theta_2))/50 + (m_3sin(\theta_3))/200 + (m_5sin(\theta_1))/100 + (m_5cos(\theta_1 - \theta_2))/200 - I_{2,xx}cos(\theta_2)^2 + I_{2,yy}cos(\theta_2)^2 - (I_{4,xx}cos(2\theta_2 + 2\theta_3 + 2\theta_4))/2 + (I_{4,yy}cos(2\theta_2 + 2\theta_3 + 2\theta_4))/2 + (m_3sin(2\theta_2 + \theta_3))/200 + (m_4cos(\theta_2 + \theta_3 + \theta_4))/100 + (m_5cos(\theta_2 + \theta_3 + \theta_4))/50 + I_{4,xy}sin(2\theta_2 + 2\theta_3 + 2\theta_4) - (m_4sin(\theta_2 + \theta_3 + \theta_4))/50 + 2I_{2,xy}cos(\theta_2)sin(\theta_2)]$$

$$m(1,2) = [I_{3,yz}cos(\theta_2 + \theta_3) + (m_5cos(\theta_1 + \theta_2 + \theta_3 + \theta_4))/400 + I_{3,xz}sin(\theta_2 + \theta_3) + (m_3cos(\theta_2 + \theta_3))/100 +$$

$$(m_5cos(\theta_1+\theta_2))/400-(m_5cos(\theta_1-\theta_2-\theta_3-\theta_4))/400+I_{2,yz}cos(\theta_2)-I_{5,yz}cos(\theta_1)+I_{2,xz}sin(\theta_2)+I_{5,xz}sin(\theta_1)+(39m_2sin(\theta_2))/80000-(m_3sin(\theta_2))/50-(m_4sin(\theta_2))/100-(m_5sin(\theta_2))/100+I_{4,yz}cos(\theta_2+\theta_3+\theta_4)-(m_5cos(\theta_1-\theta_2))/400+I_{4,xz}sin(\theta_2+\theta_3+\theta_4)-(m_4cos(\theta_2+\theta_3+\theta_4))/100-(m_4sin(\theta_2+\theta_3+\theta_4))/200-(m_5sin(\theta_2+\theta_3+\theta_4))/100]$$

$$m(1,3)=[I_{3,yz}cos(\theta_2+\theta_3)+(m_5cos(\theta_1+\theta_2+\theta_3+\theta_4))/400+I_{3,xz}sin(\theta_2+\theta_3)+(m_3cos(\theta_2+\theta_3))/100-(m_5cos(\theta_1-\theta_2-\theta_3-\theta_4))/400-I_{5,yz}cos(\theta_1)+I_{5,xz}sin(\theta_1)+I_{4,yz}cos(\theta_2+\theta_3+\theta_4)+I_{4,xz}sin(\theta_2+\theta_3+\theta_4)-(m_4cos(\theta_2+\theta_3+\theta_4))/100-(m_4sin(\theta_2+\theta_3+\theta_4))/200-(m_5sin(\theta_2+\theta_3+\theta_4))/100]$$

$$m(1,4)=[(m_5cos(\theta_1+\theta_2+\theta_3+\theta_4))/400-(m_5cos(\theta_1-\theta_2-\theta_3-\theta_4))/400+I_{4,yz}cos(\theta_2+\theta_3+\theta_4)+I_{4,xz}sin(\theta_2+\theta_3+\theta_4)-(m_4cos(\theta_2+\theta_3+\theta_4))/100-(m_4sin(\theta_2+\theta_3+\theta_4))/200-(m_5sin(\theta_2+\theta_3+\theta_4))/100]$$

$$m(2,1)=[I_{3,yz}cos(\theta_2+\theta_3)+(m_5cos(\theta_1+\theta_2+\theta_3+\theta_4))/400+I_{3,xz}sin(\theta_2+\theta_3)+(m_3cos(\theta_2+\theta_3))/100+(m_5cos(\theta_1+\theta_2))/400-(m_5cos(\theta_1-\theta_2-\theta_3-\theta_4))/400+I_{2,yz}cos(\theta_2)-I_{5,yz}cos(\theta_1)+I_{2,xz}sin(\theta_2)+I_{5,xz}sin(\theta_1)+(39m_2sin(\theta_2))/80000-(m_3sin(\theta_2))/50-(m_4sin(\theta_2))/100-(m_5sin(\theta_2))/100+I_{4,yz}cos(\theta_2+\theta_3+\theta_4)-(m_5cos(\theta_1-\theta_2))/400+I_{4,xz}sin(\theta_2+\theta_3+\theta_4)-(m_4cos(\theta_2+\theta_3+\theta_4))/100-(m_4sin(\theta_2+\theta_3+\theta_4))/200-(m_5sin(\theta_2+\theta_3+\theta_4))/100]$$

$$m(2,2)=[I_{1,zz}+I_{3,zz}+I_{4,zz}+(1521m_2)/160000+m_3/80+(9m_4)/400+m_5(sin(\theta_2+\theta_3+\theta_4)/10+sin(\theta_2)/10)^2+(m_4cos(\theta_3+\theta_4))/100+m_5(cos(\theta_2+\theta_3+\theta_4)/10+cos(\theta_1)/20+cos(\theta_2)/10)^2-(m_4sin(\theta_3+\theta_4))/50+cos(\theta_1)(I_{5,yy}cos(\theta_1)-I_{5,xy}sin(\theta_1))+(m_3sin(\theta_3))/100-sin(\theta_1)(I_{5,xy}cos(\theta_1)-I_{5,xx}sin(\theta_1))]$$

$$m(2,3)=[I_{5,xx}/2+I_{5,yy}/2+I_{3,zz}+I_{4,zz}+m_3/400+m_4/80+(9m_5)/800+(m_5cos(2\theta_1))/800+(m_5cos(\theta_1+\theta_2+\theta_3+\theta_4))/200+(m_5cos(\theta_1+\theta_2))/400+(m_4cos(\theta_3+\theta_4))/200+(m_5cos(\theta_3+\theta_4))/100+(m_5cos(\theta_1-\theta_2-\theta_3-\theta_4))/200-(m_4sin(\theta_3+\theta_4))/100+(m_3sin(\theta_3))/200+(m_5cos(\theta_1-\theta_2))/400-(I_{5,xx}cos(2\theta_1))/2+(I_{5,yy}cos(2\theta_1))/2-I_{5,xy}sin(2\theta_1)]$$

$$m(2,4)=[I_{4,zz}+m_4/80+m_5/100+(m_5cos(\theta_1+\theta_2+\theta_3+\theta_4))/400+(m_4cos(\theta_3+\theta_4))/200+(m_5cos(\theta_3+\theta_4))/100+(m_5cos(\theta_1-\theta_2-\theta_3-\theta_4))/400-(m_4sin(\theta_3+\theta_4))/100]$$

$$m(3,1)=[I_{3,yz}cos(\theta_2+\theta_3)+(m_5cos(\theta_1+\theta_2+\theta_3+\theta_4))/400+I_{3,xz}sin(\theta_2+\theta_3)+(m_3cos(\theta_2+\theta_3))/100-(m_5cos(\theta_1-\theta_2-\theta_3-\theta_4))/400-I_{5,yz}cos(\theta_1)+I_{5,xz}sin(\theta_1)+I_{4,yz}cos(\theta_2+\theta_3+\theta_4)+I_{4,xz}sin(\theta_2+\theta_3+\theta_4)-(m_4cos(\theta_2+\theta_3+\theta_4))/100-(m_4sin(\theta_2+\theta_3+\theta_4))/200-(m_5sin(\theta_2+\theta_3+\theta_4))/100]$$

$$m(3,2)=[I_{5,xx}/2+I_{5,yy}/2+I_{3,zz}+I_{4,zz}+m_3/400+m_4/80+(9m_5)/800+(m_5cos(2\theta_1))/800+(m_5cos(\theta_1+\theta_2+\theta_3+\theta_4))/200+(m_5cos(\theta_1+\theta_2))/400+(m_4cos(\theta_3+\theta_4))/200+(m_5cos(\theta_3+\theta_4))/100+(m_5cos(\theta_1-\theta_2-\theta_3-\theta_4))/200-(m_4sin(\theta_3+\theta_4))/100+(m_3sin(\theta_3))/200+(m_5cos(\theta_1-\theta_2))/400-(I_{5,xx}cos(2\theta_1))/2+(I_{5,yy}cos(2\theta_1))/2-I_{5,xy}sin(2\theta_1)]$$

$$m(3,3)=[I_{5,xx}/2+I_{5,yy}/2+I_{3,zz}+I_{4,zz}+m_3/400+m_4/80+(9m_5)/800+$$

$(m_5cos(2\theta_1))/800 + (m_5cos(\theta_1+\theta_2+\theta_3+\theta_4))/200 + (m_5cos(\theta_1-\theta_2-\theta_3-\theta_4))/200 - (I_{5,xx}cos(2\theta_1))/2 + (I_{5,yy}cos(2\theta_1))/2 - I_{5,xy}sin(2\theta_1)]$

$m(3,4) = [I_{4,zz} + m_4/80 + m_5/100 + (m_5cos(\theta_1+\theta_2+\theta_3+\theta_4))/400 + (m_5cos(\theta_1-\theta_2-\theta_3-\theta_4))/400]$

$m(4,1) = [(m_5cos(\theta_1+\theta_2+\theta_3+\theta_4))/400 - (m_5cos(\theta_1-\theta_2-\theta_3-\theta_4))/400 + I_{4,yz}cos(\theta_2+\theta_3+\theta_4) + I_{4,xz}sin(\theta_2+\theta_3+\theta_4) - (m_4cos(\theta_2+\theta_3+\theta_4))/100 - (m_4sin(\theta_2+\theta_3+\theta_4))/200 - (m_5sin(\theta_2+\theta_3+\theta_4))/100]$

$m(4,2) = [I_{4,zz} + m_4/80 + m_5/100 + (m_5cos(\theta_1+\theta_2+\theta_3+\theta_4))/400 + (m_4cos(\theta_3+\theta_4))/200 + (m_5cos(\theta_3+\theta_4))/100 + (m_5cos(\theta_1-\theta_2-\theta_3-\theta_4))/400 - (m_4sin(\theta_3+\theta_4))/100]$

$m(4,3) = [I_{4,zz} + m_4/80 + m_5/100 + (m_5cos(\theta_1+\theta_2+\theta_3+\theta_4))/400 + (m_5cos(\theta_1-\theta_2-\theta_3-\theta_4))/400]$

$m(4,4) = [I_{4,zz} + m_4/80 + m_5/100]$

A.5 Larus Nonlinear Coriolis Matrix

$C(1,1) = [-\dot{\theta}_3((m_4\cos(2\theta_2+2\theta_3+2\theta_4))/200 - I_{3,xy}\cos(2\theta_2+2\theta_3) - (3m_4\sin(2\theta_2+2\theta_3+2\theta_4))/800 + (m_5\sin(2\theta_2+2\theta_3+2\theta_4))/200 - (I_{3,xx}\sin(2\theta_2+2\theta_3))/2 + (I_{3,yy}\sin(2\theta_2+2\theta_3))/2 - (m_3\sin(2\theta_2+2\theta_3))/800 + (m_4\cos(2\theta_2+\theta_3+\theta_4))/200 + (m_4\sin(2\theta_2+\theta_3+\theta_4))/400 + (m_5\sin(2\theta_2+\theta_3+\theta_4))/200 + (m_5\sin(\theta_1+\theta_2+\theta_3+\theta_4))/400 - (m_3\cos(\theta_2+\theta_3))/200 + (m_4\cos(\theta_3+\theta_4))/200 + (m_4\sin(\theta_3+\theta_4))/400 + (m_5\sin(\theta_3+\theta_4))/200 - (m_5\sin(\theta_1-\theta_2-\theta_3-\theta_4))/400 - (m_3\cos(\theta_3))/400 - (m_3\cos(2\theta_2+\theta_3))/400 - I_{4,xy}\cos(2\theta_2+2\theta_3+2\theta_4) + (m_4\cos(\theta_2+\theta_3+\theta_4))/100 - (I_{4,xx}\sin(2\theta_2+2\theta_3+2\theta_4))/2 + (I_{4,yy}\sin(2\theta_2+2\theta_3+2\theta_4))/2 + (m_4\sin(\theta_2+\theta_3+\theta_4))/200 + (m_5\sin(\theta_2+\theta_3+\theta_4))/100) - \dot{\theta}_2((m_4\cos(2\theta_2+2\theta_3+2\theta_4))/200 - I_{3,xy}\cos(2\theta_2+2\theta_3) + (1521m_2\sin(2\theta_2))/320000 + (m_3\sin(2\theta_2))/200 + (m_4\sin(2\theta_2))/200 + (m_5\sin(2\theta_2))/200 - (3m_4\sin(2\theta_2+2\theta_3+2\theta_4))/800 + (m_5\sin(2\theta_2+2\theta_3+2\theta_4))/200 - (I_{3,xx}\sin(2\theta_2+2\theta_3))/2 + (I_{3,yy}\sin(2\theta_2+2\theta_3))/2 - (m_3\sin(2\theta_2+2\theta_3))/800 + (m_4\cos(2\theta_2+\theta_3+\theta_4))/100 + (m_4\sin(2\theta_2+\theta_3+\theta_4))/200 + (m_5\sin(2\theta_2+\theta_3+\theta_4))/100 + (m_5\sin(\theta_1+\theta_2+\theta_3+\theta_4))/400 - (m_3\cos(\theta_2+\theta_3))/200 + (m_5\sin(\theta_1+\theta_2))/400 - (m_5\sin(\theta_1-\theta_2-\theta_3-\theta_4))/400 + (39m_2\sin(\theta_2))/4000 + (m_3\sin(\theta_2))/100 + (m_4\sin(\theta_2))/100 + (m_5\sin(\theta_2))/100 - (m_3\cos(2\theta_2+\theta_3))/200 - I_{2,xy}\cos(2\theta_2) - I_{4,xy}\cos(2\theta_2+2\theta_3+2\theta_4) - (m_5\sin(\theta_1-\theta_2))/400 + (m_4\cos(\theta_2+\theta_3+\theta_4))/100 - (I_{2,xx}\sin(2\theta_2))/2 + (I_{2,yy}\sin(2\theta_2))/2 - (I_{4,xx}\sin(2\theta_2+2\theta_3+2\theta_4))/2 + (I_{4,yy}\sin(2\theta_2+2\theta_3+2\theta_4))/2 + (m_4\sin(\theta_2+\theta_3+\theta_4))/200 + (m_5\sin(\theta_2+\theta_3+\theta_4))/100) - \dot{\theta}_4((m_4\cos(2\theta_2+2\theta_3+2\theta_4))/200 - (3m_4\sin(2\theta_2+2\theta_3+2\theta_4))/800 + (m_5\sin(2\theta_2+2\theta_3+2\theta_4))/200 + (m_4\cos(2\theta_2+\theta_3+\theta_4))/200 + (m_4\sin(2\theta_2+\theta_3+\theta_4))/400 + (m_5\sin(2\theta_2+\theta_3+\theta_4))/200 + (m_5\sin(\theta_1+\theta_2+\theta_3+\theta_4))/400 + (m_4\cos(\theta_3+\theta_4))/200 + (m_4\sin(\theta_3+\theta_4))/400 + (m_5\sin(\theta_3+\theta_4))/200 - (m_5\sin(\theta_1-\theta_2-\theta_3-\theta_4))/400 - I_{4,xy}\cos(2\theta_2+2\theta_3+2\theta_4) + (m_4\cos(\theta_2+\theta_3+\theta_4))/100 - (I_{4,xx}\sin(2\theta_2+2\theta_3+2\theta_4))/2 + (I_{4,yy}\sin(2\theta_2+2\theta_3+2\theta_4))/2 + (m_4\sin(\theta_2+\theta_3+\theta_4))/200 + (m_5\sin(\theta_2+\theta_3+\theta_4))/100) - \dot{\theta}_1((m_5\sin(\theta_1+\theta_2+\theta_3+\theta_4))/400 + (m_5\sin(\theta_1+\theta_2))/400 + (m_5\sin(\theta_1-\theta_2-\theta_3-\theta_4))/400 - (m_5\cos(\theta_1))/200 + (m_5\sin(\theta_1))/200 + (m_5\sin(\theta_1-\theta_2))/400)]$

$C(1,2) = [\dot{\theta}_3((m_5\sin(2\theta_1))/800 + I_{3,xz}\cos(\theta_2+\theta_3) - I_{3,yz}\sin(\theta_2+\theta_3) - (m_3\sin(\theta_2+\theta_3))/100 + (m_5\sin(\theta_1+\theta_2))/800 + I_{4,xz}\cos(\theta_2+\theta_3+\theta_4) - I_{4,yz}\sin(\theta_2+\theta_3+\theta_4) + I_{5,xy}\cos(2\theta_1) + (m_5\sin(\theta_1-\theta_2))/800 - (m_4\cos(\theta_2+\theta_3+\theta_4))/200 - (m_5\cos(\theta_2+\theta_3+\theta_4))/100 - (I_{5,xx}\sin(2\theta_1))/2 + (I_{5,yy}\sin(2\theta_1))/2 + (m_4\sin(\theta_2+\theta_3+\theta_4))/100) - \dot{\theta}_1((m_4\cos(2\theta_2+2\theta_3+2\theta_4))/200 - I_{3,xy}\cos(2\theta_2+2\theta_3) + (1521m_2\sin(2\theta_2))/320000 + (m_3\sin(2\theta_2))/200 + (m_4\sin(2\theta_2))/200 + (m_5\sin(2\theta_2))/200 - (3m_4\sin(2\theta_2+2\theta_3+2\theta_4))/800 + (m_5\sin(2\theta_2+2\theta_3+2\theta_4))/200 - (I_{3,xx}\sin(2\theta_2+2\theta_3))/2 + (I_{3,yy}\sin(2\theta_2+2\theta_3))/2 - (m_3\sin(2\theta_2+2\theta_3))/800 + (m_4\cos(2\theta_2+\theta_3+\theta_4))/100 + (m_4\sin(2\theta_2+\theta_3+\theta_4))/200 + (m_5\sin(2\theta_2+\theta_3+\theta_4))/100 + (m_5\sin(\theta_1+\theta_2+\theta_3+\theta_4))/400 - (m_3\cos(\theta_2+\theta_3))/200 + (m_5\sin(\theta_1+\theta_2))/400 - (m_5\sin(\theta_1-\theta_2-\theta_3-\theta_4))/400 + (39m_2\sin(\theta_2))/4000 + (m_3\sin(\theta_2))/100 + (m_4\sin(\theta_2))/100 + (m_5\sin(\theta_2))/100 - (m_3\cos(2\theta_2+\theta_3))/200 - I_{2,xy}\cos(2\theta_2) - I_{4,xy}\cos(2\theta_2+2\theta_3+2\theta_4) - (m_5\sin(\theta_1-\theta_2))/400 + (m_4\cos(\theta_2+\theta_3+\theta_4))/100 - (I_{2,xx}\sin(2\theta_2))/2 + (I_{2,yy}\sin(2\theta_2))/2 - (I_{4,xx}\sin(2\theta_2+2\theta_3+2\theta_4))/2 + (I_{4,yy}\sin(2\theta_2+2\theta_3+2\theta_4))/2 + (m_4\sin(\theta_2+\theta_3+\theta_4))/200 + (m_5\sin(\theta_2+\theta_3+\theta_4))/100) - \dot{\theta}_4((m_5\sin(\theta_1+\theta_2+\theta_3+\theta_4))/800 + (m_5\sin(\theta_1-\theta_2-\theta_3-\theta_4))/800 - I_{4,xz}\cos(\theta_2+\theta_3+\theta_4) + I_{4,yz}\sin(\theta_2+\theta_3+\theta_4) + (m_4\cos(\theta_2+\theta_3+\theta_4))/200 + (m_5\cos(\theta_2+\theta_3+\theta_4))/100 - (m_4\sin(\theta_2+\theta_3+\theta_4))/100) - \dot{\theta}_2(I_{3,yz}\sin(\theta_2+\theta_3) - I_{3,xz}\cos(\theta_2+\theta_3) + (m_5\sin(\theta_1+\theta_2+\theta_3+\theta_4))/400 - I_{2,xz}\cos(\theta_2) + (m_3\sin(\theta_2+\theta_3))/100 + (m_5\sin(\theta_1+\theta_2))/400 + (m_5\sin(\theta_1-\theta_2-\theta_3-\theta_4))/400 + I_{2,yz}\sin(\theta_2) - (39m_2\cos(\theta_2))/80000 + (m_3\cos(\theta_2))/50 + (m_4\cos(\theta_2))/100 + (m_5\cos(\theta_2))/100 - \cos(\theta_1)((I_{5,xy}\cos(\theta_1))/2 - (I_{5,xx}\sin(\theta_1))/2) - \cos(\theta_1)((I_{5,xy}\cos(\theta_1))/2 + (I_{5,yy}\sin(\theta_1))/2) + \sin(\theta_1)((I_{5,xx}\cos(\theta_1))/2 + (I_{5,xy}\sin(\theta_1))/2) - \sin(\theta_1)((I_{5,yy}\cos(\theta_1))/2 - (I_{5,xy}\sin(\theta_1))/2) - I_{4,xz}\cos(\theta_2+\theta_3+\theta_4) + I_{4,yz}\sin(\theta_2+\theta_3+\theta_4) + (m_5\sin(\theta_1-\theta_2))/400 + (m_4\cos(\theta_2+\theta_3+\theta_4))/200 + (m_5\cos(\theta_2+\theta_3+\theta_4))/100 - (m_4\sin(\theta_2+\theta_3+\theta_4))/100 - (m_5\sin(\theta_1)(\cos(\theta_2+\theta_3+\theta_4)/10 + \cos(\theta_1)/20 + \cos(\theta_2)/10))/20)]$

$C(1,3) = [\dot{\theta}_2((m_5\sin(2\theta_1))/800 + I_{3,xz}\cos(\theta_2+\theta_3) - I_{3,yz}\sin(\theta_2+\theta_3) - (m_3\sin(\theta_2+\theta_3))/100 + (m_5\sin(\theta_1+\theta_2))/800 + I_{4,xz}\cos(\theta_2+\theta_3+\theta_4) - I_{4,yz}\sin(\theta_2+\theta_3+\theta_4) + I_{5,xy}\cos(2\theta_1) + (m_5\sin(\theta_1-\theta_2))/800 - (m_4\cos(\theta_2+\theta_3+\theta_4))/200 - (m_5\cos(\theta_2+\theta_3+\theta_4))/100 - (I_{5,xx}\sin(2\theta_1))/2 + (I_{5,yy}\sin(2\theta_1))/2 + (m_4\sin(\theta_2+\theta_3+\theta_4))/100) - \dot{\theta}_1((m_4\cos(2\theta_2+2\theta_3+2\theta_4))/200 - I_{3,xy}\cos(2\theta_2+2\theta_3) - (3m_4\sin(2\theta_2+2\theta_3+2\theta_4))/800 + (m_5\sin(2\theta_2+2\theta_3+2\theta_4))/200 - (I_{3,xx}\sin(2\theta_2+2\theta_3))/2 + (I_{3,yy}\sin(2\theta_2+2\theta_3))/2 - (m_3\sin(2\theta_2+2\theta_3))/800 + (m_4\cos(2\theta_2+\theta_3+\theta_4))/200 + (m_4\sin(2\theta_2+\theta_3+\theta_4))/400 + (m_5\sin(2\theta_2+\theta_3+\theta_4))/200 + (m_5\sin(\theta_1+\theta_2+\theta_3+\theta_4))/400 - (m_3\cos(\theta_2+\theta_3))/200 + (m_4\cos(\theta_3+\theta_4))/200 + (m_4\sin(\theta_3+\theta_4))/400 + (m_5\sin(\theta_3+\theta_4))/200 - (m_5\sin(\theta_1-\theta_2-\theta_3-\theta_4))/400 - (m_3\cos(\theta_3))/400 - (m_3\cos(2\theta_2+\theta_3))/400 - I_{4,xy}\cos(2\theta_2+2\theta_3+2\theta_4) + (m_4\cos(\theta_2+\theta_3+\theta_4))/100 - (I_{4,xx}\sin(2\theta_2+2\theta_3+2\theta_4))/2 + (I_{4,yy}\sin(2\theta_2+2\theta_3+2\theta_4))/2 + (m_4\sin(\theta_2+\theta_3+\theta_4))/200 + (m_5\sin(\theta_2+\theta_3+\theta_4))/100) - \dot{\theta}_4((m_5\sin(\theta_1+\theta_2+\theta_3+\theta_4))/800 + (m_5\sin(\theta_1-\theta_2-\theta_3-\theta_4))/800 - I_{4,xz}\cos(\theta_2+\theta_3+\theta_4) + I_{4,yz}\sin(\theta_2+\theta_3+\theta_4) + (m_4\cos(\theta_2+\theta_3+\theta_4))/200 + (m_5\cos(\theta_2+\theta_3+\theta_4))/100 - (m_4\sin(\theta_2+\theta_3+\theta_4))/100) + \dot{\theta}_3((m_5\sin(2\theta_1))/800 + I_{3,xz}\cos(\theta_2+\theta_3) - I_{3,yz}\sin(\theta_2+\theta_3) - (m_3\sin(\theta_2+\theta_3))/100 + I_{4,xz}\cos(\theta_2+\theta_3+\theta_4) - I_{4,yz}\sin(\theta_2+\theta_3+\theta_4) + I_{5,xy}\cos(2\theta_1) - (m_4\cos(\theta_2+\theta_3+\theta_4))/200 - (m_5\cos(\theta_2+\theta_3+\theta_4))/100 - (I_{5,xx}\sin(2\theta_1))/2 + (I_{5,yy}\sin(2\theta_1))/2 + (m_4\sin(\theta_2+\theta_3+\theta_4))/100)]$

$C(1,4) = [-\dot{\theta}_4((m_5\sin(\theta_1+\theta_2+\theta_3+\theta_4))/400 + (m_5\sin(\theta_1-\theta_2-\theta_3-\theta_4))/400 - I_{4,xz}\cos(\theta_2+\theta_3+\theta_4) + I_{4,yz}\sin(\theta_2+\theta_3+\theta_4) + (m_4\cos(\theta_2+\theta_3+\theta_4))/200 + (m_5\cos(\theta_2+\theta_3+\theta_4))/100 - (m_4\sin(\theta_2+\theta_3+$

$\theta_4))/100) - \dot{\theta}_2((m_5\sin(\theta_1+\theta_2+\theta_3+\theta_4))/800 + (m_5\sin(\theta_1-\theta_2-\theta_3-\theta_4))/800 - I_{4,xz}\cos(\theta_2+\theta_3+$
$\theta_4) + I_{4,yz}\sin(\theta_2+\theta_3+\theta_4) + (m_4\cos(\theta_2+\theta_3+\theta_4))/200 + (m_5\cos(\theta_2+\theta_3+\theta_4))/100 - (m_4\sin(\theta_2+\theta_3+$
$\theta_4))/100) - \dot{\theta}_3((m_5\sin(\theta_1+\theta_2+\theta_3+\theta_4))/800 + (m_5\sin(\theta_1-\theta_2-\theta_3-\theta_4))/800 - I_{4,xz}\cos(\theta_2+\theta_3+$
$\theta_4) + I_{4,yz}\sin(\theta_2+\theta_3+\theta_4) + (m_4\cos(\theta_2+\theta_3+\theta_4))/200 + (m_5\cos(\theta_2+\theta_3+\theta_4))/100 - (m_4\sin(\theta_2+\theta_3+$
$\theta_4))/100) - \dot{\theta}_1((m_4\cos(2\theta_2+2\theta_3+2\theta_4))/200 - (3m_4\sin(2\theta_2+2\theta_3+2\theta_4))/800 + (m_5\sin(2\theta_2+2\theta_3+$
$2\theta_4))/200 + (m_4\cos(2\theta_2+\theta_3+\theta_4))/200 + (m_4\sin(2\theta_2+\theta_3+\theta_4))/400 + (m_5\sin(2\theta_2+\theta_3+\theta_4))/200+$
$(m_5\sin(\theta_1+\theta_2+\theta_3+\theta_4))/400 + (m_4\cos(\theta_3+\theta_4))/200 + (m_4\sin(\theta_3+\theta_4))/400 + (m_5\sin(\theta_3+\theta_4))/200$
$-(m_5\sin(\theta_1-\theta_2-\theta_3-\theta_4))/400 - I_{4,xy}\cos(2\theta_2+2\theta_3+2\theta_4) + (m_4\cos(\theta_2+\theta_3+\theta_4))/100 - (I_{4,xx}\sin(2\theta_2$
$+2\theta_3+2\theta_4))/2 + (I_{4,yy}\sin(2\theta_2+2\theta_3+2\theta_4))/2 + (m_4\sin(\theta_2+\theta_3+\theta_4))/200 + (m_5\sin(\theta_2+\theta_3+$
$\theta_4))/100)]$

$C(2,1) = [\dot{\theta}_1((m_4\cos(2\theta_2+2\theta_3+2\theta_4))/200 - I_{3,xy}\cos(2\theta_2+2\theta_3) + (1521m_2\sin(2\theta_2))/320000+$
$(m_3\sin(2\theta_2))/200 + (m_4\sin(2\theta_2))/200 + (m_5\sin(2\theta_2))/200 - (3m_4\sin(2\theta_2+2\theta_3+2\theta_4))/800+$
$(m_5\sin(2\theta_2+2\theta_3+2\theta_4))/200 - (I_{3,xx}\sin(2\theta_2+2\theta_3))/2 + (I_{3,yy}\sin(2\theta_2+2\theta_3))/2 - (m_3\sin(2\theta_2+$
$2\theta_3))/800 + (m_4\cos(2\theta_2+\theta_3+\theta_4))/100 + (m_4\sin(2\theta_2+\theta_3+\theta_4))/200 + (m_5\sin(2\theta_2+\theta_3+\theta_4))/100-$
$(m_3\cos(\theta_2+\theta_3))/200 + I_{5,xz}\cos(\theta_1) + I_{5,yz}\sin(\theta_1) + (39m_2\sin(\theta_2))/4000 + (m_3\sin(\theta_2))/100+$
$(m_4\sin(\theta_2))/100 + (m_5\sin(\theta_2))/100 - (m_3\cos(2\theta_2+\theta_3))/200 - I_{2,xy}\cos(2\theta_2) - I_{4,xy}\cos(2\theta_2+2\theta_3+$
$2\theta_4) + (m_4\cos(\theta_2+\theta_3+\theta_4))/100 - (I_{2,xx}\sin(2\theta_2))/2 + (I_{2,yy}\sin(2\theta_2))/2 - (I_{4,xx}\sin(2\theta_2+2\theta_3+2\theta_4))/2$
$+(I_{4,yy}\sin(2\theta_2+2\theta_3+2\theta_4))/2 + (m_4\sin(\theta_2+\theta_3+\theta_4))/200 + (m_5\sin(\theta_2+\theta_3+\theta_4))/100) - \dot{\theta}_4((m_5\sin(\theta_1$
$+\theta_2+\theta_3+\theta_4))/800 + (m_5\sin(\theta_1-\theta_2-\theta_3-\theta_4))/800) - \dot{\theta}_3((m_5\sin(2\theta_1))/800 + (m_5\sin(\theta_1+\theta_2+\theta_3+$
$\theta_4))/400 + (m_5\sin(\theta_1+\theta_2))/800 + (m_5\sin(\theta_1-\theta_2-\theta_3-\theta_4))/400 + I_{5,xy}\cos(2\theta_1) + (m_5\sin(\theta_1-$
$\theta_2))/800 - (I_{5,xx}\sin(2\theta_1))/2 + (I_{5,yy}\sin(2\theta_1))/2) - \dot{\theta}_2(\cos(\theta_1)((I_{5,xy}\cos(\theta_1))/2 - (I_{5,xx}\sin(\theta_1))/2)+$
$\cos(\theta_1)((I_{5,xy}\cos(\theta_1))/2 + (I_{5,yy}\sin(\theta_1))/2) - \sin(\theta_1)((I_{5,xx}\cos(\theta_1))/2 + (I_{5,xy}\sin(\theta_1))/2)+$
$\sin(\theta_1)((I_{5,yy}\cos(\theta_1))/2 - (I_{5,xy}\sin(\theta_1))/2) + (m_5\sin(\theta_1)(\cos(\theta_2+\theta_3+\theta_4)/10 + \cos(\theta_1)/20+$
$\cos(\theta_2)/10))/20)]$

$C(2,2) = [\dot{\theta}_2(m_5(\cos(\theta_2+\theta_3+\theta_4)/10 + \cos(\theta_2)/10)(\sin(\theta_2+\theta_3+\theta_4)/10 + \sin(\theta_2)/10) - m_5(\sin(\theta_2+$
$\theta_3+\theta_4)/10 + \sin(\theta_2)/10)(\cos(\theta_2+\theta_3+\theta_4)/10 + \cos(\theta_1)/20 + \cos(\theta_2)/10)) - \dot{\theta}_3((m_4\cos(\theta_3+\theta_4))/100$
$+(m_4\sin(\theta_3+\theta_4))/200 - (m_3\cos(\theta_3))/200 - (m_5\cos(\theta_2+\theta_3+\theta_4)(\sin(\theta_2+\theta_3+\theta_4)/10+$
$\sin(\theta_2)/10))/10 + (m_5\sin(\theta_2+\theta_3+\theta_4)(\cos(\theta_2+\theta_3+\theta_4)/10 + \cos(\theta_1)/20 + \cos(\theta_2)/10))/10)-$
$\dot{\theta}_4((m_4\cos(\theta_3+\theta_4))/100 + (m_4\sin(\theta_3+\theta_4))/200 - (m_5\cos(\theta_2+\theta_3+\theta_4)(\sin(\theta_2+\theta_3+\theta_4)/10+$
$\sin(\theta_2)/10))/10 + (m_5\sin(\theta_2+\theta_3+\theta_4)(\cos(\theta_2+\theta_3+\theta_4)/10 + \cos(\theta_1)/20 + \cos(\theta_2)/10))/10)-$
$\dot{\theta}_1(\cos(\theta_1)((I_{5,xy}\cos(\theta_1))/2 - (I_{5,xx}\sin(\theta_1))/2) + \cos(\theta_1)((I_{5,xy}\cos(\theta_1))/2 + (I_{5,yy}\sin(\theta_1))/2)-$
$\sin(\theta_1)((I_{5,xx}\cos(\theta_1))/2 + (I_{5,xy}\sin(\theta_1))/2) + \sin(\theta_1)((I_{5,yy}\cos(\theta_1))/2 - (I_{5,xy}\sin(\theta_1))/2)+$
$(m_5\sin(\theta_1)(\cos(\theta_2+\theta_3+\theta_4)/10 + \cos(\theta_1)/20 + \cos(\theta_2)/10))/20)]$

$C(2,3) = [-\dot{\theta}_2((m_4\cos(\theta_3+\theta_4))/100 + (m_4\sin(\theta_3+\theta_4))/200 - (m_3\cos(\theta_3))/200 - (m_5\cos(\theta_2+\theta_3+$
$\theta_4)(\sin(\theta_2+\theta_3+\theta_4)/10 + \sin(\theta_2)/10))/10 + (m_5\sin(\theta_2+\theta_3+\theta_4)(\cos(\theta_2+\theta_3+\theta_4)/10 + \cos(\theta_1)/20+$
$\cos(\theta_2)/10))/10) - \dot{\theta}_3((m_5\sin(\theta_1+\theta_2+\theta_3+\theta_4))/400 + (m_4\cos(\theta_3+\theta_4))/100 + (m_4\sin(\theta_3+\theta_4))/200+$
$(m_5\sin(\theta_3+\theta_4))/100 - (m_5\sin(\theta_1-\theta_2-\theta_3-\theta_4))/400 - (m_3\cos(\theta_3))/200) - \dot{\theta}_1((m_5\sin(2\theta_1))/800+$
$(m_5\sin(\theta_1+\theta_2+\theta_3+\theta_4))/400 + (m_5\sin(\theta_1+\theta_2))/800 + (m_5\sin(\theta_1-\theta_2-\theta_3-\theta_4))/400 + I_{5,xy}\cos(2\theta_1)$
$+(m_5\sin(\theta_1-\theta_2))/800 - (I_{5,xx}\sin(2\theta_1))/2 + (I_{5,yy}\sin(2\theta_1))/2) - \dot{\theta}_4((m_5\sin(\theta_1+\theta_2+\theta_3+\theta_4))/400+$

$$(m_4\cos(\theta_3+\theta_4))/100+(m_4\sin(\theta_3+\theta_4))/200+(m_5\sin(\theta_3+\theta_4))/100-(m_5\sin(\theta_1-\theta_2-\theta_3-\theta_4))/400)]$$

$$C(2,4)=[-\dot{\theta}_1((m_5\sin(\theta_1+\theta_2+\theta_3+\theta_4))/800+(m_5\sin(\theta_1-\theta_2-\theta_3-\theta_4))/800)-\dot{\theta}_2((m_4\cos(\theta_3+\theta_4))/100+(m_4\sin(\theta_3+\theta_4))/200-(m_5\cos(\theta_2+\theta_3+\theta_4)(\sin(\theta_2+\theta_3+\theta_4)/10+\sin(\theta_2)/10))/10+(m_5\sin(\theta_2+\theta_3+\theta_4)(\cos(\theta_2+\theta_3+\theta_4)/10+\cos(\theta_1)/20+\cos(\theta_2)/10))/10)-\dot{\theta}_3((m_5\sin(\theta_1+\theta_2+\theta_3+\theta_4))/400+(m_4\cos(\theta_3+\theta_4))/100+(m_4\sin(\theta_3+\theta_4))/200+(m_5\sin(\theta_3+\theta_4))/100-(m_5\sin(\theta_1-\theta_2-\theta_3-\theta_4))/400)-\dot{\theta}_4((m_5\sin(\theta_1+\theta_2+\theta_3+\theta_4))/400+(m_4\cos(\theta_3+\theta_4))/100+(m_4\sin(\theta_3+\theta_4))/200+(m_5\sin(\theta_3+\theta_4))/100-(m_5\sin(\theta_1-\theta_2-\theta_3-\theta_4))/400)]$$

$$C(3,1)=[\dot{\theta}_1((m_4\cos(2\theta_2+2\theta_3+2\theta_4))/200-I_{3,xy}\cos(2\theta_2+2\theta_3)-(3m_4\sin(2\theta_2+2\theta_3+2\theta_4))/800+(m_5\sin(2\theta_2+2\theta_3+2\theta_4))/200-(I_{3,xx}\sin(2\theta_2+2\theta_3))/2+(I_{3,yy}\sin(2\theta_2+2\theta_3))/2-(m_3\sin(2\theta_2+2\theta_3))/800+(m_4\cos(2\theta_2+\theta_3+\theta_4))/200+(m_4\sin(2\theta_2+\theta_3+\theta_4))/400+(m_5\sin(2\theta_2+\theta_3+\theta_4))/200-(m_3\cos(\theta_2+\theta_3))/200+(m_4\cos(\theta_3+\theta_4))/200+I_{5,xz}\cos(\theta_1)+(m_4\sin(\theta_3+\theta_4))/400+(m_5\sin(\theta_3+\theta_4))/200+I_{5,yz}\sin(\theta_1)-(m_3\cos(\theta_3))/400-(m_3\cos(2\theta_2+\theta_3))/400-I_{4,xy}\cos(2\theta_2+2\theta_3+2\theta_4)+(m_4\cos(\theta_2+\theta_3+\theta_4))/100-(I_{4,xx}\sin(2\theta_2+2\theta_3+2\theta_4))/2+(I_{4,yy}\sin(2\theta_2+2\theta_3+2\theta_4))/2+(m_4\sin(\theta_2+\theta_3+\theta_4))/200+(m_5\sin(\theta_2+\theta_3+\theta_4))/100)-\dot{\theta}_4((m_5\sin(\theta_1+\theta_2+\theta_3+\theta_4))/800+(m_5\sin(\theta_1-\theta_2-\theta_3-\theta_4))/800)-\dot{\theta}_2((m_5\sin(2\theta_1))/800+(m_5\sin(\theta_1+\theta_2+\theta_3+\theta_4))/400+(m_5\sin(\theta_1+\theta_2))/800+(m_5\sin(\theta_1-\theta_2-\theta_3-\theta_4))/400+I_{5,xy}\cos(2\theta_1)+(m_5\sin(\theta_1-\theta_2))/800-(I_{5,xx}\sin(2\theta_1))/2+(I_{5,yy}\sin(2\theta_1))/2)-\dot{\theta}_3((m_5\sin(2\theta_1))/800+(m_5\sin(\theta_1+\theta_2+\theta_3+\theta_4))/400+(m_5\sin(\theta_1-\theta_2-\theta_3-\theta_4))/400+I_{5,xy}\cos(2\theta_1)-(I_{5,xx}\sin(2\theta_1))/2+(I_{5,yy}\sin(2\theta_1))/2)]$$

$$C(3,2)=[\dot{\theta}_2((m_4\cos(\theta_3+\theta_4))/100-(m_5\sin(\theta_1+\theta_2+\theta_3+\theta_4))/200-(m_5\sin(\theta_1+\theta_2))/400+(m_4\sin(\theta_3+\theta_4))/200+(m_5\sin(\theta_1-\theta_2-\theta_3-\theta_4))/200-(m_3\cos(\theta_3))/200+(m_5\sin(\theta_1-\theta_2))/400-(m_5\cos(\theta_2+\theta_3+\theta_4)(\sin(\theta_2+\theta_3+\theta_4)/10+\sin(\theta_2)/10))/10+(m_5\sin(\theta_2+\theta_3+\theta_4)(\cos(\theta_2+\theta_3+\theta_4)/10+\cos(\theta_1)/20+\cos(\theta_2)/10))/10)-\dot{\theta}_4((m_5\sin(\theta_1+\theta_2+\theta_3+\theta_4))/400-(m_5\sin(\theta_1-\theta_2-\theta_3-\theta_4))/400)-\dot{\theta}_1((m_5\sin(2\theta_1))/800+(m_5\sin(\theta_1+\theta_2+\theta_3+\theta_4))/400+(m_5\sin(\theta_1+\theta_2))/800+(m_5\sin(\theta_1-\theta_2-\theta_3-\theta_4))/400+I_{5,xy}\cos(2\theta_1)+(m_5\sin(\theta_1-\theta_2))/800-(I_{5,xx}\sin(2\theta_1))/2+(I_{5,yy}\sin(2\theta_1))/2)-\dot{\theta}_3((m_5\sin(\theta_1+\theta_2+\theta_3+\theta_4))/400-(m_5\sin(\theta_1-\theta_2-\theta_3-\theta_4))/400)]$$

$$C(3,3)=[-\dot{\theta}_2((m_5\sin(\theta_1+\theta_2+\theta_3+\theta_4))/400-(m_5\sin(\theta_1-\theta_2-\theta_3-\theta_4))/400)-\dot{\theta}_3((m_5\sin(\theta_1+\theta_2+\theta_3+\theta_4))/400-(m_5\sin(\theta_1-\theta_2-\theta_3-\theta_4))/400)-\dot{\theta}_4((m_5\sin(\theta_1+\theta_2+\theta_3+\theta_4))/400-(m_5\sin(\theta_1-\theta_2-\theta_3-\theta_4))/400)-\dot{\theta}_1((m_5\sin(2\theta_1))/800+(m_5\sin(\theta_1+\theta_2+\theta_3+\theta_4))/400+(m_5\sin(\theta_1-\theta_2-\theta_3-\theta_4))/400+I_{5,xy}\cos(2\theta_1)-(I_{5,xx}\sin(2\theta_1))/2+(I_{5,yy}\sin(2\theta_1))/2)]$$

$$C(3,4)=[-\dot{\theta}_2((m_5\sin(\theta_1+\theta_2+\theta_3+\theta_4))/400-(m_5\sin(\theta_1-\theta_2-\theta_3-\theta_4))/400)-\dot{\theta}_3((m_5\sin(\theta_1+\theta_2+\theta_3+\theta_4))/400-(m_5\sin(\theta_1-\theta_2-\theta_3-\theta_4))/400)-\dot{\theta}_4((m_5\sin(\theta_1+\theta_2+\theta_3+\theta_4))/400-(m_5\sin(\theta_1-\theta_2-\theta_3-\theta_4))/400)-\dot{\theta}_1((m_5\sin(\theta_1+\theta_2+\theta_3+\theta_4))/800+(m_5\sin(\theta_1-\theta_2-\theta_3-\theta_4))/800)]$$

$$C(4,1)=[\dot{\theta}_1((m_4\cos(2\theta_2+2\theta_3+2\theta_4))/200-(3m_4\sin(2\theta_2+2\theta_3+2\theta_4))/800+(m_5\sin(2\theta_2+2\theta_3+2\theta_4))/200+(m_4\cos(2\theta_2+\theta_3+\theta_4))/200+(m_4\sin(2\theta_2+\theta_3+\theta_4))/400+(m_5\sin(2\theta_2+\theta_3+\theta_4))/200+(m_4\cos(\theta_3+\theta_4))/200+(m_4\sin(\theta_3+\theta_4))/400+(m_5\sin(\theta_3+\theta_4))/200-I_{4,xy}\cos(2\theta_2+2\theta_3+2\theta_4)+$$

$(m_4 \cos(\theta_2+\theta_3+\theta_4))/100-(I_{4,xx} \sin(2\theta_2+2\theta_3+2\theta_4))/2+(I_{4,yy} \sin(2\theta_2+2\theta_3+2\theta_4))/2+(m_4 \sin(\theta_2+$
$\theta_3+\theta_4))/200+(m_5 \sin(\theta_2+\theta_3+\theta_4))/100)-\dot{\theta}_3((m_5 \sin(\theta_1+\theta_2+\theta_3+\theta_4))/800+(m_5 \sin(\theta_1-\theta_2-\theta_3-$
$\theta_4))/800)-\dot{\theta}_2((m_5 \sin(\theta_1+\theta_2+\theta_3+\theta_4))/800+(m_5 \sin(\theta_1-\theta_2-\theta_3-\theta_4))/800)]$

$C(4,2) = [\dot{\theta}_2((m_4 \cos(\theta_3+\theta_4))/100-(m_5 \sin(\theta_1+\theta_2+\theta_3+\theta_4))/400+(m_4 \sin(\theta_3+\theta_4))/200+$
$(m_5 \sin(\theta_1-\theta_2-\theta_3-\theta_4))/400-(m_5 \cos(\theta_2+\theta_3+\theta_4)(\sin(\theta_2+\theta_3+\theta_4)/10+\sin(\theta_2)/10))/10+$
$(m_5 \sin(\theta_2+\theta_3+\theta_4)(\cos(\theta_2+\theta_3+\theta_4)/10+\cos(\theta_1)/20+\cos(\theta_2)/10))/10)-\dot{\theta}_1((m_5 \sin(\theta_1+\theta_2+\theta_3+$
$\theta_4))/800+(m_5 \sin(\theta_1-\theta_2-\theta_3-\theta_4))/800)]$

$C(4,3) = [-\dot{\theta}_1((m_5 \sin(\theta_1+\theta_2+\theta_3+\theta_4))/800+(m_5 \sin(\theta_1-\theta_2-\theta_3-\theta_4))/800)]$

$C(4,4) = [0]$

A.6 Larus Potential Energy Gradient

$\frac{\partial V}{\partial \mathbf{q}}(1) = [(g \cdot m_1 \cos(\theta_1))/10+(g \cdot m_2 \cos(\theta_1))/10+(g \cdot m_3 \cos(\theta_1))/10+(g \cdot m_4 \cos(\theta_1))/10]$

$\frac{\partial V}{\partial \mathbf{q}}(2) = [(981 m_5(\cos(\theta_2)/10+(3\cos(\theta_4)(\cos(\theta_2)\cos(\theta_3)-\sin(\theta_2)\sin(\theta_3)))/20-$
$(3\sin(\theta_4)(\cos(\theta_2)\sin(\theta_3)+\cos(\theta_3)\sin(\theta_2)))/20))/100-g \cdot m_2(\cos(\theta_2)/400-\cos(\theta_2)^2/5$
$+\sin(\theta_2)^2/5)+g \cdot m_3((\cos(\theta_2)\sin(\theta_3))/20+(\cos(\theta_3)\sin(\theta_2))/20+\cos(\theta_2)^2/5-\sin(\theta_2)^2/5)-$
$g \cdot m_4(\sin(\theta_2)^2/5-\cos(\theta_2)^2/5-(\cos(\theta_4)(\cos(\theta_4)(\cos(\theta_2)\cos(\theta_3)-\sin(\theta_2)\sin(\theta_3))-$
$\sin(\theta_4)(\cos(\theta_2)\sin(\theta_3)+\cos(\theta_3)\sin(\theta_2))))/10+(\sin(\theta_4)(\cos(\theta_4)(\cos(\theta_2)\sin(\theta_3)+\cos(\theta_3)\sin(\theta_2))+$
$\sin(\theta_4)(\cos(\theta_2)\cos(\theta_3)-\sin(\theta_2)\sin(\theta_3))))/10+(\cos(\theta_4)(\cos(\theta_2)\sin(\theta_3)+\cos(\theta_3)\sin(\theta_2)))/10+$
$(\cos(\theta_4)(\cos(\theta_2)\cos(\theta_3)-\sin(\theta_2)\sin(\theta_3)))/20-(\sin(\theta_4)(\cos(\theta_2)\sin(\theta_3)+\cos(\theta_3)\sin(\theta_2)))/20+$
$(\sin(\theta_4)(\cos(\theta_2)\cos(\theta_3)-\sin(\theta_2)\sin(\theta_3)))/10)]$

$\frac{\partial V}{\partial \mathbf{q}}(3) = [(981 m_5((3\cos(\theta_4)(\cos(\theta_2)\cos(\theta_3)-\sin(\theta_2)\sin(\theta_3)))/20-(3\sin(\theta_4)(\cos(\theta_2)\sin(\theta_3)+$
$\cos(\theta_3)\sin(\theta_2)))/20))/100-g \cdot m_4((\sin(\theta_4)(\cos(\theta_4)(\cos(\theta_2)\sin(\theta_3)+\cos(\theta_3)\sin(\theta_2))+$
$\sin(\theta_4)(\cos(\theta_2)\cos(\theta_3)-\sin(\theta_2)\sin(\theta_3))))/10-(\cos(\theta_4)(\cos(\theta_4)(\cos(\theta_2)\cos(\theta_3)-\sin(\theta_2)\sin(\theta_3))-$
$\sin(\theta_4)(\cos(\theta_2)\sin(\theta_3)+\cos(\theta_3)\sin(\theta_2))))/10+(\cos(\theta_4)(\cos(\theta_2)\sin(\theta_3)+\cos(\theta_3)\sin(\theta_2)))/10+$
$(\cos(\theta_4)(\cos(\theta_2)\cos(\theta_3)-\sin(\theta_2)\sin(\theta_3)))/20-(\sin(\theta_4)(\cos(\theta_2)\sin(\theta_3)+\cos(\theta_3)\sin(\theta_2)))/20+$
$(\sin(\theta_4)(\cos(\theta_2)\cos(\theta_3)-\sin(\theta_2)\sin(\theta_3)))/10)+g \cdot m_3((\cos(\theta_2)\sin(\theta_3))/20+(\cos(\theta_3)\sin(\theta_2))/20)]$

$\frac{\partial V}{\partial \mathbf{q}}(4) = [(981 m_5((3\cos(\theta_4)(\cos(\theta_2)\cos(\theta_3)-\sin(\theta_2)\sin(\theta_3)))/20-(3\sin(\theta_4)(\cos(\theta_2)\sin(\theta_3)+$
$\cos(\theta_3)\sin(\theta_2)))/20))/100-g \cdot m_4((\sin(\theta_4)(\cos(\theta_4)(\cos(\theta_2)\sin(\theta_3)+\cos(\theta_3)\sin(\theta_2))+$
$\sin(\theta_4)(\cos(\theta_2)\cos(\theta_3)-\sin(\theta_2)\sin(\theta_3))))/5-(\cos(\theta_4)(\cos(\theta_4)(\cos(\theta_2)\cos(\theta_3)-\sin(\theta_2)\sin(\theta_3))-$
$\sin(\theta_4)(\cos(\theta_2)\sin(\theta_3)+\cos(\theta_3)\sin(\theta_2))))/5+(\cos(\theta_4)(\cos(\theta_2)\sin(\theta_3)+\cos(\theta_3)\sin(\theta_2)))/10+$
$(\cos(\theta_4)(\cos(\theta_2)\cos(\theta_3)-\sin(\theta_2)\sin(\theta_3)))/20-(\sin(\theta_4)(\cos(\theta_2)\sin(\theta_3)+\cos(\theta_3)\sin(\theta_2)))/20+$
$(\sin(\theta_4)(\cos(\theta_2)\cos(\theta_3)-\sin(\theta_2)\sin(\theta_3)))/10)]$