Modelling and Control of Airship with Slung Payload

by

Osama Obeid

Thesis submitted to the University of Ottawa
in partial Fulfillment of the requirements for the
Ph.D. degree in Mechanical Engineering

Department of Mechanical Engineering
Faculty of Engineering
University of Ottawa

Abstract

This study explores the modelling and control of a multibody system comprising an airship, gondola, and a slung payload. Lighter-than-air vehicles undergo inertial forces that are often neglected in heavier-than-air vehicles. These inertial forces are modelled using added mass and added inertia and there can be significant discrepancies between the values obtained empirically and those of the actual vehicle. The dynamics of the multibody system were first modelled using the Udwadia-Kalaba method. The resulting equation of motion was used to identify the added mass, added inertia, and inertia of the airship through system identification procedure. The proposed system identification method utilizes semidefinite programming with equality and inequality constraints to find any unknown parameters in the mass matrix of the multibody system. Three experiments were carried out to perform the system identification and validate the dynamic model. A comparison of reconstructed trajectories before and after applying system identification shows that the identified mass matrix produces more accurate results with 35% lower root mean squared error of position when compared with the trajectories simulated before carrying out system identification. Aerodynamic coefficients, including lift and drag coefficients, were calculated for a full-scale airship prototype using the Reynold's averaged Navier-Stokes with Spalart-Allmaras turbulence model.

Using the nonlinear dynamic model of the multibody system, two fuzzy logic controllers were developed to attenuate the payload's oscillations and maintain the payload at a desired position. Two more fuzzy logic controllers were designed to navigate the airship-gondola-slung-payload system in the longitudinal plane to a target location. An additional fuzzy logic controller was developed to deliver the payload by means of controlling the altitude. The proposed control method addresses a gap in the literature, which lacks experimental studies on airships with slung payloads using fuzzy logic control. The controller was evaluated under wind disturbance through simulations and in outdoor experiments. Despite the adverse weather conditions, the navigation fuzzy logic controller operated as intended, effectively responding to inputs and attempting to correct errors while adhering to the upper and lower bounds of the thrusters.

Acknowledgements

I would like to express my heartfelt gratitude to my supervisor Dr.Eric Lanteigne for his support and guidance throughout my journey.

I would also like to extend my sincere appreciation to my dear friends Gilmar, Ahmed, Nourah, and Eleni for the wonderful moments we shared in the lab. Those experiences will always hold a special place in my memory. I also want to thank Mahi for the time we spent together, which I truly enjoyed.

The author acknowledges the computer time and technical support provided by Niagara (www.scinethpc.ca/niagara/), Cedar (www.sfu.ca/research/supercomputer-cedar), Beluga (www.calculquebec.ca/), Graham (www.uwaterloo.ca), and Compute Canada.

Dedication

This thesis is dedicated to my family, especially my parents, whose endless support and love have been my greatest strength. I am forever grateful for their unwavering encouragement.

Contents

List of Tables

List of Figures

List of Symbols

Dynamics

c	Number of constraints
c_h	Holonomic constraint
c_{nh}	Nonholonomic constraint
$d_{VG,x}, d_{VG,z}$	Offset from CV to CG along x and z directions, respectively
$\mathbf{d_{VG}}$	Skew-symmetric matrix containing the offset of CV to CG
F_x, F_y, F_z	Force components along x, y, and z directions, respectively
$\mathbf{I}$	Symmetric inertia matrix (tensor)
$\mathbf{I_{all}}$	Skew-symmetric inertia matrix containing added inertia
i, j, k, ℓ, r	Indices
k'	Added inertia coefficient
k_1, k_2	Added mass coefficients
λ	Godola rail angle, on the curved section
L	Lagrangian
L, M, N	Moment components along x, y, and z directions, respectively
$\mathbf{M}$	Skew-symmetric mass matrix
$\mathbf{M_{all}}$	Mass matrix containing added mass coefficients
m	Mass
n	Number of degrees of freedom
p, q, r	Angular velocity components
$T, \overline{T}$	Kinetic energy

U	Potential energy
$\vec{A}$	Aerodynamic forces
$\vec{D}$	Dynamic forces
$\vec{F}$	External forces
$\vec{M}$	Moments (torque)
$\vec{Q}$	Generalized external forces and moments
$R^{\vec{P}_k}$	Resultant force at point P_k
$T^{\vec{B}_\ell}$	Resultant torque at body B_ℓ
$\vec{\Omega}$	Angular acceleration
$\vec{\omega}, \vec{V}_2$	Angular velocity
$\vec{\nu}$	Vector containing linear and angular velocity
$\vec{a}$	Linear Acceleration
$\vec{q}$	Generalized coordinates
$\vec{r}_{CG}$	Position vector from the origin, in the inertial frame, to the centre of gravity
$\vec{r}_{CV}$	Position vector from the origin, in the inertial frame, to the centre of volume
$\vec{r}_{VG}$	Position vector from the centre of volume to the centre of gravity
$\vec{V}, \vec{V}_1$	Linear velocity
u, v, w	Linear velocity components
u_r	Generalized speeds/ quasi-velocities
$\mathbb{V}$	Volume
v_r, ω_r	Partial linear and angular velocity
X^B, Y^B, Z^B	Coordinates in body frame
X^E, Y^E, Z^E	Coordinates in inertial frame
ϕ, θ, ψ	Euler angles
$\boldsymbol{\alpha}, \boldsymbol{\beta}$	Tunable feedback gain matrices for UK method
α_i, β_i	Tunable feedback gains elements for UK method
$\hat{i}, \hat{j}, \hat{k}$	Unit vectors along X^B, Y^B, Z^B
Ss	Distance travelled along the gondola rail

Aerodynamics

A	Reference area
A_L, A_M, A_N	Aerodynamic moments along X^B, Y^B, and Z^B directions, respectively
A_x, A_y, A_z	Aerodynamic forces along X^B, Y^B, and Z^B directions, respectively
α	Angle of attack
β	Side-slip angle
C_D	Coefficient of drag
$C_{D_{(\alpha=0)}}$	Coefficient of drag at zero angle of attack
C_{DC}	Cross-flow drag coefficient
C_L	Coefficient of lift
C_L, C_M, C_N	Aerodynamic moment coefficients along X^B, Y^B, and Z^B directions, respectively
$C_{n\alpha}$	Derivative of normal force with respect to the angle of attack
C_X, C_Y, C_Z	Aerodynamic force coefficients along X^B, Y^B, and Z^B directions, respectively
$(C_t)_F$	Leading-edge suction coefficient of fins
D	Diameter of the hull or envelope
l	Length of the hull or envelope
$R(\xi)$	Radius of local cross-section
ρ	Density
ν	Kinematic viscosity
p	Pressure
q_0	Dynamic pressure $= 1/2\rho U_\infty^2$
ζ	Characteristic length of the airship
ξ	Axial direction along the hull
ξ_1	Axial location along the hull where $\dfrac{dS}{d\xi}$ is minimum
ξ_v	Axial location along the hull where potential flow assumption is invalid

S	Hull cross-section area
S_F	Fin reference area
S_h	Hull reference area
$\delta_{\cdot}$	Angles of rudder or elevator, depending on subscripts
$\eta_{\cdot}$	Correction factor for finite hull (or fin, depending on the subscript)
ℓ_h	Distance from nose to hull-fin intersection point
U_∞	Freestream velocity
Re_D	Reynold's number based on the base diameter
Re_l	Reynold's number based on the hull's length
$\delta_{\cdot}$	Angles of rudder or elevator, depending on subscripts

System Identification

c	Predetermined constant value
d	Distance travelled by the airship throughout the experiment
Δ	Mass matrix convergence
ϵ	Iteration tolerance
$\vec{e}$	Error tolerance vector
$\ddot{\vec{q}}_{exp}$	Acceleration obtained from experiment
$\hat{\mathbf{M}}$	Mass matrix inverse
$\tilde{\hat{\mathbf{M}}}$	Flattened mass matrix inverse, with zero-value elements removed
$\mathbf{K}$	Matrix contains pseudo-inverse term
$\mathbf{M_{CAD}}$	Mass matrix as approximated by CAD
$\tilde{\mathbf{M}}_{\mathbf{CAD}}$	Flattened mass matrix as approximated by CAD, with zero-value elements removed
$\mathbf{M_{exp}}$	Mass matrix obtained from experiment
$\tilde{\mathbf{M}}_{\mathbf{exp}}$	Flattened mass matrix obtained from experiment, with zero-value elements removed
$\vec{r}_{force}$	Location where force is applied in the body frame
$MM(\mathbf{1})$	Mass matrix obtained from experiment 1
$MM(\mathbf{2})$	Mass matrix obtained from experiment 2
$MM(\mathbf{3})$	Mass matrix obtained from experiment 3
$NRMSE$	Normalized root mean squared error
$RMSE$	Root mean squared error

Control

CG_i	Distance from origin to the centre of gravity of region i
E	Error
$E_{\|r\|}$	Error in distance
E_{ϕ_c}	Error in slung payload angle about the axial axis
E_{ψ_B}	Error in yaw angle of blimp
E_{θ_c}	Error in slung payload angle about the lateral axis
E_{x_L}	Error in payload position in the axial direction
E_{y_L}	Error in payload position in the lateral direction
E_{χ_i}	Input
$E_{\chi_{ref}}$	Reference input
$\dot{E}_{\|r\|}$	Rate of change of error in distance
$\dot{E}_{\phi_c}$	Rate of change of error in slung payload angle about the axial axis
$\dot{E}_{\psi_B}$	Rate of change of blimp's yaw angle error
$\dot{E}_{\theta_c}$	Rate of change of error in slung payload angle about the lateral axis
$\dot{E}_{x_L}$	Rate of change of error in the payload's axial direction
$\dot{E}_{y_L}$	Rate of change of error in the payload's lateral direction
J_i	Cost function i
N	Negative membership function
NN	Negative-Negative membership function
NNN	Negative-Negative-Negative membership function
$NNNN$	Negative-Negative-Negative-Negative membership function
P	Positive membership function
PP	Positive-Positive membership function
PPP	Positive-Positive-Positive membership function
$PPPP$	Positive-Positive-Positive-Positive membership function
μ	Membership degree
u_x	Force in the axial direction
u_y	Force in the lateral direction

u_z	Force in the vertical direction
u_{mz}	Yaw moment
Z	Zero membership function
χ_i	Input
χ_{ref}	Reference input

Chapter 1

Introduction

1.1 A Brief History

It is believed that buoyant-driven vehicles were conceptualized well before Archimedes constructed a physical law for the buoyancy force (more than 2300 years ago) [1]. The first documented successful attempts of human-carrying vehicle was in the 18th century by the Montgolfier brothers, using hot air balloons [2]. However, it took an additional century before engineer Henri Giffard was able to build a full-scale airship (which was referred to as a steerable balloon) in 1852 that was powered using a light weight steam engine [2, 3]. Various variations in the structural and power configurations as well as control systems have arose ever since.

Owing to their light weight properties, hot air, hydrogen, and helium have been commonly used in airships. Although hydrogen is the lightest known gas (with density of $\rho_{hydrogen} = 0.09$ kg/m^3), its superior lifting capability is compromised by its flammability when mixed with air. Two accidents took place when the *Roma* and *Hindenburg* airships caught on fire in 1922 and 1937, respectively. As a result, hydrogen has been banned and replaced by helium which is the second lightest gas ($\rho_{helium} = 0.179$ kg/m^3). Despite the controversy about the reasons for banning hydrogen, nowadays most aviation regulations prohibit the use of a flammable gas in airships such as Federal Aviation Administration (FAA-4.48) and Transport Airship Requirements (TAR-893). In addition, the Canadian

Aviation Regulations (CAR) explicitly states "Hydrogen is not an acceptable lifting gas for use in airships." (CAR-541.7). Lately, methods of obtaining an inflammable hydrogen, by means of adding deterrent compounds, have been proposed in a Russian patent [4] to circumvent this restriction. Since hydrogen is already used in powering many road vehicles, experts are calling for lifting the restrictions on hydrogen in applications to aviation [5]. The same article [5] also highlights the financial growth that airships could provide to the Canadian economy, namely the northern side and concludes with "It is time to end the fraudulently induced ban on the use of hydrogen gas in airships and set this technology free."

1.2 Recent Developments

After a long period of being relinquished, interest in airships has resurged owing to stringent $CO2$ emissions regulations and the increasing market demand on air cargo. A report published by Boeing [6] projected a significant increase in global air cargo demand. The forecast, based on data collected over an 11-year period, predicts a rise in Revenue Ton Kilometers (RTKs) from 256 billion in 2017 to 584 billion by 2037. As a result, the initiatives and research funding in the lighter-than-air (LTA) sector directed towards civil aviation has increased in recent years [7]. In 2018, the Quebec government invested $23 [8]-$30 [9] million in Flying Whales, a France-based company that builds helium airships. Similarly, companies like Solar Ship (Fig 1.1), OceanSky Cruises, Hybrid Air Vehicles, and Lockheed Martin are developing cargo airships that would substitute for freighter aircraft. The applications of airships are not limited to cargo transport. OceanSky Cruises and Flying Whale are also planning to carry out tour trips over areas that are difficult to reach by roads such as the Amazon or North pole. Unlike airplanes, the cabin of the airship is non-pressurized which allows installing large windows or even glass floors resulting in better tour experience for passengers.

The future of civil aviation demands efficient and sustainable solutions for cargo transfer. With the increase of $CO2$ pollution, the carbon taxes will inevitably increase. The

Figure 1.1: Ongoing development of cargo airship in the market. Left: Wolverine [10], Right: Caracal [11]. Images used with permission from Solar Ship.

cargo airship could potentially replace the jet freighter aircraft due to their lower fuel consumption, capabilities to takeoff and land in areas with little manoeuvre infrastructures, and significantly reduce $CO2$ emissions compared to fixed-wing aircraft.
In addition to the capability of carrying heavy loads, airships with underneath slung-load offer a major advantage in terms of having less restrictions to the shape and size of the carried load. The slung-load can vary in scale from timber logs to wind turbine blades. Moreover, there are generally fewer airport restrictions on external load carriage compared to in-cabin air transportation. Among companies that are developing cargo airships, Flying Whales' cargo airship design is aiming to carry the load externally.

1.3 Uninhabited Airship Prototype

Owing to their ability to loiter for significant periods at a relatively lower cost, maintain longer flight duration, land within a shorter distance, and have lower noise levels compared to airplanes, airships are favorable for numerous applications such as reconnaissance, patrol, and logistic transportation [1, 12–14]. While the payload carrying capacity is approximately proportional to the cube of its linear dimensions, smaller uninhabited airships provide a convenient way to conduct experiments and simulations at a low cost.
The components of the investigated prototype are described in this chapter. The physical prototype, illustrated in Fig. 1.2, consists of a helium envelope, a gondola, and a slung-

payload. The envelope shape was optimized for rapid descent [15]. The gondola, made out of carbon fibre, is connected to the rail (keel) of the envelope by two motors (Fig. 1.3a). The motors allow the gondola to move along the keel which changes the centre of gravity (CG) of the multibody system and yields a pitching moment. Also, the gondola contains all the electronics needed to operate the airship. The electronics are discussed in the following section. In addition, four thrusters, two positioned on the sides of the gondola and two on the front and back, portrayed in Fig. 1.3b,. The gondola carries a spherical payload suspended by a tether. The weight and dimensions of each component is given in Table 1.1. It is worth noting that blimp and airship are used interchangeably throughout the thesis.

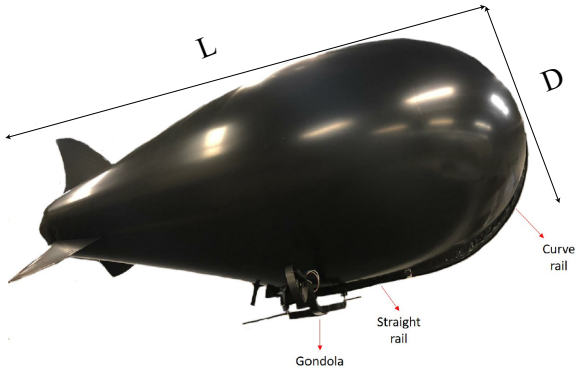
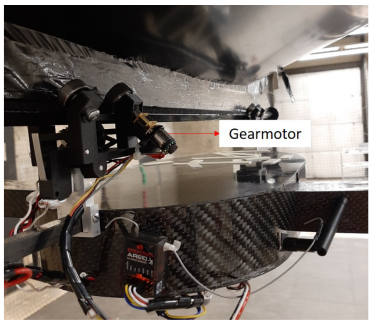
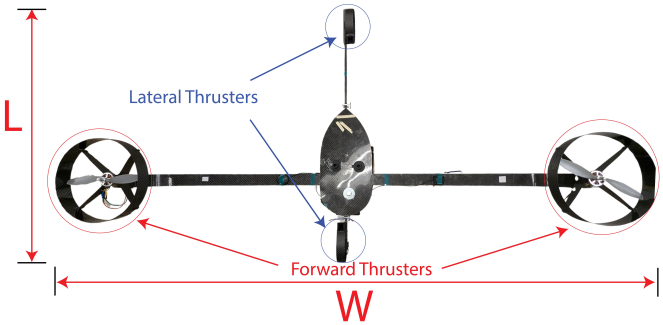


Figure 1.2: Multibody airship configuration [16].



(a) Gondola connected to the rail by motors.



(b) Gondola and thrusters. Forward and lateral thrusters are circled in red and blue, respectively.

Figure 1.3: Gondola configuration.

Table 1.1: Components of airship-gondola-payload prototype.

Component	Dimensions [m]	Weight [kg]
Blimp	$L = 4.1$ $D = 1.82$	2.66
Gondola (excluding batteries)	$L = 1$ $W = 1.95$	2.55
Tether	$L = 4$	< 0.001
Payload	$D = 0.7$	1.25
Battery	$L = 0.155$ $W = 0.048$	0.3

The airship is controlled using a Raspberry Pi 4 (RPI) with a Navio 2 Emlid Hat. The Navio 2 is a dedicated hardware developed to control uninhabited vehicles. The hardware comes with measurement sensors such as an Inertial Measurement Unit (IMU), accelerometer, and Global Positioning System (GPS) along with the necessary software packages to retrieve and integrate those measurements. An additional GPS module was added to eliminate drifting. Moreover, an optical encoder was installed to measure the payload angle. The encoder was installed on a custom designed mount, shown in Fig. 1.4, that holds the tethered payload.

MAVLink protocol is used to send Pulse-Width-Modulation (PWM) signals to the actuators. To communicate with the Navio 2 from a ground station, an Arducopter flight controller is utilized. The ground station consists of a remote controller (RC), connected to pin 0 (PPM/SB), a personal computer, and a motion capture system. Thrusters are driven by brushless motors through electronic speed controllers (ESCs). The sum of the forces produced by the forward thrusters is over 4 N while the lateral thrusters can produce up to 2 N. All thrusters are connected to reversible ESCs, allowing them to rotate in both directions. Furthermore, the gondola is powered by two brushed DC motors which are driven by a dual motor controller.

Four 14.8V Lithium–Polymer (LiPo) batteries are connected in parallel and used to power

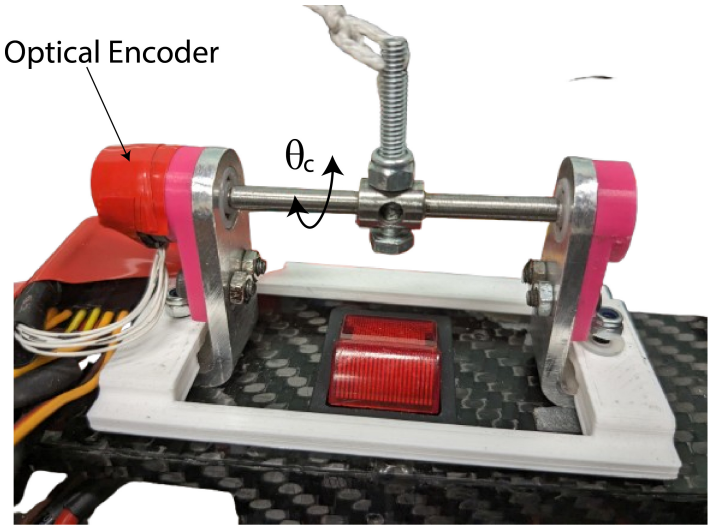


Figure 1.4: Payload mount with optical encoder.

the system. Figure 1.5 illustrates the power line (in red), communication line (in green), and actuation line (in blue) of the system's components.

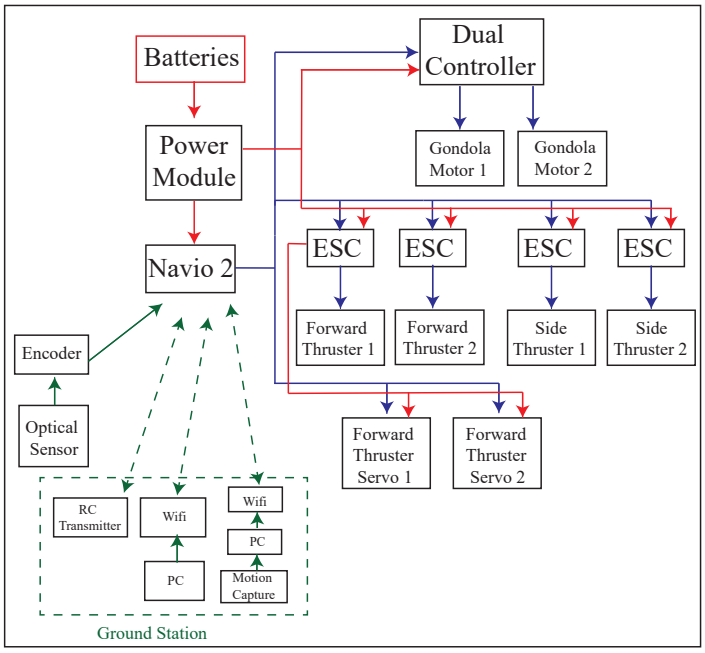


Figure 1.5: Electronics scheme. Power, communication, and actuation connections are portrayed in red, green, and blue, respectively. The dashed lines represent wireless connections.

1.4 Problems to Address and Objectives

The focus of this research is the modelling, simulation, and control of a 4.1 m long uninhabited airship with a slung payload. The physical model of the investigated uninhabited airship is adopted from Lanteigne *et al.* [17] who developed a multibody reconfigurable airship, described in Section 1.3. This design provides a mechanism to improve the manoeuvrability of airships. During the unloading phase, the airship loses weight which causes the airship to go into an uncontrollable state [18]. This issue is circumvented in the current design by streamlining the envelope design to facilitate landing when unloading.

Nevertheless, the current design of uninhabited airship, developed in [17], is still in the early stages of development. The previous dynamic model is constructed in two-dimensions. Hence, the model was expanded to three-dimensions and a slung payload was included. The aerodynamic models found in the literature are based on semi-empirical studies carried out on conventional airship bodies. Since the investigated airship body is unconventional, computational fluid dynamics (CFD) simulations were necessary for obtaining accurate aerodynamic coefficients. Moreover, trajectory tracking controllers that control the airship and account for payload swinging were developed.
To this end, the contributions of this thesis are as follows:

1. Developed a dynamic model for the airship multibody system with holonomic and nonholonomic constraints. The model effectively addressed complex actuation-dependent constraints and provided a streamlined approach for integrating them in multibody systems. Moreover, a numerical investigation was carried out to estimate the aerodynamic coefficients of the airship prototype using CFD techniques with turbulence models. The aerodynamic coefficients obtained provided valuable insights that can be utilized to enhance the design of the thrusters.

2. Developed a system identification method that estimates unknown parameters in the mass matrix of lighter-than-air vehicles, such as added mass and inertia. The proposed method employs semi-definite programming to minimize the error between the model-predicted acceleration and the experimentally obtained acceleration. Al-

though the experiments were performed in an open-loop configuration with no control actions from the thrusters, the proposed method for system identification remains applicable even in the scenario where thrusters are employed, provided that the force vector generated by the thrusters is measured in real time. Consequently, this system identification approach can be applied to a range of vehicles, including airships and underwater vehicles, when parameters in the mass matrix are unknown.

3. Developed fuzzy logic controllers for the uninhabited airship to achieve planar flight while effectively stabilizing the payload oscillations under windy conditions.

4. The airship-slung payload cargo transfer was validated by carrying out experimental tests on a prototype of the uninhabited airship. The planar flight controller was implemented on the onboard hardware and successfully tested outdoors, demonstrating feasibility.

1.5 Thesis Organization

To achieve these objectives, a review of significant developments to date on dynamic modelling (Chapter 2), system identification (Chapter 3), aerodynamics (Chapter 4), and control strategies of aircraft with slung payload (Chapter 5) is presented. This is followed by a summary table to highlight and compare the characteristics of each discussed approach. The final chapter (Chapter 6) provides a discussion on the key contributions of this thesis, along with directions for future work. It is important to emphasize that the Appendices provide supplemental content for each chapter while also serving as an integral part of the thesis.

Chapter 2

Dynamic Modelling

2.1 Background and Literature Review

Since the majority of control systems rely on precise modelling, a review of available airship models is necessary. The flight model is a mathematical description of the forces and moments (from a Newton-Euler's perspective) or total kinetic and potential energies (from a Lagrange/Hamilton's perspective) acting on the aircraft. The outcome of these models is a set of equations of motion (EOM) which are represented as Ordinary Differentiation Equations (ODEs) or Partial Differential Equations (PDEs) depending on the complexity of the system. Model fidelity is affected by parameters uncertainties, noisy measurements, neglected dynamics, and unmodelled external forces.

While dynamic models can be formulated using various techniques, this thesis will focus on five approaches: Newton-Euler (NE), Euler-Lagrange (EL), Kane, Boltzmann-Hamel (BH), and Udwadia-Kalaba (UK). The fundamental principles of each method are discussed with a particular focus on airships, followed by a table comparing all presented methods.

Newton-Euler (NE)

The Newton-Euler method is based on Newton's second law which defines the forces (F) as the product of mass (m) and acceleration (a). The following formulation is known as Euler's first and second axioms.

$$\begin{aligned} \dot{\vec{p}}_{CG} &\equiv \vec{F}_{CG} = m\ \dot{\vec{v}}_{CG} \\ \dot{\vec{h}}_{CG} &\equiv \vec{M}_{CG} = I_{CG}\ \dot{\vec{\omega}} \end{aligned} \tag{2.1}$$

where subscript CG refers to the centre of gravity of the body, p_{CG} and h_{CG} are the linear and angular momentum, respectively, M_{CG} is the moment at the body centre of gravity, I_{CG} is the moment of inertia tensor about the centre of gravity. The translational and angular velocity vectors are represented by $\vec{v_{CG}}$ and $\vec{\omega}$ respectively. The formulation requires using the entire degrees of freedom of the systems' components. Explicit expressions of constraint forces are ought to be derived when NE is used. As a result, this method becomes laborious when the system consists of multiple components. For further details about types of constraints, refer to Appendix A.7. For detailed derivation of the NE equations along with an example, readers are referred to Appendix A.

Euler-Lagrange (EL)

The Euler-Lagrange (EL) formulation is based on the mechanical energy of the system, *i.e.*, potential and kinetic energy. Using the energy in the EL enforces constraints on the configuration (*i.e.,* degrees of freedom) in an implicit manner. The general form of the Lagrangian equation is

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\vec{q}}}\right) - \frac{\partial L}{\partial \vec{q}} = \vec{Q} \tag{2.2}$$

where $\vec{Q}$ is the vector of external forces and torque acting on the systems in the generalized coordinates, L is the Lagrangian and it is defined as

$$L(\dot{\vec{q}}, \vec{q}) = T(\dot{\vec{q}}, \vec{q}) - U(\vec{q})$$

where T and U are the kinetic and potential energy, respectively, and $\vec{q}$ is the vector of generalized coordinates.

The holonomic constraints are satisfied spontaneously in the EL formulation without any further manipulations. Nevertheless, including nonholonomic constraints is non-trivial and it is often done using Lagrange multipliers [19], which involves the introduction of auxiliary variables.

To demonstrate how the EL accommodates the holonomic constraints implicitly, the following sample example is solved using the NE and the EL techniques. Figure 2.1 shows a pendulum connected to a massless rigid rod. The only forces acting on the red ball is

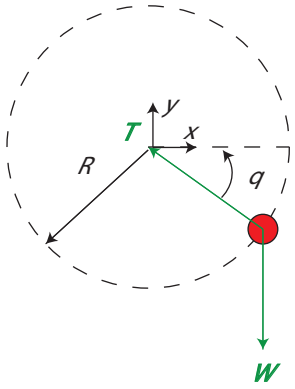


Figure 2.1: Sample problem of pendulum connected to a rigid rod, illustrating holonomic constraints. Forces are depicted in green arrows

weight and tension, where tension being the holonomic constraint in this problem. To use the EL, the generalized coordinates are ought to be determined. Clearly, there is only one degree of freedom (*i.e.,* q). To find the EOM, Equation (2.2) is used which yields (2.3)

$$\begin{aligned} L &= 0.5\ mR^2\dot{q}^2 - mgR\sin(q) \\ \frac{\partial L}{\partial \dot{q}} &= mR^2\dot{q} \\ \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial L}{\partial \dot{q}}\right) &= mR^2\,\ddot{q} \\ \frac{\partial L}{\partial q} &= -mgR\cos(q) \end{aligned} \tag{2.3}$$

Substituting (2.3) back into (2.2) gives

$$mR^2\,\ddot{q} + mgR\cos(q) = 0 \tag{2.4}$$

It is worth noting that the constraint force (i.e., tension) does not appear in (2.4).

Solving the same problem using the NE gives

$$m(-R\cos(q)\dot{q}^2 - R\sin(q)\,\ddot{q}) = -T\cos(q) \tag{2.5}$$

$$m(R\sin(q)\dot{q}^2 - R\cos(q)\,\ddot{q}) = T\sin(q) - mg \tag{2.6}$$

where the tension force (T) appears explicitly in the EOM, which requires further manipulations. Isolating T from (2.5) results in (2.7).

$$T = \frac{m}{\cos(q)}(R\cos(q)\dot{q}^2 + R\sin(q)\,\ddot{q}) \tag{2.7}$$

Substitute (2.7) into (2.6) and using some trigonometric identities, the EOM derived using the NE becomes identical to that derived earlier using the EL (2.4). Further remarks about the EL are provided in Appendix A.2.

Kane's method

Kane's method [20] is based on D'Alembert's principle where generalized coordinates and generalized speeds are used to construct the EOM. The method is simpler than NE and EL for deriving the EOM of multibody systems in vectorial form and this has numerical advantages [21, 22]. Decoupled EOM, in the space of generalized speeds derivatives ($\dot{u}_r$), are often attainable by judicious choice of generalized speeds [23]. Nonetheless, deriving a symbolic expression of the EOM is still laborious for complicated systems [21].
This method is used extensively in the literature for modelling of multibody systems, such as helicopters with underslung loads [21], shipboard operations of UAV helicopters [22], or robotic arms [24].

Unlike generalized velocities ($\dot{q}$), which are used in the EL approach, the generalized speeds (u_r) are often defined as a function of generalized coordinates (q) and generalized velocities ($\dot{q}$) [1]. As a result, properly choosing generalized speeds allows for motion constraints (nonholonomic constraints) to be encapsulated in the EOM. On the other hand,

[1] In dynamics literature, this is also known as quasi-velocities [25]

these constraints are imposed as separate equations in the NE method or as Lagrange multipliers in the EL method [19]. The derivation of Kane's method is provided in Appendix A.3.

Boltzmann-Hamel (BH)

Although it is less commonly discussed in the literature, the Boltzmann-Hamel (BH) approach is used to construct EOM for multibody systems involving nonholonomic mechanism. The method is derived from the D'Alembert Lagrange principle by assuming a linear structure for the quasi-velocity variables with respect to the generalized coordinates and generalized velocity variables. The latter assumption for quasi-velocity structure is similar to that used in Kane's method for generalized speeds. A detailed derivation is provided in Appendix A. The BH was used to derive the EOM of a multibody UAV airship [17].
This approach can be laborious as it requires obtaining an expression for the Lagrangian. Nevertheless, it is favoured over the NE and the EL when nonholonomic constraints are present. The latter is justified as the holonomic and nonholonomic constraints can be accommodated conveniently in the BH method.

Udwadia-Kalaba (UK)

A simpler dynamic modelling method has been proposed by Udwadia and Kalaba [26]. The method uses Gauss's principle to obtain explicit EOM with enforced constraints. Owing to the ease of application and capability of satisfying nonhnonolonomic constraints, the UK method will be utilized to model the airship multibody system, thus, presented here. To derive the Udwadia-Kalaba (UK) equations, Newton's description of an unconstrained system is revisited (2.8).

$$\mathbf{M}\,\ddot{\vec{q}}_{\,u} = \vec{Q} \tag{2.8}$$

where $\mathbf{M} \in \mathbb{R}^{n\times n}$ is the mass positive definite matrix, n is the number of degrees of freedom, $\ddot{\vec{q}}_{\,u} \in \mathbb{R}^n$ is the unconstrained acceleration (based on generalized coordinates), and $\vec{Q}$ is the generalized forces.

In the UK equation, the constraint forces are augmented to Newton's equation such that equation (2.8) becomes

$$\mathbf{M}\,\ddot{\vec{q}} = \vec{Q} + \vec{Q}_c \tag{2.9}$$

where $\vec{Q}_c \in \mathbb{R}^n$ is the constraint force and $\ddot{\vec{q}}$ is the constrained acceleration. The constraint forces can be either holonomic or nonholonomic[2]. Assuming smooth constraints, *i.e.,* can be differentiated, the constraints are differentiated up to the acceleration level to get the following form

$$\mathbf{A}(\vec{q}, \dot{\vec{q}}, t)\,\ddot{\vec{q}} = \vec{b}(\vec{q}, \dot{\vec{q}}, t) \tag{2.10}$$

where $\mathbf{A} \in \mathbb{R}^{m \times n}$ is the Jacobian of the constraint equation and $\vec{b} \in \mathbb{R}^m$ contains all the terms that are not multiplied by the acceleration $\ddot{q}$. To derive the UK equation, the Gauss least constraint principle is used. The constrained acceleration can be obtained by minimizing the following quadratic cost over all possible accelerations $\ddot{q}_\delta$, subject to (2.10)

$$cost = (\ddot{\vec{q}}_\delta - \ddot{\vec{q}}_u)^T \mathbf{M} (\ddot{\vec{q}}_\delta - \ddot{\vec{q}}_u) \tag{2.11}$$

After various manipulations (refer to [26] for complete derivation), the constraint forces are given as

$$Q_c = \mathbf{M}^{1/2}\,(\mathbf{A}\,\mathbf{M}^{-1/2})^\dagger\,(\vec{b} - \mathbf{A}\,\ddot{\vec{q}}_u) \tag{2.12}$$

where the dagger superscript † is the Moore-Penrose pseudo inverse. Isolating $\ddot{q}_u$ from (2.8) and substituting back into (2.12)

$$Q_c = \mathbf{M}^{1/2}\,(\mathbf{A}\,\mathbf{M}^{-1/2})^\dagger\,(\vec{b} - \mathbf{A}\mathbf{M}^{-1}\vec{Q}) \tag{2.13}$$

Equation (2.13) is substituted into (2.9) and pre-multiplied by the inverse of the mass matrix ($\mathbf{M^{-1}}$) to obtain (2.14).

$$\ddot{\vec{q}} = \mathbf{M^{-1}}\,\vec{Q} + \mathbf{M}^{-\frac{1}{2}}\,(\mathbf{A}\,\mathbf{M}^{-\frac{1}{2}})^\dagger\,(\vec{b} - \mathbf{A}\,\mathbf{M^{-1}}\,\vec{Q}) \tag{2.14}$$

To demonstrate Gauss principle of least constraint, the following example is given. Figure 2.2 depicts a multibody system consisting of a blimp, a gondola, and a slung-payload. The focus will be on the payload, which is connected to the gondola through an inelastic

[2]Refer to Appendix A.7 for further details about types of constraints

cable. Therefore, the motion of the payload is constrained by the cable. Neglecting the aerodynamic and friction forces, the only forces acting on the payload is weight and tension (Q and Q_c, respectively). The least constraint principle states that the acceleration of the constrained system must be closest (in the $L2$ norm fashion) to the unconstrained acceleration vector while satisfying the constraints. This is illustrated in Fig. 2.2 where the blue solid circles indicate some of the possible slung-load positions (also shown as blue vectors q_δ). The black solid circle is the position of the unconstrained slung-load (also shown as black vectors q_u). The magenta line is the shortest distance between q_u and q_δ. Note that the acceleration of the constrained system (q) is obtained when a possible acceleration (q_δ) satisfies Gauss's least constraint principle, consequently, $q = q_\delta$.

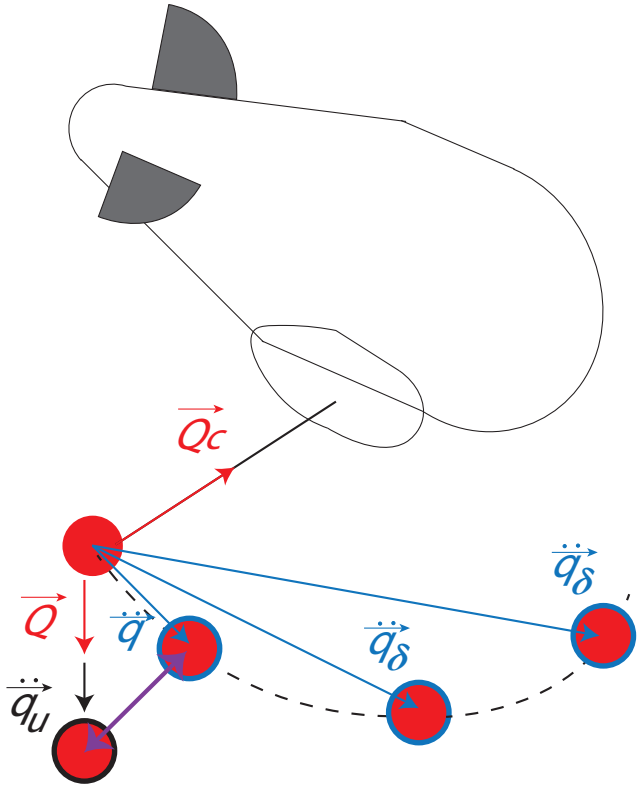


Figure 2.2: Gauss least constraint principle. Q is weight and Qc is the constraint force (*i.e.,* tension). The acceleration that satisfies Gauss least constraint principle is q whereas q_δ are possible accelerations, and q_u is the unconstrained acceleration.

Table 2.1: Comparison of dynamic modelling techniques. c_h and c_{nh} are the numbers of holonomic and nonholonomic constraints, respectively. Generalized speeds and quasi-velocities are used interchangeably to represent the velocity in the body frame while generalized velocities are in the inertial frame.

	NE	EL	Kane	BH	UK
Lagrangian construction	No	Yes	No	Yes	No
Configuration variables	Inertial frame	Gen. coordinates q	Gen. coordinates q	Gen. coordinates q	Any
Motion variables	Any frame	Gen. velocities $\dot{q}$	Gen. speeds u	Quasi velocities u	Any
Constraints equations (explicitly)	$c_h + c_{nh}$	c_{nh}	None	None	if NE is used: $c_h + c_{nh}$ if EL is used c_{nh}
Constraints encapsulated (implicitly)	None	c_h	$c_h + c_{nh}$	$c_h + c_{nh}$	if NE is used: None if EL is used c_h

Comparison

Table 2.1 provides a comparison of various modelling approaches that were discussed in the modelling section. The table highlights the dynamical features that each method can represent as well as some of the parameters required for using the method. Among the important differences are the constraint equations required to be presented explicitly for each model. In the NE method, each constraint must be introduced as an additional equation to the set of EOM. Thus, the EOM do not encapsulate any of the system's constraints by

default. On the other hand, by using generalized coordinates in the EL method, the configuration constraints (holonomic constraints) are implicitly embedded in the EOM. However, the motion constraints (nonholonomic constraints) must be introduced explicitly. A more convenient way to account for nonholonomic constraints is by using quasi-velocities to describe the motion variables, as is the case in the Kane and the BH methods. This results in encapsulating both configuration and motion constraints in the EOM. Similarly, the UK method can conveniently satisfy nonholonomic constraints. Regardless of the method used to develop the EOM, several important considerations related to airships are discussed in the following section.

2.1.1 Airship Dynamic Models

One of the earliest published attempts to derive EOM for airships was carried out by Jones *et al.* [27]. The model was used to study the stability of airships. Although the model was fairly simplified by ignoring rolling motion and assuming neutral buoyancy, Jones *et al.* were able to correctly account for the effect of virtual mass (also known as apparent mass and added mass), which was derived earlier by Lamb [28], and eventually represented these effects in the "acceleration stability derivative" form [29]. The added mass does not refer to the mass of the fluid that is being carried with the vehicle but it is rather the energy required to accelerate this fluid, when the vehicle accelerates [30]. To be more specific, as the vehicle moves, pressure-induced forces and moments are generated, causing the surrounding fluid particles, in the vicinity of the body, to oscillate [31]. The contribution of these forces is negligible in the case of heavier than air vehicles but substantial in LTA aircraft and increases with the acceleration of the body. Later on, a six degrees of freedom (DOF) model was developed in the field of hydrodynamics by Lamb [32] assuming potential flow. Lewis *et al.* [33] formulated a governing equation for underwater vehicles by constructing a Lagrangian from the energy of the vehicle and the energy of a sphere of the surrounding medium. The same principles were employed by Cook [34], Nippress *et al.* [35], and Gomes [36] to model airships' dynamics. However, Cook [34] proposed an additional simplification for airships where he assumed that the coupling of the longitudinal

and lateral motion, in terms of added mass influence, is negligible for LTA vehicles. The latter assumption resulted in significant simplification of the equations of motion, namely the added mass terms which were represented as derivative terms. The full derivations of added mass terms in derivative forms are found in Imlay [37]. The added mass derivative terms can be represented in terms of added mass factors (k_1, k_2, and k') [32, 38]. Further simplification for added mass and added inertia terms was adopted by Gomes [36], namely for the translational-angular "acceleration derivatives" terms [3]. These terms were measured experimentally and found to have insignificant effect on the investigated airship.

It would be interesting to categorize the dynamic models of airships based on their scales. Airships with volume below 200m^3 will be referred to as small-sized airships. It is very common to use the NE approach for constructing a dynamic model for small-sized airships [39–43]. On the other hand, many of the dynamic models derived for large-sized airships are based on the EL approach [34–36]. Nevertheless, there are examples in which small-sized airships were modelled using the EL [44] and large-sized airships modelled using the NE [45]. It is worth noting that the dynamic models used for large-sized airships were mostly tested on scaled down models of original airship size.

A number of recurring simplifications have been made in the above models. These include:

- Rigid body in which the aero-elastic effects are ignored [17, 39, 40, 42–44, 46, 47]
- Centre of Volume (CV) coincides with Centre of Buoyancy (CB) [17, 39, 40, 42–44, 46]
- $(X - Z)$ plane (longitudinal plane) is a plane of symmetry [17, 36, 39, 40, 42–44, 46]
- Added mass should be accounted for in the model [17, 39, 40, 42–44, 46]
- Mass and volume remain constant [17, 39, 42, 43, 47]
- The inertial frame is fixed and gravitational acceleration is constant (flat earth) [17, 39, 40, 42–44, 46]

[3] Given in Gomes [36] as $X_{\dot{q}}, Y_{\dot{p}}, Y_{\dot{r}}, Z_{\dot{q}}, L_{\dot{v}}, M_{\dot{u}}, M_{\dot{w}}, N_{\dot{v}}$. Note that the angular acceleration notations defined in this work are different from Gomes where $\dot{p} = \ddot{\phi}, \dot{q} = \ddot{\theta}, \dot{r} = \ddot{\psi}$

- Inertia of helium inside the hull is neglected [17, 39, 40, 42, 44, 46, 47].

Since the NE and the EL approaches are the most common approaches used to model airships, both methods are used, in the Appendix A.5 and A.6, to derive the EOM of an airship.

2.1.2 Slung Payload Dynamic Models

An early model for multibody dynamics of aircraft/payload was derived by applying the NE equations on the aircraft and payload separately [48]. The model was recently used to model a quadrotor slung-load system [49]. The expressions for the forces and moments acting on each system are then coupled with the cable tension force. While the cable stretching was modelled as a massless spring-damper system, the aerodynamics forces were neglected.

More recently, a hybrid approach is used to model the vehicle dynamics in uninhabited aircraft [50],[51],[52]. The hybrid approach is derived using the EL method and categorizes the model into modes, based on the tension force of the cable. The first mode is when the cable is fully taut in which the forces are transferred between the aircraft and the load. In the second mode, the cable is slack in which the payload is modelled as free-falling. The hybrid model assumes a load of point mass while the cable is mass-less and unstretchable. The system is usually treated as planar motion, resulting in 4 DOF when the cable is taut and 5 DOF when the cable is slack (due to independent aircraft/load dynamics). In the take-off stage, the payload is at rest and the cable is slack. Thus, the load has no contribution to the dynamics. The take-off stage ensures that the lift applied to the payload is optimal by positioning the aircraft on top of the payload before lifting.

$$||r_{AC} - r_L|| \geq \ell \tag{2.15}$$

where r_{AC} and r_L are the position of the aircraft and load, respectively, and ℓ is the length of the cable. The transition from slack to taut cable is called cable collision and modelled using principles of impulse and momentum (conservation of momentum) assuming a perfect

inelastic collision [53]. The latter allows to compute the states of aircraft and payload after the transition. For further details on how to compute the impulse of the collision, the equations are presented in [54]. Once the position condition is satisfied (2.15), the pulling stage takes place. In this stage, the aircraft is on top of the load and the cable is fully extended but the payload is still on the ground. The pulling stage is finished when a pre-defined height is reached. In the raise stage, the payload is already off the ground, thus, the normal force term no longer affects the dynamics of the aircraft/payload. Note that during this stage, the model assumes that the aircraft will only accelerate vertically. The quadrotor slung-load is differentially-flat, guaranteeing that any $\boldsymbol{C}^5$ continuous (six times differentiable) trajectory will have a 1-to-1 map to the states and inputs' space [55]. Nevertheless, this approach leads to a complex trajectory optimization problem since the constraints will require a non-convex optimization approach [56].

The hybrid model was simplified in [55] by assuming that the cable is always taut. Similarly, it assumes a point mass for slung load and mass-less cable. The model also neglects the aerodynamic forces and variations in yaw angles. The model is derived in a similar fashion to the hybrid model, using the Lagrange d'Alembert principle which requires finding the kinetic and potential energy of the system in addition to the virtual work. This model yields the same trajectory optimization complexity of the hybrid model.

A different dynamic model was proposed by [56] where the multibody dynamics are approximated as 3 passive constrained joints. The payload is assumed to be connected to the aircraft via two revolute and one prismatic joints. This approach yields the same behavior of cable-payload hybrid mode by imposing limits on the prismatic joint (identical to the fully taut cable mode). The proposed model is simpler in terms of task parameterizing and trajectory generation than the hybrid mode model. The model was validated in a 3D quadrotor-payload system in three experimental setups (waypoint navigation, obstacle avoidance, and payload throwing). Nevertheless, the aerodynamic forces were neglected.

Klausen *et al.* [57] used Kane's method to model a system consisting of a slung load attached to multirotor UAV. A model for each component was derived individually. The model of the multirotor UAV was adopted from [58] which is derived using the NE method.

A load is then added to the system which acts as an external force to the model of the UAV. Next, the dynamics of the suspended load are modelled as a pendulum. The load is a point mass connected to the UAV's centre of gravity by rigid rod. The displacement of the pendulum is parametrized using angles ϕ_L and θ_L which rotate about x and y axes, respectively, in the inertial frame. A linear damping term was used to model the physical swinging motion of the pendulum. Lastly, Kane's method [59] is used to find the model of the interconnected system. Although the resulting model is presented in typical matrix form, addition of further terms was necessary to avoid singularity points in the model (mainly coming from the pendulum model).

A more recent dynamic model was derived in [60] for airship/payload system. The model treats each sub-system separately by proposing two dynamic models, one for the airship and one for a controlled platform (which carries the payload). In [60], the cable length is adjustable by controlling winches on the platform whereas cable length is fixed in the platform model of [61]. The authors justify the addition of a separate payload control input to the low control authority that the airship has on the payload. The coupling between the airship and payload is introduced in the model of each sub-system as an external disturbance input. The dynamic model for the airship is adopted from the work of Fossen *et al.* [31]. Although the model seems promising as it can accommodate aerodynamic forces conveniently, it was not tested experimentally.

2.2 Methodology

Five dynamic modelling methods were discussed at the beginning of this chapter. The NE and the EL are inconvenient when nonholonomic constraints are involved. On the other hand, the Kane and the BH are laborious to derive and the EOM produced by either methods require further manipulation to be represented in a symbolic form. As an alternative, the UK method is straightforward and yields a tractable symbolic EOM. Therefore, the UK approach was adopted to model the multibody airship-gondola-slung-payload system. The components of the multibody system are modelled assuming no

constraints, after which the constraint forces are augmented.

2.2.1 Deriving the UK Equation for Blimp-Gondola-Slung-Payload

The investigated multibody system consists of three bodies: a blimp, a gondola, and a payload. Initially, each body has six degrees of freedom which results in a total of eighteen degrees of freedom for the entire multibody system (*i.e.*, $n = 18$). Imposing constraints reduces the degrees of freedom of the system[4]. In order to use (2.9), the constraint force (Q_c) must be derived. The constraint force is given in (2.12) and it depends on the constraint (2.10). Therefore, the constraint equations must be derived.

Three constraints will be imposed: a position and an orientation constraint for the blimp-gondola, and a length constraint between the gondola and the payload. Vectors involved in the blimp-gondola-slung-payload system are presented in Fig. 2.3. The position

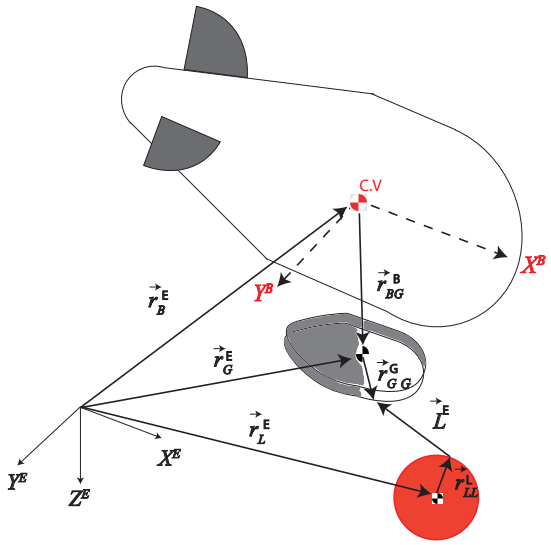


Figure 2.3: Constraint for blimp-gondola-slung-payload.

constraint between the blimp and gondola is given as

$$\vec{r}^{\,E}_B + \mathbf{R}_{1B}\, \vec{r}^{\,B}_{BG} = \vec{r}^{\,E}_G \tag{2.16}$$

[4]As discussed at the beginning of Chapter 2

where $\vec{r}_B^E$ and $\vec{r}_G^E$ are, respectively, the position vectors of the blimp and gondola in inertial frame (as shown in Fig. 2.3) and $\mathbf{R}_{1B}$ is the transformation matrix which converts the linear velocity from body to inertial frame (Appendix A.2). Vector $\vec{r}_{BG}^B$ is presented in the body frame, going from the blimp CV to the gondola CG. To distinguish between vectors in body and inertial frames, the former vectors are pre-multiplied by transformation or rotation matrices ($\mathbf{R}_1$ or $\mathbf{R}_2$ where $\mathbf{R}_2$ is the rotation matrix which converts the angular velocity from body to inertial coordinate). Removing the superscript notation for body and inertial frames, (2.16) becomes

$$\vec{C}_1 : \ \vec{r}_B + \mathbf{R}_{1B}\ \vec{r}_{BG} = \vec{r}_G \tag{2.17}$$

where C_i is used to enumerate the constraints. Differentiating (2.17) twice to get an acceleration constraint.

$$\begin{aligned}
&\frac{\mathrm{d}}{\mathrm{d}t}\vec{C}_1 : \ \mathbf{R}_{1B}\vec{V}_B + \dot{\mathbf{R}}_{1B}\ \vec{r}_{BG} + \mathbf{R}_{1B}\dot{\vec{r}}_{BG} = \mathbf{R}_{1G}\vec{V}_G \\
&\frac{\mathrm{d}^2}{\mathrm{d}t^2}\vec{C}_1 : \ \mathbf{R}_{1B}\vec{a}_B + \dot{\mathbf{R}}_{1B}\vec{V}_B + \ddot{\mathbf{R}}_{1B}\ \vec{r}_{BG} + 2\dot{\mathbf{R}}_{1B}\ \dot{\vec{r}}_{BG} + \mathbf{R}_{1B}\ \ddot{\vec{r}}_{BG} = \mathbf{R}_{1G}\vec{a}_G + \dot{\mathbf{R}}_{1G}\vec{V}_G
\end{aligned} \tag{2.18}$$

where $\vec{a}$ and $\vec{V}$ are the translational acceleration and velocity, respectively, in the body frame. Similarly, $\vec{\Omega}$ and $\vec{\omega}$ are respectively the rotational acceleration and velocity. Equation (2.18) must be rearranged so that it matches the form given in (2.10). Expanding the acceleration vector in (2.10) yields (2.19)

$$\mathbf{A}(\vec{q}, \dot{\vec{q}}) \begin{bmatrix} \vec{a}_B \\ \vec{\Omega}_B \\ \vec{a}_G \\ \vec{\Omega}_G \\ \vec{a}_L \\ \vec{\Omega}_L \end{bmatrix} = \vec{b}(\vec{q}, \dot{\vec{q}}) \tag{2.19}$$

The rearranging step will be carried out at the end of each constraint derivation.

Using chain-rule to find $\dot{\mathbf{R}}_1$ and $\ddot{\mathbf{R}}_1$

$$\dot{\mathbf{R}}_1 = \frac{\partial \mathbf{R}_1}{\partial \phi}\dot{\phi} + \frac{\partial \mathbf{R}_1}{\partial \theta}\dot{\theta} + \frac{\partial \mathbf{R}_1}{\partial \psi}\dot{\psi} \tag{2.20}$$

$$
\begin{aligned}
\ddot{\mathbf{R}}_1 = &\left(\frac{\partial^2 \mathbf{R}_1}{\partial \phi^2}\dot{\phi} + \frac{\partial^2 \mathbf{R}_1}{\partial \phi \partial \theta}\dot{\theta} + \frac{\partial^2 \mathbf{R}_1}{\partial \phi \partial \psi}\dot{\psi}\right)\dot{\phi} + \frac{\partial \mathbf{R}_1}{\partial \phi}\ddot{\phi} \\
&+ \left(\frac{\partial^2 \mathbf{R}_1}{\partial \theta \partial \phi}\dot{\phi} + \frac{\partial^2 \mathbf{R}_1}{\partial \theta^2}\dot{\theta} + \frac{\partial^2 \mathbf{R}_1}{\partial \theta \partial \psi}\dot{\psi}\right)\dot{\theta} + \frac{\partial \mathbf{R}_1}{\partial \theta}\ddot{\theta} \\
&+ \left(\frac{\partial^2 \mathbf{R}_1}{\partial \psi \partial \phi}\dot{\phi} + \frac{\partial^2 \mathbf{R}_1}{\partial \psi \partial \theta}\dot{\theta} + \frac{\partial^2 \mathbf{R}_1}{\partial \psi^2}\dot{\psi}\right)\dot{\psi} + \frac{\partial \mathbf{R}_1}{\partial \psi}\ddot{\psi}
\end{aligned}
\tag{2.21}
$$

where ϕ, θ, and ψ are Euler angles.

To present the constraint equation in a compact form, the following matrices are introduced.

$$
\begin{aligned}
\mathbf{G} &= \begin{bmatrix} \dfrac{\partial \mathbf{R_1}}{\partial \phi}\vec{r} & \dfrac{\partial \mathbf{R_1}}{\partial \theta}\vec{r} & \dfrac{\partial \mathbf{R_1}}{\partial \psi}\vec{r} \end{bmatrix} \\
\mathbf{G}_\phi &= \begin{bmatrix} \dfrac{\partial^2 \mathbf{R_1}}{\partial \phi^2}\vec{r} & \dfrac{\partial^2 \mathbf{R_1}}{\partial \phi \partial \theta}\vec{r} & \dfrac{\partial^2 \mathbf{R_1}}{\partial \phi \partial \psi}\vec{r} \end{bmatrix} \\
\mathbf{G}_\theta &= \begin{bmatrix} \dfrac{\partial^2 \mathbf{R_1}}{\partial \theta \partial \phi}\vec{r} & \dfrac{\partial^2 \mathbf{R_1}}{\partial \theta^2}\vec{r} & \dfrac{\partial^2 \mathbf{R_1}}{\partial \theta \partial \psi}\vec{r} \end{bmatrix} \\
\mathbf{G}_\psi &= \begin{bmatrix} \dfrac{\partial^2 \mathbf{R_1}}{\partial \psi \partial \phi}\vec{r} & \dfrac{\partial^2 \mathbf{R_1}}{\partial \psi \partial \theta}\vec{r} & \dfrac{\partial^2 \mathbf{R_1}}{\partial \psi^2}\vec{r} \end{bmatrix}
\end{aligned}
\tag{2.22}
$$

where $\mathbf{G}, \mathbf{G}_\phi, \mathbf{G}_\theta, \mathbf{G}_\psi \in \mathbb{R}^{3\times 3}$ are presented in the body frame. Using (2.22) simplifies the $\ddot{\mathbf{R}}_{1B}\ \vec{r}_{BG}$, $\ddot{\mathbf{R}}_{1G}\ \vec{r}_{GG}$, and $\ddot{\mathbf{R}}_{1L}\ \vec{r}_{LL}$ terms. The former term appears in the position constraint ($\vec{C}_1$) while the latter terms will appear in the following constraint equations. These terms are decomposed to (2.23)

$$
\ddot{\mathbf{R}}_1\ \vec{r} = [\ \mathbf{G}_\phi\ \mathbf{R}_2\ \vec{\omega} \qquad \mathbf{G}_\theta\ \mathbf{R}_2\ \vec{\omega} \qquad \mathbf{G}_\psi\ \mathbf{R}_2\ \vec{\omega}\ ]\ \mathbf{R}_2\ \vec{\omega} + \mathbf{G}\ (\ \dot{\mathbf{R}}_2\ \vec{\omega} + \mathbf{R}_2\ \vec{\Omega}\ ) \tag{2.23}
$$

Substituting (2.23) in (2.18) and rearranging to obtain $\mathbf{A}_1$ and $\vec{b}_1$ gives (2.24)

$$
\begin{aligned}
\mathbf{A}_1 &= [\ \mathbf{R}_{1B} \quad \mathbf{G}_{BG}\ \mathbf{R}_{2B} \quad -\mathbf{R}_{1G} \quad \mathbf{0}_{3\times 3} \quad \mathbf{0}_{3\times 3} \quad \mathbf{0}_{3\times 3}\ ] \\
\vec{b}_1 &= -\dot{\mathbf{R}}_{1B}\ \vec{V}_B + \dot{\mathbf{R}}_{1G}\ \vec{V}_G \\
&\quad - [\ \mathbf{G}_{\phi_{BG}}\ \mathbf{R}_{2B}\ \vec{\omega}_B \qquad \mathbf{G}_{\theta_{BG}}\ \mathbf{R}_{2B}\ \vec{\omega}_B \qquad \mathbf{G}_{\psi_{BG}}\ \mathbf{R}_{2B}\ \vec{\omega}_B\ ]\ \mathbf{R}_{2B}\ \vec{\omega}_B \\
&\quad - \mathbf{G}_{BG}\ \dot{\mathbf{R}}_{2B}\ \vec{\omega}_B - 2\ \dot{\mathbf{R}}_{1B}\ \dot{\vec{r}}_{BG} - \mathbf{R}_{1B}\ \ddot{\vec{r}}_{BG}
\end{aligned}
\tag{2.24}
$$

where $\mathbf{0}_{3\times 3} \in \mathbb{R}^{3\times 3}$ is a three by three zero matrix. Note that the first, third, and fifth elements[5] of $\mathbf{A_1}$ represent the terms in the constraint equation (2.18) that are multiplied

[5]Each element of $\mathbf{A_1}$ is a three by three matrix ($\mathbf{A_1} \in \mathrm{Re}^{3\times 18}$)

by the translational acceleration of the blimp, gondola, and payload, respectively. On the other hand, the second, fourth, and sixth elements of $\mathbf{A_1}$ represent the terms multiplied by the angular acceleration.

The angular velocity constraint between the blimp and gondola is necessary to ensure that any rotation applied about the blimp yields an identical rotation about the gondola (or vice-versa). The latter can be enforced by equating the angular velocity of the gondola, in the inertial frame, to the blimp.

$$\begin{aligned}
&\frac{\mathrm{d}}{\mathrm{d}t}\vec{C}_2 = \mathbf{R}_{2G}\ \vec{\omega}_G = \mathbf{R}_{2B}\ \vec{\omega}_B \\
&\frac{\mathrm{d}^2}{\mathrm{d}t^2}\vec{C}_2 = \dot{\mathbf{R}}_{2G}\ \vec{\omega}_G + \mathbf{R}_{2G}\ \vec{\Omega}_G = \dot{\mathbf{R}}_{2B}\ \vec{\omega}_B + \mathbf{R}_{2B}\ \vec{\Omega}_B
\end{aligned} \tag{2.25}$$

Rearranging to get $\mathbf{A}_2$ and $\vec{b}_2$

$$\begin{aligned}
&\mathbf{A}_2 = [\ \mathbf{0}_{3\times3} \quad \mathbf{R}_{2B} \quad \mathbf{0}_{3\times3} \quad -\mathbf{R}_{2G} \quad \mathbf{0}_{3\times3} \quad \mathbf{0}_{3\times3}\ ] \\
&\vec{b}_2 = -\dot{\mathbf{R}}_{2B}\ \vec{\omega}_B + \dot{\mathbf{R}}_{2G}\ \vec{\omega}_G
\end{aligned} \tag{2.26}$$

The length constraint between the gondola and payload is derived by first obtaining an expression for the position of the cable. The vector connecting the payload with the gondola is described as follows.

$$\vec{L} = \vec{r}_G + \mathbf{R}_{1G}\ \vec{r}^{\,G}_{GG} - \mathbf{R}_{1L}\ \vec{r}^{\,L}_{LL} - \vec{r}_L \tag{2.27}$$

where $\vec{r}_L$ is the position vector of the cable in the inertial frame, $\mathbf{R}_{1G}$ and $\mathbf{R}_{1L}$ are the transformation matrices from body to inertial frames. Recall that pre-multiplication by a transformation matrix indicates a vector in the body frame. Vectors in the body frame presented in (2.27), namely $\vec{r}^{\,G}_{GG}$ and $\vec{r}^{\,L}_{LL}$, are the vectors going from the CG of the gondola and the load, respectively, to the point where the cable is attached to each body. $\vec{L}$ is the load vector in inertial frame. Since the airship will fly at low speeds, a rigid cable will be assumed, *i.e.,* constant length[6]. It is worth noting that multiple cables could be introduced where each cable would impose an additional constraint equation.

[6]This assumption may be violated if severe wind is encountered

Since the direction of $\vec{L}$ varies with time, a more convenient constraint would be to take the length of the cable. Hence, the constraint will be defined such that

$$\begin{aligned} C_3 = ||L||^2 &= \vec{L}^T\ \vec{L} \\ &= (\vec{r}_G + \mathbf{R}_{1G}\ \vec{r}_{GG} - \mathbf{R}_{1L}\ \vec{r}_{LL} - \vec{r}_L)^T\ (\vec{r}_G + \mathbf{R}_{1G}\ \vec{r}_{GG} - \mathbf{R}_{1L}\ \vec{r}_{LL} - \vec{r}_L) \end{aligned} \tag{2.28}$$

Differentiating (2.28) twice to get a constraint on the acceleration.

$$\frac{\mathrm{d}}{\mathrm{d}t} C_3 = 2\ \vec{L}^T\ \dot{\vec{L}} = 0 \tag{2.29}$$

$$\frac{\mathrm{d}^2}{\mathrm{d}t^2} C_3 = 2\ \dot{\vec{L}}^T\ \dot{\vec{L}} + 2\ \vec{L}^T\ \ddot{\vec{L}} = 0 \tag{2.30}$$

where $\vec{L}$, $\dot{\vec{L}}$, and $\ddot{\vec{L}}$ are respectively given in (2.27), (2.31), and (2.32).

$$\dot{\vec{L}} = \mathbf{R}_{1G}\ \vec{V}_G + \dot{\mathbf{R}}_{1G}\ \vec{r}_{GG} - \dot{\mathbf{R}}_{1L}\ \vec{r}_{LL} - \mathbf{R}_{1L}\ \vec{V}_L \tag{2.31}$$

$$\ddot{\vec{L}} = \mathbf{R}_{1G}\ \vec{a}_G + \dot{\mathbf{R}}_{1G}\ \vec{V}_G + \ddot{\mathbf{R}}_{1G}\ \vec{r}_{GG} - \ddot{\mathbf{R}}_{1L}\ \vec{r}_{LL} - \mathbf{R}_{1L}\ \vec{a}_L - \dot{\mathbf{R}}_{1L}\ \vec{V}_L \tag{2.32}$$

Substituting (2.31), (2.32), and (2.23) in (2.30) yields

$$\begin{aligned} \frac{\mathrm{d}^2}{\mathrm{d}t^2} C_3 :\ & 2L^T \Big\{ \mathbf{R}_{1G}\ \vec{a}_G + \dot{\mathbf{R}}_{1G}\ \vec{V}_G \\ & + [\ \mathbf{G}_{\phi\mathbf{G}}\ \mathbf{R}_{2G}\ \vec{\omega}_G \quad \mathbf{G}_{\theta\mathbf{G}}\ \mathbf{R}_{2G}\ \vec{\omega}_G \quad \mathbf{G}_{\psi\mathbf{G}}\ \mathbf{R}_{2G}\ \vec{\omega}_G\ ]\ \mathbf{R}_{2G}\ \vec{\omega}_G \\ & + \mathbf{G}_G\ (\ \dot{\mathbf{R}}_{2G}\ \vec{\omega}_G + \mathbf{R}_{2G}\ \vec{\Omega}_G\ ) + \vec{G}_\lambda \\ & - [\ \mathbf{G}_{\phi\mathbf{L}}\ \mathbf{R}_{2L}\ \vec{\omega}_L \quad \mathbf{G}_{\theta\mathbf{L}}\ \mathbf{R}_{2L}\ \vec{\omega}_L \quad \mathbf{G}_{\psi\mathbf{L}}\ \mathbf{R}_{2L}\ \vec{\omega}_L\ ]\ \mathbf{R}_{2L}\ \vec{\omega}_L \\ & - \mathbf{G}_L\ (\ \dot{\mathbf{R}}_{2L}\ \vec{\omega}_L + \mathbf{R}_{2L}\ \vec{\Omega}_L\ ) \\ & - \mathbf{R}_{1L}\ \vec{a}_L - \dot{\mathbf{R}}_{1L}\ \vec{V}_L \Big\} \\ & + 2\ (\mathbf{R}_{1G}\ \vec{V}_G + \dot{\mathbf{R}}_{1G}\ \vec{r}_{GG} - \dot{\mathbf{R}}_{1L}\ \vec{r}_{LL} - \mathbf{R}_{1L}\ \vec{V}_L)^T\ (\mathbf{R}_{1G}\ \vec{V}_G + \dot{\mathbf{R}}_{1G}\ \vec{r}_{GG} - \dot{\mathbf{R}}_{1L}\ \vec{r}_{LL} - \mathbf{R}_{1L}\ \vec{V}_L) = 0 \end{aligned} \tag{2.33}$$

where $\vec{G}_\lambda$ contains the variation of $\mathbf{R}_{1G}$ with respect to the vectors of the gondola in the body frame. While $\vec{G}_\lambda$ is always assumed to be zero for the purposes of this thesis, the full expression is given in the appendix (A.34).

Recall that the constraint equations must be presented in the form given in (2.10). Hence, the last step is to identify $\mathbf{A}$ and $\vec{b}$ in (2.33) which yields (2.34).

$$\vec{A}_3 = 2\vec{L}^T\ [\mathbf{0}_{3\times3} \quad \mathbf{0}_{3\times3} \quad \mathbf{R}_{1G} \quad \mathbf{G}_{GG}\ \mathbf{R}_{2G} \quad -\mathbf{R}_{1L} \quad -\mathbf{G}_{LL}\ \mathbf{R}_{2L}]$$

$$
\begin{aligned}
b_3 = 2\vec{L}^T \Big\{ & \left[\begin{array}{ccc} \mathbf{G}_{\phi_{GG}}\, \mathbf{R}_{2G}\, \vec{\omega}_G & \mathbf{G}_{\theta_{GG}}\, \mathbf{R}_{2G}\, \vec{\omega}_G & \mathbf{G}_{\psi_{GG}}\, \mathbf{R}_{2G}\, \vec{\omega}_G \end{array} \right] \mathbf{R}_{2G}\, \vec{\omega}_G \\
& + \mathbf{G}_{GG}\, (\, \dot{\mathbf{R}}_{2G}\, \vec{\omega}_G + \mathbf{R}_{2G}\, \vec{\Omega}_G\,) + \vec{G}_\lambda \\
& - \left[\begin{array}{ccc} \mathbf{G}_{\phi_{LL}}\, \mathbf{R}_{2L}\, \vec{\omega}_L & \mathbf{G}_{\theta_{LL}}\, \mathbf{R}_{2L}\, \vec{\omega}_L & \mathbf{G}_{\psi_{LL}}\, \mathbf{R}_{2L}\, \vec{\omega}_L \end{array} \right] \mathbf{R}_{2L}\, \vec{\omega}_L \\
& - \mathbf{G}_{LL}\, (\, \dot{\mathbf{R}}_{2L}\, \vec{\omega}_L + \mathbf{R}_{2L}\, \vec{\Omega}_L\,) \Big\} \\
& - 2\dot{\vec{L}}^T\, \dot{\vec{L}}
\end{aligned}
\tag{2.34}
$$

Combining all constraints in one matrix gives

$$
\mathbf{A} = \begin{bmatrix} \mathbf{A_1} \\ \mathbf{A_2} \\ \vdots \\ \mathbf{A_c} \end{bmatrix} \qquad \vec{b} = \begin{bmatrix} \vec{b_1} \\ \vec{b_2} \\ \vdots \\ \vec{b_c} \end{bmatrix} \tag{2.35}
$$

where c is the number of constraints. The dimensions of the Jacobian matrices $\mathbf{A}_1$, $\mathbf{A}_2 \in \mathbb{R}^{3\times n}$, and $\vec{A}_3 \in \mathbb{R}^{1\times n}$ where n is the degrees of freedom of the multibody system (*i.e.*, $n = 18$).

Substituting $\mathbf{A}$ and $\vec{b}$ back into (2.14) yields the equation of motion of a constrained blimp-gondola-payload system. To solve the EOM, recall (2.14)

$$
\ddot{\vec{q}} = \mathbf{M^{-1}}\, \vec{Q} + \mathbf{M}^{-\frac{1}{2}}\, (\mathbf{A}\, \mathbf{M}^{-\frac{1}{2}})^\dagger\, (\vec{b} - \mathbf{A}\, \mathbf{M^{-1}}\, \vec{Q}\,)
$$

where

$$
\ddot{\vec{q}} = \begin{bmatrix} \vec{a}_B \\ \vec{\Omega}_B \\ \vec{a}_G \\ \vec{\Omega}_G \\ \vec{a}_L \\ \vec{\Omega}_L \end{bmatrix}
$$

$$\mathbf{M} = \begin{bmatrix} \mathbf{M_B} & -\mathbf{d_{VG}}\,\mathbf{M_B} & \mathbf{0_{3\times3}} & \mathbf{0_{3\times3}} & \mathbf{0_{3\times3}} & \mathbf{0_{3\times3}} \\ \mathbf{d_{VG}}\,\mathbf{M_B} & \mathbf{I_B} & \mathbf{0_{3\times3}} & \mathbf{0_{3\times3}} & \mathbf{0_{3\times3}} & \mathbf{0_{3\times3}} \\ \mathbf{0_{3\times3}} & \mathbf{0_{3\times3}} & \mathbf{M_G} & \mathbf{0_{3\times3}} & \mathbf{0_{3\times3}} & \mathbf{0_{3\times3}} \\ \mathbf{0_{3\times3}} & \mathbf{0_{3\times3}} & \mathbf{0_{3\times3}} & \mathbf{I_G} & \mathbf{0_{3\times3}} & \mathbf{0_{3\times3}} \\ \mathbf{0_{3\times3}} & \mathbf{0_{3\times3}} & \mathbf{0_{3\times3}} & \mathbf{0_{3\times3}} & \mathbf{M_L} & \mathbf{0_{3\times3}} \\ \mathbf{0_{3\times3}} & \mathbf{0_{3\times3}} & \mathbf{0_{3\times3}} & \mathbf{0_{3\times3}} & \mathbf{0_{3\times3}} & \mathbf{I_L} \end{bmatrix} \tag{2.36}$$

$$\mathbf{d_{VG}} = \begin{bmatrix} 0 & -d_{VG,z} & 0 \\ d_{VG,z} & 0 & -d_{VG,x} \\ 0 & d_{VG,x} & 0 \end{bmatrix}$$

where $\mathbf{d_{VG}}$ is the skew-symmetric matrix containing the distance between CV and CG of the blimp.

$$\mathbf{M_B} = \begin{bmatrix} (1+k_1)m_B & 0 & 0 \\ 0 & (1+k_2)m_B & 0 \\ 0 & 0 & (1+k_2)m_B \end{bmatrix} \quad \mathbf{I_B} = \begin{bmatrix} I_{Bx} & 0 & 0 \\ 0 & (1+k')I_{By} & 0 \\ 0 & 0 & (1+k')I_{Bz} \end{bmatrix}$$

$$\mathbf{M_G} = \begin{bmatrix} m_G & 0 & 0 \\ 0 & m_G & 0 \\ 0 & 0 & m_G \end{bmatrix} \quad \mathbf{I_G} = \begin{bmatrix} I_{Gx} & 0 & 0 \\ 0 & I_{Gy} & 0 \\ 0 & 0 & I_{Gz} \end{bmatrix}$$

$$\mathbf{M_L} = \begin{bmatrix} m_L & 0 & 0 \\ 0 & m_L & 0 \\ 0 & 0 & m_L \end{bmatrix} \quad \mathbf{I_L} = \begin{bmatrix} I_{Lx} & 0 & 0 \\ 0 & I_{Ly} & 0 \\ 0 & 0 & I_{Lz} \end{bmatrix}$$

$$\vec{Q} = \vec{A} + \vec{G} + \vec{U} - \vec{D}$$

where k_1, k_2, and k' are the added mass and added moment factors. $\vec{A}$, $\vec{D}$, $\vec{G}$, and $\vec{U}$ are the aerodynamic, dynamic, gravitation and buoyancy, and actuation forces, respectively. The dynamic term, $\vec{D}$, is derived in the appendix (A.78), the aerodynamic forces are derived from numerical simulations in Chapter 4. Gravitation and buoyancy forces are derived in the following subsection whereas the actuation forces are described in Chapter 5.

The gondola has a single degree of freedom, *i.e.,* it can only move along the rail. While the gondola is on the straight section of the rail, the frame of reference of the gondola is coincident with the blimp's frame. However, the orientation of the gondola differs from the blimp as the gondola reaches the curved section of the rail. Hence, the transformation and rotation matrices ($\mathbf{R}_{1G}$ and $\mathbf{R}_{2G}$, respectively) vary. The transformation and rotation matrices as well as vectors in the body frame $\vec{r}_{BG}$ (from 2.17) can be described for the straight section

$\mathbf{R}_{1G} = \mathbf{R}_{1G\ straight} = \mathbf{R}_{1B}$ (identical to (A.30)).

$\mathbf{R}_{2G} = \mathbf{R}_{2G\ straight} = \mathbf{R}_{2B}$ (identical to (A.31)).

$\vec{r}_{BG} = [Ss, 0, dzG]^T, \quad \dot{\vec{r}}_{BG} = \left[\dot{Ss}, 0, 0\right]^T, \quad \ddot{\vec{r}}_{BG} = \left[\ddot{Ss}, 0, 0\right]^T$

where Ss is the distance travelled by the gondola with respect to the blimp and dzG is the vertical distance (in z direction) between the blimp and the gondola.

On the other hand, the transformation and rotation matrices for the curved section are given as

$$\mathbf{R}_{1G} = \mathbf{R}_{1G\ curved} = \begin{bmatrix} \cos(\lambda) & 0 & \sin(\lambda) \\ 0 & 1 & 0 \\ -\sin(\lambda) & 0 & \cos(\lambda) \end{bmatrix} \mathbf{R}_{1G\ straight}$$

where λ is the angle between the vertical axis of the blimp (z axis) and the position of the gondola with respect to the curved rail (Fig. 2.4). The resulting transformation matrix ($\mathbf{R}_{1G\ curved}$) is presented in Appendix (A.32)

$$\mathbf{R}_{2G} = \mathbf{R}_{2G\ curved} = \begin{bmatrix} 1 & \sin(\phi)\tan(\theta+\lambda) & \cos(\phi)\tan(\theta+\lambda) \\ 0 & \cos(\phi) & -\sin(\phi) \\ 0 & \sin(\phi)\sec(\theta+\lambda) & \cos(\phi)\sec(\theta+\lambda) \end{bmatrix}$$

From Fig. 2.4, the rail angle (λ) can be described as (2.37)

$$\lambda = \frac{Ss - x_{curve}}{R_{curve}} \tag{2.37}$$

where x_{curve} is the distance from the blimp CV to the centre of the curved section along the x-axis in the body frame, R_{curve} is the radius of the curved section. The curved section

begins when $Ss \geq 0.3$. Given that the gondola is actuation-dependent, Ss is obtained using an encoder.

The position vector between the blimp and gondola ($\vec{r}_{BG}$) along the curved rail and its derivatives are given in (2.38)

$$\begin{aligned}
\vec{r}_{BG} &= \begin{bmatrix} x_{curve} + R_{curve}\sin(\lambda) \\ 0 \\ R_{curve}\cos(\lambda) \end{bmatrix} \\
\dot{\vec{r}}_{BG} &= \begin{bmatrix} \cos(\lambda)\dot{S}s \\ 0 \\ -\sin(\lambda)\dot{S}s \end{bmatrix} \\
\ddot{\vec{r}}_{BG} &= \begin{bmatrix} \cos(\lambda)\ \ddot{S}s - \frac{\sin(\lambda)\ \dot{S}s^2}{R_{curve}} \\ 0 \\ -\sin(\lambda)\ \ddot{S}s - \frac{\cos(\lambda)\ \dot{S}s^2}{R_{curve}} \end{bmatrix}
\end{aligned} \tag{2.38}$$

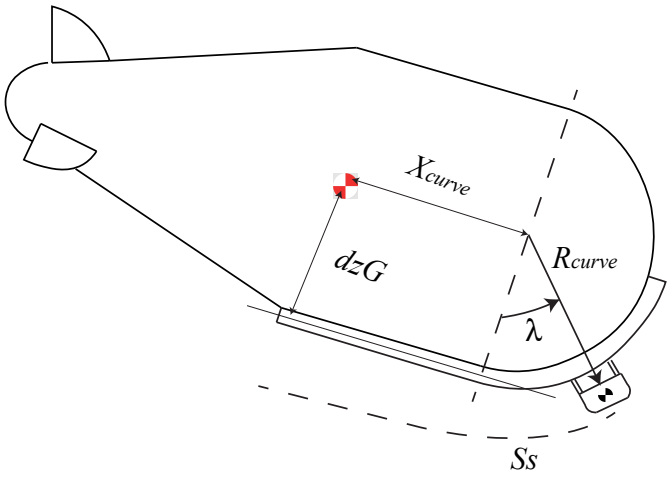


Figure 2.4: Demonstration of angle λ when gondola is at the curved section of the rail.d_{zG} is the vertical distance, in the body frame, between the blimp and the straight section of the rail.

2.2.2 Gravitational and Buoyancy Forces

At the beginning of Section 2.1.1, it was assumed that the centre of buoyancy is coincident with CV. As a result, the buoyancy force acts on the CV and it is given as

$$\vec{G} = \begin{bmatrix} m_B\ \vec{g} \\ \mathbf{d_{VG}}\ m_B\ \vec{g} \\ m_G\ \vec{g} \\ \vec{0} \\ m_L\ \vec{g} \\ \vec{0} \end{bmatrix} - \begin{bmatrix} (\rho - \rho_H)\ V_B\ \vec{g} \\ \vec{0} \\ \vec{0} \\ \vec{0} \\ \vec{0} \\ \vec{0} \end{bmatrix} \tag{2.39}$$

where m is the mass, $\vec{0}$ is a zero column vector ($\vec{0} \in \mathbb{R}^{3\times 1}$), $\mathbf{d_{VG}}$ is the skew-symmetric matrix containing the distance between the CV and the CG of the blimp (A.78), ρ and ρ_H are, respectively, the density of air and helium, and V_B is the volume of the blimp. To account for the variation of air density with respect to altitude, a density-elevation model was adopted from [17]. The model assumes a linear relationship between air density and elevation[7] to approximate the air density at a given altitude.

$$\begin{aligned} \rho &= \rho_{ref} - \Delta\rho\Delta Z \\ \Delta\rho &= -1.164 \times 10^{-4}\ kg/m^4 \end{aligned} \tag{2.40}$$

where ρ_{ref} is the air density at a reference point, $\Delta\rho$ is the slope of the density-elevation relationship, and ΔZ is the variation of altitude with respect to the reference point. The gravitational acceleration vector ($\vec{g}$) is defined in the body frame such that

$$\vec{g} = \mathbf{R_1^T} \begin{bmatrix} 0 \\ 0 \\ 9.81 \end{bmatrix} \tag{2.41}$$

2.3 Results and Discussion

Two examples of blimp-gondola-payload multibody system are presented here to exhibit the fidelity of the UK modelling method with the previously derived constraints. The first

[7] The relationship is based on Ottawa city elevation

example consists of blimp-gondola with a slung spherical payload attached via a massless cable. The parameters of the blimp-gondola, listed in Table 2.2 were obtained from the physical prototype that is investigated in this thesis. The integration is carried out using *ode45* in MATLAB which is an adaptive time-step solver that combines Runge-Kutta 4th order scheme along with the Fehlberg 5th order scheme to control the time-step.

The multibody system starts at the origin while the altitude for the blimp, gondola, payload is $z_{B0} = -10$ m, $z_{G0} = -9.06$ m, and $z_{L0} = -7.56$ m, respectively. The multibody system moves with an initial axial velocity of 1 m/s and a constant lift force, in the inertial frame, that is capable of maintaining the multibody system at a fixed altitude. A positive side force (along the Y axis), in the inertial frame, is exerted on the blimp CV as a step input for 1 second. Since the side force is applied at the blimp CV, a yawing moment is produced about the blimp CG. The resulting trajectory over a period of 7 seconds is presented in top view ($X - Y$ plane) in Fig. 2.5.

Table 2.2: Parameters of multibody used in this study

Parameter	Value	Units
m_B	2.66	kg
m_G	3.3	kg
m_L	1	kg
$I_{B\ x}$	0.87	kg m^2
$I_{B\ y} = I_{B\ z}$	3.72	
$I_{G\ x}$	0.01	kg m^2
$I_{G\ y} = I_{G\ z}$	0.14	
$I_{L\ x} = I_{L\ y} = I_{L\ z}$	0.1	kg m^2
x_{curve}	0.3	m
$R_{curve} = dzG$	0.94	m
Ss	0	m
$\vec{r}_{BG}$	$[0,\ 0,\ 0.94]^T$	m
$\vec{r}_{GG}$	$[0,\ 0,\ 0]^T$	m
$\vec{r}_{LL}$	$[0,\ 0,\ 0]^T$	m
k_1	0.1664	-
k_2	0.69	-
k'	0.3364	-

Examining Fig. 2.5, it can be seen that the multibody system was initially heading

east along $+x$-axis. The influence of the side force is quickly exhibited as the multibody system starts to yaw and roll. The rolling is portrayed in front view at selected locations and presented in small figures adjacent to the main figure. The rolling is more pronounced for the slung payload in comparison to the blimp-gondola, which is expected given the pendulum-like behaviour of the slung payload. The roll angle (ϕ) alternates with time, going from positive roll at location 2 to negative roll at location 4. On the other hand, the yaw angle (ψ) is positive throughout the trajectory. Location 3 shows a phase shift between the blimp-gondola and slung payload roll angles where the roll angle for the blimp-gondola is negative while the slung payload angle is positive.

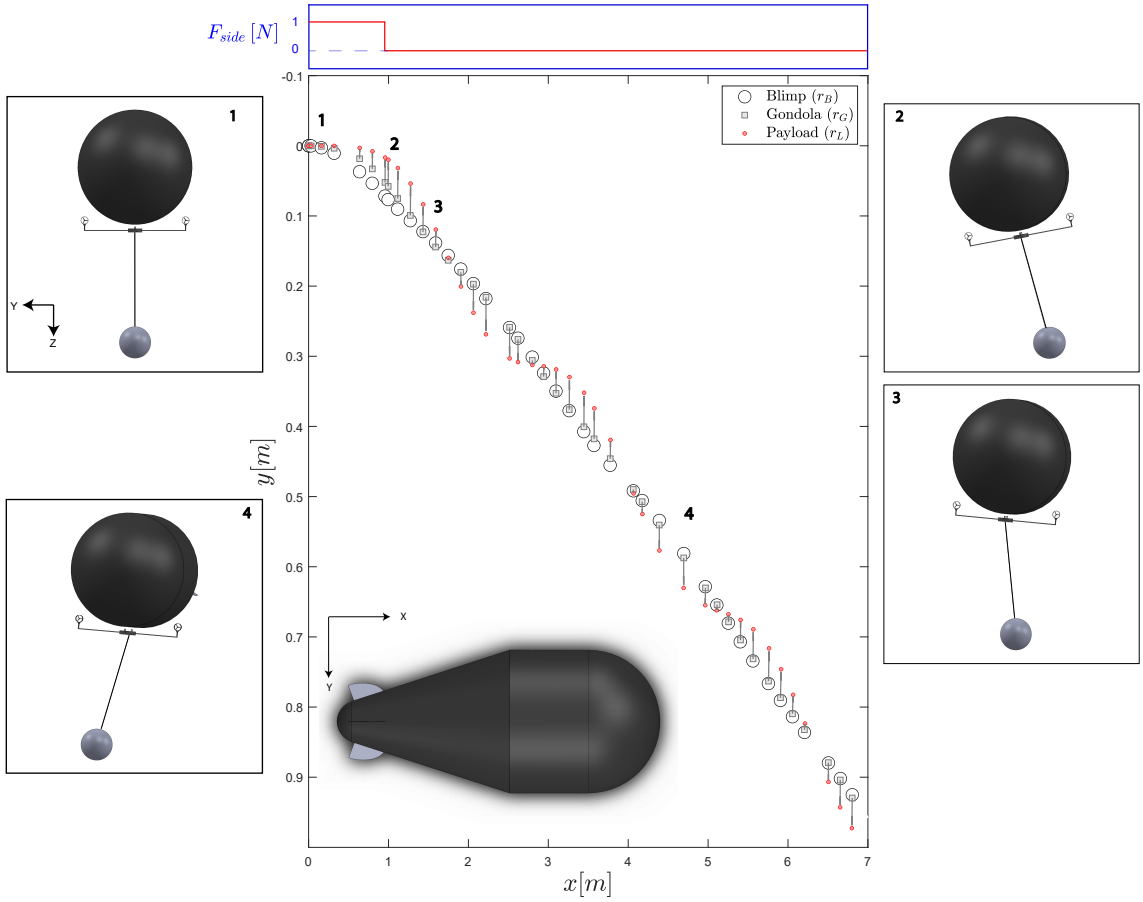


Figure 2.5: Position of blimp (circle), gondola (square), and payload (red small circle) viewed in X-Y plane. The grey solid line represents the cable that connects the payload to the gondola. Exaggerated orientations are portrayed in front view for selected locations. The applied side force over time is illustrated on top.

Position and Euler angles as well as body velocities are illustrated in Fig. 2.6. The axial velocity for all bodies is decreasing owing to drag. Moreover, the lateral velocity of the blimp increases over a period of 1 second after which it decreases. The increase in lateral velocity is attributed to the step side force. A similar trend is observed for the lateral velocity of the gondola. However, the payload exhibits a swinging behaviour where the lateral velocity varies in a sinusoidal manner. The swinging of the slung payload is further demonstrated by examining the angular velocity, about the x-axis (ω_x), of the blimp and gondola which also manifests a sinusoidal behaviour.

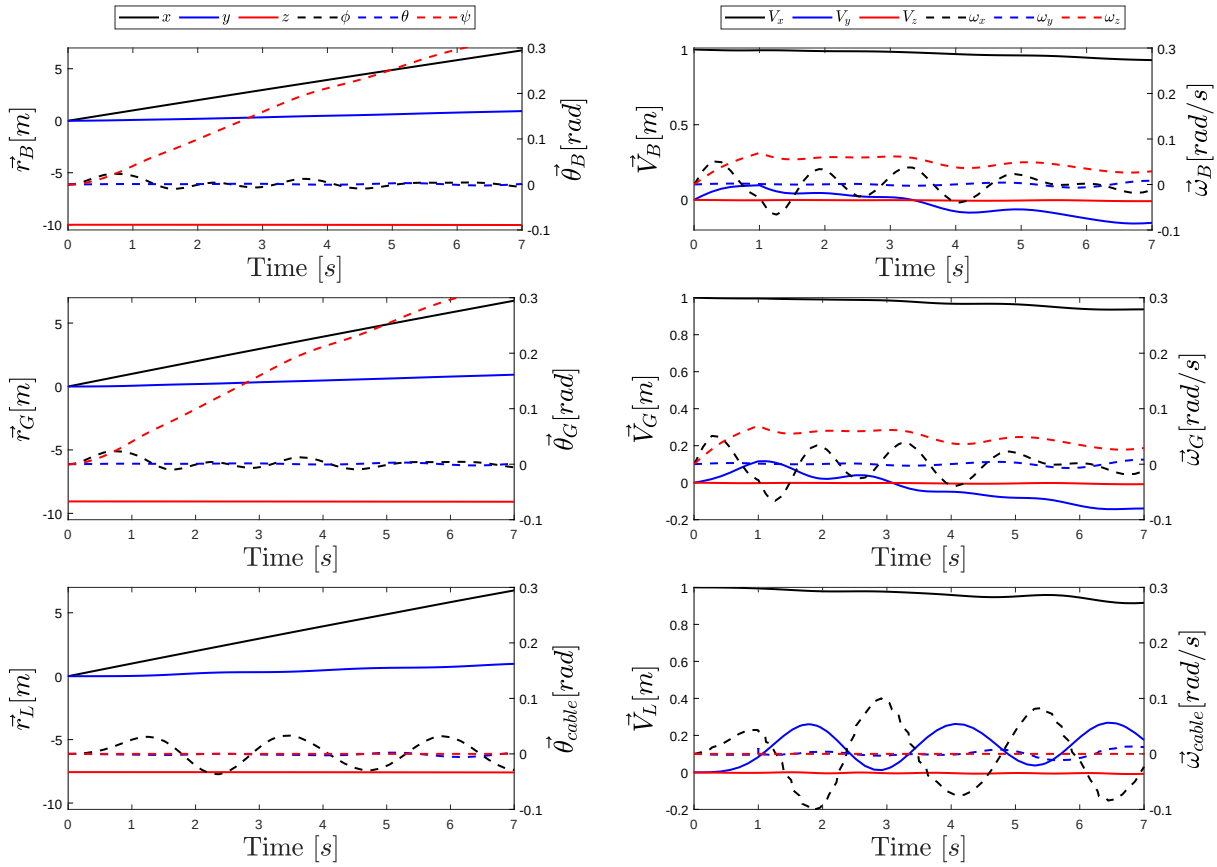


Figure 2.6: Results obtained from the first example. Top to bottom: blimp, gondola, and payload, respectively. Left column: Position ($\vec{r}$) and Euler angles ($\vec{\theta}$). Right column: Translational and angular velocity ($\vec{V}$ and $\vec{\omega}$, respectively).

In the second example, the side force is applied for 3 seconds while the physical parameters and initial conditions presented in Table 2.2 are unchanged. Figure 2.7 delineates the

trajectory of the multibody over a period of 7 seconds. Location 1 shows a trend where the slung payload exhibits a yawing angle in the X-Y plane. This trend is more pronounced in this example owing to momentum gained from the applied side force. The ramifications of applying side force for longer periods can also be seen in Fig. 2.8 where the payload exhibits an increasing velocity envelope in the lateral direction. Moreover, since the side force is applied for a longer period, the blimp undergoes a larger yaw angles compared to the previous example. As a result, the axial component of the side force, projected to the body frame, increases. This yields a slight increase in the axial velocity of the blimp and gondola, which is demonstrated in Fig. 2.8.

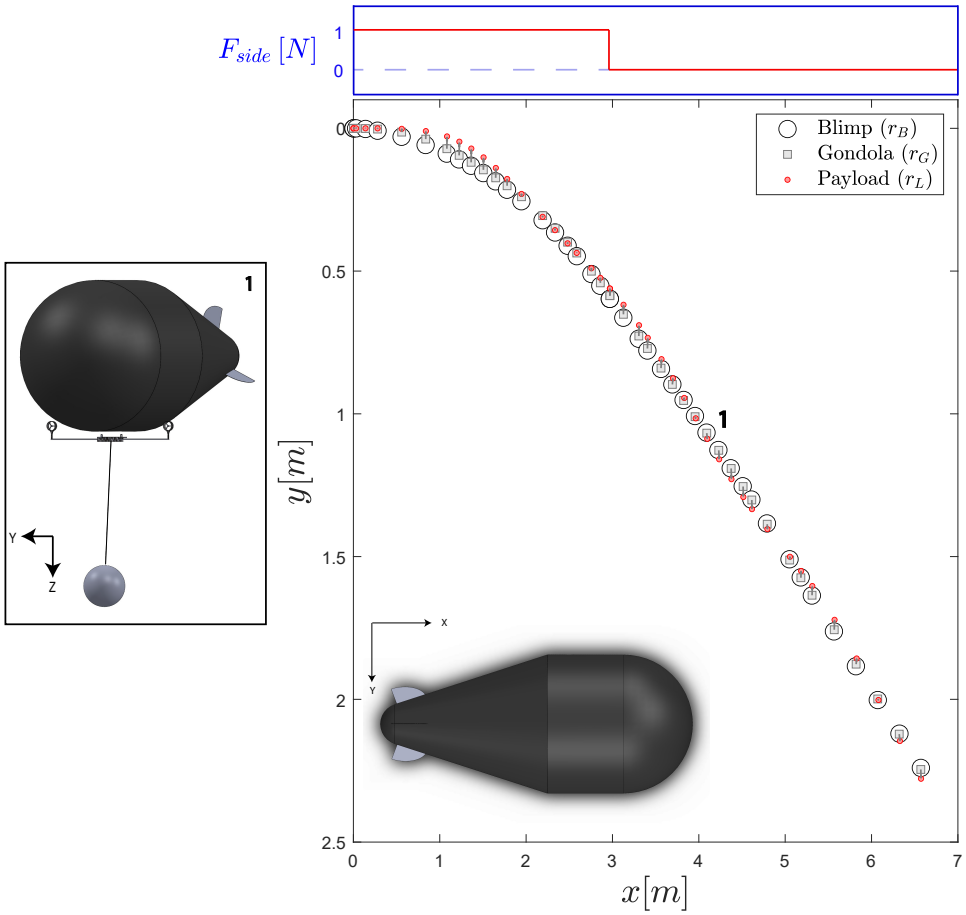


Figure 2.7: Trajectory produced by the second example. Position of blimp (circle), gondola (square), and payload (red small circle) viewed in X-Y plane. The applied side force over time is illustrated on top.

It is worth noting that Euler angles ($\vec{\theta}_L$) and angular velocity ($\vec{\omega}_L$) are calculated about the payload CG. Since the cable is assumed to be connected at the CG of the payload, the cable will not exert any moment about the payload CG. However, the swinging of the slung payload, demonstrated in Fig. 2.5-2.8, is calculated about the load vector $\vec{L}$ (2.27) using (2.42).

$$\vec{\theta}_c = \begin{bmatrix} \arctan\left(\frac{\vec{L}(2)}{\vec{L}(3)}\right) \\ \arctan\left(\frac{\vec{L}(1)}{\vec{L}(3)}\right) \\ \psi_L \end{bmatrix} \tag{2.42}$$

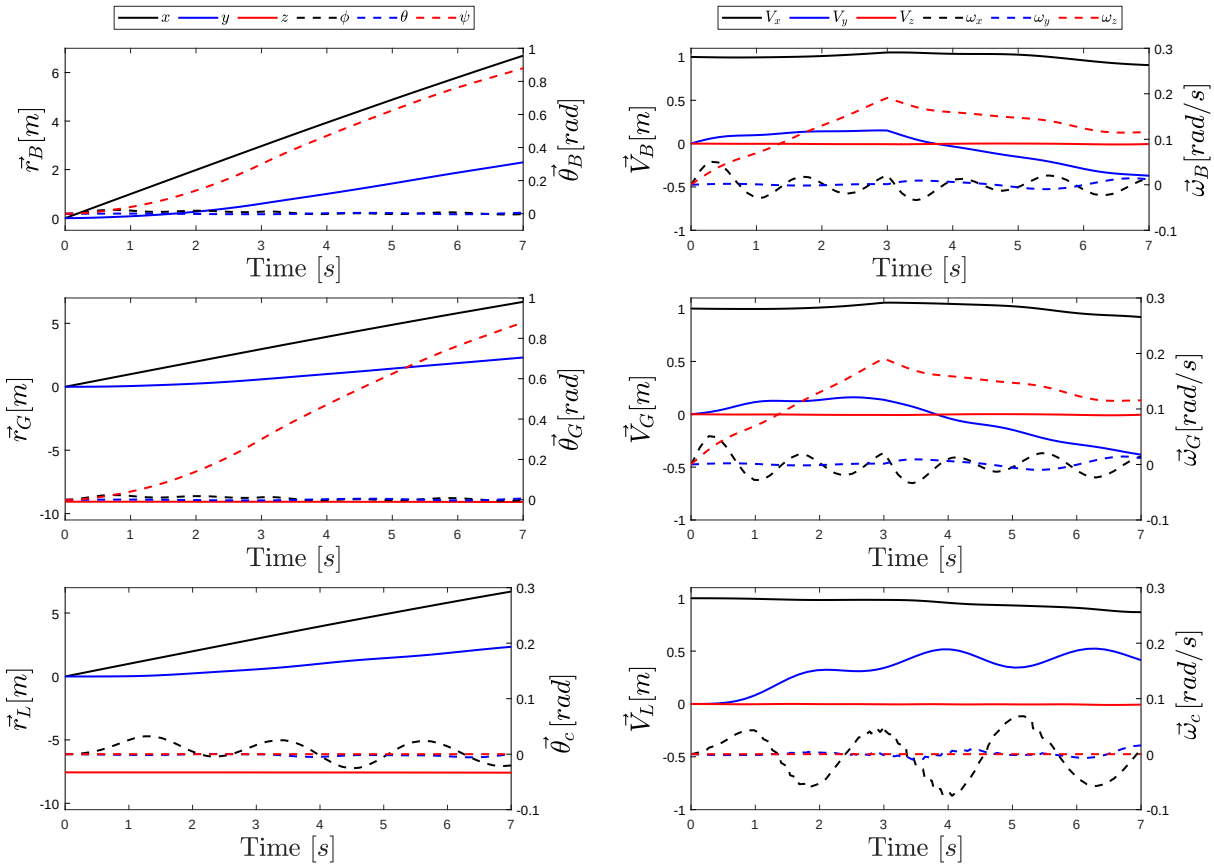


Figure 2.8: Results obtained from the second example. Top to bottom: blimp, gondola, and payload, respectively. Left column: Position ($\vec{r}$) and Euler angles ($\vec{\theta}$). Right column: Translational and angular velocity ($\vec{V}$ and $\vec{\omega}$, respectively).

2.3.1 Accumulation of numerical integration errors

While the UK method is relatively easy to implement, the errors caused by numerical integration need to be addressed. Errors from numerical integration are often categorized into three types: round off, local, and global errors. The round off errors are mainly attributed to the machine precision and is often overcome by employing double precision format in the machine memory. On the other hand, local and global errors are caused by the discretization scheme (order of Taylor series truncation) and step size. These errors can be reduced by using higher order schemes but will always exist nevertheless. Since the UK method imposes constraints on the acceleration level, the constraints at the position level would drift away with time due to numerical integration errors (namely local and global errors). Various methods were proposed to remedy this issue and ensure that the constraints are satisfied throughout the simulation. Baumgarte [62] proposed a method to stabilize the errors associated with integrating a second order differential equation of the form (2.43). Since the constraint equations used in the UK approach are also second order differential equations of the form (2.43), Baumgarte's method can be applied to stabilize the constraint equations. A more recent approach was proposed by Bisgaard [63] for slung payload constraint. The method assumes a virtual spring-damper which applies a force/moment proportional to the error of computed wire length and its rate of change with time[8]. The spring-damper forces/moments are applied to the dynamic equation, not the constraint equations.

Since the investigated multibody system has various types of constraints (*i.e.,* not only length constraints), Baumgarte's method is favourable. Hence, Baumgarte's stabilization approach is applied to the blimp-gondola-slung payload system.

[8]The wire length and its rate of change are the constraints at the position and velocity levels, respectively, for the slung payload constraint

Baumgarte's stabilization method

The method relies on a simple state feedback approach where the constraint equations are modified from (2.43) to (2.44)

$$\ddot{\vec{C}}_i = 0 \tag{2.43}$$

$$\ddot{\vec{C}}_i = -2\alpha_i \, \dot{\vec{C}}_i - \beta_i^2 \, \vec{C}_i \tag{2.44}$$

where α_i and β_i are tunable feedback gains[9] and i is used to numerate the constraints. The state equation is given as

$$\begin{bmatrix} \dot{\vec{C}} \\ \ddot{\vec{C}} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\beta_i^2 & -2\alpha_i \end{bmatrix} \begin{bmatrix} \vec{C} \\ \dot{\vec{C}} \end{bmatrix}$$

Substituting (2.44) into (2.10) yields

$$\mathbf{A} \, \ddot{\vec{q}} = \vec{b} - 2\alpha_i \, \dot{\vec{C}}_i - \beta_i^2 \, \vec{C}_i \tag{2.45}$$

Recall that $\mathbf{A}$ and $\vec{b}$ combine all constraints as in (2.35). Similarly, $\vec{C}$, $\dot{\vec{C}}$, $\boldsymbol{\alpha}$, and $\boldsymbol{\beta}$ should have a similar structure

$$\vec{C} = \begin{bmatrix} \vec{C}_1 \\ \vec{C}_2 \\ \dots \\ \vec{C}_p \end{bmatrix} \qquad \dot{\vec{C}} = \begin{bmatrix} \dot{\vec{C}}_1 \\ \dot{\vec{C}}_2 \\ \vdots \\ \dot{\vec{C}}_p \end{bmatrix}$$

$$\boldsymbol{\alpha} = 2 \begin{bmatrix} \alpha_1 & 0 & \dots & 0 \\ 0 & \alpha_2 & \dots & 0 \\ & & \ddots & \\ 0 & 0 & \dots & \alpha_p \end{bmatrix} \qquad \boldsymbol{\beta} = \begin{bmatrix} \beta_1^2 & 0 & \dots & 0 \\ 0 & \beta_2^2 & \dots & 0 \\ & & \ddots & \\ 0 & 0 & \dots & \beta_p^2 \end{bmatrix}$$

where p is the row-space of $\mathbf{A}$, $\boldsymbol{\alpha}$ and $\boldsymbol{\beta} \in \mathbb{R}^{p \times p}$. It is usually common to choose a single α and β for a constraint despite its dimension. For example, if the constraint is a three dimension position constraint ($\vec{C} \in \mathbb{R}^{3\times 1}$), then $\alpha_1 = \alpha_2 = \alpha_3$ and $\beta 1 = \beta 2 = \beta 3$. The values of α and β are given in Table 2.3

Finally, the UK equation with stabilization terms becomes

$$\ddot{\vec{q}} = \mathbf{M^{-1}} \, \vec{Q} + \mathbf{M}^{-\frac{1}{2}} \, (\mathbf{A} \, \mathbf{M}^{-\frac{1}{2}})^{\dagger} \, (\vec{b} - \boldsymbol{\alpha} \, \dot{\vec{C}} - \boldsymbol{\beta} \, \vec{C} - \mathbf{A} \, \mathbf{M^{-1}} \, \vec{Q}) \tag{2.46}$$

[9] Refer to [64] for a systematic approach to tune the feedback gains.

Table 2.3: Numerical stabilization terms

Parameter	Value
$\alpha_1, \alpha_2, \alpha_3$	4
$\alpha_4, \alpha_5, \alpha_6$	0
α_7	4
$\beta 1, \beta 2, \beta 3$	10
$\beta 4, \beta 5, \beta 6$	0
$\beta 7$	10

2.4 Contributions

The Udwadia-Kalaba method was used to model the dynamics of the multibody blimp-gondola-slung-payload system. This technique provides a convenient means to satisfy both holonomic and nonholonomic constraints where the constraint forces are represented explicitly, solely relying on the generalized coordinates and their time derivatives (velocities), without necessitating any additional auxiliary variables. Three constraints were derived and imposed: position and orientation constraints between the airship and gondola, and a length constraint between the gondola and slung-payload. Deriving and implementing the blimp-gondola constraints presented two primary challenges: first, the gondola constraints were actuation-dependent. The second challenge having a piecewise continuous rail with a curved section. The first challenge was addressed by incorporating the distance travelled along the gondola (Ss) as measured by an encoder. To tackle the second challenge, the position vector between the blimp and the gondola was reformulated by introducing a rail angle (λ), which is a function of the distance travelled along the curved rail.

To conclude, this chapter provides a detailed guide on applying the Udwadia-Kalaba method to a multibody dynamical system connected with various types of constraints and offers valuable insights into addressing actuation-controlled constraints.

Chapter 3

System Identification

This chapter discusses various methods from the literature to identify the physical parameters of single and multibody systems. Calculation of physical parameters such as the inertia tensor is essential to accurately describe the dynamic model of a system. Unfortunately, the calculated mass and inertia can differ substantially from those measured experimentally. While the parameters computed from CAD generated models and the actual prototype are expected sources of discrepancies, lighter-than-air (LTA) vehicles are subjected to the added mass (virtual mass) and added inertia which are estimated based on the airship geometry (as discussed in Section 2.1.1). Therefore, common system identification methods are first highlighted then a new approach is proposed to identify the mass matrix from experimental data using Semi-Definite Programming (SDP) with linear equality and inequality constraints.

3.1 Background and Literature Review

The dynamic modelling of multibody systems involves various physical parameters, including the inertia tensor and virtual mass/inertia in LTA application. Calculating the latter involves assumptions such as potential flow as well as geometric simplifications (refer to Chapter 2 for further details). Similarly, the inertia tensor is often estimated using geometric modelling and approximations followed by refinements throughout experimentation.

The problem of identifying unknown parameters in the system can be formulated in two ways: inverse dynamics identification and direct dynamics identification [65]. In the inverse dynamic formulation, generalized forces are described as a function of the system position, velocity, and acceleration. On the other hand, the acceleration of the system is described as a function of the system position, velocity, and generalized forces in the direct dynamic formulation.

The most common approach for estimating inertial parameters is the least-squares method [66], [65] which is based on regression techniques and it is often formulated as inverse dynamics identification problem[1]. There are many variations of least-squares method such as weighted least-squares and total least-squares [67]. Recursive least-squares was used to estimate the inertia parameters of UAV helicopter based on simulation results [68]. While the estimated inertia tensor was quite close to the UAV manufacturer values, adding noise to the measurements deteriorated the estimation. Least-squares is known to be sensitive to noisy measurements which can substantially affect the fidelity of the parameter estimation [65],[68]. Moreover, Extended Kalman Filter was used to identify inertia parameters and control gains of a multirotor UAV with adjustable rotor arms' length [69]. To excite the system, the authors had to develop multiple dynamical models for each experimental test where some of the tests had the UAV pinned with an axial bearing to constrain the motion. The identified parameters were used in a simulation and showed good agreement with experimental data. While the proposed method was capable of identifying the system parameters, including control gains, it is cumbersome to carry out. Furthermore, [70] identified the inertia of a small UAV quadrotor by examining the response of the closed-loop control system in the frequency-domain. While frequency-domain techniques are generally more robust to noise in measured data, the resulting dynamic model has to be linearized in the time domain.

Recently, optimization-based approaches are being utilized to identify the system parameters. [71] formulated the identification problem as a Semi-Definite Program (SDP) with linear matrix inequality constraints for links of a robot. Eigenvalue decomposition was

[1]Since the unknown parameters, namely mass and inertia, are linear in the inverse dynamic formulation, the identification problem can be represented as a linear least-square regression

applied on the inertia tensor and triangle inequalities were enforced on the diagonal eigenvalue matrix. However, it is not clear how would the proposed identification method work when the system is subject to nonholonomic constraints.

The following section discusses a simpler approach to identify the mass matrix of a multibody system with nonholonomic constraints using SDP.

3.2 Methodology

The author proposes to identify the mass matrix elements of a multibody system using SDP with equality and inequality constraints. The technique that is being proposed utilizes the structure of the mass matrix to identify the inertia, added inertia, and added mass of the blimp-gondola system.

3.2.1 Mathematical Formulation

To carry out the system identification process, the EOM derived in the previous chapter (Chapter 2) is recalled.

$$\ddot{\vec{q}} = \mathbf{M^{-1}}\ \vec{Q} + \mathbf{M}^{-\frac{1}{2}}\ (\mathbf{A}\ \mathbf{M}^{-\frac{1}{2}})^{\dagger}\ (\vec{b} - \mathbf{A}\ \mathbf{M^{-1}}\ \vec{Q}) \tag{3.1}$$

Using (3.1) would yield a direct dynamics identification problem. The mass matrix ($\mathbf{M}$) contains the unknown parameters, such as the inertia as well as the virtual mass and virtual inertia of the blimp, that are ought to be identified. However, there are two challenges with the current form. The first challenge is having the inverse of the unknown parameter (the mass matrix) present in the equations which hinders the SDP formulation. The second challenge is having a pseudo-inverse of the unknown parameter. Since the inverse of a positive-definite matrix is also a positive-definite, the first challenge can be easily resolved by changing the unknown parameter from $\mathbf{M}$ to $\hat{\mathbf{M}}$ where

$$\hat{\mathbf{M}} = \mathbf{M^{-1}} \tag{3.2}$$

To overcome the second challenge, the pseudo-inverse term in (3.1) is substituted with (3.3)

$$\mathbf{M}^{-\frac{1}{2}}\ (\mathbf{A}\ \mathbf{M}^{-\frac{1}{2}})^{\dagger} = \mathbf{K}\ \mathbf{A}^{\dagger} \tag{3.3}$$

where

$$\mathbf{K} = \hat{\mathbf{M}}^{\frac{1}{2}}\ (\mathbf{A}\ \hat{\mathbf{M}}^{\frac{1}{2}})^{\dagger}\ \mathbf{A} \tag{3.4}$$

and $\mathbf{K}$ is solved iteratively, as will be discussed later. Therefore, (3.1) becomes

$$\ddot{\vec{q}} = \hat{\mathbf{M}}\ \vec{Q} + \mathbf{K}\ \mathbf{A}^{\dagger}\ (\vec{b} - \mathbf{A}\ \hat{\mathbf{M}}\ \vec{Q}) \tag{3.5}$$

Equation (3.5) is now in a form that can be used in SDP to identify the inverse mass matrix ($\hat{\mathbf{M}}$). The SDP problem is introduced as follows

$$\begin{aligned} \text{minimize} \quad & \text{cost function} \\ \text{subjected to} \quad & \hat{\mathbf{M}} \succeq 0 && (3.6) \\ & \hat{\mathbf{M}}(i,j) = c && (3.7) \\ & \left|\frac{\ddot{\vec{q}}_{exp} - \ddot{\vec{q}}}{\ddot{\vec{q}}_{exp}}\right| = \vec{e} && (3.8) \end{aligned}$$

where $\ddot{\vec{q}}_{exp}$ is the acceleration obtained from experiments and $\ddot{\vec{q}}$ is given in (3.5), $\vec{e}$ is a vector containing the error tolerance that is set by the user, and c is a constant value. The inequality constraint enforces $\hat{\mathbf{M}}$ to be a semi-definite matrix and the equality constraint enforces the elements of $\hat{\mathbf{M}}$ to be constant values[2]. Equality constraints considered in this application include fixing the mass of the gondola and equating the blimp's moment of inertia in the lateral axis (I_{By}) to that in the vertical axis (I_{Bz}). Since this is a feasibility problem, the cost function is set to zero. The algorithm rejects solutions if the experimental acceleration is zero ($\ddot{\vec{q}}_{exp} = 0$) and the numerator is nonzero in (3.8).

The SDP problem is inserted into Drake [72] and solved using MOSEK optimization package [73]. Drake is an open source toolbox that offers a convenient language to formulate optimization problems then calls upon suitable optimization packages (such as MOSEK, Gurobi, SNOPT, Ipopt, etc) to solve them. The system ID algorithm is explained in Algorithm 1 where ($\mathbf{K}$) is initialized using an estimated mass matrix followed by initialization

[2]Constant values include the measured mass of gondola and zeroes where needs to be (refer to (2.36))

of (Δ) which represents the convergence of the mass matrix to an acceptable (ϵ[3]) value that is defined by the user. The convergence of the mass matrix (Δ) is normalized by the elements of the estimated mass matrix. The estimated mass matrix may be obtained from theoretical or semi-empirical work. The operation (./) refers to element-wise division and $\mathbf{M_{CAD}}$ is the mass matrix as approximated by CAD. The tilde symbol indicates a flattened matrix, representing the transformation of a multidimensional array into a one-dimensional vector while eliminating all zero-value elements.

Algorithm 1 System ID using SDP

1: Initialize $\mathbf{K}$ (3.4) with an estimation of $\hat{\mathbf{M}}$
2: Initialize Δ with a nonzero value.
3: **while** $\Delta > \epsilon$ **do**
4: Enforce inequality (3.6) and equality constraints (3.7-3.8)
5: Solve the SDP problem to find $\hat{\mathbf{M}}_{\mathbf{new}}$
6: Calculate $\mathbf{K}_{\mathbf{new}}$ based on $\hat{\mathbf{M}}_{\mathbf{new}}$
7: Define $\tilde{\hat{\mathbf{M}}}$, $\tilde{\hat{\mathbf{M}}}_{\mathbf{new}}$, and $\tilde{\mathbf{M}}_{\mathbf{CAD}}$ as the flattened versions of the corresponding matrices where all zero-value elements are eliminated.
8: Update $\Delta = \sum(|\tilde{\hat{\mathbf{M}}}_{\mathbf{new}} - \tilde{\hat{\mathbf{M}}}|./|\tilde{\mathbf{M}}_{\mathbf{CAD}}|)$
9: Set $\hat{\mathbf{M}} = \hat{\mathbf{M}}_{\mathbf{new}}$ and $\mathbf{K} = \mathbf{K}_{\mathbf{new}}$
10: **end while**

3.2.2 Experimentation and Implementation

As explained in the previous section, SDP is utilized to identify the mass matrix by minimizing the error between the acceleration predicted by the model ($\dddot{\vec{q}}$) and the acceleration obtained experimentally ($\dddot{\vec{q}}_{exp}$). Three experiments were carried out to validate the dynamical model derived in Chapter 2 and discover the unknown parameters in the system, namely the moment of inertia, added mass and added inertia factors of the blimp. Experiments were carried out inside the Autonomous Agent Flight Lab at the University

[3] will be referred to as iteration tolerance

of Ottawa using an OptiTrack motion capture system, depicted in Fig. 3.1. The blimp and gondola are treated as two rigid bodies with two representative points in the motion capture software. The representative point is determined based on the centre of the body shape that is being tracked. The number of reflectors attached to the body tends to affect the location of the body centre. However, the dependence of body centre on the number of reflectors becomes insignificant when sufficient reflectors are used. The blimp featured over twenty reflectors whereas the gondola had ten, strategically distributed to ensure optimal tracking by the cameras throughout the tests.

Figure 3.1: Physical prototype inside the experiment workspace. The cameras, emitting blue light, are used to capture the motion of the multi-body.

The experiments were designed to excite various states of the system while respecting the size limitations of the workspace. Owing to space limitation of the workspace, the trajectory of each experiment is presented over a period of 5 seconds. The applied force in each experiment is described as follows:

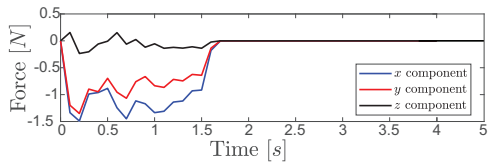
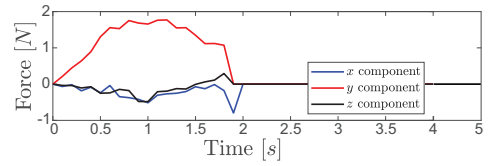
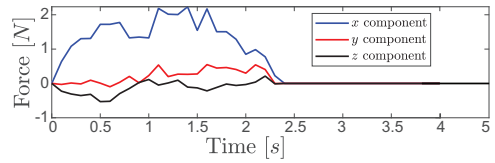
1. In Experiment 1, a force is applied on the gondola at an angle $\approx 45^\circ$ in $X-Y$ plane for 1.6 seconds.

2. In Experiment 2, a lateral force (Y direction) is applied to the blimp for 1.9 seconds.

3. In Experiment 3, the multibody system was given an initial velocity. In addition, a force was applied for 2.3 seconds at the gondola's arm.

Experiment 1 was carried out by applying a force on the gondola C.G and the force was exerted on the blimp in Experiment 2. Both experiments started with stationary initial conditions. Experiment 3 was initiated with a non-zero velocity and the force was exerted on the propeller arm. The characteristics of each experiment and location of applied force are listed in Table 3.1.

Table 3.1: Initial conditions and force application during experiments. The location of applied force ($\vec{r}_{force}$) is given in body frame.

Exp.	Initial Velocity	Force Direction and Time	Force Location
1	$\vec{V}_B = \vec{V}_G = [0,0,0]^T$		$\vec{r}_{force} = \begin{bmatrix} 0.2072 \\ 0.1331 \\ 0.0962 \end{bmatrix}$
2	$\vec{V}_B = \vec{V}_G = [0,0,0]^T$		$\vec{r}_{force} = \begin{bmatrix} -0.0210 \\ 0.8514 \\ -0.1975 \end{bmatrix}$
3	$\vec{V}_B = \begin{bmatrix} 0.0258 \\ -0.0145 \\ -0.0384 \end{bmatrix}$ $\vec{V}_G = \begin{bmatrix} -0.0311 \\ -0.0202 \\ -0.034 \end{bmatrix}$		$\vec{r}_{force} = \begin{bmatrix} -0.0077 \\ -0.7046 \\ 0.0371 \end{bmatrix}$

It is worth noting that all experiments were performed in open loop with no control actions from the thrusters. The gondola was positioned on the rail manually to obtain a zero degree pitch of the blimp. Additional weight was placed on top of the gondola, near

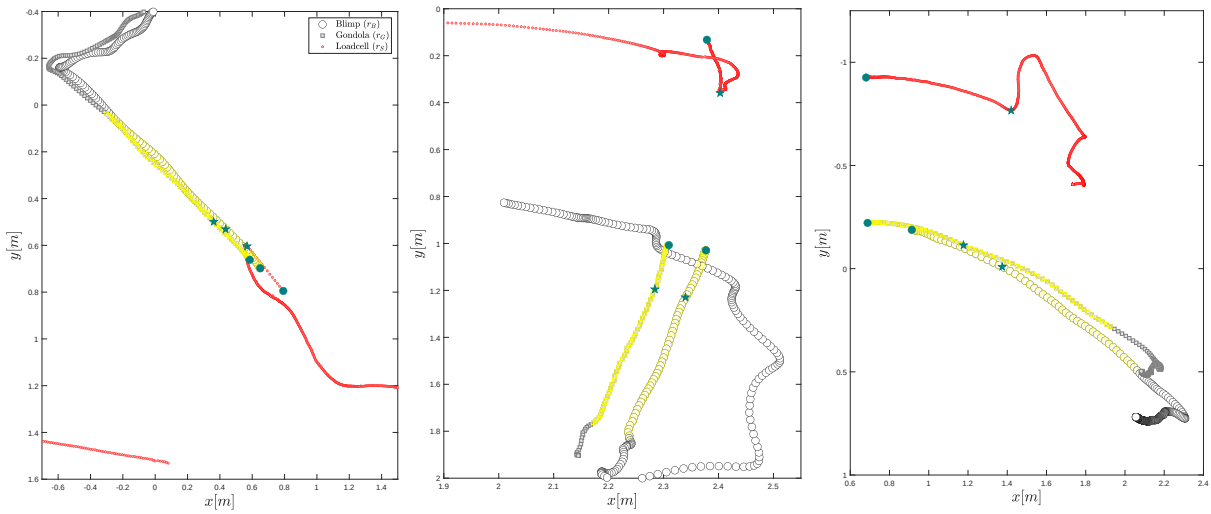
the gondola's CG, to maintain a constant altitude.

The applied forces are measured using a custom-designed load cell and manually exerted by hand. Since the load cell only provides the force magnitude, it was necessary to trace the trajectory of the load cell using the motion capture system to extract the direction of the applied force. Hence, a total of four reflectors were attached to the load cell mount. Differentiating the trajectory of the load cell with time and normalizing yields a vector with a unit magnitude at each time step. The applied force magnitude, obtained from load cell, is then projected into the resulting unit vectors to get the applied force vectors in time. The load cell mount design is shown in Appendix B. The sampling time of the motion capture system and the load cell is 8 ms and 100 ms, respectively. Figure 3.2 depicts the trajectory of experiments one, two, and three. The start and end of the applied force are indicated by a green circle and a star, respectively. The red symbols represent the trajectory of the load cell, while the black symbols depict the trajectory of the blimp. Points highlighted in yellow denote the duration of data collection, providing a clear visualization of the experiments' timeline.

The motion capture system relies on tracing reflectors that are attached to the investigated bodies. If the reflectors are out of the cameras' sight, as illustrated in Fig. 3.2a between $y = 1.2$ m to $y = 1.45$ m for the load cell, the tracing is lost. Nevertheless, the number of points collected throughout the experiments were sufficient to trace the trajectories of the blimp, gondola, and the load cell.

3.3 Results and Discussion

Data collected from experiments, explained in previous section, are presented here for discussion. The motion capture system provides the position and orientation of each body involved in the experiment (*i.e.,* blimp, gondola, and load cell). Velocities and accelerations are computed numerically using a second order finite difference scheme. To highlight the orientation of the blimp, connection lines between the blimp and gondola were added on



(a) Experiment 1. (b) Experiment 2. (c) Experiment 3.

Figure 3.2: Trajectory obtained from experiment 1 to 3. Points highlighted in yellow elucidate the period of data collection, while the green circle and star mark the beginning and end of the force application, respectively.

top of the figures. To ensure clarity and avoid crowded figures, the results presented are a subset of the collected data points.

To identify the parameters inside the mass matrix (inertia, added mass, and added inertia of the blimp), matrix ($\mathbf{K}$) (3.4) was initialized based on an estimation of mass matrix. The estimation of ($\hat{\mathbf{M}}$) is obtained from previous work [16] which is based on Solidworks. Error tolerance (e) and iteration tolerance (ϵ) were set to 0.05 (5%) and 0.01 (1%), respectively. These values were chosen as they represent the lowest thresholds at which the solver was able to successfully converge to a solution. The system identification process was carried out on data acquired from experiments one, two, and three. Mass matrices identified from Experiment 1, 2, and 3 will be referred to as MM **(1)**, MM **(2)**, and MM **(3)**, respectively. To assess each identified mass matrix, the trajectory of each experiment is reconstructed using MM **(1)**, MM **(2)**, and MM **(3)**. The reconstructed trajectories are then compared with the experimental data. Normalized root mean squared error (NRMSE) is calculated for all cases in order to quantify the difference between experimental results and reconstructed trajectories. The root mean squared error (RMSE), normalized root

mean squared error (NRMSE), and distance travelled (d) are defined in (3.9), (3.10), and (3.11), respectively.

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(\|(\vec{r}_B)_{Exp}[i] - (\vec{r}_B)_{Sim}[i]\|^2 + \|(\vec{r}_G)_{Exp}[i] - (\vec{r}_G)_{Sim}[i]\|^2\right)} \tag{3.9}$$

$$NRMSE = \frac{RMSE}{d} \tag{3.10}$$

$$d = \sum_{i=1}^{N-1} \|(\vec{r}_B)_{Exp}[i+1] - (\vec{r}_B)_{Exp}[i]\|^2 \tag{3.11}$$

where the subscripts Exp and Sim refer to experiment and simulation results, respectively. Distance travelled by the airship throughout the experiment is denoted by d, index i refers to timestep and N is the total number of timesteps. RMSE is normalized so that the error is represented as a percentage of distance travelled. To maintain clarity and cohesion, the figures presented here will focus solely on the X-Y plane. The trajectories in both the $X-Y$ and $X-Z$ planes are provided in Appendix B.2. The identified mass matrix that yields the minimal RMSE (3.9), summed over all trajectories, is selected to be the mass matrix of the multibody system. Lastly, trajectories simulated using the selected mass matrix are compared with simulations before performing system ID. It is important to highlight that all system identification tests were conducted at low speeds due to the space limitations of the workspace.

Experiment 1

In Experiment 1, a force is applied on the gondola at an angle $\approx 45^\circ$ in $X-Y$ plane for 1.6 seconds. The identified mass matrix from Experiment 1, 2, and 3 (MM **(1)**, MM **(2)**, and MM **(3)**, respectively) are used to reconstruct the trajectory of Experiment 1. The reconstructed trajectory of Experiment 1 is demonstrated in Fig. 3.3 and compared to experimental results. The reconstructed trajectories are in good agreement with the experiment where NRMSE is below 3% for all identified mass matrices. The lowest NRMSE is 2.55% which is achieved using MM **(3)** while the highest is 2.93% using MM **(2)**.

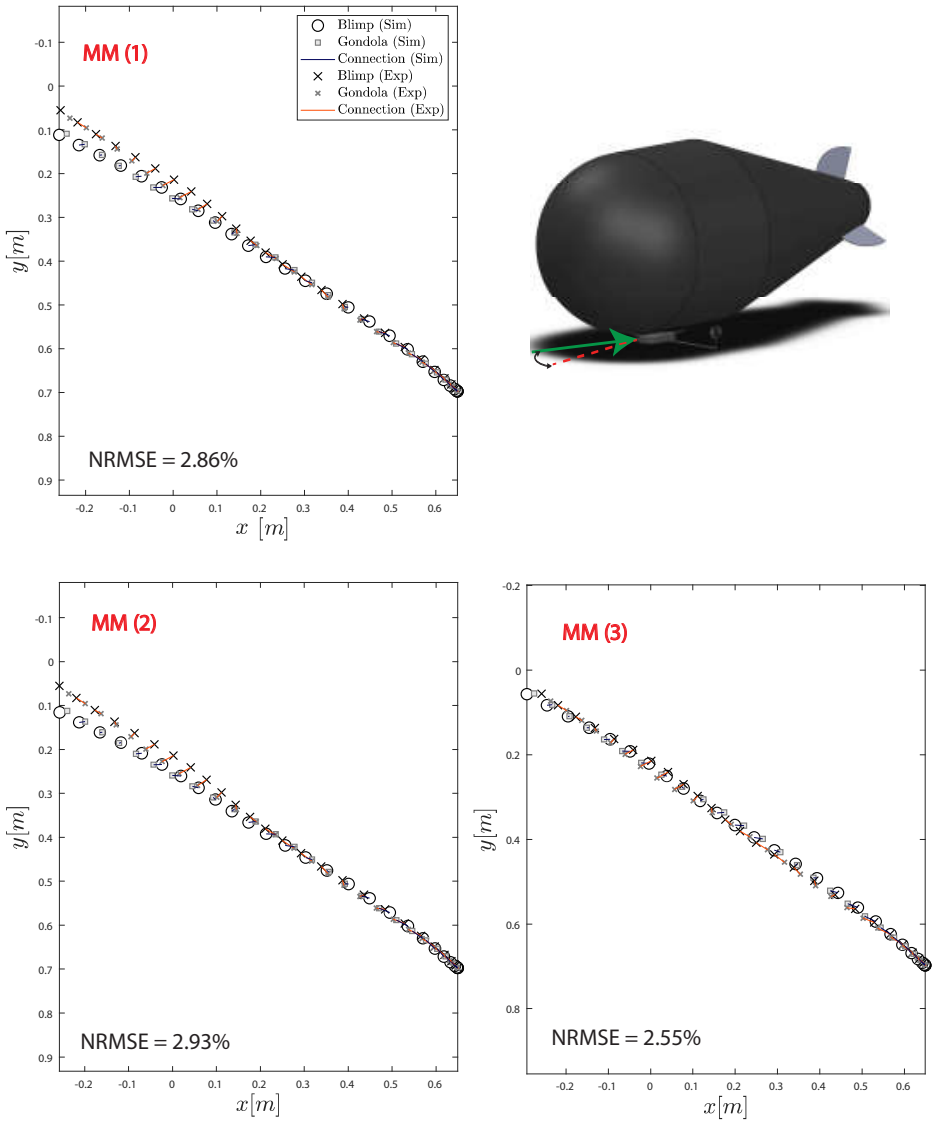


Figure 3.3: Reconstruction of Experiment 1 using MM **(1)**, MM **(2)**, and MM **(3)**. The applied force is displayed in the top right corner.

Experiment 2

A lateral force (Y direction) is applied directly to the blimp for 1.9 seconds in Experiment 2. From top to bottom, Fig. 3.4 illustrates the reconstructed trajectories of Experiment 2 using MM **(1)**, MM **(2)**, and MM **(3)**, respectively. The variation in NRMSE is marginal where the highest error was 1.1%.

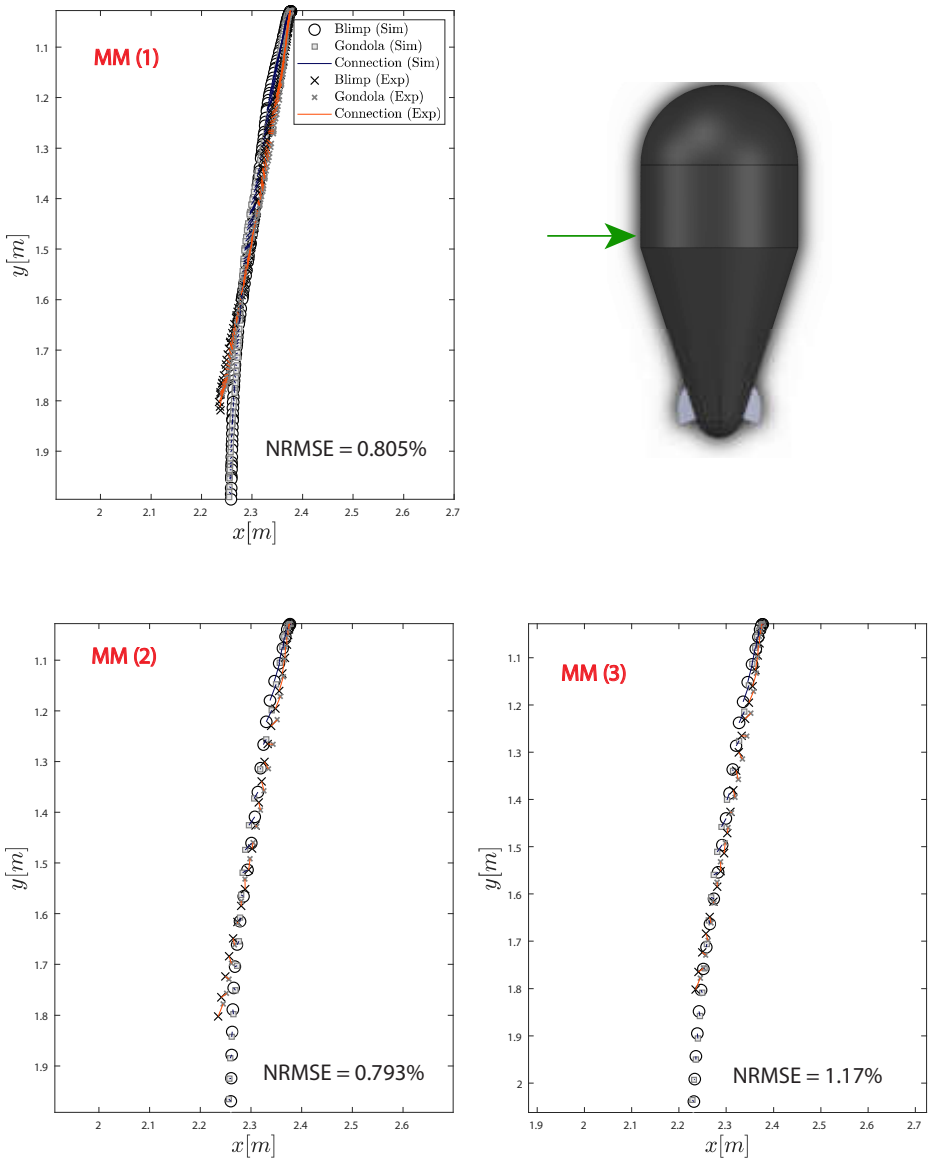


Figure 3.4: Results obtained from Experiment 2 using MM **(1)**, MM **(2)**, and MM **(3)**. The applied force is displayed in the top right corner.

Experiment 3

Experiment 3 exhibits a rotation about the Z axis (yawing moment). The force was applied for 2.3 seconds at a the propeller's arm which is connected to the gondola. In addition, the multibody system was given an initial velocity. Figure 3.5 demonstrates the reconstructed trajectory of Experiment 3 using MM **(1)** at top, MM **(2)** in the middle, and MM **(3)** at

the bottom. The minimal NRMSE was attained by MM **(3)** with an error percentage of 3.92% of the distance travelled while the error for MM **(1)** and MM **(2)** is around 5%.

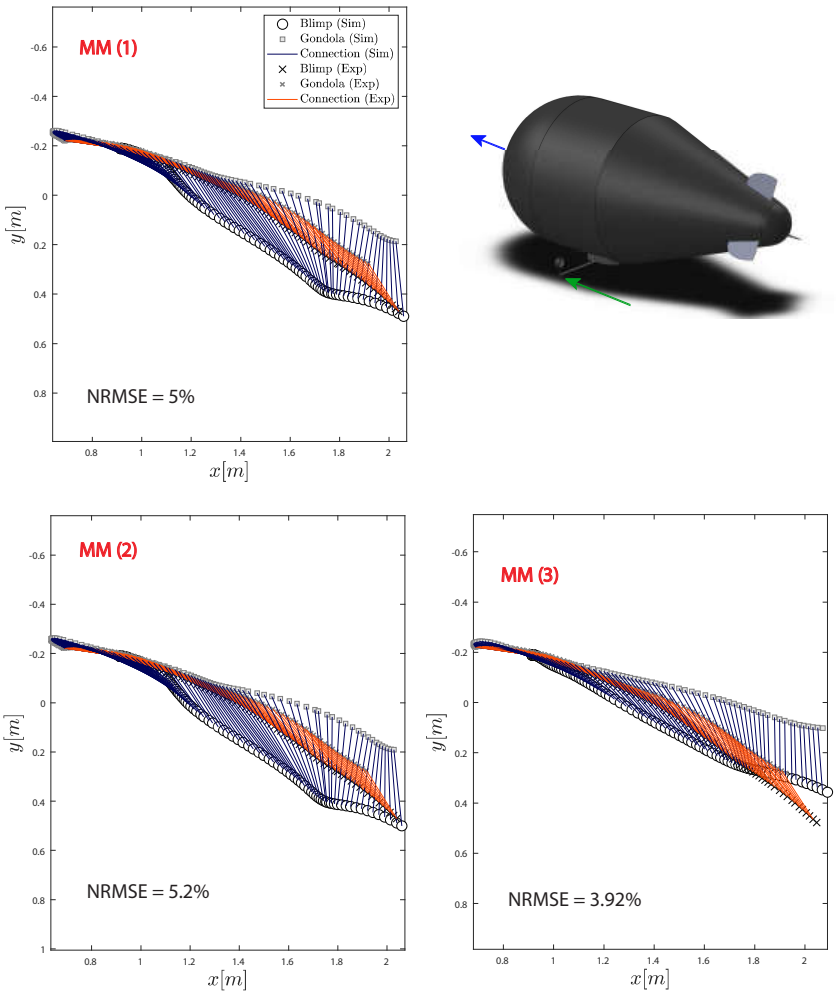


Figure 3.5: Results obtained from Experiment 3 using MM **(1)**, MM **(2)**, and MM **(3)**. The applied force is displayed in the top right corner where green and blue arrows represent force and initial velocity, respectively.

From the identified mass matrices (MM **(1)**, MM **(2)**, and MM **(3)**), the one that exhibits the minimal RMSE will be chosen. Figure 3.6 illustrates the RMSE that each mass matrix yields for each reconstructed trajectory. The highlighted bars represent the sum of RMSE that each mass matrix produces. While MM **(1)**, MM **(2)**, and MM **(3)**, respectively, indicate the mass matrices identified using Experiment 1, 2, and 3, MM **(0)**

refers to the initial mass matrix used before applying system ID. Clearly, MM **(3)** attains the lowest RMSE among other identified mass matrices. Using MM **(3)** reduces the RMSE by about 35% in comparison to MM **(0)** (mass matrix before system ID).

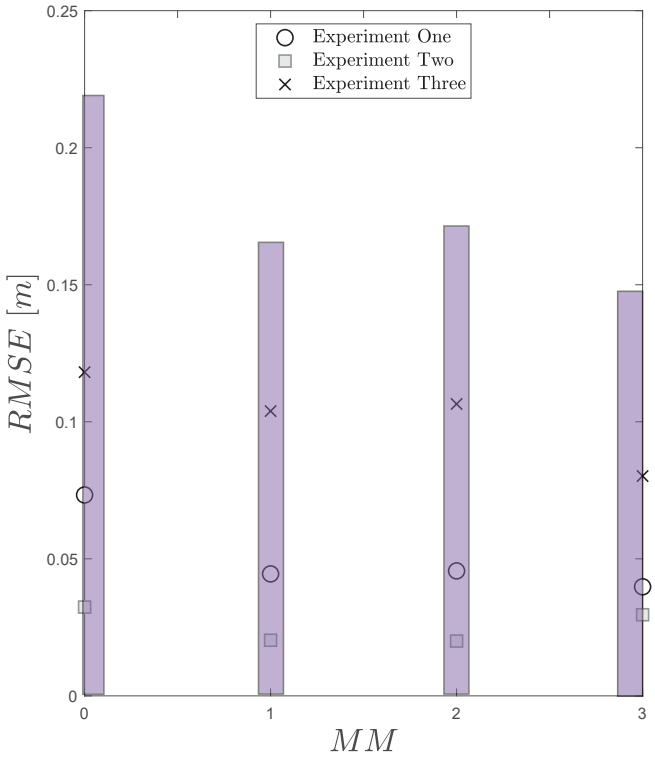


Figure 3.6: RMSE of simulated trajectories using MM **(0)** (before system ID), MM **(1)**, MM **(2)**, and MM **(3)**. Bars represent the sum of RMSE over all experiments.

The identified mass matrices from Experiments 1, 2, and 3 (MM**(1)**, MM**(2)**, and MM**(3)**, respectively) are shown in Table 3.2 and Table 3.3 for the blimp and gondola, respectively. It is worth reminding the reader that the values of mass and inertia vary along the principal axes of the blimp due to added mass and added inertia. While the variation in the blimp mass and inertia components is minor between MM **(1)** and MM **(2)**, the values obtained in MM **(3)** are relatively lower. However, the mass and inertia components of the gondola are quite similar among all experiments. Since the mass of the gondola is measured experimentally, it was imposed as a constraint during the formulation of the optimization problem. Hence, having identical mass components from all experiments was

expected for the gondola.

Table 3.2: Selected indices of identified mass matrices from Experiments 1, 2, and 3. The selected indices represent the mass and inertia components for the blimp.

	$\mathbf{M_B}$ (kg)			$\mathbf{I_B}$ (kg.m^2)		
MM	$\mathbf{M}(1,1)$	$\mathbf{M}(2,2)$	$\mathbf{M}(3,3)$	$\mathbf{M}(4,4)$	$\mathbf{M}(5,5)$	$\mathbf{M}(6,6)$
(1)	3.9761	6.4	6.4	0.9816	5.1	5.1
(2)	4	6.52	6.52	0.971	5.1	5.1
(3)	3.7518	4.84	4.84	0.95	4.094	4.094

Table 3.3: Selected indices of identified mass matrices from experiments one, two, and three. The selected indices represent the mass ($\mathbf{M}(7,7), \mathbf{M}(8,8)$, and $\mathbf{M}(9,9)$) and inertia ($\mathbf{M}(10,10), \mathbf{M}(11,11)$, and $\mathbf{M}(12,12)$) components for the gondola.

	$\mathbf{M_G}$(kg)			$\mathbf{I_G}$ (kg.m^2)		
MM	$\mathbf{M}(7,7)$	$\mathbf{M}(8,8)$	$\mathbf{M}(9,9)$	$\mathbf{M}(10,10)$	$\mathbf{M}(11,11)$	$\mathbf{M}(12,12)$
(1)	4.154	4.154	4.154	0.14105	0.1504	0.1504
(2)	4.154	4.154	4.154	0.133	0.147	0.147
(3)	4.154	4.154	4.154	0.1366	0.147	0.147

To evaluate the identified mass matrix, all experiments are simulated and compared with simulations before carrying out system ID, *i.e.,* using MM **(0)**. Figures 3.7-3.9 depict the reconstructed trajectories before system ID (on the left) and after system ID (on the right) along with experimental results. For Experiment 1 (Fig. 3.7), the NRMSE was 4.7% before system ID which reduces to 2.55% after applying system ID. The reconstructed trajectory for Experiment 2 demonstrates a reduction in NRMSE decreasing from 1.28% before system ID to 1.17% afterwards. A substantial improvement is observed in Experiment 3 where the NRMSE goes down from 5.77% to 3.92% after system ID. It is clear from the reconstructed trajectories that the identified mass matrix produces more accurate results with lower NRMSE in comparison with those produced before system ID.

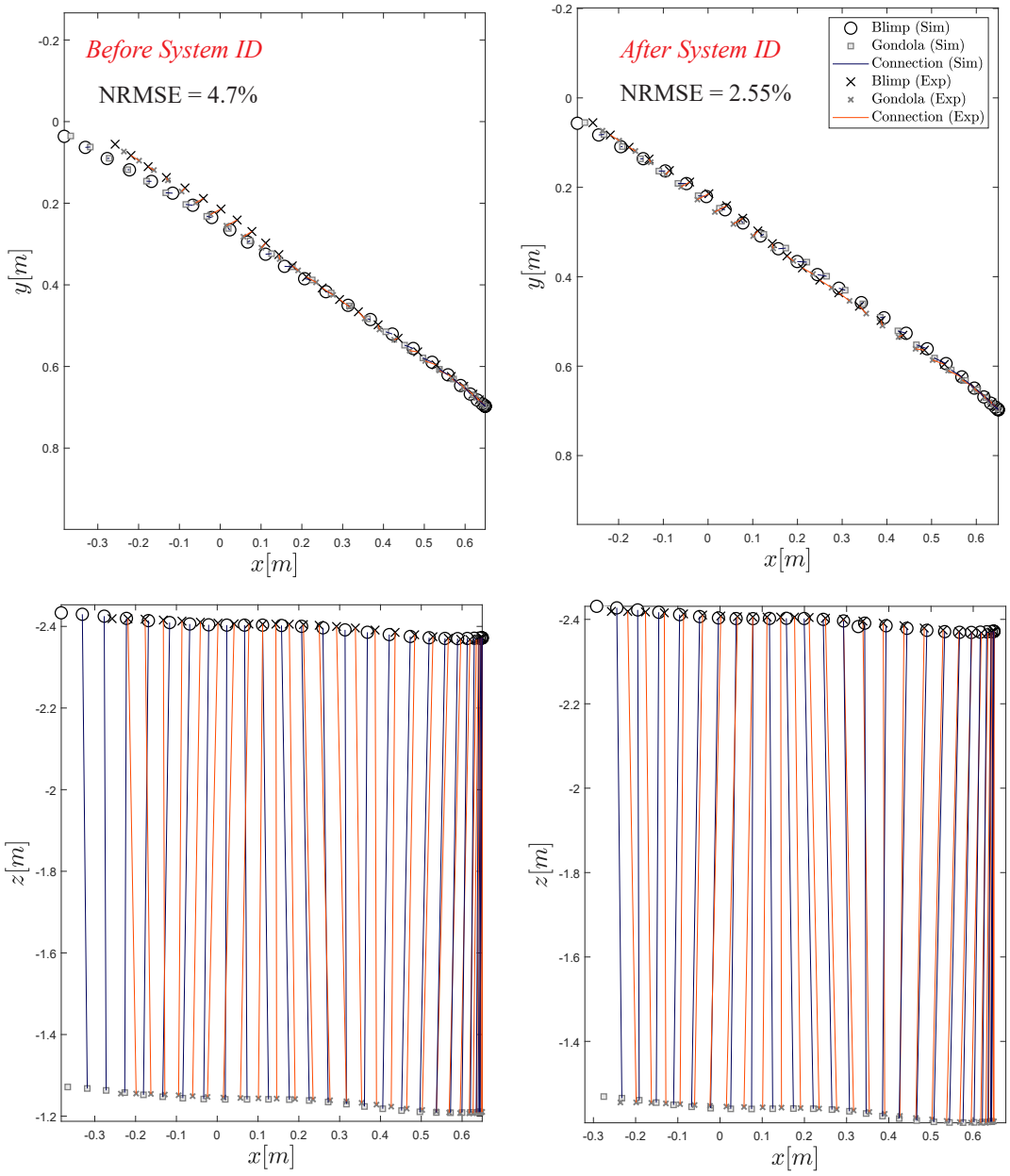


Figure 3.7: Results obtained from Experiment 1. Right and left columns are respectively the trajectories with and without system ID.

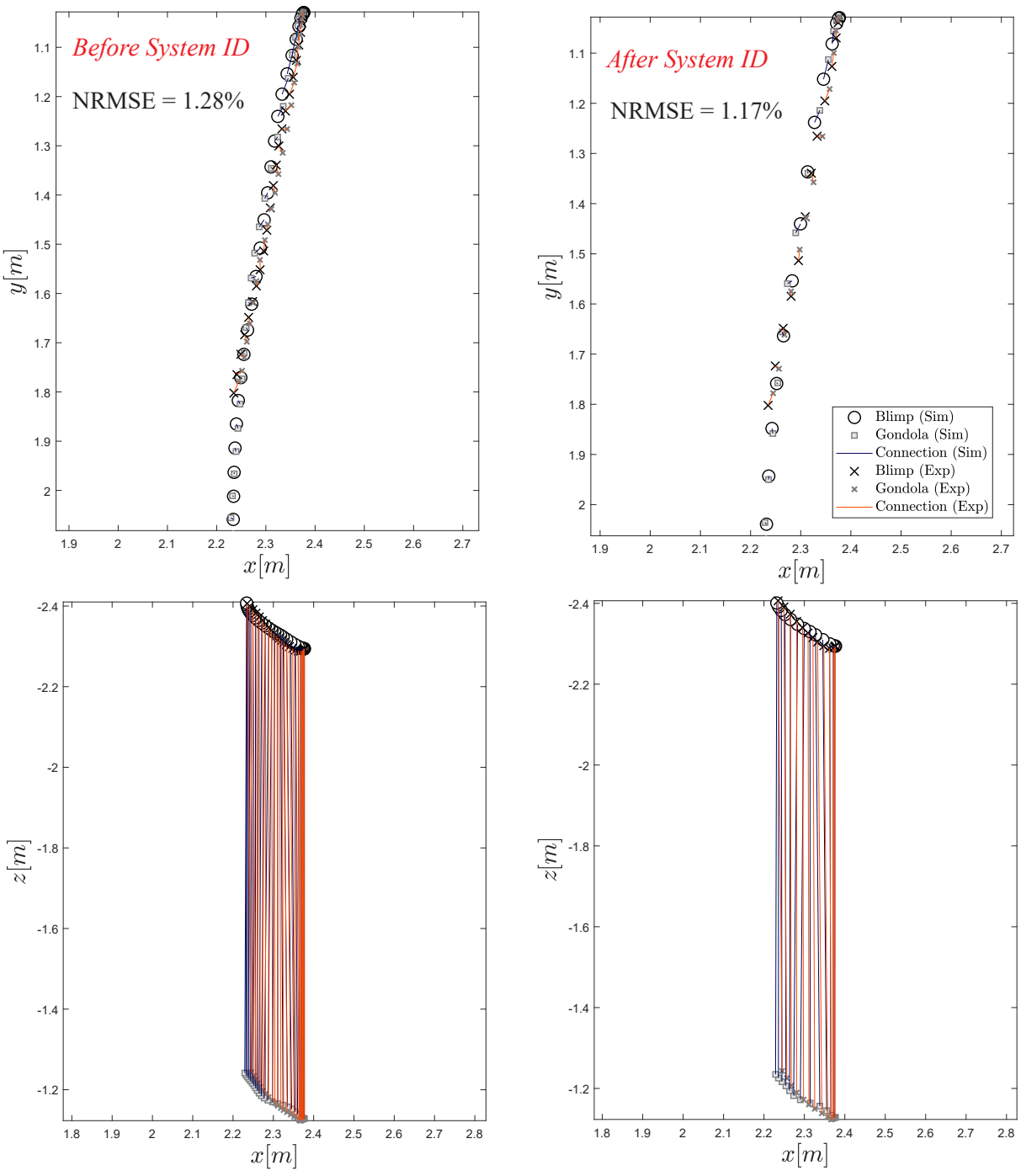


Figure 3.8: Results obtained from Experiment 2. Right and left columns are respectively the trajectories with and without system ID.

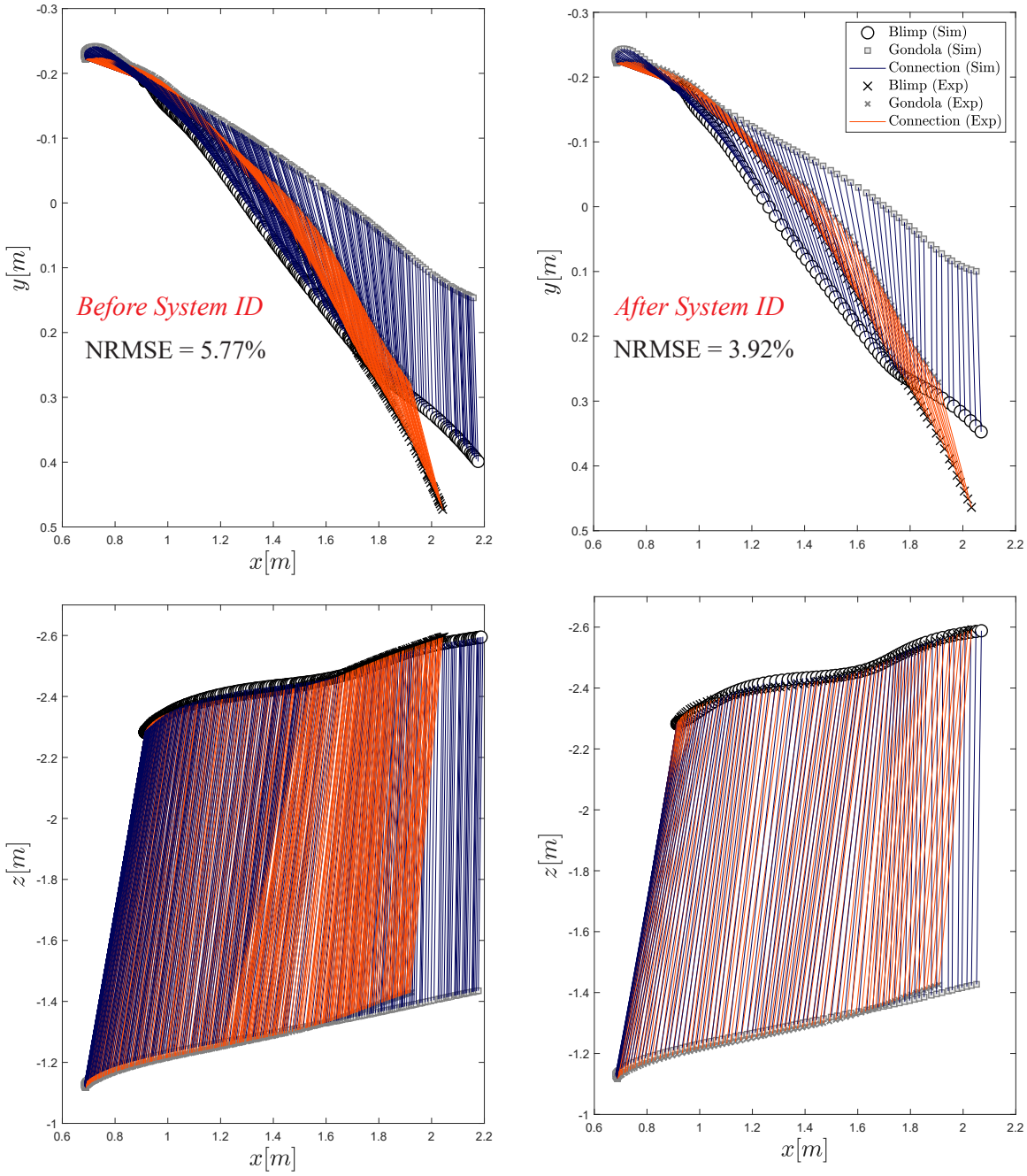


Figure 3.9: Results obtained from Experiment 3. Right and left columns are respectively the trajectories with and without system ID.

The reproducibility of the proposed system identification method warrants discussion. Since the applied force was manually generated by hand, achieving consistent and repeat-

able results is challenging. A more statistically robust approach would involve applying thrust forces directly. However, this option was unavailable for the investigated airship due to the absence of force sensors on the thrusters. In addition, the absence of a statistical analysis in the evaluation of the system identification results introduces some limitations regarding the robustness and reliability of the data. Without such analysis, it becomes challenging to quantify the level of uncertainty, variability, or potential biases present in the identified mass matrix. To ensure a more reliable and robust system identification process, future studies should consider integrating statistical methods or repeating the same experiment multiple times.

3.4 Contributions

A new approach has been developed to identify the parameters in the mass matrix of a multibody system through optimization, combined with a methodology for conducting open-loop experimental tests for buoyant systems. The trajectories obtained from these tests are utilized in the identification process. The system identification problem is formulated as a direct dynamics problem and solved using semi-definite programming with equality and inequality constraints. The method allows to identify unknown parameters in the mass matrix of a multibody system. Equality constraints are introduced to minimize the difference between accelerations that are predicted by the model and those acquired experimentally. By utilizing the structure of the EOM and iteratively solving pseudo-inverse terms (namely $\mathbf{K}$), the moment of inertia, added inertia, and added mass of the multibody autonomous LTA vehicle can be identified using SDP.

Among the main challenges encountered when performing system identification of multibody system is the dynamic coupling of the system. The latter is often circumvented by decoupling of the system, which might not be feasible in a practical scenario, or constraining the motion of the system so that the dynamic model is simplified. Nevertheless, the proposed method can conveniently accommodate the dynamical constraints of the multibody system without the need to simplify or decouple the system dynamics. This allows for system identification to be carried out in a wider range of experimental scenarios.

The proposed method works by minimizing the error between the acceleration predicted by the model and the acceleration obtained experimentally. Although the experiments were performed in open-loop with no control actions from the thrusters, the proposed method for system identification remains applicable even in the scenario where thrusters are employed, provided that the force vector generated by the thrusters can be measured in real time[4]. Consequently, this system identification approach can be applied to a range of vehicles, including airships and underwater vehicles, particularly in scenarios where parameters such as added mass and inertia within the mass matrix are unknown.

[4]The investigated prototype lacked onboard sensors to measure thrust, which limited the ability to apply force directly through the thrusters.

Chapter 4

Aerodynamics

Traditionally, aerodynamic models have been derived from theoretical work [27, 74–76] or combination of theory and experiment (semi-empirical approach) [77–80]. These models can significantly overestimate or underestimate the aerodynamics of uninhabited airships with unconventional geometries. Therefore, a numerical simulation was carried out to obtain the aerodynamic coefficients, which were subsequently compared to models found in the literature. It is important to note that there is a slight abuse of notation in this chapter, where the symbols α and β are used to represent the angle of attack and the side slip angle, respectively.

4.1 Literature Review

The majority of theoretical work was derived from potential flow theory. The latter implies that the flow is inviscid, incompressible and irrotational. In addition, the flow field is treated as steady (transients are neglected) and attached over the entire model (*i.e.,* no separation). The normal pressure distribution around the hull, from which aerodynamic forces and moments can be derived, was obtained by Munk based on slender body theory [75].

$$\frac{dF_N}{d\xi} = (k_2 - k_1) q_0 \frac{dS}{d\xi} \sin(2\alpha) \tag{4.1}$$

where ξ is the axial direction along the hull, S is the hull cross-section area, k_1 and k_2 are respectively the axial and lateral coefficients of added mass, α angle of attack, and q_0 is the dynamic pressure. After Munk, many efforts were invested in developing the model [36, 77–79, 81–83]. Owing to the inviscid flow assumption which potential flow theory adopts, Allen and Perkins [77] modelled the normal force F_N using a semi-empirical approach. The viscosity effects were added to the model as an additional term to Munk's equation. The additional term in (4.2) was derived from the steady drag force acting on an infinitely long cylinder, of identical diameter to the hull, in a cross-flow. The relation was inspired by the theoretical work of Jones [84] in which the viscous flow across an infinite cylinder was found to be independent of flow parallel to the cylinder.

$$\frac{dF_N}{d\xi} = (k_2 - k_1)\ q_0\ \frac{dS}{d\xi} \sin(2\alpha) \cos(\frac{\alpha}{2}) + 2\ q_0\ R\ \eta\ C_{DC} \sin^2(\alpha) \tag{4.2}$$

From the normal force distribution (4.2), the same authors also deduced the coefficients of lift and forebody drag (4.3-4.4) [85]. The lift and drag forces are calculated by projecting each term in the normal force equation (4.2) onto the direction that is, respectively, perpendicular and parallel to the stream-wise direction.

$$\begin{aligned} C_L = \frac{1}{A}\{&(k_2 - k_1) \sin(2\alpha) \cos(\frac{\alpha}{2}) \int_0^l \frac{dS}{d\xi} d\xi \\ &+ \sin^2(\alpha) \cos(\alpha)\ \eta \int_0^l 2R\ C_{DC}\ d\xi\ \} \end{aligned} \tag{4.3}$$

$$\begin{aligned} C_D = \frac{1}{A}\{&(k_2 - k_1) \sin(2\alpha) \sin(\frac{\alpha}{2}) \int_0^l \frac{dS}{d\xi} d\xi \\ &+ C_{D_{(\alpha=0)}} \cos^3(\alpha) + \sin^3(\alpha)\ \eta \int_0^l 2R\ C_{DC}\ d\xi\ \} \end{aligned} \tag{4.4}$$

where A is the reference area (which is used to normalize lift and drag coefficients), C_{DC} is the cross-flow drag coefficient of the infinite circular cylinder, $C_{D_{(\alpha=0)}}$ is the drag coefficient at zero angle of attack, η is a correction factor to account for the finite length of the body and it depends on the fineness ratio, R is the radius of the local cross-section, and l is the length of the hull. It is worth noting that the first term in (4.2) was multiplied by $\cos(\frac{\alpha}{2})$.

The latter can be shown from the work of Ward [86] where the cross flow forces take place at an angle between two axes. The axis that is normal to the stream-wise direction and the axis normal to the body axial direction (thus $\frac{\alpha}{2}$).

Further modifications were carried out by Hopkins [78] who divided the flow around the airship into two sections. A normal force section ($0 \leq \xi \leq \xi v$) and a drag force section ($\xi v \leq \xi \leq l$). The location along the axial direction where the potential flow assumption becomes invalid is denoted by ξv and l is the length of the hull, as defined earlier. The aerodynamic normal force at the front section was calculated using Munk's model (4.1). Whereas Hopkins determined that the force acting on the remaining section of the body is attributed to cross-flow drag. The normal force distribution at low angles of attack could be expressed as:

$$\frac{dF_N}{d\xi} = \begin{cases} 2(k_2 - k_1)\ q_0\ \alpha\ \dfrac{dS}{d\xi} & \text{for } 0 \leq \xi \leq \xi_v \\ 2q_0\ R\ \eta\ C_{DC}\ \alpha^2 & \text{for } \xi v < \xi < l \end{cases} \tag{4.5}$$

where the value of ξ_v is computed empirically from wind tunnel experiments

$$\xi_v = 0.378l + 0.5271\xi_1 \tag{4.6}$$

and ξ_1 indicates the position where the value of $\dfrac{dS}{d\xi}$ is minimum. It is worth noting that the units for ξ_v, ξ_1, and l should be expressed in feet.

Owing to this division, the approach of Hopkins[78] yields a discontinuity in the normal force distribution at distance ξ_v in the axial direction. In addition to dividing the flow field into two sections and restricting the angles of attack to low values, the value of C_{DC} was set as a constant of 1.2 in the Hopkins method [78]. On the other hand, C_{DC} was bounded in the range of 0.3 and 1.2 in Allen's approach [77]. As a result, the normal force distributions generated by these methods were different. Both models were validated through a series of wind-tunnel experiments. The results obtained from (4.5) are in better agreement with the experiments. However, due to lack of experimental data for angles above $\alpha = 20°$, using Hopkins model requires finding ξ_v empirically (either experimentally or numerically) for for high angles of attack. On the other hand, (4.2) can be easily extended up to $\alpha = 180°$.

DeLaurier and Schenck [81] included the shear layer vortex shedding effect in the stability analysis. Jones and Delaurier [79] extended the work of Allen and Perkins by accounting for the loads on the fin of the aircraft, as derived by Wardlaw [87]. The forces on the fin were modelled as two points acting on the centre of pressure and centre of area of the fin. From (4.2) normal force F_N and drag force F_D can be obtained and include two additional terms.

$$\begin{aligned} F_N =& q_0\{\ (k_2-k_1)\ \eta_k\ \sin(2\alpha)\cos(\frac{\alpha}{2})\int_0^{\ell_h}\frac{dS}{d\xi}d\xi \\ &+ C_{DC}\sin(\alpha)\sin|\alpha|\int_0^{\ell_h}2Rd\xi \\ &+ S_F[\ (C_{n\alpha})_F\ \eta_F\ \frac{\sin(2\alpha)}{2}+\ (C_{DC})_F\ \sin(\alpha)\sin|\alpha|\ ]\ \} \end{aligned} \tag{4.7}$$

$$\begin{aligned} F_D =& q_0\{\ -(k_2-k_1)\ \eta_k\ \sin(2\alpha)\sin(\frac{\alpha}{2})\int_0^{\ell_h}\frac{dS}{d\xi}d\xi \quad (1) \\ &+ \cos^2(\alpha)[\ (C_{D_{(\alpha=0)}})_h\ S_h\ +(C_{D_{(\alpha=0)}})_F\ S_F\ ] \quad (2) \\ &- S_F(C_t)_F\ \} \quad (3) \end{aligned} \tag{4.8}$$

where ℓ_h is the distance from nose to hull-fin intersection point, S_h and S_F are the reference area of the hull and fin, respectively, C_{DC} and $C_{D_{(\alpha=0)}}$ are respectively cross-flow drag coefficient and drag coefficient at zero angle of attack, $(C_{n\alpha})$ is the derivative of normal force coefficient with respect to angle of attack. $(.)_F$ refers to the fin. Besides the two additional terms added to the normal force equation, hull and fin efficiency factors η_k and η_F, respectively, were introduced to account for interference between hull and fin. Moreover, rewriting the $\sin(\alpha)^2$ term as $\sin(\alpha)\sin|\alpha|$ helps dealing with negative angles of attack. Similarly, the drag force equation (4.8) presented in [79] was modified in comparison to (4.4). Besides the sign and the efficiency of the hull (η_k), the first term (1) in (4.8) is identical to (4.4). However, the contribution of the fin drag was added to the second term (2) in (4.8). A notable discrepancy is the multiplication of the second term by $\cos^2(\alpha)$ in (4.8) as opposed to $\cos^3(\alpha)$ in (4.4). The last term in (4.4) was replaced by (3) $(C_t)_F$ in (4.8) which represents the leading-edge suction coefficient of the fins and it acts as a correction factor to the Wardlaw model since the fins' leading edge is not sharp. Readers are referred to [79] for calculating $(C_t)_F$.

Following the Jones and Delaurier model, Mueller and Paluszek [88] derived the aerody-

namic forces and moments of an airship at an angle of attack and side-slip angles. The effects of the control surfaces, namely rudders and elevators, were included in the model. The aerodynamic forces and moment in the body axis frame are described in section C.1 of Appendix C.

The differences are minute between [79] (4.7)-(4.8) and [88] (C.1)-(C.3). To begin, the term (3) in (4.8) was removed from (C.1). In addition, the effect of the gondola was added in (C.7) to term (2) and the whole term is multiplied by $\cos^2(\beta)$ to account for the side-slip. Also, the effects of the gondola cross-sectional drag were included in the aerodynamic model of the lateral forces (C.11) while neglected in the vertical axis (C.15). The forces along the lateral axis are modelled identically to the forces in the z direction (normal forces) in (C.2) - (C.3), but the lateral forces now depend on the side-slip and rudder angles instead of angle of attack and elevator angles.

Recoskie [89] modified the aerodynamic moments of the [88] model (C.4)-(C.6) in order to account for damping forces that take place when the aircraft is undergoing an angular motion (such as yaw) while hovering, *i.e.*, where the translational velocity is zero. Since the dynamic pressure is zero when $U_\infty = 0$, all aerodynamic forces will cancel out. Therefore, the additional terms in the aerodynamic moments are functions of angular velocity instead of translational velocity (C.27)-(C.29). Although the aerodynamic forces were adopted from [88], Recoskie neglected the effect of the control surfaces (C.24)-(C.26). Equations derived by Recoskie are supplied in Section C.2 of Appendix C.

A summary table of all discussed aerodynamic models is presented, in a progressive order, in Table 4.1. Blue coloured text prepended by a plus sign refers to the additional terms introduced in the model with respect to the preceding one. In contrast, omitted terms are portrayed by a negative sign followed by red coloured text. Except Munk, all discussed aerodynamic models have unknown aerodynamic coefficients. Methods used to calculate the aerodynamic coefficients are discussed next.

Table 4.1: Historical development of airship aerodynamic model. Blue "+" and red "−" colours represent the added and omitted terms, respectively, of a given model relative to its preceding model.

	Normal Force	Drag Force
Munk[75] (4.1)	$F_N = q_0\{(k_2 - k_1)\sin(2\alpha)I_1\ \}$	(-)
Allen & Perkins[85] (4.3)-(4.4)	Multiplied (4.1) by $\cos\left(\frac{\alpha}{2}\right)$ + Viscous effects by accounting for cross flow drag. + Finite hull length effects.	Multiplied (4.1) by $\sin\left(\frac{\alpha}{2}\right)$ + Viscous effects by accounting for cross and parallel flow drag. + Finite hull length effects.
Hopkins[78] (4.5)	Divided the flow into two regions; $0 \leq x < x_v$ is dominated by F_N $x_v \leq x \leq l$ dominated by C_{DC}	
Jones & Delaurier[79] (4.7)-(4.8)	+ Forces of fins (two points force) + Correction term for fin (similar to that introduced by Allen for finite hull effects).	Multiplied the parallel flow drag term by $\cos^2$ instead of $\cos^3$ + Fin tip effects. - Cross flow drag effects
Mueller & Paluszek[88] (C.1)-(C.3)	+ Side-slip angle effects + Control surfaces effects (rudders and elevators)	+ Side-slip angle effects + Control surfaces effects (rudders and elevators) - Fin tip effects
Recoskie[89] (C.24)-(C.29)	+ Aerodynamic moments due to angular velocity (C.27)-(C.29) - Control surface effects (rudders and elevators)	

4.1.1 Aerodynamic Coefficients: Estimation and Calculation

As demonstrated in the previous section, aerodynamic models rely on aerodynamic coefficients, namely C_D and C_L. These coefficients depend on the geometry of the body, angle of attack, side-slip angle, the roughness of the body surface, and Reynolds number. It is a usual practice to calculate the aerodynamic coefficients experimentally using wind/wa-

ter tunnels [36, 90–92], numerically using computational fluid dynamics (CFD) techniques [93–100], or less often by semi-empirical methods that combine slender body/finite wing theories along with experimental data fitting [101].

Hoerner [101] derived an expression for drag coefficient (C_D) of a bare hull based on slender body theory and experimental data of various airships. Drag coefficients were fitted based on the airships' thickness ratio D/l, and Reynolds number.

$$C_{DV} = \{\ 0.172(l/D)^{1/3} + 0.252(D/l)^{1.2} + 1.032(D/l)^{2.7}\ \} Re_l^{-1/6} \tag{4.9}$$

where C_{DV} is the drag coefficient of the bare hull based on the volume of the hull, D is the maximum diameter of the hull, l is the length of the hull, and Re_l is the Reynolds number based on hull length. The Reynold's number Re_ζ is given by (4.10)

$$Re_\zeta = \rho U_\infty \zeta / \mu \tag{4.10}$$

where U_∞ is the freestream velocity, ζ is the characteristic length, ρ is the fluid density, and μ is the dynamic viscosity. Hoerner suggested that (4.9) is practical for $Re_l > 5 \times 10^6$ [101]. It is important to note the characteristic length, ζ, can be chosen differently when calculating the Reynolds number, depending on the choice of the author. For example, it is common in the literature of airships to use the hull length ($\zeta = l$), the diameter of the hull cross-section ($\zeta = D$), or the cubic root of hull volume ($\zeta = V_h^{1/3}$) as the characteristic length when calculating Reynolds number. Hence, when comparing results of different studies, it is essential to use a consistent base for Reynolds number. Similarly, drag coefficient C_D can be either normalized by surface area S_h (C_{DS}) or $V_h^{2/3}$ (C_{DV}) of the airship hull [1].

Since the angle of attack is not considered in the given relationship, it can be anticipated that (4.9) performs poorly as the angle of attack is altered. Khoury [103] used (4.9) to compare the drag coefficient of Skyship 500 with an experimental study carried out by Bailey [104]. C_{DV} obtained from flight test was twice that calculated using (4.9). However, Khoury attributed the discrepancy to the additional components on the airship, namely

[1] Young [102] proposed a relationship to convert drag coefficient from C_{DS} to C_{DV}. The formula was tested on two different airships and gave accurate results within 0.8% error margin [103].

the tail and gondola. Moreover, Hoerner proposed a relationship to calculate lift coefficient (C_L) based on C_D and angle of attack.

$$\frac{C_L}{C_D} = sin^2(\alpha)cos(\alpha) \tag{4.11}$$

Equation (4.11) was evaluated for 6:1 prolate body[2] and compared with high fidelity numerical simulation [105]. The results were acceptable for very low Reynolds number ($Re_D = 50$) but deviated significantly as Reynolds number is increased.

Experimental studies of airships aerodynamics are hampered by several challenges. In wind tunnel experiments, a scaled down model is often used in lieu of the full-size airship which reduces the Reynolds number. As a result, the flow regime in which the experiment is carried out could be different from that encountered in real flight conditions. Moreover, the model is often mounted on a tunnel structure, such as a sting, which alters the flow field. Furthermore, measurement techniques are limited in terms of tracking the spatial and temporal development of the vortices in the leeward side of the airship.

On the other hand, CFD provides an alternative to wind tunnel testing where the aerodynamic coefficients as well as the unsteady flow characteristics are calculated through numerical simulations. Early CFD studies used panel methods which are primarily based on inviscid potential flow theory in conjunction with a boundary-layer correction procedure to account for viscous effects [106]. Owing to its computational efficiency and simplicity, panel methods are useful tools for simulating attached flows, mostly at very low angles of attack. Nevertheless, they oversimplify the complex turbulent flow over the airship.
Lutz et al [94] used panel methods after adding a wake model and a separation criterion based on experiments. The method was only able to exhibit few flow features of turbulence and it was very sensitive to the assigned separation location. Jefferson *et al* [107] utilized XFLR5 software which is based on potential flow theory and lifting line theory [108] to estimate the lift, drag, and side force coefficients (C_L, C_D, and C_Y, respectively) for YEZ-2A airship. The results were compared with the wind tunnel tests in Gomes [36]. The agreement of the aerodynamic coefficients with the experiment is acceptable for low angles

[2] 6:1 prolate body is a bluff body with fineness ratio of $\frac{l}{D} = 6$.

of attack. As the angle of attack increases, C_L values remain in good agreement with the experiment but the error increases remarkably for C_D and C_Y.

The alternatives to panel methods require solving Navier Stokes (N-S) equations. Fully solving N-S equations[3] is known as Direct Numerical Simulation (DNS). Since the investigated airship operates at a Reynolds number significantly greater than $10,000$, DNS is unattainable due to stringent grid requirements. To relax the grid requirements, turbulence modelling is introduced. In Large Eddy Simulation (LES), the scales of motion are categorized into two groups; large scale and small scale. The large scale is resolved by the computational grid. The unresolved small-scale motion, which is smaller than grid spacing, is modeled via sub-grid-scale models. Although LES provides a feasible alternative to DNS, it still requires using High Performance Computers (HPC) for computations. Thus, Reynolds Averaged Navier Stokes (RANS) models are more common for industrial use.

Voloshini *et al* [96] provided the historical development of common Reynolds Averaged Navier Stokes (RANS) turbulent models. Based on the reviewed turbulent models, the authors selected four models, all of which are based on eddy-viscosity assumption[4]; Spalart-Allmaras (SA), standard and Menter SST $k-\omega$, and $k-\epsilon$. RANS turbulence models were implemented in a CFD study to calculate the aerodynamic coefficients for a classic scaled-down Zepplin airship operating at $Re_l = 7.91 \times 10^5$ on a mesh size of 4.5 million cells. Simulation was carried out over three angles of attack, $\alpha = -0.4°, 11.62°$, and $35.62°$. The results produced by the simulation were compared with experimental data. Errors in C_D in comparison with experimental studies will be referred to as ΔC_D. While predicted C_L values were in good agreement with experiment, C_D values diverged significantly at higher angles of attack. At $\alpha = -0.4°$, the error in drag coefficient produced by SA and $k-\epsilon$ was the lowest, with a value of $\Delta C_D = 6.9\%$ in both models. In comparison, standard $k-\omega$ and Menter SST $k-\omega$ achieved $\Delta C_D = 10\%$ and 11.7%, respectively. Nevertheless, all models performed poorly when the angle of attack is increased where the minimum error at $\alpha = 35.62°$ was $\Delta C_D = 52.6\%$ using the $k-\epsilon$ model. While the RANS SA model was computationally faster than the rest, the paper concludes that all investigated RANS

[3] Up to Kolmogorov length scale[109]
[4] Also known as Boussinesq Hypothesis[110]

models are only valid for attached or mildly separated flows.

Kanoria *et al* [99] carried out a computational study of flow over the ZHIYUAN-1 airship (hull, fins, and gondola) at $Re_l = 14.955 \times 10^6$. RANS-SA and LES-WALE turbulence models were implemented using a computational domain of 2.8 million cells and 8.4 million cells, respectively. Comparing the obtained C_L curve with the experimental study showed that both methods agree with the wind tunnel values for $\alpha < 14°$ beyond which the values deviate. On the other hand, LES demonstrated agreement with the experimental study for C_D at $0° \leq \alpha \leq 10°$ while C_D values predicted by RANS matched over $12° \leq \alpha \leq 20°$. Both models failed to predict the values of C_D at $\alpha > 20°$.

Carrión *et al* [111] investigated flow over Airlander of Hybrid Air Vehicles using unsteady RANS (URANS) with explicit-algebraic $k-\omega$, Menter SST $k-\omega$, and $\gamma - Re_\theta$ turbulence models. The study investigated three body configurations; 6:1 prolate body at $Re_l = 4.2 \times 10^6$, bare hull consisting of 3 merged prolate bodies (see Fig. 1.1), and full airship (hull, fins, and strakes) at $Re_l = 3 \times 10^6$. The computational domain for 6:1 spheroid consists of 40 million cells while the full airship is only 31 million cells. Since experimental results are only available for 6:1 prolate body, the turbulence models and mesh convergence were validated using 6:1 prolate geometry at $\alpha = 20°$. The distributions of pressure coefficient C_p and skin friction coefficient C_f in the azimuthal direction were acquired and contrasted with an experimental and numerical studies from the literature. For C_p distribution, RANS-explicit-algebraic $k-\omega$ and Menter SST $k-\omega$ resulted in good agreement with the experimental data whereas $\gamma - Re_\theta$ performed poorly. The deficiency of $\gamma - Re_\theta$ is attributed to the delayed prediction of flow transition. It is worth noting that the flow was tripped near the nose in the wind tunnel testing. Hence, the boundary layer was mostly turbulent over the entire prolate body. The latter could justify the inaccuracy of the results obtained by the $\gamma - Re_\theta$ turbulence model as it is a free transition model[5]. On the other hand, explicit-algebraic $k-\omega$ model accomplished a more accurate C_f distribution than Menter SST $k-\omega$ especially in the vicinity of the secondary and tertiary vortices (illustration of vortices is shown in Fig. C.1). Unfortunately, the study does not

[5]Free transition models predict the point at which the flow transitions from laminar to turbulent. Other models have predetermined transition locations which are set by the user

provide a comparison of aerodynamic coefficients with experimental data.

While submarines operate under different Reynolds number, the review will be focused on simulations of submarines that are operating at a similar Reynolds number to airships. Several turbulence models have been employed in numerical investigations of flow over prolate bodies and submarines. Reviewed turbulence models include RANS-free-transition [98], RANS-$k-\omega$ [112], RANS-$k-\epsilon$ [95, 112], RANS-SA [113–115], URANS [116, 117], LES-WALE [115], LES-Smagorinsky [95, 117, 118], and Detached Eddy Simulation (DES) [114, 115, 119]. In addition, recent CFD studies for flow over 6:1 prolate bodies at a relatively low Reynolds number used DNS [105, 120–125]. Given that these same modelling techniques are applicable to simulating flow over airships, it is valuable to examine how accurately they predict the aerodynamic coefficients. This review can provide insights into the effectiveness of these models for airship simulations at comparable Reynolds numbers. Suman *et al* [98] used RANS to assess the sensitivity of the flow over ZHIYUAN-1 bare hull using two transition models. Computed aerodynamic coefficients were then compared with experimental data. The study found two trends for C_D with respect to transition location. As the transition point is moved downstream, C_D decreases until a critical point is reached. Downstream of that critical location, the trend reverses where the values of C_D now increases substantially. This can be explained by flow separation. Since separation is more likely to occur earlier in the case of a laminar boundary layer, delaying the transition results in premature laminar separation which causes an increase in pressure drag. Moreover, the influence of the angle of attack on aerodynamic coefficients was discussed. The transition point was assumed to be fixed irrespective of the angle of attack. This assumption will introduce errors to the results since it has been established in the literature [126] that the transition point is dependent on angle of attack. As expected, values of C_D and C_L deviated from experimental values at $\alpha > 5°$.

Andersson *et al* [123] investigated flow over 6:1 prolate body using DNS. The study covers a broad range of Reynolds number, $Re_D = 10 - 3000$ (equivalent to $Re_l = 60 - 18,000$) at $\alpha = -45°$. Although airships operate at a much higher Reynolds number (in the order of $Re_l = 10^6$), the flow phenomena observed over the investigated range of

Reynolds numbers will certainly help in understanding the complexity of the flow field. Various computational meshes were implemented, varying in size from 13.06 million cells (for $Re_D = 50$) to a maximum of 748.7 million cells (for $Re_D = 3000$). The results obtained delineates the flow development as the Reynolds number is increased. At low Reynolds number, $Re_D < 800$, the flow field is steady, laminar, and the wake-vortex is symmetric about the longitudinal plane ($X - Z$ plane). The incipience of unsteadiness and wake-vortex asymmetry begin to show up mildly at $800 < Re_D \leq 1200$. As Reynolds number is increased to $Re_D = 3000$, the flow is distinctively unsteady and the time-averaged wake-vortex asymmetry [6] becomes prominent. As a result of asymmetry, a side force, in the lateral direction, is generated with a magnitude of 75% of drag force.

Another interesting phenomenon is the bi-stability of the asymmetry where the wake-vortex is arbitrarily deflected to one of either body sides. The latter is known as pitchfork bifurcation and it has also been observed in flows over sharp nose slender bodies at high angles of attack [127, 128]. However, the aspect ratio of such bodies is often $l/D > 10$. In addition, aerodynamic coefficients show a trend of decreasing C_D at $50 \leq Re_D \leq 1200$ while C_L was only slightly affected. Nevertheless, the magnitude of lift and drag coefficients increase as the Reynolds number is increased to $Re_D = 3000$. Owing to lack of experimental data at the investigated angle of attack and Reynolds number, the simulation data could not be further validated.

In an endeavour to cover a wider range of Reynolds number at $\alpha = -45°$, Strandenes *et al* simulated flow over 6:1 prolate body at $Re_D = 4000$ [124] and $Re_D = 8000$ [125] using DNS. The studies give a further insight about the flow instabilities such as Kelvin-Helmholtz, as well as delineating the evolution of vortices on the leeward side. Also, the bi-stable nature of wake-vortex asymmetry was demonstrated where the calculated side force at $Re_D = 4000$ was compared with $Re_D = 3000$ (from [123]). The side force manifested a change in sign, indicating a switch in the direction of the asymmetry.

Besides the turbulence modelling, the validity of numerical simulations are dictated by several factors including the physical model, grid resolution [98], solution schemes [129], convergence criteria [130], and the coupling between these factors.

[6] Further details about wake-vortex asymmetry formation around slender bodies can be found in [127].

Based on the reviewed studies for estimating aerodynamic coefficients of airships, prolate bodies, and submarines, the following was observed:

1. Most of the CFD studies used RANS for high Reynolds number flow. RANS models were only able to produce acceptable aerodynamic coefficients at low angles of attack ($\alpha < 10°$). When the flow field exhibits high levels of unsteadiness and separation, RANS models fail to predict accurate lift and drag coefficients. In addition, RANS models are sensitive to the prescribed transition location, as demonstrated by [98].

2. LES studies were carried out on coarse computational grids. It is crucial to have a sufficiently dense grid in order to resolve the turbulent flow features including the wake-vortex asymmetry [127]. Hence, LES studies using a coarse grid also failed to accurately predict C_L and C_D.

3. Due to stringent grid requirements, studies that used DNS were restricted to low Reynolds number ($10 \leq Re_D \leq 10,000$).

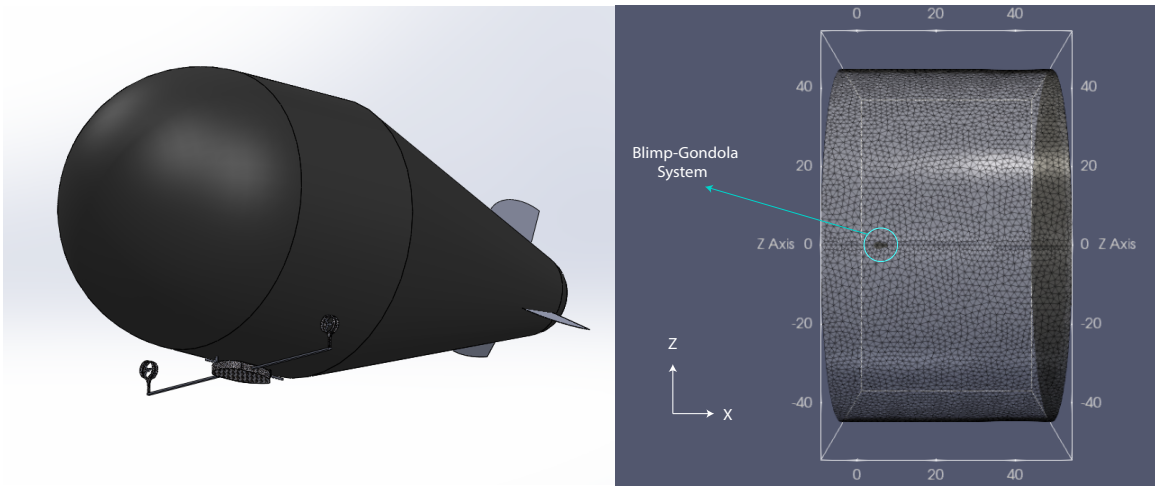
4. The influence of side force (C_Y) at high angles of attack ($\alpha \geq 30°$), discussed in [123, 124] for a 6:1 prolate body, could not be predicted in any of the airships simulations.

4.2 Methodology

Since the investigated LTA blimp is designed to operate at low angles of attack, the RANS-SA turbulence model was used to estimate the aerodynamic coefficients of the airship system. In terms of computational power, RANS is the least demanding among other CFD turbulence modelling methods discussed earlier. Nevertheless, utilization of HPC is still necessary since the simulations were performed on the full-scale prototype. Hence, the numerical investigation was carried out on Compute Canada HPCs using Star-CCM+ software. Given the computational demands of CFD simulations and the airship's design for operation at low angles of attack, the simulations will be conducted within a range of $-5°$ to $5°$ for both the angle of attack and side-slip angles.

4.2.1 Physical Model and Computational Model

The airship model was developed in SolidWorks, then exported to Pointwise for meshing, and finally transferred to STAR-CCM+ for simulation. The CAD model of the airship and the dimensions of the computational domain are delineated in Fig. 4.1a and Fig. 4.1b, respectively. The distance from the inlet to the nose of the hull is $5D$, tail to the outlet is $27.5D$, and the radial radius is $30D$. The size of the computational dimensions was selected based on two factors. The first factor is related to the size of the body where previous studies demonstrated that the flow development is influenced by the size of the computational domain [127]. The second factor is related to the boundary conditions used (depicted in Fig. 4.2). For instance, The outlet flow condition uses extrapolation which may affect the flow field in the vicinity of the outlet boundary. Hence, the outlet boundary should be moved far away from the body to minimize its effect on the aerodynamic forces.



(a) CAD model of the airship system. (b) Computational domain.

Figure 4.1: Airship model used to carry out the simulations.

Recall that the cross-section diameter and length of the hull are $D = 1.82$ m and $l = 4.1$ m, respectively. The airship will be operating at a constant uniform speed of $U_\infty = 8.33$ m/s $= 30$ km/h . Reynold's number is calculated based on hull length (4.12) and hull diameter (4.13)

$$Re_l = \frac{\rho\, U_\infty\, l}{\mu} = 2.338 \times 10^6 \tag{4.12}$$

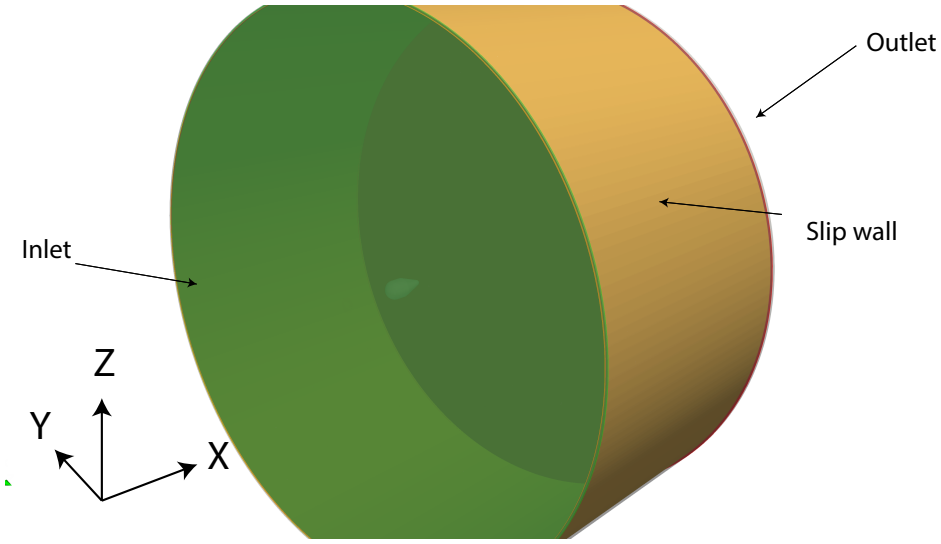


Figure 4.2: Computational domain, demonstrating the boundary conditions.

$$Re_D = \frac{\rho\, U_\infty\, D}{\mu} = 1.038 \times 10^6 \tag{4.13}$$

where $\rho = 1.225$ kg m^{-3} and $\mu = 1.789 \times 10^{-5}$ kg m^{-1} s^{-1} are the air density and dynamic viscosity, respectively. Reynolds number will specify whether the boundary layer will be laminar or turbulent. According to [131], the flow over 6:1 prolate spheroid was found to undergo a turbulent separation in the range of $2 \times 10^6 \leq Re_l \leq 2.8 \times 10^6$. Since the investigated Reynold number is within the critical Reynold number, it is possible that the boundary layer is turbulent. A turbulent boundary layer requires stringent surface meshing along the body, namely the grid spacing in the streamwise and azimuthal directions. To avoid expensive surface meshing, a turbulent boundary layer modelling, known by $y+$ wall treatment, is implemented. $y+$ wall treatment models turbulent boundary layer quantities such as turbulent production, dissipation, as well as wall shear stress [132].

A grid sensitivity study is carried out to assess the influence of the mesh on the aerodynamic coefficients. Four unstructured grids were generated: 20×10^6 cells, 60×10^6 cells, 120×10^6 cells, and 148×10^6 cells. The variation in aerodynamic coefficients was found to be negligible between the last two grids, hence, no further grid refinements were performed.

The investigated angles of attack (α) and side slip angles (β) are $-5°, -2°, 0, 2°$, and $5°$. The analysis is repeated over all combinations of angles which resulted in a total of 25 simulations. For each simulation, a new grid is generated corresponding to the angle of

attack and side slip angle.

4.2.2 Solution Method

Simulations are carried out using Siemens Star-CCM+. The solver uses SIMPLE scheme for pressure-velocity coupling. A second-order central differencing scheme was applied for spatial discretization while the RANS-SA [133] turbulence model was solved using first order upwind scheme. While the aerodynamic coefficients converged after $20,000$ iterations, the flow field was solved for an additional $10,000$ iterations to ensure complete convergence.

4.2.3 Results

Aerodynamics coefficients obtained from simulations are presented here. The aerodynamic coefficients are defined as follows

$$C_D = \frac{D}{0.5\ \rho\ V^2\ Vol^{2/3}} \tag{4.14}$$

$$C_S = \frac{S}{0.5\ \rho\ V^2\ Vol^{2/3}} \tag{4.15}$$

$$C_L = \frac{L}{0.5\ \rho\ V^2\ Vol^{2/3}} \tag{4.16}$$

$$C_{Mx} = \frac{M_x}{0.5\ \rho\ V^2\ Vol^{2/3}\ D} \tag{4.17}$$

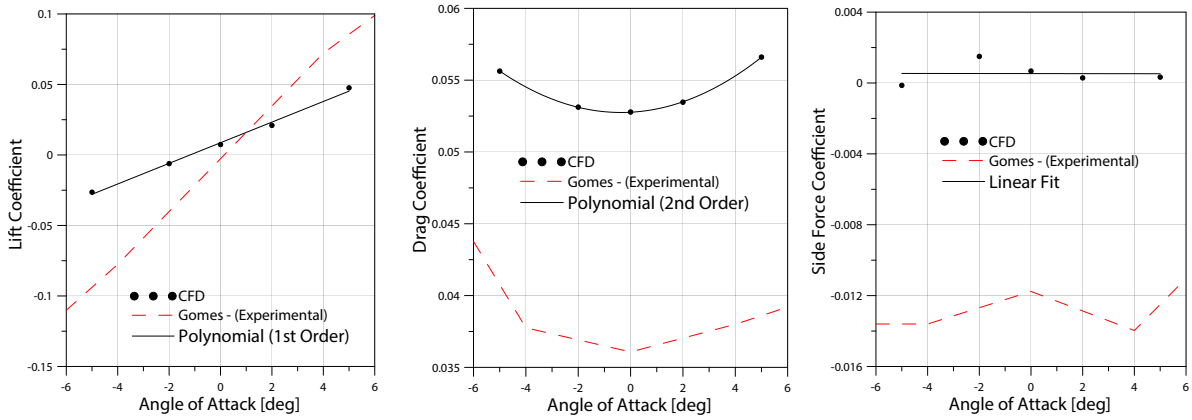
$$C_{My} = \frac{M_y}{0.5\ \rho\ V^2\ Vol^{2/3}\ D} \tag{4.18}$$

$$C_{Mz} = \frac{M_z}{0.5\ \rho\ V^2\ Vol^{2/3}\ D} \tag{4.19}$$

where C_D, C_S, and C_L are, respectively, drag, side force, and lift coefficients. Similarly, C_{Mx}, C_{My}, and C_{Mz} are the moment coefficients about x, y, and z axes in the aerodynamic frame. The aerodynamic frame is illustrated in Fig. 4.2. V is the resultant velocity, Vol is the volume of the airship's envelope, and D is the envelope diameter.

Figure 4.3 compares the acquired aerodynamic coefficients with experimental data obtained by means of wind tunnel tests on the YEZ-2A airship by Gomes [36]. Data shown at $-5° \leq \alpha \leq 5°$ and zero side slip angle. It is worth noting that the airship investigated

in this study is different from that used by Gomes [36]. In addition to differences in geometry, the wind tunnel data was performed on the bare airship envelope, *i.e.*, without gondola or fins. On the other hand, CFD data obtained here includes the effect of gondola, fins/tail, as well as propellers. Nevertheless, the trend shown is consistent with the wind tunnel tests. The lift coefficient changes linearly with angle of attack (Fig. 4.3a) while drag coefficient varies in parabolic fashion (Fig. 4.3b). Since the side slip angle is fixed at zero, the variation of side force coefficient is insignificant (Fig. 4.3c). Clearly, the drag for the investigated multibody system is higher due to additional components that were not considered in Gomes. The remaining angles are presented in Appendix C.3 along with the moment coefficients.



(a) Lift coefficient (C_L) (b) Drag coefficient (C_D) (c) Side force coefficient (C_S)

Figure 4.3: Comparison of aerodynamic coefficients at $-5° \leq \alpha \leq 5°$ and $\beta = 0$.

To include the acquired aerodynamic coefficients in the dynamic model derived earlier, the data must be fitted. The MATLAB "fit" function was used to perform the fitting. $poly11$ was used to fit lift and side force coefficients while $poly22$ was utilized to fit the drag coefficients. Fitted data is shown in Fig. 4.4-4.6 along with CFD data. The polynomial fit captures the CFD data closely within the investigated range of angles. The reader should note that the angles must be in degrees when using the fitted models.

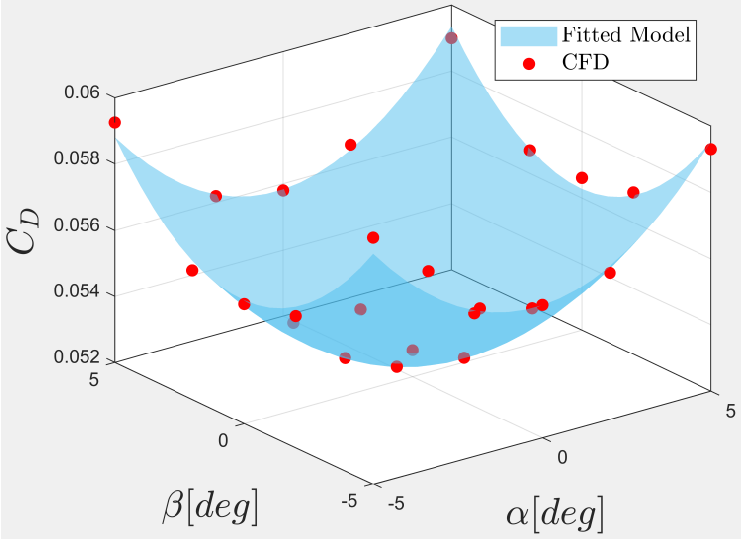


Figure 4.4: Second order polynomial representing the fitted drag coefficient.

$C_D = 0.05274 + 5.583 \times 10^{-5}\ \alpha - 1.593 \times 10^{-5}\ \beta + 0.1301 \times 10^{-3}\ \alpha^2 - 1.599 \times 10^{-7}\ \alpha\ \beta + 0.1269 \times 10^{-3}\beta^2$

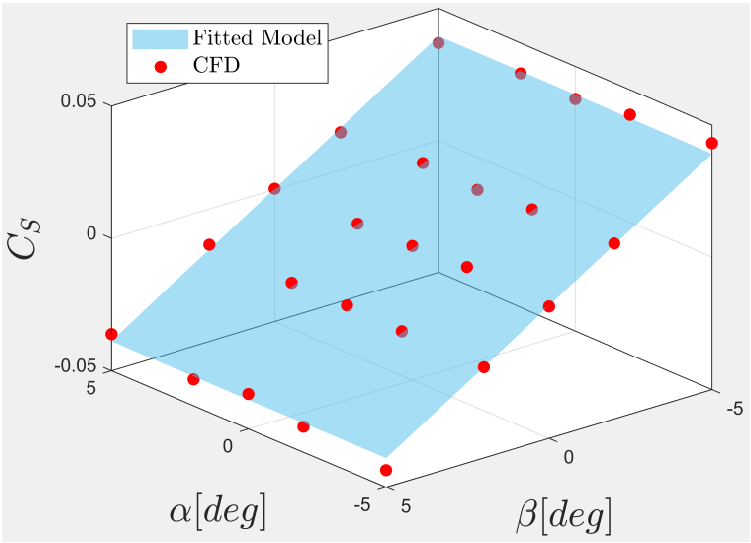


Figure 4.5: First order polynomial representing the fitted side force coefficient.

$C_S = 0.222 \times 10^{-3} + 1.396 \times 10^{-5}\ \alpha - 0.007822\ \beta.$

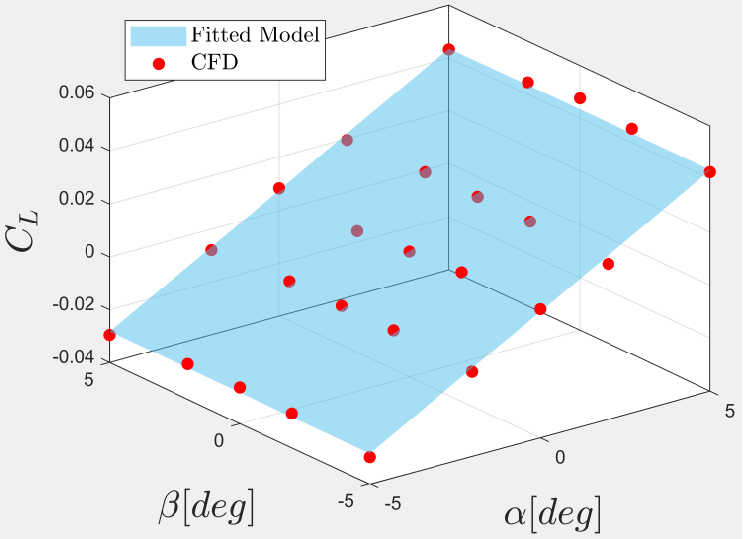


Figure 4.6: First order polynomial representing the fitted lift coefficient.
$C_L = 0.007573 + 0.007175\ \alpha + 9.594 \times 10^{-6}\ \beta$.

The moment coefficients about y and z axes were fitted with *poly*11 (Fig. 4.7-4.8) while the moment about x axis was found to be negligible.

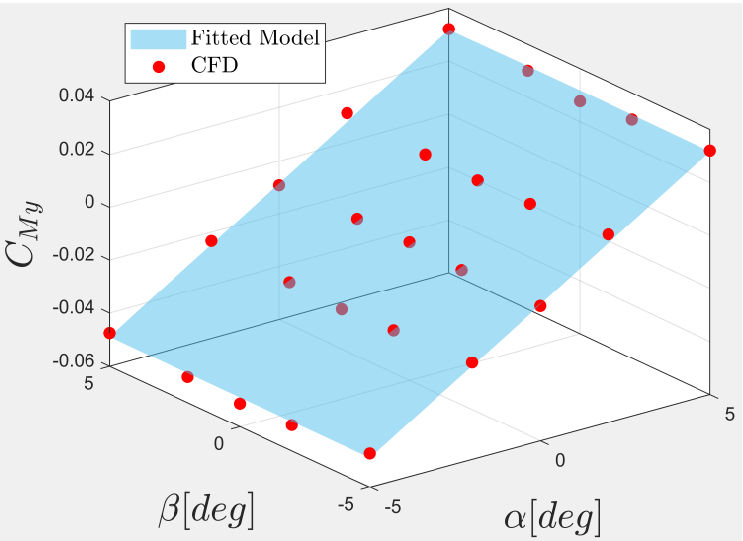


Figure 4.7: First order polynomial representing the fitted moment coefficient about y axis.
$C_{My} = -0.008533 + 0.008066\ \alpha + 6.369 \times 10^{-7}\ \beta$.

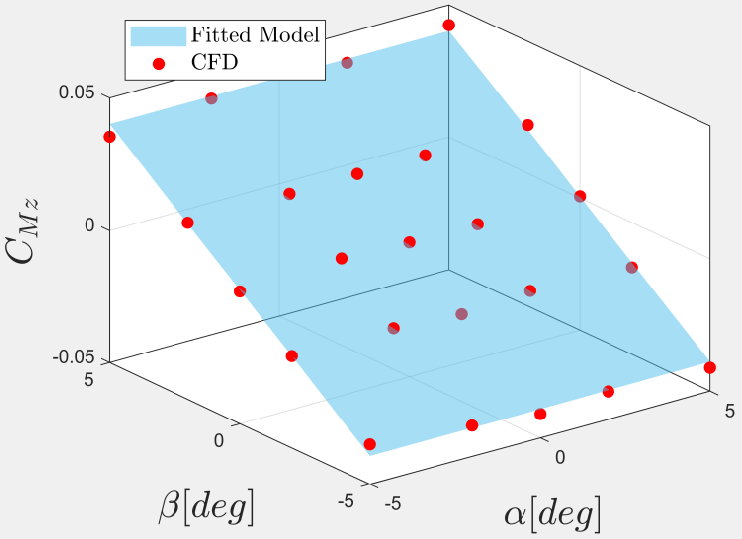


Figure 4.8: First order polynomial representing the fitted moment coefficient about z axis. $C_{Mz} = 0.0005087 + 2.155 \times 10^{-5}\ \alpha + 0.007935\ \beta$.

4.3 Contributions

Reynold's averaged Navier-Stokes equations along with the Spalart-Allmaras turbulence model were used to calculate the aerodynamic coefficients of the airship multibody system. The study was carried out at five different angles of attack and side-slip angles within the range of $-5° \leq \alpha, \beta \leq 5°$. This selection of angles is intended to partially cover the operational conditions of the multibody system. The aerodynamic coefficients obtained in this chapter provide valuable insights that can be utilized to enhance the design of the thrusters, ensuring the propulsion system operates efficiently under the airship's intended environmental conditions while fulfilling its intended mission.

Moreover, this chapter highlights the distinctions among the empirical, semi-empirical, and numerical aerodynamic models. It also offers guidance for those conducting computational studies, detailing which method to choose based on their research objectives.

Chapter 5

Slung Payload Control and Trajectory Tracking

With the developed dynamic model which incorporates the aerodynamic forces, the holonomic and nonholonomic constraint forces, and the added mass/inertia effects, a control strategy can be effectively designed to test the airship prototype with a slung payload. The computational capabilities of the onboard computer are limited, therefore, a review of control methods for aircraft with a slung-payload is presented to evaluate the most favourable strategy for the given hardware. That is, the control method should be computationally inexpensive, relatively easy to implement, and can be reliably tuned through simulations.

5.1 Literature Review

Various controlling architectures have been proposed to control aircraft with a slung load. These architectures can be broadly divided into actuated and under-actuated methods. An actuated system has at least as many actuators as degrees of freedom (DOFs) [1] while under-actuated systems have fewer actuators than degrees of freedom [2].

For the actuated methods, some models propose adding an active control input that directly

[1] Each degree of freedom can be independently controlled by an actuator.
[2] Not all DOFs can be directly controlled.

controls the slung payload such as an aerodynamic surface [134] or a mounted winch system designed to retract and manage the payload cables [135][60]. As an example, sliding mode and proportional-derivative (PD) controllers were used in [60] to stabilize the airship and the slug-load, respectively. While the paper only discusses simulation results during the loading/unloading phase without wind disturbance, the designed PD controller was able to track the prescribed slung load trajectory. Kang *et al.* [136] applied active control techniques to UAV helicopter with a slung load in order to transport the load to a moving goal with minimal error margins. The authors used three controllers' components: proportional-derivative controller that tracks the UAV position, feedback linearization of slung load dynamics to eliminate load oscillations, and an adaptive neural network (NN) to account for the system's uncertainty and external disturbances. The controller was able to correct for moderate disturbances but the performance degrades as the external disturbance is increased. The controller was only tested via simulations.

Nevertheless, the majority of recent control strategies found in literature utilize the coupling between the aircraft and payload in order to either alleviate or harness the swinging of the payload using an under-actuated control architecture. The design of under-actuated control strategies can be based on: payload trajectory following, aircraft trajectory following, or aircraft trajectory following with payload swing feedback. Since the prototype at hand is classified as an under-actuated system, each category is discussed in depth.

5.1.1 Payload Trajectory Following

The inputs to payload trajectory following controllers are the desired payload position, cable angle, or both. The controller then computes the corresponding aircraft attitude and position. In [137], [138], [139] the payload trajectory is designed to attenuate the slung load oscillations, resulting in a safer trajectory. Conversely, [55],[140] leverage the swinging dynamics of the slung load in order to achieve higher manoeuvrability and energy efficient trajectory. By employing a camera on the quadrotor, Tang *et al.* [55] were able to approximate the position of the slung load with respect to the aircraft using an Extended Kalman Filter (EKF). This allowed overcoming some of the challenges found in previous

works, such as having to constrain the system to planar workspace [140] and/or relying on external motion capturing tools [52].

A geometric controller was used in [141] [55] which is defined in a coordinate-free space and requires no linearization of the system dynamics. However, the controller has seven gain matrices in total, two for the quadrotor attitude controller (k_R, k_Ω), two for the payload attitude controller ($k_p, k_{\dot{p}}$), and three for the payload position controller (k_x, k_v, and k_i). The desired trajectory is fed to the payload position controller which computes the appropriate payload angle and angular velocity by using the data collected from the camera, Inertial Measurement Unit (IMU), and Vicon motion system. This information is then passed to the payload attitude controller that calculates the quadrotor orientation, angular velocity, and the required thrust vector. Consequently, thrust and moment inputs are determined by the quadrotor attitude controller. The robustness of the controller constants were first examined by varying the cable length (l) and trajectory periods (T) in a circular trajectory at a fixed altitude and payload path. That is, the quadrotor orientation, velocity, and path were allowed to change in order to maintain the altitude and path of the payload unaltered. It was reported that the payload control becomes harder as the trajectory period decreases. Although, the detection of the payload location is more difficult when the cable length increases, the control is relatively simpler. The average load angle and angular velocity errors were less than 2° and $0.2\frac{1}{s}$, respectively. Trajectory estimation error remained low throughout the investigated trajectories. However, the tracking error increased as the velocity increased. The paper did not specify the computational requirements for implementing this controller; nevertheless, a multibody system with more than six DOF would likely demand significant computational power.

Omar [142] developed a fuzzy logic controller (FLC) to suppress the oscillations of a helicopter slung payload. The rules of the controller are derived from the input shaping technique. Input shaping is a control technique that modifies the input signal to a system into a specific waveform to pre-compensate for its dynamics, thereby minimizing the oscillations from the system's natural frequencies. FLC was tuned using Particle swarm optimization (PSO), which is a global stochastic optimization method [143]. While FLC

was able to suppress the payload oscillations with initial payload disturbance, the paper does not present experimental results nor computational requirements for implementation. Also, the proposed FLC suffers from a lag-effect since it is based on input shaping logic.

5.1.2 Aircraft Trajectory Following

In the aircraft trajectory following approach, the aircraft trajectory is designed to optimize the slung load motion while no feedback control is applied to eliminate/dampen the swinging of the payload. Instead, input shaping could be applied in order to mitigate the swinging, as in [144], [145].

Klausen [57] *et al.* used a nonlinear backstepping technique to control a multirotor with a slung load. This control technique was implemented in two steps: first, two Lyapunov function candidates (LFC) are determined for the UAV and the slung load, separately. The LFCs are then superposed to compute the LFC of the whole system. After omitting singularity points from the system (by creating a perturbed model), stabilizing functions are introduced. The stabilizing functions are determined by setting the derivative of the LFC to zero. For further details about this controller, readers can refer to [146]. The controller was only tested in a Matlab simulation with and without an input-shape filter. As expected, including an input-shape filter was found to be necessary to minimize the swinging of the load.

Foehn *et al.* [56] adopted a nonlinear control technique from [58] which is similar to the nonlinear controller discussed in the previous section. From a high level description, the controller consists of position, attitude, and velocity controllers. The trajectory is first given to the position controller which computes the required thrust and orientation that achieve the desired trajectory. From current orientation, the attitude controller calculates the required velocity to rotate the vehicle to the desired attitude. Lastly, the acceleration controller determines the required torque to be applied to the vehicle. The controller and trajectory planer were tested experimentally based on three criteria: the first is the ability to navigate from point to point, the second is to avoid obstacles, and the third is to drop the payload. The results show faithful tracking to the nominal trajectory in all tested

scenarios. However, since the controller closes the loop on the aircraft, it is not robust to external disturbances on the slung-payload.

5.1.3 Aircraft Trajectory Following and Payload Swing Feedback

This approach is similar to the previous approach, but a feedback controller is applied to modify the desired motion trajectory such that it induces minimal load swinging motion.

Klausen *et al.* [147] used a backstepping technique in addition to a delayed feedback controller, adopted from [148], on a hexacopter to damp the payload oscillations. The payload is a cylindrical shaped metal piece with a mass of 250 g. Based on the current deflected payload angle, the delayed feedback controller sends a correction signal that is superposed with the input shaping signal. The addition of these two signals is then sent to the main controller block. The controller is tested experimentally with and without the swing damping (delayed feedback). The experiment was carried out using a trajectory that has the shape of ∞. While the tracking error was only slightly affected, the angular displacement of the payload was drastically attenuated when feedback is activated. It is worth noting that wind disturbance, with an average speed of 4 m/s, was present during the experimental testing. While no information was provided about the computational resources needed to implement this controller, the backstepping technique is often computationally demanding [149].

Fielding *et al.* [49] implemented input shaping within a cascaded control framework consisting of three interconnected controller blocks: an oscillation controller, a position controller, and an altitude controller. The oscillation controller block uses the same geometric controller described earlier in the work of [55],[141]. The position controller is adopted from [136] and it determines the thrust force which is a combination of feedforward and feedback elements for both aircraft and load swing tracking. The attitude controller is a nonlinear controller which retains a similar structure of the linear proportional-integral-derivative (PID) controller and it is based on the work of [150]. The additional nonlinear terms are incorporated into the controlled moment to counteract the angular acceleration induced by the desired angular velocity. The evaluation of the controller was carried out

by means of numerical simulations and showed favourable results in terms of mitigating load oscillations after u-turn manoeuvre and maintaining low tracking error.

Omar *et al.* [151] used FLC along with PID to suppress payload oscillations for a quadcopter with a slung payload. While the FLC rules were similar to that in [142], the tuning was carried out using a genetic algorithm. The PID control output, which controls the quadrotor position and attitude, is added to the fuzzy anti-swing control output. While the numerical simulations show improved payload attenuation in planar flight, the paper does not present experimental results.

5.1.4 Comparison

A summary of reviewed control strategies is presented in Table 5.1. Many of the presented under-actuated control strategies use geometric nonlinear controllers to track the aircraft and/or the payload trajectories which can be challenging to derive and implement. For quadrotors, geometric nonlinear controllers were found to be practical in achieving severe aerobatic manoeuvres including swinging of the payload. Nevertheless, such manoeuvres are not favourable in the case of airships as they could lead to adverse outcomes. Moreover, most of the experimentally tested controllers discussed in the literature review were tuned using empirical methods which can be laborious. Therefore, there is a need for a control method that is both easy to implement and computationally inexpensive for uninhabited aircraft with limited computational capabilities. To address this need, the author proposes a fuzzy logic control method that is relatively simple and efficient to implement on the airship's onboard computer. The fuzzy logic controller is discussed in the following Section 5.2.

Table 5.1: Control strategies for a slung load UAV.

	Tracking	Control Strategy	Advantages	Challenges/Shortages
Actuated Payload Control	UAV	Sliding mode control [60], PD [136]	Provides a direct control authority over the payload	Requires extra actuators/motors [136] Feedback linearization is computationally expensive
	Payload	PD [60], LQR [134], feedback linearization [136]		
Under-actuated Payload Control	Payload trajectory following	Geometric nonlinear [55, 152] FLC [142]	[55, 152] Robust,tested experimentally. [142] Simple to implement	[55, 152] Hard to derive, unknown computational req. [142] Not tested experimentally.
	Aircraft trajectory following	Geometric nonlinear controller [56]	Robust to UAV external disturbances, tested experimentally	Hard to derive, unknown computational req.
	Aircraft trajectory following + payload feedback	For multirotor UAV: Geometric nonlinear [49], backstepping [147], PID [151] For payload swing: Geometric controller [49], delayed feedback [148], FLC [151], delayed feedback + gain scheduling[147]	Robust to external disturbances	[147] Backstepping is computationally expensive. [148, 151] delayed feedback induces oscillations during transients and exhibits lag-effect. [49, 151] has not been tested experimentally.

5.2 Methodology

Given the nature of the problem and the presence of disturbances in the operating environment, a linear controller is unlikely to provide satisfactory performance, particularly without continuous linearization, which is computationally demanding. In light of this, and based on the review of underactuated systems with slung payloads, it is clear that the development of a nonlinear control technique is essential. For the purposes of this research, the control technique should be designed to be both effective and relatively simple to implement on the onboard controller. While fuzzy controllers have been utilized to control the attitude of airships [153, 154], height and attitude for multi-rotors [155], and payload suppression for quadrotors [151], there appears to be no experimental research, to the best of the author's knowledge, that investigates the application of fuzzy controllers for an airship with a slung load. The fuzzy logic controller should fill a gap in the literature by providing a control method that is straightforward to derive, easy to tune through simulations, and computationally efficient for implementation on the airship's onboard hardware.
To fulfil the objectives of the thesis, fuzzy controllers were developed for three flight modes: stationary payload (hover), planar flight ($X-Y$ plane), and payload delivery. In the stationary payload mode, the controller should maintain the position of the payload at the origin while minimizing oscillations caused by external disturbances. In the planar flight mode, a controller navigates to a desired payload position while minimizing payload oscillations. For payload delivery, an altitude controller descends to the desired reference altitude.

To satisfy the prescribed flight modes, five fuzzy controllers were designed, two for stationary payload, two for planar navigation, and one for payload delivery. It is worth noting that the control techniques discussed in this chapter for stationary payload mode fall into the Payload Trajectory Following category while planar flight and payload delivery belong to the Aircraft Trajectory Following category. The controllers are evaluated by performing point-to-point tasks while undergoing external disturbances. Following the design of the controllers and their assessment through simulations, an experimental outdoor test was conducted. This test aimed to validate both the model and the control strategies

developed in this thesis, while also establishing the airship's operational flight envelope.

5.2.1 General description of FLC with an example of designing a FLC for a cart-pendulum.

Fuzzy logic control (FLC) provides a relationship between inputs and outputs via fuzzy sets. The design process consists of four stages: fuzzification, rule base, inferencing, and defuzzification. Each stage is explained in detail in Appendix D.1. A simple example is presented in this section to illustrate the FLC processes. Consider the following example where a fuzzy controller is designed to alleviate the oscillations of a pendulum attached to a cart. To simplify the example, assume that the pendulum can only rotate in the plane with an angle θ and the cart is only allowed to move in the axial direction. The goal of the controller is to maintain θ close to zero. Two inputs were used in this example, error and rate of change of error. Error is defined to be the deviation of θ from zero.

$$E = \theta - \theta_{ref} = \theta - 0 = \theta \tag{5.1}$$

$$\dot{E} = \dot{\theta} \tag{5.2}$$

The output is the cart velocity that is needed, in the axial direction, to attenuate the pendulum oscillation. To employ a fuzzy controller, the processes of fuzzification, rule base establishment, inference, and defuzzification must be performed.

Fuzzification Three linguistic terms were used to denote membership functions (MF), negative (N), zero (Z), and positive (P). To keep the example simple, triangular membership functions are used. Note that many MF shapes are found in the literature and the triangular shape is among the most common [156]. Refer to Appendix D.1 for further details about other MF shapes. Having more MF will increase the control resolution but at the cost of being computationally more demanding. The assigned MF are shown in Fig. 5.1. Similarly, the fuzzification of the output is given in Fig. 5.2. In both figures the vertical axis (μ) represents the membership degree to which a given input belongs to a fuzzy set.

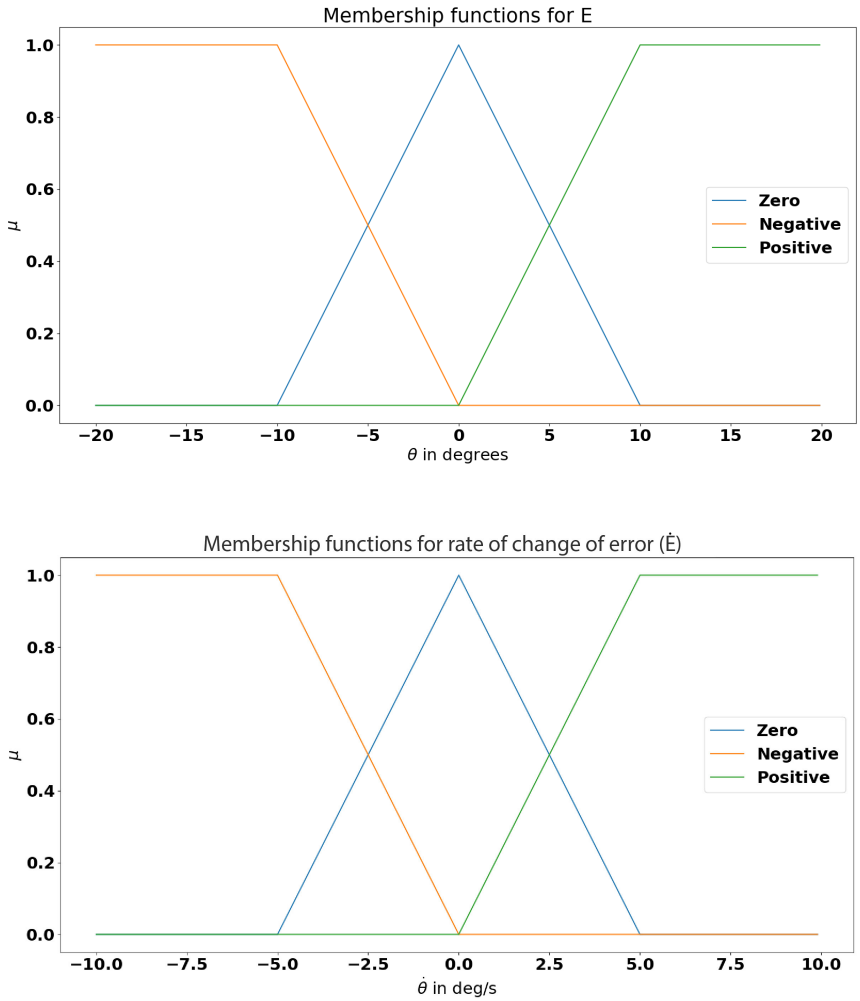


Figure 5.1: Membership functions for input. On the vertical axis, μ represents the membership degree

Rule Base The rules are often acquired through trial and error or expert knowledge. In the work of [142], time-delayed feedback of payload swing angles and angular velocity were used to construct the rules. However, proportional-derivative (PD) logic was used for this example. Also, it is worth mentioning that the output of [142] is distance whereas the output in the given example is the velocity of the cart. In this example, the rules are obtained from PD logic, given in Table 5.2. The top row represents the fuzzy set of the pendulum angles while the left column is the rate of change of pendulum angle fuzzy sets. In this example, the focus is on the rules that are circled in red. The circled rules, from

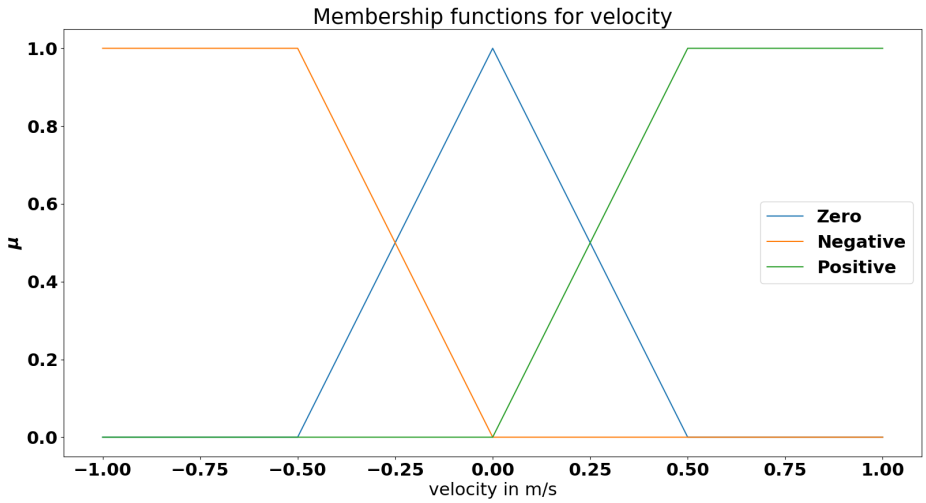


Figure 5.2: Membership functions for output. On the vertical axis, μ represents the membership degree

Table 5.2: Rule base to fuzzy controller

	N	Z	P
N	Z	N	Z
Z	Z	Z	Z
P	Z	P	Z

top to bottom, translate to the following

1. **if** $(angle = 0)$ **and** $(angular\ velocity < 0)$ **then** $(velocity < 0)$
2. **if** $(angle = 0)$ **and** $(angular\ velocity = 0)$ **then** $(velocity = 0)$
3. **if** $(angle = 0)$ **and** $(angular\ velocity > 0)$ **then** $(velocity > 0)$

Now assuming that the measured angle error of the pendulum and its rate of change, shown in Fig. 5.3, are $E = -6°$ and $\dot{E} = -2.5°/s$, respectively. The inputs are translated to fuzzy sets $E = [0.6, 0.4, 0]$ and $\dot{E} = [0.5, 0.5, 0]$ where each entity represents the degree of membership of the assumed inputs ($E = -6°$ and $\dot{E} = -2.5°/s$) to their respective fuzzy sets ($[N\ Z\ P]$).

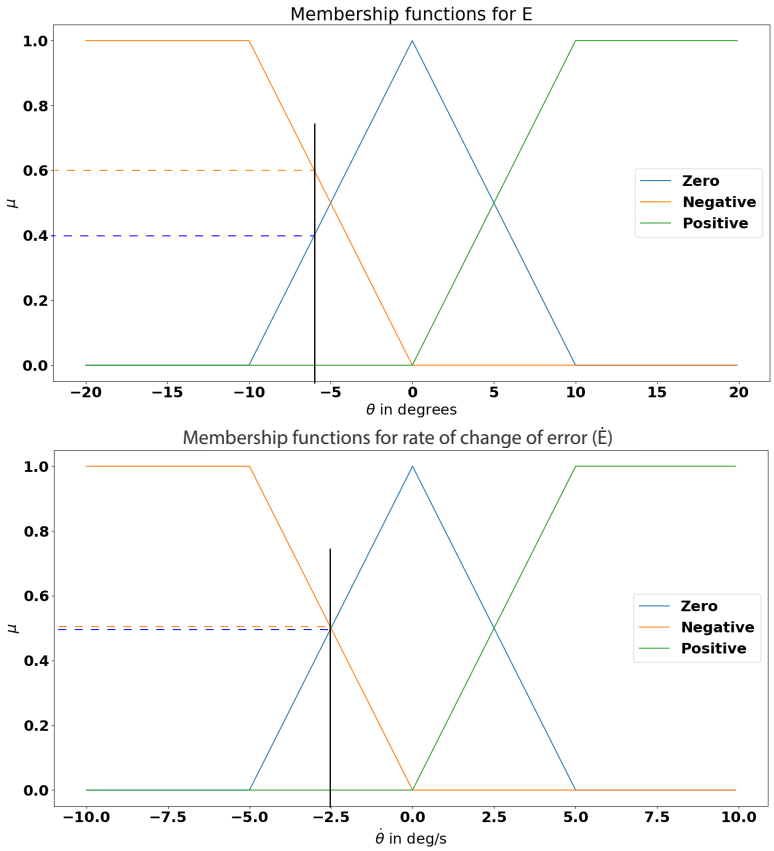


Figure 5.3: Degree of membership to each fuzzy set, based on the assumed inputs in the given example

Inferencing There are four rules that are active for the given inputs. Implication of firing rules can be written as follows

$$\mu_{velocity} = \mathbf{min}(\mu_E, \mu_{\dot{E}})$$

$$N \textbf{ and } N = Z \Rightarrow Z = \mathbf{min}(0.6, 0.5) = 0.5 \tag{5.3}$$

$$N \textbf{ and } Z = N \Rightarrow N = \mathbf{min}(0.6, 0.5) = 0.5 \tag{5.4}$$

$$Z \textbf{ and } N = N \Rightarrow N = \mathbf{min}(0.4, 0.5) = 0.4 \tag{5.5}$$

$$Z \textbf{ and } Z = Z \Rightarrow Z = \mathbf{min}(0.4, 0.5) = 0.4 \tag{5.6}$$

where μ_E, $\mu_{\dot{E}}$, and $\mu_{velocity}$ are, respectively, the degree of membership of cart angle error (input), rate of change of angle error(input), and the velocity of the cart (output). Implication is determined by taking the **minimum** firing degree of membership. Aggregation is then carried out by taking the **maximum** firing degree of membership. The activated

fuzzy sets, shown in (5.3)-(5.6), are aggregated to

$$Z = \mathbf{max}(0.5, 0.4) = 0.5$$

$$N = \mathbf{max}(0.5, 0.4) = 0.5$$

Defuzification Lastly, defuzzification is carried out using centre of gravity equation (D.2) to find the crisp output value, as demonstrated in Fig. 5.4

$$v_{cart} = \frac{S_1\ CG_1 + S_2\ CG_2}{S_1 + S_2} = -0.303 \text{ m/s}. \tag{5.7}$$

where v_{cart} is the velocity of the cart, S_1 and S_2 are the areas represented in amber and blue, respectively. Similarly, CG_1 and CG_2 correspond to the distance from the origin to, respectively, the centre of gravity of regions highlighted in amber and blue.

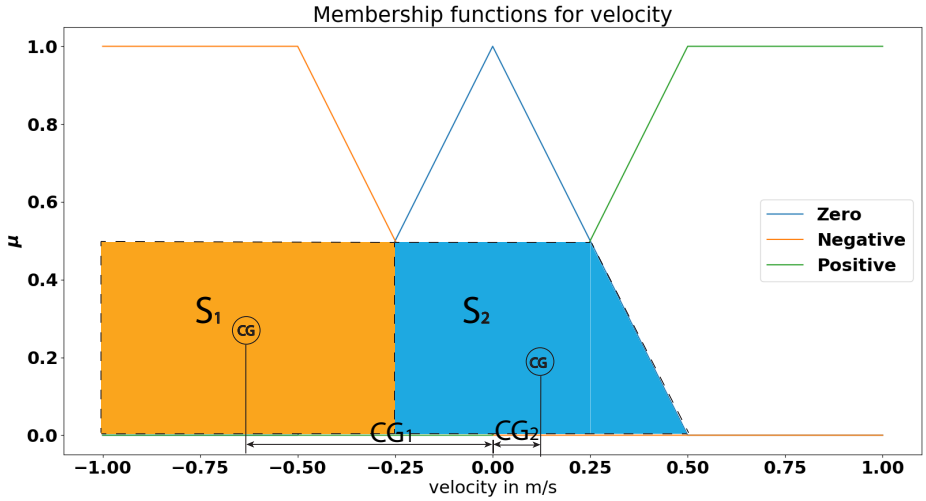


Figure 5.4: Centre of gravity of two regions, used to calculate the output crisp value.

5.2.2 Fuzzy Logic Controller for Stationary Payload

Due to the differing dynamics of the blimp in the axial and lateral directions, two separate fuzzy logic controllers (FLCs) were developed for the stationary payload mode: one to control the axial direction, while the other focused on the lateral direction of the payload. While it is possible to combine both FLCs into one FLC by superposing the inputs and outputs of the two controllers into one, it was found that having two separate controllers

yielded faster computations when implemented onboard. Since the purpose of the stationary payload controller is to attenuate the payload swinging while maintaining the position error of the payload close to zero, each FLC will have four inputs and one output. The inputs are the error of the payload's position, angle, and their rates of change while the output is the force along the controlled direction. The force outputs are effectuated by two forward and two lateral thrusters, respectively controlling the axial and lateral directions of the airship[3]. The sum of the forces produced by the forward thrusters is approximately 4 N while the lateral thrusters can produce up to 2 N. Since thrusters produce higher force magnitudes in one direction, this force asymmetry should be accommodated in the design process of the controllers. To account for the worst-case scenario, the upper bounds were set based on the maximum forces the thrusters can generate in their least efficient direction: 3 N for the forward thrusters and 1.5 N for the lateral thrusters. The thrust values were measured using propeller thrust stand [157] as detailed in Appendix D.6. The process of fuzzification is elaborated next.

Initially, five membership functions (MFs) were defined for each input and seventeen MFs for the output. With this many MFs and inputs, it was found that the hardware (RPi) takes a substantial amount of time to compute an output[4]. Therefore, the number of MFs was reduced to three and nine for the inputs and output, respectively. A conventional way of naming the MFs is to use negative, zero, and positive for inputs that are, respectively, less than, equal, and larger than zero [156]. The same classification is adopted for the output MF while iteratively labelling the various levels of negative and positive MFs. That is, an output value that falls below negative is called negative-negative (or "NN" for brevity) all the way to negative-negative-negative-negative ("NNNN") as shown in Fig. 5.5. Similarly, the positive MFs have four levels starting with "P" up to "PPPP". All MFs are chosen to be triangular-shaped for two key reasons. First, their linear nature makes them easier to define, implement, and compute, reducing computational complexity and resulting in faster performance, particularly in real-time applications. Second, triangular MFs provide clear and simple boundaries for fuzzy sets, which enhances the interpretability of the system

[3]Recall that the system is underactuated, hence, the payload position can only be (indirectly) controlled through the thrusters installed on the gondola.

[4]Each computation takes around 2 seconds on Raspberry Pi 4B

and makes it more straightforward to adjust and tune, contributing to overall ease of use in design and optimization.

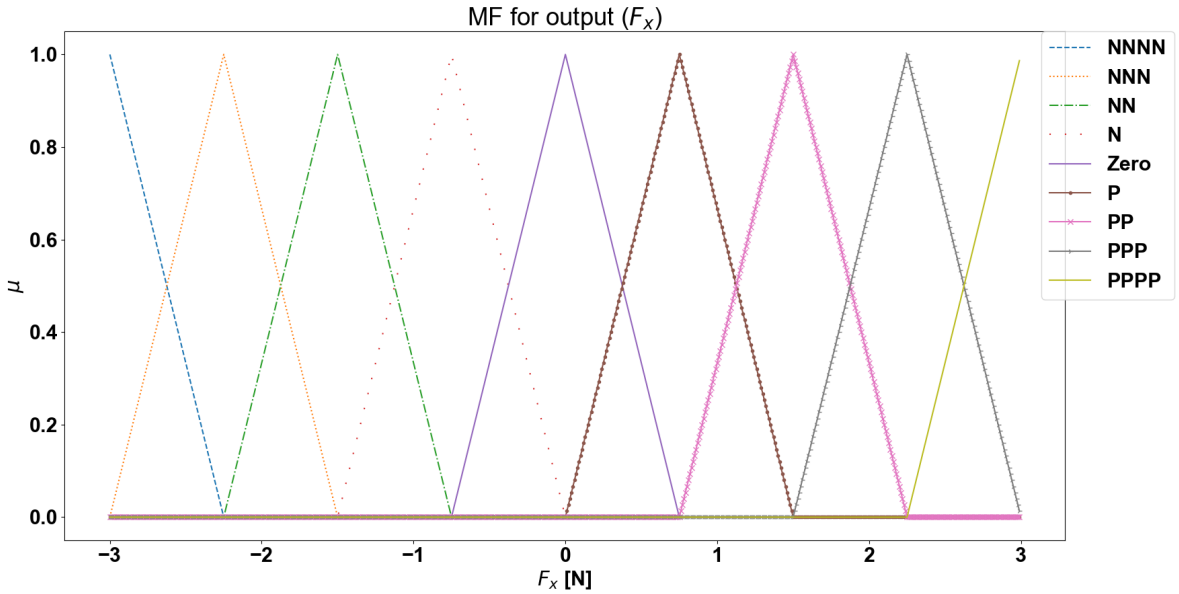


Figure 5.5: Fuzzy sets for output force in the axial direction.

The range of the input and output variables is given in Table 5.3 for axial and lateral directions. The range of input variables is determined through optimization while the output range is dictated by the physical limitations of the thrusters used.

Table 5.3: Input and output variables of stationary payload FLC in the axial and lateral directions.

	Variables		Range (Min — Max)		
	Axial	Lateral	Axial	Lateral	Units
Input	E_{θ_c}	E_{ϕ_c}	$-\pi/2$ — $\pi/2$		[rad]
	$\dot{E}_{\theta_c}$	$\dot{E}_{\phi_c}$	-1 — 1		[rad/s]
	E_{x_L}	E_{y_L}	-1 — 1		[m]
	$\dot{E}_{x_L}$	$\dot{E}_{y_L}$	-1 — 1		[m/s]
Output	u_x	u_y	-3 — 3	-1.5 — 1.5	[N]

Recall that the error (E) and its rate of change ($\dot{E}$) are defined as

$$E = \chi - \chi_{ref}$$
$$\dot{E} = \dot{\chi}$$

where χ and χ_{ref} are, respectively, the measured and desired input.

It is worth noting that the input range serves as a scaling factor, where a narrower input range will carry greater weight in the FLC, resulting in a higher output compared to inputs with a wider range. For example, the payload's position error has more impact on the force output over the payload's angle error (Table 5.3).

Moreover, the distribution of MF needs to be tuned. That is, each triangular-shaped MF consists of three parameters: the starting point, middle point, and end point. The tuning of these parameters was carried out using Particle Swarm Optimization (PSO) in MATLAB. It is important to highlight that other global optimization methods, such as Genetic Algorithm, could also be employed as an alternative. The optimal solution is determined by minimizing cost functions (5.8)-(5.9) that were prescribed for the investigated vehicle. The cost function used to tune the FLC is a quadratic function of the controller inputs summed over the simulation time. Using a quadratic cost function for optimization is a widely accepted approach in the literature, often considered fundamental in control optimization. Its popularity stems from its simplicity, smoothness, and ability to penalize larger errors more effectively. As such, the author adopted a quadratic cost function to tune the FLCs, leveraging its well-understood properties to ensure that substantial deviations from desired behaviour are minimized efficiently during optimization.

$$J_1 = \int_0^{t_f} \left({E_{\theta_c}}^2 + {\dot{E}_{\theta_c}}^2 + {E_{x_L}}^2 + {\dot{E}_{x_L}}^2 \right) dt \tag{5.8}$$

$$J_2 = \int_0^{t_f} \left({E_{\phi_c}}^2 + {\dot{E}_{\phi_c}}^2 + {E_{y_L}}^2 + {\dot{E}_{y_L}}^2 \right) dt \tag{5.9}$$

where t_f is the simulation time. Given that the optimization process requires numerous simulation attempts, the simulation time was set to 40 seconds ($t_f = 40s$). This duration was determined by carefully considering the time needed for the controller to stabilize the system in a physical setup while also maintaining a computationally manageable simulation length. The distribution of the optimized MFs are given in Appendix D.2

The rules for the stationary payload controller are based on proportional-derivative logic (5.10)-(5.11)

$$u_x = E_{\theta_c} + \dot{E}_{\theta_c} - (E_x + \dot{E}_x) \tag{5.10}$$

$$u_y = E_{\phi_c} + \dot{E}_{\phi_c} - (E_y + \dot{E}_y) \tag{5.11}$$

The opposite relationship, between the angle error and position error of the payload, is introduced to alleviate the oscillations that would occur as a result of an aggressive force output as will be demonstrated in the simulations. In addition, (5.10)-(5.11) are used to construct the rules for the stationary payload FLC by exploiting the following operations:

$$\begin{aligned} P + P &= PP & \qquad P - N &= PP \\ N + N &= NN & \qquad N - P &= NN \\ Zero \pm Zero &= Zero & \qquad P + N &= Zero \\ P - P &= Zero & \qquad N - N &= Zero \end{aligned}$$

With these operations, the rules can be presented in the linguistic form. For example, if all inputs were Positive (P), the output, based on (5.10)-(5.11), would be

$$u = P + P - (P + P) = Zero \tag{5.12}$$

Reformulating (5.12) to the linguistic form gives (5.13)

$$\textbf{if} \ \ (\chi_1 = P) \ \ \textbf{and} \ \ (\dot{\chi}_1 = P) \ \ \textbf{and} \ \ (\chi_2 = P) \ \ \textbf{and} \ \ (\dot{\chi}_2 = P) \ \ \textbf{then} \ \ (u = Zero) \tag{5.13}$$

where the subscript is used to enumerate the inputs. For instance, $\chi_1 = E_{\theta_c}$ and $\chi_2 = E_{x_L}$. Iterating the fuzzy sets over all possible combinations yields a total of 81 rules for each controller. All rules are included in Appendix D.2, Table D.1.

The inferencing method is adopted from Mamdani and Assilian [158] which utilizes *minimum* for implication and *maximum* for aggregation, as discussed in the example provided at the beginning of the chapter.

The last step is defuzzification which is used to find the crisp output (a force) required to stabilize the payload. Defuzzification is carried out using the centre of gravity method (D.2) and explained in Appendix. D.1.

5.2.3 Fuzzy Logic Controller for Planar Navigation

Two fuzzy logic controllers (FLCs) have been developed to control and navigate the multi-body airship in the $X-Y$ plane. One FLC is tasked with delivering thrust, which is called thrust controller, while the other is responsible for controlling yaw. Both controllers utilize the forward thrusters, with the yaw controller relying on differential thrust to produce the necessary yaw moment. The thrust controller receives three inputs and generates a single output. The inputs are the distance error to the goal ($E_{|\vec{r}|}$), the rate of change of distance error ($\dot{E}_{|\vec{r}|}$), and the yaw angle error. The latter input was used to ensure that the planar controller only operates when the yaw angle error is within a prescribed margin[5]. The distance is calculated by taking the norm of the position vector to the goal. The distance error and its rate of change are defined in (5.14)-(5.15)

$$E_{|\vec{r}|} = \sqrt{(x_B - x_{ref})^2 + (y_B - y_{ref})^2 + (z_B - z_{ref})^2} \tag{5.14}$$

$$\dot{E}_{|\vec{r}|} = \sqrt{{V_{x_B}}^2 + {V_{y_B}}^2 + {V_{z_B}}^2} \tag{5.15}$$

Five triangular-shaped MFs are used for the fuzzification of distance error and its rate of change. On the other hand, three MFs are used for the yaw angle. The output is fuzzified using nine triangular-shaped MFs. As mentioned before, having more MFs adds more resolution to the control output at the cost of increasing the computational demand. It was found that increasing the number of MFs beyond nine caused substantial computational delays. Therefore, nine MFs were utilized to maintain a smooth control flow. The range of input and output variables is presented in Table 5.4. While the input range is tuned using PSO, the output range is dependent on actuation capabilities of the physical prototype. If the input exceeds the specified range, the algorithm selects the nearest value within that

[5]That is $-12.5^\circ \leq E_{\psi_B} \leq 12.5^\circ$.

range. For example, if the distance error $E_{|\vec{r}|} = 5$ m, the algorithm will adjust the input to $E_{|\vec{r}|} = 2.5$ m, as this represents the upper bound.

Table 5.4: Input and output variables of thrust FLC.

	Variables	Range (Min — Max)	
Input	$E_{\|\vec{r}\|}$	0 — 2.5	[m]
	$\dot{E}_{\|\vec{r}\|}$	0 — 0.5	[m/s]
	E_{ψ_B}	$-\pi/2$ — $\pi/2$	[rad]
Output	u_x	-3 — 3	[N]

When the yaw angle exceeds the prescribed margin, the yaw controller takes over. The yaw controller takes two inputs and delivers one output. The inputs are the yaw error (E_{ψ_B}) and its rate of change ($\dot{E}_{\psi_B}$) whereas the output is a yaw moment (u_{mz}). Similar to the thrust controller, five and nine MFs were used for the fuzzification of the inputs and output, respectively. Table 5.5 shows the designated range for input and output variables. The limits for yaw moment (u_{mz}) are determined by multiplying the thrust limits by the perpendicular distance to the thrusters.

Table 5.5: Input and output variables of yaw FLC.

	Variables	Range (Min — Max)	
Input	E_{ψ_B}	-0.2π — 0.2π	[rad]
	$\dot{E}_{\psi_B}$	-0.16π — 0.16π	[rad/s]
Output	u_{mz}	-2.5 — 2.5	[N m]

To optimize the MF distribution and input range of thrust and yaw FLCs, PSO was used to minimize the cost functions associated with thrust and yaw FLCs, denoted as J_3 (5.16) and J_4 (5.17), respectively.

$$J_3 = \int_0^{t_f} ({E_{|\vec{r}|}}^2 + {\dot{E}_{|\vec{r}|}}^2)\, dt \tag{5.16}$$

$$J_4 = \int_0^{t_f} ({E_{\psi_B}}^2 + {\dot{E}_{\psi_B}}^2)\, dt \tag{5.17}$$

where t_f is the simulation time and it is set to 40 seconds. The optimized membership function distribution is presented in Appendix D.3.

The rules for planar navigation FLC are derived from the principles of the PD control strategy (5.18)-(5.19). In this approach, both the thrust and yaw moment increase in response to a growing negative error in distance and yaw angle, as well as their respective rates of change.

$$u_x = (E_{|\vec{r}|} + \dot{E}_{|\vec{r}|}) \tag{5.18}$$

$$u_{mz} = -(E_{\psi_B} + \dot{E}_{\psi_B}) \tag{5.19}$$

Rules are tabulated and presented in Appendix.D.3, Table D.2 and Table D.3. The inferencing and defuzzification methods are identical to those used in stationary payload FLC.

A schematic for the desired payload position controller is illustrated in Fig. 5.6. The

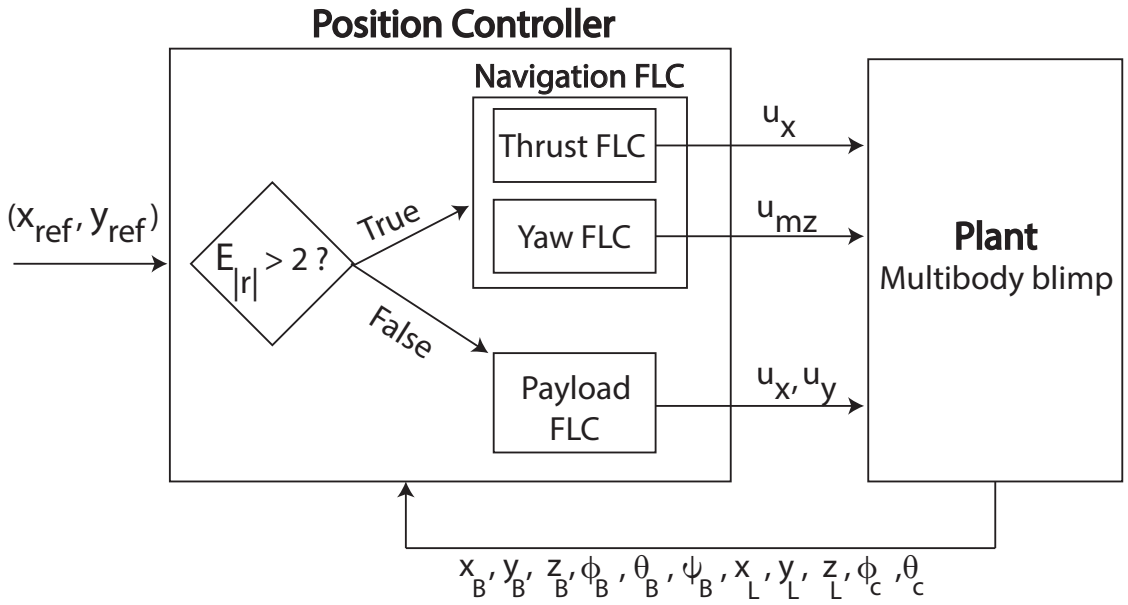


Figure 5.6: Schematic of the position controller delineating the sequence of operations among the thrust, yaw, and payload FLCs

navigation FLC operates until the airship is within two metres of the target then the payload FLC assumes control to attenuate the oscillations and stabilize the slung payload

at the desired position. The two-metre threshold was selected based on a parametric study, which demonstrated that the performance of the payload FLC deteriorates beyond this distance. Reducing the threshold below two metres did not yield significant improvements in performance, and it also poses practical challenges for experimental testing, as the GPS accuracy is limited to approximately two metres in optimal conditions.

5.2.4 Fuzzy Logic Controller for Payload Delivery

A dedicated fuzzy logic controller (FLC) was designed for payload delivery, using two inputs: altitude error (E_{z_B}) and its rate of change ($\dot{E}_{z_B}$). The controller generates a vertical force applied to adjust altitude. To apply this vertical force, servos rotate the forward thrusters to point upward. Five MF were used for each input and nine MF for the output. Following the same optimization steps as previous controllers, the membership function distribution and input range were optimized using PSO to minimize the cost function (5.20).

$$J_5 = \int_0^{t_f} ({E_{z_B}}^2 + {\dot{E}_{z_B}}^2)\, dt \tag{5.20}$$

The MF range for delivery FLC is presented in Table 5.6 while the optimized MFs are given in Appendix D.4.

Table 5.6: Input and output variables of payload delivery FLC.

	Variables	Range (Min — Max)
Input	E_{z_B}	-2 — 2 $[m]$
	$\dot{E}_{z_B}$	-0.5 — 0.5 $[m/s]$
Output	u_z	-3 — 3 $[N]$

The rules for the FLC are deduced from PD logic given in (5.21)

$$u_z = -(E_{z_B} + \dot{E}_{z_B}) \tag{5.21}$$

The complete set of rules are given in Appendix D.4. The inferencing and defuzzification methods are, respectively, Mamdani and centre of gravity.

5.3 Results and Discussion

As mentioned at the beginning of the chapter, the controllers were tested in point-to-point hover missions by considering the following scenarios:

1. Starting the simulation with non-zero payload angle (θ_c).
2. Applying a step excitation to the payload.
3. Demanding a desired payload position.
4. Demanding a desired altitude.

The first two scenarios were tested using FLC for stationary payload while the third scenario utilized the planar navigation FLC. The last test case was achieved using the payload delivery FLC. Additional simulations were carried out with wind disturbances. These tests will provide valuable insights into the controller's performance under ideal conditions and wind disturbances.

5.3.1 Simulation Results

Results obtained from simulations are presented for the three scenarios mentioned earlier. The control output is introduced to the EOM through $\vec{U}$ (refer to Chapter 2). All simulations were run for 40 seconds. While other durations could have been selected for the simulations, the author chose to maintain the same duration as that used when tuning the FLCs. Unless otherwise mentioned, the initial conditions for simulations are given in Table 5.7

Non-zero initial payload angle

The stationary payload controller is used in this scenario to stabilize the payload position at the origin while minimizing the oscillations. The results are presented in Fig. 5.7 where the payload angles are initialized with $\phi_c = 0, \theta_c = 60°$. The controller was able to stabilize

Table 5.7: Initial conditions for simulation with controller

Parameter	Value	Units
$\vec{r}_B$	$[0, 0, 0]$	m
$\vec{r}_G$	$[0, 0, 1.16]$	m
$\vec{r}_L$	$[0, 0, 2.16]$	m
$\vec{V}_B = \vec{V}_G = \vec{V}_L$	$[0, 0, 0]$	m/s
$\vec{\theta}_B = \vec{\theta}_G = \vec{\theta}_L$	$[0, 0, 0]$	rad
$\phi_c = \theta_c$	0	rad
$\vec{\omega}_B = \vec{\omega}_G = \vec{\omega}_L$	$[0, 0, 0]$	rad/s

the position of the payload within ± 0.2 m from the origin.

Far right of Fig. 5.7 depicts the payload oscillations during the simulation time.

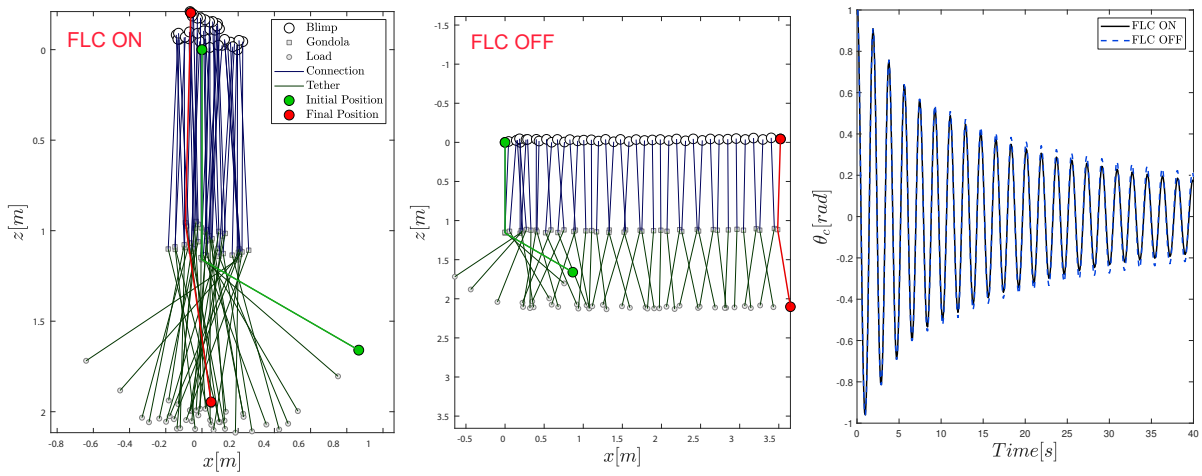


Figure 5.7: Simulation results demonstrating the stationary payload FLC. Initial payload angles $\phi_c = 0, \theta_c = 60°$.

The oscillations diminish over time until they settle within a confined angle range of $\pm 11°$ ($-11° \leq \theta_c \leq 11°$) by the end of the simulation. The results suggest that the payload oscillations are just slightly lower when FLC is active. Since the objective of the experiment is to maintain the payload stationary, the position of the payload is given more emphasis

over the oscillations which would take more time to attenuate.

Step excitation applied at the payload

A step force of one Newton is applied to the payload for one second. Figure 5.8 portrays the results when a step force is applied along the axial and lateral directions. In both cases, the stationary payload FLC stabilized the position of the payload within 0.01 m from the origin with minimal oscillations.

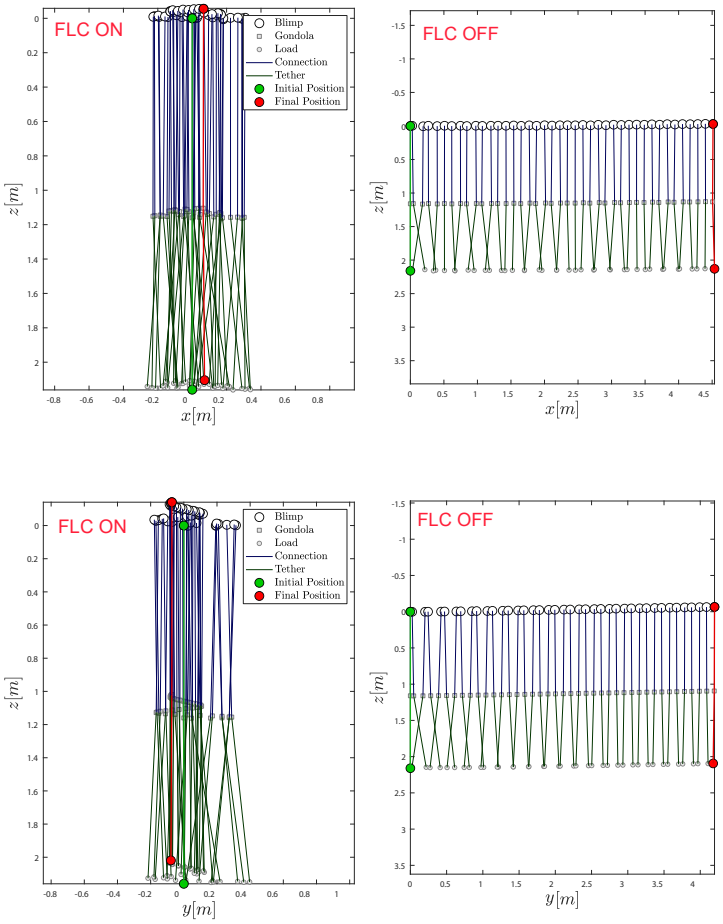


Figure 5.8: Simulation results demonstrating the stationary payload FLC when a step force is applied to the payload along the axial (top) and lateral (bottom) directions.

Desired payload position

The previous tests focused on maintaining the payload position near the origin using stationary payload FLC. In this test, a desired payload position, in X-Y plane, needs to be achieved. To fulfil this task, FLC for planar navigation is utilized along with the stationary payload FLC, as described earlier at the end of Fuzzy Logic Controller for Planar Navigation.

Figure 5.9 illustrates the results obtained when a desired position of eight metres in the axial and lateral directions is demanded. Although there is a minor overshoot, the desired position is eventually reached with very little oscillations. The overshoot can be attenuated with further tuning of MF.

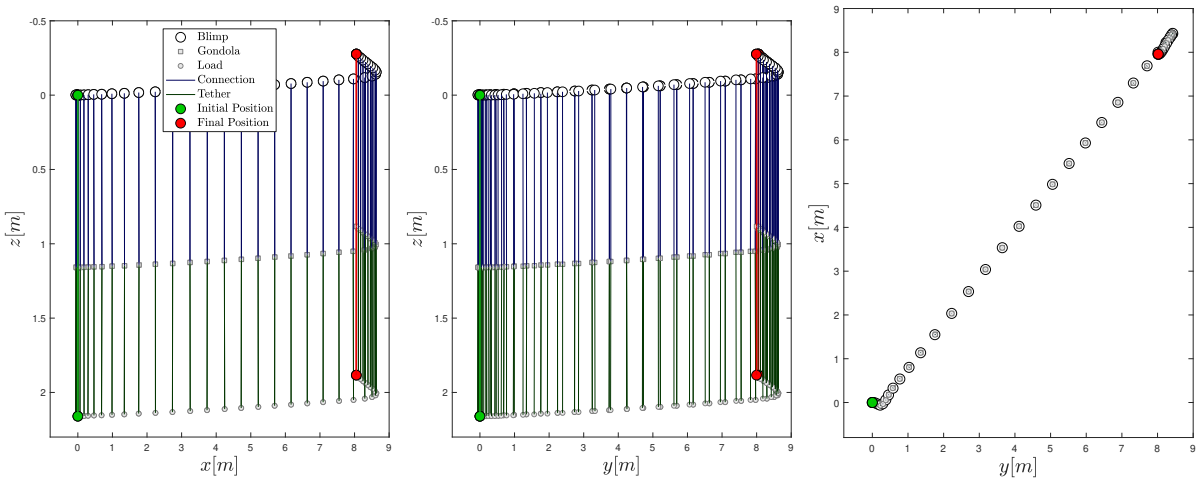


Figure 5.9: Simulation results demonstrating the planar navigation FLC when the desired position is eight metres in both axial and lateral directions ($x = 8, y = 8$).

Two additional runs were carried out using the Dryden wind turbulence model at two wind speeds: ten and twenty kilometres per hour. Figures 5.10 and 5.11 demonstrate the results obtained using position controller for wind speeds 10 km/h and 20 km/hr, respectively. A similar overshoot is observed at 10 km/h wind speed with a slight deviation in the altitude. As the wind speed increases to 20 km/h, the overshoot is attenuated but the error in the axial direction increased by roughly half a metre. The rise in axial

position error is caused by a significant spike in the axial component of wind velocity towards the end of the simulation (not shown). Nevertheless, the FLC shows robustness against wind disturbance up to 20 km/h. As wind speeds increased beyond 20 km/h, the controller struggled to maintain position. This is attributed to the limitations in the thrust capabilities of the motors.

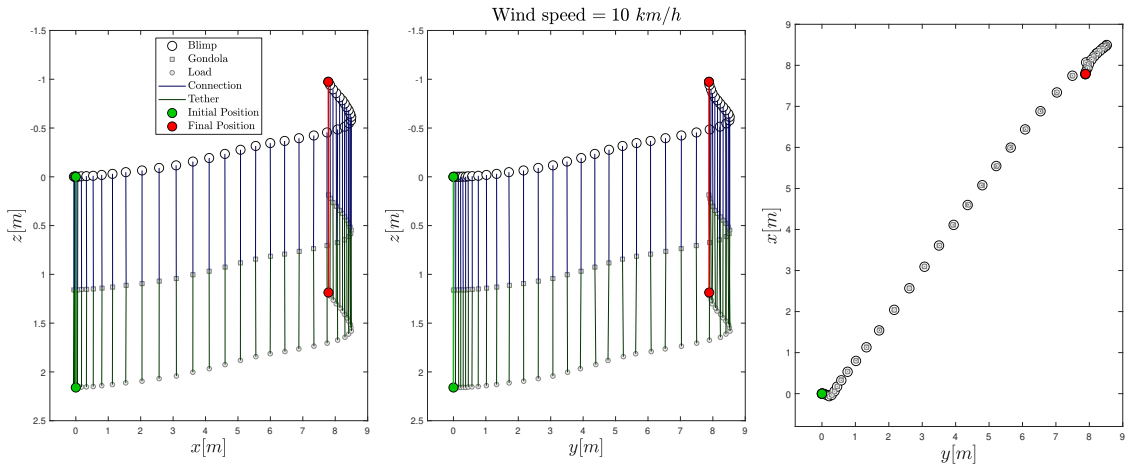


Figure 5.10: Position controller with wind disturbance of 10 km/h. The desired position is eight metres in both axial and lateral directions ($x = 8$ m, $y = 8$ m).

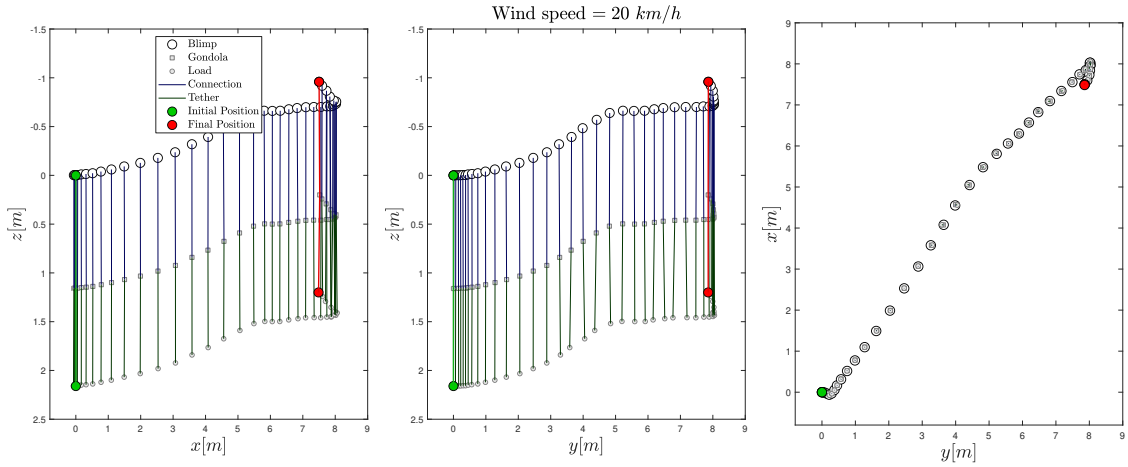


Figure 5.11: Position controller with wind disturbance of 20km/h. The desired position is eight metres in both axial and lateral directions ($x = 8$ m, $y = 8$ m).

Desired altitude

The goal in this test is to deliver the payload by controlling the altitude using the payload delivery FLC.

Figure 5.12 delineates the results for a target altitude of two meters. While the controller successfully reached the desired altitude, a minor deviation occurred in the axial direction. Although an additional FLC could be developed to actuate the servos for thrust vectoring, the current altitude FLC uses a fixed servo angle, limiting its ability to counteract these offsets in the longitudinal plane.

Controlling the servos to achieve thrust vectoring was inconvenient with the current unmanned airship prototype for two main reasons. First, the servos are restricted to a 120-degree rotation range. Second, the orientation of the servos may shift if the thrusters are removed and reinstalled, necessitating remodelling of the PWM-to-servo angle relationship.

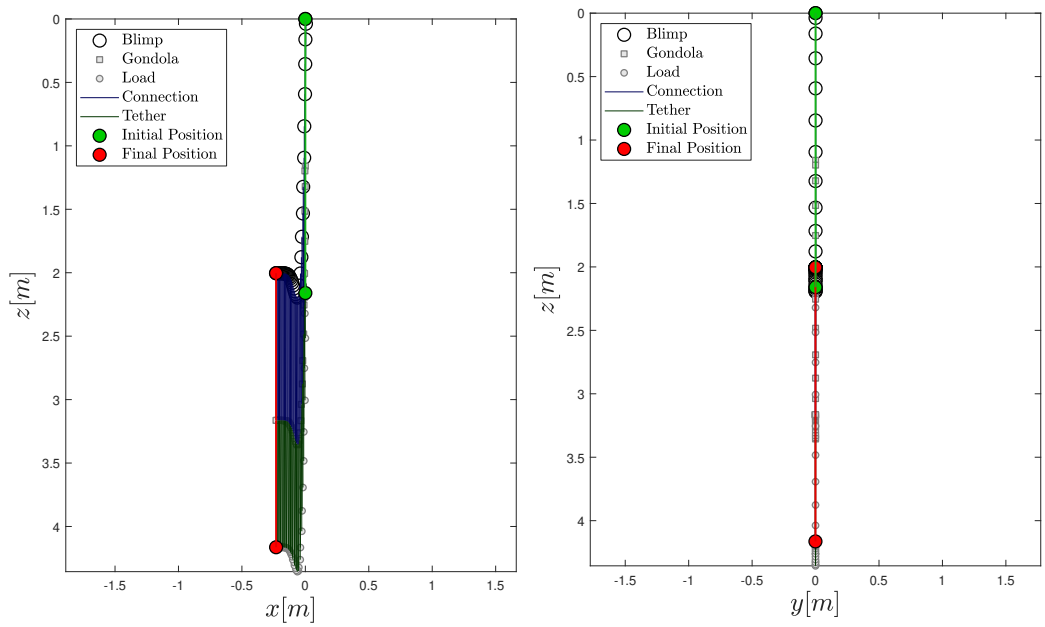


Figure 5.12: Simulation results demonstrating the payload delivery FLC.

5.3.2 Experimental Setup and Flight Test Results

The experimental flight tests were carried out in a farm field of approximately 9000 m^2 in the township of Russell, Ontario. The average wind speed was approximately 12 km/h with gusts that were almost twice the wind speed. It is worth noting that the gust speed was not measured but approximated based on the weather forecast. Figure 5.13 presents the wind data at the Ottawa airport, located approximately 30 km from the test site, with the time frame corresponding to the experiment highlighted.

Hourly Data Report for September 30, 2024

TIME LST	Temp °C	Dew Point °C	Rel Hum %	Precip. Amount mm	Wind Dir 10's deg	Wind Spd km/h	Visibility km	Stn Press kPa	Hmdx	Wind Chill	Weather
00:00	15.8	15.8	100		35	11	24.1	100.37			NA
01:00	15.2	14.9	98		36	14	24.1	100.40			Clear
02:00	14.7	12.9	89		1	17	24.1	100.41			NA
03:00	14.1	12.1	88		2	15	24.1	100.44			NA
04:00	13.3	11.5	88		1	14	24.1	100.49			Clear
05:00	12.4	11.3	93		1	10	24.1	100.56			NA
06:00	11.9	10.9	93		1	10	24.1	100.61			NA
07:00	13.0	11.7	91		1	11	24.1	100.67			Clear
08:00	14.7	11.3	80		4	12	24.1	100.71			NA
09:00	16.5	12.0	75		3	8	24.1	100.73			NA
10:00	18.2	11.8	66		9	14	24.1	100.70			Clear
11:00	19.6	12.0	61		7	13	24.1	100.67			NA
12:00	21.2	12.6	58		3	12	24.1	100.62			NA
13:00	22.7	13.6	56		3	8	24.1	100.55	26		Clear
14:00	22.5	12.9	54		8	12	24.1	100.50	25		NA
15:00	23.0	11.4	48		8	12	24.1	100.47	25		NA
16:00	22.6	12.0	51		8	11	24.1	100.45	25		Clear

Figure 5.13: Wind data showing recorded wind speed at experiment day. Obtained from the Canadian Climate Data [159].

With large wind gusts, testing the stationary payload controller proved highly challenging, especially given that the controller's maximum thrust output was limited to 3 N. Consequently, it was determined that only the navigation controller would be tested in the direction of the wind. The schematic of navigation FLC using the onboard sensors is presented in Fig. 5.14. The figure illustrates the workflow of the position controller, em-

phasizing the conditions that trigger the navigation FLC. Only one optical sensor was used to measure the slung payload's angle (θ_c), hence, only the axial position for the payload was available (x_L). Refer to Chapter 1.3 for further details about the onboard sensors and to Appendix D.5 for calculations of axial payload position.

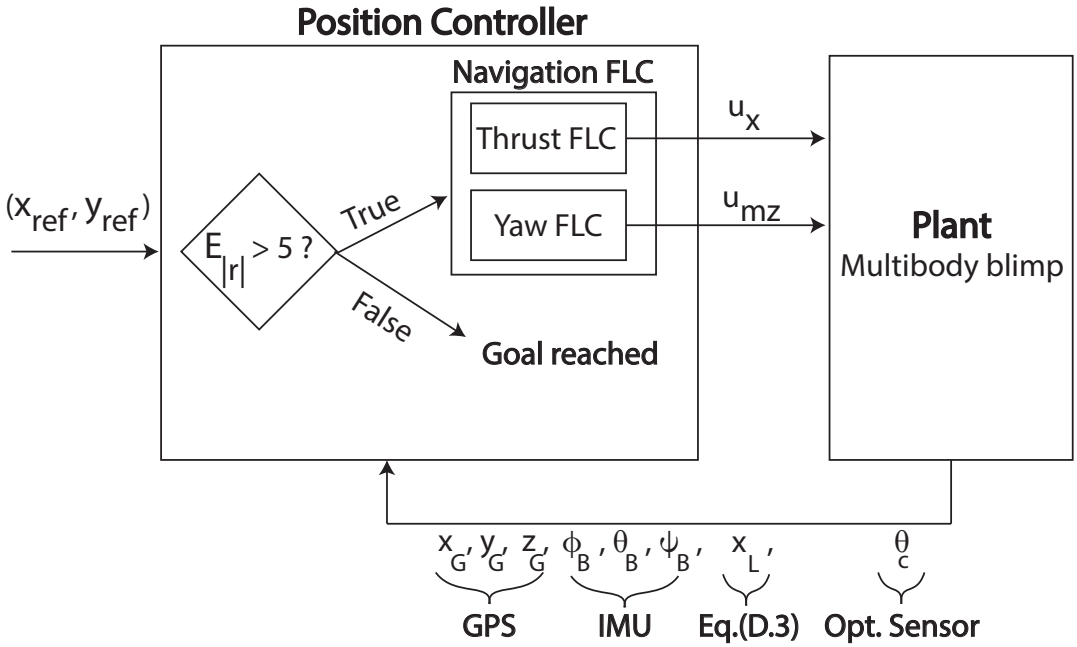


Figure 5.14: Schematic illustrating the position controller integrated into the flight controller.

Before each test, the compass was calibrated and the GPS fidelity was verified to be accurate within 5 m margin of error. As the envelope is filled with helium, counterweights with a total mass of 0.335 kg were added to the gondola such that the airship reaches neutral buoyancy at an altitude of approximately 11 m. This elevation was sufficient to fly the airship above the trees to ensure clear line of sight with the GPS satellites. Moreover, the payload is attached to the gondola by a 4-metre long tether, with an additional 7-metre tether for security. This extension serves as a safety measure, allowing it to be grabbed by hand in case of an emergency. The procedure to start the test is described as follows: one person holds the airship by the tether while another person prepares the connections to start the control code. A hotspot was used to connect the onboard computer to a ground station. The control code includes a 45-second delay before the controller activates,

allowing the person holding the airship sufficient time to release it while the operator at the ground station sends the code to the background of the onboard computer. Sending the control code to the background allows for the code to run if wifi connection is lost. While the code operates in the background at a rate of 5 Hz[6], it continuously monitors for a pre-configured RC signal, which functions as a kill switch for the controller and can be activated in case of an emergency. Figure 5.15 shows the blimp while it is being prepared for testing.

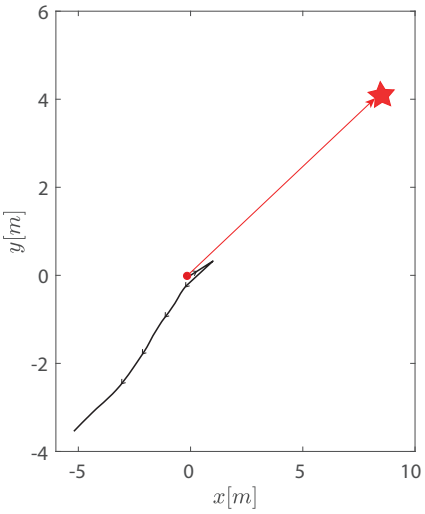
Figure 5.15: Left: blimp's envelope being inflated with helium. Right: blimp with slung payload in neutral buoyancy.

In the first navigation control test, the desired position was given as a coordinate point, which is illustrated in Fig. 5.16. The circle represents the starting point while the star is the goal. The wind during this test was a headwind, and the gusts were strong enough to push the airship backward opposite to the goal. At the beginning of the test, the controller managed to move the airship in the direction of the goal. However, as the gust intensifies, the thrusters could not produce enough thrust to counter the wind, causing the airship to move backwards. The test was terminated when the airship was pushed far away in the opposite direction it is supposed to go, as depicted in Fig. 5.16b. The control output is

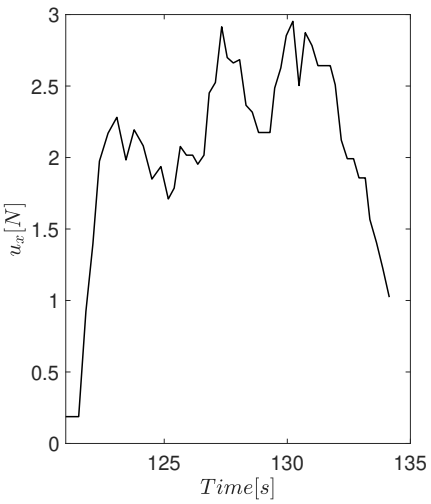
[6]The highest sampling rate the flight controller was able to achieve.

presented in Fig. 5.16c where the controller output is almost saturated, yet it was not enough to counter the wind.

(a) Satellite view



(b) $X - Y$ trajectory.

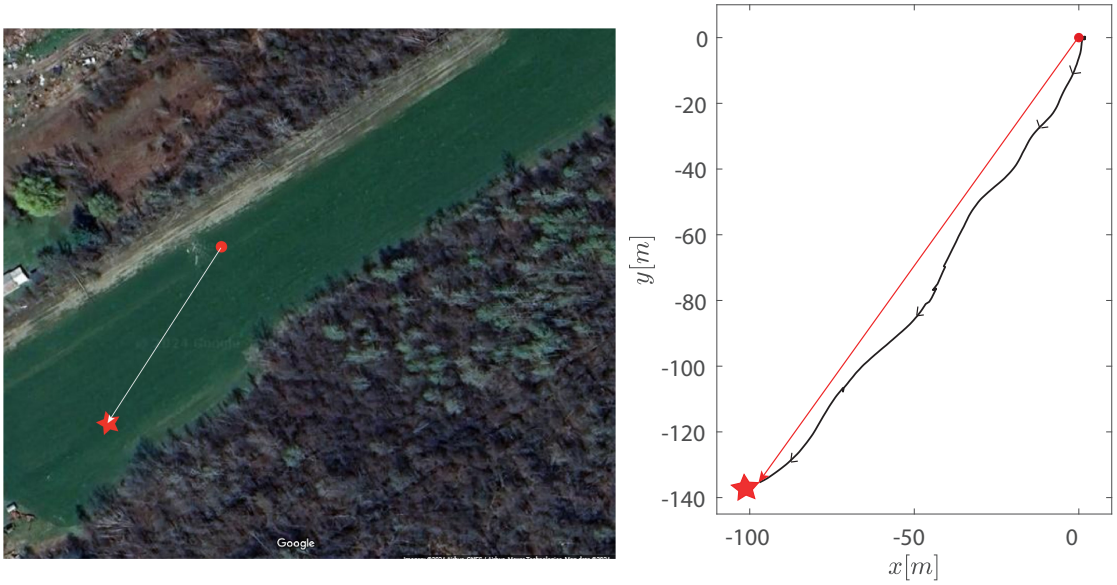


(c) Control output (u_x).

Figure 5.16: The first navigation control test, high headwind. Red circle is the starting point while star is the goal. Black and red solid lines, respectively, represent the airship and desired trajectories.

Results similar to those shown in Fig. 5.16 were acquired several times before a break in the wind occurred. Figure 5.17 illustrates a successful trajectory tracking test under an estimated 5 to 10km/h wind speed (based on qualitative observations by the participants

during the test) along with the satellite view. It is clear that the controller performs much better under low gust.



(a) Satellite view

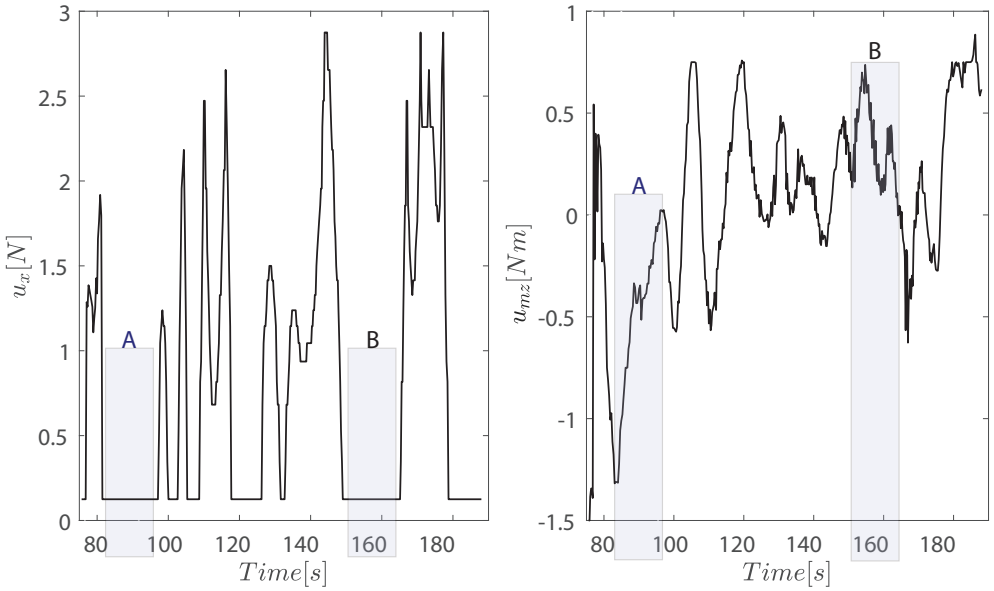
(b) $X - Y$ trajectory.

Figure 5.17: The successful navigation control test, low tailwind. Red circle is the starting point while star is the goal. Black and red solid lines, respectively, represent the airship and desired trajectories.

Examining Fig. 5.18 which manifests the controller inputs and outputs during the successful experiment, it is clear that there are times where the control force output defaults to zero in the axial direction (Fig. 5.18a). This happens whenever the error in the yaw angle exceeds a threshold[7]. For example, the yaw error goes as high as 1 rad at the highlighted region A and as low as -0.5 rad at region B (Fig. 5.18b). In both cases, the yaw error exceeds the prescribed threshold, resulting in zero axial force. Conversely, the control moment output (u_{mz}), which applies differential thrust to correct the yaw error, increases in the highlighted regions where the yaw error is pronounced (Fig. 5.18c). Once the yaw error is compensated, the control force output returns to generating thrust, propelling the airship toward the goal. The position error ($E_{|r|}$) decreases consistently throughout the flight reaching a value of roughly 5 m towards the end (Fig. 5.18d). Given that the GPS

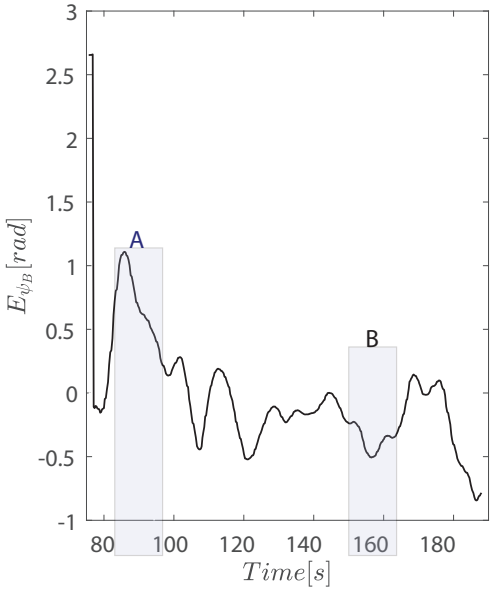
[7]When $|E_{\psi_B}| > 0.2$ *radians*

accuracy is within five metres, the test is deemed complete and terminated upon reaching the five metres mark.

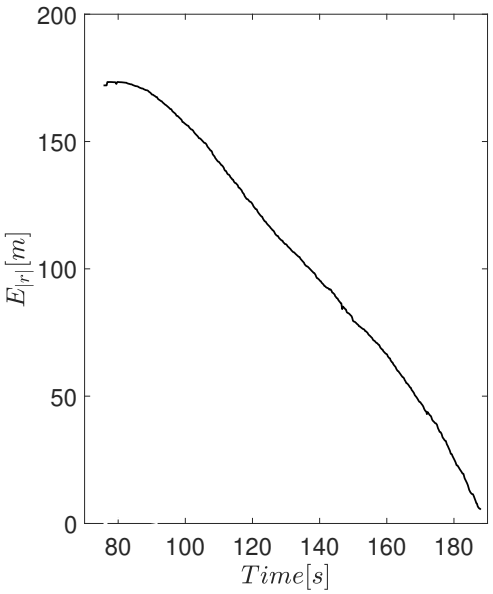


(a) Axial force output (u_x).

(b) Yaw moment output (u_{mz}).



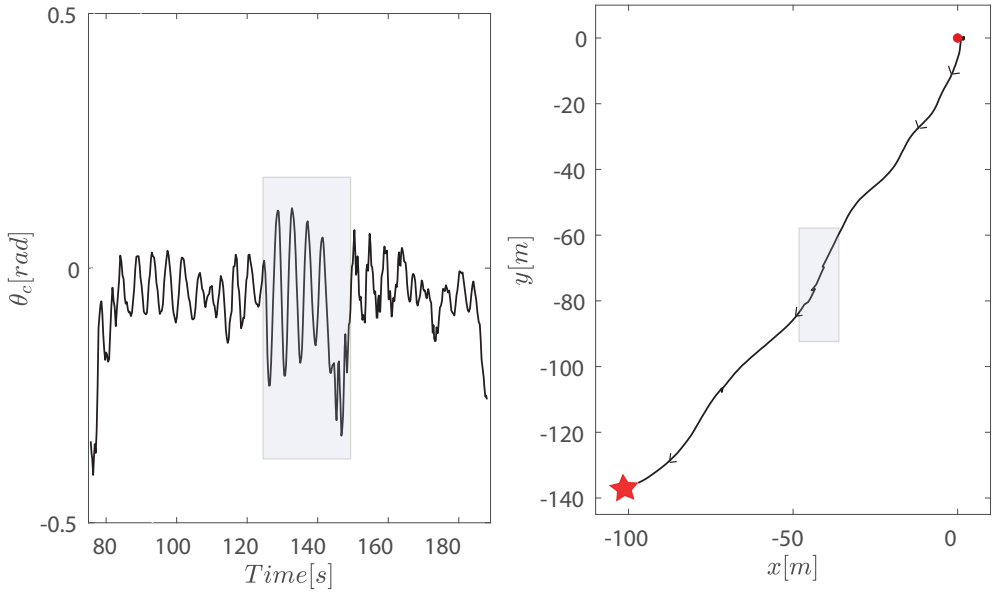
(c) Error in yaw angle (E_{ψ_B}).



(d) Error in position ($E_{|r|}$).

Figure 5.18: Control inputs and outputs during the successful navigation control test, low tailwind. The highlighted areas A and B represent selected instances where the yaw angle threshold was exceeded.

Figure 5.19 delineates the slung payload angle θ_c along with the planar trajectory during the successful test. Besides the first 5 seconds where the airship is being released, the payload oscillation is between -0.1 rad $\leq \theta_c \leq 0.1$ rad which is acceptable. In Fig. 5.19a, the swinging of the payload increases in the highlighted region although not by much. This increase is attributed to the gust and its effect can also be observed on the $X - Y$ trajectory highlighted in Fig. 5.19b.



(a) Payload cable angle (θ_c)

(b) $X - Y$ trajectory.

Figure 5.19: Payload oscillation during the successful navigation control test. The highlighted region illustrates the influence of gusts on both the payload's angle and the planar trajectory.

Despite the limited scope of the experimental tests caused by adverse weather conditions and time constraints, the overall performance of the controller can still be evaluated using the recorded data. Overall, the navigation controller operated as intended, effectively responded to inputs and attempted to correct errors while adhering to the upper and lower bounds of the thrusters. It is important to highlight that these were the very first outdoors experiments for this multibody blimp-gondola-payload system. In other words, the controller was purely tuned via simulations yet it was able to produce acceptable outcome in experimental tests. It is also clear from the experiments that the motors are not pow-

erful enough to counter gust speeds that are over $20km/h$, a finding that is corroborated by the simulations presented in Section 5.3.1 on page 104. Attempts to test the altitude controller were unsuccessful due to unfavourable weather conditions and the requirement for the blimp to be in neutral buoyancy, approximately 11 metres in altitude. At lower altitudes, the thrusters were not able to generate sufficient downward force to counteract the buoyant lift, making payload delivery tests unattainable under these circumstances.
Although the rules for FLC were based on PD logic, the disadvantages of PD control, such as chattering effect, do not appear to be significant in the conducted tests.
Furthermore, it is worthwhile to briefly discuss the stability of the proposed control method. Although proving the stability of a fuzzy logic controller for all state space can be challenging, a more manageable form of stability, in the sense of Lyapunov, can be demonstrated by conducting a series of simulations and identifying a Lyapunov function candidate that satisfies the conditions of the Lyapunov's theorem. Data collected from these simulations would establish stability for a subset of the state space. A more comprehensive investigation into the stability aspects of the proposed fuzzy logic controllers should be performed.

5.4 Contributions

Fuzzy logic controllers were developed to navigate an uninhabited blimp carrying a slung payload while effectively minimizing the payload's swinging. The proposed control method addresses a gap in the literature, which lacks experimental studies on airships with slung payloads using fuzzy logic control. With a reliable dynamic model of the multibody blimp-gondola-slung-payload, FLCs were tuned through numerical simulations and could operate successfully with no experimental training. The navigation controller was implemented on onboard hardware and successfully tested in an outdoor environment, demonstrating the practicality of the proposed approach.

Chapter 6

Thesis Contributions and Future Work

The key contributions of this research are:

1. Developed a dynamic model for the airship multibody system with holonomic and nonholonomic constraints. Moreover, a numerical investigation was carried out to estimate the aerodynamic coefficients of the airship prototype using CFD techniques with turbulence models.

2. Developed a system identification method that accurately estimates unknown parameters in the mass matrix of light-than-air vehicles, such as added mass and inertia. The proposed method is applicable with or without control actions, provided that the force vector generated by the controller is measured in real time. Consequently, this system identification approach can be applied to a range of vehicles, including airships and underwater vehicles, when parameters in the mass matrix are unknown.

3. Developed fuzzy logic controllers for the uninhabited multibody system to achieve planar flight, stabilizing the payload oscillations under windy conditions.

4. The airship slung load cargo transfer was validated by carrying out experimental tests on a prototype of the uninhabited airship. The planar flight controller was implemented on the onboard hardware and successfully tested outdoors, demonstrating feasibility.

The Udwadia-Kalaba method was applied to model the dynamics of the multibody blimp-gondola-slung-payload system, offering an efficient way to meet both holonomic and nonholonomic constraints. This approach expresses constraint forces directly through generalized coordinates and velocities, without needing auxiliary variables. Three constraints were established: position and orientation constraints between the airship and gondola, and a length constraint between the gondola and payload. The implementation encountered two main challenges: first, the actuation-dependent nature of gondola constraints, and second, the piecewise continuous rail with a curved segment. The first challenge was resolved by incorporating the gondola's travel distance using encoder measurements. To address the curved rail constraint, the blimp-gondola position vector was redefined by introducing a rail angle which is a function of the distance travelled along the rail. This approach effectively addressed the actuator-dependant constraints associated with the multibody system and constitutes the first contribution of the research.

While this chapter discussed most of the holonomic and nonholonomic constraints associated with the blimp-gondola-slung-payload multibody system, one constraint remained unaddressed: the inequality constraint of the gondola's rail. In other words, there are no constraints embedded in the dynamic model that defines the rail's starting and ending points. Future work may be invested to enforce the inequality constraint using diffeomorphism approach.

Second, a novel approach was developed for identifying parameters in multibody systems through optimization. In this case, the system identification problem was formulated as a direct dynamics problem and subsequently solved using semi-definite programming (SDP) with equality and inequality constraints. This method allows for the identification of unknown parameters in the mass matrix of a multibody system. More specifically, equality constraints were introduced to minimize discrepancies between the accelerations predicted by the model and those obtained experimentally. By taking advantage of the structure of the equations of motion and iteratively solving pseudo-inverse terms (namely, the matrix $\mathbf{K}$), parameters such as the moment of inertia, added inertia, and added mass of a multibody lighter-than-air (LTA) vehicle can be accurately identified using SDP.

One of the primary challenges encountered in multibody system identification lies in treat-

ing dynamic coupling. Traditionally, this is addressed by decoupling the system or constraining its motion, which are often cumbersome. The proposed method conveniently accommodates the dynamical constraints of the multibody system without requiring any simplifications or decoupling. As a result, it enables system identification to be conducted across a wider range of experimental scenarios. This constitutes the second contribution of the research.

The aerodynamic characteristics of the blimp-gondola system were analyzed using Reynolds-Averaged Navier-Stokes (RANS) equations in conjunction with the Spalart-Allmaras turbulence model. Specifically, aerodynamic coefficients were calculated for five different angles of attack and side-slip angles within the range of $-5^\circ \leq \alpha, \beta \leq 5^\circ$. This selection of angles is intended to partially cover the operational conditions of the multibody system. The resulting aerodynamic coefficients offer essential insights for optimizing future thruster designs, ensuring that the propulsion system operates efficiently and meets performance requirements under the intended environmental conditions. Moreover, the chapter highlights the distinctions among the empirical, semi-empirical, and numerical aerodynamic models. It also provides practical guidance for selecting the appropriate CFD technique based on specific research objectives.

Finally, the last contribution of this thesis is the development and implementation of Fuzzy logic controllers (FLCs) to navigate an unmanned blimp with a slung payload, minimizing payload swing during flight. This control method fills a gap in the literature, where experimental studies on airships with slung payloads using fuzzy logic are scarce. Using a dynamic model of the blimp-gondola-slung-payload system, FLCs were refined through numerical simulations, then implemented on onboard hardware and successfully tested outdoors, demonstrating feasibility.

The experiments highlighted areas for future improvements, including integrating more powerful motors and using weather stations for precise gust measurement. Due to processing limitations, the controller's operation was restricted to 5 Hz. Potential solutions include using a pre-compiled language, embedding the control code directly into the flight controller's firmware, or upgrading to a more powerful controller, such as the Pixhawk 6X

with RPI or CUAV V5+. These alternatives were recommended as better available options at the time this thesis was written.

Bibliography

[1] Philip V Hunt. *Advanced Airship Technologies and Design Approaches*. American Institute of Aeronautics and Astronautics, Incorporated, 2015.

[2] Lennart AT Ege and Kenneth Munson. *Balloons and Airships, 1783-1973: Editor of the English Edition Kenneth Munson*. Blandford Press, 1973.

[3] Donald E Ryan Jr. The airship's potential for intertheater and intratheater airlift. Technical report, AIR Univ Maxwell AFB AL School of Advanced Airpower Studies, 1992.

[4] Viktor Z. Shkoldychenko Vilen V. Azatyan, Timur R. Timerbulatov. Gas compound used to prevent inflammation and explosion of hydrogen-air mixtures, RU. Patent 2441685C2, Feb. 2012.

[5] Barry Prentice. Remember the hindenburg? forget the hindenburg! *Financial Post*, 2020.

[6] Boeing Commercial Airplanes. World air cargo forecast 2018–2037. In *Technical Report*. Seattle, WA: The Boeing World Air Cargo Forecast Team, Boeing Commercial, 2018.

[7] Carl-Oscar Lawaczeck. Looking back at the past decade. *OceanSky Aviation*, 2020.

[8] Ambrose Evans-Pritchard. How airships could provide the future of green transport. *The Telegraph*, 2020.

[9] Brian Dunn. Quebec defends $30 million flying whales investment. *Skies Magazine*, 2020.

[10] Wolverine. *Solar Ship - Smart Remote*. https://www.solarship.com/wolverine.html.

[11] Caracal. *Solar Ship - Smart Remote*. https://www.solarship.com/caracal.html.

[12] Lin Liao and Igor Pasternak. A review of airship structural research and development. *Progress in Aerospace Sciences*, 45(4-5):83–96, 2009.

[13] Jinjun Rao, Zhenbang Gong, Jun Luo, and Shaorong Xie. Unmanned airships for emergency management. In *IEEE International Safety, Security and Rescue Robotics Workshop, 2005*, pages 125–130. IEEE, 2005.

[14] Ahmed Ghanmi and Abderrahmane Sokri. Airships for military logistics heavy lift: A performance assessment for northern operation applications. *Defence R&D Canada—CORA, Technical Memorandum*, 63, 2010.

[15] Vinay Singh and Eric Lanteigne. Design and shape optimization of unmanned, semi-rigid airship for rapid descent using hybrid genetic algorithm. In *2019 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 1099–1107. IEEE, 2019.

[16] Gilmar Tuta Navajas. Modelling and pitch control of a re-configurable unmanned airship. Master's thesis, Université d'Ottawa/University of Ottawa, 2021.

[17] Eric Lanteigne and Joshua O'Reilly. Multibody dynamic modeling and control of an unmanned aerial vehicle under non-holonomic constraints. In *2020 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 316–321. IEEE, 2020.

[18] Office of Assistant Secretary of Defense (Research and United States. Engineering). Hybrid airships: Operational concepts. *Homeland Security Digital Library*, Jan 2013. https://www.hsdl.org/?view&did=728732.

[19] I Neimark Ju and NA Fufaev. Dynamics of nonholonomic systems (translations of mathematical monographs vol 33)(providence, ri: American mathematical society). 1972.

[20] Thomas R Kane and David A Levinson. *Dynamics, Theory and Applications*. McGraw Hill, 1985.

[21] Rajib Shekhar Pal. Modelling of helicopter underslung dynamics using kane's method. *IFAC-PapersOnLine*, 53(1):536–542, 2020.

[22] Alexander Schock and Robert Langlois. Spatial simulation of shipboard operations for skid-equipped rotary-wing aircraft.

[23] L Sandino, Manuel Bejar, and Anibal Ollero. Tutorial for the application of kane's method to model a small-size helicopter. In *Proceedings of the 1st Workshop on Research, Development and Education on Unmanned Aerial Systems*, pages 162–173, 2011.

[24] Azmin Sham Rambely, Norhafiza Ab Halim, and Rokiah Rozita Ahmad. A numerical comparison of lagrange and kane's methods of an arm segment. In *International Journal of Modern Physics: Conference Series*, volume 9, pages 68–75. World Scientific, 2012.

[25] Jonathan M Cameron and Wayne J Book. Modeling mechanisms with nonholonomic joints using the boltzmann-hamel equations. *The International Journal of Robotics Research*, 16(1):47–59, 1997.

[26] Firdaus E Udwadia and Robert E Kalaba. *Analytical Dynamics*. 1996.

[27] R. Jones and D. H. Williams. The stability of airship. Technical Report ARC R & M 751, June 1921.

[28] Horace Lamb. The inertia coefficients of an ellipsoid moving in fluid. *Advisory Committee for Aeronautics, Reports and Memoranda No. 623, London, UK*, 1918.

[29] R. Jones. Acceleration derivatives in the case of a body moving in an ideal fluid. Technical Report ARC R & M 748, April 1921.

[30] Peter G Thomasson. Equations of motion of a vehicle in a moving fluid. *Journal of Aircraft*, 37(4):630–639, 2000.

[31] Fossen Thor. Guidance and control of ocean vehicles, 1994.

[32] Horace Lamb. Hydrodynamics, ed. 6-th. *NY Dover Publ*, 194(5), 1945.

[33] DJ Lewis, JM Lipscombe, and PG Thomasson. The simulation of remotely operated underwater vehicles. In *ROV'84*, pages 245–252, 1984.

[34] MV Cook. The linearised small perturbation equations of motion for an airship. Technical report, 1990.

[35] K Nippress and S Gomes. Estimation of the flight dynamic characteristics of the yez-2a. In *8th Lighter-Than-Air Technology Conference*, page 3173, 1989.

[36] Sérgio BV Gomes. *An Investigation of the Flight Dynamics of Airships with Application to the YEZ-2A*. PhD thesis, Cranfield Institute of Technology, 1990.

[37] Frederick H Imlay. The complete expressions for added mass of a rigid body moving in an ideal fluid. Technical report, David Taylor Model Basin, Washington DC, 1961.

[38] John Nicholas Newman. *Marine Hydrodynamics*. The MIT Press, 1977.

[39] Leonardo Solaque and Simon Lacroix. *Airship Control*, pages 147–188. Springer Berlin Heidelberg, Berlin, Heidelberg, 2007.

[40] E. Hygounenc and P. Soueres. Lateral path following gps-based control of a small-size unmanned blimp. In *2003 IEEE International Conference on Robotics and Automation (Cat. No.03CH37422)*, volume 1, pages 540–545 vol.1, 2003.

[41] Peter Funk, Thorsten Lutz, and Siegfried Wagner. Experimental investigations on hull-fin interferences of the lotte airship. *Aerospace Science and Technology*, 7(8):603–610, 2003.

[42] Eric Lanteigne, Ahmad Alsayed, Dominic Robillard, and Steven G Recoskie. Modeling and control of an unmanned airship with sliding ballast. *Journal of Intelligent & Robotic Systems*, 88(2-4):285–297, 2017.

[43] Ali Mansur and Eric Lanteigne. Pitch tracking for an airship with moving gondola using backstepping control. In *2020 6th International Conference on Mechatronics and Robotics Engineering (ICMRE)*, pages 209–215. IEEE, 2020.

[44] E. C. de Paiva, S. S. Bueno, S. B. V. Gomes, J. J. G. Ramos, and M. Bergerman. A control system development environment for aurora's semi-autonomous robotic airship. In *Proceedings 1999 IEEE International Conference on Robotics and Automation (Cat. No.99CH36288C)*, volume 3, pages 2328–2335 vol.3, 1999.

[45] Shiqian Liu, Yuanjun Sang, and Hongbin Jin. Robust model predictive control for stratospheric airships using lpv design. *Control Engineering Practice*, 81:231–243, 2018.

[46] S. B. V. Gomes and J. G. Ramos. Airship dynamic modeling for autonomous operation. In *Proceedings. 1998 IEEE International Conference on Robotics and Automation (Cat. No.98CH36146)*, volume 4, pages 3462–3467 vol.4, 1998.

[47] Emmanuel Hygounenc, Il-Kyun Jung, Philippe Soueres, and Simon Lacroix. The autonomous blimp project of laas-cnrs: Achievements in flight control and terrain mapping. *The International Journal of Robotics Research*, 23(4-5):473–511, 2004.

[48] Luigi S Cicolani. *Equations of motion of slung-load systems, including multilift systems*, volume 3280. NASA, 1992.

[49] Sean Fielding and Meyer Nahon. Input shaped trajectory generation and controller design for a quadrotor-slung load system. In *2019 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 162–170. IEEE, 2019.

[50] C. de Crousaz, F. Farshidian, M. Neunert, and J. Buchli. Unified motion control for dynamic quadrotor maneuvers demonstrated on slung load and rotor failure tasks.

In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2223–2229, 2015.

[51] L Sciavicco, B Siciliano, L Villani, and G Oriolo. Robotics: Modelling, planning and control, ser. advanced textbooks in control and signal processing, 2011.

[52] Koushil Sreenath, Nathan Michael, and Vijay Kumar. Trajectory generation and control of a quadrotor with a cable-suspended load-a differentially-flat hybrid system. In *2013 IEEE International Conference on Robotics and Automation*, pages 4888–4895. IEEE, 2013.

[53] Morten Bisgaard, Jan Dimon Bendtsen, and Anders la Cour-Harbo. Modeling of generic slung load system. *Journal of Guidance, Control, and Dynamics*, 32(2):573–585, 2009.

[54] P. J. Cruz, M. Oishi, and R. Fierro. Lift of a cable-suspended load by a quadrotor: A hybrid system approach. In *2015 American Control Conference (ACC)*, pages 1887–1892, 2015.

[55] Sarah Tang, Valentin Wüest, and Vijay Kumar. Aggressive flight with suspended payloads using vision-based control. *IEEE Robotics and Automation Letters*, 3(2):1152–1159, 2018.

[56] Philipp Foehn, Davide Falanga, Naveen Kuppuswamy, Russ Tedrake, and Davide Scaramuzza. Fast trajectory optimization for agile quadrotor maneuvers with a cable-suspended payload. 2017.

[57] Kristian Klausen, Thor I Fossen, and Tor Arne Johansen. Nonlinear control of a multirotor uav with suspended load. In *2015 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 176–184. IEEE, 2015.

[58] Robert Mahony, Vijay Kumar, and Peter Corke. Multirotor aerial vehicles: Modeling, estimation, and control of quadrotor. *IEEE Robotics and Automation Magazine*, 19(3):20–32, 2012.

[59] Olav Egeland and Jan Tommy Gravdahl. *Modeling and Simulation for Automatic Control*, volume 76. Marine Cybernetics Trondheim, Norway, 2002.

[60] Fida Ben Abdallah, Naoufel Azouz, Lotfi Beji, and Azgal Abichou. Modeling of a heavy-lift airship carrying a payload by a cable-driven parallel manipulator. *International Journal of Advanced Robotic Systems*, 16(4):1729881419861769, 2019.

[61] Fida Ben Abdallah, Salim Hima, Naoufel Azouz, Lotfi Beji, and Azgal Abichou. Modeling and control of an airship-mounted crane for freight transportation. *IFAC-PapersOnLine*, 51(9):452–457, 2018.

[62] Joachim Baumgarte. Stabilization of constraints and integrals of motion in dynamical systems. *Computer Methods in Applied Mechanics and Engineering*, 1(1):1–16, 1972.

[63] Morten Bisgaard. *Modeling, Estimation, and Control of Helicopter Slung Load System*. Department of Control Engineering, Aalborg University, 2008.

[64] Shih-Tin Lin and Jiann-Nan Huang. Stabilization of baumgarte's method using the runge-kutta approach. *J. Mech. Des.*, 124(4):633–641, 2002.

[65] Quentin Leboutet, Julien Roux, Alexandre Janot, Julio Rogelio Guadarrama-Olvera, and Gordon Cheng. Inertial parameter identification in robotics: A survey. *Applied Sciences*, 11(9):4303, 2021.

[66] Bruno Siciliano, Oussama Khatib, and Torsten Kröger. *Springer Handbook of Robotics*, volume 200. Springer, 2008.

[67] Sabine Van Huffel and Joos Vandewalle. *The Total Least Squares Problem: Computational Aspects and Analysis*. SIAM, 1991.

[68] Khaled S Hatamleh, Ou Ma, and Robert Paz. A uav model parameter identification method: A simulation study. *International Journal of Information Acquisition*, 6(04):225–238, 2009.

[69] Sangheon Lee, Wonmo Chung, and Hungsun Son. Online parameter identification framework for a multirotor uav: Application to an arm stretchable morphing multirotor. *Mechanical Systems and Signal Processing*, 166:108468, 2022.

[70] Wei Wei, Kelly Cohen, and Mark B Tischler. System identification and controller optimization of a quadrotor uav. In *Proceedings of the AHS International's 71st Annual Forum and Technology Display*, 2015.

[71] Cristóvão D. Sousa and Rui Cortesão. Inertia tensor properties in robot dynamics identification: A linear matrix inequality approach. *IEEE/ASME Transactions on Mechatronics*, 24(1):406–411, 2019.

[72] Russ Tedrake and the Drake Development Team. Drake: Model-based design and verification for robotics, 2019.

[73] MOSEK ApS. Mosek optimization suite, 2019.

[74] Louis Bryant Tuckerman. Notes on aerodynamic forces on airship hulls. 1923.

[75] Max M Munk. The aerodynamic forces on airship hulls. Technical Report NACA TR-184, Springfield, VA, January 1924.

[76] Ralph H Upson and WA Klikoff. *Application of Practical Hydrodynamics to Airship Design*. 1931.

[77] H Julian Allen and Edward W Perkins. *Characteristics of Flow Over Inclined Bodies of Revolution*. National Advisory Committee for Aeronautics, 1951.

[78] Edward J Hopkins. A semi-empirical method for calculating the pitching moment of bodies of revolution at low mach numbers. 1951.

[79] SP Jones and JD DeLaurier. Aerodynamic estimation techniques for aerostats and airships. *Journal of Aircraft*, 20(2):120–126, 1983.

[80] Yuwen Li and Meyer Nahon. Modeling and simulation of airship dynamics. *Journal of Guidance, Control, and Dynamics*, 30(6):1691–1700, 2007.

[81] J DeLaurier and D Schenck. Airship dynamic stability. In *3rd Lighter-Than-Air Systems Technology Conference*, page 1591, 1979.

[82] J Evans and J DeLaurier. The shenandoah flies again-a computer simulation. In *4th Lighter-Than-Air Conference*, page 1325, 1981.

[83] J DeLaurier and K Hui. Airship survivability in atmospheric turbulence. In *4th Lighter-Than-Air Conference*, page 1323, 1981.

[84] Robert T Jones. Effects of sweepback on boundary layer and separation. 1947.

[85] H Julian Allen and Edward W Perkins. A study of effects of viscosity on flow over slender inclined bodies of revolution. 1951.

[86] GN Ward. Supersonic flow past slender pointed bodies. *The Quarterly Journal of Mechanics and Applied Mathematics*, 2(1):75–97, 1949.

[87] A.B. Wardlaw. High-angle-of-attack missile aerodynamics. *Lecture Series*, 98(5-1):(5–53), 1979.

[88] Joseph Mueller, Michael Paluszek, and Yiyuan Zhao. Development of an aerodynamic model and control law design for a high altitude airship. In *AIAA 3rd "Unmanned Unlimited" Technical Conference, Workshop and Exhibit*, page 6479, 2004.

[89] Steven Recoskie. *Autonomous Hybrid Powered Long Ranged Airship for Surveillance and Guidance*. PhD thesis, Université d'Ottawa/University of Ottawa, 2014.

[90] Theodor Von Karman. Calculation of pressure distribution on airship hulls. Technical report, NATIONAL ADVISORY COMMITTEE FOR AERONAUTICS WASHINGTON DC, 1930.

[91] Hugh B Freeman and John B Wheatley. *Pressure-Distribution Measurements on the Hull and Fins of a 1/40-Scale Model of the US Airship "Akron"*. US Government Printing Office, 1932.

[92] Xiao-Liang Wang, Gong-Yi Fu, Deng-Ping Duan, and Xue-Xiong Shan. Experimental investigations on aerodynamic characteristics of the zhiyuan-1 airship. *Journal of Aircraft*, 47(4):1463–1468, 2010.

[93] K Wong, J DeLaurier, and L Zhiyung. An application of source-panel and vortex methods for aerodynamic solutions of airship configurations. In *6th Lighter-Than-Air Systems Conference*, page 874, 1985.

[94] T Lutz, P Funk, A Jakobi, and S Wagner. Aerodynamic investigations on inclined airship bodies. In *Proceedings of the International Airship Convention and Exhibition*, pages 26–28, 1998.

[95] Kamal El Omari, Éric Schall, Bruno Koobus, and Alain Dervieux. *Numerical Investigations of Turbulent Flow Past a Generic Airship*. PhD thesis, INRIA, 2006.

[96] Vitaly Voloshin, Yong K Chen, and Rajnish K Calay. A comparison of turbulence models in airship steady-state cfd simulations. *arXiv Preprint arXiv:1210.2970*, 2012.

[97] Kevin Shields. Cfd applications in airship design. Master's thesis, Graduate Theses, Dissertations, and Problem Reports. 2185. West Virginia University, 2010.

[98] Sawan Suman, Sunil Lakshmipathy, and Rajkumar S Pant. Evaluation of assumed-transition-point criterion in context of reynolds-averaged simulations around lighter-than-air vehicles. *Journal of Aircraft*, 50(2):450–456, 2013.

[99] Akshay A Kanoria, K Panchal, Rocky Dongre, and Murali Damodaran. Computational modelling of aerodynamic characteristics of airships in arbitrary motion. In *22nd AIAA Lighter-Than-Air Systems Technology Conference*, page 3230, 2015.

[100] Jefferson L Mendonça Junior, Jonatas S Santos, Maurício A Morales, Luiz C Goes, Stojan Stevanovic, and Rodrigo A Santana. Airship aerodynamic coefficients estimation based on computational method for preliminary design. In *AIAA Aviation 2019 Forum*, page 2982, 2019.

[101] Sighard F Hoerner. Fluid-dynamic drag: Theoretical, experimental, and statistical information. *Copyright by: SF Hoerner Fluid Dynamics, Vancouver, Printed in the USA, Card Number 64-19666*, 1965.

[102] Arthur D Young. The calculation of the total and skin friction drags of bodies of revolution at zero incidence. Technical report, HM Stationery Office, 1939.

[103] Gabriel Alexander Khoury. *Airship Technology*, volume 10. Cambridge University Press, 2012.

[104] DB Bailey. Patrol airship concept evaluation (pace), naval air development center, report no. nadc-85019-60. Technical report, 62241N/F41-411/DH814, USA, 1985.

[105] Fengjian Jiang, José P Gallardo, and Helge I Andersson. The laminar wake behind a 6:1 prolate spheroid at 45 degree incidence angle. *Physics of Fluids*, 26(11):113602, 2014.

[106] J. Richard Shanebrook and William J. Sumner. Entrainment theory for axisymmetric, turbulent, incompressible boundary layers (entrainment theory for incompressible turbulent boundary layer velocity and drag on bodies of revolution employed in fuselage, submersible and cowlings for propulsion design). *Journal of Hydronautics*, 4(4):159–160, 1970.

[107] Jefferson L. Mendonça Junior, Jonatas S. Santos, Maurício A. Morales, Luiz C. Goes, Stojan Stevanovic, and Rodrigo A. Santana. *Airship Aerodynamic Coefficients Estimation Based on Computational Method for Preliminary Design.*

[108] Pijush K. Kundu, Ira M. Cohen, and David R. Dowling. Chapter 14 - aerodynamics. In Pijush K. Kundu, Ira M. Cohen, and David R. Dowling, editors, *Fluid Mechanics (Sixth Edition)*, pages 773–817. Academic Press, Boston, sixth edition edition, 2016.

[109] Nikolaos D. Katopodes. Chapter 8 - turbulent flow. In Nikolaos D. Katopodes, editor, *Free-Surface Flow*, pages 566–650. Butterworth-Heinemann, 2019.

[110] François G Schmitt. About boussinesq's turbulent viscosity hypothesis: Historical remarks and a direct evaluation of its validity. *Comptes Rendus Mécanique*, 335(9-10):617–627, 2007.

[111] M Carri, M Biava, R Steijl, GN Barakos, D Stewart, et al. Computational fluid dynamics challenges for hybrid air vehicle applications. *Progress in Flight Physics–Volume 9*, 9:43–80, 2017.

[112] Lafayette Taylor, Ramesh Pankajakshan, WR Briley, and David Whitfield. Scalable parallel implicit algorithm for advanced turbulence closures. In *43rd AIAA Aerospace Sciences Meeting and Exhibit*, page 876, 2005.

[113] Shin Hyung Rhee and Takanori Hino. Numerical simulation of unsteady turbulent flow around maneuvering prolate spheroid. *AIAA Journal*, 40(10):2017–2026, 2002.

[114] George S Constantinescu, Hugo Pasinato, You-Qin Wang, James R Forsythe, and Kyle D Squires. Numerical investigation of flow past a prolate spheroid. *J. Fluids Eng.*, 124(4):904–910, 2002.

[115] Christer Fureby and Andreas Karlsson. Les of the flow past a 6:1 prolate spheroid. In *47th AIAA Aerospace Sciences Meeting Including The New Horizons Forum and Aerospace Exposition*, page 1616, 2009.

[116] Nathan Scott and Earl Duque. Unsteady reynolds-averaged navier-stokes predictions of the flow around a prolate spheroid. In *42nd AIAA Aerospace Sciences Meeting and Exhibit*, page 55, 2004.

[117] Niklas Alin, Christer Fureby, S Svennberg, William Sandberg, R Ramamurti, N Wikstrom, Rikard Bensow, and Tobias Persson. 3d unsteady computations for submarine-like bodies. In *43rd AIAA Aerospace Sciences Meeting and Exhibit*, page 1104, 2005.

[118] N Wikström, U Svennberg, N Alin, and C Fureby. Large eddy simulation of the flow around an inclined prolate spheroid. *Journal of Turbulence*, 5:029–029, 2004.

[119] Rupesh B Kotapati-Apparao, Kyle D Squires, and James R Forsythe. Prediction of a prolate spheroid undergoing a pitchup maneuver. In *AIAA Papper 2003-0269 41 St Aerospace Sciences Meeting and Exhibit*. Citeseer, 2003.

[120] George K El Khoury, Helge I Andersson, and Bjørnar Pettersen. Crossflow past a prolate spheroid at reynolds number of 10000. *Journal of Fluid Mechanics*, 659:365, 2010.

[121] Marian Zastawny, George Mallouppas, Fan Zhao, and Berend van Wachem. Derivation of drag and lift force and torque coefficients for non-spherical particles in flows. *International Journal of Multiphase Flow*, 39:227–239, 2012.

[122] Fengjian Jiang, José P Gallardo, Helge I Andersson, and Zhiguo Zhang. The transitional wake behind an inclined prolate spheroid. *Physics of Fluids*, 27(9):093602, 2015.

[123] Helge I Andersson, Fengjian Jiang, and Valery L Okulov. Instabilities in the wake of an inclined prolate spheroid. In *Computational Modelling of Bifurcations and Instabilities in Fluid Dynamics*, pages 311–352. Springer, 2019.

[124] Håkon Strandenes, Fengjian Jiang, Bjørnar Pettersen, and Helge I Andersson. Near-wake of an inclined 6:1 spheroid at reynolds number 4000. *AIAA Journal*, 57(4):1364–1372, 2019.

[125] Håkon Strandenes, Fengjian Jiang, Bjørnar Pettersen, and Helge I Andersson. Low-frequency oscillations in flow past an inclined prolate spheroid. *International Journal of Heat and Fluid Flow*, 78:108421, 2019.

[126] Taeyoung Han and VC Patel. Flow separation on a spheroid at incidence. *Journal of Fluid Mechanics*, 92(4):643–657, 1979.

[127] Osama Obeid, Ibraheem AlQadi, and Jaber AlMutairi. Investigation of asymmetric flow past a slender body at high angles of attack. *Theoretical and Computational Fluid Dynamics*, 33(5):481–508, 2019.

[128] M. D. Zeiger, D. P. Telionis, and P. P. Vlachos. Unsteady separated flows over three-dimensional slender bodies. *Progress in Aerospace Sciences*, 40(4):291–320, 2004.

[129] Russell M. Cummings, James R. Forsythe, Scott A. Morton, and Kyle D. Squires. Computational challenges in high angle of attack flow prediction. *Progress in Aerospace Sciences*, 39(5):369–384, 2003.

[130] M. J. Siclari and F. Marconi. Computation of navier-stokes solutions exhibiting asymmetric vortices. *AIAA Journal*, 29(1):32–42, 1991.

[131] Seungki Ahn. An experimental study of flow over a 6 to 1 prolate spheroid at incidence (ph.d. thesis). 1992.

[132] Siemens Digital Industries Software. Simcenter STAR-CCM+ documentation, version 2021.1. Siemens 2021. https://docs.sw.siemens.com/documentation/external/PL20200805113346338/en-US/userManual/userguide/html/index.html#page/STARCCMP%2FGUID-478651F1-9901-41B6-8193-3052E3CCE9DB.html%23wwID0EKTFSB.

[133] Philippe Spalart and Steven Allmaras. A one-equation turbulence model for aerodynamic flows. In *30th Aerospace Sciences Meeting and Exhibit*, page 439, 1992.

[134] Reuben Raz, Aviv Rosen, and Tuvia Ronen. Active aerodynamic stabilization of a helicopter/sling-load system. *Journal of Aircraft*, 26(9):822–828, 1989.

[135] So-Ryeok Oh, Ji-Chul Ryu, and Sunil K Agrawal. Dynamics and control of a helicopter carrying a payload using a cable-suspended robot. 2006.

[136] Keeryun Kang, JVR Prasad, and Eric Johnson. Active control of a uav helicopter with a slung load for precision airborne cargo delivery. *Unmanned Systems*, 4(03):213–226, 2016.

[137] Markus Bernard and Konstantin Kondak. Generic slung load transportation system using small size helicopters. In *2009 IEEE International Conference on Robotics and Automation*, pages 3258–3264. IEEE, 2009.

[138] Maria Eusebia Guerrero, DA Mercado, Rogelio Lozano, and CD García. Passivity based control for a quadrotor uav transporting a cable-suspended payload with minimum swing. In *2015 54th IEEE Conference on Decision and Control (CDC)*, pages 6718–6723. IEEE, 2015.

[139] Pedro O Pereira, Manuel Herzog, and Dimos V Dimarogonas. Slung load transportation with a single aerial vehicle and disturbance removal. In *2016 24th Mediterranean Conference on Control and Automation (MED)*, pages 671–676. IEEE, 2016.

[140] Sarah Tang and Vijay Kumar. Mixed integer quadratic program trajectory generation for a quadrotor with a cable-suspended payload. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2216–2222. IEEE, 2015.

[141] Koushil Sreenath, Taeyoung Lee, and Vijay Kumar. Geometric control and differential flatness of a quadrotor uav with a cable-suspended load. In *52nd IEEE Conference on Decision and Control*, pages 2269–2274. IEEE, 2013.

[142] Hanafy M Omar. Designing anti-swing fuzzy controller for helicopter slung-load system near hover by particle swarms. *Aerospace Science and Technology*, 29(1):223–234, 2013.

[143] James Kennedy. The particle swarm: Social adaptation of knowledge. In *Proceedings of 1997 IEEE International Conference on Evolutionary Computation (ICEC'97)*, pages 303–308. IEEE, 1997.

[144] J. Potter, W. Singhose, and M. Costelloy. Reducing swing of model helicopter sling load using input shaping. In *2011 9th IEEE International Conference on Control and Automation (ICCA)*, pages 348–353, 2011.

[145] N. S. Zúñiga, F. Muñoz, M. A. Márquez, E. S. Espinoza, and L. R. G. Carrillo. Load transportation using single and multiple quadrotor aerial vehicles with swing load attenuation. In *2018 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 269–278, 2018.

[146] Hassan K Khalil and Jessy W Grizzle. *Nonlinear Systems*, volume 3. Prentice Hall Upper Saddle River, NJ, 2002.

[147] Kristian Klausen, Thor I Fossen, and Tor Arne Johansen. Nonlinear control with swing damping of a multirotor uav with suspended load. *Journal of Intelligent & Robotic Systems*, 88(2):379–394, 2017.

[148] Morten Bisgaard, Anders la Cour-Harbo, and Jan Bendtsen. Swing damping for helicopter slung load systems using delayed feedback. In *AIAA Guidance, Navigation, and Control Conference*, page 5795, 2009.

[149] Muhammad Amir Tahir, Imran Mir, and Tauqeer Ul Islam. Control algorithms, kalman estimation, and near actual simulation for uavs: State of art perspective. *Drones*, 7(6):339, 2023.

[150] Farhad Goodarzi, Daewon Lee, and Taeyoung Lee. Geometric nonlinear pid control of a quadrotor uav on se (3). In *2013 European Control Conference (ECC)*, pages 3845–3850. IEEE, 2013.

[151] Hanafy M Omar and Saad MS Mukras. Developing geno-fuzzy controller for suppressing quadrotor slung-load oscillations. *Ain Shams Engineering Journal*, 14(8):102051, 2023.

[152] B. Yu, K. Gamagedara, S. Kim, T. Lee, and J. Suk. Geometric control and experimental validation for a quadrotor uav transporting a payload. In *2020 59th IEEE Conference on Decision and Control (CDC)*, pages 201–207, 2020.

[153] Laith M Alkurdi. Fuzzy control of a robotic blimp. Master's thesis, MS Thesis, University of Edinburgh, 2011.

[154] Rami Al-Jarrah, Radouane Ait Jellal, and Hubert Roth. Blimp based on embedded computer vision and fuzzy control for following ground vehicles. *IFAC Proceedings Volumes*, 46(29):7–12, 2013.

[155] Omar Rodríguez-Abreo, Juvenal Rodríguez-Reséndiz, A. García-Cerezo, and José R. García-Martínez. Fuzzy logic controller for uav with gains optimized via genetic algorithm. *Heliyon*, 10(4):e26363, 2024.

[156] Fakhreddine Karray, Fakhreddine O Karray, and Clarence W De Silva. *Soft Computing and Intelligent Systems Design: Theory, Tools, and Applications*. Pearson Education, 2004.

[157] Tyto Robotics Inc. Tyto robotics thrust stands and test equipment, 2024.

[158] Ebrahim H Mamdani and Sedrak Assilian. An experiment in linguistic synthesis with a fuzzy logic controller. *International Journal of Man-Machine Studies*, 7(1):1–13, 1975.

[159] Environment and Natural Resources of Canada. Hourly data report for september 30, 2024.

[160] L. Meirovitch and M. K. Kwak. State equations for a spacecraft with maneuvering flexible appendages in terms of quasi-coordinates. *Applied Mechanics Reviews*, 42(11S):S161–S170, 11 1989.

[161] Leonard Meirovitch. *Dynamics and Control of Structures*. John Wiley & Sons, 1990.

[162] Sergei Alekseevich Chaplygin. On the theory of motion of nonholonomic systems. the reducing-multiplier theorem. *Regular and Chaotic Dynamics*, 13(4):369–376, 2008.

[163] J. Piedboeuf. Kane's equations or jourdain's principle? In *Proceedings of 36th Midwest Symposium on Circuits and Systems*, pages 1471–1474 vol.2, 1993.

[164] Leonard Meirovitch. *Methods of Analytical Dynamics*. Courier Corporation, 2010.

[165] NA Fufaev. Theory of the motion of systems with rolling. *Journal of Applied Mathematics and Mechanics*, 49(1):49–53, 1985.

[166] Donald T. Greenwood. *Equations of Motion: Differential Approach*, page 217–288. Cambridge University Press, 2003.

[167] S. I. Sagatun and T. I. Fossen. Lagrangian formulation of underwater vehicles' dynamics. In *Conference Proceedings 1991 IEEE International Conference on Systems, Man, and Cybernetics*, pages 1029–1034 vol.2, 1991.

[168] Olvi L Mangasarian. Equivalence of the complementarity problem to a system of nonlinear equations. *SIAM Journal on Applied Mathematics*, 31(1):89–92, 1976.

[169] AL Eterovic and KJ Bathe. On the treatment of inequality constraints arising from contact conditions in finite element analysis. *Computers & Structures*, 40(2):203–209, 1991.

[170] R Rosenberg. *Analytical Dynamics*. Springer, 1977.

[171] TE Connectivity. *FX29 Compact Compression Load Cell*, 2022.

[172] Arduino. *Arduino Mega 2560*, 2022.

[173] Osama Ahmad Obeid. Computational study of wake-vortex asymmetry around slender body at high angles of attack. Master's thesis, King Abdulaziz University, 2017.

[174] Jan Jantzen. Design of fuzzy controllers. *Technical University of Denmark, Department of Automation, Bldg*, 326:362–367, 1998.

[175] Michio Sugeno. *Industrial Applications of Fuzzy Control*. Elsevier Science Inc., 1985.

APPENDICES

Appendix A

Dynamics and Kinematics

A.1 Newton-Euler (N-E)

For an origin point in a rotating frame, recall that the time derivative with respect to an inertial frame is related to the time derivative in the body-fixed frame by the following relationship

$$\frac{^E d}{dt}\vec{r} = \frac{^B d}{dt}\vec{r} + \vec{\omega} \times \vec{r} \tag{A.1}$$

$$\frac{^E d}{dt}\vec{\omega} = \frac{^B d}{dt}\vec{\omega} + \vec{\omega} \times \vec{\omega} = \frac{^B d}{dt}\vec{\omega} \tag{A.2}$$

where $\vec{r}$ and $\vec{\omega}$ are the position and angular velocity vectors, respectively. E and B superscripts refer to, respectively, earth (inertial frame) and body-fixed frames.

Figure A.1 demonstrates the position vectors $r\vec{_{CV}}$ and $r\vec{_{CG}}$ from inertial frames to CV and CG respectively. Position vector $r\vec{_{VG}}$ goes from the CV to CG of the body. The relation between these vectors and their time derivatives are given next. Note that the time derivatives are taken with respect to the inertial frame, unless otherwise mentioned, and will be referred to using an over-dot on top of an arrow $\dot{\vec{r}}$.

$$\begin{aligned} \vec{r}_{CG} &= \vec{r}_{CV} + \vec{r}_{VG} \\ \dot{\vec{r}}_{CG} &= \dot{\vec{r}}_{CV} + \dot{\vec{r}}_{VG} \end{aligned} \tag{A.3}$$

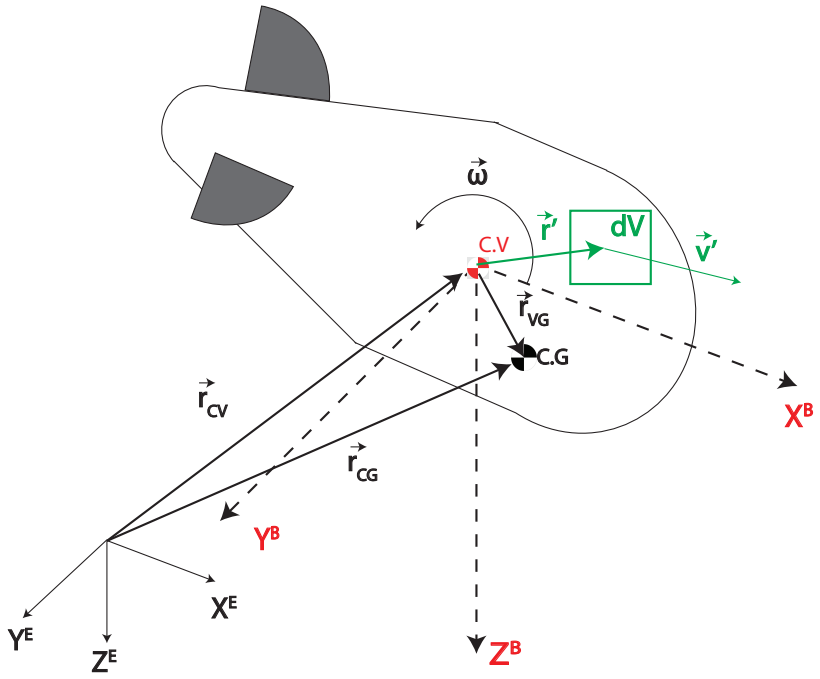


Figure A.1: Sketch illustrating the inertial frame (X^E, Y^E, Z^E) and the body frame at CV (X^B, Y^B, Z^B).

For single rigid body motions, $\vec{r}_{VG}$ is constant therefore $\frac{{}^B d}{dt}\vec{r}_{VG}$ cancels out due to rigid body assumption.

$$\begin{aligned}
\vec{V}_{CG} &= \vec{V}_{CV} + \vec{\omega} \times \vec{r}_{VG} \\
\dot{\vec{V}}_{CG} &= \dot{\vec{V}}_{CV} + \dot{\vec{\omega}} \times \vec{r}_{VG} + \vec{\omega} \times \dot{\vec{r}}_{VG} \\
\vec{a}_{CG} &= \frac{{}^B d}{dt}\vec{V}_{CV} + \vec{\omega} \times \vec{V}_{CV} + (\frac{{}^B d}{dt}\vec{\omega} + \cancelto{0}{\vec{\omega} \times \vec{\omega}}) \times \vec{r}_{VG} + \vec{\omega} \times (\vec{\omega} \times \vec{r}_{VG}) \\
\vec{a}_{CG} &= \vec{a}_{CV} \quad + \vec{\omega} \times \vec{V}_{CV} + \vec{\Omega} \times \vec{r}_{VG} \quad + \vec{\omega} \times (\vec{\omega} \times \vec{r}_{VG})
\end{aligned} \tag{A.4}$$

where $\vec{a}$ and $\vec{\Omega}$ are the translational and angular acceleration vectors, respectively. Substituting (A.4) in (2.1) yields the translational motion

$$m[\vec{a}_{CV} + \vec{\omega} \times \vec{V}_{CV} + \vec{\Omega}_{CV} \times \vec{r}_{VG} + \vec{\omega} \times (\vec{\omega} \times \vec{r}_{VG})] = F_{CV} \tag{A.5}$$

Note if the origin is chosen to be CG instead of CV, then $\vec{r}_{VG} = 0$ which simplifies equation

(A.5) to

$$m[\vec{a}_{CG} + \vec{\omega} \times \vec{V}_{CG}] = F_{CG} \tag{A.6}$$

On the other hand, the derivation of the angular motion starts by defining the angular momentum ($\vec{h}_{CV}$)

$$\vec{h}_{CV} = \int_{\mathbb{V}} \vec{r}\,' \times \vec{v}\,' \, \rho \, d\mathbb{V} \tag{A.7}$$

where $\vec{r}\,'$ is a vector going from CV to an arbitrary point on the body which has a local velocity $\vec{v}\,'$, ρ is the density of the body, $\mathbb{V}$ is the volume of the body, and $d\mathbb{V}$ is an infinitesimal volume. Differentiating the angular momentum with respect to time gives (A.8)

$$\dot{h}_{CV} = \int_{\mathbb{V}} \dot{\vec{r}}\,' \times \vec{v}\,' \, \rho \, d\mathbb{V} + \underbrace{\int_{\mathbb{V}} \vec{r}\,' \times \dot{\vec{v}}\,' \, \rho \, d\mathbb{V}}_{\vec{M}_{CV}\,(\text{moment vector})} \tag{A.8}$$

It is possible to redefine $\vec{v}\,'$ by examining Fig. A.1 such that $\vec{v}\,' = \dot{\vec{r}}\,' + \vec{V}_{CV}$. Rearranging gives (A.9)

$$\dot{\vec{r}}\,' = \vec{v}\,' - \vec{V}_{CV} \tag{A.9}$$

Substituting (A.9) in (A.8)

$$\dot{h}_{CV} = \vec{M}_{CV} + \int_{\mathbb{V}} (\vec{v}\,' - \vec{V}_{CV}) \times \vec{v}\,' \, \rho \, d\mathbb{V} \tag{A.10}$$

The term $\vec{v}\,' \times \vec{v}\,'$ is zero. It is possible to take $\vec{V}_{CV}$ out of the integral and substituting (A.9) again for $\vec{v}\,'$

$$\dot{\vec{h}}_{CV} = \vec{M}_{CV} - \vec{V}_{CV} \times \int_{\mathbb{V}} \vec{v}\,' \, \rho \, d\mathbb{V} + \cancelto{0}{\int_{\mathbb{V}} \vec{v}\,' \times \vec{v}\,' \, \rho \, d\mathbb{V}} \tag{A.11}$$

$$\dot{\vec{h}}_{CV} = \vec{M}_{CV} - \vec{V}_{CV} \times \int_{\mathbb{V}} (\dot{\vec{r}}\,' + \vec{V}_{CV}) \, \rho \, d\mathbb{V} \tag{A.12}$$

$$\dot{\vec{h}}_{CV} = \vec{M}_{CV} - \vec{V}_{CV} \times \int_{\mathbb{V}} \dot{\vec{r}}\,' \, \rho \, d\mathbb{V} \tag{A.13}$$

Recall that the position vector of an arbitrary point on the body to CG can be defined as

$$\begin{aligned} \vec{r}_{VG} &= \frac{1}{m} \int_{\mathbb{V}} \vec{r}\,' \, \rho \, d\mathbb{V} \\ m \, \vec{r}_{VG} &= \int_{\mathbb{V}} \vec{r}\,' \, \rho \, d\mathbb{V} \end{aligned} \tag{A.14}$$

In order to eliminate the integral term on the RHS of (A.13), equation A.14 is differentiated

$$m\ \dot{\vec{r}}_{VG} = \int_{\mathbb{V}} \dot{\vec{r}}'\ \rho\ d\mathbb{V} \tag{A.15}$$

$$m\ (\vec{\omega} \times \vec{r}_{VG}) = \int_{\mathbb{V}} \dot{\vec{r}}'\ \rho\ d\mathbb{V} \tag{A.16}$$

Substituting (A.16) in (A.13) yields

$$\dot{\vec{h}}_{CV} = \vec{M}_{CV} - \vec{V}_{CV} \times m(\omega \times \vec{r}_{VG}) \tag{A.17}$$

Equations (A.7) and (A.9) will be used again where

$$\begin{aligned}
\vec{h}_{CV} &= \int_{\mathbb{V}} \vec{r}\,' \times (\vec{V}_{CV} + \dot{\vec{r}}')\ \rho\ d\mathbb{V} \\
\vec{h}_{CV} &= \int_{\mathbb{V}} \vec{r}\,' \times (\vec{V}_{CV} + \vec{\omega} \times \vec{r}\,')\ \rho\ d\mathbb{V} \\
\vec{h}_{CV} &= \int_{\mathbb{V}} \vec{r}\,' \times \vec{V}_{CV}\ \rho\ d\mathbb{V} + \int_{\mathbb{V}} \vec{r}\,' \times (\vec{\omega} \times \vec{r}\,')\ \rho\ d\mathbb{V} \\
\vec{h}_{CV} &= \underbrace{(\int_{\mathbb{V}} \vec{r}\,'\ \rho\ d\mathbb{V})}_{(m\ \vec{r}_{VG})\ \text{from (A.14)}} \times \vec{V}_{CV} + \underbrace{\int_{\mathbb{V}} \vec{r}\,' \times (\vec{\omega} \times \vec{r}\,')\ \rho\ d\mathbb{V}}_{(I_{CV}\ \vec{\omega})\ \text{product of inertia tensor with angular velocity}}
\end{aligned} \tag{A.18}$$

$$\vec{h}_{CV} = (m\ \vec{r}_{VG}) \times \vec{V}_{CV} + I_{CV}\ \vec{\omega} \tag{A.19}$$

Now differentiating (A.19) while assuming mass and I_{CV} are constant with respect to time[1]

$$\dot{h}_{CV} = m(\vec{\omega} \times \vec{r}_{VG}) \times \vec{V}_{CV} + m\ \vec{r}_{VG} \times (\vec{a}^{B}_{CV}\ +\ \vec{\omega}\ \times \vec{V}_{CV}) + I_{CV}\ \vec{\Omega}^{B} + \vec{\omega} \times (I_{CV}\ \vec{\omega}) \tag{A.20}$$

One last manipulation can be used to simplify the angular momentum equation where

$$(\vec{\omega} \times \vec{r}_{VG}) \times \vec{V}_{CV} = -\vec{V}_{CV} \times (\vec{\omega} \times \vec{r}_{VG}) \tag{A.21}$$

Equating (A.17) to (A.20) and using (A.21) result in the following component form of Euler equation

$$I_{CV}\ \vec{\Omega}_{CV} + \vec{\omega} \times (I_{CV}\ \vec{\omega}) + m\ \vec{r}_{VG} \times (\vec{a}_{CV}\ +\ \vec{\omega} \times \vec{V}_{CV}) = \vec{M}_{CV} \tag{A.22}$$

[1]For rigid bodies, mass is assumed to be constant. On the other hand, the inertia tensor I_{CV} is only constant when using body-fixed frame of reference.

The inertia tensor I_{CV} is defined as

$$I_{CV} = \begin{bmatrix} I_x & -I_{xy} & -I_{xz} \\ -I_{yx} & I_y & -I_{yz} \\ -I_{zx} & -I_{zy} & I_z \end{bmatrix} \tag{A.23}$$

I_x, I_y, and I_z are the moments of inertia of X^B, Y^B, Z^B, respectively. Moments of inertia and their products $(I_{xy}, I_{xz}, I_{yx}, I_{yz}, I_{zx}, I_{zy})$ are defined at the end of Appendix A. While the N-E equations (A.5) and (A.22) can be represented in the inertial or body frame, the variable $(\vec{a}_{CV}, \vec{V}_{CV}, \vec{\Omega}_{CV}, \vec{\omega})$ are obtained with respect to the body frame.

A.2 Euler-Lagrange (EL)

The general form of the Lagrangian equation is

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\vec{q}}}\right) - \frac{\partial L}{\partial \vec{q}} = \vec{Q} \tag{A.24}$$

where Q is the vector of external forces and torque acting on the systems in the geenralized coordinats, L is the Lagrangian and it is defined as

$$L(\dot{\vec{q}}, \vec{q}) = T(\dot{\vec{q}}, \vec{q}) - U(\vec{q}) \tag{A.25}$$

where T and U are the kinetic and potential energy, respectively and $\vec{q}$ is the vector of generalized coordinates. As long as the generalized coordinates are used and the number generalized coordinates are equal to the number of DOF, Lagrange equation (A.24) is valid for any number of interconnected bodies. However, there are some nuisances that should be highlighted.

While it is often advantageous to construct the equations of motion in body-fixed frame[2], it is inconvenient to use body-fixed variables in the given form of Lagrange equation. Let $\vec{V}$ be the body-fixed velocity vector

$$\vec{V} = \{u \; v \; w \; p \; q \; r\} \tag{A.26}$$

[2]For reasons mentioned before, such as exploiting the symmetry of the body and the principal axes all of which yield a more compact EOM.

the integration of the body-fixed velocity vector ($\int_0^t \vec{V} dt$) does not give the position vector and has no clear physical interpretation. In other words, the time integral of the body-fixed velocity vector does not yield generalized coordinates. Since the Lagrangian is differentiated with respect to $\vec{q}$ and $\dot{\vec{q}}$ in Eq. (A.24), the translational and angular velocities in the body-fixed frame must be introduced as functions of $\vec{q}$ and $\dot{\vec{q}}$. Although it is possible to execute the latter, the resulting expression is intractable. A more convenient approach is applying Kirchoff's equations, a special case of Quasi Lagrange method. Kirchoff uses quasi-coordinates which leads to a much more manageable expression. Details about the derivation of Kirchoff's equations are described in [160]. The resulting Kirchoff's equations are presented next in vector form.

$$\begin{aligned} \frac{d}{dt}(\frac{\partial T}{\partial \vec{V}_1}) + \vec{V}_2 \times \frac{\partial T}{\partial \vec{V}_1} &= \vec{F} \\ \frac{d}{dt}(\frac{\partial T}{\partial \vec{V}_2}) + \vec{V}_2 \times \frac{\partial T}{\partial \vec{V}_2} + \vec{V}_1 \times \frac{\partial T}{\partial \vec{V}_1} &= \vec{M} \end{aligned} \tag{A.27}$$

where $\vec{V}_1 = [u\ v\ w]$ and $\vec{V}_2 = [p\ q\ r]$ are the linear and angular velocity in the body-fixed frame, respectively. It is worth noting that Kirchoff's equations do not account for potential energy (that is, gravitational forces are neglected). To account for gravitational forces, the Quasi-Lagrangian should be used, which gives the following equations [161]

$$\begin{aligned} \frac{d}{dt}(\frac{\partial L}{\partial \vec{V}_1}) + \vec{V}_2 \times \frac{\partial L}{\partial \vec{V}_1} - \mathbf{R_1}^T \frac{\partial L}{\partial \vec{q_1}} &= \vec{F} \\ \frac{d}{dt}(\frac{\partial L}{\partial \vec{V}_2}) + \vec{V}_2 \times \frac{\partial L}{\partial \vec{V}_2} + \vec{V}_1 \times \frac{\partial L}{\partial \vec{V}_1} - \mathbf{R_2}^T \frac{\partial L}{\partial \vec{q_2}} &= \vec{M} \end{aligned} \tag{A.28}$$

where $\vec{q_1} = [x\ y\ z]$, $\vec{q_2} = [\phi\ \theta\ \psi]$, $\mathbf{R_1}$ (Eq. A.30) and $\mathbf{R_2}$ (Eq. A.31) are the linear and angular velocity transformation matrices, respectively (included in Appendix A).

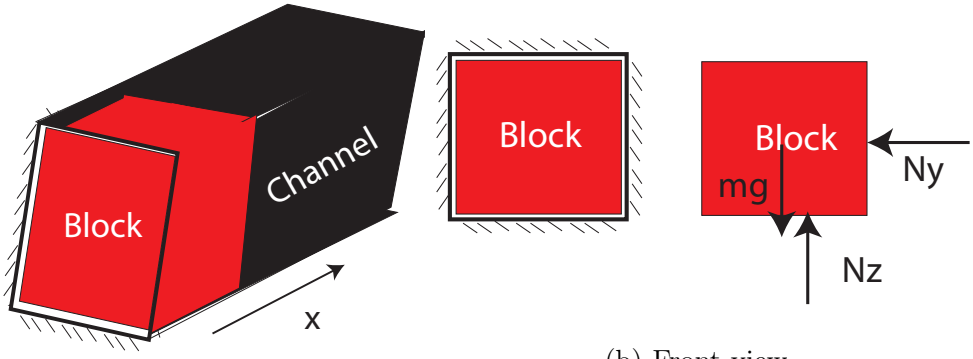
Moments of inertia and product of inertia Moments of inertia and their products ($I_{xy}, I_{xz}, I_{yx}, I_{yz}, I_{zx}, I_{zy}$) are defined as

$$\begin{aligned} I_x &= \int_{\mathbb{V}} (y^2 + z^2)\rho d\mathbb{V} \qquad & I_{xy} = I_{yx} &= \int_{\mathbb{V}} xy\rho d\mathbb{V} \\ I_y &= \int_{\mathbb{V}} (x^2 + z^2)\rho d\mathbb{V} \qquad & I_{yz} = I_{zy} &= \int_{\mathbb{V}} yz\rho d\mathbb{V} \\ I_z &= \int_{\mathbb{V}} (x^2 + y^2)\rho d\mathbb{V} \qquad & I_{xz} = I_{zx} &= \int_{\mathbb{V}} zx\rho d\mathbb{V} \end{aligned} \tag{A.29}$$

Explicit holonomic constraints in N-E formulation Figure A.2 depicts a block that moves inside a channel. The walls of the channel impose constraints on the motion of the block, reducing its DOF to one. Using the free body diagram in Fig. A.2b, and summing the forces in z direction yields

$$ma_z = -mg + N_z$$

where N_z is a constraint force, m is mass of block, a_z is the acceleration in $z - direction$, and g is gravitational acceleration. In order to obtain the EOM, further manipulations are required to eliminate the constraint force N_z, which appears in the EOM explicitly.



(a) Problem setup

(b) Front view.

Figure A.2: Toy problem demonstrating the explicit appearance of constraints in N-E applications

Transformation and rotation matrices Transformation and rotation matrices R_1 and R_2, respectively, are used to convert the linear and angular velocity from body to inertial coordinate.

$$\mathbf{R_1} = \begin{bmatrix} \cos(\theta)\cos(\psi) & \sin(\phi)\sin(\theta)\cos(\psi) - \cos(\phi)\sin(\psi) & \cos(\phi)\sin(\theta)\cos(\psi) + \sin(\phi)\sin(\psi) \\ \cos(\theta)\sin(\psi) & \sin(\phi)\sin(\theta)\sin(\psi) + \cos(\phi)\cos(\psi) & \cos(\phi)\sin(\theta)\sin(\psi) - \sin(\phi)\cos(\psi) \\ -\sin(\theta) & \sin(\phi)\cos(\theta) & \cos(\phi)\cos(\theta) \end{bmatrix} \tag{A.30}$$

$$\mathbf{R_2} = \begin{bmatrix} 1 & \sin(\phi)\tan(\theta) & \cos(\phi)\tan(\theta) \\ 0 & \cos(\phi) & -\sin(\phi) \\ 0 & \sin(\phi)\sec(\theta) & \cos(\phi)\sec(\theta) \end{bmatrix} \tag{A.31}$$

$$\mathbf{R_{1G\ curved}} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \tag{A.32}$$

where

$$\begin{aligned}
a_{11} &= \cos(\lambda)\cos(\psi)cos(\theta) - \sin(\lambda)\sin(\theta) \\
a_{12} &= \cos(\theta)\sin(\lambda)\sin(\phi) - \cos(\lambda)(\cos(\phi)\sin(\psi) - \cos(\psi)\sin(\phi)\sin(\theta)) \\
a_{13} &= \cos(\lambda)(\sin(\phi)\sin(\psi) + \cos(\phi)\cos(\psi)\sin(\theta)) + \cos(\phi)\cos(\theta)\sin(\lambda) \\
a_{21} &= \cos(\theta)\sin(\psi) \\
a_{22} &= \cos(\phi)\cos(\psi) + \sin(\phi)\sin(\psi)\sin(\theta) \\
a_{23} &= \cos(\phi)\sin(\psi)\sin(\theta) - \cos(\psi)\sin(\phi) \\
a_{31} &= -\cos(\lambda)\sin(\theta) - \cos(\psi)\cos(\theta)\sin(\lambda) \\
a_{32} &= \sin(\lambda)(\cos(\phi)\sin(\psi) - \cos(\psi)\sin(\phi)\sin(\theta)) + \cos(\lambda)\cos(\theta)\sin(\phi) \\
a_{33} &= \cos(\lambda)\cos(\phi)\cos(\theta) - \sin(\lambda)(\sin(\phi)\sin(\psi) + \cos(\phi)\cos(\psi)\sin(\theta))
\end{aligned}$$

Additional terms for $\ddot{\mathbf{R}}_{1G}$ when the gondola is on the curved rail When the gondola is on the curved rail, the variation of angle λ with respect to time must be considered. Therefore, additional terms must be added to the original definition of $\ddot{\mathbf{R}}_1\ \vec{r}$ given in (2.23). The new expression is given as

$$\ddot{\mathbf{R}}_{1G\ curved}\ \vec{r}_{GG} = \ddot{\mathbf{R}}_{1G\ straight}\ \vec{r}_{GG} + \vec{G}_\lambda \tag{A.33}$$

$$\vec{G}_\lambda = 2\ \left[\frac{\partial^2 \mathbf{R}_{1G\ curved}}{\partial\lambda\partial\phi}\ \dot{\lambda}\ \vec{r}_{GG} \quad \frac{\partial^2 \mathbf{R}_{1G\ curved}}{\partial\lambda\partial\theta}\ \dot{\lambda}\ \vec{r}_{GG} \quad \frac{\partial^2 \mathbf{R}_{1G\ curved}}{\partial\lambda\partial\psi}\ \dot{\lambda}\ \vec{r}_{GG}\right]\ \mathbf{R}_{2G}\ \omega_G \tag{A.34}$$

$$+ \frac{\partial \mathbf{R}_{1G\ curved}}{\partial\lambda}\ \ddot{\lambda}\ \vec{r}_{GG} + \frac{\partial^2 \mathbf{R}_{1G\ curved}}{\partial\lambda^2}\ \dot{\lambda}^2\ \vec{r}_{GG}$$

where

$$\dot{\lambda} = \frac{\dot{S}s}{R_{curve}} \qquad \ddot{\lambda} = \frac{\dot{S}\dot{s}}{R_{curve}}$$

Clearly, $\vec{G}_\lambda = 0$ if $\dot{\lambda} = \ddot{\lambda} = 0$ or $\vec{r}_{GG} = \vec{0}$.

A.3 Kane's method

Before introducing Kane's method, generalized speeds (u_r) and partial velocities are defined.

$$[u_1 ... u_n]^T = \mathbf{Y}(\vec{q})\ [\dot{q_1} ... \dot{q_n}]^T + \vec{Z}(\vec{q}) \tag{A.35}$$

where $\mathbf{Y}$ is $n \times n$ matrix, n is the number of generalized coordinates, $\vec{Z}$ is a vector, $\vec{q}$ is generalized coordinates. Rearranging (A.35) to get the generalized velocities

$$[\dot{q_1} ... \dot{q_n}]^T = \mathbf{W}\ [u_1 ... u_n]^T + \vec{X} \tag{A.36}$$

where $\mathbf{W} = \mathbf{Y}^{-1}$ and $\vec{X} = \mathbf{Y}^{-1}\vec{Z}$.

To include motion constraints into EOM, the following equation is used

$$\mathbf{A}(\vec{q})\ [u_1 ... u_n]^T + \vec{B}(\vec{q}) = 0 \tag{A.37}$$

where $\mathbf{A}$ and $\vec{B}$ are functions of generalized coordinates ($\vec{q}$). Note that the motion constraints should be expressed as linear relationships with generalized speeds[3].

Having motion constraints, c_{nh}, implies that there will be c_{nh} dependent generalized speeds. In other words, one could represent the dependent c_{nh} generalized speeds in terms of the independent $p = n - c_{nh}$ generalized speeds. This leads to the following expression

$$[u_{p+1}...u_n]^T = \mathbf{C}(\vec{q})\ [u_1...u_p]^T + \vec{D}(\vec{q}) \tag{A.38}$$

Using equation (A.38) eliminates dependent generalized speeds while applying (A.37) to the last m generalized speeds satisfies the motion constraints.

To introduce partial velocities, let $\dot{\vec{r}}^{P_k}$ and $\vec{\omega}^{B_\ell}$ be the linear and angular velocity, respectively, of point P_k and body B_ℓ in the inertial frame. k and ℓ are, respectively, the number of points and bodies in a multibody system. The linear and angular velocity, in inertial frame, can be represented as a function of generalized speeds and partial velocities such that

$$\dot{\vec{r}}^{P_k} = \frac{\partial \vec{r}^{P_k}}{\partial \vec{q}}\dot{\vec{q}} + \frac{\partial \vec{r}^{P_k}}{\partial t} \tag{A.39}$$

Similarly for $\vec{\omega}^{B_\ell}$. Substituting (A.36) in (A.39) gives

$$\begin{aligned} \dot{\vec{r}}^{P_k} &= \underbrace{[v_1...v_p]}_{\textit{partial linear velocity}}\ [u_1...u_p] + v_t \\ \vec{\omega}^{B_\ell} &= \underbrace{[\omega_1...\omega_p]}_{\textit{partial angular velocity}}\ [u_1...u_p] + \omega_t \end{aligned} \tag{A.40}$$

where v_t and ω_t are called the *velocity remainder* which contain all the terms that are not a function of generalized speeds u_r. The name *partial velocity* comes from the fact that $v_r^{P_k} = \dfrac{\partial \dot{\vec{r}}^{P_k}}{\partial u_r}$ and $\omega_r^{B_\ell} = \dfrac{\partial \vec{\omega}^{B_\ell}}{\partial u_r}$.

For the airship model shown in Fig. A.1, there is only one point on one rigid body ($k = \ell = 1$) therefore the subscripts P_k and B_ℓ can be removed. To have a further insight

[3]This type of systems is known as Nonholonomic Chaplygin type [162]

about the selection of generalized speeds and partial velocities, recall equation (A.1)

$$\begin{aligned}
\dot{\vec{r}}_{CG} =& \{ \{R_1^{-1}\}_{1st} \{\dot{q}\}_{1st\rightarrow\ 3rd} + \dot{\theta} d_{VG,z} \}\hat{i} \\
&+\{ \{R_1^{-1}\}_{2nd} \{\dot{q}\}_{1st\rightarrow\ 3rd} - \dot{\phi}(d_{VG,z} - d_{VG,x}) \}\hat{j} \\
&+\{ \{R_1^{-1}\}_{3rd} \{\dot{q}\}_{1st\rightarrow\ 3rd} - \dot{\psi} d_{VG,x} \}\hat{k} \\
\vec{\omega} =& \{ \{R_2^{-1}\}_{1st} \{\dot{q}\}_{4th\rightarrow\ 6th} \}\hat{i} \\
&+\{ \{R_2^{-1}\}_{2nd} \{\dot{q}\}_{4th\rightarrow\ 6th} \}\hat{j} \\
&+\{ \{R_2^{-1}\}_{3rd} \{\dot{q}\}_{4th\rightarrow\ 6th} \}\hat{k}
\end{aligned} \tag{A.41}$$

where $\hat{i}, \hat{j}, \hat{k}$ are unit vectors along X^E, Y^E, and Z^E, respectively. The subscripts $1st, 2nd$, and $3rd$ are respectively the first, second, and third rows of the inverse of rotation matrices. Similarly, $\{\dot{q}\}_{\ell\rightarrow k}$ are the rows of $\dot{q}$ from ℓ to k. The generalized speeds are defined in the following way

$$\begin{aligned}
u_1 &\triangleq \{R_1^{-1}\}_{1st} \{\dot{q}\}_{1st\rightarrow\ 3rd} \\
u_2 &\triangleq \{R_1^{-1}\}_{2nd} \{\dot{q}\}_{1st\rightarrow\ 3rd} \\
u_3 &\triangleq \{R_1^{-1}\}_{3rd} \{\dot{q}\}_{1st\rightarrow\ 3rd} \\
u_4 &\triangleq \dot{\phi} = \{R_2^{-1}\}_{1st} \{\dot{q}\}_{4th\rightarrow\ 6th} \\
u_5 &\triangleq \dot{\theta} = \{R_2^{-1}\}_{2nd} \{\dot{q}\}_{4th\rightarrow\ 6th} \\
u_6 &\triangleq \dot{\psi} = \{R_2^{-1}\}_{3rd} \{\dot{q}\}_{4th\rightarrow\ 6th}
\end{aligned} \tag{A.42}$$

Rearranging (A.42) to get generalized velocities

$$\begin{aligned}
\dot{q}_1 &= (R_1)_{1st} \{u\}_{1st\rightarrow 3rd} \\
\dot{q}_2 &= (R_1)_{2nd} \{u\}_{1st\rightarrow 3rd} \\
\dot{q}_3 &= (R_1)_{3rd} \{u\}_{1st\rightarrow 3rd} \\
\dot{q}_4 &= u_4 + sin(q_4)tan(q_5)u_5 + cos(q_4)tan(q_5)u_6 \\
\dot{q}_5 &= cos(q_4)u_5 - sin(q_4)u_6 \\
\dot{q}_6 &= sin(q_4)cos(q_5)u_5 + cos(q_4)sec(q_5)u_6
\end{aligned} \tag{A.43}$$

Hence, from (A.40),(A.41), and (A.42) the partial velocities are

$$\begin{aligned}
v_1 &= \hat{i} \\
v_2 &= \hat{j} \\
v_3 &= \hat{k} \\
v_4 &= (d_{VG,x} - d_{VG,z})\hat{j} \\
v_5 &= d_{VG,z}\hat{i} - d_{VG,x}\hat{k} \\
v_6 &= 0 \\
\omega_\ell &= 0 \qquad (\ell = 1 \dots 3) \\
\omega_4 &= \hat{i} \\
\omega_5 &= \hat{j} \\
\omega_6 &= \hat{k}
\end{aligned} \tag{A.44}$$

Note that equations (A.42) are not unique, meaning it is possible to redefine generalized speeds, and subsequently partial velocities, in different forms.

In order to derive EOM, generalized active (external) and inertia forces are required. For point P_k and body B_ℓ, the generalized active force is defined as the dot product of partial velocity with resultant force (R^{P_k}). Similarly, the generalized active torque is the dot product of partial angular velocity with resultant torque (T^{B_ℓ}).

$$\begin{aligned}
F_r^{P_k} &= v_r^{P_k} \cdot R^{P_k} \qquad (r = 1 \cdots p) \\
F_r^{B_\ell} &= \omega_r^{B_\ell} \cdot T^{B_\ell}
\end{aligned} \tag{A.45}$$

The r-th generalized active force is the summation of all active forces (A.45) being applied at all points P_k and all bodies B_ℓ

$$F_r = \sum_{k=1}^{k_{total}} F_r^{P_k} + \sum_{\ell=1}^{\ell_{total}} F_r^{B_\ell} \qquad (r = 1 \cdots\ p) \tag{A.46}$$

On the other hand, generalized inertia forces and torques are given as

$$\begin{aligned}
(F^{P_k})_r^* &= v_r^{P_k} \cdot (R^{P_k})^* \qquad (r = 1 \cdots p) \\
(F^{B_\ell})_r^* &= \omega_r^{B_\ell} \cdot (T^{B_\ell})^*
\end{aligned} \tag{A.47}$$

where $(R^{P_k})^*$ and $(T^{B_\ell})^*$ are the RHS of Newton-Euler equations but with a negative sign. Linear and angular acceleration ($\ddot{\vec{r}}^{P_k}$ and $\dot{\vec{\omega}}^{B_\ell}$, respectively) are in inertial frame, m^{P_k} is the mass of P_k, $I^{B_\ell^{CG}}$ is the moment of inertia of B_ℓ at its centre of mass.

$$\begin{aligned}(R^{P_k})^* &= -m^{P_k}\,\ddot{\vec{r}}^{P_k} \\ (T^{B_\ell})^* &= -I^{B_\ell^{CG}}\dot{\vec{\omega}}^{B_\ell} - \vec{\omega}^{B_\ell} \times (I^{B_\ell^{CG}}\vec{\omega}^{B_\ell})\end{aligned} \tag{A.48}$$

In a similar way to (A.46), the r-th generalized inertia force is

$$F_r^* = \sum_{k=1}^{k_{total}} (F_r^{P_k})^* + \sum_{\ell=1}^{\ell_{total}} (F_r^{B_\ell})^* \qquad (r = 1 \cdots\ p) \tag{A.49}$$

Since the applied forces R^{P_k} and moments T^{B_ℓ} in equation (A.45) represent the LHS of Newton-Euler equations (2.1) while the RHS of Newton-Euler (2.1) is represented by $-(R^{P_k})^*$ and $-(T^{B_\ell})^*$ in (A.48), then the following relationship can be deduced

$$\begin{aligned}R^{P_k} + (R^{P_k})^* &= 0 \\ T^{B_\ell} + (T^{B_\ell})^* &= 0\end{aligned} \tag{A.50}$$

Following Jourdain's principle [163] by projecting equations (A.50) into the vector space of partial velocities [4] ($v_r^{P_k}$ and $\omega_r^{B_\ell}$) and summing both equations over all P_k and B_ℓ gives

$$\sum_{k=1}^{k_{total}} v_r^{P_k} \cdot \{\ R^{P_k} + (R^{P_k})^*\ \} + \sum_{\ell=1}^{\ell_{total}} \omega_r^{B_\ell} \cdot \{\ T^{B_\ell} + (T^{B_\ell})^*\ \} = 0 \tag{A.51}$$

Clearly, substituting (A.46) and (A.49) into (A.51) yields (A.52)

$$F_r + F_r^* = 0 \tag{A.52}$$

A.4 Boltzmann-Hamel (B-H)

Starting from d'Alembert Lagrange principle

$$\sum_{k=1}^{n} (\frac{d}{dt}\frac{\partial T}{\partial \dot{q_k}} - \frac{\partial T}{\partial q_k} - Q_k)\delta q_k = 0 \tag{A.53}$$

[4] In Jourdain's work, the Jacobian matrix was used instead of partial velocities. However, as shown in [163], the columns of the Jacobian matrix are identical to the partial velocities introduced by Kane.

where T is the kinetic energy, q_k and $\dot{q}_k$ are the k-th generalized coordinate and generalized velocity, respectively. Q_k is the k-th generalized active force or torque (applied external force/torque), and δq_k is the virtual displacement in q_k direction. For the holonomic case, the term in parenthesis is equated to zero for all k which yields Euler-Lagrange equation (A.24). This is valid as long as q_k are independent of each other, *i.e.,*

$$\delta q_k \cdot \delta q_j = 0 \qquad\qquad (k \neq j)$$

Nonetheless, that is not the case for nonholonomic constraints where the generalized coordinates are often dependent on each other. Therefore, equation (A.53) shall be modified to accommodate nonholonomic constraints. This is done by introducing "quasi-velocity", velocities in the body-frame, to the set of generalized variables. The quasi-velocities are defined as affine functions of generalized velocities and generalized coordinates; The same procedure, with a slightly different notations, of generalized speeds (A.35) and generalized velocities (A.36) is used here to define quasi-velocity.

$$\begin{aligned} [u_1 ... u_n]^T &= \mathbf{Y^{-1}}(\vec{q}, t)\ [\dot{q_1} ... \dot{q_n}]^T + \vec{Z}(\vec{q}, t) \\ u_j &= \sum_{i=1}^{n} Y_{ji}(\vec{q}, t)\dot{q}_i + Z_j(\vec{q}, t) \qquad (j = 1 \cdots n) \end{aligned} \tag{A.54}$$

u_j are the quasi-velocities, n is the number of DOF, $\mathbf{Y}$ and $\mathbf{Z}$ are functions of generalized coordinates and time. In order to maintain the same notations used in the literature for B-H derivation, $\mathbf{Y^{-1}}$ is used here instead of $\mathbf{Y}$ (the inverse of that defined in Kane's approach). To account for nonholonomic constraints, the same technique that was used in Kane's method (A.37) is used here where additional c_{nh} quasi-velocities are added and then set to zero. These additional quasi-velocities characterize the nonholonomic constraints. Let $p = n - c_{nh}$, the first p equations of u_r are given by A.54 (for $j = 1...p$) while the last $p - n$ equations are given by A.55.

$$u_j = \sum_{i=1}^{n} Y_{ji}(\vec{q}, t)\dot{q}_i + Z_j(\vec{q}, t) = 0 \qquad (j = p + 1 \cdots n) \tag{A.55}$$

The generalized velocities can be obtained from the quasi-velocities by rearranging

$$\begin{aligned}[\dot{q}_1 ... \dot{q}_n]^T &= \mathbf{W}(\vec{q}, t)\ [u_1 ... u_n]^T + \vec{X}(\vec{q}, t) \\ \dot{q}_i &= \sum_{j=1}^{p} W_{ij}(\vec{q}, t) u_j + X_i(\vec{q}, t) \qquad (i = 1 \cdots n)\end{aligned} \tag{A.56}$$

where $\mathbf{W^{-1}Y} = \mathbf{I}$, the identity matrix, $\vec{X} = -\mathbf{Y}\vec{Z}$. In order to account for rheonomic constraints, $\mathbf{Y}, \vec{Z}, \mathbf{W}$, and $\vec{X}$ are presented here as functions of generalized coordinates q and time[5]. The goal is to replace the kinetic energy $T(q, \dot{q})$ in Eq. (A.53) by $\overline{T}(q, u)$ [164] such that $\overline{T}$ is now a function of quasi-velocities instead of generalized velocities.

$$\frac{\partial T}{\partial \dot{q}_k} = \sum_{i=1}^{n} \frac{\partial \overline{T}}{\partial u_i} \frac{\partial u_i}{\partial \dot{q}_k} = \sum_{i=1}^{n} \frac{\partial \overline{T}}{\partial u_i} Y_{ki} \tag{A.57}$$

where $Y_{ki} = \dfrac{\partial u_i}{\partial \dot{q}_k}$ and $\mathbf{Y}^{-1} = Y_{ji}$. Differentiating A.57 with respect to time

$$\frac{d}{dt}\frac{\partial T}{\partial \dot{q}_k} = \sum_{i=1}^{n} Y_{ki} \frac{d}{dt}\frac{\partial \overline{T}}{\partial u_i} + \sum_{i=1}^{n} \frac{\partial \overline{T}}{\partial u_i} \dot{Y}_{ki} \tag{A.58}$$

$$\begin{aligned}\frac{\partial T}{\partial q_k} &= \frac{\partial \overline{T}}{\partial q_k} + \sum_{i=1}^{n} \frac{\partial \overline{T}}{\partial u_i} \frac{\partial u_i}{\partial q_k} \\ &= \frac{\partial \overline{T}}{\partial q_k} + \sum_{i=1}^{n} \frac{\partial \overline{T}}{\partial u_i} \underbrace{\left(\frac{\partial u_i}{\partial q_k}\Big|_{\dot{q}_k = 0} + \sum_{\ell=1}^{n} \frac{\partial Y_{\ell i}}{\partial q_k} \dot{q}_\ell\right)}_{\textcircled{1}}\end{aligned} \tag{A.59}$$

To better understand the manipulation of $\dfrac{\partial u_i}{\partial q_k}$ in Eq. (A.59), recall Eq. (A.54) and differentiate with respect to q

$$\sum_{i=1}^{n} \frac{\partial u_i}{\partial q_k} = \frac{\partial \mathbf{Y}(q, t)}{\partial q} \dot{q} + \frac{\partial \vec{Z}(q, t)}{\partial q} \tag{A.60}$$

Equation (A.60) is equivalent to (1) term. This can be shown by decomposing (1) into two parts; $\dfrac{\partial Y_{\ell i}}{\partial q_\ell}\dot{q}_\ell$ and $\dfrac{\partial u_i}{\partial q_k}\Big|_{\dot{q}_k = 0}$. For the first part, differentiate u in (A.54) twice; Once with respect to $\dot{q}$ and once with respect to q. The result is then multiplied by $\dot{q}$.

$$\frac{\partial}{\partial q_k} \sum_{i=1}^{n} \left(\frac{\partial u_i}{\partial \dot{q}_k}\right) \dot{q}_k = \sum_{i=1}^{n} \sum_{\ell=1}^{n} \frac{\partial}{\partial q_k} (Y_{\ell i}) \dot{q}_\ell = \frac{\partial \mathbf{Y}(\vec{q}, t)}{\partial \vec{q}} \dot{\vec{q}} \tag{A.61}$$

[5]Note that time was omitted in the derivation of Kane's equations but it can be introduced in an identical manner.

For the second term, taking the derivative of (A.54) with respect to q_k when $\dot{q}_k = 0$ gives

$$\sum_{i=1}^{n} \frac{\partial u_i}{\partial q_k}|_{\dot{q}_k=0} = \cancelto{0}{\mathbf{Y}(q,t)\,\dot{q}} + \frac{\partial \vec{Z}(\vec{q},t)}{\partial \vec{q}} \tag{A.62}$$

Clearly, summing (A.61) and (A.62) gives (A.60) which is equivalent to (1) in (A.59). To eliminate $\dot{q}$ in (A.61), (A.56) is substituted. The resulting equation is a function of quasi-velocities. Collecting terms and substituting (A.58) and (A.59) into (A.53) yields

$$\delta\vec{\theta}(\frac{d}{dt}\frac{\partial \overline{T}}{\partial \vec{u}} + \frac{\partial \overline{T}}{\partial \vec{u}}\,\vec{\eta} - \mathbf{W}^{-1}\frac{\partial \overline{T}}{\partial \vec{q}} - \mathbf{W^{-1}}\vec{Q}) = 0$$

where

$$\begin{aligned}
&\delta\vec{\theta} = \delta\vec{q}\mathbf{Y}\\
&\eta = \gamma_{r\ell}\ u_\ell + \gamma_r\\
&\gamma_{r\ell} = -\gamma_{\ell r} = \sum_{i=1}^{n}\sum_{k=1}^{n}(\frac{\partial Y_{ji}}{\partial q_k} - \frac{\partial Y_{jk}}{\partial q_k})W_{k\ell}W_{ir}\\
&\gamma_r = \sum_{i=1}^{n}\sum_{k=1}^{n}(\frac{\partial Y_{ji}}{\partial q_k} - \frac{\partial Y_{jk}}{\partial q_k})X_k W_{ir} + \sum_{i=1}^{n}(\frac{\partial Y_{ji}}{\partial t} - \frac{\partial Z_j}{\partial q_i})W_{ir}
\end{aligned} \tag{A.63}$$

Expressions $\gamma_{r\ell}$ and γ_r are adopted from [165, 166] by using transpositional relationship, explained in Chapter III of [19]. The new form (A.63) is now a function of quasi-coordinates, θ_j. With this form, it is possible to equate the rows of the parenthesis term to zero if the quasi-coordinates are independent of each other. This can be done in a similar fashion to Kane's approach where dependent quasi-velocities are rewritten as linear functions of independent quasi-velocities. (A.38). Setting the parenthesis term to zero gives (A.64)

$$\frac{d}{dt}\frac{\partial \overline{T}}{\partial \vec{u}} + \frac{\partial \overline{T}}{\partial \vec{u}}\,\vec{\eta} - \mathbf{W}^{-1}\frac{\partial \overline{T}}{\partial \vec{q}} = \mathbf{W^{-1}}\vec{Q} \tag{A.64}$$

Since the potential energy is not a function of generalized velocity, the kinetic energy $\overline{T}$ can be replaced with the Lagrangian ($L = \overline{T} - V$) in all previous manipulations. Therefore,

the general B-H equation, in index form, is given as follows

$$\begin{aligned}\frac{d}{dt}\left(\frac{\partial L}{\partial u_r}\right) &- \sum_{i=1}^{n}\frac{\partial L}{\partial q_i}W_{ir} \\ &+ \sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{k=1}^{n}\sum_{\ell=1}^{p}\frac{\partial L}{\partial u_j}\left(\frac{\partial Y_{ji}}{\partial q_k}-\frac{\partial Y_{jk}}{\partial q_i}\right)W_{k\ell}W_{ir}u_\ell \\ &+ \sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{k=1}^{n}\frac{\partial L}{\partial u_j}\left(\frac{\partial Y_{ji}}{\partial q_k}-\frac{\partial Y_{jk}}{\partial q_i}\right)X_k W_{ir} \\ &+ \sum_{i=1}^{n}\sum_{j=1}^{n}\frac{\partial L}{\partial u_j}\left(\frac{\partial Y_{ji}}{\partial t}-\frac{\partial Z_j}{\partial q_i}\right)W_{ir} = \sum_{i=1}^{n}Q_i W_{ir}\end{aligned} \tag{A.65}$$

where Q_i is the applied force or torque, expressed in generalized coordinates q_i.

The kinetic energy term in the Lagrangian should be constructed using quasi-velocities, as discussed earlier. However, the derivative of the Lagrangian with respect to the quasi-velocities ($\frac{\partial L}{\partial u_j}$) in (A.65) must be computed before enforcing the last c_{nh} quasi-velocities to zero (A.55).

A.5 Airship EOM using N-E

Starting with Newton-Euler (2.1), the model will be constructed about CV of the airship. Expressing the equations of motion in body-frame (Fig. A.1) simplifies the inertia tensor by exploiting the geometrical property of the airship, namely symmetry in the longitudinal plane. As a result, the products of inertia $I_{xy} = I_{yx} = I_{zy} = I_{yz} = 0$.
In order to expand equations (A.5) and (A.22), the following notations, depicted in Fig. A.3, will be used to represent the components of forces, moments, and velocities.

Forces about CV	$\vec{F}_{CV} = [\ F_x \quad F_y \quad F_z\ ]^T$
Moments about CV	$\vec{M}_{CV} = [\ L \quad M \quad N\ ]^T$
Linear velocity in body frame	$\vec{V}_{CV} = [\ u \quad v \quad w\ ]^T$
Linear acceleration in body frame	$\vec{a}_{CV} = [\ \dot{u} \quad \dot{v} \quad \dot{w}\ ]^T$

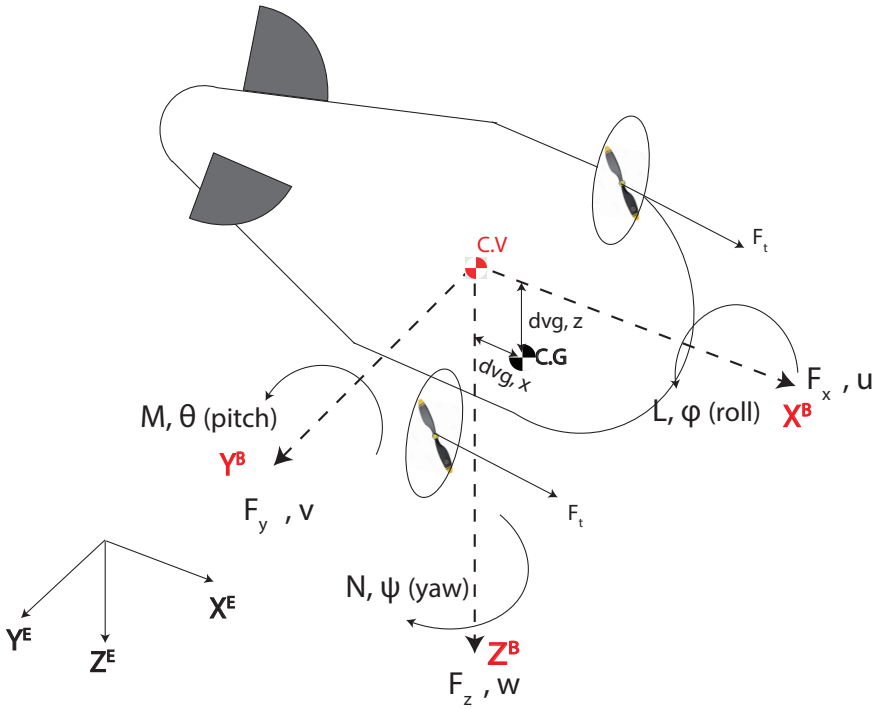


Figure A.3: Forces, moments, and velocity components along X^B, Y^B, and Z^B directions in the body frame of reference.

Angular velocity in body frame	$\vec{\omega}_{CV} = [\ \omega_x \quad \omega_y \quad \omega_z\ ]^T$
Angular acceleration in body frame	$\vec{\Omega}_{CV} = [\ \dot{\omega}_x \quad \dot{\omega}_y \quad \dot{\omega}_z\ ]^T$
CG position with respect to CV	$\vec{r}_{VG} = [\ d_{VG,x} \quad 0 \quad d_{VG,z}\ ]^T$

Using this notation, the cross products terms, as well as the product of the inertia tensor with angular acceleration, in (A.5) and (A.22) are derived and presented in vectorial form

$$\vec{\omega} \times \vec{V}_{CV} = \{\ \omega_y w - v\omega_z\ \}\hat{i} - \{\ \omega_x w - u\omega_z\ \}\hat{j} + \{\ \omega_x v - u\omega_y\ \}\hat{k} \tag{A.66}$$

$$\vec{\omega} \times \vec{r}_{VG} = \{\ \omega_y d_{VG,z}\ \}\hat{i} - \{\ \omega_x d_{VG,z} - \omega_z d_{VG,x}\ \}\hat{j} + \{\ -\omega_y d_{VG,x}\ \}\hat{k} \tag{A.67}$$

$$\vec{\Omega} \times \vec{r}_{VG} = \{\ \dot{\omega}_y d_{VG,z}\ \}\hat{i} - \{\ \dot{\omega}_x d_{VG,z} - \dot{\omega}_x d_{VG,x}\ \}\hat{j} + \{\ -\dot{\omega}_y d_{VG,x}\ \}\hat{k} \tag{A.68}$$

$$\begin{aligned}\vec{\omega}\times(\vec{\omega}\times\vec{r}_{VG}) = &\{\ -\omega_y^2 d_{VG,x} + \omega_z\omega_x d_{VG,z} - \omega_z^2 d_{VG,x}\ \} && \hat{i}\\ &+\{\ \omega_y\omega_x d_{VG,x} + \omega_y\omega_z d_{VG,z}\ \} && \hat{j}\\ &+\{\ -\omega_x^2 d_{VG,z} + \omega_x\omega_z d_{VG,x} - \omega_y^2 d_{VG,z}\ \} && \hat{k}\end{aligned} \tag{A.69}$$

$$\begin{aligned}m\vec{r}_{VG}\times\vec{a}_{CV} = &\{\ -m\dot{v}d_{VG,z}\ \} && \hat{i}\\ &-\{\ md_{VG,x}\dot{w} - md_{VG,z}\dot{u}\ \} && \hat{j}\\ &+\{\ md_{VG,x}\dot{v}\ \} && \hat{k}\end{aligned} \tag{A.70}$$

$$\begin{aligned}m\vec{r}_{VG}\times(\vec{\omega}\times\vec{V}_{CV}) = &\{\ md_{VG,z}(-\omega_z u + \omega_x w)\} && \hat{i}\\ &-\{\ md_{VG,x}(\omega_x v - \omega_y u) - md_{VG,z}(\omega_y w - \omega_z v)\ \} && \hat{j}\\ &+\{\ md_{VG,x}(\omega_z u - \omega_x w)\ \} && \hat{k}\end{aligned} \tag{A.71}$$

$$\begin{aligned}I_{CV}\vec{\Omega}_{CV} = &\{\ I_x\dot{\omega}_x - I_{xy}\dot{\omega}_y - I_{xz}\dot{\omega}_z\} && \hat{i}\\ &+\{\ -I_{yx}\dot{\omega}_x + I_y\dot{\omega}_y - I_{yz}\dot{\omega}_z\ \} && \hat{j}\\ &+\{\ -I_{zx}\dot{\omega}_x - I_{zy}\dot{\omega}_y + I_z\dot{\omega}_z\ \} && \hat{k}\end{aligned} \tag{A.72}$$

$$\begin{aligned}I_{CV}\vec{\omega}_{CV} = &\{\ I_x\omega_x - I_{xy}\omega_y - I_{xz}\omega_z\} && \hat{i}\\ &+\{\ -I_{yx}\omega_x + I_y\omega_y - I_{yz}\omega_z\ \} && \hat{j}\\ &+\{\ -I_{zx}\omega_x - I_{zy}\omega_y + I_z\omega_z\ \} && \hat{k}\end{aligned} \tag{A.73}$$

$$\begin{aligned}\vec{\omega}\times(I_{CV}\vec{\omega}_{CV}) = &\{\ -I_{zx}\omega_x\omega_y - I_{zy}\omega_y^2 + I_z\omega_y\omega_z + I_{yx}\omega_z\omega_x - I_y\omega_y\omega_z + I_{yz}\omega_z^2\} && \hat{i}\\ &-\{\ -I_{zx}\omega_x^2 - I_{zy}\omega_y\omega_x + I_z\omega_x\omega_z - I_x\omega_x\omega_z + I_{xy}\omega_y\omega_z + I_{xz}\omega_z^2\ \} && \hat{j}\\ &+\{\ -I_{yx}\omega_x^2 + I_y\omega_y\omega_x - I_{yz}\omega_z\omega_x - I_x\omega_y\omega_x + I_{xy}\omega_y^2 + I_{xz}\omega_z\omega_y\ \} && \hat{k}\end{aligned} \tag{A.74}$$

where $\hat{i}, \hat{j}$, and $\hat{k}$ are unit vectors along the X^B, Y^B, and Z^B, respectively.

Substituting back the terms (A.66)-(A.74) into Newton-Euler equations (A.5) and (A.22) yields

$$m\{\ \dot{u} - v\omega_z + w\omega_y - d_{VG,x}(\omega_y^2 + \omega_z^2) + d_{VG,z}(\omega_x\omega_z + \dot{\omega}_y)\} = \qquad F_x$$

$$
\begin{aligned}
m\{\ \dot{v} - w\omega_x + u\omega_z + d_{VG,z}(\omega_y\omega_z - \dot{\omega}_x) + d_{VG,x}(\omega_x\omega_y + \dot{\omega}_z)\} = \quad & F_y \\
m\{\ \dot{w} - u\omega_y + v\omega_x - d_{VG,z}(\omega_y^2 + \omega_x^2) + d_{VG,x}(\omega_x\omega_z - \dot{\omega}_y)\} = \quad & F_z
\end{aligned}
\tag{A.75}
$$

$$
\begin{aligned}
I_x\dot{\omega}_x + (I_z - I_y)\omega_y\omega_z - (\dot{\omega}_z + \omega_x\omega_y)I_{xz} + (\omega_z^2 - \omega_y^2)\cancelto{0}{I_{yz}} + (\omega_x\omega_z - \dot{\omega}_y)\cancelto{0}{I_{xy}} \\
-md_{CV,z}(\dot{v} - w\omega_x + u\omega_z) = L \\
I_y\dot{\omega}_y + (I_x - I_z)\omega_x\omega_z - (\dot{\omega}_x + \omega_z\omega_y)\cancelto{0}{I_{xy}} + (\omega_x^2 - \omega_z^2)I_{zx} + (\omega_y\omega_x - \dot{\omega}_z)\cancelto{0}{I_{yz}} \\
+m\{d_{CV,z}(\dot{u} - v\omega_z + w\omega_y) - d_{CV,x}(\dot{w} - u\omega_y + v\omega_x)\ \} = M \\
I_z\dot{\omega}_z + (I_y - I_x)\omega_y\omega_x - (\dot{\omega}_y + \omega_x\omega_z)\cancelto{0}{I_{yz}} + (\omega_y^2 - \omega_x^2)\cancelto{0}{I_{xy}} + (\omega_y\omega_z - \dot{\omega}_x)I_{zx} \\
+md_{CV,x}(\dot{v} - w\omega_x + u\omega_z) = N
\end{aligned}
\tag{A.76}
$$

Owing to the symmetry in the longitudinal plane, I_{xy} and I_{yz} are equal to zero. A more compact form of the equations of motion can be obtained by reintroducing the variables in matrix form. Furthermore, the added mass terms, which were discussed earlier for LTA vehicles, can be introduced to the mass and inertia matrices. The compact form for the equation of motion is defined as

$$
\mathbf{M}\dot{\vec{\nu}} + \vec{D}(\vec{\nu}) = \vec{F} \tag{A.77}
$$

where $\mathbf{M}$ is the mass matrix, $\vec{D}$ is the dynamic forces containing Coriolis and Centrifugal terms, $\vec{F}$ is the sum of external forces. The external forces are aerodynamic forces $\vec{A}(\vec{\nu})$, buoyancy forces $\vec{G}(\mathbf{R_1})$, $\mathbf{R_1}$ is the transformation matrix (appendix A.30), and input forces $\vec{U}$. Control inputs applied to the airship through propellers and gondola actuators are all contained in $\vec{U}$. The structure of $\vec{G}$ and $\vec{U}$ is discussed later in a separate section.

$$
\vec{F} = \vec{A}(\vec{\nu}) + \vec{G}(\mathbf{R_1}) + \vec{U}
$$

The new notation is given as follows

$$\textbf{\textit{Forces and moments}} \qquad \vec{F} = [\ F_{CV} \quad M_{CV}\ ] = [\ F_x \quad F_y \quad F_z \quad L \quad M \quad N\ ]^T$$

$$\textbf{\textit{Linear and angular velocity}} \qquad \vec{\nu} = [\ \vec{V} \quad \vec{\omega}\ ] = [\ u \quad v \quad w \quad \omega_x \quad \omega_y \quad \omega_z\ ]^T$$

$$\textbf{\textit{Linear and angular acceleration}} \qquad \dot{\vec{\nu}} = [\ \vec{a} \quad \vec{\Omega}\ ] = [\ a_x \quad a_y \quad a_z \quad \Omega_x \quad \Omega_y \quad \Omega_z\ ]^T$$

$$\textbf{\textit{Mass matrix}} \qquad \mathbf{M} = \begin{bmatrix} \mathbf{M_{all}} & -m\mathbf{d_{VG}} \\ m\mathbf{d_{VG}} & \mathbf{I_{all}} \end{bmatrix}$$

$$\mathbf{M_{all}} = \begin{bmatrix} (1+k_1)m & 0 & 0 \\ 0 & (1+k_2)m & 0 \\ 0 & 0 & (1+k_2)m \end{bmatrix}$$

$$\mathbf{d_{VG}} = \begin{bmatrix} 0 & -d_{VG,z} & 0 \\ d_{VG,z} & 0 & -d_{VG,x} \\ 0 & d_{VG,x} & 0 \end{bmatrix}$$

$$\mathbf{I_{all}} = \begin{bmatrix} I_x & 0 & -I_{xz} \\ 0 & I_y(1+k') & 0 \\ -I_{xz} & 0 & I_z(1+k') \end{bmatrix}$$

$$\begin{aligned} \vec{D}(\vec{\nu}) = \mathbf{C}(\vec{\nu})\vec{\nu} &= \begin{bmatrix} \vec{\omega} \times \mathbf{M_{all}}\vec{V} - m\vec{\omega} \times (\mathbf{d_{VG}}\vec{\omega}) \\ \vec{\omega} \times \mathbf{I_{all}}\vec{\omega} + \vec{\omega} \times m\mathbf{d_{VG}}\vec{V} + \vec{V} \times (\mathbf{M_{all}}\vec{V} - m\mathbf{d_{VG}}\vec{\omega}) \end{bmatrix} \\ &= \begin{bmatrix} \vec{\omega} \times \mathbf{M_{all}}\vec{V} + m\vec{\omega} \times (\vec{\omega} \times \vec{r}_{VG}) \\ \vec{\omega} \times \mathbf{I_{all}}\vec{\omega} + m\vec{r}_{VG} \times (\vec{\omega} \times \vec{V}) \end{bmatrix} \end{aligned} \tag{A.78}$$

where k_1 and k_2 are, respectively, the axial (OX^B) and lateral (OY^B) added mass coefficient, while k' is the added inertia coefficients. The added mass and added inertia coefficients are now combined with the body mass and inertia in $\mathbf{M_{all}}$ and $\mathbf{I_{all}}$. Consequently, equations (A.78) are different from those derived earlier (A.75,A.76). $\mathbf{C}(\vec{\nu})$ is a parametrization matrix that can be represented in various forms [167]. Expansion of $\vec{D}(\vec{\nu})$ is given in appendix (A.88).

A.6 EOM using E-L

Recall that it is inconvenient to use Lagrange equations to derive an equation of motion in body-fixed frame, owing to non-integrable velocity. Alternatively, Kirchoff's equations are used (A.27) for which we have $V_1 = \vec{V}$, $V_2 = \vec{\omega}$.

$$\begin{aligned} \frac{d}{dt}(\frac{\partial T}{\partial \vec{V}}) + \vec{\omega} \times \frac{\partial T}{\partial \vec{V}} &= \vec{F}_{CV} \\ \frac{d}{dt}(\frac{\partial T}{\partial \vec{\omega}}) + \vec{\omega} \times \frac{\partial T}{\partial \vec{\omega}} + \vec{V} \times \frac{\partial T}{\partial \vec{V}} &= \vec{M}_{CV} \end{aligned} \tag{A.79}$$

The kinetic energy (T) is defined as such

$$T = \frac{1}{2}\left(\vec{V}^T \mathbf{M}_{all}\, \vec{V} - \vec{V}^T m\mathbf{d}_{VG}\, \vec{\omega} + \vec{\omega}^T m\mathbf{d}_{VG}\, \vec{V} + \vec{\omega}^T \mathbf{I}_{all}\, \vec{\omega}\right) \tag{A.80}$$

Substituting the kinetic energy equation (A.80) into (A.79) results in

$$\begin{aligned} &\frac{d}{dt}(\mathbf{M_{all}}\vec{V} - m\mathbf{d_{VG}}\vec{\omega}) + \vec{\omega} \times (\mathbf{M_{all}}\vec{V} - m\mathbf{d_{VG}}\vec{\omega}) = \vec{F}_{CV} \\ &= \underbrace{(\mathbf{M_{all}}\vec{a} - m\mathbf{d_{VG}}\vec{\Omega})}_{\text{identical to the first three rows of } \mathbf{M}\dot{\vec{\nu}}} + \underbrace{\vec{\omega} \times (\mathbf{M_{all}}\vec{V} - m\mathbf{d_{VG}}\vec{\omega})}_{\text{identical to the first row of } \vec{D}(\vec{\nu}) \text{ (A.78) or the first three equations in (A.88)}} = \vec{F}_{CV} \end{aligned}$$

$$\begin{aligned} &\frac{d}{dt}(\mathbf{I_{all}}\vec{\omega} + m\mathbf{d_{VG}}\vec{V}) + \vec{\omega} \times (\mathbf{I_{all}}\vec{\omega} + m\mathbf{d_{VG}}\vec{V}) + \vec{V} \times (\mathbf{M_{all}}\vec{V} - m\mathbf{d_{VG}}\vec{\omega}) = \vec{M}_{CV} \\ &= \underbrace{(\mathbf{I_{all}}\vec{\Omega} + m\mathbf{d_{VG}}\vec{a})}_{\text{identical to the last three rows of } \mathbf{M}\dot{\vec{\nu}}} + \underbrace{\vec{\omega} \times (\mathbf{I_{all}}\vec{\omega} + m\mathbf{d_{VG}}\vec{V}) + \vec{V} \times (\mathbf{M_{all}}\vec{V} - m\mathbf{d_{VG}}\vec{\omega})}_{\text{identical to the second row of } \vec{D}(\vec{\nu}) \text{ (A.78) or the last three equations in (A.88)}} = \vec{M}_{CV} \end{aligned} \tag{A.81}$$

It is also possible to include the external forces and moments in the Lagrangian. For example, the buoyancy forces $\vec{G}(\mathbf{R_1})$, would be introduced into the Lagrangian as a potential energy term

$$\begin{aligned} L &= T - U \\ \vec{G}(\mathbf{R_1}) &= \frac{\partial U}{\partial q} \end{aligned} \tag{A.82}$$

Since Kirchoff's equations only consider the kinetic energy, it would be necessary to use the Quasi-Lagrange equation (A.28) to include the potential energy.

A.7 Holonomic and Nonholonomic Constraints

Kinematics play an essential role in constructing mathematical models for physical systems. Familiarity with kinematics terminologies facilitates the classification of the investigated system. There are generally two types of dynamics constraints; Holonomic and nonholonomic constraints. Holonomic constraints refer to the type of constraints that restrain possible configurations of the system. Subsequently, the number of DOF of the system is reduced. Holonomic constraints are commonly presented in the following form

$$C(q) = 0 \tag{A.83}$$

$$C(q, t) = 0 \tag{A.84}$$

where (A.83) and (A.84) are called scleronomic and rheonomic constraints, respectively. The holonomic constraints are functions of state (q), as in (A.83), and possibly time, as in (A.84).

On the other hand, the nonholonomic constraints do not reduce the DOF nor do they restrain the configurations of a system. However, nonholonomic constraints affect the manner in which a system reaches a desired configuration. Thus, the term "motion constraints" is used interchangeably with nonholonomic constraints. The general nonholonomic form is given as

$$C(q, \dot{q}, t) = 0 \tag{A.85}$$

In addition to generalized coordinates and time, the nonholonomic constraints are functions of generalized velocities ($\dot{q}$) as well. The latter occurs owing to having non-integrable terms. Caution should be taken when classifying nonholonomic constraints as they can be confused with holonomic constraints in some situations. The following conditions demonstrate the cases in which a constraint is called nonholonomic. Firstly, if the constraint equation (A.85) can not be reduced to the form given in (A.84), *i.e.,* contains non-integrable terms. For example, Lagrange equations (A.24) can be presented as a constraint of the form (A.86), yet it is not considered nonholonomic.

$$C(q, \dot{q}, \ddot{q}, t) = 0 \tag{A.86}$$

Secondly, inequality constraints, shown in (A.87), are also considered nonholonomic [6] [170]. Thirdly, using velocities in body-frame (quasi-velocities) while formulating EOM. Using velocity in body frame often simplifies the EOM [25]. However, velocities in body frame are non-integrable, as discussed in section 2.1.

$$C(q, \dot{q}, t) \leq 0 \tag{A.87}$$

Dynamic term for airship with 6 DOF, constructed about CV

$$\begin{aligned}
D_x &= (1+k_2)\, mw\omega_y - (1+k_2)\, m\omega_z\, v - m\left(d_{VG,x}\ \left({\omega_z}^2 + {\omega_y}^2\right) - d_{VG,z}\, \omega_z\, \omega_x\right) \\
D_y &= (1+k_1)\, mu\omega_z - (1+k_2)\, m\omega_x\, w + m\left(d_{VG,x}\, \omega_x\, \omega_y + d_{VG,z}\, \omega_z\, \omega_y\right) \\
D_z &= (1+k_2)\, mv\omega_x - (1+k_1)\, m\omega_y\, u - m\left(d_{VG,z}\ \left({\omega_x}^2 + {\omega_y}^2\right) - d_{VG,x}\, \omega_z\, \omega_x\right) \\
D_L &= (1+k')\left(I_z - I_y\right)\omega_z\, \omega_y - I_{xz}\, \omega_x\, \omega_y + m d_{VG,z}\ \left(\omega_x\, w - u\omega_z\right) \\
D_M &= \left(I_x - (1+k')\, I_z\right)\omega_x\, \omega_z + I_{xz}\ \left({\omega_x}^2 - {\omega_z}^2\right) + m\left(d_{VG,z}\ \left(-v\omega_z + w\omega_y\right) + d_{VG,x}\ \left(-v\omega_x + u\omega_y\right)\right) \\
D_N &= \left((1+k')\, I_y - I_x\right)\omega_y\, \omega_x + I_{xz}\, \omega_y\, \omega_z + m d_{VG,x}\ \left(-\omega_x\, w + u\omega_z\right)
\end{aligned} \tag{A.88}$$

[6] Some references, such as [168, 169] , treat inequality constraints as complementarity constraints which are presented as $0 \leq a \perp b \geq 0$.

Appendix B

System Identification

B.1 Custom Load Cell Mount

To identify unknown parameters in the system, namely moment of inertia, added mass, and added inertia of the blimp, a set of experiments were carried out. A sequence of forces are applied to the multibody system and the response of the system is traced. The time history of the applied force is measured using a load cell [171]. The load cell is powered by a 9V battery and produces an analog signal between $0.5 - 5V$. The load cell is connected to Arduino Mega 2560 [172] for reading and storing the output signal. Moreover, the load cell requires a mount in order to be used conveniently. A hinged arm was designed to convey the applied force to a uni-directional load cell while making sure that the friction at the hinge is minimal. The mount is depicted in Fig. B.1. It is worth noting that a total of four reflectors were attached to the load cell mount to trace the motion and orientation of the load cell.

After assembling the load cell apparatus, calibration was performed by placing an object with a known weight on the load cell and recording the output voltage. Repeating this process using multiple weights yields a plot of voltage vs force that can be nicely represented with a linear fit (Fig.B.2).

During the experiment, data is stored on Arduino's EEPROM by first setting all bytes

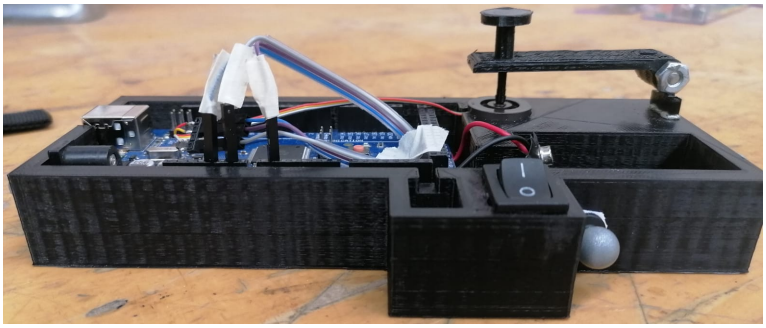


Figure B.1: Load cell mount used to measure the applied force.

to zero using script B.1 followed by saving the signal that is received from the load cell (through analog pin) using script B.2. Finally, the data is retrieved from Arduino using reading script B.3.

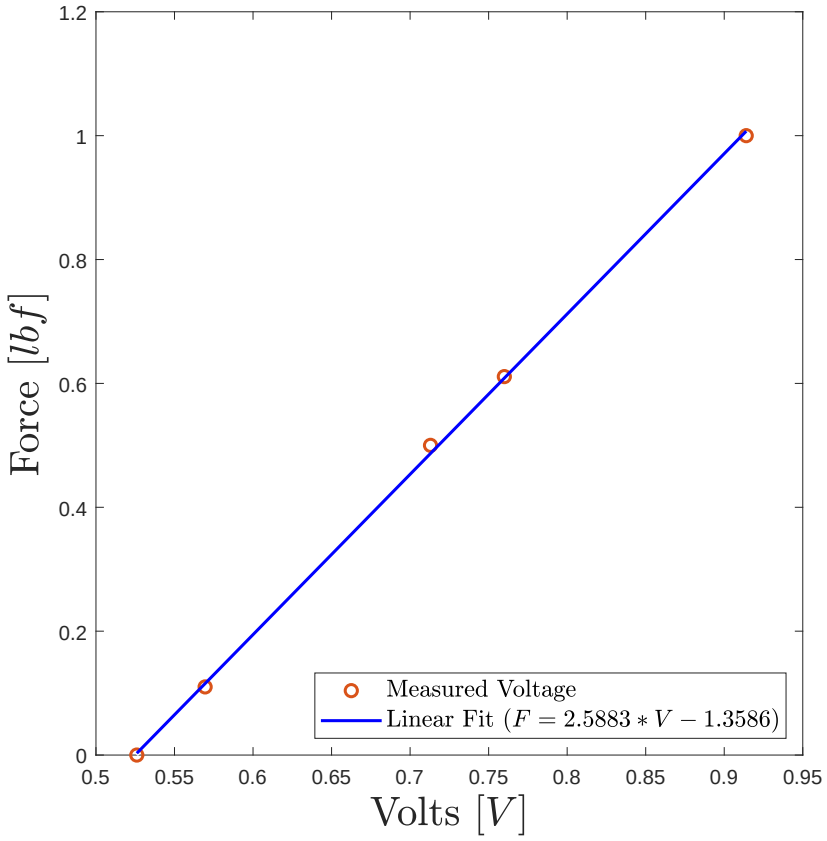


Figure B.2: Load cell calibration.

Script B.1: Arduino script to zero out the EEPROM

```
/*
 * EEPROM Clear
 * Sets all of the bytes of the EEPROM to 0.
 * Please see eeprom_iteration for a more in depth
 * look at how to traverse the EEPROM.
 * This example code is in the public domain.
 */
#include <EEPROM.h>
void setup() {
  // initialize the LED pin as an output.
  pinMode(13, OUTPUT);
  /***
    Iterate through each byte of the EEPROM storage.
    Larger AVR processors have larger EEPROM sizes, E.g:
    - Arduino Duemilanove: 512 B EEPROM storage.
    - Arduino Uno:       1 kB EEPROM storage.
    - Arduino Mega:      4 kB EEPROM storage.
    Rather than hard-coding the length, you should use the pre-provided length
        function.
    This will make your code portable to all AVR processors.
  ***/
  for (int i = 0 ; i < EEPROM.length() ; i++) {
    EEPROM.write(i, 0);
  }
  // turn the LED on when we're done
  digitalWrite(13, HIGH);
}
void loop() {
  /** Empty loop. **/
}
```

Script B.2: Arduino script to write on EEPROM

```
/*
 * EEPROM Write
 * Stores values read from analog input 1 into the EEPROM.
 * These values will stay in the EEPROM when the board is
 * turned off and may be retrieved later by another sketch.
 */
#include <EEPROM.h>
/** the current address in the EEPROM (i.e. which byte we're going to write to
    next) **/
int addr = 0;
void setup() {
  /** Empty setup. **/
}
void loop() {
  /***
    Need to divide by 4 because analog inputs range from
    0 to 1023 and each byte of the EEPROM can only hold a
    value from 0 to 255.
  ***/
  int val = analogRead(1) / 4;
  /***
    Write the value to the appropriate byte of the EEPROM.
    these values will remain there when the board is
    turned off.
  ***/
  EEPROM.write(addr, val);
  /***
    Advance to the next address, when at the end restart at the beginning.
    Larger AVR processors have larger EEPROM sizes, E.g:
    - Arduno Duemilanove: 512b EEPROM storage.
    - Arduino Uno:      1kb EEPROM storage.
```

```
    - Arduino Mega:     4kb EEPROM storage.
    Rather than hard-coding the length, you should use the pre-provided length
        function.
    This will make your code portable to all AVR processors.
  ***/
  addr = addr + 1;
  if (addr == EEPROM.length()) {
    addr = 0;

  }
  /***
    As the EEPROM sizes are powers of two, wrapping (preventing overflow) of an
    EEPROM address is also doable by a bitwise and of the length - 1.
    ++addr &= EEPROM.length() - 1;
  ***/
  delay(100); //in miliseconds
}
```

Script B.3: Arduino script to read bytes stored on EEPROM

```
#include <EEPROM.h>
// start reading from the first byte (address 0) of the EEPROM
int address = 0;
byte value;
void setup() {
  // initialize Serial and wait for port to open:
  Serial.begin(9600);
  while (!Serial) {
    ; // wait for Serial port to connect. Needed for native USB port only
  }
}
void loop() {
  // read a byte from the current address of the EEPROM
  value = EEPROM.read(address);

  Serial.print(address);
  Serial.print("\t");
  Serial.print(value, DEC);
  Serial.println();
  /***
    Advance to the next address, when at the end restart at the beginning.
    Larger AVR processors have larger EEPROM sizes, E.g:
    - Arduino Duemilanove: 512 B EEPROM storage.
    - Arduino Uno:       1 kB EEPROM storage.
    - Arduino Mega:      4 kB EEPROM storage.

    Rather than hard-coding the length, you should use the pre-provided length
        function.
    This will make your code portable to all AVR processors.
  ***/
  address = address + 1;
```

```
  if (address == EEPROM.length()) {
    address = 0;
  }
  /***
    As the EEPROM sizes are powers of two, wrapping (preventing overflow) of an
    EEPROM address is also doable by a bitwise and of the length - 1.

    ++address &= EEPROM.length() - 1;
  ***/
  delay(100); //in miliseconds
}
```

B.2 System Identification Results

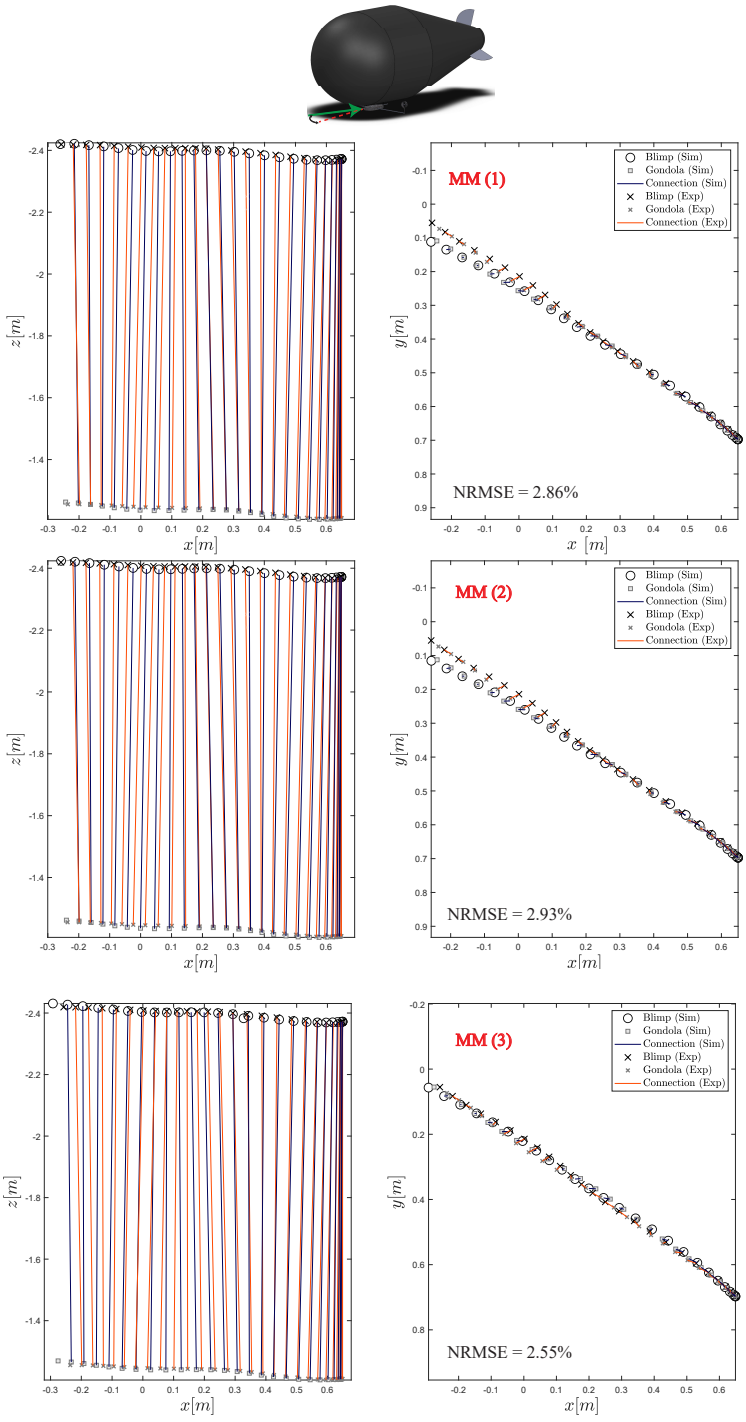


Figure B.3: Reconstruction of Experiment 1 using MM **(1)**, MM **(2)**, and MM **(3)**. The applied force is shown at top.

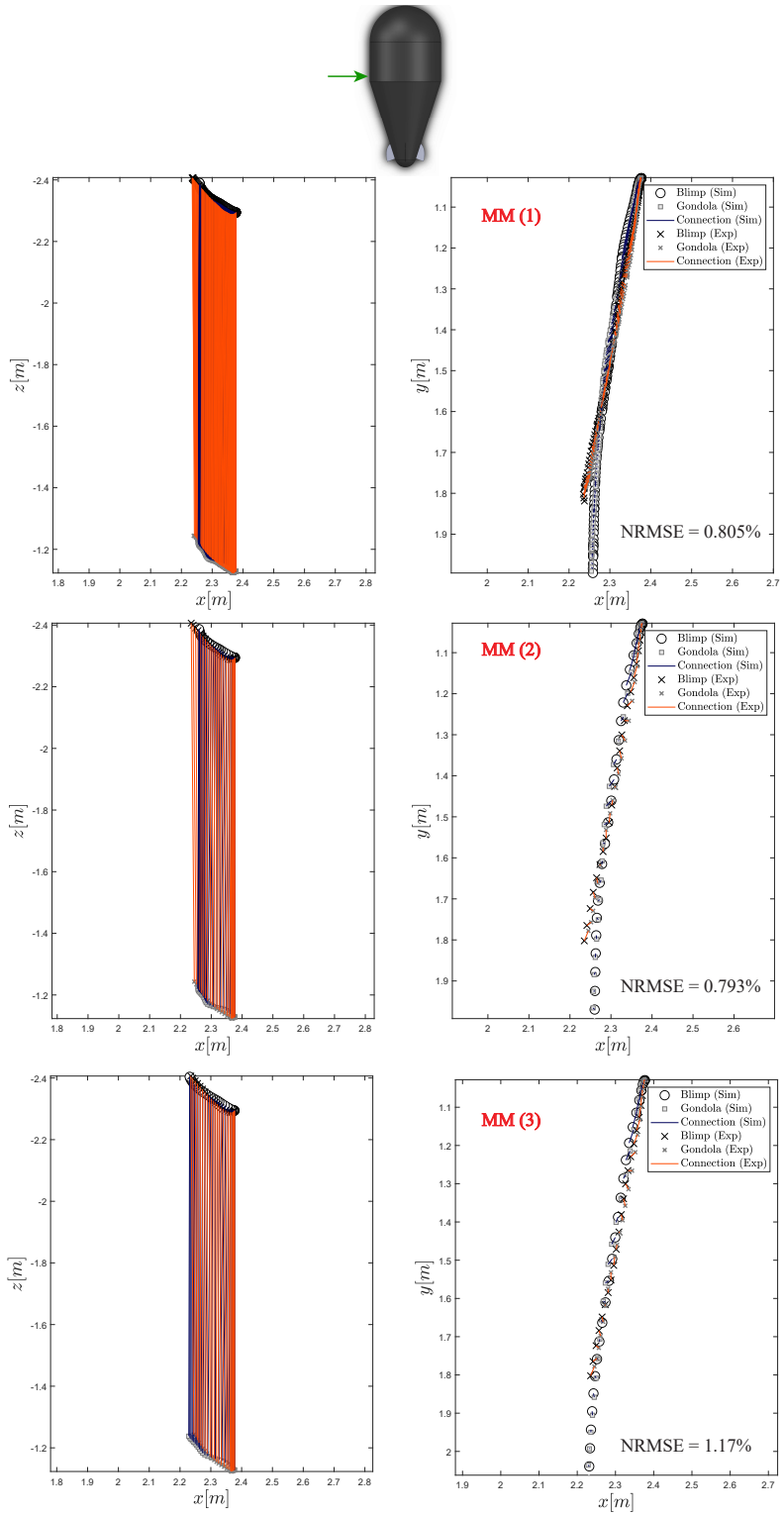


Figure B.4: Results obtained from Experiment 2 using MM **(1)**, MM **(2)**, and MM **(3)**. The applied force is shown at top.

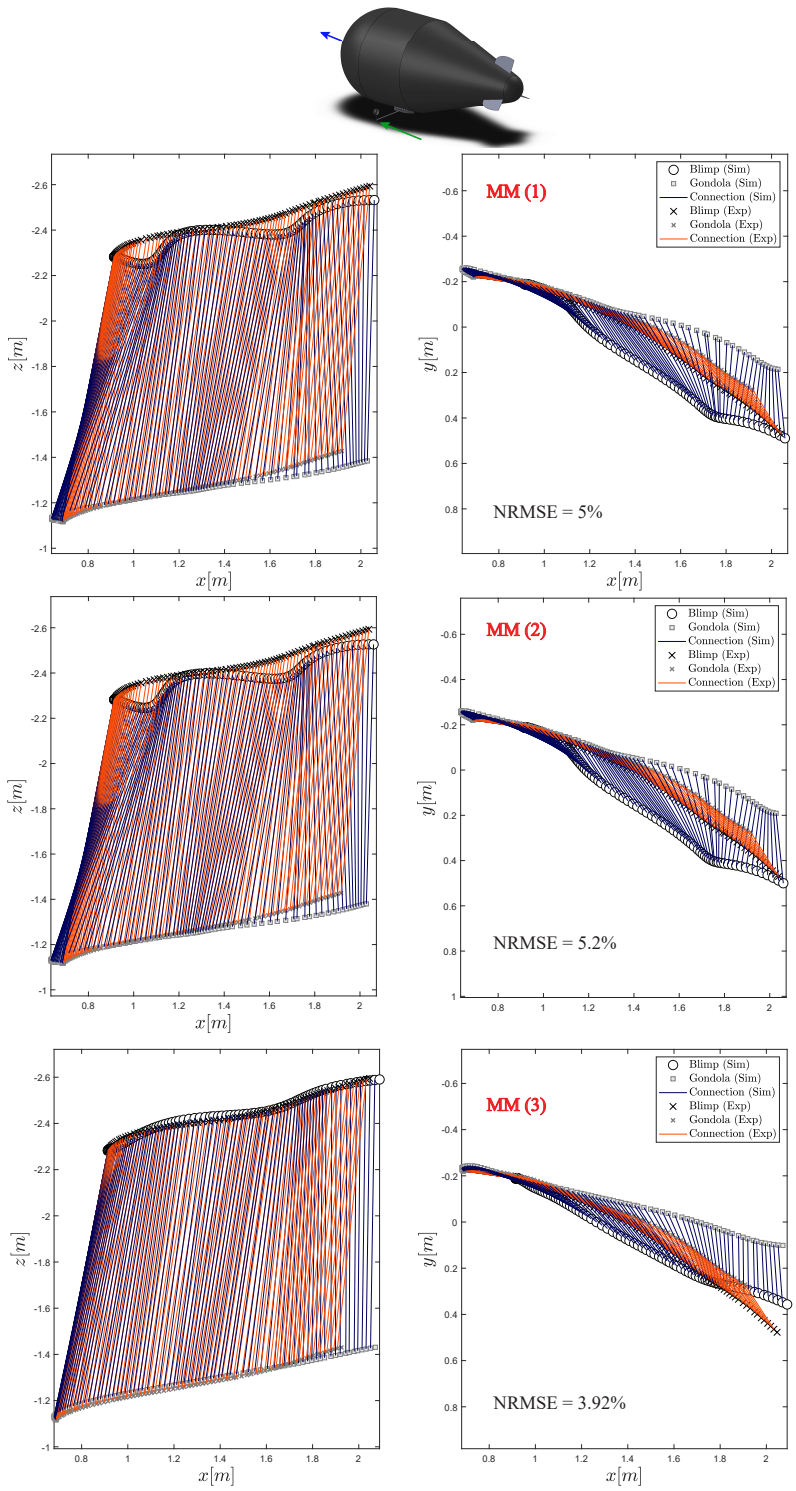


Figure B.5: Results obtained from Experiment 3 using MM **(1)**, MM **(2)**, and MM **(3)**. The applied force is shown at top where green and blue arrows represent force and initial velocity, respectively.

Appendix C

Aerodynamics

C.1 Aerodynamic Models of Airship - Mueller and Paluszek

Aerodynamic forces and coefficients used in C.1-C.6 [88]

$$A_X = q_0\{\ C_{X_1}\ \cos^2(\alpha)\cos^2(\beta) + C_{X_2}\sin(2\alpha)\sin(\frac{\alpha}{2})\ \} \tag{C.1}$$

$$\begin{aligned} A_Y = q_0\{\ & C_{Y_1}\ \sin(2\beta)\cos\left(\frac{\beta}{2}\right) + C_{Y_2}\ \sin(2\beta) \\ & + C_{Y_3}\ \sin(\beta)\sin|\beta| + C_{Y_4}\ (\delta_{RUDT} + \delta_{RUDB})\ \} \end{aligned} \tag{C.2}$$

$$\begin{aligned} A_Z = q_0\{\ & C_{Z_1}\ \sin(2\alpha)\cos(\frac{\alpha}{2}) + C_{Z_2}\sin(2\alpha) \\ & + C_{Z_3}\ \sin(\alpha)\sin|\alpha| + C_{Z_4}\ (\delta_{ELVL} + \delta_{ELVR})\ \} \end{aligned} \tag{C.3}$$

$$\begin{aligned} A_L = q_0\{\ & C_{L_1}\ (\delta_{ELVL} - \delta_{ELVR} + \delta_{RUDB} - \delta_{RUDT}) \\ & + C_{L_2}\sin(\beta)\sin|\beta|\ \} \end{aligned} \tag{C.4}$$

$$\begin{aligned} A_M = q_0\{\ & C_{M_1}\ \cos\left(\frac{\alpha}{2}\right)\sin(2\alpha) + C_{M_2}\sin(2\alpha) \\ & + C_{M_3}\sin(\alpha)\sin|\alpha| + C_{M_4}(\delta_{ELVL} + \delta_{ELVR})\} \end{aligned} \tag{C.5}$$

$$\begin{aligned} A_N = q_0\{\ & C_{N_1}\ \cos\left(\frac{\beta}{2}\right)\sin(2\beta) + C_{N_2}\sin(2\beta) \\ & + C_{N_3}\sin(\beta)\sin|\beta| + C_{N_4}(\delta_{ELVL} + \delta_{ELVR})\}\} \end{aligned} \tag{C.6}$$

where $C_{X_{i=1,2}}, C_{Y_{j=1\ldots4}}, C_{Z_j}, C_{L_i}, C_{M_j}, C_{N_j}$ correspond to the aerodynamic coefficients of the different bodies of the airship and are listed in Appendix C, β is the side-slip angle, and ξ

is the axial direction. The airship is assumed to have two rudders (top and bottom) and two elevators (left and right); The angles of the top and bottom rudders are δ_{RUDT} and δ_{RUDB}, respectively. Similarly, δ_{ELVL} and δ_{ELVR} are respectively left and right elevator angles.

$$C_{X_1} = -[(C_{D_{(\alpha=0)}})_h \; S_h + (C_{D_{(\alpha=0)}})_F \; S_F \; + (C_{D_{(\alpha=0)}})_g \; S_g] \tag{C.7}$$

$$C_{X_2} = (k_2 - k_1) \; \eta_k \; I_1 \; S_h \tag{C.8}$$

$$C_{Y_1} = C_{X_2} \tag{C.9}$$

$$C_{Y_2} = -0.5(C_{n\alpha})_F \; S_F \; \eta_F \tag{C.10}$$

$$C_{Y_3} = -[(C_{DC})_h \; J_1 \; S_h + (C_{DC})_F \; S_F \; + (C_{DC})_g \; S_g] \tag{C.11}$$

$$C_{Y_4} = -0.5(C_{n\delta})_F \; S_F \; \eta_F \tag{C.12}$$

$$C_{Z_1} = C_{X_2} \tag{C.13}$$

$$C_{Z_2} = C_{Y_2} = -0.5(C_{n\alpha})_F \; S_F \; \eta_F \tag{C.14}$$

$$C_{Z_3} = -[(C_{DC})_h \; J_1 \; S_h + (C_{DC})_F \; S_F] \tag{C.15}$$

$$C_{Z_4} = C_{Y_4} = -0.5(C_{n\delta})_F \; S_F \; \eta_F \tag{C.16}$$

$$C_{L_1} = (C_{n\delta})_F \; S_F \; \eta_F \; l_{f_3} \tag{C.17}$$

$$C_{L_2} = -(C_{DC})_g \; S_g \; l_{gz} \tag{C.18}$$

$$C_{M_1} = (k_2 - k_1) \; \eta_k \; I_3 \; S_h \; L \tag{C.19}$$

$$C_{M_2} = -0.5(C_{n\alpha})_F \; S_F \; \eta_F \; l_{f_1} \tag{C.20}$$

$$C_{M_3} = -[(C_{DC})_h \; J_2 \; S_h \; L + (C_{DC})_F \; S_F \; l_{f_2}] \tag{C.21}$$

$$C_{M_4} = -0.5(C_{n\delta})_F \; S_F \; \eta_F \; l_{f_1} \tag{C.22}$$

$$C_{N_j} = -C_{M_j} \tag{C.23}$$

where

$I_1 = \dfrac{1}{S_h} \int_0^{\ell_h} \frac{dS}{d\xi} d\xi$

$$I_3 = \frac{1}{S_h\ l_h} \int_0^{\ell_h} \xi\ \frac{dS}{d\xi} d\xi$$
$$J_1 = \frac{1}{S_h} \int_0^{\ell_h} 2R(\xi) d\xi$$
$$J_2 = \frac{1}{S_h\ l_h} \int_0^{\ell_h} 2R(\xi)\xi d\xi.$$

C.2 Aerodynamic Models of Airship - Recoskie

Aerodynamic model of airships, derived by Recoskie [89]

$$A_X = q_0\{\ C_{X_1}\ \cos^2(\alpha)\cos^2(\beta) + C_{X_2}\sin(2\alpha)\sin(\frac{\alpha}{2})\ \} \tag{C.24}$$

$$\begin{aligned} A_Y =& q_0\{\ C_{Y_1}\ \sin(2\beta)\cos\left(\frac{\beta}{2}\right) + C_{Y_2}\ \sin(2\beta) \\ &+ C_{Y_3}\ \sin(\beta)\sin|\beta|\ \} \end{aligned} \tag{C.25}$$

$$\begin{aligned} A_Z =& q_0\{\ C_{Z_1}\ \sin(2\alpha)\cos(\frac{\alpha}{2}) + C_{Z_2}\sin(2\alpha) \\ &+ C_{Z_3}\ \sin(\alpha)\sin|\alpha|\ \} \end{aligned} \tag{C.26}$$

$$A_L = q_0\{\ -C_{L_2}\sin(\beta)\sin|\beta|\ \} + \frac{1}{2}\rho C_{L_{\dot{\phi}}}\dot{\phi}|\dot{\phi}| \tag{C.27}$$

$$\begin{aligned} A_M =& q_0\{\ C_{M_1}\ \cos\left(\frac{\alpha}{2}\right)\sin(2\alpha) + C_{M_2}\sin(2\alpha) \\ &+ C_{M_3}\sin(\alpha)\sin|\alpha|\ \} + \frac{1}{2}\rho C_{M_{\dot{\theta}}}\dot{\theta}|\dot{\theta}| \end{aligned} \tag{C.28}$$

$$\begin{aligned} A_N =& q_0\{\ C_{N_1}\ \cos\left(\frac{\beta}{2}\right)\sin(2\beta) + C_{N_2}\sin(2\beta) \\ &+ C_{N_3}\sin(\beta)\sin|\beta|\} + \frac{1}{2}\rho C_{N_{\dot{\psi}}}\dot{\psi}|\dot{\psi}| \end{aligned} \tag{C.29}$$

$$C_{L_{\dot{\phi}}} = -2(C_{DC})_F\ S_F\ d_{f,z}^3 \tag{C.30}$$

$$C_{M_{\dot{\theta}}} = -[(C_{DC})_F\ S_F\ d_{f,x2}^3 + \frac{l_{v,z}l_{v,x}^4}{240}] \tag{C.31}$$

$$C_{N_{\dot{\psi}}} = C_{M_{\dot{\theta}}} = -[(C_{DC})_F\ S_F\ d_{f,x2}^3 + \frac{l_{v,z}l_{v,x}^4}{240}] \tag{C.32}$$

Demonstration of wake-vortex asymmetry on a slender body at high angles of attack.

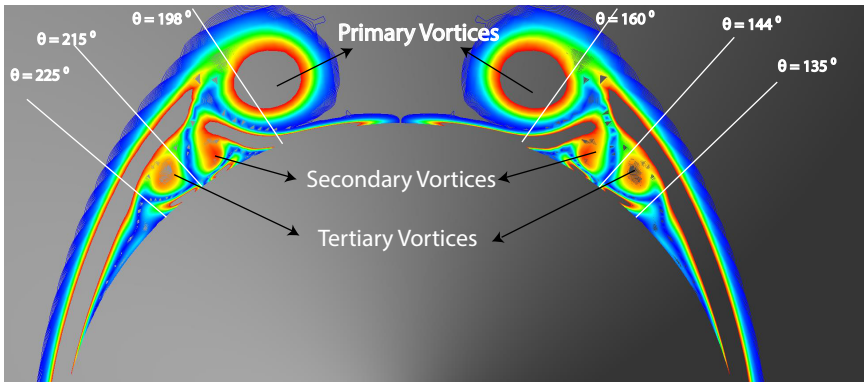


Figure C.1: Vortices developed on the leeward side of a slender body. Obtained from [173]

C.3 Aerodynamic Coefficients From CFD

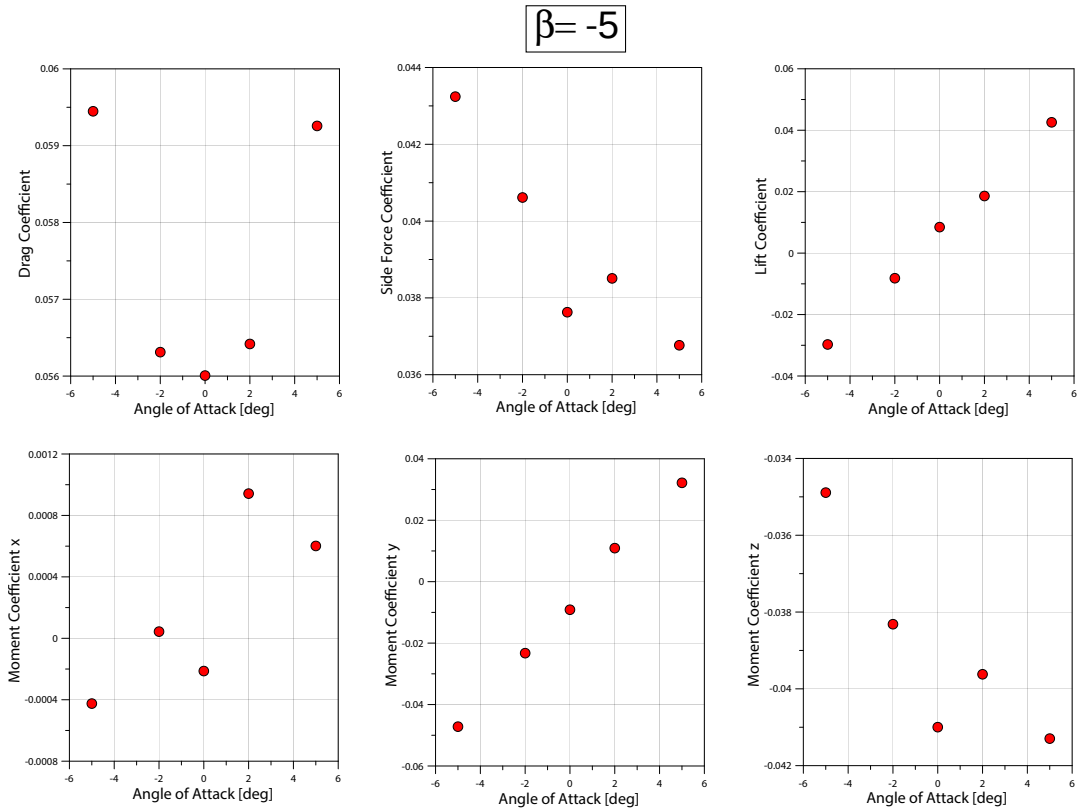


Figure C.2: Comparison of aerodynamic coefficients at $-5° \leq \alpha \leq 5°$ and $\beta = -5°$.

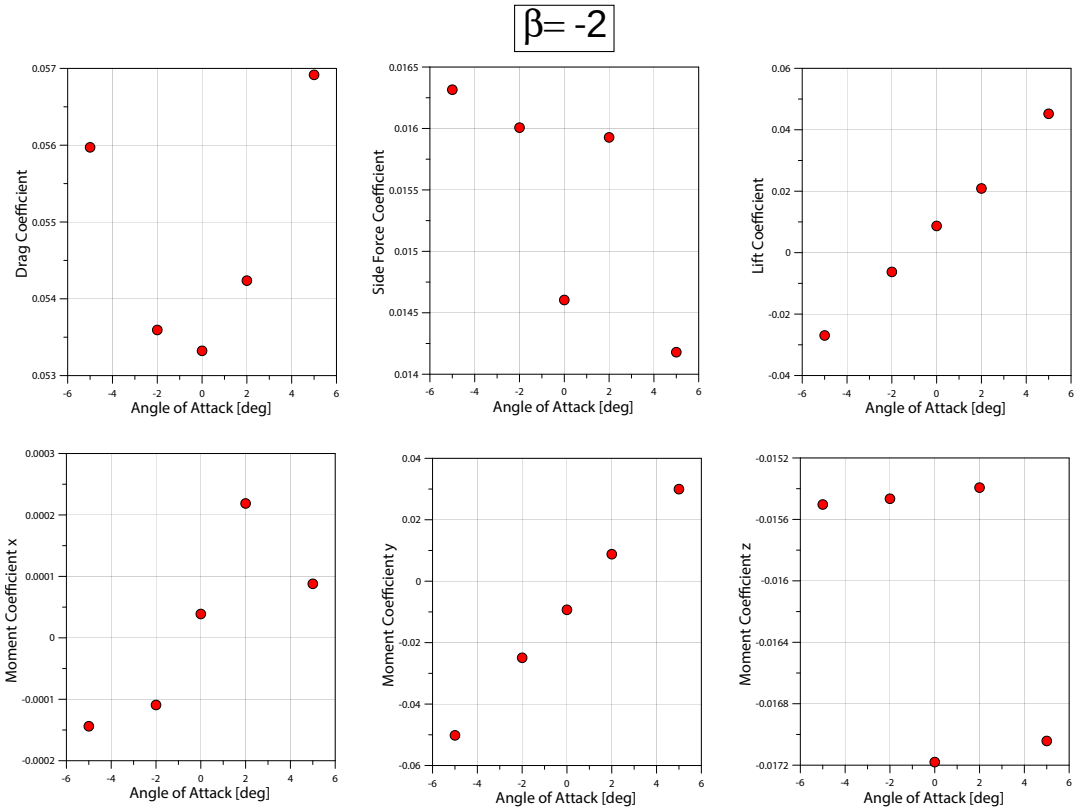


Figure C.3: Comparison of aerodynamic coefficients at $-5° \leq \alpha \leq 5°$ and $\beta = -2°$.

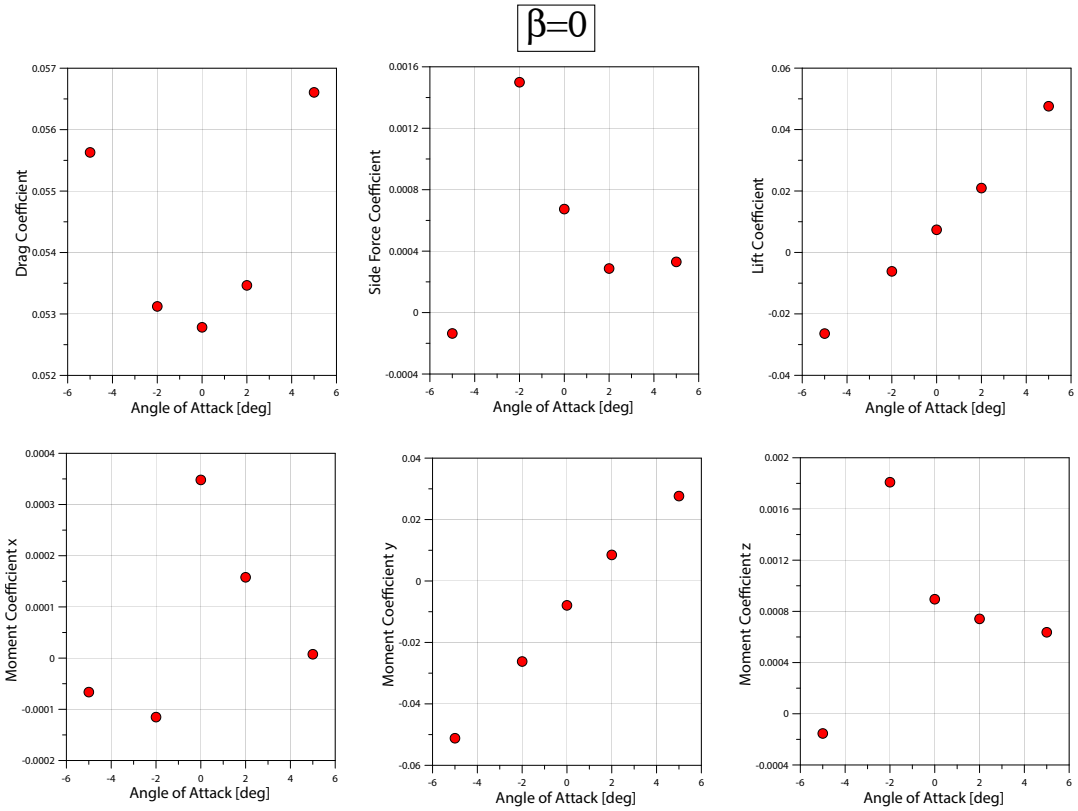


Figure C.4: Comparison of aerodynamic coefficients at $-5° \leq \alpha \leq 5°$ and $\beta = 0°$.

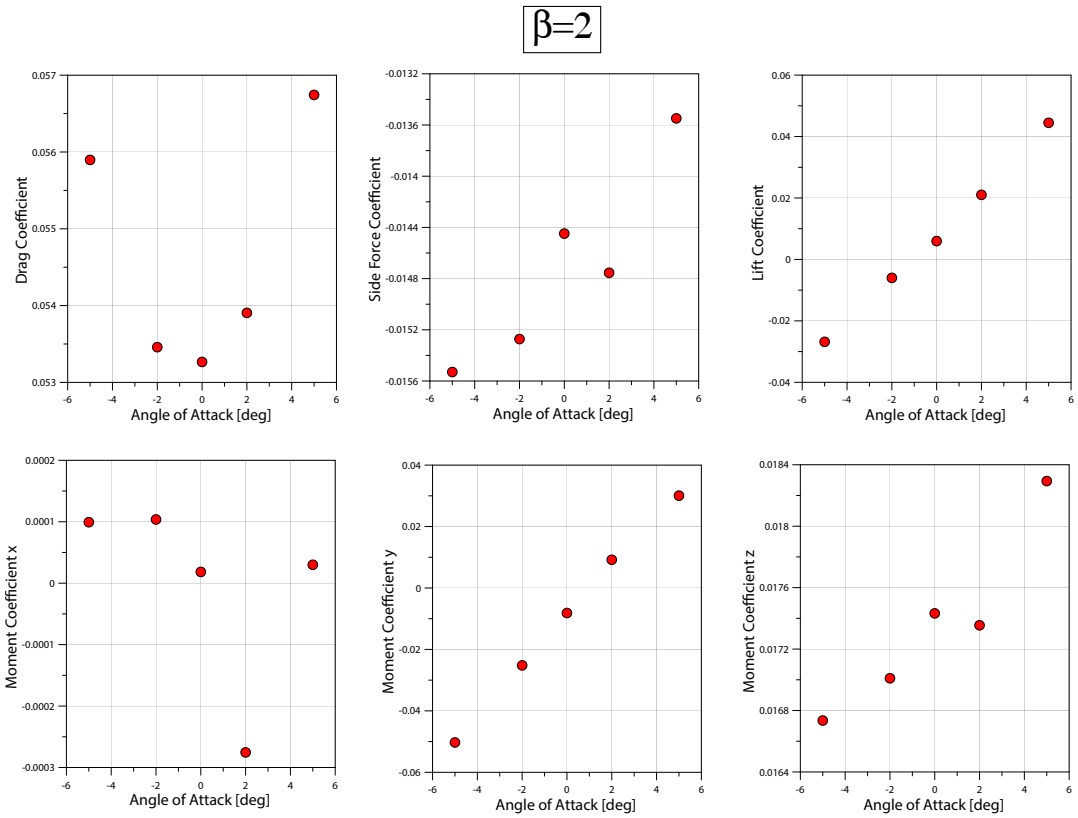


Figure C.5: Comparison of aerodynamic coefficients at $-5° \leq \alpha \leq 5°$ and $\beta = 2°$.

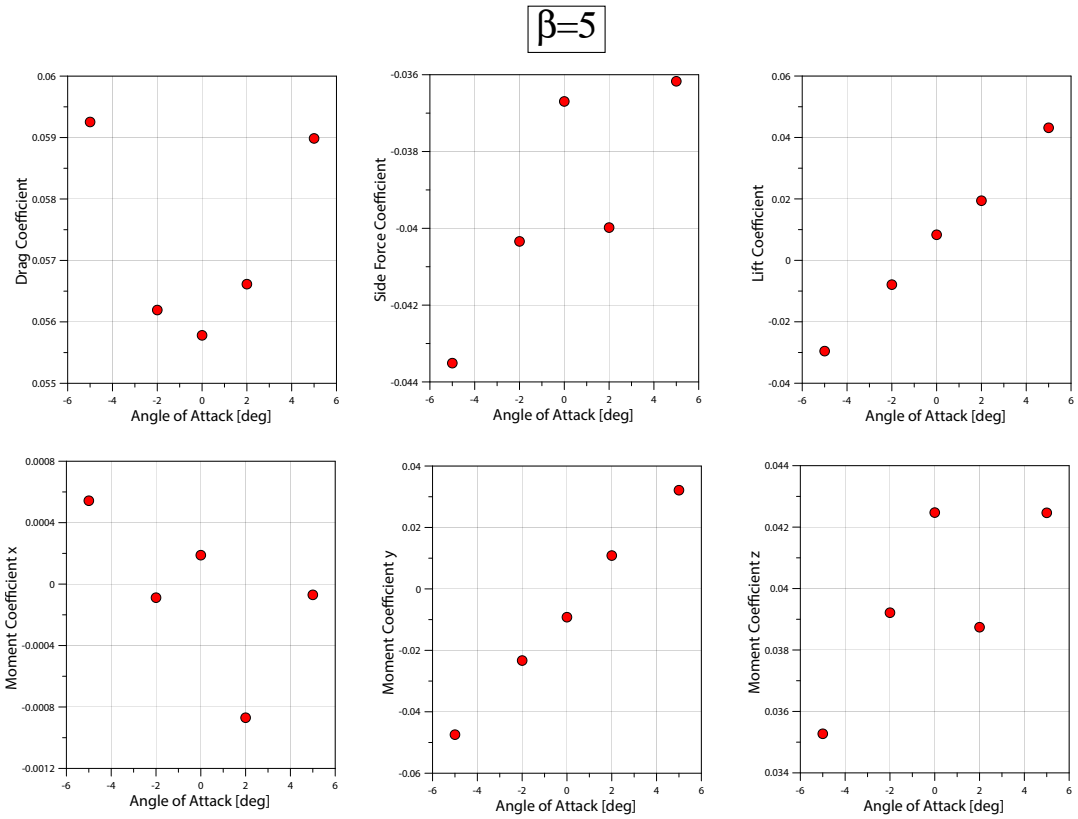


Figure C.6: Comparison of aerodynamic coefficients at $-5^\circ \leq \alpha \leq 5^\circ$ and $\beta = 5^\circ$.

Appendix D

Control Strategy

D.1 General Description of Fuzzy Logic Control

Fuzzy logic control (FLC) provides a relationship between inputs and outputs via interpolation in vague environment (fuzzy sets). The control rules are expressed verbally through linguistic variables. A fuzzy controller consists of four processes, fuzzification, rule base, inference, and defuzzification. However, it is often desirable to pre-process the inputs by filtering out noise and normalizing the values to standard ranges. Similarly, the output values can be post-processed by re-scaling the normalized values to physical units. Figure D.1 illustrates the building blocks of a fuzzy controller.

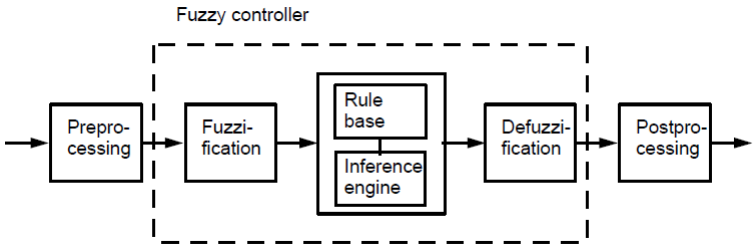


Figure D.1: Fuzzy control scheme, adopted from [174]

In the fuzzification process, the input values are converted from numbers (crisp value) to degrees of membership functions (MF), μ. Hence, every input value is represented in

terms of the degree of membership of each linguistic term (syntax rule). An example will be provided after defining the rest of the processes.

It is necessary to have a set of semantic rules dictating the relationship between the inputs and the outputs, referred to as rule base. The semantic rules are often described in if(condition)-then(conclusion) form. Connectives are used, such as **and**/**or**, when multiple inputs are used. In fuzzy logic, **and** is prominently treated as **minimize** or algebraic product of MF. On the other hand, **or** acts as **maximize** or probabilistic sum. For example,

$$\textbf{A and B} = \textbf{min(A,B)}$$

$$\textbf{A or B} = \textbf{max(A,B)}$$

Alternatively

$$\textbf{A and B} = \textbf{A} \times \textbf{B}$$

$$\textbf{A or B} = \textbf{A} + \textbf{B} - \textbf{A} \times \textbf{B}$$

where **A** and **B** are degrees of MF. Moreover, MF quantify the belongingness of an input to the fuzzy set. A fuzzy set may contain multiple MF. Membership functions exist in many shapes, most commonly singleton[1], triangular, or Gaussian. Some of the commonly used MF shapes are illustrated in Fig.D.2. For more MF shapes, readers are advised to look at [156]. It is worth noting that the set of rules and shape of MF are determined by expert knowledge and/or trial and error.

The decision-making process in fuzzy logic is referred to as inferencing. The inference engine consists of two operations: implication and aggregation. The most common inferencing methods are Mamdani[158] and Sugeno[175]. The implication of a rule is the "likelihood" of that rule to occur. Implication serves as a scaling factor to the firing strength of a rule. Furthermore, aggregation[2] is the process of combining multiple inputs using **maximum** or **sum** operations. The aggregation operation is identical to the fuzzification if a single input is used.

[1]Singleton MF are mainly used in the fuzzy sets of an output and it has the shape of an impulse function.
[2]Also called accumulation in some references

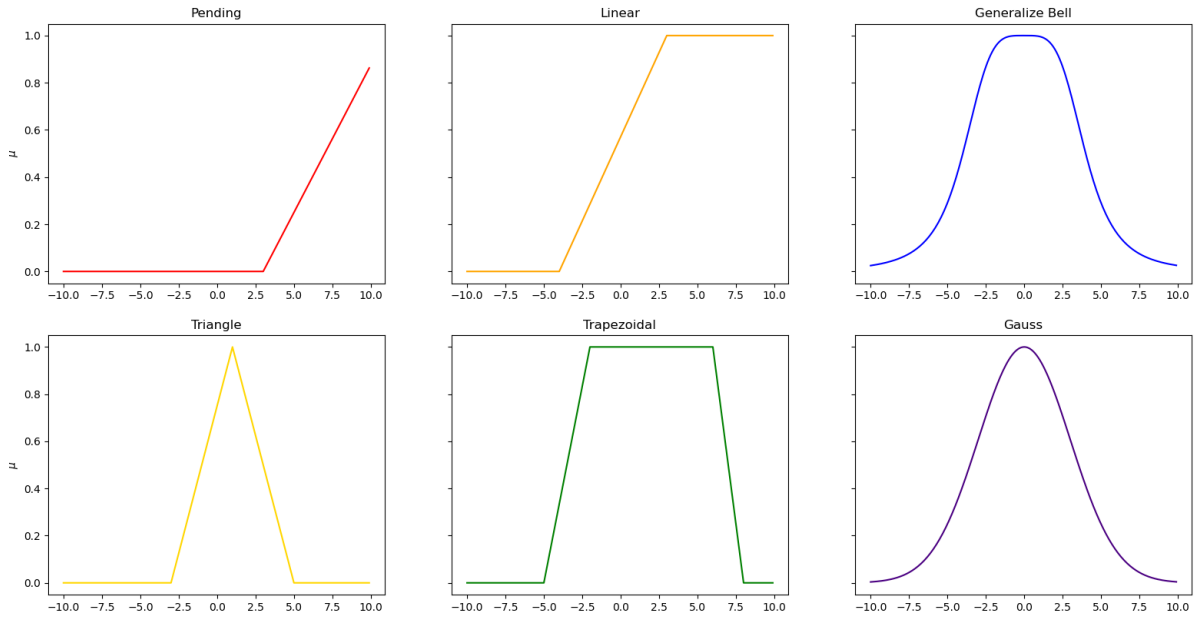


Figure D.2: Membership functions used in rule base process

The final stage in a fuzzy controller is defuzzification where the fuzzy set is converted to a single crisp output value. Defuzzification can be carried out using various approaches. The most common approach uses the centre of gravity under the fuzzy set, given in the following form

$$u = \frac{\int_{\xi} \mu(\xi)\xi}{\int_{\xi} \mu(\xi)} \tag{D.1}$$

where u is the crisp output value, ξ is the fuzzy output value (a point in the universe of discourse), and $\mu(\xi)$ is the degree of membership of the fuzzy output value (ξ). Equation (D.1) is often presented in discretized form (D.2)

$$u = \frac{\sum_i^N S_i \; CG_i}{\sum_i^N S_i} \tag{D.2}$$

where S_i is the area of discretized region i, CG_i is the distance along the control axis (fuzzy output axis) to the centre of gravity of region i, and N is the total number of discretized regions.

D.2 Membership Functions and Rules for Stationary Payload FLC

The distribution of membership functions for the inputs of the stationary payload FLC is illustrated in Fig. D.3 and Fig. D.4 for the axial direction, while the corresponding MFs for the lateral direction are presented in Fig. D.5 and Fig. D.6. The midpoints of the N and P membership functions were adjusted for all inputs, with some undergoing more pronounced shifts than others. In contrast, the endpoints of the N and P functions remained unchanged. However, the endpoints of the zero membership function (MF) were slightly modified, except for the payload angle error along the X-axis and its rate of change.

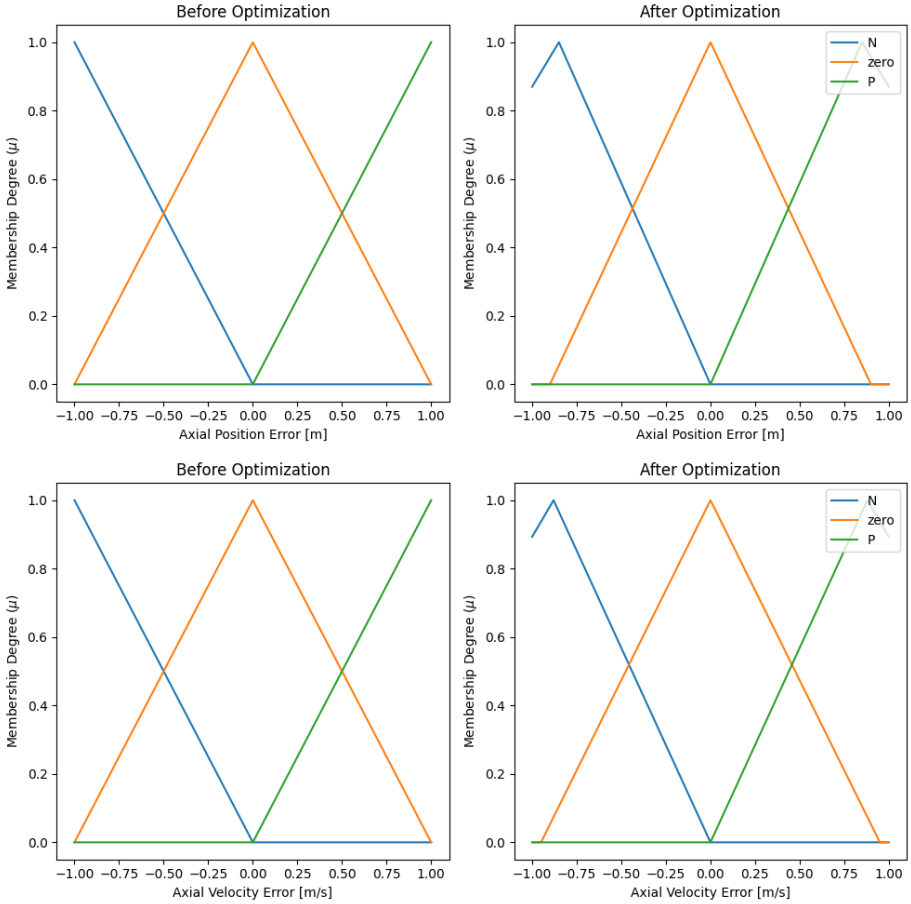


Figure D.3: Membership function distribution for axial position error and its rate of change before and after optimization.

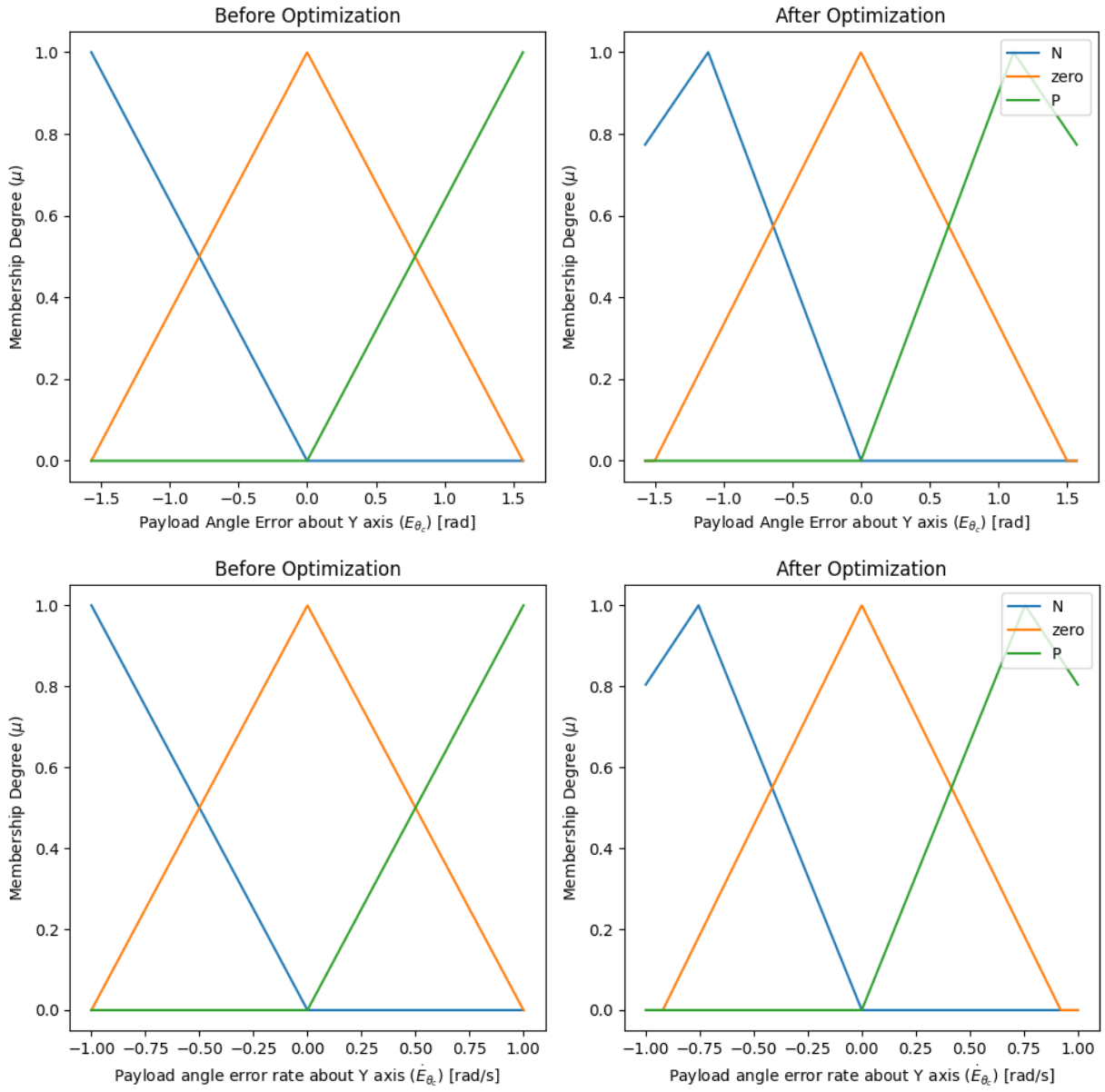


Figure D.4: Membership function distribution for payload angle error and its rate of change about Y axis before and after optimization.

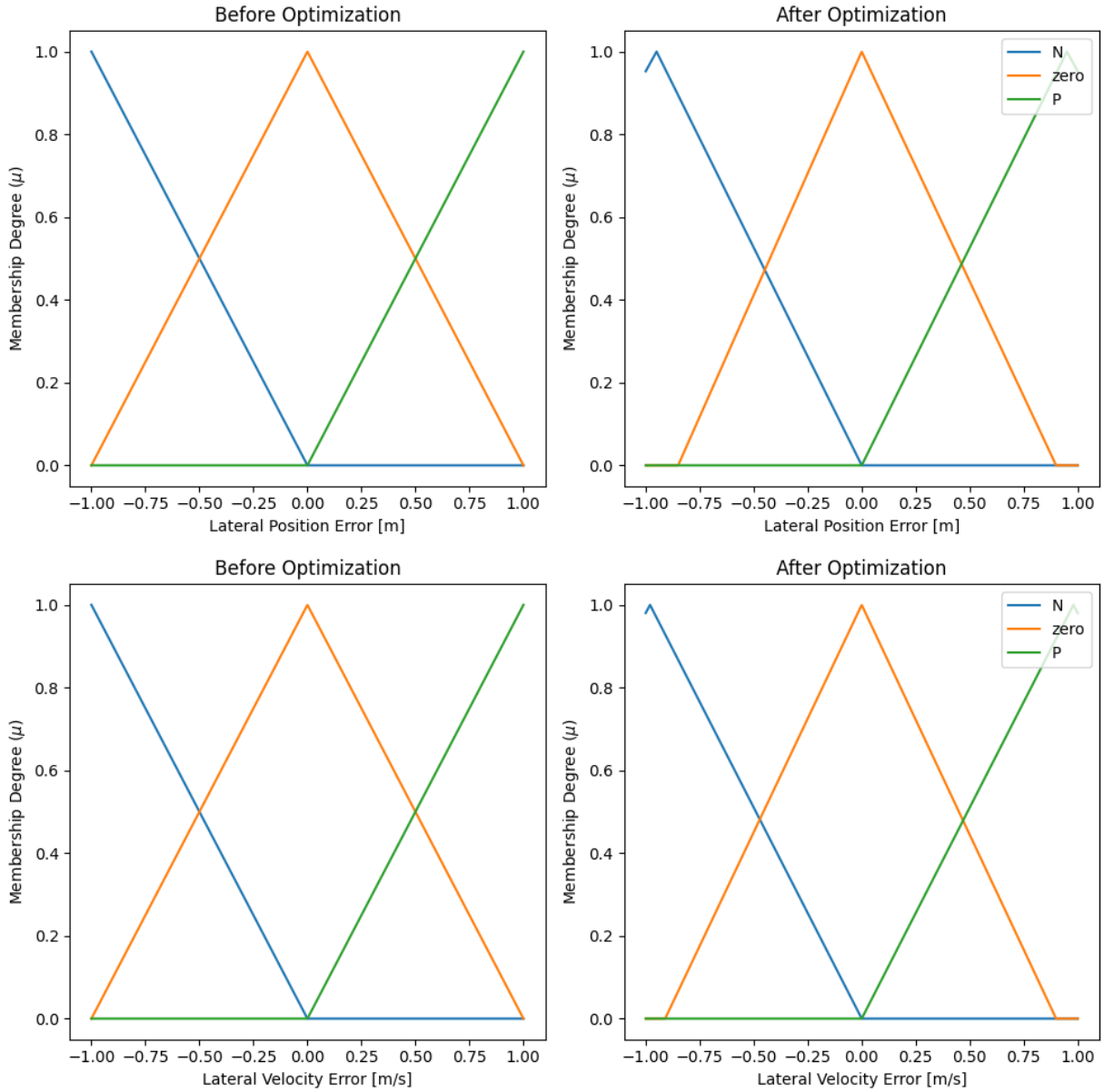


Figure D.5: Membership function distribution for lateral position error and its rate of change before and after optimization.

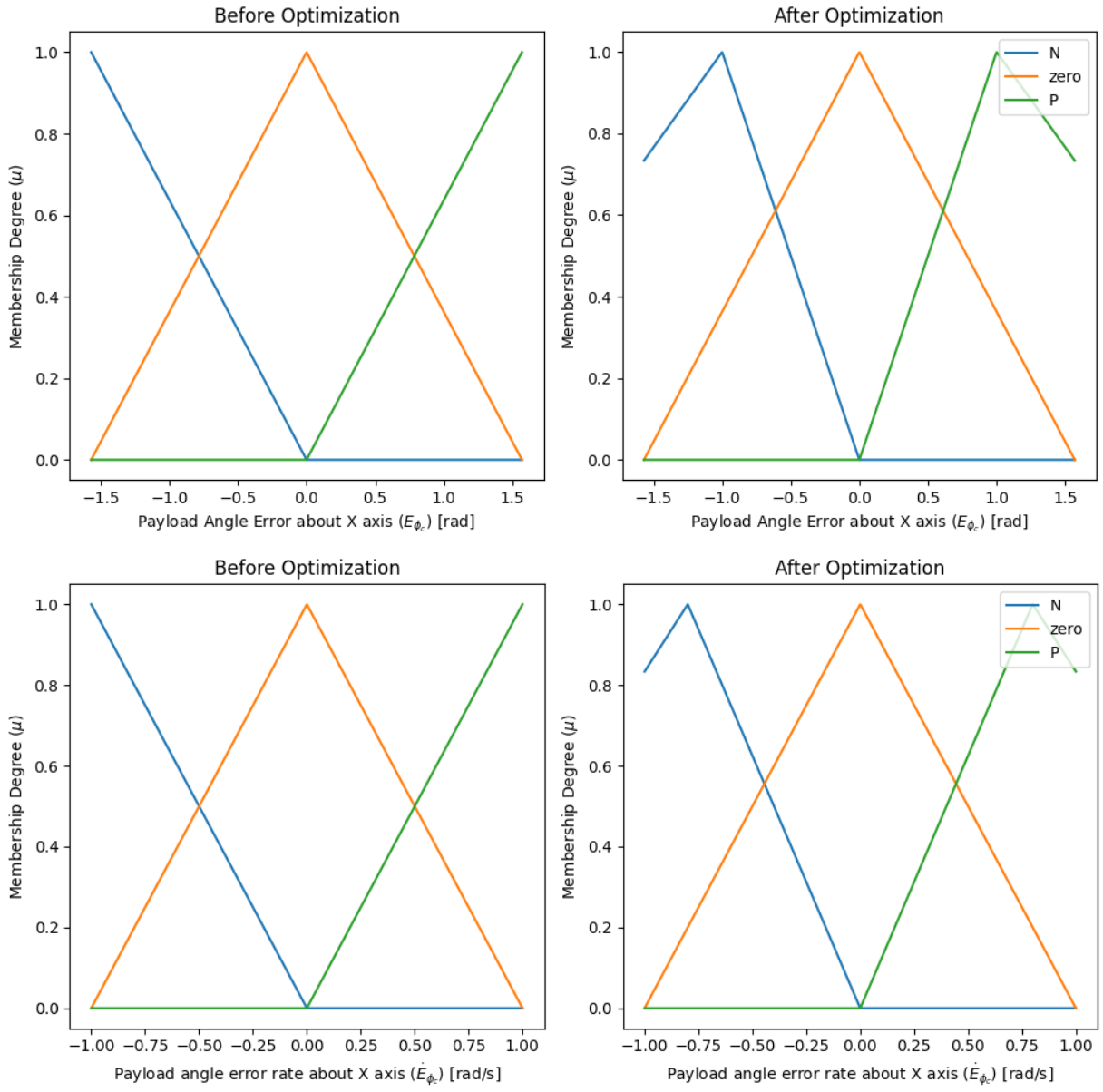


Figure D.6: Membership function distribution for payload angle error and its rate of change about X axis before and after optimization.

Since there are 4 inputs, each with 3 MF, the number of rules can be calculated using this simple formula ($nMF^{\,inputs} = 3^4 = 81$) where nMF is the number of MF. A more general formula, that can be used when the number of MF varies among the inputs, is ($nMF_1 \times nMF_2 \times ...$) where the inputs are iterated in the subscript. Rules for payload FLC along the axial direction are presented in Table D.1. The same rules apply for the lateral

direction.

Table D.1: Rules for payload FLC along the axial direction. Membership functions P, Z, and N are represented with 1, 0, and -1, respectively.

	$\dot{E}_{\theta_c}$	1	1	1	0	0	0	-1	-1	-1
	E_{θ_c}	1	0	-1	1	0	-1	1	0	-1
$\dot{E}_{x_L}$	E_{x_L}									
1	1	0	-1	-2	-1	-2	-3	-2	-3	-4
1	0	1	0	-1	0	-1	-2	-1	-2	-3
1	-1	2	1	0	1	0	-1	0	-1	-2
0	1	1	0	-1	0	-1	-2	-1	-2	-3
0	0	2	1	0	1	0	-1	0	-1	-2
0	-1	3	2	1	2	1	0	1	0	-1
-1	1	2	1	0	1	0	-1	0	-1	-2
-1	0	3	2	1	2	1	0	1	0	-1
-1	-1	4	3	2	3	2	1	2	1	0

D.3 Membership Functions and Rules for Navigation FLC

The membership function distributions for navigation controller inputs are illustrated in Fig D.7 before and after optimization. The optimization process resulted in shifting the midpoints of PP, PPP, and PPPP MFs. This suggests that the optimizer reassigned error values above 0.75 m and 0.15 m/s, respectively for $E_{|\vec{r}|}$ and $\dot{E}_{|\vec{r}|}$, to different fuzzy categories while keeping the overall structure unchanged. Similarly, the midpoints for N and P MFs were altered for yaw error and its rate of change, as depicted in Fig D.8. The optimized decision-making thresholds were found to enhance the control performance, as determined by the cost function over the simulation time.

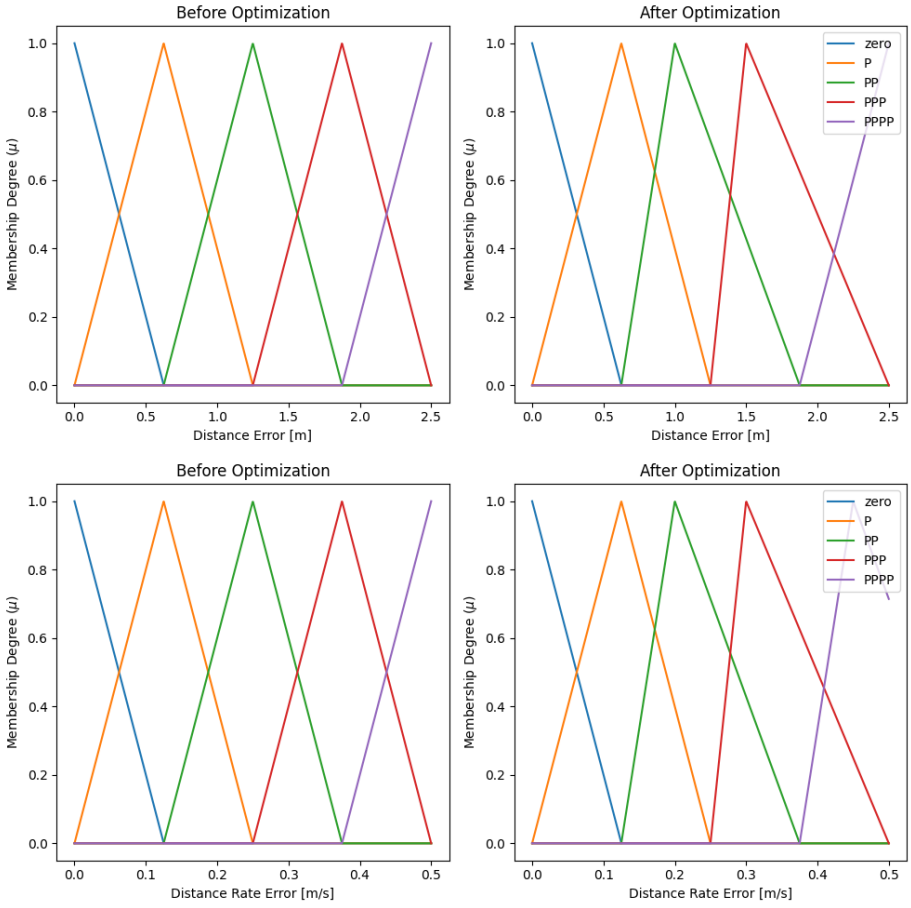


Figure D.7: Membership function distribution for distance and its rate of change error before and after optimization.

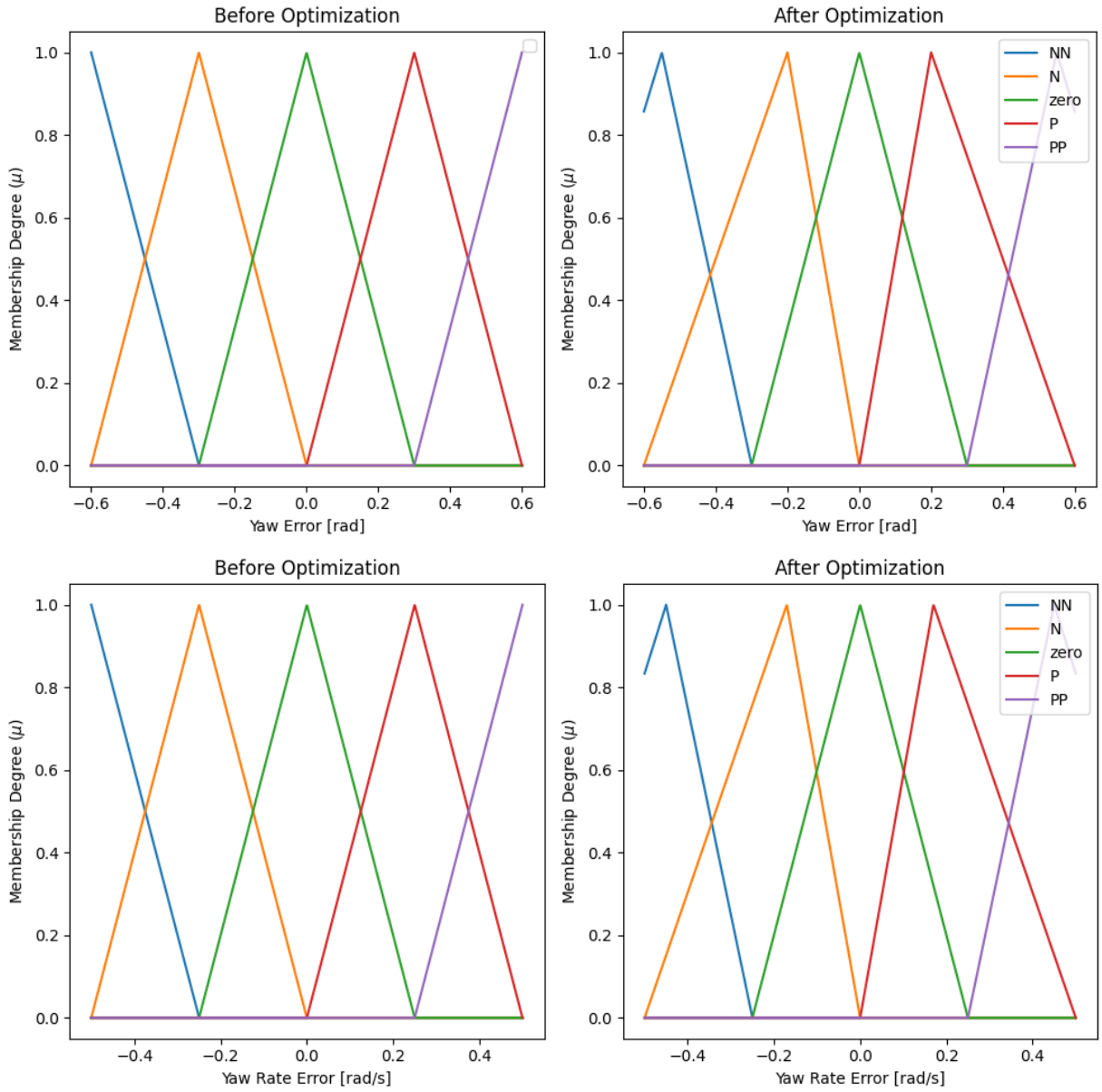


Figure D.8: Membership function distribution for yaw and yaw rate error before and after optimization.

The rules for thrust FLC are presented in Table D.2 which shows two inputs: the distance error ($E_{|\vec{r}|}$) and its rate of change ($\dot{E}_{|\vec{r}|}$). The rules for the yaw controller with E_{ψ_B} and $\dot{E}_{\psi_B}$ as inputs are given in Table D.3.

Table D.2: Rules for thrust FLC. Membership functions Z, P, PP, PPP, and PPPP are represented with 0, 1, 2, 3, and 4, respectively, all the way to 8.

	$E_{\vert\vec{r}\vert}$	0	1	2	3	4
$\dot{E}_{\vert\vec{r}\vert}$						
0		0	1	2	3	4
1		1	2	3	4	5
2		2	3	4	5	6
3		3	4	5	6	7
4		4	5	6	7	8

Table D.3: Rules for yaw FLC. Membership functions PP, P, Z, N, NN are represented with 2, 1, 0, and -1, -2, respectively.

	E_{ψ}	-2	-1	0	1	2
$\dot{E}_{\psi}$						
-2		4	3	2	1	0
-1		3	2	1	0	-1
0		2	1	0	-1	-2
1		1	0	-1	-2	-3
2		0	-1	-2	-3	-4

D.4 Membership Functions and Rules for Payload Delivery FLC

Figure D.9 illustrates the membership function distributions for the payload delivery controller's inputs both before and after optimization. As a result of the optimization process, the midpoints of the NN, N, P, and PP membership functions were adjusted.

The rules for payload delvery FLC are presented in Table D.4 which shows two inputs: the distance error ($E_{|\vec{r}|}$) and its rate of change ($\dot{E}_{|\vec{r}|}$). The same rules applies for the yaw controller with E_{ψ_B} and $\dot{E}_{\psi_B}$ as inputs.

Table D.4: Rules for payload delivery FLC. Membership functions PP, P, Z, N, NN are represented with 2, 1, 0, and -1, -2, respectively.

	E_{z_B}	-2	-1	0	1	2
$\dot{E}_{z_B}$						
-2		4	3	2	1	0
-1		3	2	1	0	-1
0		2	1	0	-1	-2
1		1	0	-1	-2	-3
2		0	-1	-2	-3	-4

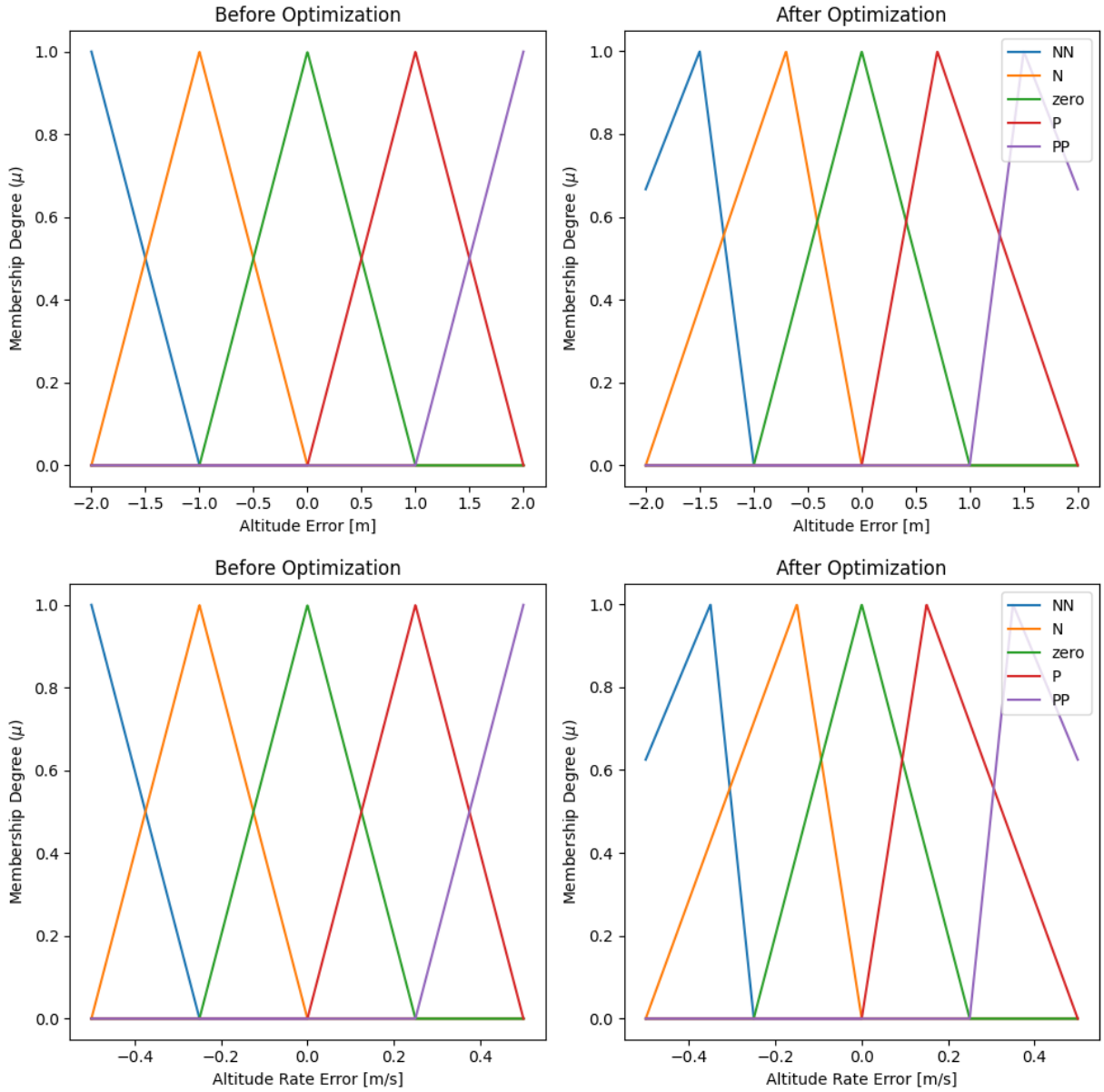


Figure D.9: Membership function distribution for altitude and altitude rate errors before and after optimization.

D.5 Calculating the Axial Payload Position

Since the GPS can only provide the position of the gondola, the axial position of the slung payload, in the inertial frame, can be approximated using the following kinematic

relationship:

$$x_L = x_G + |\vec{L}|\ \cos\psi_B \cos\theta_B \sin\theta_c$$
$$+|\vec{L}|\ (\cos\psi_B \cos\phi_B \sin\theta_B + \sin\psi_B \sin\phi_B) \cos\theta_c \tag{D.3}$$

where $|L|$ is the tether length and θ_c is the tether angle in $X - Z$ plane. Note that (D.3) is only valid when the tether is taught.

D.6 Modelling of Thrusters Force

The force generated by the thrusters was modelled using a propeller thrust stand [157]. This device measures the forces produced in static conditions as the PWM signal is varied. Consequently, a model illustrating the relationship between force and PWM signal was developed by altering the PWM signal and observing the corresponding force output. Since the force generated by a given PWM signal can vary with battery voltage, meaning that the same PWM signal can yield different forces at different voltage levels, it was necessary to repeat those tests at various battery voltages. It is also important to note that the thrusters exhibit directional preference, meaning they produce higher thrust in one direction over the other. This requires separate models for positive and negative thrust directions. Reversible ESCs (Electronic Speed Controllers) were utilized to rotate the thrusters in the less preferred direction. Data collected, along with polynomial fit models for thrust versus PWM, are shown in Fig. D.10 - Fig. D.13 at two selected voltages. The selected voltages cover the majority of the operating vehicle range. Figures D.10 and D.10 illustrate the thrust generated by a single forward thruster in both the preferred (positive) and least preferred (negative) directions. Similarly, Fig. D.12 and Fig. D.13 illustrate the thrust-PWM relationship of a single lateral thruster under the same conditions.

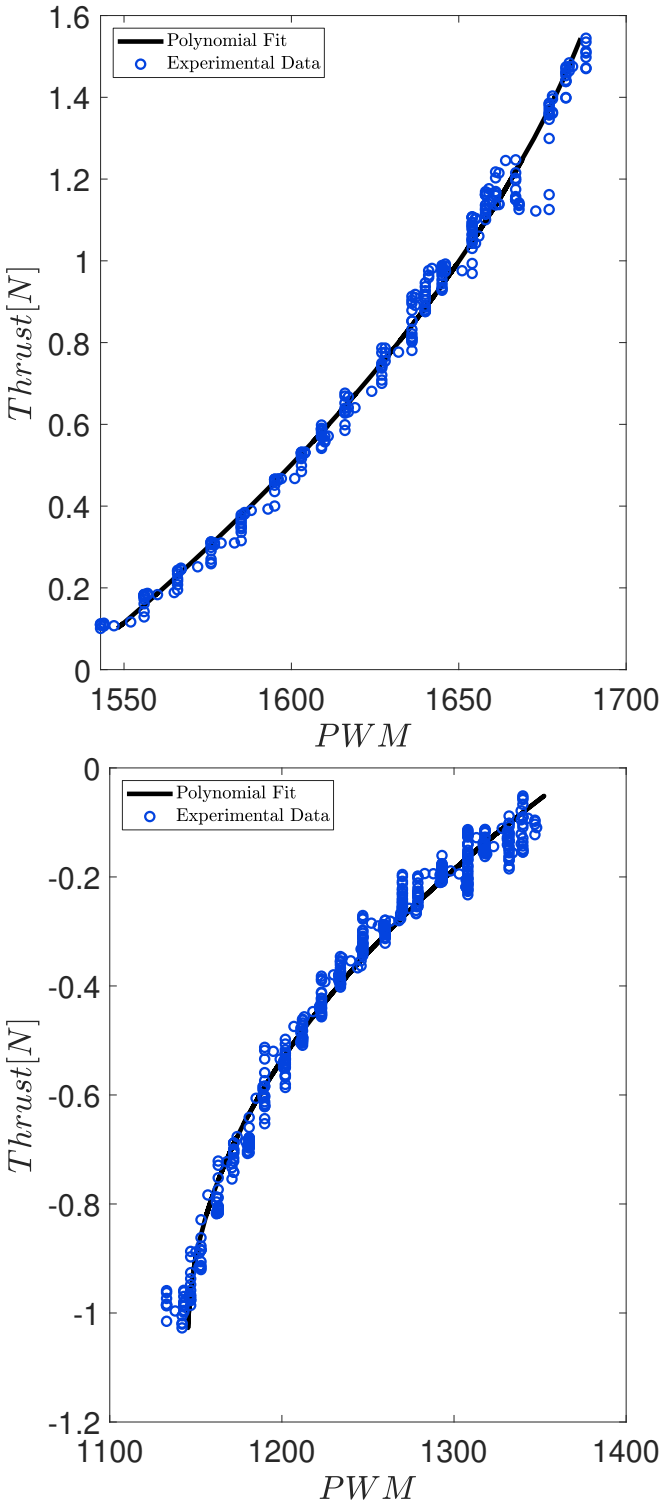


Figure D.10: Thrust as a function of PWM signal at 15.5 volts for a single forward thruster.

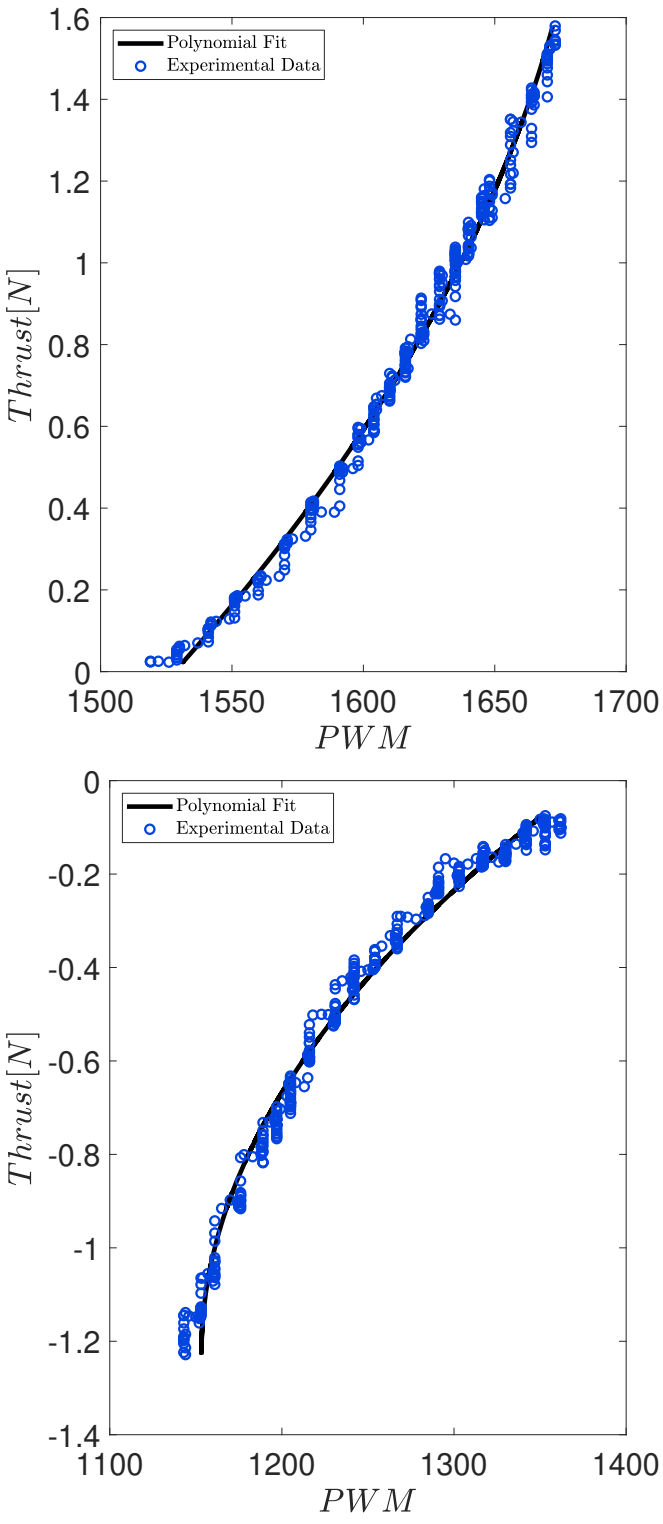


Figure D.11: Thrust as a function of PWM signal at 16.7 volts for a single forward thruster.

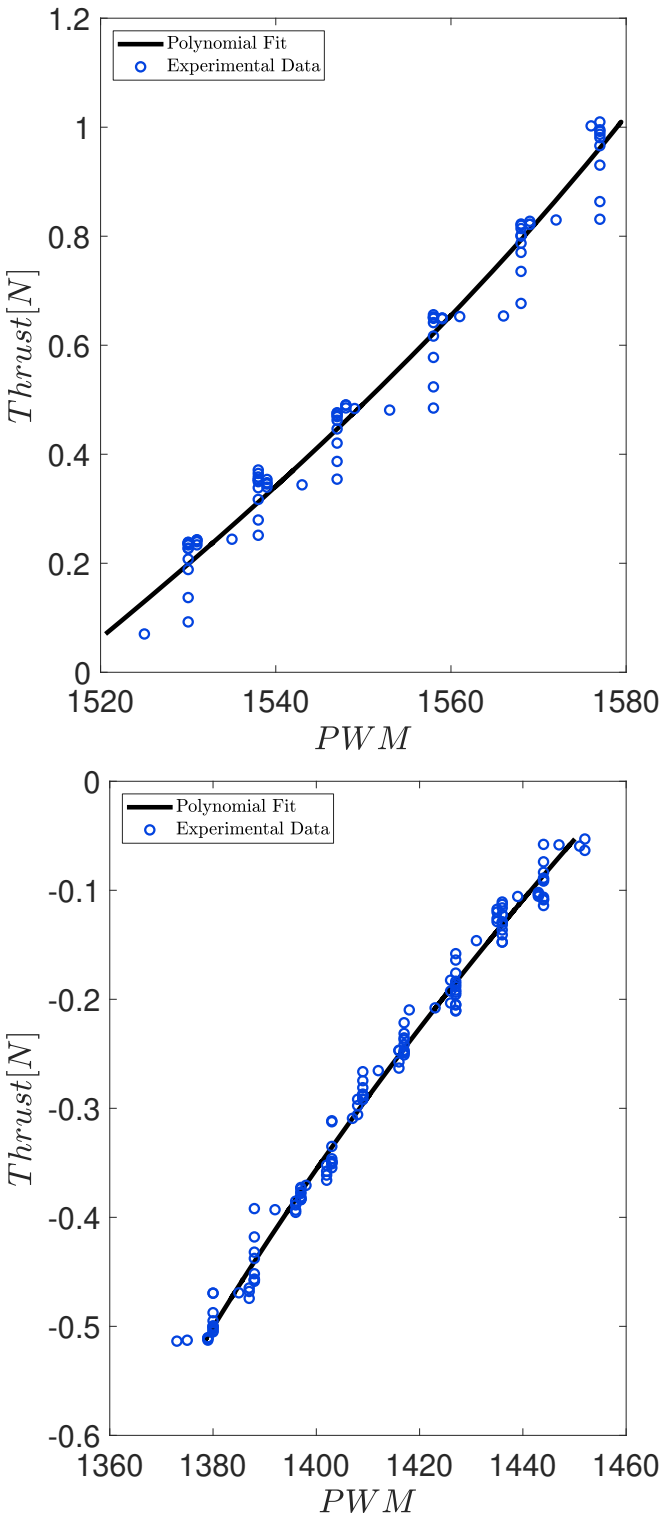


Figure D.12: Thrust as a function of PWM signal at 15.5 volts for a single lateral thruster.

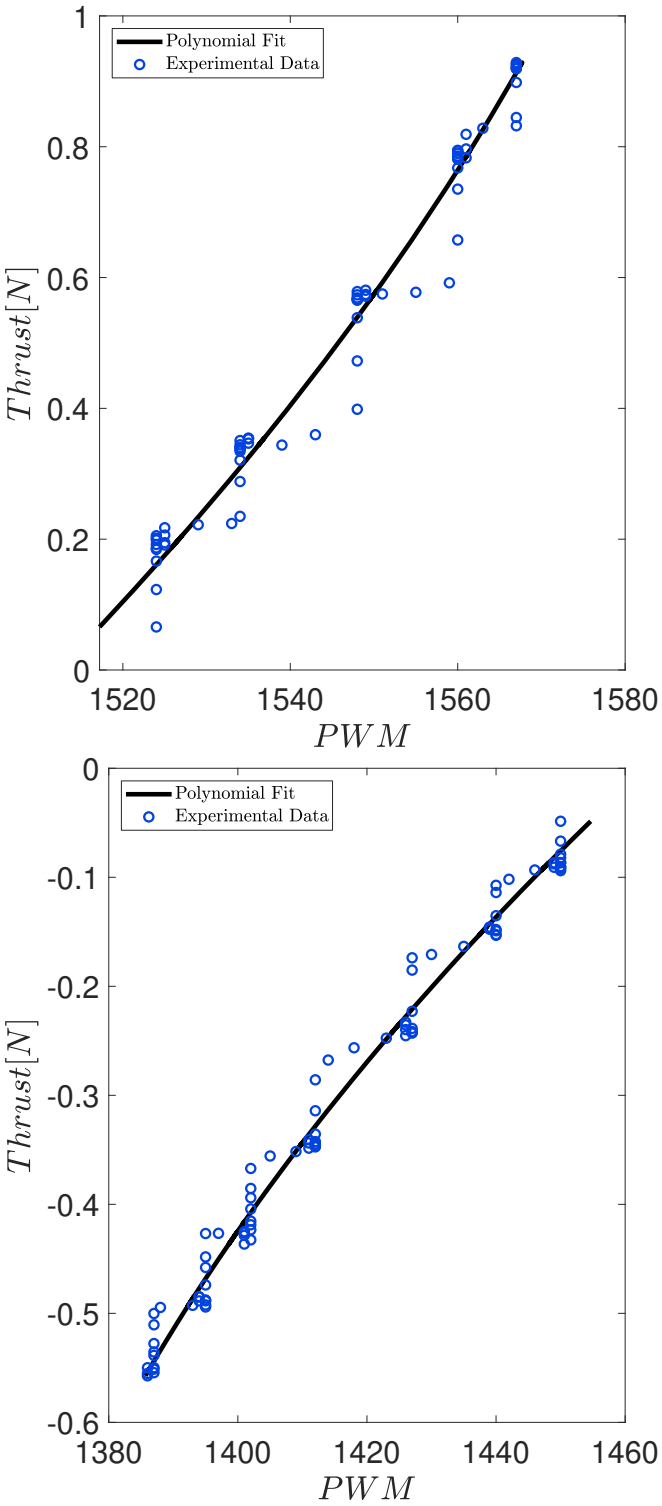


Figure D.13: Thrust as a function of PWM signal at 16.7 volts for a single lateral thruster.