Minimum Energy Control Strategy for an Uninhabited Airship with a Moving Gondola

Ali Mansur

Thesis submitted to the University of Ottawa

in partial fulfillment of the requirements for the

Doctorate in Philosophy in Mechanical Engineering

Ottawa-Carleton Institute for Mechanical and Aerospace Engineering

University of Ottawa

Acknowledgments

I am blessed and I thank God for every day for everything that happens for me and for the completion of this stage of my academic life. This thesis would not have been possible without the assistance and support of incredible people who were a part of my PhD program. I would like first to acknowledge my supervisor Dr. Eric Lanteigne for being an amazing supervisor. His guidance and support have been invaluable throughout my research. Words cannot express my thankfulness for his patience, guidance, encouragement, and constructive criticism of this thesis. Also, for all the resources he has provided to the completion of this thesis. I also will never be able to adequately express my gratitude to my wife, Tiser, for believing in me and being patient with the less time I spent with her and my kids during my Ph.D studies. I cannot forget the support of my friends Dr. Rajab Legnain, Dr. Tamer Beitelmal and Dr. Dhaou Said, for their advice and constant support. I would also like to thank my committee members of my Ph.D for taking the time to read and comment on this dissertation.

Lastly and most importantly, I dedicate this thesis to my parents who have raised, supported, taught and loved me throughout my entire life.

Abstract

An airship with a moving gondola is investigated with the goal of achieving a large pitch angle and of minimizing the total energy consumption required to goal position. The airship in this study is equipped with different actuation tools such as a moving gondola and a vectoring thrust, which can be used for various flight modes. The efficiency of the actuation methods employed is studied and compared in various flight scenarios, based on the airship's ability to reach the desired position while consuming the least amount of energy. The nonlinear dynamic model is derived using Newton-Euler equations. Backstepping and incremental nonlinear dynamic inversion (INDI) controllers are designed to track a desired trajectory by controlling the position of the gondola and the thrust. The dynamic models are then implemented and simulated in the Matlab/Simulink to evaluate the effectiveness of the controllers in different environmental conditions. The simulation results show the effectiveness of the controllers used, and a larger pitch angle of -89^{o}, can be reached thanks to the movement of the gondola to the front of the airship on the curved keel. The airship prototype was used for the experimental test to evaluate the pitch tracking performance of each of the controllers. The experimental results show that the prototype used can generate a -90^{o} pitch angle, thereby improving manoeuvrability and allowing for rapid changes in altitude. The energy model is developed to evaluate and compare the energy required by the airship for ascent, cruise, and descent flights, using different actuation methods. The effectiveness of the composite control strategy is demonstrated by completing the flight mission with the least amount of energy consumed. An optimization method is then developed to find the optimum design configuration to reduce the cost function, based on energy consumption, of the different flight scenarios while always respecting the design constraints. The Heuristic technique is used to obtain the optimal flight trajectory based on the platform's ability to complete the desired mission while minimising energy consumption. The results show that for pitch tracking, the vectored thrust has a rapid response, and the required thrust is high. Therefore, this configuration requires more energy than the moving gondola control configuration, in all cases studied. The composite configuration is found to be the most efficient method for completing the flight trajectory with the least amount of energy. The total energy consumption of the entire flight is reduced by about 17% by using the optimization algorithm to select the best actuation method for each flight mode.

Table of Contents

Nomenclature

$\mathbf{M}$	Mass matrix
$\mathbf{D}$	Dynamics vector
$\mathbf{E}$	Uncertainty vector
$\mathbf{x}_v$	State vector in body frame
$\mathbf{U}$	External forces and moments and defined
$\mathbf{A}$	Aerodynamic vector
$\mathbf{G}$	Gravitational and buoyancy vector
$\boldsymbol{R}_1$	directional cosine matrix
$\mathbf{F}_u$	Propulsive force vector.
$\mathbf{v}_w$	Total wind vector
$\mathbf{v}$	Linear velocities vector
$\boldsymbol{\omega}$	**A**ngular velocities vector
x	Position along OX
y	Position along OY
z	Position along OZ
α	Angle of attack
ϕ	Roll angle
θ	Pitch angle
ψ	Yaw angle
$\mathbf{B}$	Input matrix
$\mathbf{u}$	Control input vector
s_g	Gondola's position
F_x, F_y, F_z	Thrust components
ρ_{air}	Density of the air
z_g	Distance between the CV to the CM of the gondola along z
x_m	Distance between the CV to CM of the propeller along x
$d_{r,x}$,	Distance between the CV to the CM of fins along x
$d_{r,z}$	Distance between the CV to the CM of fins along z

L_r	Distance between the CV to the CM of the rail
m_g	Mass of the gondola
m_r	Mass of the rail
m_e	Mass of the envelope
m_f	Mass of the fins
T_R	Right thrust
T_L	Left thrust
γ	Vectored thrust angle
J	Jacobian matrix
P_m	Power required to overcome the drag
P_G	Power required to move the gondola
P_s	Power required to rotate servo motor
P_c	Constant electrical power
E_{tot}	Total energy required
C_{DV}	Drag coefficient
k_d	Drag factor
R_e	Reynolds number
δ_{air}	Dynamic viscosity of the air
η_p	Propulsion system efficiency
F_r	Rolling force
τ	Torque of the servo motors
ω_s	The angular velocity of the servo motor
K_1, K_2	Positive definite design matrices
$\dot{S}$	Gondola repositioning rate
ϕ_A	Measured roll angle from the accelerometer
θ_A	Measured pitch angle from the accelerometer
C_{Dho}	Hull zero-incidence drag coefficient
C_{Dfo}	Fin zero-incidence drag coefficient
C_{Dgo}	Gondola zero-incidence drag coefficient
C_{Dch}	Hull cross-flow drag coefficient

C_{Dcf}	Fin cross-flow drag coefficient
C_{Dcg}	Gondola cross-flow drag coefficient
$\left(\frac{\delta C_L}{\delta \alpha}\right)_f$	Derivative of fin lift-coefficient with respect to angle of attack
$\left(\frac{\delta C_L}{\delta \delta_{E,R}}\right)_f$	Derivative of fin lift-coefficient with respect to fin angle
S_h	Hull reference area, $V^{2/3}$
S_f	Fin reference area
S_g	Gondola reference area
$d_{f,x1}$	Distance between CV to GC of fins along *x*
$d_{f,z}$	Distance between CV to AC of fins along *z*
$d_{c,z}$	Distance between CV to AC of gondola along *z*
η_f	Fin efficiency factor
η_h	Hull efficiency factor
m	Airship mass
V	Airship volume
D	Airship diameter
L	Airship length
$d_{g,z}$	Distance between CV to gondola CG along z
$d_{p,y}$	Distance between propeller centers to the x - z plane along y
$d_{p,z}$	Distance between propeller centers to gondola CG along z
I_x	Moment of inertia about OX
I_y	Moment of inertia about OY
I_z	Moment of inertia about OZ
k_1	Lamb's inertia ratio along OX
k_2	Lamb's inertia ratio along OY or OZ
$K`$	Lamb's inertia ratio about OY or OZ

Acronyms

LAT	Linear Analysis Tool
LTA	Lighter Than Air
HTA	Heavier Than air
UAV	Unmanned Aerial Vehicle
LQR	Linear Quadratic Regulator
AURORA	Autonomous Unmanned Remote Monitoring Robotic Airship
DIVA	*Dirigível Instrumentado para Vigilância Aérea*
CLF	Control Lyapunov Function
CFD	Computational Fluid dynamics
SLC	Sequential Loop Closure
SMC	Sliding Mode Control
MIMO	Multi-Input / Multi-Output
ETDPOF	Exact Tracking Error Dynamics Passive Output Feedback
NDI	Nonlinear Dynamic Inversion
INDI	Incremental Nonlinear Dynamic Inversion
CM	Centre of Mass
CV	Centre of volume
PWM	Pulse Width Modulation
IMU	Inertial Measurement Unit
ESC	Electrical Speed Controller

List of Figures

List of Tables

Chapter 1: **Introduction**

In the past twenty years, there has been a resurgence in airship research. A growing safety trend is to keep humans out of danger especially in the field of Search and Rescue (SAR), military operations, and firefighting, etc. Researchers have been working impetuously to design new airship configurations and propose different methods for trajectory tracking and feedback control design [1–5]. Conventional airships are driven by a set of thrusters along the length of the hull with either elevators and rudders or thrusters for maneuver and altitude control. Larger airships use air ballasts (also referred to as ballonets) for altitude and pitch control. For smaller vehicles, the typical actuator configuration is two thrusters oriented along the longitudinal axis for forward motion and a single thruster oriented along the vertical axis of the airship for altitude control. Other configurations include thrusters in the longitudinal axis that can be rotated along the lateral axis [1]. However, with new lightweight materials and manufacturing methods, many unconventional airship designs have been either proposed or tested. These vehicles have notable features that include the use of aerodynamic helium envelopes, multiple envelopes, segmented buoyancy cells, or utilize hybrid winged configurations to achieve lift. A rapid altitude change is a challenge for large-volume airships. Typical actuators, such as aerodynamic control surfaces, vectored thrust, and ballonet, cannot function very well. Therefore, development of new actuators and relevant controllers are necessary. Changing the center of gravity position in order to control airship's longitudinal motion by a moving-mass control method is introduced to stratospheric airships. Moving-mass control can change the external moment by displacing the masses, such that the attitude of aircraft is changed. Moving mass method has been used in different successful

applications, such as, in the study of spacecraft, reentry vehicles, underwater vehicles, kinetic kill vehicles, saucerlike air vehicles, and missiles as well as in low-altitude airships [1–7].

1.1 Research Motivation and Contribution Areas

The limitations of autonomous landing and take off the airship offer interesting research opportunities to search for robust vehicle architectures and control algorithms to perform fully autonomous flight. The concept of shifting the centre of gravity by moving mass to increase the system's manoeuvrability and generate a rapid ascent and descent without needing to add considerable number of components motivated this work to study the airship developed by Lanteigne [13]. The airship demonstrated that a large pitch angle could be reached using moving gondola (or payload). The moving mass control has been studied and applied in different applications however, none of the studies were able to produce a pitch angle of -90^{o}.

The airship prototype used in this study is shown in Figure 3.3 and it has an extended keel to the front end of the airship, allowing the gondola to move to the front and generate a large pitch angle. This airship equipped with different actuation methods (moving gondola and vectoring thrust) that can be used to preform different flight modes. These actuation methods offer interesting research opportunities to study the efficiency of the different control configurations, at various flight scenarios based on achieving the goal position with the least amount of energy consumption or the shortest arrival time. For long endurance, weight and energy costs must be reduced. The planning of optimal trajectories is a particular challenge for airship operations due to the different wind condition. As a result, the trajectory must be carefully planned and optimized in order to ensure that the desired goal be reached within acceptable performance bounds of flight time and energy consumption. Therefore, select the optimal method for the mission is one of the main objectives of this thesis.

The aim of this thesis is to present a complete study of modeling, control, optimization and flight test of an airship with a moving gondola. The main contributions are as follows:

1. The airship architecture is investigated by analysis, design and simulations.
2. The non-linear dynamic model is derived using Newton-Euler equations.
3. A backstepping and incremental nonlinear dynamic inversion controllers are designed for the purpose of tracking a desired trajectory by controlling the position of the gondola and the thruster angle. The performance of both controllers is then simulated under different environmental conditions.
4. The flight tests are conducted to evaluate the pitch tracking performance of each of the controllers as well as the capability of the moving gondola to produce a large pitch angle.
5. The analysis of the efficiency of actuators available in the prototype including the composite actuators to complete flight missions at various flight scenarios based on minimizing the energy consumption to reach the goal position with the shortest arrival time.
6. The development of the optimal flight trajectory based on the capability of platform to reach the desired mission with minimum energy consumption. This method is amenable for onboard implementation.

The tracking performances of the airship architecture (Model 1) by using the backstepping and INDI controller are published in 2020 6th International Conference on Mechatronics and Robotics Engineering (ICMRE) and in 2020 Canadian Society for Mechanical Engineering (CSME). The published manuscripts are included in Appendix H.

1.2 Thesis Organization

The remainder of the thesis is organized as follows:

Chapter 2 discusses a review of existing airship models and controls.

Chapter 3 describes the airship architectures proposed and derivation of dynamic models as well as the calculation of the energy consumption during the flight mission.

Chapter 4 presents the backstepping and INDI controllers used for trajectory tracking.

Chapter 5 discusses and compares the numerical simulation results of the two controllers.

Chapter 6 presents the evaluation and comparison of the energy required by the airship to perform different flight missions using different actuation methods.

Chapter 7 introduces an optimization method to find the optimal flight trajectory based on the capability of platform to reach the desired mission with minimum energy consumption.

Chapter 8 describes the experimental platform of the prototype used for flight test (the physical properties, hardware, and software).

Chapter 9 presents and discusses the flight test results.

Chapter 10 discusses the conclusion and the future work.

Chapter 2: **Literature Review**

Airships have been used extensively at the beginning of the 20th century for transportation and military purposes because they can loiter and sustain long endurance flights over long distances. Conventional airships are driven by a set of thrusters along the length of the hull and elevator and rudders at the tail. Larger airships use air ballasts (also referred to as ballonets) for altitude and pitch control. With the introduction of lighter, smaller, and more powerful energy storage systems, actuators, sensors and computing hardware, the research and development of uninhabited airships has gained momentum for applications [14]. For conventional Lighter than Air vehicles, the typical actuator configuration is two thrusters oriented along the longitudinal axis for forward motion and a single thruster oriented along the vertical axis of the airship for altitude control. Other configurations include thrusters in the longitudinal axis that can be rotated along the lateral axis [15]. However, with new lightweight materials and manufacturing methods, many unconventional airship designs have been either proposed or tested. These vehicles have notable features that include the use of aerodynamic helium envelopes, multiple envelopes, segmented buoyancy cells, or utilize hybrid winged configurations to achieve lift [3–5]. The small size uninhabited radio-controlled models, which have size of a few tens of cubic meters, their domain of operation is currently essentially restrained to advertising or aerial photography. However, their properties make them a very suitable support to develop heterogeneous air/ground robotics systems and they are easy to operate. Also, they can safely fly at very low altitudes (down to a few meters). In addition, they can hover a long time over a particular area, while being able to fly at a few tens of kilometers per hour while consuming less energy [19].

Airships fall into three main structural categories, namely, rigid airships, semi-rigid airships and non-rigid airships. These types are defined by the shape of the envelope support structure. Rigid airships are airships that have an internal framework that sustains their shape. Semi-rigid airships are similar to non-rigid airships in their structure with an addition of a rigid keel or a truss that extends from the nose to the tail along the bottom surface of the airship. A broad review of uninhabited airship platforms and structures is presented in Appendix A for reference.

2.1 Modelling

In the history of aviation, the airship was the first vehicle used to demonstrate that controlled flight was possible. Therefore, understanding their flight behavior, design their control systems, and optimize their flight trajectories motivated researchers over the past several decades. There are multiple German papers on the topic that date to the early 20th Century. DeLaurier investigated the dynamic stability of several airship designs [20]. DeLaurier applied the analytical model and assumptions to three example airships, (U.S. Navy, Airships ZR-1 and ZR-4). The next interesting study was done by Gomes in 1990 [21]. He presented the first known non-linear dynamic model for an airship derived based on a Newton-Euler approach, which has become an essential starting point for other future research. Gomes used the wind-tunnel aerodynamic coefficient results to ACSL (Advanced Continuous Simulation Language) non-linear simulation program to obtain aerodynamic data for the YEZ-2A airship. The wind-tunnel is shown in Figure 2.1. In 2011, Li et al. [22] published a comprehensive review of airship dynamics covering many decades [21]. They applied different prediction approaches for different aerodynamic effects to airships based on a rigid-body assumption.

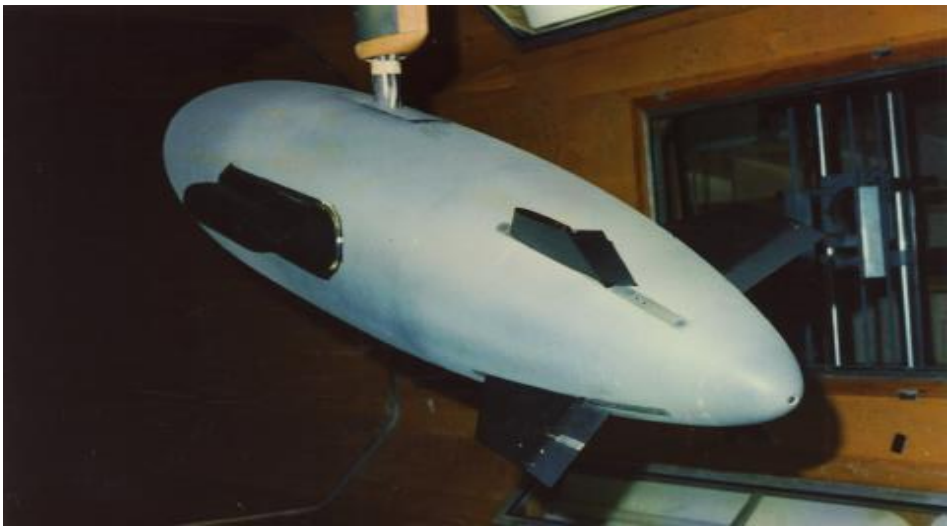


Figure 2.1 YEZ-2A 1:75 scale model in the test section of the 8ft x 6ft wind tunnel [21]

The dynamic modelling of conventional Heavier Than Air (HTA) aircraft has been broadly studied and become highly reliable. However, for a variety of reasons, limitations exist when applying these HTA dynamics models to airships. First, the airships have lift generation mechanism which is different from the HTA aircrafts due to the use of light lifting gas. Certain solid–fluid interaction forces can be neglected for HTA aircraft, such as buoyancy and those related to the inertia of the surrounding air, but these forces become important for airships. Second, airships have very different shape and structure compared with HTA aircraft.

An accurate model of the airship dynamics is the first requirement of an accurate control system. Over many decades, many efforts have been made to investigate the airship dynamic model. Different approaches presented in literature to improve the dynamic model of the airships. These modelling methods could be classified into two groups: The Newton-Euler method, and the Lagrangian formulation. The Newtonian approach is vector oriented since everything is derived from Newton's second law [23]. The Lagrangian description uses generalized coordinates (one for each degree of freedom), all of which must be independent. One advantage with the Lagrangian approach is to deal with two scalar energy functions $\boldsymbol{W}$ (kinetic energy) and $\boldsymbol{Ep}$ (potential energy).

The Lagrangian equations were used to develop the equation of motion for many Lighter Than Air (LTA) vehicles [4], [10], [23–27]. There are some airship mathematical models were described by the Newtonian non-linear equations of motion [28–31]. Azinheira et al. [33] investigated the wind influence on airship dynamics via a Newtonian approach. A few years later, they discovered that [33] lacks an inertial moment term, and they corrected the moment result using a Lagrangian approach in [34].

Unlike modelling of rigid body airships, few publications are available concerning the modelling of non-rigid airships. Salim et al. [35], [36] presented dynamic modelling of small autonomous non rigid airships, using the Newton-Euler approach. They discussed the motion in six degrees of freedom. Their model was to introduce the effects of the vertical and horizontal control surfaces. Wu et al. [37] performed wind tunnel test measurements with a typical non-rigid airship model. They established a loosely coupled procedure for the fluid–structure interaction (FSI) of airships on the basis of computational fluid dynamics and nonlinear finite element analysis methods.

Yuwen Li [24] conducted a study at McGill University in Canada, they presented two derived models for an airship. They used series of prediction methods to predict the aerodynamic and dynamic characteristics of an airship. It was found that the most significant aerodynamic effect on an airship was the Munk moment in the added-mass terms which tends to destabilize the pitch and yaw rotations, while other aerodynamic forces are normal to the centerline stabilize the airship. The numerical results were compared to existing Computational fluid dynamics (CFD) wind-tunnel and flight test data. The comparison showed that the aerodynamic computational model can lead to reasonable prediction of the aerodynamic and dynamic characteristics of an airship.

The engine selection for an aircraft is an important part of the complete aircraft design process. Like any conventional or unconventional aircraft design, basic engine requirements are largely dictated by the selected mission profile and estimation of required aerodynamic parameters. Accurate estimation of drag force is important for thrust calculations and estimation of power requirements for engines [38]. There is always an option of using either combustion engines or electric motors to propel the airship. Usually two engines on the sides of the gondola have been used [4], [13], [26], [34], [38–42]. The Figure 2.2 shows the DIVA airship powered by two motors on each side of the gondola, with vectoring capability ranging from -30^{o} to +120^{o}.

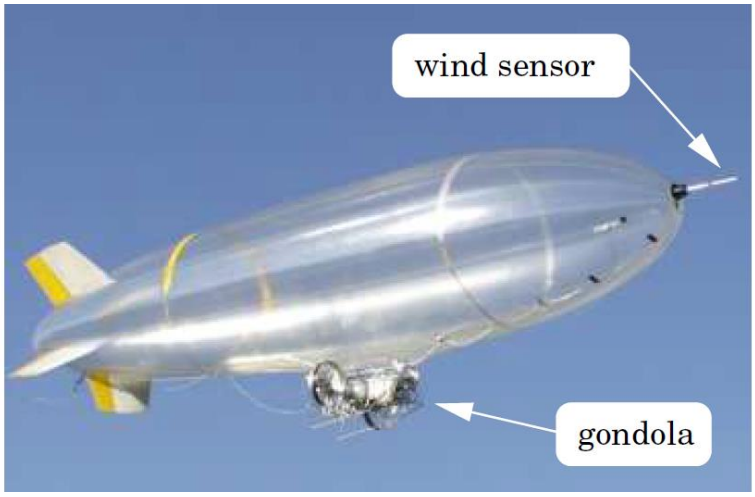
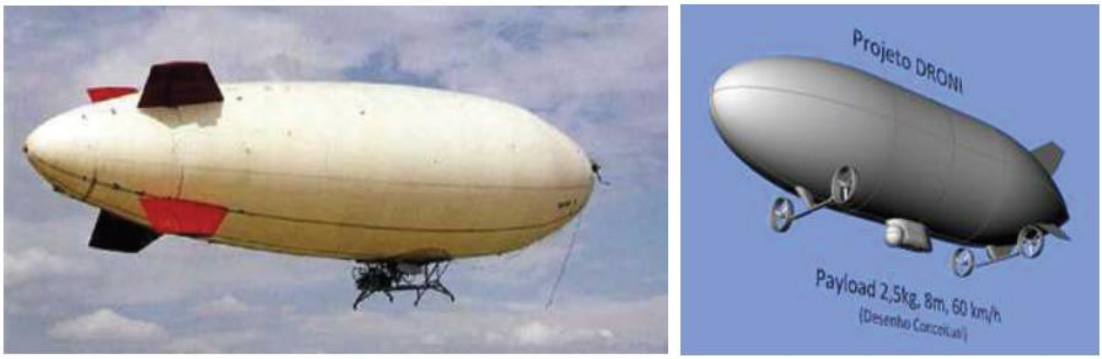


Figure 2.2 DIVA airship with two engines [39]

The thrusters are usually connected through a movable axle that can tilt up and down. Some airships have used three thrusters to have the possibility to control the altitude and yaw angle while hovering [18], [43], [44].

Several airships have used four engines to provide the required thrust during the flight mission. Arias adapted the dynamic model of AURORA [45], [46] to consider four engines instead of two, and the four thrusters have the ability to rotate 20 degrees around their axes, which allows each

thruster to generate a lateral force component besides the longitudinal and vertical components when thrusters are vectorized upward [47]. Azinheira et al. [48] addressed the lateral control of DRONI (Dirigível Robotico de Concepção Inovadora). Airship DRONI is a project code-name of an autonomous 10 m long airship powered by four vectored electric motors. With the purpose of flying in the remote areas of the Amazon forest, DRONI is an evolution of the AURORA airship [3], as seen in Figure 2.3.



a) b)

Figure 2.3 a) AURORA project and b) DRONI project [47]

Battipede et al. [31] presented unconventional airship based on six propellers, moved by electrical motors, suitably set in order to produce the desired forces and moments, necessary to control and maneuver the airship both in hovering and forward flight as seen in Figure 2.4. In particular, two vertical propellers provide the vertical thrust for climbing, descent and pitching maneuvers, while four thrust-vectoring propellers mounted on rotating arms allow to control the lateral-directional attitude of the airship.

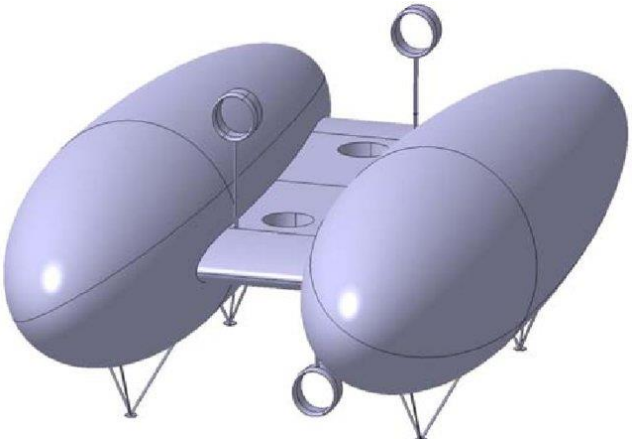


Figure 2.4 The Nautilus new concept unmanned airship [49]

In this study, two airship architectures are used. The first model proposed is designed based on the airship prototype used in [13] which is shown in Figure 2.5. This prototype has a strait rigid keel fixed at the bottom of the helium envelope allowing the gondola to move forward and backward on the keel by a DC gearmotor to generate descending or ascending pitch. The second model is designed based on shape optimization methodology for rapid descent with lower drag and optimum surface area [50]. Both architectures will be discussed in detail in Section 3.1.

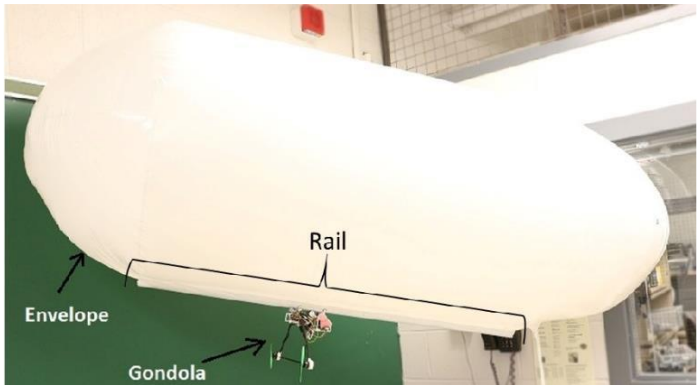


Figure 2.5 The airship prototype [13]

2.2 Control

Flight control is the first problem to address in order to provide an airship with the ability to autonomously achieve any flight mission. Control approaches can be categorized into linear control design techniques and nonlinear control techniques. The classical linear control design techniques have been used to solve flight control problems for many years. The linear methods do not take into account the real system conditions such as wind gusts or sensor measurement noise and abrupt of the different use of actuators necessary within each flight region. These model parameter errors and disturbances are a key issue in the choice of the controller. However, if the automatic control system is to cover the complete aerodynamic range from hover to cruise flight, then the control solution must be able to cope with the nonlinear and under- actuated airship dynamics [27]. In this case, it is sometimes difficult to obtain practical controllers based on linear design techniques. In the literature, the linear control approaches have been used to solve flight control problems for many airships.

One of the first autonomous experimental flight of an airship resulted from the implementation of a PID heading controller, along with an automatic altitude control, for path-tracking through pre-defined points in altitude and attitude [40]. Similar controllers were used in many other airships including the Aurora where a successful flight was achieved using a PI controller to track a set of predefined waypoints [29] [40 – 49]. The horizontal trajectory was controlled automatically by the onboard system, while altitude was controlled manually by the ground pilot. The controller's proportional and integral gains were obtained by trial and error. The flight trajectory was validated by simulation results and showed good agreement.

Lanteigne et al. [13] used a PID controller to control the airship pitch using the gondola position. They built a miniature uninhabited airship with a moving platform to address the limited altitude

maneuverability of the airships. They examined the theoretical and experimental differences between altitude and pitch variations generated using aerodynamic control surfaces (elevator) versus configuration changes achieved by repositioning the gondola along the longitudinal axis of the helium envelope.

De Paiva et al. simulated airship (AURORA) environment to study the airship behavior in a variety of flight modes and simulate different control strategies [54]. They used a simple robust PID controller which proved to assure performance specifications to the pitch control loop for a class of uncertain systems derived from large variations in the operating conditions. The Figure 2.6 shows the pitch attitude control structure (PID) for the linearized uncertain model. The results of nonlinear simulations showed absence of overshoots and good tracking properties of the pitch angle.

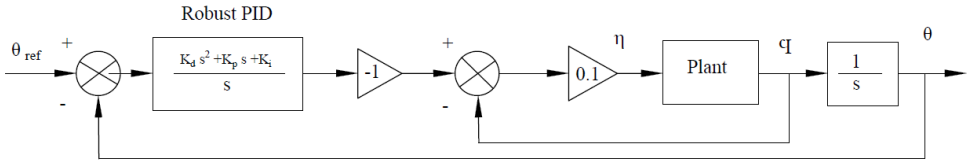


Figure 2.6: Pitch attitude control structure for the linearized uncertain model [54].

The PID controllers have been widely used not only for the airship applications but also for UAV quadrotors, helicopters and robots in the industrial applications. Bouabdallah studied the design and control of quadrotors with application to autonomous flying. He developed five different controller techniques including the PID controller. The PID controller proved to be well adapted to the quadrotor when flying near hover [58].

In the presence of significant nonlinear effects, any linear control approach can only be guaranteed to be locally stable and it may be difficult to achieve the desired performance or even guarantee stability. In these cases, nonlinear control techniques must be used to account for dynamic

nonlinearity in the system. Some of the more common methods used for nonlinear UAV control are backstepping, linear quadratic regulator, dynamic inversion, incremental nonlinear dynamic inversion sliding mode control, gain scheduling.

Recoskie in [60] developed a backstepping based control law and compared it to a conventional PID control for Five trajectory following on the basis of control effort and tracking error for each of the cases. The two flight controllers were developed to track the trajectories generated from the previously developed trajectory planner backstepping was shown to have a 50% decrease in tracking error versus PID. He made considerations to prevent actuator saturation and address error coupling. The energy consumption, however, was found at least 50% higher during trajectory following for both controllers in all cases. This increase can be attributed to the effects of introducing the full dynamic model and uncertainty.

Masar et al. [61] presented the design procedure for a gain-scheduled Linear Quadratic Regulator (LQR) controller for an autonomous airship. Simulation results showed a good performance of the designed control system, and it would be tested on the real airship.

LQR controllers have been used not only for airships but also for helicopters. Good position tracking performance has been obtained using these controllers. Bergerman et al. [62] used LQR controller as an inner loop to stabilize the unstable poles of the identified linear model of the robotic RMAX helicopter. This controller was then combined with a feedback linearization controller that decoupled the linear dynamics of the lateral, longitudinal, vertical, and heading axes and enabled trajectory tracking. The proposed cascaded position and heading controller for a robotic RMAX helicopter is shown in Figure 2.7. Both a sequential loop closure and linear quadratic regulator point-to-point controller were implemented on the linearized dynamic model of a mid-sized airship by Kulczycki et al. [63]. The LQR controller greatly outperformed the SLC

(sequential loop closure) controller in time-of-completion and overall input required for purely altitude climb maneuvers.

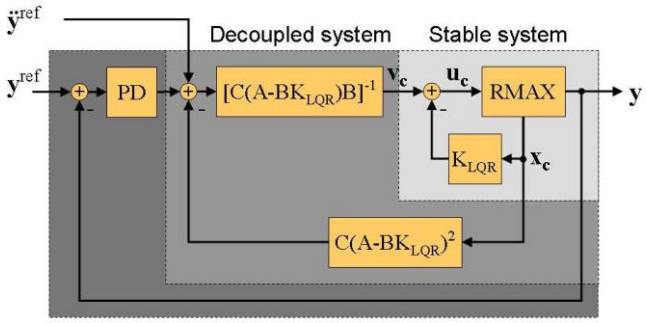


Figure 2.7: System-level block diagram of the cascaded controller [62].

Dynamic Inversion, also known as feedback linearization, involves on line approximate linearization of a nonlinear plant via feedback [4], [23], [26], [63–68]. Moutinho and Azinheira in [26], [64] used Dynamic Inversion (DI) approach as a solution for the longitudinal and lateral control systems of the AURORA airship. A Dynamic Inversion controller was implemented with desired dynamics given by a linear optimal compensator. The stability of the nonlinear system was analyzed applying Lyapunov's stability theory. The results obtained illustrated the overall system robustness. An autonomous flight controller was developed to guide the Titan airship along a three dimensional trajectory by Acosta and Joshi [67]. They used a combination between a proportional-derivative controller for position control and a nonlinear dynamic inversion controller for velocity control. Simulation results showed the autonomous flight controller is able to successfully pilot the airship along the desired trajectory within mission error tolerances and adapted to increase robustness in the presence of parameter uncertainty. Yedavalli et al. [69] studied the robustness of the control system across the entire flight envelope. The proposed method was presented to determine the robustness of a baseline dynamic inversion control system developed for a

representative space-maneuvering vehicle. The stability of the nominal system was determined by using the Lyapunov function approach for nonlinear systems.

Incremental nonlinear dynamic inversion (INDI) control is a variation on the nonlinear dynamic inversion (NDI). The NDI is a nonlinear control method which eliminates the system nonlinearity by means of feedback and results into partly or entirely linearized closed-loop system dynamics, where conventional linear control techniques can then be applied. The INDI method retains the high-performance characteristics of NDI while reducing model dependency, increasing robustness and reducing computational complexity [1-2]. The INDI only requires knowledge of the system kinematics and the actuator dynamics. The dynamic model is replaced with measurements of the system dynamics [72]. The INDI controller solves the incremental form of equations of motion using acceleration feedback and generates a control law substantially reducing the controller's dependence on complex, and sometimes inaccurate, models such as those describing the aerodynamic phenomena in flight dynamics. Feedback of angular accelerations eliminates the sensitivity to model mismatch, greatly increasing the performance of the system compared with conventional nonlinear dynamic inversion [73]. The incremental nonlinear dynamic inversion method has been successfully used to control various aerospace systems and shown desirable robust performance to aerodynamic model uncertainties [4–8]. The concept of INDI was first suggested in [77]. They examined the behaviour of NDI control law on the VAAC Harrier. During flight test a poor quality angular accelerations signals were observed and filtering and differentiating the angular rates led to oscillatory closed-loop behaviour. They suggested that the preferred solution would be to use an independent angular acceleration signal. An extension of Smith's approach to reformulate the dynamic inversion to be less dependent on the onboard model was developed in [78]. They chose two different sources to get independent data of the state

acceleration vector: linear accelerometers and numerically differentiated angular rates. Angular acceleration data was derived from a unique implementation of linear accelerometer measurements, in addition to the differentiated angular rates. Recently, the INDI methodology has been studied at Delft University of Technology [73]. INDI was adopted to control the attitude of a fixed-wing aircraft and, when compared to NDI, demonstrated improved robustness to model uncertainties. More recently INDI control theory was used to achieve a complete quadrotor controller in [79]. INDI showed improvements with the control of a time varying quadrotor model. The INDI controller was able to maintain the quadrotor trajectory while performing some degree of aggressive manoeuvres.

INDI control was applied to the lateral control of an under actuated airship with uncertain dynamics model [48]. The stability and robustness of the proposed sensor-based control solution was illustrated with representative simulation results, including wind disturbances and a path following loop. The INDI controller performance was evaluated by adapting the DRONI's geometry with four-propellers configuration of the AS800 of [3]. The controller was successfully able to follow the path with a maximum error of 10 m in the presence of wind disturbances.

The derivations of INDI control law in literature are based on the time scale separation principle, which considers that the controls can change significantly faster than the states when the sampling frequency is high [4–8]. The time scale separation principle uses the fact that the time constants of the inner and outer loops are different and hence the control laws for inner and outer loops can be designed independently. The closed-loop system stability of a general linear system controlled by INDI was investigated in [80]. The analytical stability analysis showed that implementing discrete-time INDI with a smaller sampling time resulted in larger stability margins regarding system characteristics and controller gains.

Backstepping is one of the most popular nonlinear controller which is based on Control Lypanov Functions (CLF) [81]. Backstepping has been used in UAVs and proven to offer robustness in the presence of parametric uncertainties [81–89]. A backstepping methodology was adapted to design a closed-loop trajectory-tracking controller for an under-actuated airship [85]. The authors proposed a time-varying stabilizing controller of the position and the orientation of an underactuated autonomous airship. They found that the airship could not be stabilized to a point using continuous pure-state feedback law. However, the stabilization problem was solved with an explicit homogeneous time varying control law, based on an averaging approach. Yang et al. [91] proposed a backstepping control approach to tackle the planar station-keeping problem of a stratospheric airship platform. The stability of the closed-loop control system was proven by the Lyapunov theorem. The simulation results illustrated the effectiveness and robustness of the proposed control approach.

Hygounenc and Soueres [86] provided a complete solution to the control problem of airships in case of very low perturbations. They proposed a flight decomposition allowing to define canonical navigation phases from takeoff to landing. For each phase a reduced model was determined, and a controller was designed on the base of backstepping techniques. The control technique used allowed to drive the system to a steady state. In Murguia-Rendon at et. [88], the authors addressed and solved the trajectory tracking problem for the planar dynamics of a thrust vectored airship. The proposed controller was based on a combination of two nonlinear control design techniques, backstepping and Exact Tracking Error Dynamics Passive Output Feedback (ETDPOF). The closed–loop stability was preserved in the presence of unmodeled dynamics.

A backstepping controller for trajectory tracking control of a high-altitude unmanned airship was proposed by Bang at el. [92]. Liesk at el. studied the design of a combined backstepping and

Lyapunov controller for the attitude, velocity and height control of an unmanned, unstable, fin-less airship [87]. The controller performance was verified using a simulation and then tested in outdoor flight tests. The simulations showed that the designed controller could successfully stabilize the airship's attitude and velocity in wind conditions to be expected during outdoor flight tests. During the flight tests, the controller performed similarly to the simulations with the exception of height control.

Sliding mode control (SMC) is another nonlinear control technique [14], [26], [91–93]. This method involves high speed switching to keep the system on a hyper surface (in state space) called a sliding surface so the system trajectory exhibits desirable behavior when confined to this hyper surface. Different control gain is applied depending on whether the current state is above or below the sliding surface. In [93], the authors proposed a neural network self-gain-scheduling sliding mode control (NNSSMC) approach to design the trajectory controller for robotic airships in the presence of parametric variations and external disturbances. The proposed control approach showed increase the robustness against system variations and decrease the chattering phenomenon of the sliding mode control (SMC). In addition, the stability and convergence of the closed-loop controller were proven by using the Lyapunov theorem. Figure 2.8 shows the block diagram of SMC, where η_c and η are the commanded trajectory and true trajectory, respectively.

Yang et al. [96] introduced the conceptual design, dynamics modelling and attitude controller of the stratospheric telecommunication platform. The dynamic model of the platform was derived from the Newton–Euler formulation, and the sliding mode control (SMC) approach was employed to develop the attitude-tracking controller in the presence of uncertainties and external disturbances. The simulation results verified the effectiveness and robustness of the proposed control scheme. In [95], the authors presented the design of two MIMO Sliding Mode Controllers

for the guidance and navigation system of the AURORA airship (for the longitudinal and lateral modes). The MIMO SMC design proposed was adapted from the approach presented in Healey and Lienard [97]. The two controllers were derived (for the longitudinal and the lateral modes) based on linearized models of the airship. The results showed that the approach was a strong and robust tool for the design of a single global control scheme for unmanned vehicles.

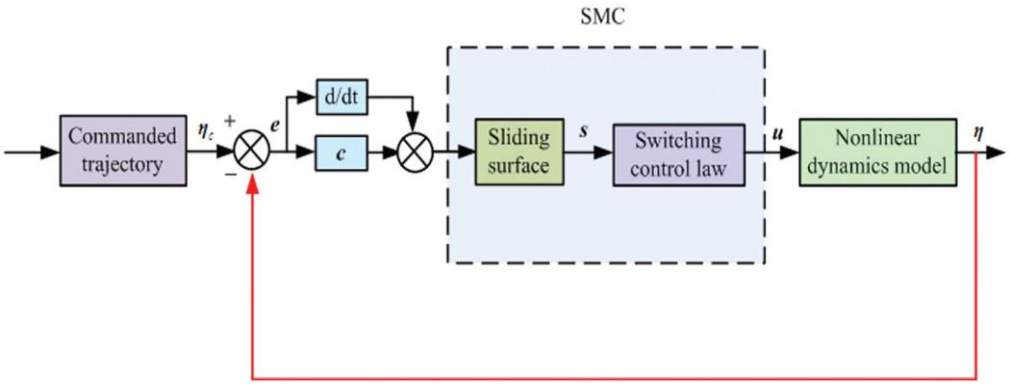


Figure 2.8: Block diagram of SMC [93].

Gain Scheduling solution is another important nonlinear control approach which optimizes the linear state-space controller parameters to various operating points of the airship [13], [15], [54], [91], [97–102]. In [27], A path-tracking gain-scheduling controller was designed and its performance and robustness were evaluated in the simulation environment described for a complete airship mission consisting of vertical takeoff and landing, cruise flight and ground-hover, under realistic wind disturbances. In this paper, the authors comprehensively addressed the relevant aspects of the control design methodology, the airship modelling, the dynamics analysis over the flight envelope and the step-by-step design of a gain-scheduling controller. The simulation results demonstrated the robustness of the gain-scheduling path-tracking control and the suitability of the approach. General features of the gain scheduling approach and in terms of its applications in flight control and automotive engine control was described by Rugh and Shamma in the Survey Paper [102].

Some other nonlinear methods have also been proposed for airships at which include, Transverse feedback linearization [104], [105], Poincare map [105–107], Lagrangian control approach [109], Fuzzy sliding mode control method [51], [109–111].

For a specific control application, combinations of controller techniques can be used for airship autonomous control. Yang et al. [113] combined the backstepping control with the sliding-mode control to assure the robustness against parametric uncertainties and external disturbances. Beji and Abichou [81] used a combination between the integrator backstepping approach and Lyapunov theory to improve the stability of the airship. Liesk et al. [87] used an integral backstepping algorithm for attitude control and Lyapunov theory for velocity control. Yang et al. [113] addressed the problem of positioning control for an autonomous airship in the presence of model uncertainties and external disturbances by using a combination between the backstepping control and the sliding-mode control. Their proposed control scheme provided a promising approach for the positioning control of an airship.

In this study, the model developed in [13] will be modified to reflect the new vehicle architectures. The new architectures provide an alternative solution to over-actuation and ballonets for the rapid altitude changes required when landing or avoiding obstacles. The pitch angle of the airship due to the changes in the gondola's position without using rudder and elevator will be analyzed in this study. For this new architecture, backstepping and INDI controllers will be developed and evaluated. These controllers were chosen because they are appropriate for a nonlinear system such as the airship. Another reason for selecting the INDI controller is model independence which improves the robustness of this controller to model mismatches.

Chapter 3: **Airship Architecture and Dynamic Modelling**

This chapter is organized as follows; first, two airship architectures used in this study are presented in section 3.1, the nonlinear dynamic model is presented in section 3.2. Then, the energy consumption is presented in section 3.3.

3.1 Airship Architectures

Two airship architectures used in this study are presented in this section. The first model proposed (Model 1) is designed based on the airship prototype used in [13]. The Model 1 is shown in Figure 3.1 and it has the thrusters are located along the center of volume (CV) of the airship to minimize the pitch moment caused by the thrust. In addition, the keel extends to the front of the airship to allow the gondola to move all the way to the front of the airship to increase the maximum pitch angle to -90^{o}. The gondola position along the keel is controlled using a DC gearmotor.

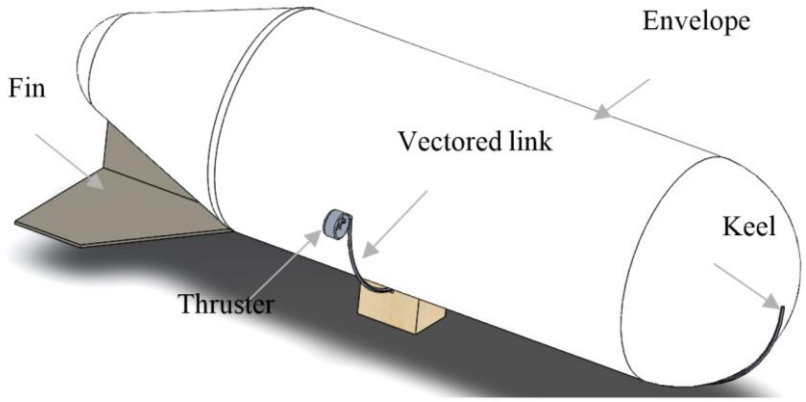


Figure 3.1 Model 1 airship design

The gondola has a weight of 0.67 kg and the centre of mass (CM) located at 0.53 m below the CV based on the SolidWorks model. The remaining physical properties are given in Appendix B.

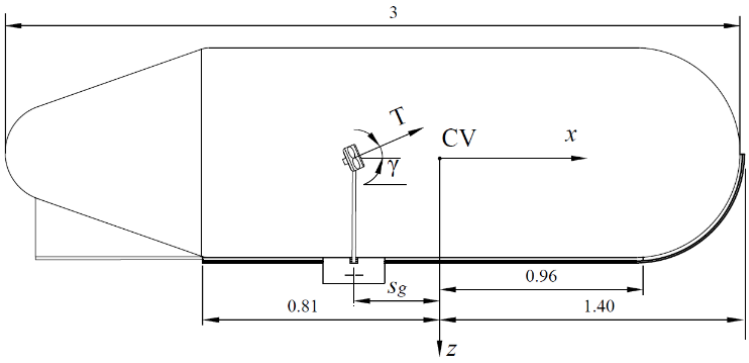


Figure 3.2 Side view of the proposed airship with dimensions in (meters)

The second model was designed based on shape optimization methodology for rapid descent with lower drag and optimum surface area [50]. This model has the thrusters located closer to the gondola to minimize the mass and inertial of the gondola as well as to facilitate manufacturing, handling, and storage. The CAD model of the proposed airship is shown in Figure 3.3. This model has a similar drive mechanism for the gondola as in the Model 1, where the gondola moves on a rail fixed to the keel of the helium envelope. The rail is also strait along the mid-section of the vehicle and circular along the bow. However, Model 2 is different in shape and size that allows more payload to be carried. The dimensions of airship (Model 2) are shown in Figure 3.4 and the physical properties of the airship are listed in the Appendix B. The propulsion system is mounted on a tilting system with servo motor. The tilting system can provide a tilting angle from $\zeta = -10^{o}$ to $\zeta = +110^{o}$.

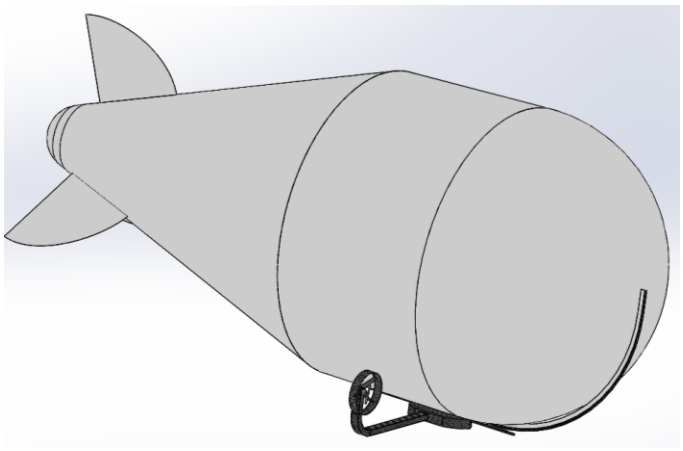


Figure 3.3 Model 2 airship design

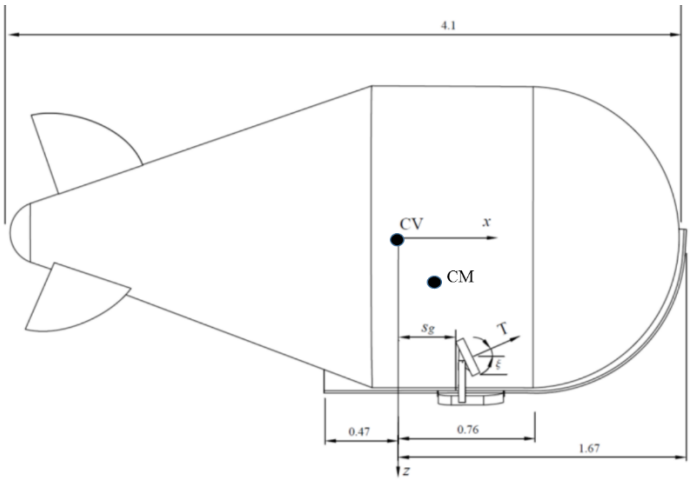


Figure 3.4 Model 2 airship dimensions (meters)

3.2 Nonlinear Dynamic Model

The non-linear dynamic model of the proposed architectures was developed using Newton-Euler equations as derived by Gomes [21], with addition from Recoskie [60], and modifications to account for the curved rail sections and thruster locations. In order to apply trajectory tracking control strategies, a complete dynamic model of the airship was developed based on the physical characteristics of the prototype platforms as well as empirical coefficients derived from similar airships. The dynamic model was developed based on the assumption that the airship is considered

a rigid body such that aeroelastic effects can be ignored, which is justified by the fact that the envelope is pressurized and the structure of the airship is supported by the curved keel.

3.2.1 Axis and Notation

For describing the airship motion, two frames of references are defined to describe the airship motion: the inertial reference axis and the airship body axis. Both models use the same reference frames and model variables, with the exception of Model 1, which has no vertical distance between thruster and CV. The two reference frames used in this study for the development of the equations of motion with respect to the physical characteristics of each model are shown in Figure 3.5.

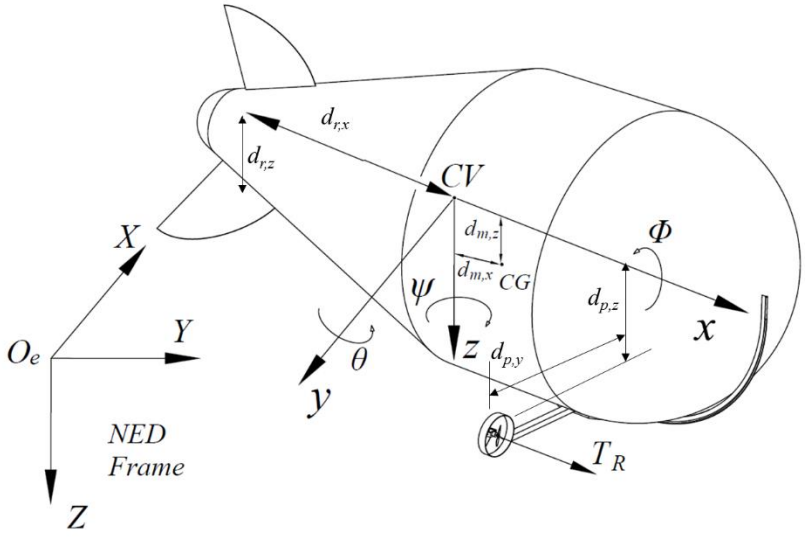


Figure 3.5 Body axis convention and inertia axis of Model 2

The body axis system is defined at the CV of the airship with the *x*-axis along the centerline and pointing to the nose, the *z*-axis is positive downward and the *y*-axis is positive in the direction perpendicular to *xz*-plane as in Figure 3.5. The orientation of the body axis system is represented by Euler (roll $\emptyset$, pitch θ and yaw ψ) angles. The reason of choosing the origin at the CV is due to the possible shift of the center of gravity (CG) during flight.

The equation of motion of the model's reference system in Figure 3.5 is defined as,

$$\mathbf{M}\ddot{\mathbf{x}}_{\mathrm{v}} + \mathbf{D}\left(\dot{\mathbf{x}}_{\mathrm{v}}\right) = \mathbf{E} + \mathbf{U} \tag{3.1}$$

where, **M** is the 6x6 mass matrix, **D** is the 6x1 dynamics vector, **E** is the 6x1 model uncertainty vector, $\mathbf{x}_{\mathrm{v}}$ is the 6x1 state vector of linear and angular velocities expressed in the body frame located at the CV, **U** is the 6x1 external forces and moments and defined as,

$$\mathbf{U} = \mathbf{A}\left(\dot{\mathbf{x}}_{\mathrm{v}}\right) + \mathbf{G}\left(\boldsymbol{R}_{1}\right) + \mathbf{F}_{\boldsymbol{u}} \tag{3.2}$$

where, **A** is the 6x1 aerodynamic vector, **G** is the 6x1 gravitational and buoyancy vector, $\boldsymbol{R}_1$ is the directional cosine matrix and $\mathbf{F}_{\mathrm{u}}$ is the 6x1 propulsive force vector.

The state convention is written as

Positions: $\mathbf{x} = \begin{bmatrix} x & y & z & \phi & \theta & \psi \end{bmatrix}^T$

Velocities: $\dot{\mathbf{x}} = \begin{bmatrix} \mathbf{v} & \boldsymbol{\omega} \end{bmatrix}^T, \quad \mathbf{v} = \begin{bmatrix} \dot{x} & \dot{y} & \dot{z} \end{bmatrix}^T, \quad \boldsymbol{\omega} = \begin{bmatrix} \dot{\phi} & \dot{\theta} & \dot{\psi} \end{bmatrix}^T$

where **v** is vector of linear velocities and **ω** is angular velocities of the vehicle defined in the moving frame of reference located at the CV of the helium envelope. Both the position and the velocity vectors can be defined in either the earth or the vehicle reference frames, which have the subscripts g and v, respectively. In some instances, the repetitive vehicle frame subscripts have been removed for clarity.

3.2.2 Inertia matrix

The mass matrix contains the true mass and inertias of the airship as well as added mass terms. After applying simplifications due to vehicle symmetry, the mass matrix is given by,

$$\mathbf{M} = \begin{bmatrix} \mathrm{M_a} & -m\mathbf{d}_{\mathrm{CG}} \\ m\mathbf{d}_{\mathrm{CG}} & \mathrm{J_a} \end{bmatrix} = \begin{pmatrix} m_x & 0 & 0 & 0 & md_{m,z} & 0 \\ 0 & m_y & 0 & -md_{m,z} & 0 & md_{m,x} \\ 0 & 0 & m_z & 0 & -md_{m,x} & 0 \\ 0 & -md_{m,z} & 0 & J_x & 0 & -J_{xz} \\ md_{m,z} & 0 & -md_{m,x} & 0 & J_y & 0 \\ 0 & md_{m,x} & 0 & -J_{xz} & 0 & J_z \end{pmatrix} \tag{3.3}$$

where $\mathbf{d}_{\mathrm{CG}}$ is a skew symmetric matrix containing distances between the CV and CM. $\mathrm{M_a}$ and $\mathrm{J_a}$ are the added mass and added inertia matrices and are computed from the sum of the true mass and inertia.

$$\mathbf{d}_{\mathrm{CG}} = \begin{bmatrix} 0 & -d_{m,z} & 0 \\ d_{m,z} & 0 & -d_{m,x} \\ 0 & d_{m,x} & 0 \end{bmatrix} \tag{3.4}$$

$$m_x = (1 + k_1)m \tag{3.5}$$

$$m_y = m_z = (1 + k_2)m \tag{3.6}$$

$$J_x = I_{f,x} + I_{g,x} \tag{3.7}$$

$$J_y = (1 + k')I_{f,y} + I_{g,y} \tag{3.8}$$

$$J_z = (1 + k')I_{f,z} + I_{g,z} \tag{3.9}$$

$$J_{xz} = I_{f,xz} + I_{g,xz} \tag{3.10}$$

where the mass of the airship m is equal to the sum of the mass of the gondola, the rail, the envelope, the fins and electric motors

$$m = m_{\mathrm{g}} + m_r + m_e + m_f + m_m$$

The motors are attached to both sides of the gondola therefore, due to the symmetry, the mass center of the gondola includes m_g and m_m.

The centres of masses of each airship are computed as follows:

$$d_{m,x} = \frac{m_g \times s_g + \left(2 \times m_m \times x_m\right) + m_f \times d_{r,x}}{m}$$
$$d_{m,z} = \frac{m_g \times z_g + m_r \times L_r + m_f \times d_{r,z}}{m} \tag{3.11}$$

m_f is considered here the total mass of all fins.

$$\left.\begin{aligned} z_g &= d_{p,z} \\ x_m &= s_g \end{aligned}\right\} \quad \text{straight rail}$$

$$\left.\begin{aligned} z_g &= d_{p,z} \times cos\zeta \\ x_m &= s_g - \left(d_{p,z} \times sin\zeta\right) \end{aligned}\right\} \quad \text{curved rail}$$

where, z_g is the distance between the CV to the CM of the gondola along z, x_m is distance between the CV to CM of the propeller along x, ζ is the angle between the link and z axis when the gondola is in the curved link as shown in Figure 3.6. $d_{r,x}$,*and* $d_{r,z}$, are the distance between the CV to the CM of fins along x and z respectively, and L_r is the distance between the CV to the CM of the rail.

Model 1 the straight rail from $-0.81 \leq s_g \leq 0.96$ and the curved rail when $s_g > 0.96$.

and Model 2 the straight rail from $-0.47 \leq s_g \leq 0.76$ and the curved rail when $s_g > 0.76$.

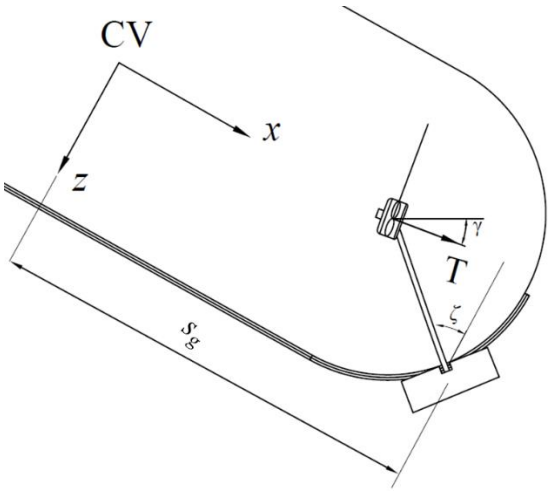


Figure 3.6 the gondola in the curved section

The masses and the inertias of the fixed components as indicated by the subscript *f* represent those of the envelope and rail and were found based on the dimensions of the prototype SolidWorks model, while virtual masses terms k_1, k_2, and k' above are approximated using the works of Lamb [114] and Munk [115]. The geometric and physical properties of the simulated airships are listed in Appendix B.

3.2.3 Dynamics vector

The 6×1 column matrix that contains the dynamic terms associated with inertial linear and angular velocities is given by,

$$\mathbf{D}(\dot{\mathrm{x}}_{\mathrm{v}}) = \begin{bmatrix} D_X & D_Y & D_Z & D_L & D_M & D_N \end{bmatrix}^T \tag{3.12}$$

where the axis convention is defined in Figure 3.5. The forces and moments in equation (3.12) are derived based on Newton's laws for rigid body motion about the CV in the body frame.

The components of the dynamics vector simplified due to the airship symmetry about the xz plane and become [21],

$$D_X = m_z\,\dot{z}\,\dot{\theta} - m_z\,\dot{y}\,\dot{\psi} + m\,(d_{m,x}[\dot{\theta}^2 + \dot{\psi}^2] - d_{m,z}\,\dot{\psi}\,\dot{\phi}) \tag{3.13}$$

$$D_Y = m_x\,\dot{x}\,\dot{\psi} - m_z\,\dot{\phi}\,\dot{z} - m\,(d_{m,x}\,\dot{\phi}\,\dot{\theta} + d_{m,z}\,\dot{\psi}\,\dot{\theta}) \tag{3.14}$$

$$D_Z = m_y\,\dot{y}\,\dot{\phi} - m_x\,\dot{x}\,\dot{\theta} + m\,(d_{m,z}[\dot{\theta}^2 + \dot{\phi}^2] - d_{m,x}\,\dot{\psi}\,\dot{\phi}) \tag{3.15}$$

$$D_L = (J_z - J_y)\,\dot{\psi}\,\dot{\theta} - J_{xz}\,\dot{\phi}\,\dot{\theta} + m\,d_{m,z}(\dot{\phi}\,\dot{z} - \dot{x}\,\dot{\psi}) \tag{3.16}$$

$$D_M = (J_x - J_z)\,\dot{\phi}\,\dot{\psi} + J_{xz}(\dot{\phi}^2 + \dot{\psi}^2) + m\,(d_{m,z}[\dot{z}\,\dot{\theta} - \dot{y}\,\dot{\psi}) + d_{m,\mathrm{x}}[\dot{x}\,\dot{\theta} - \dot{y}\,\dot{\phi}]) \tag{3.17}$$

$$D_N = (J_y - J_x)\,\dot{\theta}\,\dot{\phi} + J_{xz}\,\dot{\theta}\,\dot{\psi} + m\,d_{m,x}(\dot{x}\,\dot{\psi} - \dot{\phi}\,\dot{z}) \tag{3.18}$$

3.2.4 Aerodynamics vector

The aerodynamics vector acts to dampen linear and rotational rates of change in the vehicle. In the present study, the coefficients within the aerodynamic terms were determined from the airship geometry using the system identification method proposed by Mueller et al. who presented an alternative method to system identification using airship geometry [51]. Many aerodynamic models in the literature are linear combinations of terms containing the angle of attack, the angle of sideslip, and the control surface deflections, all of which are multiplied by the square of the trimmed vehicle velocity. These models can be adequate for medium to high speed flight but have no effect at hover since aerodynamic dampening is reduced asymptotically with relative vehicle speed [21]. Gomes [21] and Ashraf et al. [41] present rotational pitch and yaw damping terms but neglect the envelope effects which are on the same order of magnitude as the fins. This research incorporates the aerodynamic model derived based on the work of both Jones and Mueller [51], [116] with additional rotational dampening added. The aerodynamic vector is given by,

$$\mathbf{A} = \begin{bmatrix} A_X & A_Y & A_Z & A_L & A_M & A_N \end{bmatrix}^{\mathrm{T}} \tag{3.19}$$

The components of the aerodynamic vector are defined as:

$$A_X = q\left[C_{X1}\cos^2\alpha\cos^2\beta + C_{X2}\left(\sin(2\alpha)\sin(\alpha/2) + \sin(2\beta)\sin(\beta/2) + C_{X3}\right)\right] \tag{3.20}$$

$$A_Y = q\left[C_{Y1}\cos(\beta/2)\sin(2\beta) + C_{Y2}\sin(2\beta) + C_{Y3}\sin(\beta)\sin\left(|\beta|\right)\right] \tag{3.21}$$

$$A_Z = q\,[C_{Z1}\cos(\alpha/2)\sin(2\alpha) + C_{Z2}\sin(2\alpha) + C_{Z3}\sin(\alpha)\sin\left(|\alpha|\right)] \tag{3.22}$$

$$A_L = q\,[C_{L1}\sin(\beta)\sin\left(|\beta|\right)] + 0.5\rho_a C_{L3}(\dot{\phi}|\dot{\phi}| + \dot{\psi}|\dot{\psi}|) \tag{3.23}$$

$$A_M = q\,[C_{M1}\cos(\alpha/2)\sin(2\alpha) + C_{M2}\sin(2\alpha) + C_{M3}\sin(\alpha)\sin(|\alpha|)] + 0.5\rho_a C_{M5}\dot{\theta}|\dot{\theta}| \tag{3.24}$$

$$A_N = q\left[C_{N1}\cos(\beta/2)\sin(2\beta) + C_{N2}\sin(2\beta) + C_{N3}\sin(\beta)\sin(|\beta|)\right] + 0.5\rho_a C_{N5}\dot{\psi}|\dot{\psi}| \tag{3.25}$$

where the steady state dynamic pressure is defined as $q = \frac{1}{2}\rho_a v_0^2$ and the angles of attack and side slip relative to the air are defined as $\alpha = arctan\left(\frac{\dot{z}}{\dot{x}}\right)$ and $\beta = arcsin\left(\frac{\dot{y}}{\sqrt{\dot{x}^2+\dot{y}^2+\dot{z}^2}}\right)$, respectively. The last term on each aerodynamic moment are the added rotational dampening that are proportional to the angular rates squared. The coefficients of drag are,

$$C_{X1} = -[C_{Dh_0}S_h + C_{Df_0}S_f + C_{Dg_0}S_g] \tag{3.26}$$

$$C_{X2} = -C_{Y1} = -C_{Z1} = (k_2 - k_1)\eta_k I_1 S_h \tag{3.27}$$

$$C_{Y2} = C_{Z2} = -0.5(\frac{\delta C_L}{\delta\alpha})_f S_f \eta_f \tag{3.28}$$

$$C_{Y3} = -[C_{Dch}J_1 S_h + C_{Dcf}S_f + C_{Dcg}S_g] \tag{3.29}$$

$$C_{Y4} = C_{Z4} = -0.5(\frac{\delta C_L}{\delta\alpha})_l S_f \eta_f \tag{3.30}$$

$$C_{Z3} = -[C_{Dch}J_1 S_h + C_{Dcf}S_f] \tag{3.31}$$

$$C_{L1} = C_{Dcg}S_g d_{g,z} \tag{3.32}$$

$$C_{L2} = -2C_{Dcf}S_f d_{f,z}^3 \tag{3.33}$$

$$C_{L3} = -C_{Dcg}S_g d_{gz} D^2 \tag{3.34}$$

$$C_{M1} = -C_{N1} = -(k_2 - k_1)\eta_k I_3 S_h L \tag{3.35}$$

$$C_{M2} = -C_{N2} = -0.5(\frac{\delta C_L}{\delta \alpha})_f S_f \eta_f d_{f,x} \tag{3.36}$$

$$C_{M3} = -(C_{Dch} J_2 S_h L + C_{Dcf} S_f d_{f,x}) \tag{3.37}$$

$$C_{N3} = C_{Dch} J_2 S_h L + C_{Dcf} S_f d_{f,x} + C_{Dcg} S_g s \tag{3.38}$$

$$C_{M4} = -C_{N4} = -0.5(\frac{\delta C_L}{\delta \delta_{E,R}})_l S_f \eta_f d_{f,x} \tag{3.39}$$

$$C_{M5} = -C_{Dcf} S_f d_{f,x}^3 \tag{3.40}$$

$$C_{N5} = -[C_{Dcf} S_f d_{f,\mathrm{x}}^3 - C_{Dcg} S_g s^3] \tag{3.41}$$

All other variables and coefficients are summarized in Appendix B and including references to where they can be determined for any airship platform using a combination of semiempirical models and geometric functions.

3.2.5 Gravitational and buoyancy vector

The gravitational and buoyancy vector accounts for gravitational and buoyancy forces applied to the vehicle. Since these forces align with the axes of the earth reference frame, they must be premultiplied by a rotation matrix to convert them to the body reference frame. The centre of buoyancy is assumed same as the CV. The gravitational and buoyancy vector **G** is given by,

$$\mathbf{G}(\boldsymbol{R}_1) = \begin{bmatrix} m\mathbf{g} \\ m\mathbf{d}_{\mathbf{CG}}\mathbf{g} \end{bmatrix} - \begin{bmatrix} \rho_a V\mathbf{g} \\ \rho_a V\mathbf{d}_{\mathbf{CG}}\mathbf{g} \end{bmatrix} \tag{3.42}$$

where V is the volume of the helium envelope, $\mathbf{d}_{\mathrm{CG}}$ is defined in equation (3.4), and **g** is the gravitational vector expressed in the body frame and $\boldsymbol{R}_1$ is a rotation matrix (also known as the 123 or roll-pitch-yaw Euler sequence),

$$\mathbf{g} = \begin{bmatrix} g_1 & g_2 & g_3 \end{bmatrix}^T = \boldsymbol{R}_1^T \begin{bmatrix} 0 & 0 & g \end{bmatrix}^T \tag{3.43}$$

$$\boldsymbol{R}_1 = \begin{bmatrix} \cos\theta\cos\psi & \sin\phi\sin\theta\cos\psi - \cos\phi\sin\psi & \cos\phi\sin\theta\cos\psi + \sin\phi\sin\psi \\ \cos\theta\sin\psi & \sin\phi\sin\theta\sin\psi + \cos\phi\cos\psi & \cos\phi\sin\theta\sin\psi - \sin\phi\cos\psi \\ -\sin\theta & \sin\phi\cos\theta & \cos\phi\cos\theta \end{bmatrix} \tag{3.44}$$

In component form, the gravitational and buoyancy vector becomes,

$$G_X = (m - \rho_a V) g_1 \tag{3.45}$$

$$G_Y = (m - \rho_a V) g_2 \tag{3.46}$$

$$G_Z = (m - \rho_a V) g_3 \tag{3.47}$$

$$G_L = -m d_{mz} g_2 \tag{3.48}$$

$$G_M = m (d_{mz} g_1 - d_{mx} g_3) \tag{3.49}$$

$$G_N = m\, d_{mx}\, m\, g_2 \tag{3.50}$$

3.2.6 Uncertainty terms

Several parameters in the airship equations of motion, as well as the calculations based on the size and weight measurements of the actual blimp are subject to uncertainty. There is also uncertainty in the model formulation. Additionally, the uncertainty arises from assumptions made about the vehicle aerodynamics and the actuator dynamics. These variations will result in an unknown, higher order aerodynamic model. Therefore, the uncertainty vector is modelled as,

$$\boldsymbol{E} = \sigma_1 (\mathrm{A} + \mathrm{F}_u)^{\mathrm{T}} \tag{3.51}$$

where σ_1 is a 6x1 vector of uniformly distributed random percentages within the range of ±10%. This range was chosen by Recoskie [60] based on the variation between methods to determine empirically derived constants. It was difficult to validate the magnitude and distribution of the

uncertainties [60]. Most of uncertainty parameters cannot directly be measured, and thus requires to be constructed from the measurable airship states. Accuracy of the obtained parameters are highly dependent on the quality of the sensor measurements [117].

3.2.7 Input force vector

The input force vector is expressed in form of 6x1 matrix and contains the propulsive forces and moments. The following Input force equation is derived for the vectored link system and written as,

$$\mathbf{F}_u = \begin{bmatrix} F_x \\ F_y \\ F_z \\ M_x \\ M_y \\ M_z \end{bmatrix} = \begin{bmatrix} (T_R + T_L)\cos\gamma \\ 0 \\ -(T_R + T_L)\sin\gamma \\ (T_R - T_L)\, d_{p,y}\sin\gamma \\ (T_R + T_L)\left[\sin\gamma\ (s_g + d_{p,z}\sin\gamma) - \cos\gamma\ (d_{g,z} + d_{p,z}\cos\gamma)\right] \\ (T_R - T_L)\, d_{p,y}\cos\gamma \end{bmatrix} \tag{3.52}$$

where T_R and T_L are the right and left propeller thrust, γ the vectored thrust angle as shown in Figure 3.6, and $d_{p,y}$ and $d_{p,z}$ are defined in Appendix B for Model 2. However, $d_{p,z}$ is equal to zero in Model 1 because the thrusters are in line with the CV.

3.2.8 Kinematics

The body fixed frame origin was chosen at center of volume (assumed to coincide with the gross center of lift). Vehicle kinematics are applied in tandem to determine the vehicle's velocity and positional changes with respect to the earth reference frame. Coordinates and trajectories in the earth reference frame can be determined from the equivalent states in body reference frame pre-multiplied by the Jacobian matrix J,

$$\dot{\mathbf{x}}_{\mathrm{g}} = \mathrm{J}\dot{\mathbf{x}}_{\mathrm{v}} + \mathbf{v}_{\mathrm{w}} = \begin{bmatrix} \boldsymbol{R}_1 & 0_{3x3} \\ 0_{3x3} & \boldsymbol{R}_2 \end{bmatrix} \dot{\mathbf{x}}_{\mathrm{v}} + \mathbf{v}_{\mathrm{w}} \tag{3.53}$$

where $\boldsymbol{R}_1$ is the rotation matrix described in equation (3.44), $\mathbf{v}_{\mathrm{w}}$ is the total wind (steady + gusts) in a 6x1 vector and, and, $\boldsymbol{R}_2$ is the rotation matrix described by

$$\boldsymbol{R}_2 = \begin{bmatrix} 1 & \sin\phi\tan\theta & \cos\phi\tan\theta \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi\sec\theta & \cos\phi\sec\theta \end{bmatrix} \tag{3.54}$$

3.2.9 Wind disturbances

In order to evaluate the controller robustness in the presence of wind disturbances $\mathbf{v}_w$, the wind disturbances were modeled by using a Dryden model. The disturbances enter into the dynamics of the airship in equation (3.53). The equation (3.53) defined $\dot{\mathbf{x}}_{\mathrm{g}}$ as the velocity of the airship relative to the ground, $\dot{\mathbf{x}}_{\mathrm{v}}$ as the velocity of the airship relative to the body frame, and $\mathbf{v}_{\mathrm{w}}$ as the wind velocity. The wind was modeled as a constant wind field plus turbulence. The turbulence was generated by passing white noise through a filter to create Dryden model. The model was simulated included a turbulent gust with an intensity of 2 m/s, in addition to a wind gust with gust amplitude of 2 m/s. These values are suitable for low and medium altitudes and light and moderate turbulence [118], and represent 33% of the maximum speed of the airship.

The output of the total wind vector resulted by Dryden model is represented as,

$$\mathbf{v}_w = \mathbf{v}_{wc} + \mathbf{v}_{w\mathrm{g}}$$

where $\mathbf{v}_{wc}$ is a constant wind, and $\mathbf{v}_{w\mathrm{g}}$ is a stochastic process that represents wind gusts and other atmospheric disturbances.

The wind velocity $\mathbf{v}_w$ in (3.53).expressed in inertia reference frame is added to the model in the longitudinal motions as,

$$\mathbf{v}_w = \begin{bmatrix} \dot{x}_w \\ \dot{y}_w \\ \dot{z}_w \end{bmatrix}$$

The lateral wind effect on Y axis is neglected in the simulations and experimental tests and it is assumed zero ($\dot{y}_w$=0).

3.3 Energy consumption

Energy consumption is one of the important aspects involved in airship optimization because it directly affects the endurance of the vehicle. Several factors can influence the energy consumption of an airship including the atmospheric environment, payload power requirements, the size of the airship, and the power and propulsion systems' efficiencies [119]. Also, rotational changes require energy and cannot be considered negligible and some maneuvers can even cause instability which can consume additional energy to correct and follow the trajectory [60]. The constraints such as power consumption, weight and size play an important role in UAVs, particularly in small size, light and low cost UAVs. Most existing literature on UAVs focuses on large, high altitude airship platforms used for communication and surveillance [120], [121]. However, the energy efficient and low altitude uninhabited airships have not largely been explored to date [60]. Tseng et al. [122] conducted an empirical study to model the energy consumption of drones, considering various flight scenarios. They observed large power fluctuations due to repeating vertical movements. For long endurance missions or transportation purposes, minimize the weight and power requirements of the configuration is critical in airships. The required propulsive power depends mainly on the aerodynamic drag of the airship hull, which accounts for about two-thirds of the total drag. Even

a small reduction in hull drag can result in a significant energy savings, which in turn, will lead to maximize the efficiency of airships. During the aerodynamic design of an airship, it is important to find a drag-minimized envelope for the long range goal [1]. Gertler and Marzocca conducted experimental work to determine a low-drag submarine contour for conventional and unconventional airships [123]. In addition to this many other researchers have studied the impact of drag reduction of the airship hull [50], [116], [124], [125].

3.3.1 Energy required for the airship

The power required for an airship includes the power needed to supply the sensors, all support systems and in particular the propulsion system. Propulsion system in this study consists of two brushless DC (BLDC) motors with two propellers installed on the shafts of the motors, one motor on each side of the envelope as shown in Figure 3.7. Each thruster is mounted on a tilting system with servo motor. A driver carriage containing DC gear motors and a driven carriage as shown in Figure 3.8. These actuators provide the means to control the position of the gondola on the keel.

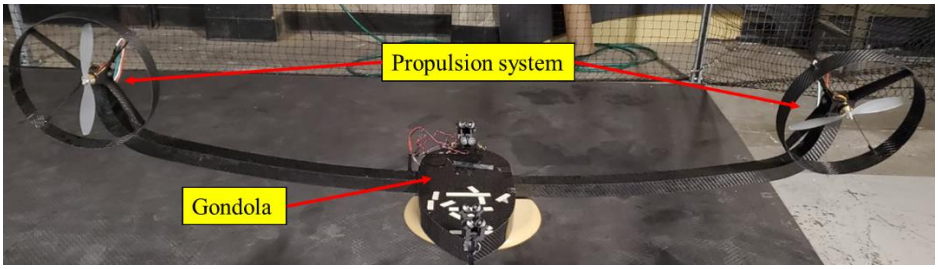


Figure 3.7 The propulsion system

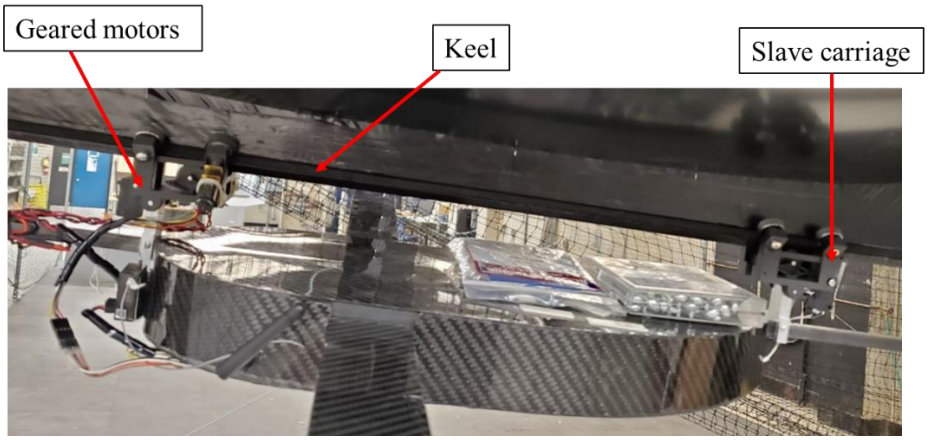


Figure 3.8 The drive mechanism of the airship

The on-board electronics power is a constant electrical power P_c that is consumed by the on-board electronics (camera, minicomputer, telemetry, etc.). The power consumed by the on-board electronics effects the overall flight endurance and can be up to 65% for dirigibles [60].

The total power required for the airship can be then calculated as:

$$P_{tot} = P_m + P_G + P_s + P_c \tag{3.55}$$

where P_m is the power required to overcome the drag, P_G is the power required to move the gondola and P_s is the power required to rotate servo motor. However, not all systems consumed power during the mission objectives. For example, the drive mechanism of the gondola does not consume power when it is not in motion. Unlike the propulsion system, that consumes power for the duration of the flight time since it is the only system that generates thrust to move the airship to overcome the drag.

Therefore, the total energy required E_{tot} for an airship during the trajectory from the start time t_{start} and t_{end} can be calculated as:

$$E_{tot} = \sum_i P_i \cdot \Delta t_i \tag{3.56}$$

where i is the different system that consumes power, Δt_i the amount of time that each system operates during the whole trajectory.

3.3.2 Propulsion system power

In order to calculate the required energy, the airship drag force acting on the airship at the maximum operating speed (V) needs to be calculated. The drag force F_d is calculated as:

$$F_d = \frac{1}{2}\rho_{air}\, C_{DV} V_e^{2/3}\, V^2 \tag{3.57}$$

where ρ_{air} the density of the air (kg/m^3), V_e is the volume of the envelope (m^3), C_{DV} the drag coefficient of the entire airship can be calculated as:

$$C_{DV} = \frac{C_{DVe}}{k_d} \tag{3.58}$$

where C_{DVe} is the volumetric drag coefficient of an envelope and can be estimated using the semiempirical equation suggested by Cheeseman [16] as:

$$C_{Dve} = \frac{0.172\left(\sqrt[3]{\left(\frac{L}{D}\right)}\right) + \left(\frac{0.256}{(L/D)_e^{1.2}}\right) + \left(\frac{1.032}{(L/D)_e^{2.7}}\right)}{R_e^{1/6}} \tag{3.59}$$

and k_d is the drag factor and is found in literature [126] to be 0.5243 and L and D is the length and diameter of the airship respectively. R_e is the Reynolds number and it is expressed as:

$$R_e = \frac{\rho_{air}\, V_w\, D}{\delta_{air}} \tag{3.60}$$

where V_w is velocity of the wind and δ_{air} is dynamic viscosity of the air.

Once the total airship drag force has been calculated, the actual power required P_m (Watt) for the motors to overcome the drag of airship can be determined as:

$$P_m = F_d \cdot V / \eta_p \tag{3.61}$$

η_p is the propulsion system efficiency which is considered 75% based on thruster static bench testing.

3.3.3 Gondola's drive mechanism power

There are two BLDC motors to move the gondola. The power required to move the gondola P_G along the rigid keel based on satisfaction of the needs of the mission, is calculated as:

$$P_G = F_r \cdot \dot{S} \tag{3.62}$$

where F_r is the rolling force from the motors that moves the gondola along the keel (friction force required to rotate the wheel), $\dot{S}$ is the speed of the gondola. The rolling force is calculating by using Newton's second law. The simplified forces acting on the gondola are shown in Figure 3.9.

$$F_r = m(\ddot{S} + \mathrm{g}\ \sin\theta + \frac{1}{5}\mathrm{g}\ \cos\theta\,\mu_s + \frac{4}{5}\ \mathrm{g}\ \cos\theta\,\mu_{roll}) \tag{3.63}$$

where m is the total mass of the moving parts (Payload weight) attached to the gondola, $\ddot{S}$ is the acceleration of the gondola in the direction along the keel and θ is the pitch angle in the body frame. μ_s is friction coefficient between the rubber wheels and the keel and μ_{roll} is the rolling coefficient between the steel wheels and the keel. The weight carried by each carriage assembly is approximated by (W/2).

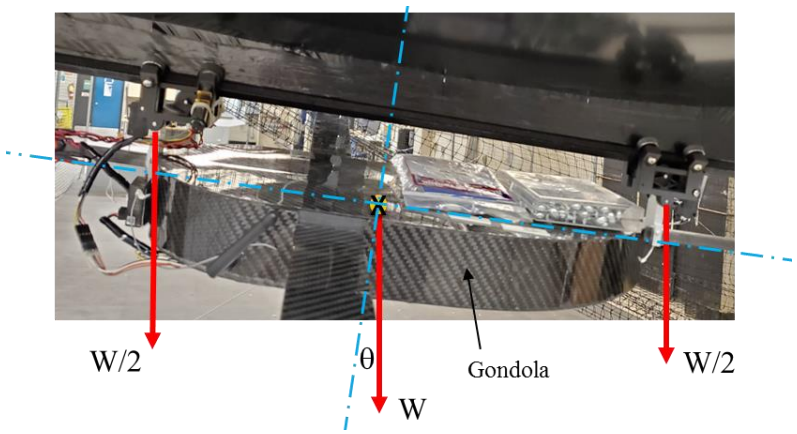


Figure 3.9 Free body diagram of forces acting on the gondola and the carriages

The power required to move the gondola P_G by two motors can be obtained by substituting equation (3.63) in equation (3.62),

$$P_G = m(\ddot{S} + g\ \sin\theta + \frac{1}{5} g\ \cos\theta\, \mu_s + \frac{4}{5}\ g\ \cos\theta\, \mu_{roll}) \cdot \dot{S} \qquad (3.64)$$

The motor power is a linear function of the gondola speed and thus is only required during gondola repositioning. Equation (3.64) shows that the power required to move the gondola is independent of velocity of the airship. However, the primary factors affecting energy consumption are the payload weight and the gondola repositioning rate $\dot{S}$.

3.3.4 Tilting system power

The propulsion system (BLDC motor paired with propeller) is mounted on a tilting system as shown in Figure 3.10. The servo angles are defined with respect to the x axis (horizontal) of the body frame and the negative direction of z axis which defines positive rotations. The tilting system with the propulsion system can produce pitching moment, lift, and forward thrust at the same time, whereas differential thrust can generate rolling and yawing moments. The power required to rotate servo motor P_s as required by the mission, is calculated as:

$$P_s = \eta \,.\, \tau \cdot \omega_s \tag{3.65}$$

where τ is the torque required to rotate the propulsion system weight, which equals to the product of the weight and the distance between the center of mass of the propulsion system and the center of rotation of the motor. The angular velocity of the servo motor ω_s (deg/s) is defined in the data sheet of the motor. η is the efficiency that account for friction losses in the gears and tilting bracket and it is considered to be constant in this study η = 0.80.

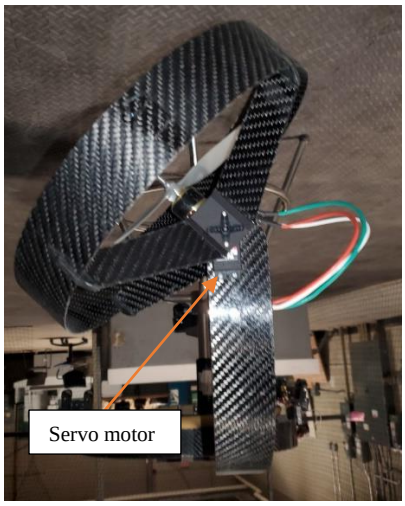


Figure 3.10 the tilting system

Chapter 4: **Airship Control**

Backstepping and incremental nonlinear dynamic inversion controllers were designed for the purpose of tracking a desired trajectory by controlling the position of the gondola and the thrust. The performance of both controllers was simulated on both airship models then the energy consumption of the prototype Model 2 was evaluated. The following sections describe the theory and implementation details of each controller followed by simulation and experimental results.

4.1 Backstepping controller

The backstepping method is a nonlinear approach which uses Control Lyapunov Functions (CLF). The nonlinear system is broken down into smaller subsystems and the Lyapunov function is applied for each of the subsystems. The backstepping approach can be applied to multi input multi output systems since the control law is derived in few steps [127]. Global asymptotic stability of the tracking control law is proven by applying Lyapunov stability analysis

The first backstepping variable e_1 is defined as

$$e_1 = \mathbf{x}_g - \mathbf{x}_d \tag{4.1}$$

where $\mathbf{x}_g$ is the 6x1 airship's position vector in the earth fixed frame and $\mathbf{x}_d$ is the 6x1 desired state. The virtual control is chosen by introducing the second backstepping variable e_2.

$$\dot{\mathbf{x}}_g = J\dot{\mathbf{x}}_v = e_2 + \alpha_1 \tag{4.2}$$

where α_1 is a stabilizing function and J is the Jacobian matrix that defined in (3.53). The time differentiation of equation (4.1) and substitution of (4.2) gives

$$\dot{e}_1 = \dot{\mathbf{x}}_g - \dot{\mathbf{x}}_d = J\dot{\mathbf{x}}_v - \dot{\mathbf{x}}_d = e_2 + \alpha_1 - \dot{\mathbf{x}}_d \tag{4.3}$$

Let us consider the Lyapunov function candidate V_1 for the first step to be,

$$V_1 = \frac{1}{2} e_1^T e_1 \tag{4.4}$$

and its time derivative is

$$\dot{V}_1 = \frac{1}{2} e_1^T \dot{e}_1 \tag{4.5}$$

To make $\dot{V}_1$ negative definite with respect to e_1, the stabilizing function can be selected as

$$\alpha_1 = \dot{\mathbf{x}}_d - K_1 e_1 \tag{4.6}$$

Where $K_1 = K_1^T > 0$ is a 6X6 positive definite design matrix. By substituting equations (4.3) and (4.6) in equation (4.5)

$$\dot{V}_1 = e_1^T e_2 - e_1^T K_1 e_1 \tag{4.7}$$

Equations (4.2) and (4.3) are rearranged as

$$\dot{e}_1 = e_2 - K_1 e_1 \tag{4.8}$$

$$e_2 = J\dot{\mathbf{x}}_v - \dot{\mathbf{x}}_d + K_1 e_1 \tag{4.9}$$

The time derivative of e_2 is

$$\dot{e}_2 = \dot{J}\dot{\mathbf{x}}_v + J\ddot{\mathbf{x}}_v - \ddot{\mathbf{x}}_d + K_1 \dot{e}_1 \tag{4.10}$$

The second step of the backstepping controller is designed to drive the virtual state e_2 as defined in (4.9) to zero. Let us consider the second Lyapunov function candidate as

$$V_2 = V_1 + \frac{1}{2} e_2^T e_2 \tag{4.11}$$

Its time derivative is

$$\dot{V}_2 = \mathrm{e}_1^{\mathrm{T}}\mathrm{e}_2 - \mathrm{e}_1^{\mathrm{T}}K_1\mathrm{e}_1 + \mathrm{e}_2^{\mathrm{T}}\dot{\mathrm{e}}_2 \tag{4.12}$$

Taking the time derivative of e_2 and substituting $\dot{\mathbf{x}}_{\mathrm{v}}$ from equation (3.1) to equation (4.12)

$$\dot{V}_2 = \mathrm{e}_1^{\mathrm{T}}\mathrm{e}_2 - \mathrm{e}_1^{\mathrm{T}}K_1\mathrm{e}_1 + \mathrm{e}_2^{\mathrm{T}}(\dot{\mathrm{J}}\dot{\mathbf{x}}_{\mathrm{v}} + \mathrm{J}\,\mathbf{M}^{-1}[-\mathbf{D}(\dot{\mathbf{x}}_{\mathrm{v}}) + \mathbf{U}] - \ddot{\mathbf{x}}_{\mathrm{d}} + K_1\dot{\mathrm{e}}_1) \tag{4.13}$$

The time derivative the second Lyapunov function $\dot{V}_2$ is negative definite in e_1 and e_2, if the terms between brackets are equal to ($-\mathrm{e}_1$ $-K_2$ e_2), where $K_2 = K_2^T > 0$ is another 6x6 positive definite design matrix. If the virtual control is chosen such that,

$$\dot{\mathrm{e}}_2 = -\mathrm{e}_1 - K_2\mathrm{e}_2 \tag{4.14}$$

and the time derivative of second Lyapunov candidate function becomes

$$\dot{V}_2 = -\mathrm{e}_1^{\mathrm{T}}K_1\mathrm{e}_1 - \mathrm{e}_2^{\mathrm{T}}K_2\mathrm{e}_2 \tag{4.15}$$

The second Lyapunov function $\dot{V}_2$ can be made negative definite by choosing the positive definite matrices K_1 and K_2. Thus, according to the Lyapunov stability theory, the airship system is globally asymptotically stable.

By setting equation (4.13) equal to equation (4.15) and substituting $\ddot{\mathbf{x}}_{\mathrm{v}}$ from equation (3.1) and then solving for the desired control force **U** we get

$$\mathbf{U} = \mathbf{M}\mathrm{J}^{-1}\left(-K_2\mathrm{e}_2 - \mathrm{e}_1 - \dot{\mathrm{J}}\dot{\mathbf{x}}_{\mathrm{v}} + \ddot{\mathbf{x}}_{\mathrm{d}} - K_1\mathrm{e}_2 + K_1^2\mathrm{e}_1\right) + [\mathbf{D}(\dot{\mathbf{x}}_{\mathrm{v}})] \tag{4.16}$$

The uncertainty terms are removed from the desired control because they are unpredictable. However, they are added as 10% error to the virtual masses since they are approximated using the works of Lamb [114]

The control allocation problem with aerodynamic forces and moments can be formulated according to the control law of equation (4.16) and equation (3.2).

$$\mathbf{U} = \begin{bmatrix} F^T & M^T \end{bmatrix}^T \tag{4.17}$$

where the external forces $F = [u_1\ u_2\ u_3]^T$ and the external moments $M = [u_4\ u_5\ u_6]^T$. By substituting the external forces and the external moments from equation (3.2) in equation (4.17) we get,

$$\mathbf{U} = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \end{bmatrix} = \begin{bmatrix} (T_R + T_L)cos\gamma & + & G_X \\ 0 & + & G_Y \\ -(T_R + T_L)sin\gamma & + & G_Z \\ (T_R - T_L).d_{p,y}.sin\gamma & + & G_L \\ (T_R + T_L).[sin\gamma(s_g + d_{p,z}.sin\gamma) - cos\gamma(d_{g,z} + d_{p,z}.cos\gamma)] & + & G_M \\ (T_R - T_L).d_{p,y}.cos\gamma & + & G_N \end{bmatrix} \tag{4.18}$$

For simplicity, we set $T_R = T_L$ *and* $F_x = (T_R + T_L)\ cos\ \gamma$, and $F_z = (T_R + T_L)\ sin\ \gamma$. In addition, the desired roll and yaw angles are chosen here to be equal to zero and the airship is symmetric about the x-z plane. Therefore, $u_2 = u_4 = u_6 = 0$.

The real control inputs, $\mathbf{u} = [s_g, F_x, F_z]^T$. Then the equation become,

$$\mathbf{U=Bu}$$

where **B** contains the terms directly related with **u**. Using basic trigonometric geometry γ can be found from the relation between F_x, F_z.

$$\gamma = tan^{-1}\left(\frac{-F_z}{F_x}\right)$$

The block diagram of the backstepping controller is shown in Figure 4.1.

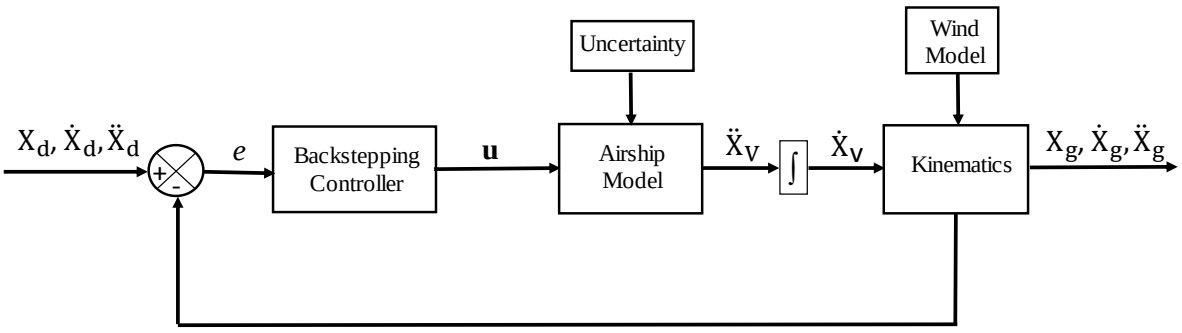


Figure 4.1 Block diagram of the backstepping controller

4.2 Incremental Nonlinear Dynamics Inversion Approach

The controller solves the incremental form of equations of motion using acceleration feedback and generates a control law substantially reducing the controller's dependence on complex, and sometimes inaccurate, models such as those describing the aerodynamic phenomena in flight dynamics. The INDI controller only requires knowledge of the kinematics and the actuator dynamics, and replaces the dynamic model with measurements of the system dynamics [72]. The foundation of INDI is that only small portions of the model (control derivatives) are required for the design of the controller. Instead of computing the total control input directly, the INDI methodology calculates the increment of the control input for every sample time based on the system states at the previous time step. This concept has been used for the purpose of coping with nonlinear control derivatives, and to reduce the impact of model mismatch [73].

The INDI methodology is applied to the trajectory tracking of the vehicle architectures proposed in this study. The controller developed in [48] was modified for longitudinal motion, and the effect of the moving gondola for rapid altitude changes for ascending or landing an airship is studied and discussed here.

To avoid reader confusion, the variables in this section will be used in the same way that they are used in literature [79], [128–130].

The variables for describing the INDI methodology are:

$\dot{\mathbf{X}}$ state acceleration

$\dot{\mathbf{X}}_0$ state acceleration at the previous time step

$\dot{x}$ the velocity

x the position

$\mathbf{u}$ the control input

$\mathbf{u}_0$ the control input at the previous time step

v_r virtual control variable

The subscript (ω) represents the rotational equations of motion

The subscript (p) represents the linear equations of motion.

The (INDI) controller can be derived from a general nonlinear system:

$$\dot{\mathbf{X}} = f(\mathbf{X}) + G(\mathbf{X})\mathbf{u} \tag{4.19}$$

where $\dot{\mathbf{X}}$ is state acceleration, $\mathbf{X} \in \mathbb{R}^n$ is the state vector, $\mathbf{u} \in \mathbb{R}^m$ is the control input, and $G \in \mathbb{R}^{nxm}$ is the control effectiveness matrix. A standard Taylor series expansion provides the following first-order approximation of $\dot{\mathbf{X}}$ at the previous state and control $\mathbf{X}_0$, $\mathbf{u}_0$ respectively,

$$\dot{\mathbf{X}} = \mathrm{f}(\mathbf{X_0}) + \mathrm{G}(\mathbf{X_0})\mathbf{u_0} + \frac{\partial}{\partial \mathrm{x}}\left[\mathrm{f}(\mathbf{X}) + \mathrm{G}(\mathbf{X})\mathbf{u}\right]\Big|_{\substack{\mathrm{x}=\mathrm{x}_0 \\ \mathrm{u}=\mathrm{u}_0}} (\mathbf{X} - \mathbf{X_0}) + \frac{\partial}{\partial \mathrm{u}}\left[\mathrm{G}(\mathbf{X})\mathbf{u}\right]\Big|_{\substack{\mathrm{x}=\mathrm{x}_0 \\ \mathrm{u}=\mathrm{u}_0}} (\mathbf{u} - \mathbf{u_0}) + \mathrm{H.O.T} \tag{4.20}$$

Assuming a high sampling frequency, the changes of state variation $\mathbf{X} \approx \mathbf{X_0}$ of equation (4.20) and (H.O.T) the higher order terms can be neglected. The dynamics of the previous state $\dot{\mathbf{X}}_0$ are:

$$\dot{\mathbf{X}}_0 = \mathrm{f}(\mathbf{X_0}) + \mathrm{G}(\mathbf{X_0})\mathbf{u_0} \tag{4.21}$$

$$H_u = \frac{\partial}{\partial \mathrm{u}}\left[\mathrm{G}(\mathbf{X})\mathbf{u}\right]\Big|_{\substack{\mathrm{x}=\mathrm{x}_0 \\ \mathrm{u}=\mathrm{u}_0}}$$

H_u is the partial derivative of [G(**X**)**u**] at $\mathbf{X} = \mathbf{X}_0$ and $\mathbf{u} = \mathbf{u}_0$. The incremental form of the dynamic equation is thus simplified as,

$$\dot{\mathbf{X}} = \dot{\mathbf{X}}_0 + H_u(\mathbf{u} - \mathbf{u_0}) \tag{4.22}$$

Equation (4.22) is a linear approximation of the equation of motion around $\mathbf{u}_0$ and $\mathbf{X}_0$ for small time increments, describing the changes in angular accelerations as a function of control increments. By setting $\dot{\mathbf{X}}$ is the virtual control variable $\boldsymbol{\nu_r}$, the controller can then be designed in the incremental form as,

$$\mathbf{u} = \mathbf{u_0} + H_u^{-1}(\boldsymbol{\nu_r} - \dot{\mathbf{X}}_0) \tag{4.23}$$

where $\dot{\mathbf{X}}_0$ is assumed to be measurable and the input control matrix H_u has to be invertible. The total input can be obtained by adding the previous input $\mathbf{u}_0$ to the calculated increment

$$\mathbf{u} = \mathbf{u_0} + \Delta\mathbf{u}$$

It is important to notice that this implicit control law design is not entirely independent on the model since changes in f(**X**) are reflected in measurements of $\dot{\mathbf{X}}_\mathbf{0}$. Therefore, this implicit control law design is more dependent on accurate measurements or accurate estimates of $\dot{\mathbf{X}}_\mathbf{0}$ the acceleration, and $\mathbf{u}_0$ the control input.

The acceleration reference for the INDI controller $\boldsymbol{v_r}$ is obtained as follow. The position error is calculated from the difference between the reference trajectory and the feedback of the vehicle's position, and then the desired velocity ($\dot{\boldsymbol{x}}_d$) towards a goal position is calculated by a proportional controller by multiplying the position error with a gain k_{px}.

$$\dot{\boldsymbol{x}}_{\boldsymbol{d}} = \left(\boldsymbol{x}_{\boldsymbol{d}} - \boldsymbol{x}\right) k_{px}$$

Finally, the velocity error, resulting from subtracting the desired velocity from the feedback velocity of the vehicle, is then multiplied with a gain k_{pv} to calculate the acceleration reference.

$$\boldsymbol{v_r} = \left(\dot{\boldsymbol{x}}_{\boldsymbol{d}} - \dot{\boldsymbol{x}}\right) k_{pv} \tag{4.24}$$

The proportional control is used to correct for errors in position and the velocity and provides the acceleration reference for the INDI controller. Then (4.23) becomes,

$$\mathbf{u} = \mathbf{u_0} + H_u^{-1}\left(\left[\left(\boldsymbol{x}_{\boldsymbol{d}} - \boldsymbol{x}\right) k_{px} - \dot{\boldsymbol{x}}\right] k_{pv} - \dot{\mathbf{X}}_\mathbf{0}\right) \tag{4.25}$$

The equation shows that the dependency of the closed-loop system on the accurate knowledge of the airship model is largely decreased, improving robustness against model uncertainties. Therefore, the changes in f(**X**) are reflected in $\dot{\mathbf{X}}_0$, and the control mainly required the measurements of $\dot{\mathbf{X}}_0$ and $\mathbf{u}_0$, making this control approach more dependent on the sensor measurements.

Usually, the acceleration $\dot{\mathbf{X}}_0$ is not measured directly and the sensors are not widely available, especially for small uninhabited aircrafts [79]. Different filtering approaches have been used to solve the problem of indirect measurement of the accelerations. According to [76] the acceleration $\dot{\mathbf{X}}_0$ can be calculated by passing the velocity $\dot{\boldsymbol{x}}$ through a second order low pass filter. Two control loops are designed to control the position and the orientation of the airship.

4.2.1 Translational Control Loop

In translational control loop, the vehicle's flight position is controlled by the vectored thrust $\mathbf{F}_u$ to track the reference trajectory. The vectored thrust is defined as,

$$\mathbf{F}_u = \begin{bmatrix} F_x \\ F_y \\ F_z \end{bmatrix} = \begin{bmatrix} T_R\ cos\gamma_R + T_L cos\gamma_L \\ 0 \\ -T_R sin\gamma_R + T_L sin\gamma_L \end{bmatrix} \tag{4.26}$$

where F_x, F_y, F_z are the components of the thrust in body fixed frame in x, y and z axes, respectively, and defined here as the control variables. T_R and T_L are the right and left propeller thrust, and γ_R and γ_L the right and left angles of the thrusters. The thrust component in y-axis F_y, is considered in this study equal to zero.

$$\mathbf{u}_F = \begin{bmatrix} F_x \\ F_z \end{bmatrix},\ g_1(\mathbf{X}) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad \text{thus} \quad \mathbf{F}_u = g_1(\mathbf{X})\ \mathbf{u}_F$$

Equation (3.1) can be rewritten into the following control affine form of equation (4.19) for the linear motion,

$$\ddot{\mathbf{x}}_v = \mathbf{M}^{-1}[\mathbf{A}(\dot{\mathbf{x}}_v) + \mathbf{G}(\boldsymbol{R}_1) - \mathbf{D}(\dot{\mathbf{x}}_v)] + \mathbf{M}^{-1}\ \mathbf{F}_u$$

$$\mathrm{f}\ (\mathbf{X}_v) = \mathbf{M}^{-1}[\mathbf{A}(\dot{\mathbf{x}}_v) + \mathbf{G}(\boldsymbol{R}_1) - \mathbf{D}(\dot{\mathbf{x}}_v)]\ ,\ H_{u1} = \mathbf{M}^{-1}\ g_1(\mathbf{X}) \tag{4.27}$$

where $\ddot{\mathbf{x}}_{\mathbf{v}} = [\ddot{\mathbf{x}}_{\mathbf{x}} \quad \ddot{\mathbf{x}}_{\mathbf{z}}]^{\mathbf{T}}$

The incremental form of the translational control is,

$$\mathbf{u}_F = \mathbf{u}_{F0} + \Delta \mathbf{u}_F \tag{4.28}$$

where $\mathbf{u}_{F0}$ is the previous input. The airship's flight path control can be calculated by using the translational equation part of motion (3.1) and (4.23). The subscript (v) in (4.27) represents the linear equations of motion used to distinguish between the linear and rotational equations.

4.2.2 Pitch Control Loop

The objective of this control loop is to control the pitch angle by controlling the pitching moment that generated by moving the gondola's position. The angular rate is controlled by the moments of gravitational and buoyancy vector **G** due to the movement of the gondola

$$\mathbf{G}_M = \begin{bmatrix} G_\phi \\ G_\theta \\ G_\psi \end{bmatrix} \tag{4.29}$$

where G_ϕ and G_ψ are the moment components of gravitational and buoyancy vector around the x axis and z axis in body frame respectively, and are considered here equal zero in order to study the effect of the gondola position on the altitude control in the x-z plane. G_θ is the moment component of gravitational and buoyancy vector around the y axis. The gravitational and buoyancy forces applied to the vehicle must be premultiplied by a rotation matrix to convert them to the body reference frame. After some manipulations, the final format of G_θ is,

$$G_\theta = [R\,(3,1) \times \mathrm{g}\,(d_{m,z}\,(m - m_\mathrm{g}) + m_\mathrm{g}\,d_{p,z})] - [R(3,3) \times \mathrm{g}\,(d_{m,x}\,(m - m_\mathrm{g}) + m_\mathrm{g}\,s_\mathrm{g})] \tag{4.30}$$

where $d_{m,x}$ and $d_{m,z}$ are defined in equation (3.11) and it is a function of s_g. The input s_g can be calculated by substituting $d_{m,x}$ and $d_{m,z}$ in equation (4.30) and solve for s_g.

$$G_\theta = g_2(\mathbf{x})\,\mathbf{u}_g$$

In this loop, the control variable is the gondola's position $\mathbf{u}_\mathrm{g} = s_\mathrm{g}$ and $g_2(\mathbf{X})$ contains the terms directly related with $\mathbf{u}_\mathrm{g}$. The equation shows that the movement of the moving mass s_g is the primary factor affecting the pitching moment.

Equation (3.1) can be rewritten into the following control affine form equation (4.19) for the angular motion,

$$\ddot{\mathbf{x}}_\omega = \mathbf{M}^{-1}[\mathbf{A}(\dot{\mathbf{x}}_\omega) + \mathbf{F}_u - \mathbf{D}(\dot{\mathbf{x}}_\omega)] + \mathbf{M}^{-1}\mathbf{G}(\boldsymbol{R}_1)$$

$$\mathrm{f}(\mathbf{x}_\omega) = \mathbf{M}^{-1}[\mathbf{A}(\dot{\mathbf{x}}_\omega) + \mathrm{F}_u - \mathbf{D}(\dot{\mathbf{x}}_\omega)]\,,\; H_{u2} = \mathbf{M}^{-1}\,g_2(\mathbf{x}) \tag{4.31}$$

The incremental form of the longitudinal control is,

$$s_g = s_{g0} + \Delta s_g \tag{4.32}$$

The airship's pitch angle control can be calculated by using the rotational part equation of motion (3.1) and equation (4.23). In case of the airship Model 2, the thrust location is below the CV that generates a pitching moment, therefore the term M_y in (3.52) is added to (4.30) in order for its effect to be taken into consideration. The subscript (ω) represents the rotational equations of motion used to distinguish between the linear and rotational equations. The angular acceleration

$\dot{\mathbf{x}}_{\omega 0}$ can be measured by onboard sensors or by passing the angular velocity $\dot{\omega}$ through a second order low pass filter.

Using a filter leads to a delay which should be compensated. However, the measured acceleration in both loops were filtered by a second order low pass filter equation to keep all signals synchronized in control equations and to prevent performance degradation of the controller due to the time delay [76]. The closed loop of the INDI control system is shown in Figure 4.2.

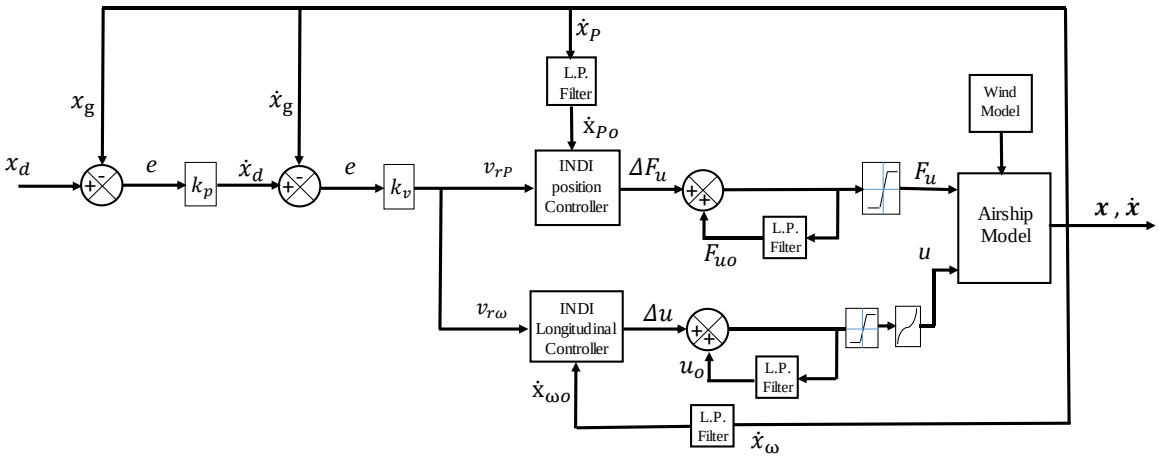


Figure 4.2 Block diagram of INDI control system

Chapter 5: **Numerical Simulation**

Dynamic simulations were developed in the Matlab/Simulink environment to evaluate the effectiveness of the nonlinear dynamic model discussed in Section 3.2. The Runge-Kutta solver was used in the simulation with a fixed time step of 0.1 seconds. The reference trajectory is described by the red line in Figure 5.1. It was filtered using a second order filter to smoothen the corners, as described by the blue line in Figure 5.1, and avoid aggressive changes in states similar to [131]. The motion is only in the longitudinal plane in order to test ascent, descent and cruise manoeuvres. The trajectory was generated to produce the main flight trajectory modes (ascent, cruise flight, descent) and to illustrate the effectiveness of the designed controllers at tracking the trajectory in the different flight modes. The negative sign in the z-axis represents the ascending flight mode in the inertia frame.

In the simulation process, the thrusters are equal on both sides (left and right) and they are each equal to T ($T_R = T_L$ = T). The propeller thrust is limited to -0.26 N < T < 2.6 N, based on the thruster static bench testing prototype presented in Appendix G. The initial CV position of the airship is assumed to be at sea level, (x_g, y_g, z_g) = (0, 0, 0) m in the inertial reference frame and the initial desired roll. Pitch and yaw angles are chosen to be equal to zero ($\phi = \theta = \psi$ =0). The simulation is run for 75 seconds, starting with an initial velocity of v_o = 0 m/s, an initial thrust value of zero and the initial position of the gondola at s_g = 0.

Trajectory tracking results are obtained in an ideal no-wind environment as well as in the presence of wind. The wind model used in the simulation is based on the Dryden model, as discussed in Section 3.2.9.

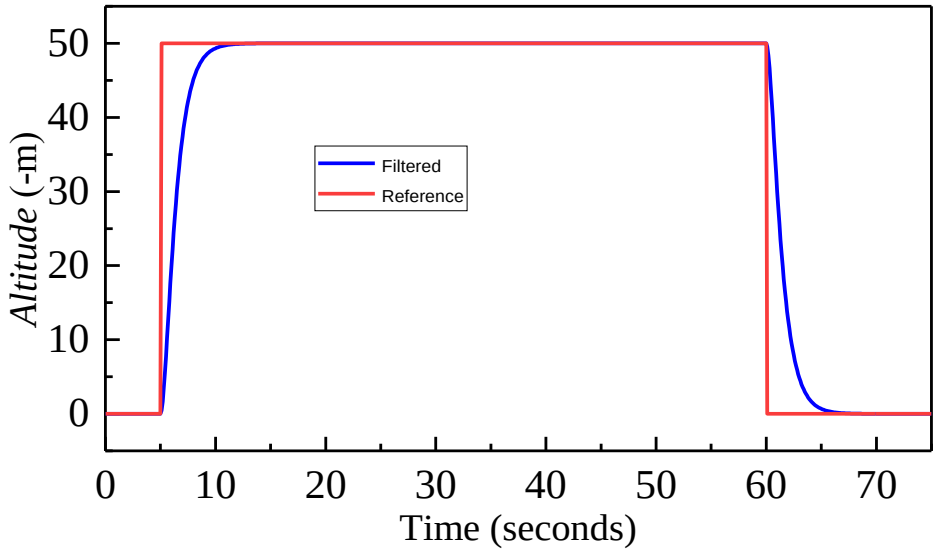


Figure 5.1 Reference and filtered trajectories

5.1 The dynamic simulation Model 1

The closed loop of the backstepping control system is shown in Figure 5.2, where the real control input is calculated from the control command **U**, in Equation (4.16). A rate limiter was added to the closed loop system, as was output saturation, in order to respect the geometry and vehicle capability and avoid saturation. A maximum gondola repositioning rate of $\dot{s}_g$ = 0.3 m/s and a limit of -0.8 < s_g< 1.4 m were applied. The controller gains were selected based on trial and error to obtain a desired closed loop response (K_1 = 9 x I_{6x6}, K_2 = 23 x I_{6x6}).

The real control inputs are: $\mathbf{u} = [s_g, F_x, F_z]^T$.

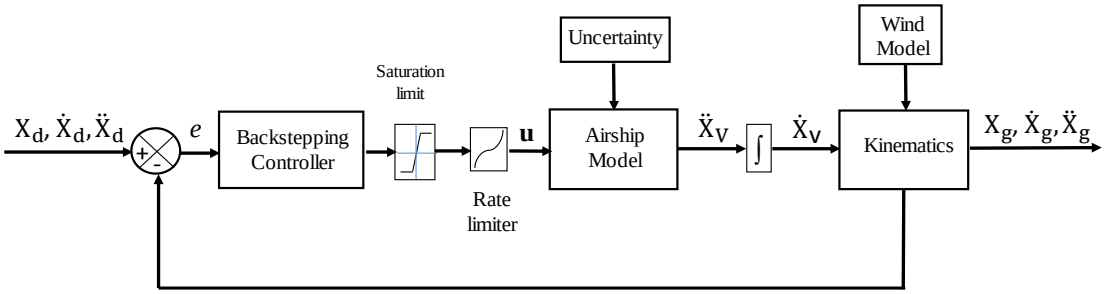


Figure 5.2 Block diagram of backstepping control system

Figure 5.3 shows the tracking performance of the proposed controller during the no-wind simulation. The figure shows a smooth horizontal motion for 5 seconds, starting from x = 0 m. There is then a change in altitude, which causes small oscillations due to the motion of the gondola, as seen in the x position curve between t = 5 seconds and t = 9 seconds during the ascent. The cruise mode starts at t = 10 seconds until t = 60 seconds and the desired altitude of 50 m is successfully tracked. The airship then descends, and again small oscillations are observed between t = 60 seconds and t = 65 seconds. The results of the gondola's position control are shown in Figure 5.4 and the results of the pitch responses to the movements of the gondola, in degrees, are shown in Figure 5.5. The gondola moves quickly backwards at the start of the simulation to adjust to the small differences between the desired and actual pitch angles and keeps almost below the CG to track the 0^o reference angle for approximately 5 seconds. Then, the airship quickly adjusts to the rapid changes in the desired pitch angle caused by the variations in altitude.

As a result, the controller moves the gondola backward at the fastest rate possible and holds the position at the saturation limit of -0.81 cm, for about three seconds, to follow the reference input. The reference angle returns to zero once the reference altitude is reached, allowing the gondola to quickly adjust to changes in desired pitch and move forward to almost s_g = 0 cm. The airship tracks the reference trajectory with only a small overshoot during the change in altitude at the end of the descent and ascent modes, as seen Figure 5.5. The oscillatory motions seen during the ascent and the descent are caused by the rapid movement of the gondola and the aerodynamic drag induced by the changes in pitch in (3.24). Figure 5.4 shows that the gondola was able to move from s_g = -0.81 m to s_g = 1.2 m and generate pitch angles of 73^o and -88^o respectively.

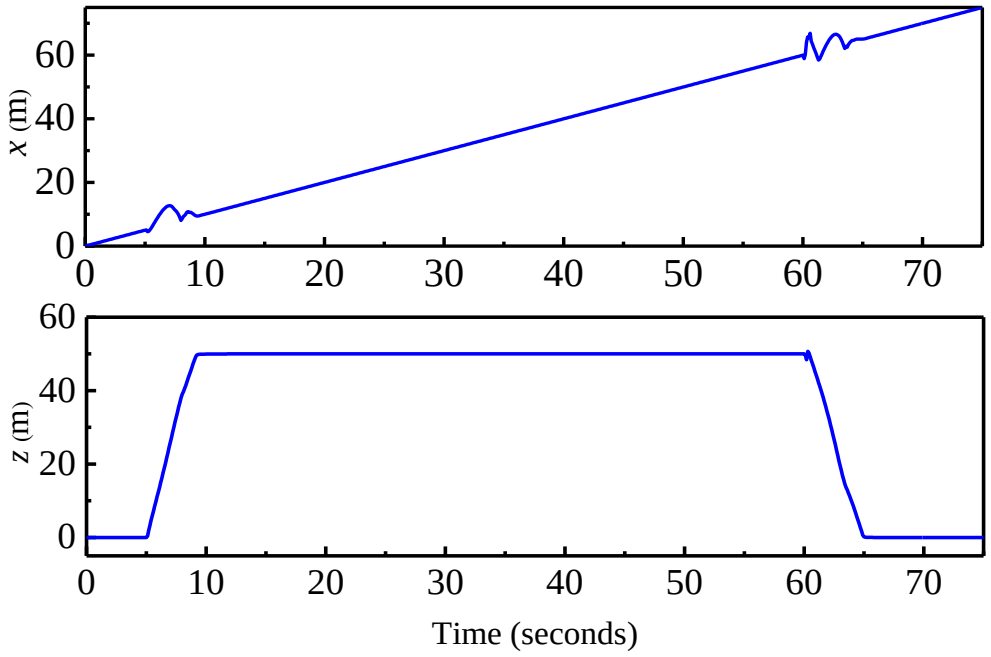


Figure 5.3 Airship position Model 1 using backstepping control with no-wind

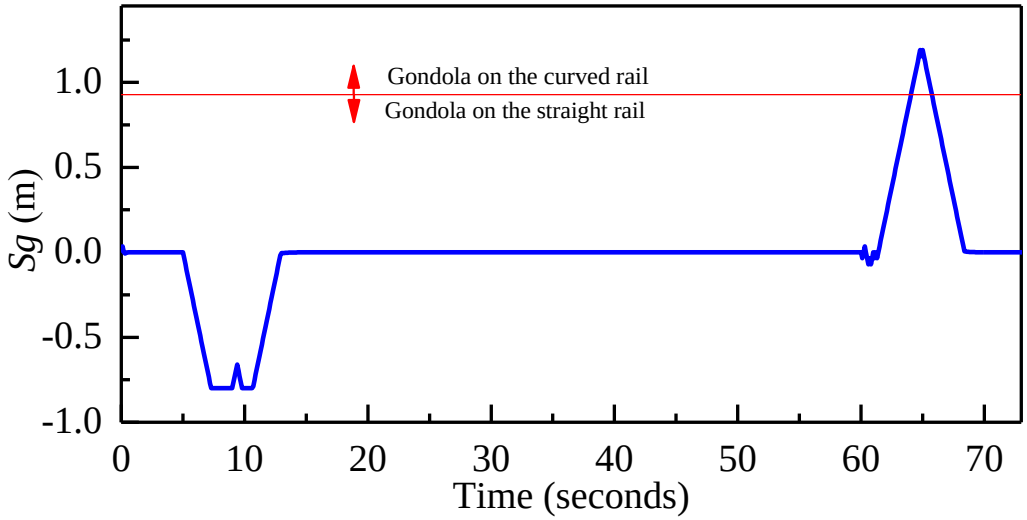


Figure 5.4 Gondola position Model 1 using backstepping control with no-wind

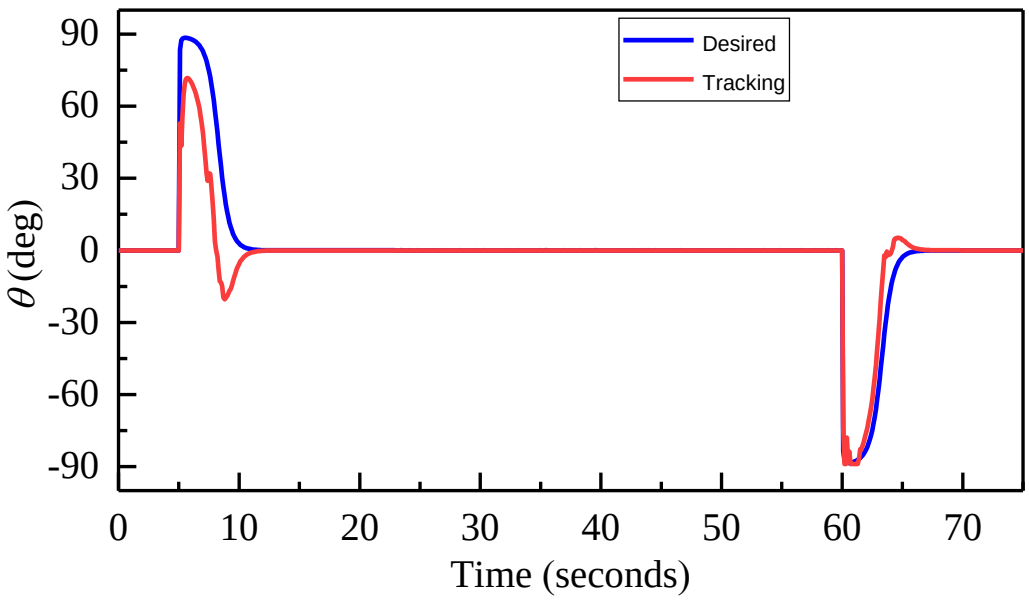


Figure 5.5 The reference pitch angle and simulated pitch Model 1 using backstepping control with no-wind

The tracking performance of the backstepping controller during the simulation in the presence of wind disturbances is shown in Figure 5.6. The results of the gondola's position control under wind disturbances, along with the pitch responses to the gondola's movements, in degrees, are shown in Figure 5.7 and Figure 5.8, respectively. The results show the robustness of the proposed control while subjected to aggressive manoeuvres. The airship remains stable with only small oscillations, even in the presence of unknown wind disturbances with a maximum steady state error of 5^{o}.

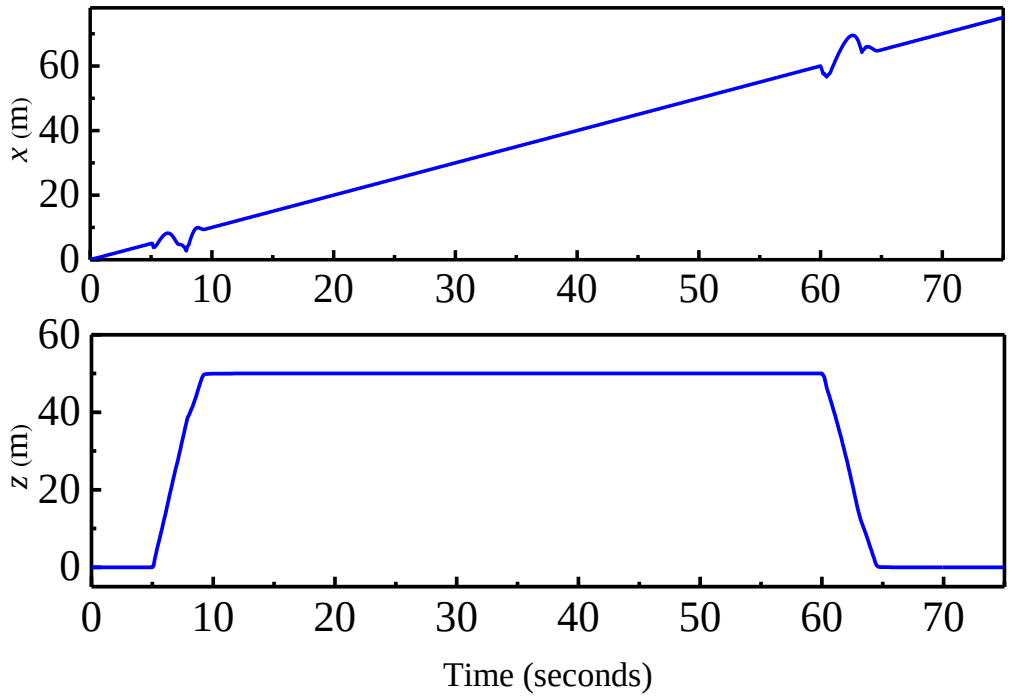


Figure 5.6 Airship position under wind disturbances Model 1 using backstepping control

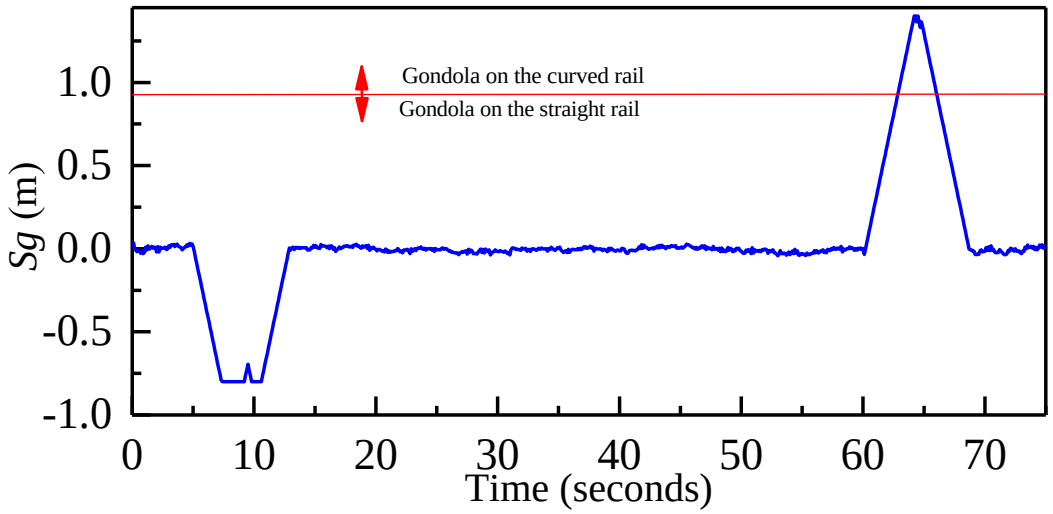


Figure 5.7 Gondola position under wind disturbances Model 1 using backstepping control

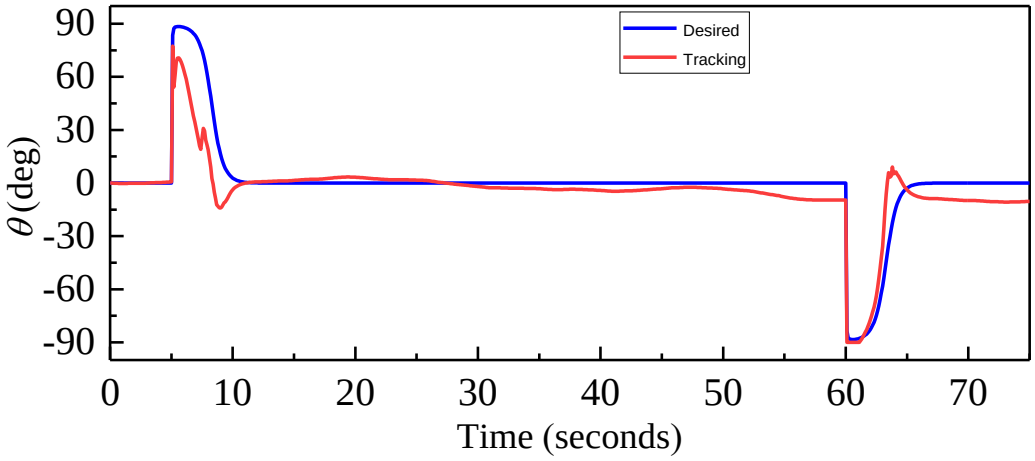


Figure 5.8 The reference pitch angle and simulated pitch under wind disturbances Model 1 using backstepping control

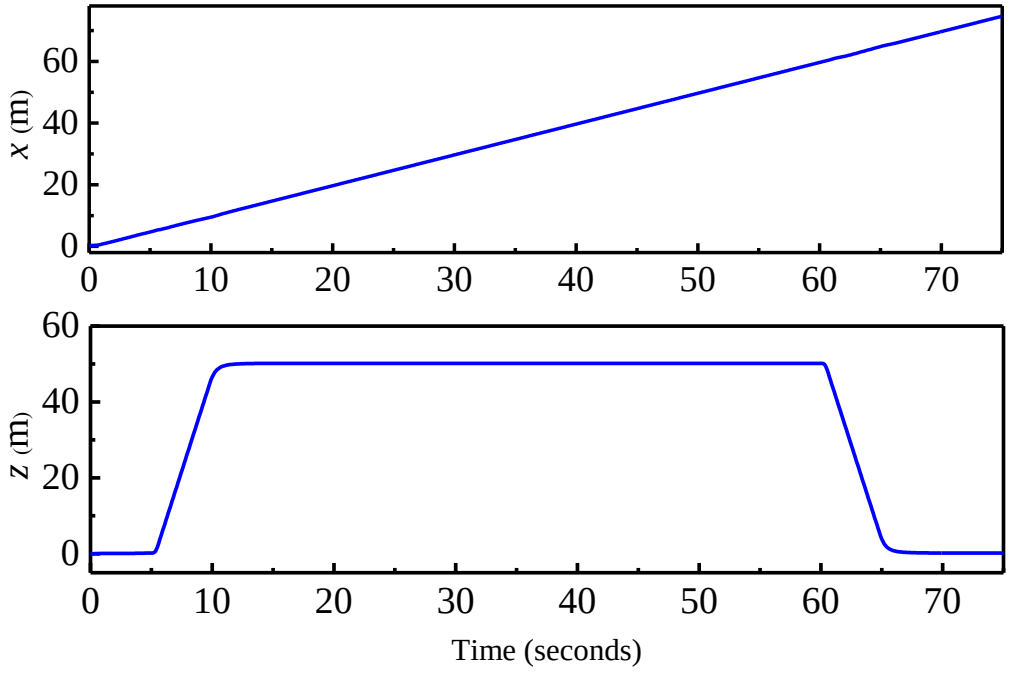


Figure 5.9 Airship position Model 1 with no-wind using backstepping control

The tracking performance of the INDI controller was then evaluated using the same reference, trajectory and initial position. The controller gains were selected heuristically, by trial and error, in order to obtain a desired closed loop response. The position gains and the velocity gains of the

translational control are K_{xp} = 3, K_{zp} = 2, K_{xv} = 5, K_{zv} = 6 and the longitudinal controller gains are K_{θ} = 8, $K_{\dot{\theta}}$ = 20. The acceleration $\dot{\mathbf{X}}_0$ was estimated by passing the velocity $\dot{x}$ through a second order low pass filter equation with a natural frequency of ω_n = 25 and damping of ζ_r = 0.7 rad/s. The natural frequency and the damping were selected by trial and error. The closed loop of the INDI control system is shown in Figure 4.2.

The simulation results in no wind conditions are presented in Figure 5.9 to Figure 5.11 and the simulation results under wind disturbances are presented in Figure 5.12 to Figure 5.14. Figure 5.9 shows the tracking performance of the INDI controller during the simulation. The tracking performance is satisfactory with an error between ± 0.2 m in x direction during the ascent and the descent.

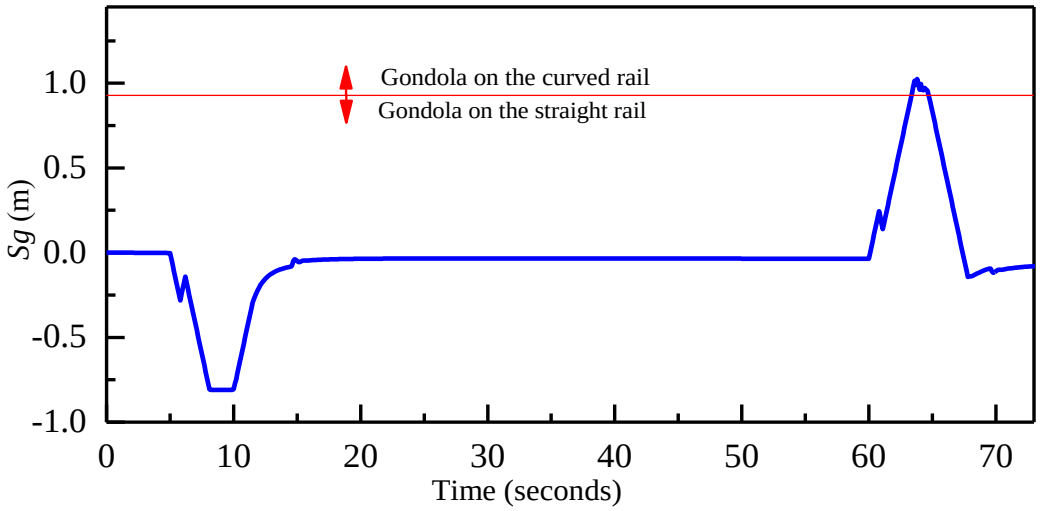


Figure 5.10 Gondola position Model 1 using INDI control with no-wind

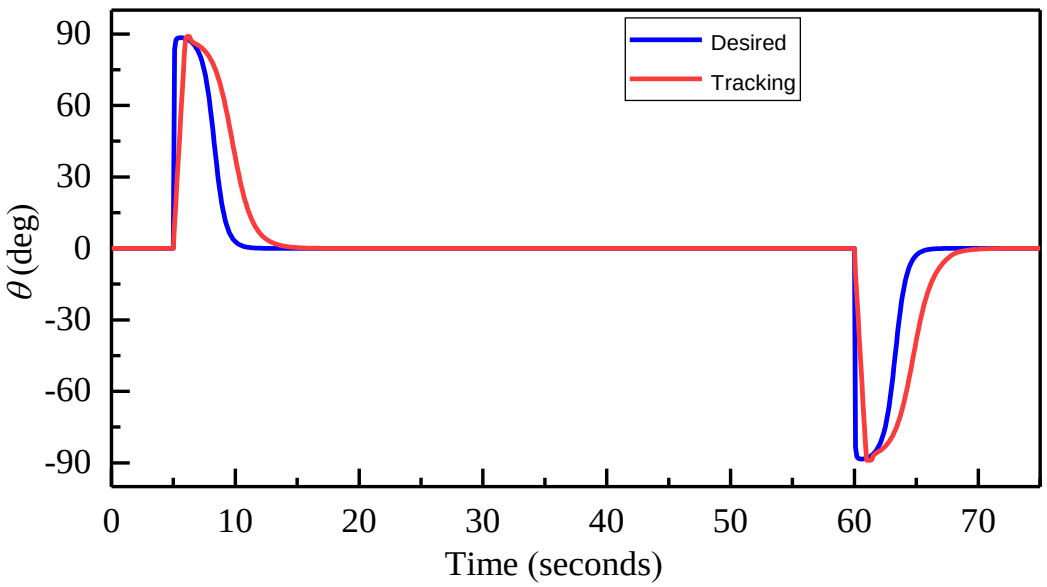


Figure 5.11 The reference pitch angle and simulated pitch Model 1 using INDI control with no-wind

The tracking performance of the INDI controller during the simulation in the presence of wind disturbances is shown in Figure 5.12. The reference trajectory was tracked successfully with an error of + 0.3 m during the ascent mode and -0.4 m in the descent mode in x direction. The reference pitch angle is tracked very well, and very small oscillations are observed during the rapid altitude change in the ascent and the descent modes. The steady-state error due to the wind disturbances is about 4^{o}, as can be seen in Figure 5.14.

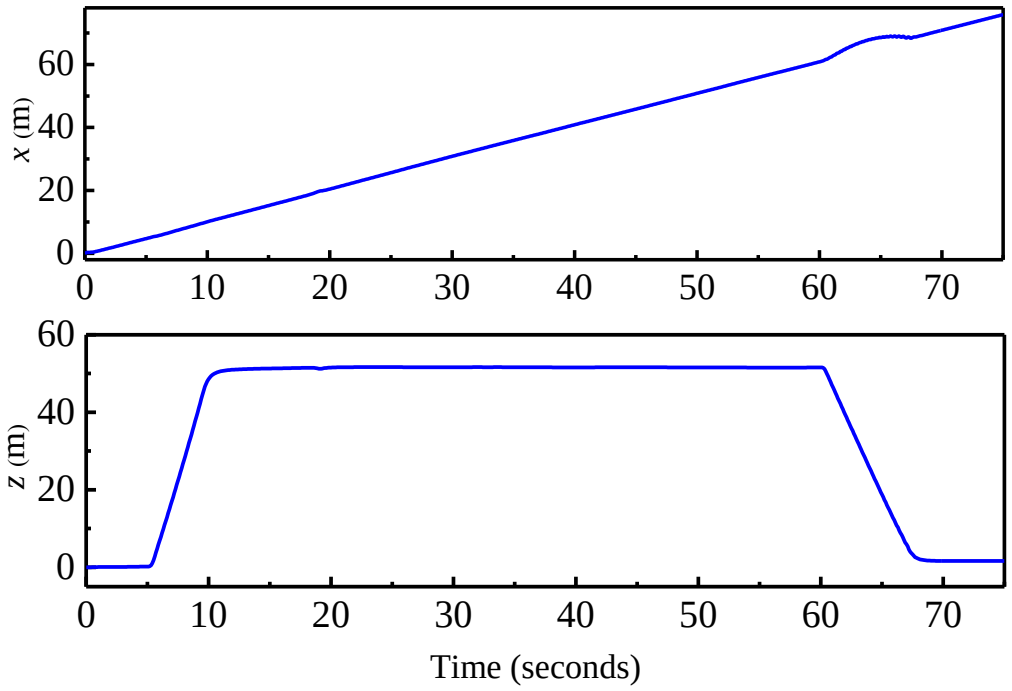


Figure 5.12 Airship position under wind disturbances Model 1 using INDI control

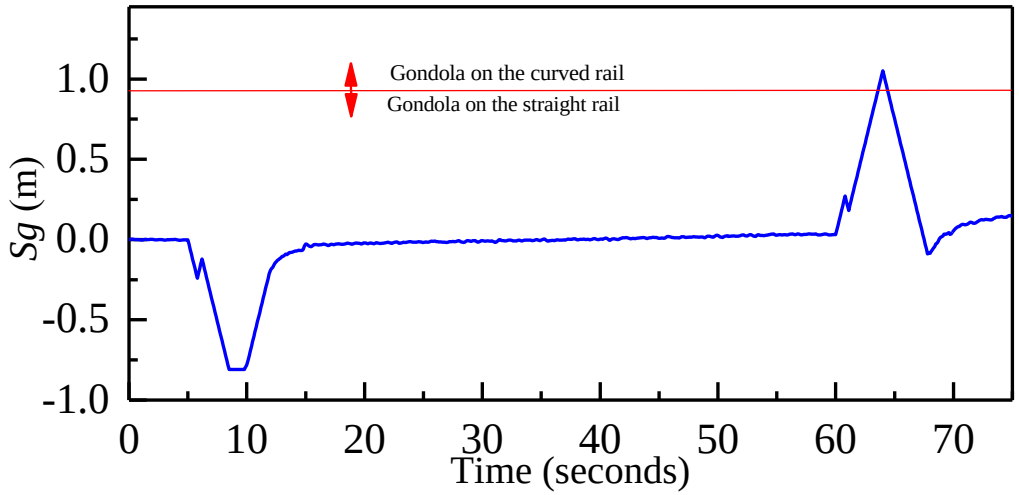


Figure 5.13 Gondola position under wind disturbances Model 1 using INDI control

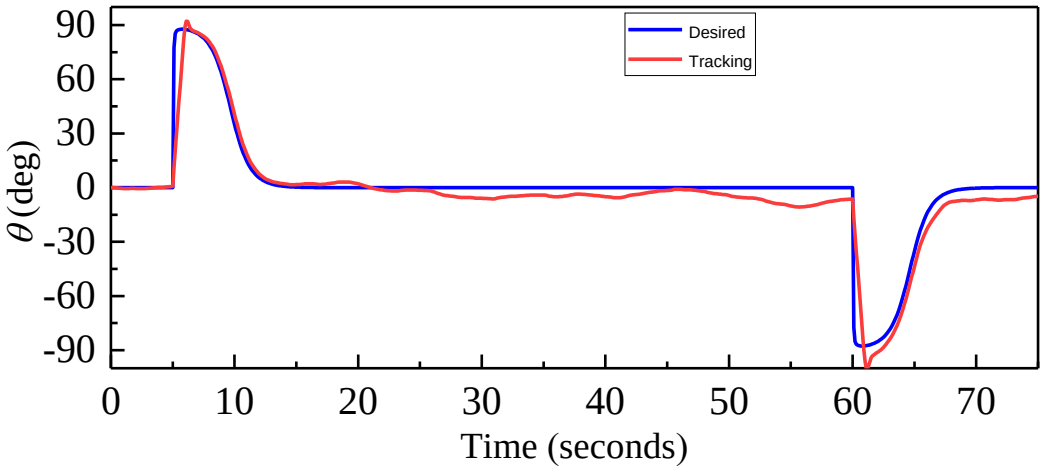


Figure 5.14 The reference pitch angle and simulated pitch under wind disturbances Model 1 using INDI control

5.2 The dynamic simulation of the airship Model 2

A similar simulation procedure as the one used for the airship Model 1 was adopted to illustrate the performance of both controllers on the airship Model 2. The closed loop of the backstepping architecture in Figure 5.2 was used. Controller gains were selected by trial and error, to ensure that the desired tracking performance could be achieved while satisfying the constraints caused by the mechanical limits of the airship (K_1 = 10 x I_{6x6}, K_2 = 23 x I_{6x6}).

The real control inputs are $\mathbf{u} = [s_g, F_x, F_z]^T$ and similar initial conditions and run time as those used with Model 1 are considered. The results of the backstepping controller are presented in Figure 5.15 to Figure 5.20. Figure 5.15 shows the tracking performance of the proposed controller, during the simulation, in x and z directions. Due to the rapid manoeuvre that causes the control to saturate in the x and z directions, there are noticeable oscillations during the ascent and descent in the x direction. The control output of the gondola's position is shown in Figure 5.16 and the results of the pitch's responses to the movement of the gondola, in degrees, are shown in Figure 5.17. The

pitch that was caused by the thrusters, due to their location below the CV, caused the gondola to rapidly move backwards to approximately s_g = - 0.20 m, at the start of the simulation, as shown in Figure 5.16. The rapid movement of the gondola is necessary in order to maintain the given altitude. There are also noticeable oscillations during the ascent and descent, as seen in Figure 5.17, which are caused by the rapid movement of the gondola and the aerodynamic drag moment induced by the changes in pitch. The results show that the gondola was able to move from s_g = - 0.47 m to s_g = 1.1 m and generate a pitch angle of 72^o and -89^o during the ascent and descent, respectively.

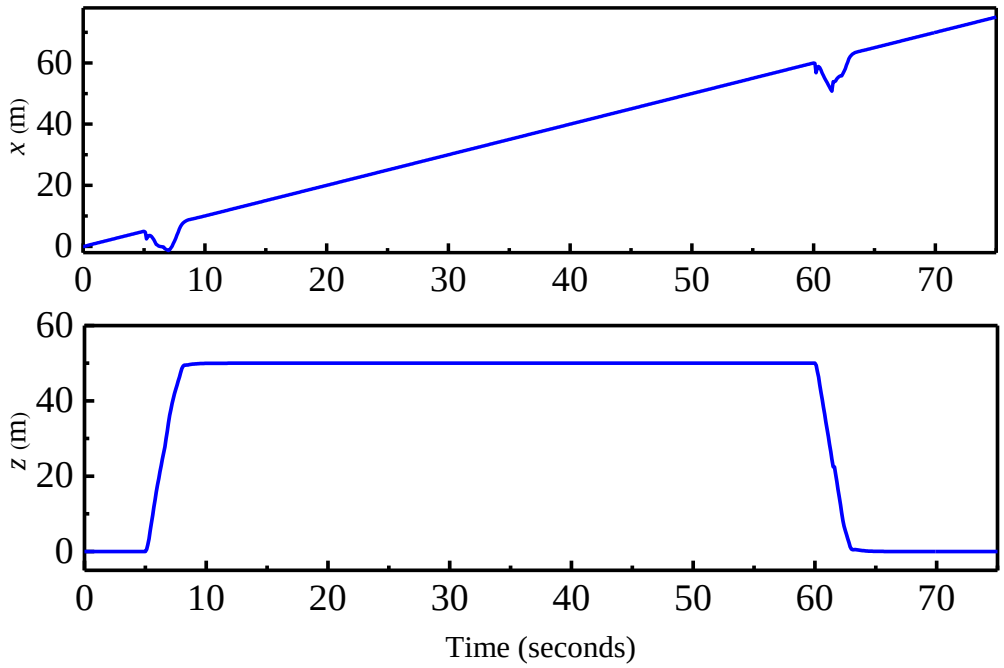


Figure 5.15 Airship position Model 2 using backstepping control with no-wind

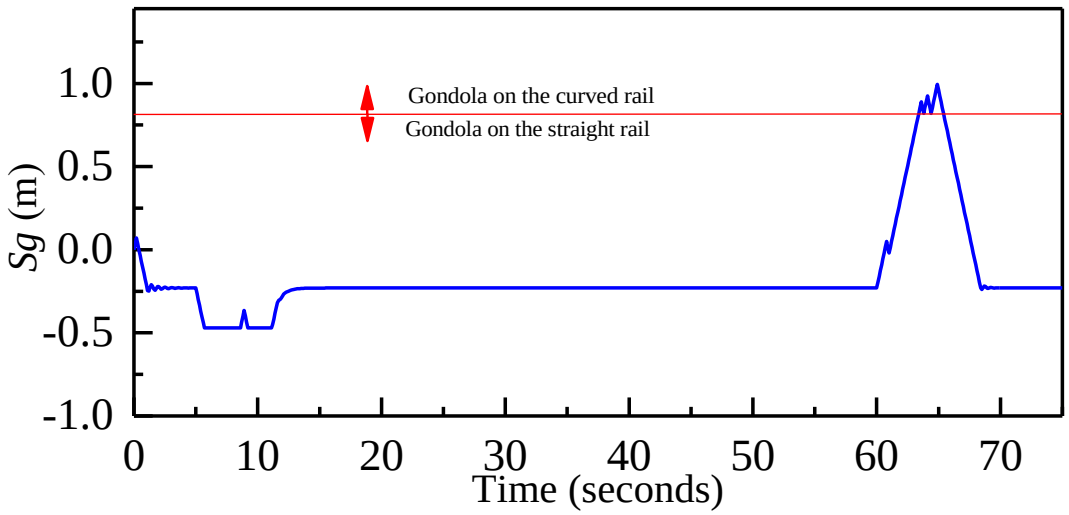


Figure 5.16 Gondola position Model 2 using backstepping control with no-wind

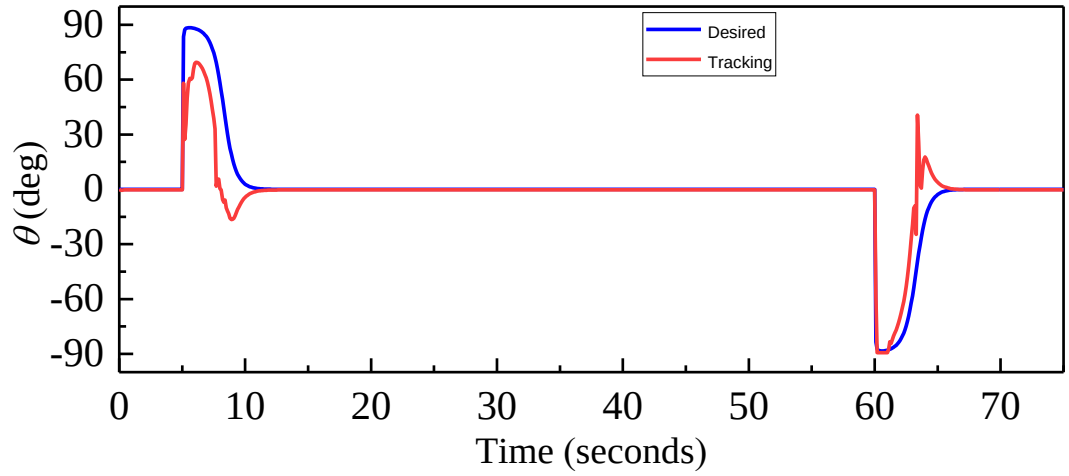


Figure 5.17 The reference pitch angle and simulated pitch Model 2 using backstepping control with no-wind

The tracking performance of the backstepping controller was evaluated in the presence of wind. The simulation results are presented in Figure 5.18 to Figure 5.20. The low amplitude oscillations caused by the wind disturbances can be observed in the gondola's position. The maximum amplitude error of the pitch, due to the wind disturbances, was successfully tracked at about 7°, as can be seen in Figure 5.20.

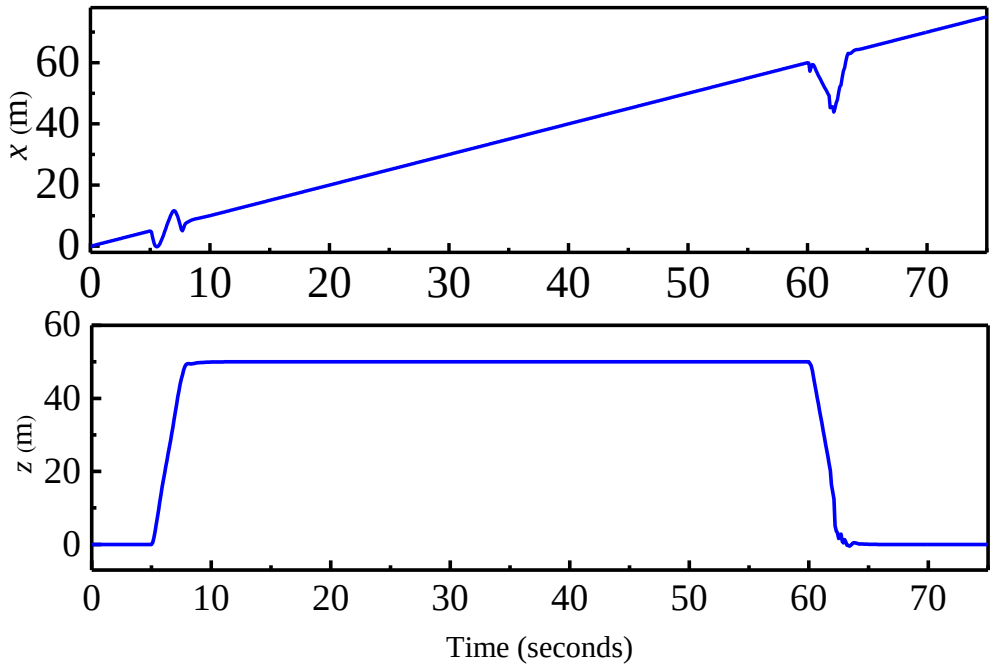


Figure 5.18 Airship position under wind disturbances Model 2 using backstepping control

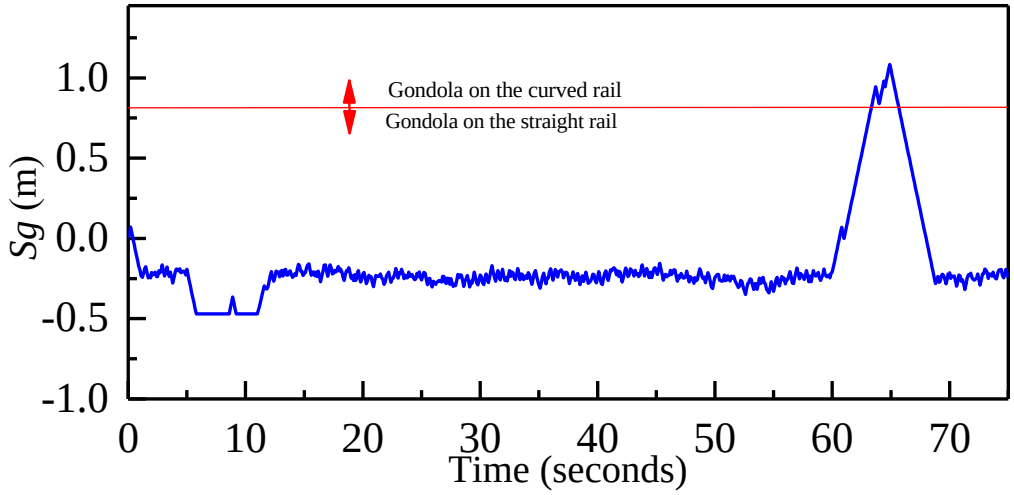


Figure 5.19 Gondola position under wind disturbances Model 2 using backstepping control

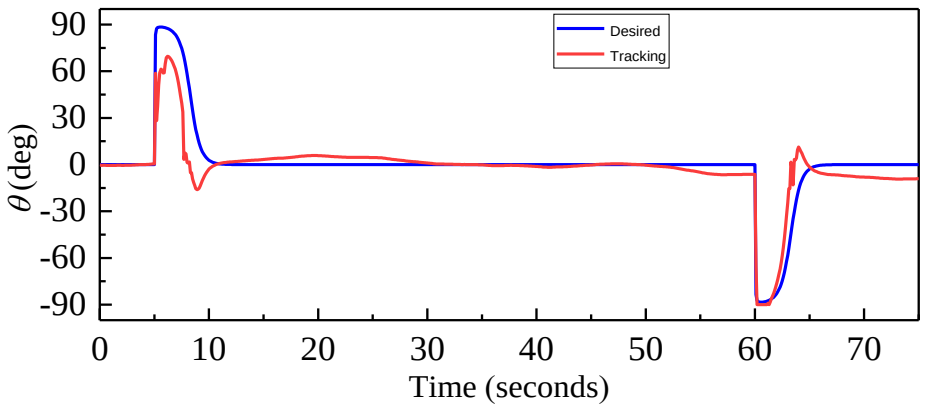


Figure 5.20 The reference pitch angle and simulated pitch under wind disturbances Model 2 using backstepping control

The simulation results for the airship Model 2 using the INDI controller are presented in Figure 5.21 to Figure 5.26. The position and the velocity gains of the translational control are K_{xp} = 0.3, K_{zp} = 0.5, K_{xv} = 1.2, K_{zv} = 6, and the longitudinal controller gains are K_θ = 2, $K_{\dot{\theta}}$ =10. The acceleration $\dot{\mathbf{X}}_0$ was estimated to be similar to that of Model 1. The closed loop of the INDI control system is shown in Figure 4.2. The position tracking performance is satisfactory with an error between ± 0.3 m in x direction during the altitude changes. The control output of the longitudinal loop is shown in Figure 5.22. As with the backstepping controller, the gondola moves backward to s_g = 0.17 m at the beginning of the simulation as a result of the pitch moment generated by the location of the thrust. The pitch responses to the gondola's movements, in degrees, are shown in Figure 5.23. It can be noted that the reference pitch angle is tracked very well, and no oscillations are observed. Figure 5.23 shows a steady-state error of 3^o. This steady-state error is always produced when using only proportional control to design the reference acceleration in (4.24). The error signal gets smaller as the system converges to the set-point, therefore proportional control times the error signal will be small enough that it can force the system to zero.

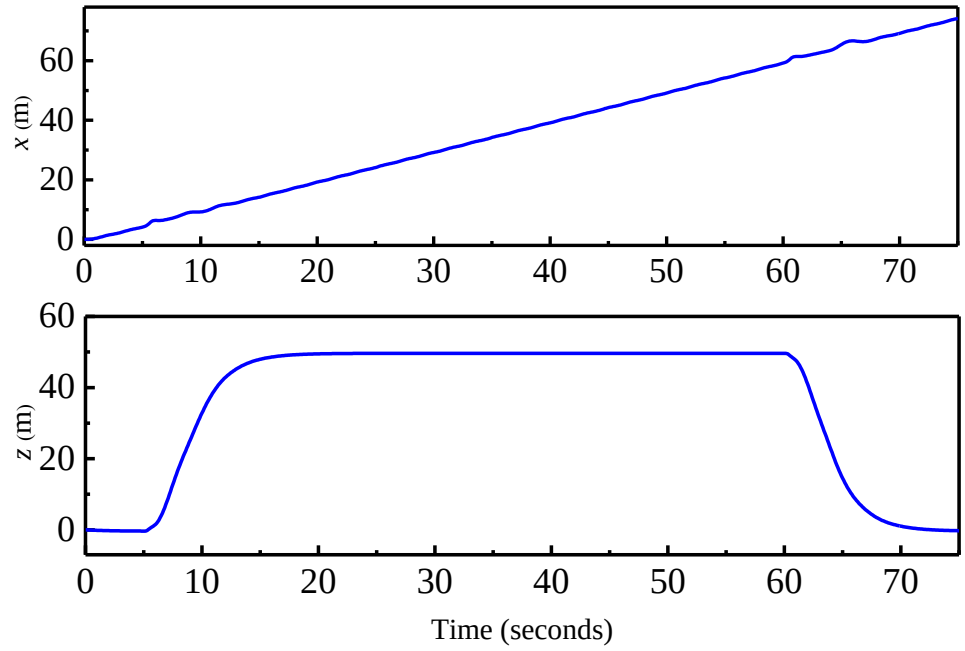


Figure 5.21 Airship position Model 2 using INDI control with no-wind

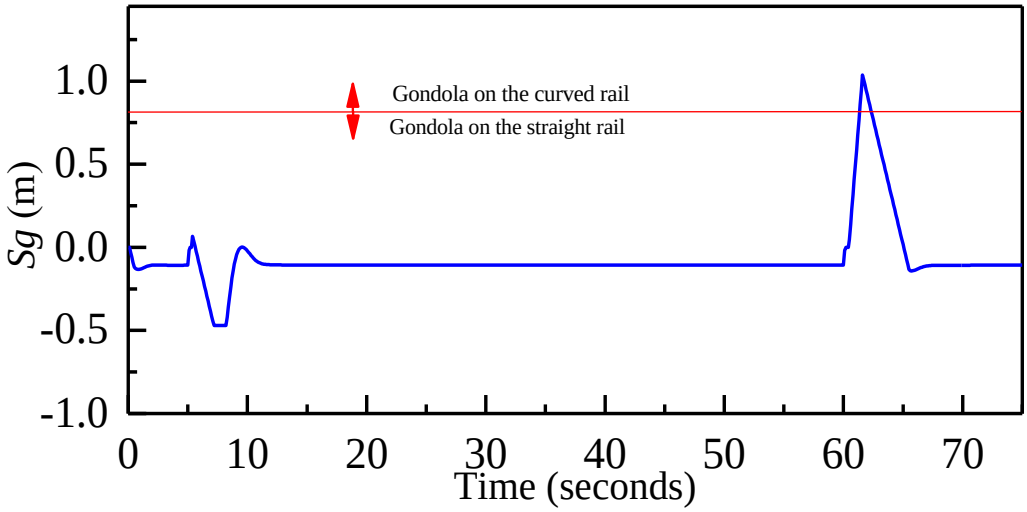


Figure 5.22 Gondola position Model 2 using INDI control with no-wind

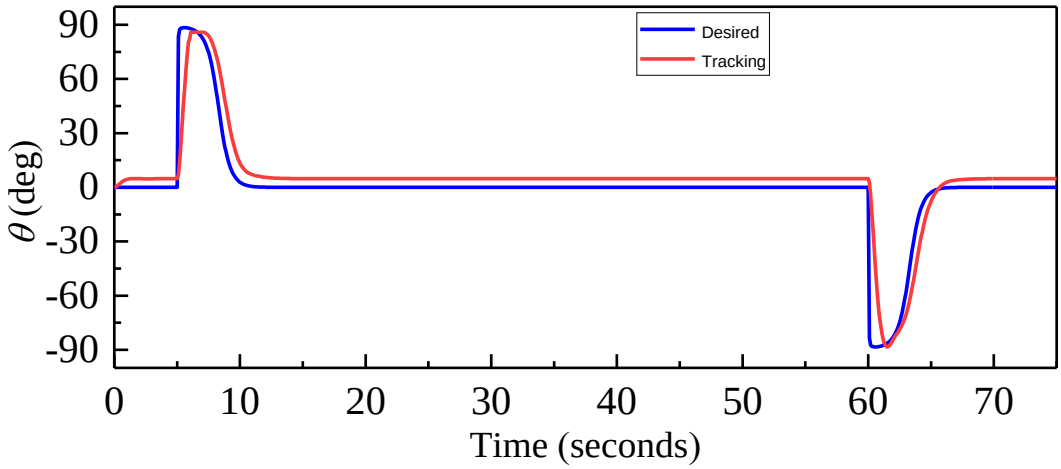


Figure 5.23 The reference pitch angle and simulated pitch Model 2 using INDI control with no-wind

The results of the INDI controller in the presence of wind disturbances are presented Figure 5.24 to Figure 5.26. The tracking errors in altitude and position are observed during the longitudinal maneuvers with an error of ± 5 m during the ascent mode and 16 m in the descent mode in x direction, as illustrated in Figure 5.24. The oscillations during the ascent and descent in the x direction are due to the rapid manoeuvre that causes the control to saturate in the x and z directions. The control inputs are shown in Figure 5.25, with saturations observed during the ascent of the airship. The pitch was successfully tracked in the presence of wind disturbances with a steady-state error of about 6^{o}, as can be seen in Figure 5.26.

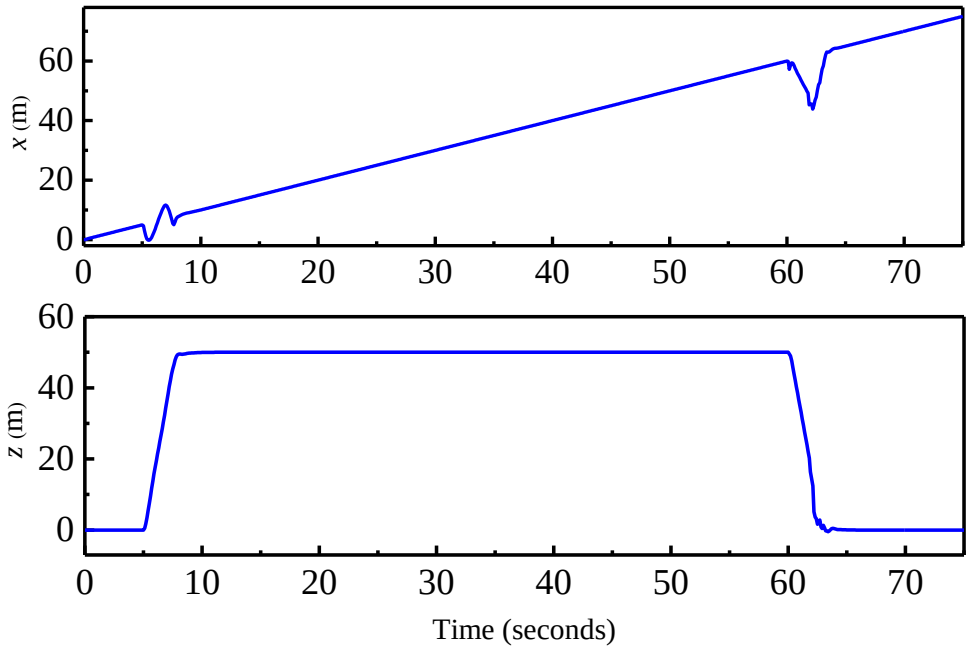


Figure 5.24 Airship position under wind disturbances Model 2 using INDI control

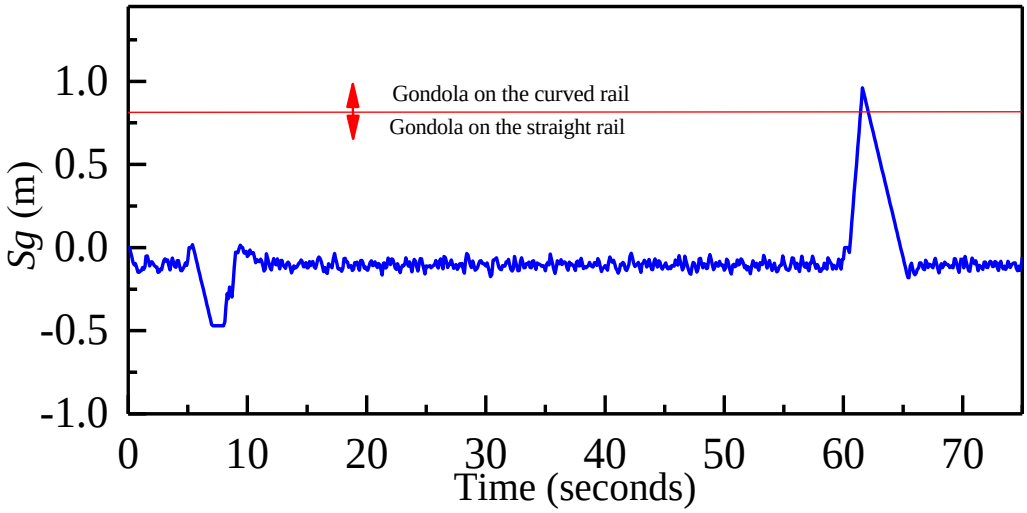


Figure 5.25 Gondola position under wind disturbances Model 2 using INDI control

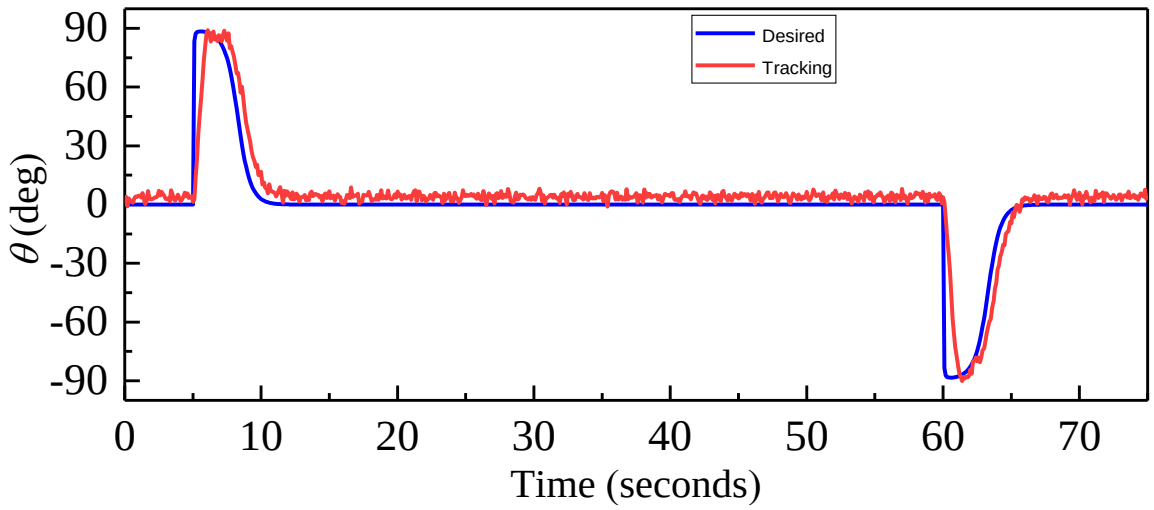


Figure 5.26 The reference pitch angle and simulated pitch under wind disturbances Model 2 using INDI control

The simulation results showed that both controllers were able to accomplish the entire mission successfully with noticeable different performances. The airship Model 1 was able to generate pitch angles of 73^o and -88^o during the ascent and descent modes respectively. The maximum pitch angle during the scent was generated as a result of moving the gondola backward to its saturation limit of s_g = -0.81 m. In case of airship Model 2, the airship was able to generate pitch angles of 72^o and -89^o during the ascent and descent modes when the gondola moves from s_g = -0.47 m to s_g = 1.1 m respectively.

The backstepping results in all cases show smaller deviations from the reference trajectory. The results of INDI control show that the control was also able to track the reference trajectory with more steady state errors. These errors can be caused by two reasons: due to the possible bias on the actuator measurements and using the P-control to design the virtual control input.

Chapter 6: **Energy Consumption Analysis**

The energy model developed in Section 3.3 is used to evaluate and compare the energy required by the airship during ascent, cruise, and descent flights, using two different actuation methods. As previously mentioned, the actuation methods utilized are centre of gravity modifications using gondola position control and thrust vectoring. In the current platform, the moving gondola is developed for pitch tracking using the gondola's position as a control input. The vectored thrust is developed for altitude control and uses the thrust angle as a control input. Each actuator is evaluated individually for pitch and altitude tracking and the energy consumption of each actuator is investigated. In the current study, the efficiency of each actuator at completing the flight missions will be compared. The purpose of this comparison is to demonstrate the effectiveness of the moving gondola used in the current platform when compared to that of the fixed gondola used in other airships with different pitch control systems (e.g., thrust vectoring). Furthermore, the effectiveness of a composite control strategy that integrates both actuators to achieve similar flight missions is also included in the comparison. The performances of the composite configurations of the two controllers for trajectory tracking are shown in Chapter 5.

The purpose of the comparison made in this same chapter is to study the energy consumption needed to achieve a successful flight mission using the two controllers described in Chapter 4.

The composite control is developed in Sections 4.1 and 4.2. The gondola actuator control differs from the composite control in that the vectored angle γ is considered zero and thrusters are considered parallel to the horizontal axis of the body frame of the airship, during the entire flight mission. As a result, the thrust components in the body frame in Equation (4.18) become $F_x = 2T$ and $F_z = 0$, and the real control input becomes, $\mathbf{u} = [s_g, T]^T$.

On the other hand, in the individual vectored thrust actuator for pitch control, the position of the gondola s_g is considered zero and the thrust components in Equation (4.18) become $T_R = T_L = T$ *and* $F_x = (T_R + T_L)\ cos\ \gamma$, and $F_z = (T_R + T_L)\ sin\ \gamma$. However, the real control input becomes, $\mathbf{u} = [T, \gamma]^T$.

The total control effort or energy required for the entire trajectory will be determined by the numerical simulation. The total energy consumption of the airship during all flight modes is the integral of the total required power defined in Equation (3.50).

The objective function of each actuation method is to minimize the total energy consumption.

The total energy using only the gondola actuator E_G is,

$$E_G = (P_m + P_c) \times t_{0-f} + P_G \times t_G \qquad (6.1)$$

where the power of each system is described in Section (3.3). t_{0-f} is the total time needed to reach the goal and t_G is the running time of the gondola's motors.

The total energy required when using only the thrust vectored actuator $E_{V.T}$ is,

$$E_{V.T} = (P_m + P_c) \times t_{0-f} + P_s \times t_s \qquad (6.2)$$

where t_s is the running time of the servo motors.

The total energy E_{tot} when using the composite control is,

$$E_{tot} = (P_m + P_c) \times t_{0-f} + P_G \times t_G + P_s \times t_s \qquad (6.3)$$

$$E_i = \sum_i P_i \cdot \Delta t_i$$

Extensive simulations are conducted using similar case scenarios to evaluate the energy consumption of the current platform for each actuation method. In all cases studied, the motions of the airship are divided into ascent, cruise, and descent modes, and the energy consumed in each flight mode is analysed. Both backstepping and INDI controllers are used to evaluate the tracking performance of the different actuators.

The airship begins from a flight level of z = 0 m and ascends to 50 m, in the ascent mode, at different reference pitch angles. The airship then cruises for 50 m in the x-direction, at an altitude of 50 m, before landing (reaching an altitude of zero) in descent mode at different pitch angles.

All simulations use different airship speeds at each pitch angle, in order to measure the energy consumption required in different possible flight scenarios as well as to evaluate the capability of the different actuation methods of the current platform to achieve their mission's objectives.

The actuator constraints applied in the simulation are based on the physical properties of the actual airship.

During the ascent, the actuator constraints are;

- The servo angle: $\zeta \leq 7^o$
- The rearmost position of the gondola: s = - 25 cm
- The resultant of thrust: - 0.26 N $\leq$ T $\leq$ 2.6 N

During the descent the actuator constraints are;

- The servo angle: $\zeta \leq -100^o$
- The front position of the gondola: s $\leq$ 150 cm
- The resultant of thrust: - 0.26 N $\leq$ T $\leq$ 2.6 N

Weather conditions can strongly influence the energy consumption of the airship; therefore, the evaluation is conducted in various wind conditions. The wind disturbances are produced by using

a Dryden model, as discussed in Section (3.2.9), and originate from the opposite direction than that of the flight (directions x and z) to simulate a worst-flight scenario, as seen in Figure 6.1 and Figure 6.2.

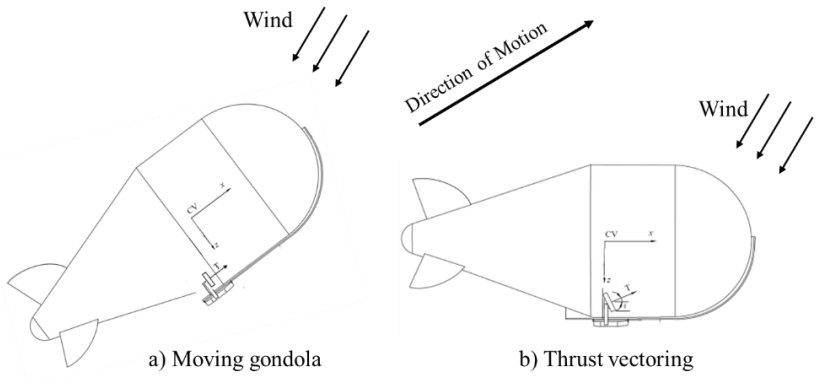


Figure 6.1 Direction of motion during the ascent mode

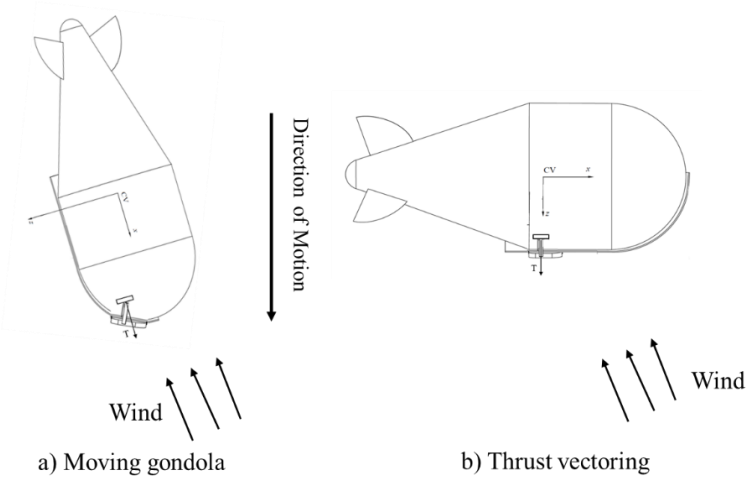


Figure 6.2 Direction of motion during the descent mode

The total energy of the airship is calculated based on the equations discussed in Section (3.3). The quantity of constant energy being consumed by the actual on-board electronics is calculated using current and voltage measurements obtained with the fully charged battery (12.6 V). The measurements of the on-board components are listed in Table 6.1. The constant energy in (3.55)

is calculated as the sum of all the power consumed by the on-board components multiplied by the travel time, from the start of the mission until the airship reaches its destination. Similarly, the propulsion system consumes power for the duration of the flight time. The power required by the propulsion system is calculated from (3.61) and the overall efficiency η_p is considered to be 75%, based on thruster static bench testing.

The energy consumed by the moving gondola is mostly consumed during the movement of the gondola to control the pitch angle and significantly reduced when the pitch angle is tracked, unless an adjustment of the position due to external disturbances is required.

Similarly, in vectored thrust configuration, the servo motors only consume power when they move to the required angle.

Table 6.1 Power consumption of the on-board electronic components

	Component Description	Quantity	Current (A)	Voltage (V)	Power (W)
1	Raspberry Pi 3 Model B+	1	2.5	5	12.5
2	Counter Click	1	$5x10^{-6}$	5	0.000025
3	Magnetic Encoder	1	0.0002	5	0.001
4	MPU6050	1	0.0039	5	0.0195
5	RF 7020 Radio Frequency Module	1	0.4	3.3	1.32
6	Arduino Compatible 5V Relay Module	1	0.0714	5	0.357
4	AR610 Aircraft Receiver	1	2	5	10

The simulation results of the energy required by the airship to reach the reference altitude for the different actuation methods are obtained by using backstepping and INDI controllers. The simulations of both controllers are subjected to the same initial conditions and gains, as described

in Section (5.2). The results during different flight missions and with varying airship speeds are shown in Figure 6.3 throughFigure 6.13. In addition, the calculation of the energy needed for each mission is tabulated in Appendix E. The results covered an airship speed ranging from 1 m/s to 5 m/s in disturbance-free conditions and in windy conditions.

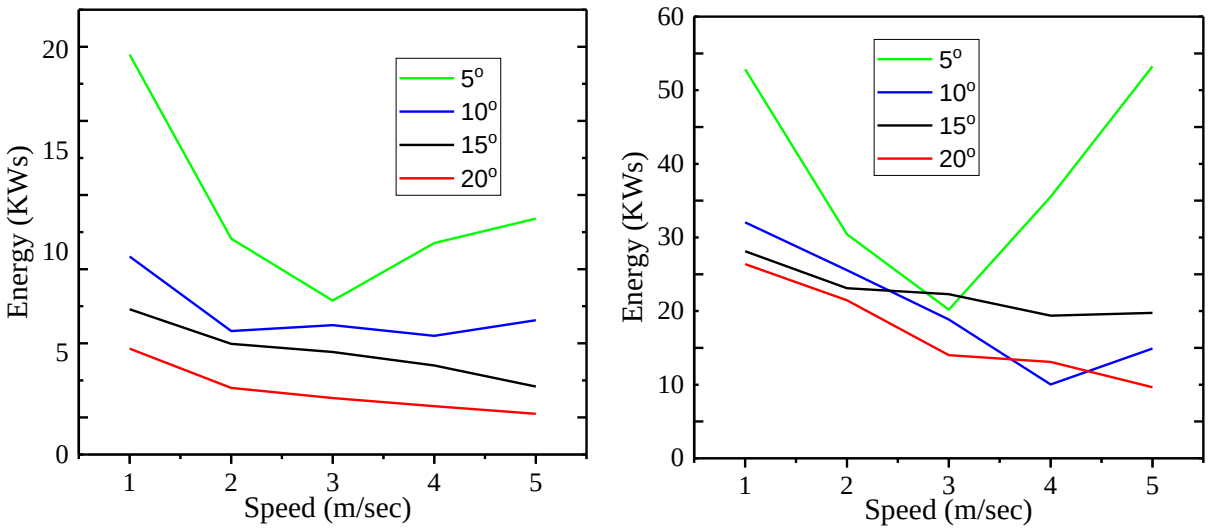
6.1 Results of Backstepping control

The results of the different acuation methods are illustrated for different flight modes.

a- Gondola Configuration

The results obtained during the ascent mode at different flight speeds, with no external disturbances using the gondola configuration, are shown in Figure 6.3 (a). The figure shows that during the ascent, as speed increases, energy consumption decreases; this is due to the shorter travel time. It is not surprising that travel time has a significant impact on energy consumption, as seen in Equation (6.1) where the constant energy is about 60% of the total energy and is consumed throughout the flight. In addition, the results show that as the pitch angle increases, the airship reaches the reference altitude faster and consumes less energy. The energy consumed at a velocity of 3 m/s and a pitch angle of 5^o is less than the energy consumed at the other velocities, as shown in Figure 6.3, since the pitch angle is generated by the thrust and the energy consumed by the gondola's movement is nearly zero in calm conditions. The energy required for the cruise mode, in calm conditions, as well as the travel time in each scenario, are shown in Table E.1 in Appendix E. The results in the table show that the airship reached an altitude of -50 m in 96 seconds with a velocity of 3 m/s, when the pitch angle was 10^o and the conditions were calm. The total energy consumed was 6.39 KWs. As expected, energy consumption in the presence of wind disturbances increases for all flight scenarios, as shown in Figure 6.3 (b). As seen in Table E.2 in Appendix E, in another scenario with windy conditions, the airship reached an altitude of -50 m in 100 seconds

with a velocity of 3 m/s when the pitch angle was 10°. The total energy consumed was 18.86 KWs. Because of the wind disturbances, the gondola's running time t_G increases, as does the energy consumption in Equation (6.1), as a result of the movement of the gondola to stabilize the airship and track the reference pitch. The gondola's movement causes a variation in the angle of attack, which results in a change in the aerodynamic moment. Consequently, the drag force in Equation (6.1) increases, leading to higher energy consumption. The results in Table E.1 and Table E.2 also show the amount of energy required for the cruise mode in different wind conditions. The results show that in this mode, when using only the gondola configuration, the airship is less sensitive to wind, consuming 0.857 KWs in calm conditions and 0.914 KWs in windy conditions, at a velocity of 2 m/s.

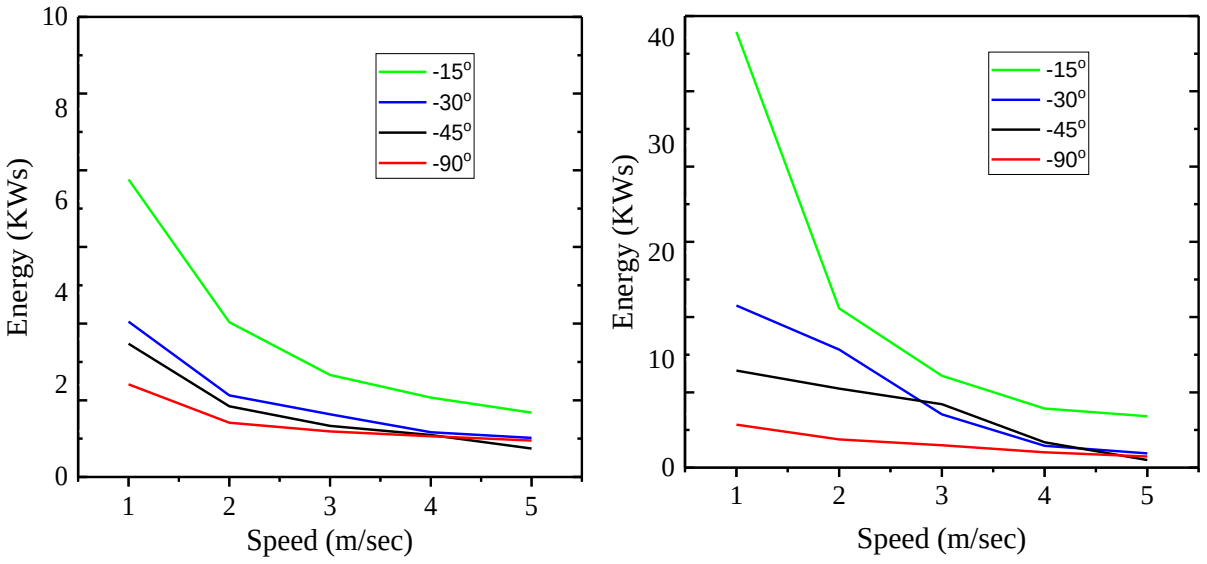


a) calm condition b) windy condition

Figure 6.3 Energy consumption at different pitch angles and different flight speeds during the ascent using Gondola configuration

The simulation results also include the data for the descent in different conditions and possible flight scenarios, with respect to the physical properties of the prototype. After the airship cruises for 50 m, it descends from an altitude of -50 m to zero m, at various pitch angles, including -90 °.

As a result, the gondola moves to the front end of the airship. Figure 6.4 shows the energy consumption during the descent, at different pitch angles, flight speeds and wind conditions, using the gondola configuration. The energy consumption results in Table E.3 and Table E.4 in Appendix E show that the airship consumed less energy when it descended by -90 ^{o} compared to when it descended by other pitch angles, as reorienting the airship to the direction of motion reduces the angle of attack, therefore reducing the drag force significantly and consequently also the energy consumption. In addition, the shorter travel time also reduces the total energy consumption.



a) calm condition b) windy condition

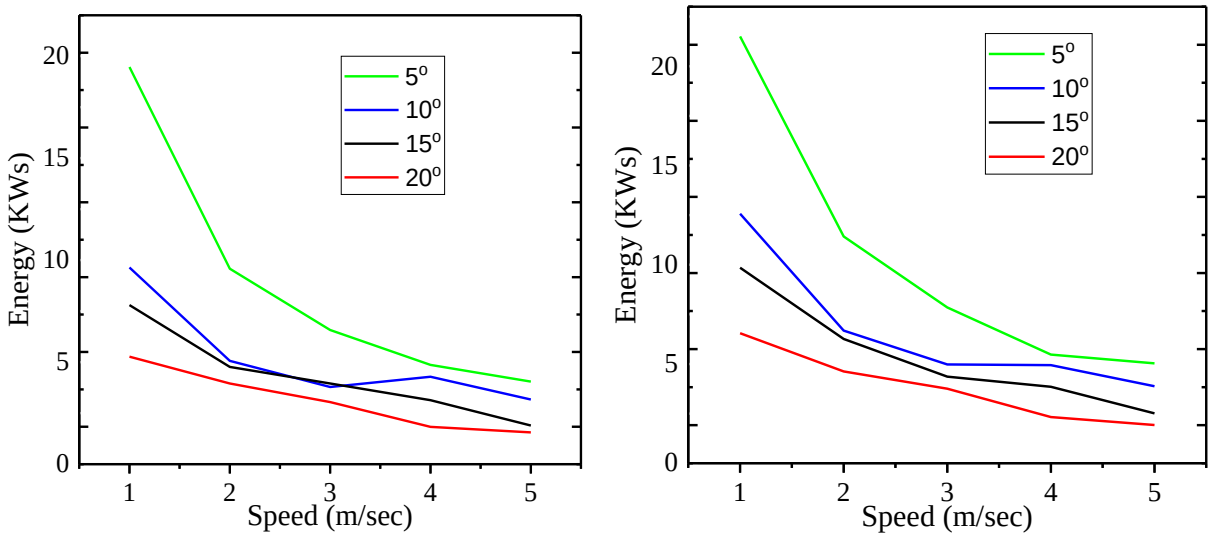
Figure 6.4 Energy consumption at different pitch angles and different flight speeds during the descent using Gondola configuration

b- Vectored Thrust Configuration

Energy consumption during the different flight missions is also evaluated using the vectored thrust actuator. The total energy is calculated using Equation (6.2). The simulation results are obtained using similar case scenarios to those in part a, in order to compare each actuation method's ability to complete the mission with minimal energy consumption. During the ascent, the thrust generates a positive pitch moment due to its location below the CV of the airship. A rapid response in the

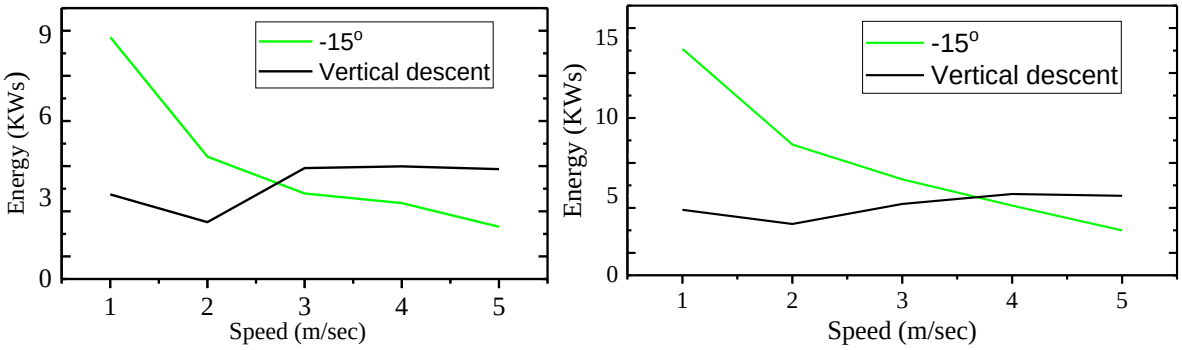
pitch angle is usually observed when using the vectored thrust for pitch tracking. However, in this case, the pitch control requires a continuous action, which consumes a significant amount of energy. The energy consumption results obtained during the ascent and cruise modes, in different environmental conditions, are shown in Figure 6.5. In addition, energy consumption and travel times are illustrated in Table E.5 and Table E.6 in Appendix E. Using the vectored thrust at a velocity of 2 m/s and a pitch of 10^o, in calm conditions, the airship required 5.5 KWs of energy during the ascent, compared to the 6 KWs needed with the gondola actuator. At low speeds, the vectored thrust is more efficient than the gondola actuator. For example, at 1 m/s with a pitch of 5^o, the vectored thrust consumed about 22 KWs, whereas the gondola required 42 KWs. During the descent mode, the pitch angle cannot be tracked with the vectored thrust actuator on this platform due to its location below the CM, but the airship can still reach the required altitude. This configuration, on the other hand, can efficiently perform a vertical landing with a pitch angle near zero degrees. The energy consumption results obtained during the descent mode, in different environmental conditions, are shown in Figure 6.6. Also, the energy required and the travel times are illustrated in Table E.7 and Table E.8 in Appendix E. In a vertical descent, the vectored thrust angle tilts to -90^o, and the corresponding vehicle angle of attack also reaches approximately -90^o. As a result of the large angle of attack, the drag force increases, resulting in higher energy consumption. The airship descended from an altitude of -50 m at nearly zero pitch angle, to an altitude of zero in 25 seconds. With a velocity of 2 m/s, it consumed 2.1 KWs of energy in no-wind conditions, while it consumed 3.224 KWs in approximately 28 seconds in windy conditions. Figure 6.5 shows that when the reference pitch is -15^o the airship consumes more energy and requires 194 seconds to reach an altitude of zero, compared to 50 seconds for a vertical descent at 1 m/s. Pitch tracking using the vectored thrust configuration is not possible when the pitch angle

is less than zero degrees because the thrust is located below the center of mass of the airship and cannot generate a negative pitch moment.



a) calm condition b) windy condition

Figure 6.5 Energy consumption at different pitch angles and different flight speeds during the ascent using vectored thrust configuration

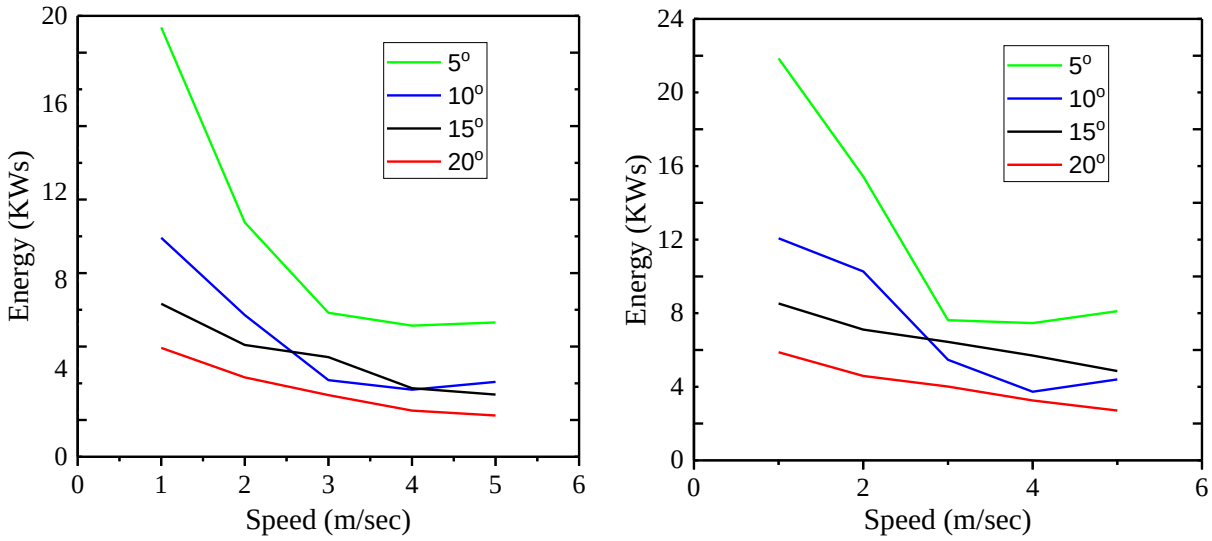


a) calm condition b) windy condition

Figure 6.6 Energy consumption at different pitch angles and different flight speeds during the descent using vectored thrust configuration

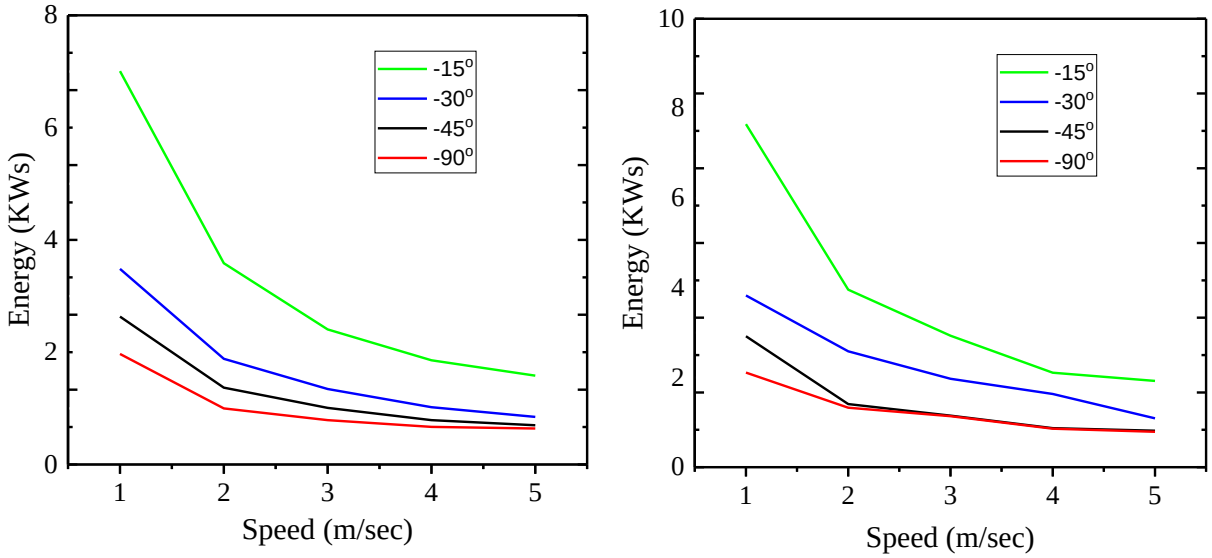
c- Composite Configuration (Gondola+Vectoring Thrust)

The simulation results of the composite configuration are obtained using case scenarios like those used in a and b. The energy consumption results for the different flight modes, under various environmental conditions, are illustrated in Figure 6.7 and Figure 6.8. It can be noticed that there are very small differences in the results obtained with the composite configuration and the gondola configuration, in all flight modes, but especially in calm conditions. For example, in the ascent mode, with a 10^{o} pitch angle and a velocity of 1 m/s, energy consumption was 9.9 KWs with the composite configuration and 9.7 KWs with the gondola configuration. However, in most of the cases when conditions are windy, energy consumption is lower by 30 % when using the composite configuration. In this case, the pitch can simultaneously be controlled by the vectored thrust and the moving mass. The vectored thrust causes the pitch to change instantaneously based on the thrust value which in turn helps to stabilize the airship faster and limit the gondola's movements to track the pitch angle. As a result, reducing the gondola's operating time in Equation (6.3) lowers the energy consumption of the system. The total energy consumption and travel time for the different flight modes, under various environmental conditions, are shown in Tables E.9 through E.12 in the Appendix E. In all flight scenarios, the composite configuration allows for shorter travel times than the other configurations. During the descent, the composite configuration consumes less energy than the other methods. In the presence of wind disturbances, it takes 13 seconds for the airship to reach an altitude of zero with a -90^{o} pitch angle, and as shown in Figure 6.8, it consumes 0.790 KWs at 5 m/s compared to 3.9KWs with the gondola configuration.



a) calm condition b) windy condition

Figure 6.7 Energy consumption at different pitch angles and different flight speeds during the ascent using composite control configuration



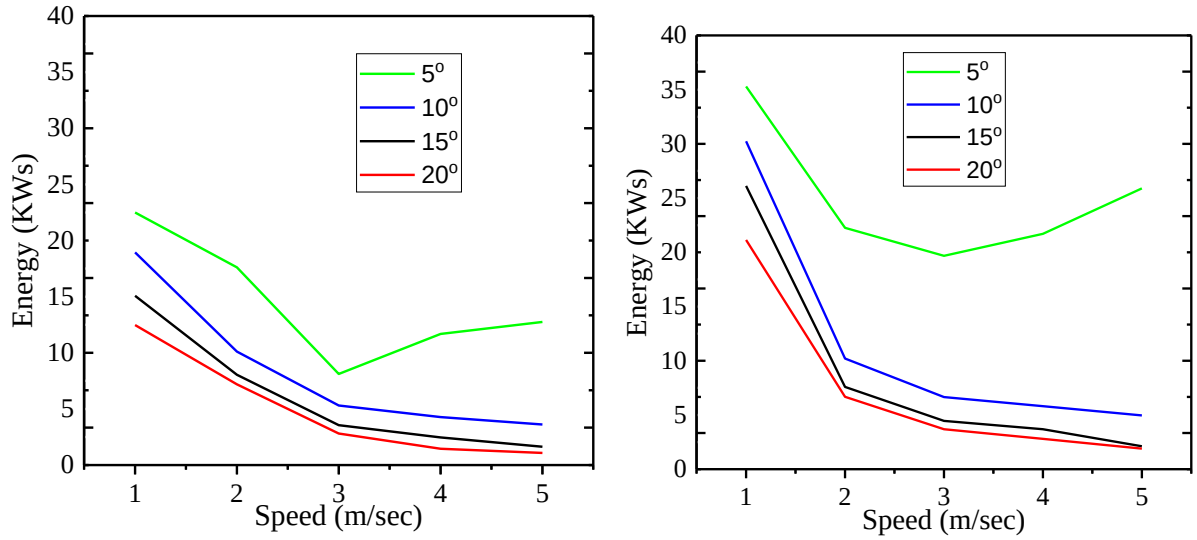
a) calm condition b) windy condition

Figure 6.8 Energy consumption at different pitch angles and different flight speeds during the descent using composite control configuration

6.2 Results of INDI Control

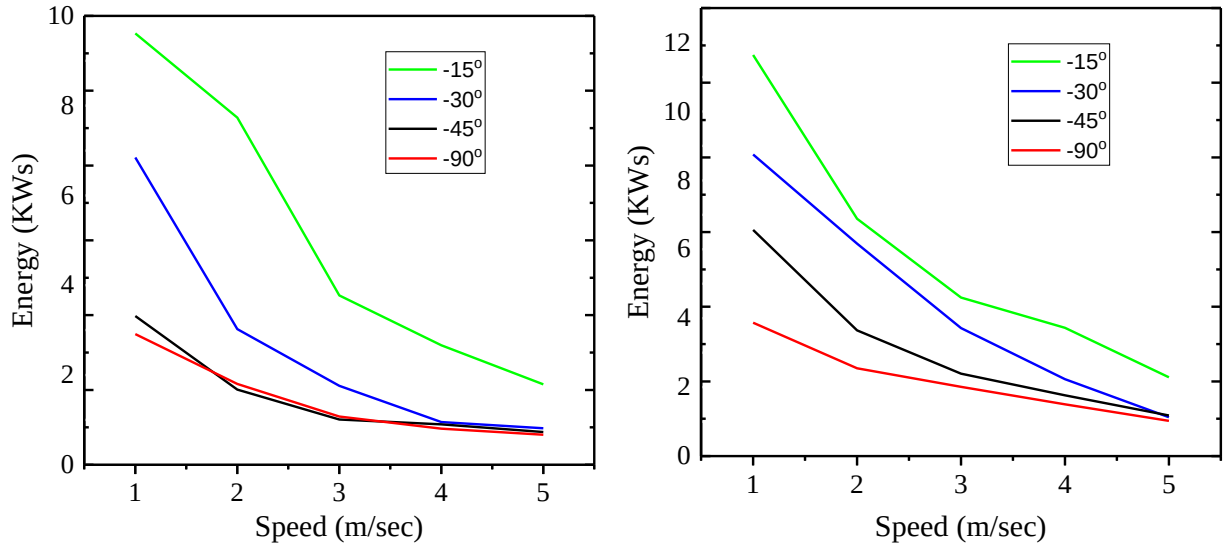
As in the previous section, INDI control is used to evaluate the capability of the different actuation methods available in the current platform to accomplish the flight mission. It is also used to analyse the energy consumption during the flight and to compare the control effort with that of backstepping. The energy consumption results of the different actuation methods and different flight modes of the INDI control are illustrated in Figure 6.9 through Figure 6.13. The INDI control completed the flight mission with minimal energy consumption when using the gondola and composite configurations during the descent mode (consuming 2.77KWs when using the composite method and 3.72KWs when using the gondola method), at a pitch angle of -15^{o} and a velocity of 4 m/s. However, it continues to saturate when using only the vectored thrust actuator for the descent Mode. In this case, the INDI is essentially tracking the acceleration, and when the acceleration error is detected, the INDI controller will increment the control inputs in order to rapidly begin tracking the reference again. As, in this case, the pitch angle cannot be tracked during the descent, the system becomes unstable and oscillates, increasing energy consumption. Therefore, the energy results of this actuator during the descent mode are not considered because it is not an efficient method. The vectored thrust method shows higher energy requirements during the ascent in the presence of wind disturbances than other actuator methods. Figure 6.11 (b) shows that the energy required at low speed is the highest, with about 47 KWs at a pitch of 5^{o}. However, as expected, the consumption is lower in calm conditions, with approximately 42.9 KWs and 20 KWs in the case of backstepping. The total energy consumption and the travel times for the different flight modes, under various environmental conditions and with the use of INDI control, are shown in Tables E.13 through E.22 in the appendix E.

The results of both controllers show that all actuation methods were capable of successfully completing the ascent and cruise missions. However, the energy consumed when using the INDI control during the ascent mode in the presence of wind and with the various actuation methods is lower than with backstepping, as illustrated in Figure 7.3. The results show that in the presence of wind, the most efficient control during the ascent is provided by the INDI control when using the gondola actuator. The total energy consumed in this situation, as shown in Table E.14, is 2.1 KWs. In the case of the cruise mode, the most efficient control is the backstepping control when vectored thrust is used, consuming 0.44 KWs. The lowest consumption in the descent mode is also the backstepping control, but this time with the composite configuration, consuming about 0.8 KWs.



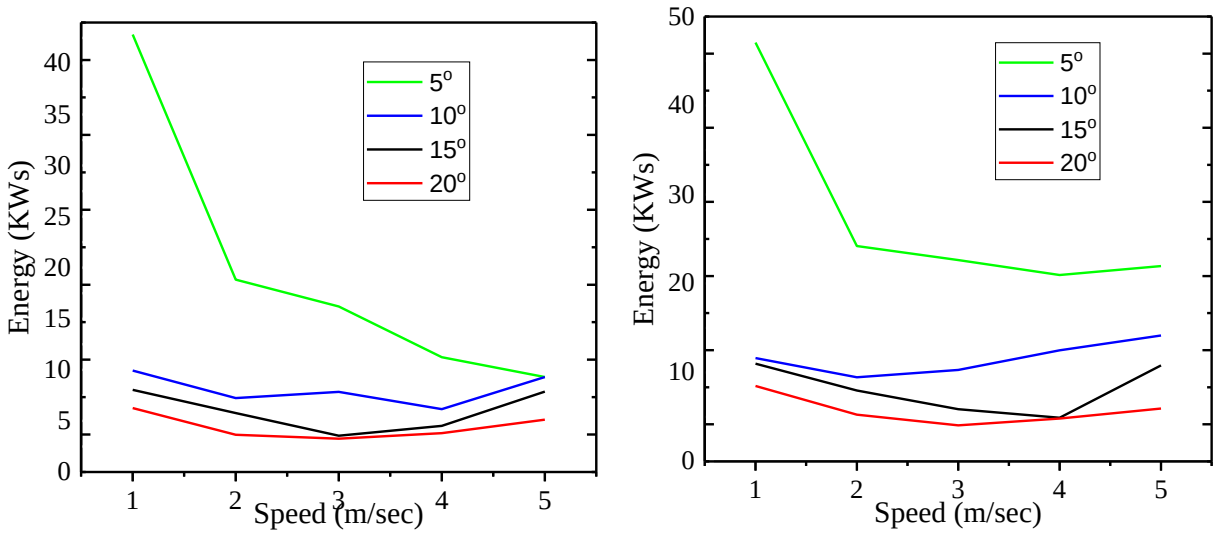
a) calm condition b) windy condition

Figure 6.9 Energy consumption at different pitch angles and different flight speeds during the ascent using Gondola configuration



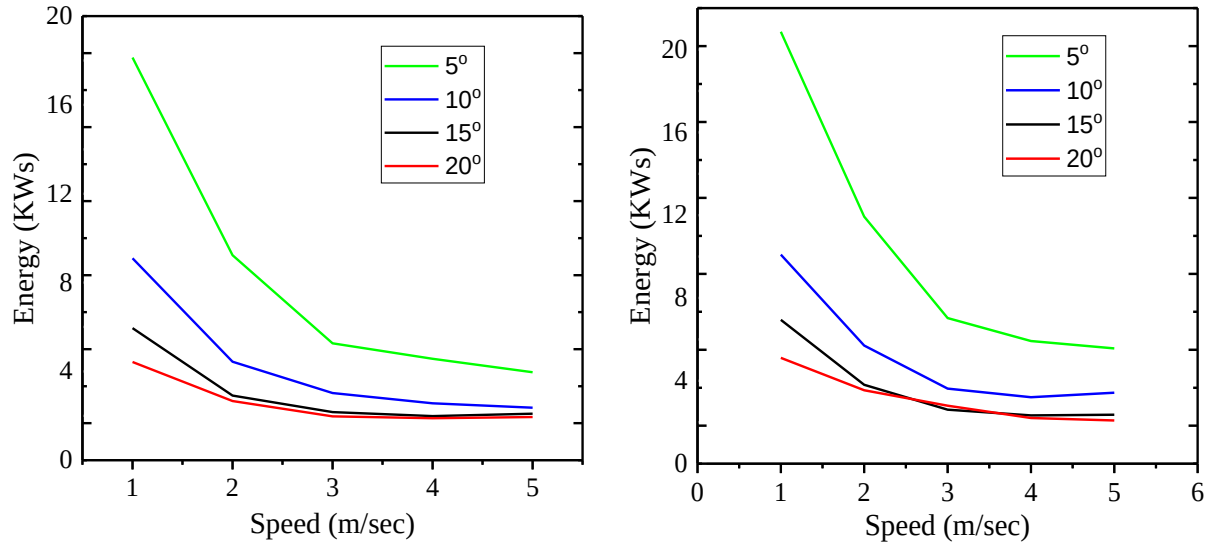
a) calm condition b) windy condition

Figure 6.10 Energy consumption at different pitch angles and different flight speeds during the descent using Gondola configuration



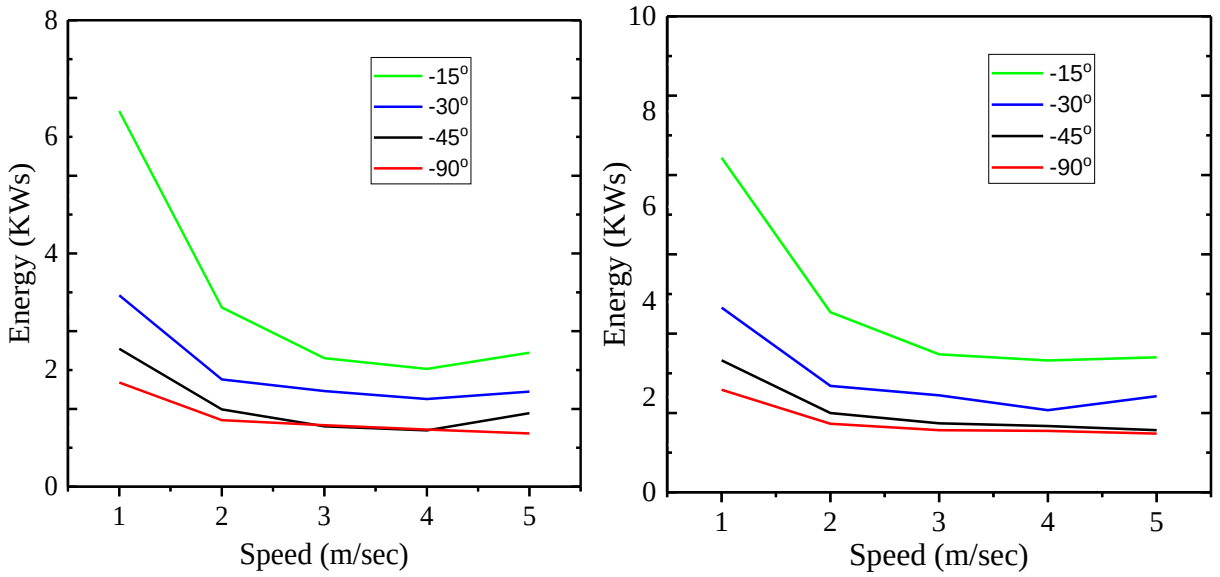
a) calm condition b) windy condition

Figure 6.11 Energy consumption at different pitch angles and different flight speeds during the ascent using vectored thrust configuration



a) calm condition b) windy condition

Figure 6.12 Energy consumption at different pitch angles and different flight speeds during the ascent using composite configuration



a) calm condition b) windy condition

Figure 6.13 Energy consumption at different pitch angles and different flight speeds during the descent using composite configuration

Chapter 7: **Evaluation of Optimal Solutions**

The results obtained from the controllers with the different actuation methods in the previous chapter are used in this chapter to develop the optimal flight trajectory based on the capability of the platform to complete the required mission with minimal energy consumption. The simulations were carried out in different possible scenarios enabling the optimization to be performed quickly due to the simplified functions used. In this work, different control methods and different actuation methods are used to achieve the maximum system performance in terms of energy efficiency. Therefore, in this section, an optimization method is used to find the optimal design configuration that minimizes the cost function in Equation (7.5) for a minimum energy flight trajectory. Usually, this search process is conducted under certain design constraints to ensure that the optimization process converges on a minimum value.

Several methods for optimizing UAV energy consumption have been suggested [129–131]. However, optimization algorithms typically take more time to execute as they are, most of the time, mathematically difficult to solve. Furthermore, some real-world processes cannot be adequately modeled using linear optimization techniques. Therefore, in this study a heuristic technique is used to obtain the optimal solution. A heuristic algorithm is a specific type of algorithm that conducts a search of the solution space. It may make use of certain aspects of the problem to avoid having to search through every possible solution, or it may update its search method based on the input it receives from previously found solutions. The algorithm begins with a set of control input values and uses various methods to refine the values and decrease the value of the associated cost function, until the best solution is found.

The main advantage of the heuristic approach is that it offers a quick solution that is easy to understand and implement. Heuristic algorithms are practical, serving as fast and feasible short-term solutions to planning and scheduling problems. Moreover, its success is ensured by the extensive simulations that cover different possible scenarios.

The formulation of trajectory optimization problems for various airship flight modes will be considered. The airship dynamic model described in Equation (3.1) can be expressed as:

The main advantage of the heuristic approach is that it offers a quick solution, which is easy to understand and implement. Heuristic algorithms are practical, serving as fast and feasible short-term solutions to planning and scheduling problems. Moreover, its success is ensured by the extensive simulations that cover different possible scenarios.

The formulation of trajectory optimization problems for various airship flight modes will be considered. The airship dynamic model described in equation (3.1) can be expressed as:

$$\dot{\mathrm{X}}(t) = f(\mathrm{x}(t), \mathrm{u}(t), t)$$

where X(t) is the state vectors, $t \subset [t_0, t_f]$, t_0 is the start time t_f is the final time, and u(t) is the control vector, $t \subset [t_0, t_f]$.

The states of the system have dynamic constraints due to Newton's equations. Therefore, constant bounds were enforced on the states and control inputs over the time interval $[t_0, t_f]$.

In this study, the optimal trajectories are considered to be those that require minimal energy. The sum of the energy required by the different systems in this platform are mentioned in Equations (6.1- 6.3). However, in this prototype, the propulsion system consumes the most energy during flight because it is the only system that generates thrust to move the airship and overcome drag.

The energy required by the propulsion system during the flight trajectory is the integral of the required power equation (3.61) based on the control effort,

$$E_p = \int_{t_0}^{t_f} \frac{V.T}{\eta_p} dt$$

The cost function is the sum of energy consumption of each flight mode and can be described as:

$$J = C_t\, t_f + C_e\, \sum_{i=1}^{n} E_i \qquad (7.4)$$

$$\text{where} \quad C_t + C_e = 1,$$

where C_t and C_e are time and energy weighting coefficients, respectively, that are chosen based on the operator priority of each objective and can range from 0 to 1. For pure minimum energy, $C_t = 0$ and $C_e = 1$. n is the total number of the system in the current platform.

The problem is summarized as:

$$\min_{\mathrm{u}(t), t_f} \quad J \qquad (7.5)$$

$$\text{Subject to,} \quad \mathrm{X}_L \leq \mathrm{X}(t) \leq \mathrm{X}_U, \;\; t > t_0$$

$$\mathrm{u}_L \leq \mathrm{u}(t) \leq \mathrm{u}_U, \;\; t > t_0$$

where the subscripts L and U are the lower and upper limits, respectively. The optimization problems for various types of airship flights for the different actuation methods are considered to choose a control history u(t) that minimizes the cost function J, which subjects to the equations of motion over time interval [t_0, t_f].

The algorithm implementation is shown in the flowchart in Figure 7.1. The algorithm takes the initial criteria for each flight mode based on the reference trajectory. The first subroutine is

repeated for each controller to find the optimal action for a given state, based on the results of the different flight scenarios. The solution of the optimal trajectory is selected based on the minimization of the cost function equation (7.5). For each flight mode, the optimal solution involves selecting: 1) the pitch angle, 2) the airship speed, 3) the actuation method, 4) the control approach. The optimal solution is chosen based on the operator priority in Equation (7.5). In this case, solving for pure minimum energy is used (C_t =0, C_e =1). The algorithm outputs the best result after the converging criteria is met at the end of each flight mode. The simulation stops at the end of the landing mode (z_f = 0 m). Figure 7.2 shows the implementation of the heuristic technique in the closed loop system. As explained above, the output of the optimal strategy block is the optimal action (control approach, actuation method, reference trajectory). The controllers are then used to control the gondola and the thrust to track the desired trajectory. The optimal strategy block starts by first selecting the minimum-energy solutions found in the prior numerical results, based on the environmental conditions. Then, the minimum energy trajectory is generated for each flight mode based on their initial condition. For each mode, the optimal actuation method and the controller will then be selected in order to execute the flight mission.

This approach is not well-suited for an online implementation due to the computational demands and the risk of not obtaining feasible solutions quickly enough to use them. However, the solutions were found by first generating optimal solutions offline or by adding an optimal strategy block to each controller approach separately in the closed loop. The optimal strategy block is used in the outer loop of the closed loop system for each controller, to select the optimal actuation method and track the desired trajectory, as described above. This method greatly simplifies the optimization and allows the system to rapidly compute the optimal trajectory for online implementation.

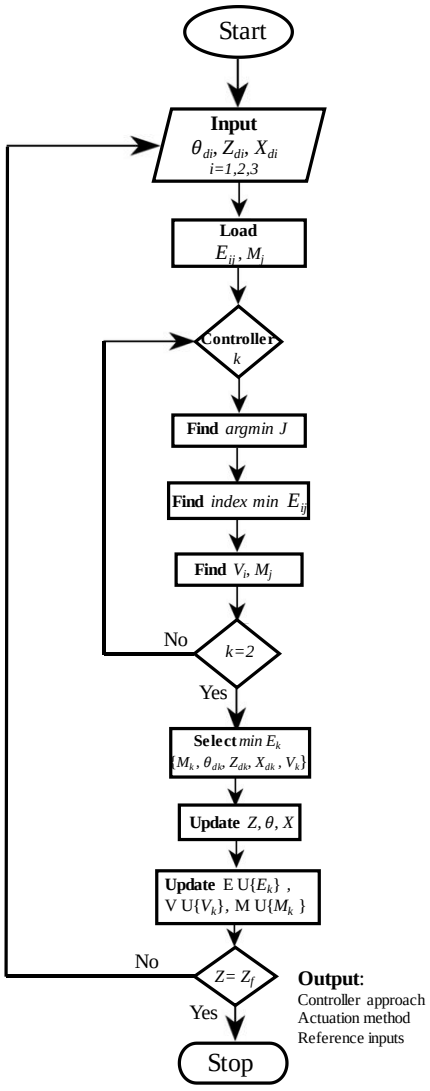


Figure 7.1 flowchart of the optimization methodology

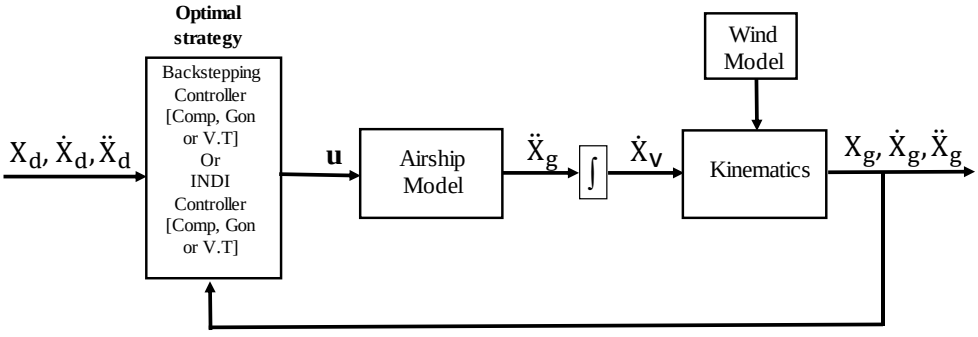


Figure 7.2 Control System Architecture
(comp. - composite, Gon. - gondola, V.T - vectored thrust)

Figure 7.3 illustrates the flight scenario where a minimum energy flight is required in the presence of wind disturbances. The flight started the ascent mode with initial conditions of (x_0, z_0, θ_0, v_0), which were updated during the cruise mode to (x_1, z_1, θ_1, v_1) and finally to (x_2, z_2, θ_2, v_2) for the descent mode. The figure shows that the optimal solution for the ascent is the INDI control using only the gondola actuation, with a speed of 5 m/s and a pitch angle of 20^o. The airship travels in the ascent mode for 31 seconds until the altitude of -50 m is reached. Once the airship reaches the reference altitude, the system chooses, based on the optimal solution, the backstepping control using only the vectored thrust actuator with a pitch angle of nearly zero and a speed of 5 m/s. The airship flies 50 m in x-direction, taking approximately 11 seconds. At the end of the cruise mode, the airship switches to the backstepping control system using the composite actuators and begins a descent flight with a -90^o pitch angle and a speed of 5 m/s. The airship completes the flight mission in 55 seconds and consumes a total of 3295 Ws of energy.

An example of an optimized flight in the presence of wind disturbances and based on a chosen reference trajectory is shown in Figure 7.4. In this scenario, the ascent mode inputs are a pitch angle of 10^o and a velocity of 4 m/s, while the cruise mode input is 2 m/s and the descent mode inputs are a -45^o pitch angle and a velocity of 4 m/s. The minimum energy trajectory solution is chosen based on the input for each mode by selecting the optimal actuation method as well as the control approach for each flight mode. It is accomplished by first using the INDI control with the composite configuration in the ascent mode, where the airship travels for 77 seconds to reach the reference altitude. Then, the airship cruises for 27 seconds using only the gondola actuator with the backstepping control. Finally, the airship descends for 20 seconds using the composite configuration with the backstepping control.

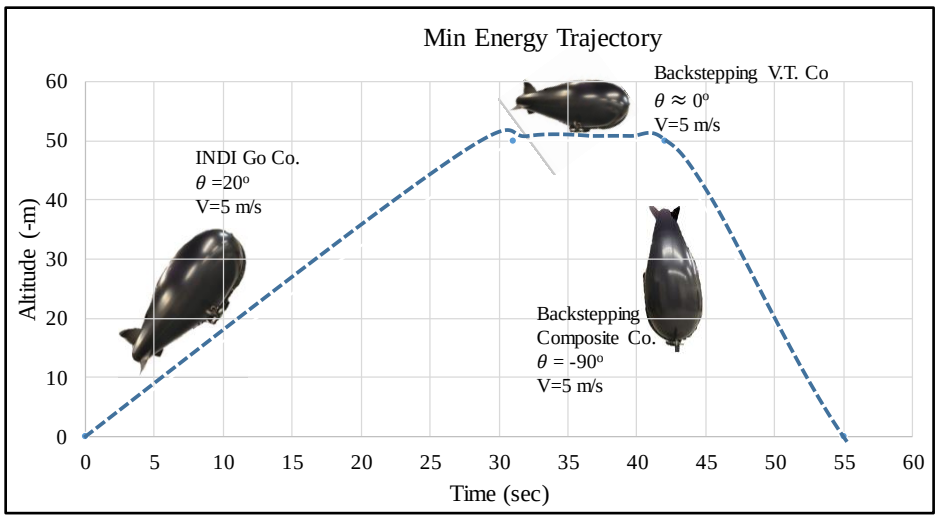


Figure 7.3 Minimum Energy Trajectories in the presence of wind

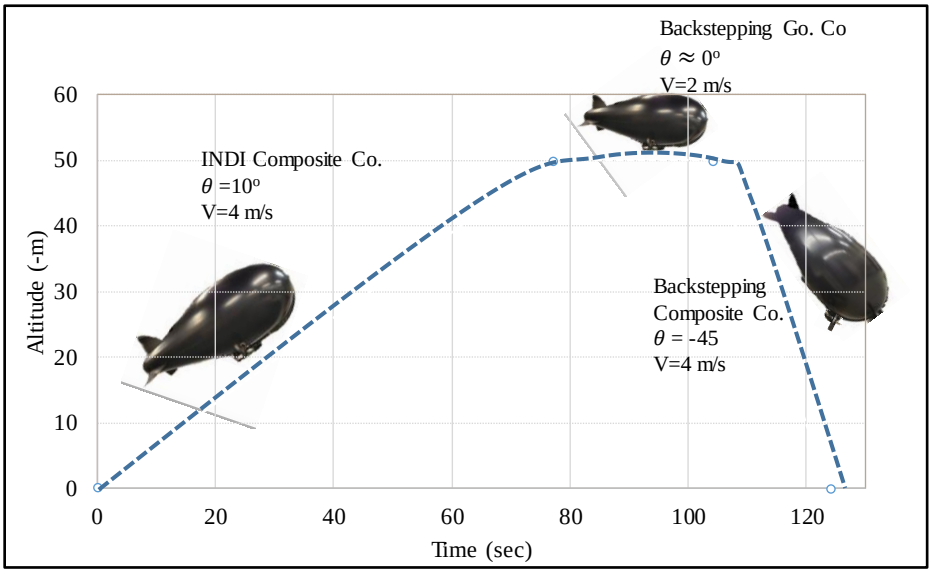


Figure 7.4 Minimum energy trajectories based on reference input in the presence of wind

The results of the energy consumption during the ascent mode demonstrate the efficiency of all the configurations used. As can be seen, the results of the INDI control using the gondola and the composite configurations were found to require less energy than the backstepping control when using the same configurations, as shown in the two scenarios mentioned above. In calm conditions, the vectored thrust actuator results show similar energy consumption to those of the gondola, yet

it consumes more energy in windy conditions. During cruise mode, there is a very small difference in energy consumption for all configurations. The composite configuration shows the lowest energy consumed during the descent mode for both controllers, while backstepping control is more effective than INDI control in the descent mode.

The results also show that wind conditions are a major factor in energy consumption, as seen in the different tables in Appendix E. In addition, traveling time has a proportional impact on energy consumption as seen in Equations (6.1- 6.3). The constant power in these equations is consumed throughout the flight time, however, as the travel time increases, the aggregate amount of energy consumed by on-board components naturally rises as well, resulting in an increase in total energy consumption. Another factor influencing energy consumption is the propulsion system. This system consumes power throughout the flight time since it is the only system that generates the thrust to move the airship and enable it to overcome the drag. Therefore, minimizing the drag and/or the airship velocity can result in significant energy savings from the propulsion system, as seen in Equation (3.61). The results show that the energy consumed by the drive mechanism of the gondola is very low compared to the energy consumed by the propulsion system. Equation (3.63) shows that the power required to move the gondola is independent of the velocity of the airship. However, the primary factors affecting energy consumption in this system are the payload weight and the gondola repositioning rate $\dot{S}$. In this study, the payload weight is considered constant, therefore the primary factor affecting energy consumption during the movement of the gondola is the gondola repositioning rate. In the thrust vector control configuration, the energy is consumed by the tilting system and the propulsion system. Together, the tilting system and the propulsion system can produce a pitching moment, lift, and create a forward thrust at the same time. The results show that altitude tracking overshoots when only the vectored thrust is used, thus indicating

that the vectored thrust has a rapid response, and that the required thrust is high. Accordingly, this configuration requires more energy than the moving gondola control configuration, in all study cases, because the pitch response of the moving gondola mechanism is very stable and is less affected by the flight conditions.

The comparison of the minimum energy trajectory in windy conditions with each individual actuator, for each controller, is illustrated in Table 7.1. The results show that the composite control required the least amount of energy to complete the flight trajectory for both controllers. The INDI shows a small improvement between the composite and the gondola actuators. In the case of the INDI using the vectored thrust, the airship was unstable during the descent therefore the table only shows the results for the ascent and cruise modes.

Table 7.1 The minimum energy trajectory of each actuators in windy condition

Actuators / Controllers	Gondola KWs	Vectored Thrust KWs	Composite KWs
Backstepping Control	11.275	5.597	3.968
INDI Control	4.284	> 5.932	4.279

Using the algorithm above to find the optimal trajectory improves the energy consumption of the entire flight by approximately 17%.

It is evident that the composite configuration control provides the most effective control solution for the proposed airship, enabling it to complete the flight missions while respecting the defined criteria.

Chapter 8: **Experimental Platform**

In this Chapter, the equipment used for the experimental test will be discussed. The experimental test platform used in this research can be classified into three components: the physical properties of the airship, the hardware, and the software. These components will be discussed in detail in the following sections.

8.1 The physical properties of the airship

The airship was designed based on shape optimization methodology for rapid descent with lower drag and optimum surface area. The actual airship is shown in Figure 8.1. This prototype was fabricated in the mechanical engineering laboratory of the University of Ottawa. The physical properties of the airship are listed in Appendix B. The airship architecture is composed of two main components: a helium envelope with a rigid keel and moving gondola capable of travelling along the keel.

The envelope of the airship is made of a polyester polyurethane fabric with a thickness of 0.1016 mm. This is one of the most suitable materials to be used for uninhabited airship envelopes. However, the airship still encounters the loss of buoyancy over time. The loss of lifting gas affects buoyancy and produces a decrease in altitude. Therefor, it is important to include in the model a deferential term of the buoyancy and the total mass loss rate to keep a state of equilibrium. The buoyancy loss test was performed in the lab and it is discussed in the Appendix C.

Figure 8.1 Airship prototype

The result of buoyancy loss test shows that the loss of lifting gas affects buoyancy and produces a decrease in altitude. This can play a major role in the controllability during the descent. Therefore, the model requires an additional term to include the contribution of the total mass loss rate.

The gondola is made of carbon fiber of 5 mm thickness and it is shown in Figure 8.2. The shape of the gondola is streamlined to reduce the drag and the corners are smoothed to avoid skin friction drag. The propulsion systems are attached to the gondola in both sides by a link made of carbon fiber as well. The top view of the gondola when it is attached to a test keel is shown in Figure 8.3. All the sensors and the microcontroller are mounted inside the gondola and will be discussed in next section.

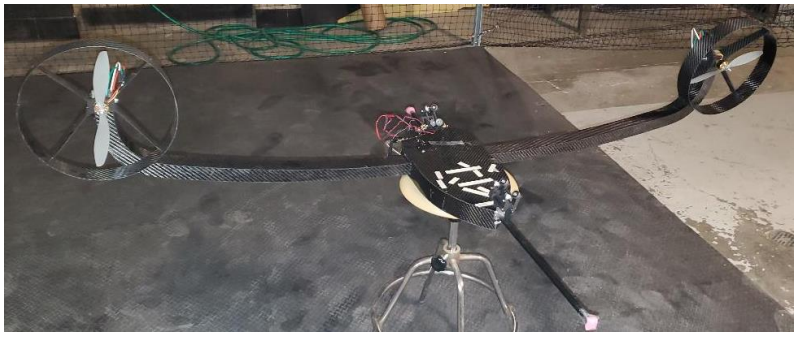


Figure 8.2 the gondola and drive mechanisms

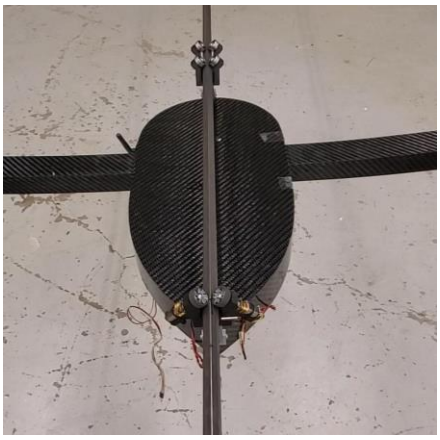


Figure 8.3 top view of the gondola

8.2 Hardware

The hardware architecture on-board the airship includes a set of sensors and devices for data processing and data link with the ground station. The onboard components include a CPU, sensors, actuators, and a communications subsystem are shown in Figure 8.4. Most of the components are mounted inside the gondola as seen in Figure 8.5.

All the components used in the current airship are commercially available and are listed in Table 8.1 along with the references to where they can be found. However, they are all discussed in detail in the Appendix C.

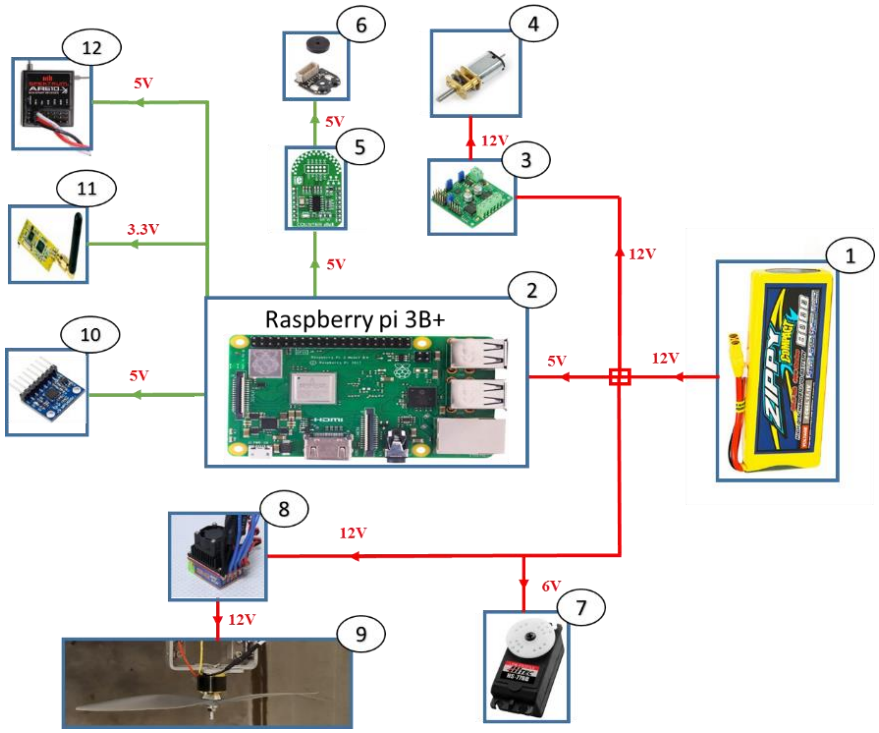


Figure 8.4 hardware architecture

Figure 8.5 on-board hardware inside the gondola

Table 8.1 hardware components

	Component description	Quantity	References
1	ZIPPY Compact 8000mAh Battery	2	[135]
2	Raspberry Pi 3 Model B+	1	[136]
3	TReX Jr Dual Motor Controller	1	[137]
4	Brushed DC gearmotors 1000:1	2	[138]
5	Counter click	1	[139]
6	Magnetic Encoder	1	[140]
7	HS-77BB Servo motor	2	[141]
8	Standard 45A car ESC	2	[142]
9	Rimfire 400 Brushless motor	2	[143]
10	MPU6050	1	[144]
11	Radio Frequency Module	2	[145]
12	AR610 Aircraft Receiver RF 7020	1	[146]

8.2.1 MPU6050 (10)

The (Inertial Measurement Unit) IMU used in this platform is MPU6050 and it shown in Figure 8.6. The IMU is powered by 5V from the Raspberry Pi and communicates with the Raspberry Pi using Inter-Integrated Circuit (I2C) protocol. The IMU is composed of 3-axis gyroscope and 3-axis accelerometer to estimate the airship's attitude and angular rates. The gyroscope provides angular velocity raw data $(\dot{\phi}, \dot{\theta}, \dot{\psi})_{\text{gyro}}$ and the accelerometer provides axial acceleration raw data $(\ddot{X}, \ddot{Y}, \ddot{Z})_{\text{acc}}$. The IMU is assumed to be located in the center of the gondola in order to provide the

most accurate data. However, the IMU output contains biases and noise. Therefore, it is important to initialize the sensor data to eliminate the bias and the noise before using them in the controller. However, this bias can be removed by initializing the sensors. The initialization process is to ensure that the accelerometer should be reading zero acceleration in the X and Y direction, and 9.81 m/s^2 in the Z direction, and the gyroscope should read zero rotational velocities in all three axes. The accelerometer and the gyroscope can both provide the same angular position however, the gyroscope drift over time due to the integration of the angular velocity and accelerometer is noisy at high frequency. In order to calculate an accurate orientation data, the gyroscope and accelerometer data are combined. The celebrated gyroscope date is integrated by multiply the angular rate measurement by the sample time to get the angular position. The angular position can be calculated from the accelerometer date by using the following equations;

$$\phi_A = \tan^{-1}\left(\frac{\ddot{Y}_{acc}}{\sqrt{\ddot{X}_{acc}^2 + \ddot{Z}_{acc}^2}}\right)$$

$$\theta_A = \tan^{-1}\left(\frac{\ddot{X}_{acc}}{\sqrt{\ddot{Y}_{acc}^2 + \ddot{Z}_{acc}^2}}\right)$$

where

ϕ_A is the measured roll angle from the accelerometer

θ_A is the measured pitch angle from the accelerometer

In order to calculate an accurate orientation value, a complimentary filter is used. The filter is very easy and light to implement making it perfect for embedded systems. The idea of the complimentary filter is to pass the accelerometer data through a low-pass filter and the gyroscope

data through a high-pass filter and combine them to give the final angular rate as seen in Figure 8.7. The measured angles by using the complementary filter are;

$$\theta = k * \left(\theta + \theta_{gyro}\right) + (k-1)\theta_{acc}$$

$$\phi = k * \left(\phi + \phi_{gyro}\right) + (k-1)\phi_{acc}$$

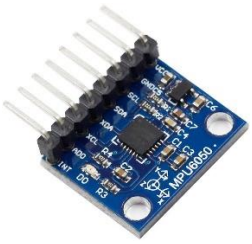


Figure 8.6 MPU6050

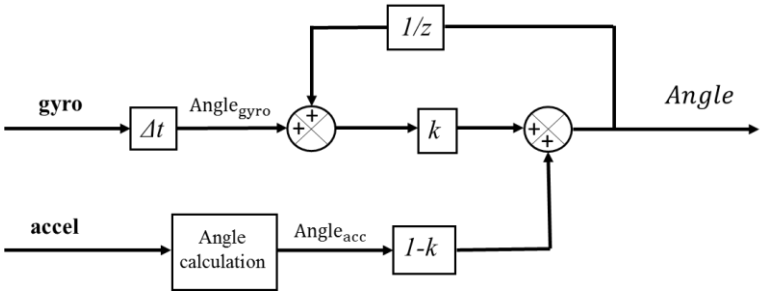


Figure 8.7 Complimentary filter implementation

The constant *k* in the equations is a tunable constant (0< *k* <=1) and it is chosen experimentally to ensure that the measurements are correct and not drifting. The filter will update the pitch and roll angles every iteration by taking *k* % of the gyroscope data and add (*k*-1) % of the angle calculated by the accelerometer. This will ensure the smoothness and quick updates of the integrated gyro measurements to remove the bias before growing too large. Using the complementary filter compensates for pitch and roll drift but not the yaw drift. This drift will result in a growing error in yaw and yaw angular rate. These errors cause control output saturation and airship instability

after a while. Most of commercially available IMU sensors use magnetometers to sense earth magnetic field direction and provide an absolute measurement of the yaw angle however it is out of scope of this study.

8.3 Software

Airship control software is implemented in Matlab/Simulink environment. Matlab/Simulink runs on the ground station (laptop) provides the communication and visualization mechanism in real time between the user and the system onboard (Raspberry Pi) of the airship. In order to establish the communication between the Simulink and the Raspberry Pi, Simulink support package for Raspberry Pi should be installed first. The dynamic model of the system can be developed on the Simulink and deploy it to the Raspberry Pi using the automatic code generation for standalone execution on the Raspberry Pi. The algorithm runs on the Raspberry Pi can read and send sensors data to the ground station and execute the autonomous flight control strategies and sending commands to the actuators. The ground station sends commands and mission paths to the onboard system and it also receives sensor data from the airship, displaying them in real time during actual flight by an AR610 receiver which is connected to the ground station through a serial port as illustrated in Figure 8.8. The data is received at a baud rate of 57600 by the receiver on the ground station to provide a real-time data of the airship states. An Arduino compatible 5V relay module is used to switch between the automatic and manual modes. This relay module takes a control signal between 2.5 and 5 Volts and controls a DC or AC signal, either closing the circuit when the control signal is on or opening the circuit when the control signal is off.

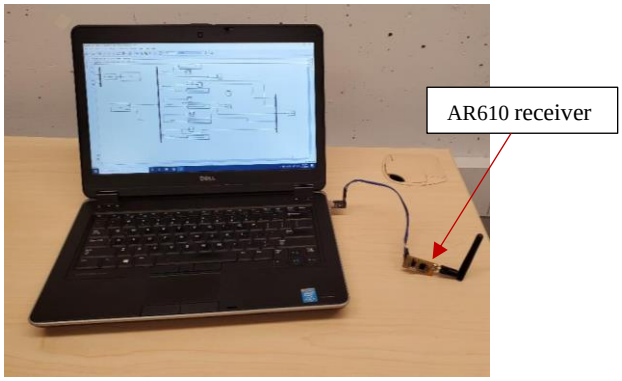


Figure 8.8 Ground station

Figure 8.9 Arduino compatible 5V relay module [141]

In the automatic mode, a block diagram to switch between enable and disable the automatic mode, when enabling the automatic mode, the ESCs receive the PWM signals from the Raspberry Pi and provide power to the motors according to the flight mission.

In the manual mode, a disable is chosen in the ground station and the Remote Control (RC) is turned ON. The ESCs in the manual mode receive the PWM signals from the AR610 Aircraft Receiver and provide power to the motors according to the user inputs.

The radio control (RC) is used by the operator for manual control to take over the control of the airship in case of a software or a hardware failure. The Spektrum DXe RC used in this research is shown in Figure 8.10. Once the Remote Control (RC) turned ON, the manual mode is activated and override the onboard system control. The RC send the signals to the ESCs of drive system

through the Receiver. The wireless radio connection with the ground station and the onboard control hardware are shown in Figure 8.11.

Figure 8.10 Spektrum DXe RC

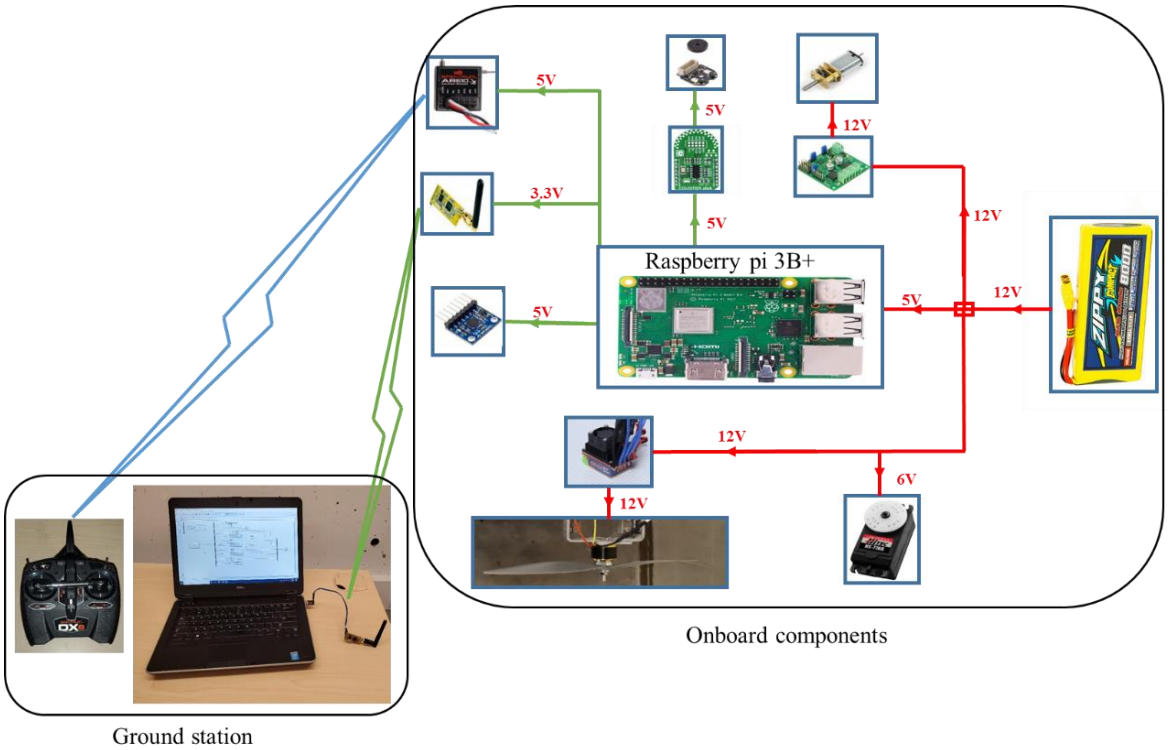


Figure 8.11 onboard components of the airship and ground station

8.4 Experimental setup

In this section, the experimental setup is used to evaluate the controllers' approaches developed in Chapter 4. The controllers are implemented onboard of the airship platform proposed in Figure 2.3 and several experimental tests are performed. All tests were performed in the laboratory in the University of Ottawa and were limited to pitch trajectory tracking due to the limitation of the test area. The test area used for the experimental tests is shown in Figure 8.12. Flights in the campus sports dome were not permitted.

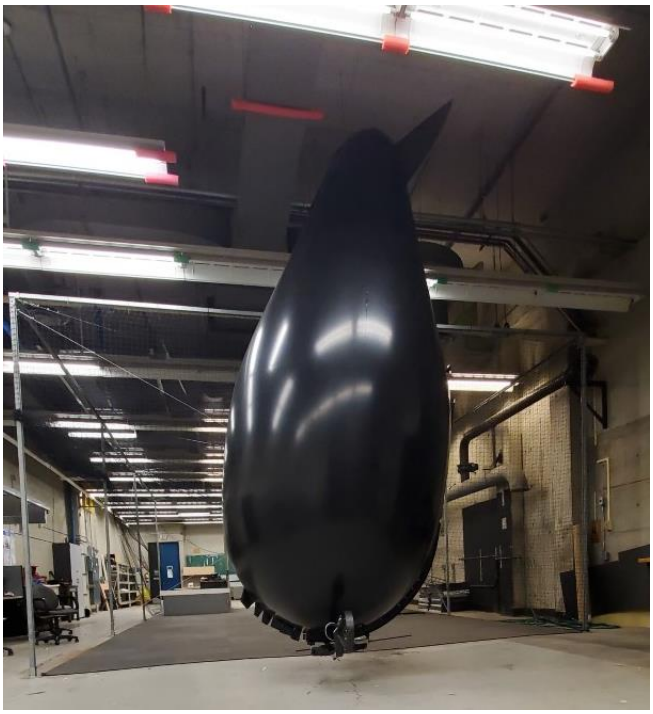


Figure 8.12 Lab test area

8.4.1 Design of experiments

The objective of these flight tests is to show that the designed airship, equipped with the moving gondola, can be controlled by the proposed controllers and is capable of achieving large pitch angle variations. The reference trajectories considered in the experimental tests are sinusoidal trajectory and step reference at different pitch angles and different environmental conditions. The controllers

were tested on the first iterations of the gondola positioning mechanism. This mechanism relies on friction between the driving wheels on the gondola and the rail on the airship. The high frictions lead to numerous driving motor failures when moving from high to low pitch angles. Therefore, consistent results could only be obtained for small repeating pitch variations or large pitch down variations. As a result, the airship was first set to track a sinusoidal reference input with amplitude of 5^o and then to track different reference angle inputs (10^o, -10^o, -20^o, -30^o, -40^o, -50^o, -60^o, -65^o, -80^o). The reference signal was created on the ground station by the signal builder block in the Simulink. Each test is terminated when the airship tracks and maintains the desired angle.

The gondola travels along two sections of the keel, straight and curved sections. The pitch reading is taken directly from the IMU sensor along the straight section of the keel because the IMU is parallel to the longitudinal axis. However, when the gondola reaches the curved section, sensor readings alone can not be used. In this section, the pitch angle is obtained by subtracting the curvature angle θ_{Geo} from the IMU reading.

$$\theta = \theta_{imu} - \theta_{Geo} \qquad s_g > 47 \text{ cm} \tag{8.1}$$

where θ_{Geo} (expressed in radians) is calculated as the gondola position divided by the radius of curvature *R*:

$$\theta_{Geo} = \frac{(s_g - 47)}{R}$$

Due to the uncertainties in curvature measurement, an empirical method was used to verify (8.1). At 10 cm increments of the gondola position along the curved keel, the relationship between the sensor angle and the manually measured angle of the airship was recorded, and a fifth order

polynomial interpolation was computed. The results of the empirical method shown in Figure 8.13 along with the results of (8.1).

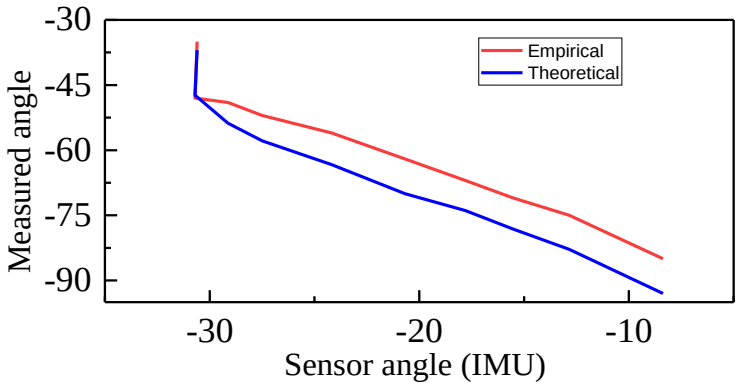


Figure 8.13 empirical and theoretical measured angle of the airship in degrees

The tracking results were obtained by using both airship pitch angle calculation methods on the curved keel. The initial position of the gondola for each test is $s_g = 0$ with $\theta = 0^o$. Once the model is uploaded to the Raspberry Pi through Wi-Fi communication, the system is ready to execute the reference commands from the ground station.

8.4.2 Implementation

The dynamic model of the system and the controller are developed first on Simulink and then uploaded to the Raspberry Pi using the automatic code generation for standalone execution on the Raspberry Pi. The sampling time was set to 0.01seconds, the Raspberry Pi runs at 100 Hz and the IMU sends the data at same frequency. The RF is connected to a laptop where a Simulink interface allowed a person on the ground station to send reference commands and receive sensors data and the control measurements on the airship in real time. The communications between the ground station and the airship are via a 20 Hz shared frequency RF link. In addition, the log data stored in the Raspberry Pi can be accessed and saved wirelessly from the ground station. All the actuators

and the sensors were tested first on the platform individually to be calibrated and to check their performances. The encoder model was uploaded first on the Raspberry Pi that runs at 20 Hz to test the gondola's position and the wireless communication between the Raspberry Pi and the ground stations was 10 Hz. In this case, the time delay between sending the reference commands and receiving the sensor data is very small and it is barely noticeable. Similarly, when testing the thrust and the servo model, the best performance with almost no time delay was found when the Raspberry Pi was running at 20 Hz and the wireless communication was at 10 Hz. However, when all the components were integrated in one model, using similar frequencies produced excessive time delays. Therefore, the effective frequency utilized for all components integrated in one model was 100 Hz and the wireless communication between the Raspberry Pi and the ground stations was 20 Hz. However, there is still 1-2 seconds delay in radio communications.

Figure 8.14 shows the architecture of the embedded control system of the all components. The airship position was estimated and updated by using Kalman filter as described in Appendix C. However, the GPS model was removed from the platform since the experiments were conducted indoor and the position control could not be performed in the test area. Different safety procedures were considered before conducting the experimental test including a mixed manual/automatic control mode as discussed in the Section 7.3. In addition, two safety switch circuit breakers are added to the driving mechanism of the gondola as shown in Figure 8.15. Each end of the gondola has a safety switch installed to prevent the gondola from moving further than either end of the keel if the system fails for any reason.

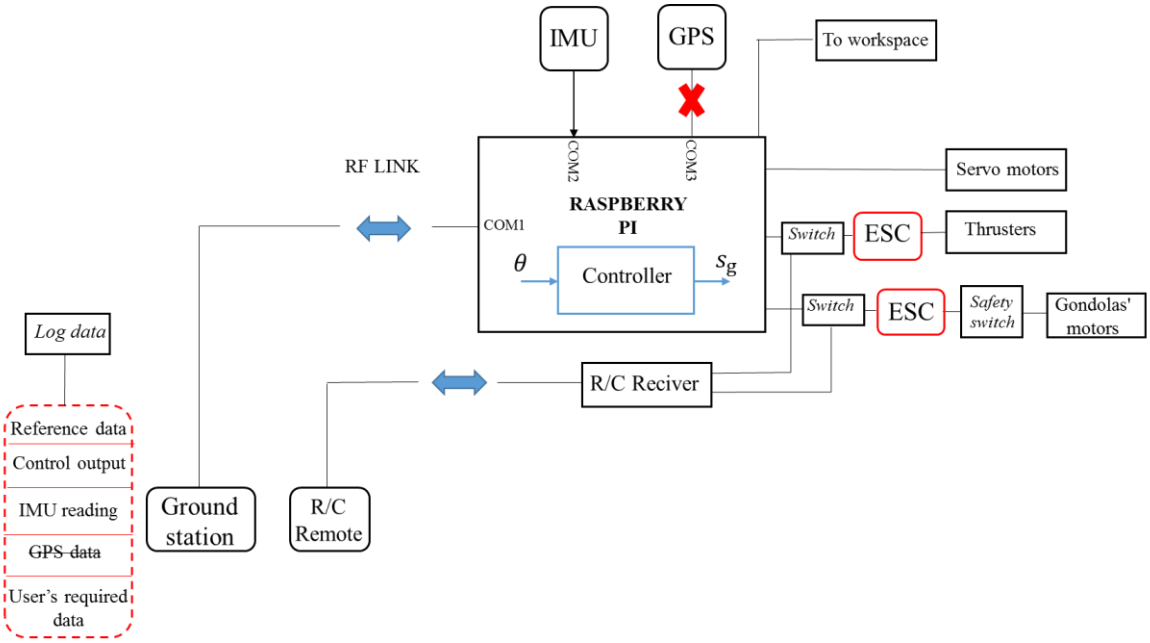


Figure 8.14 Architecture of the embedded control system

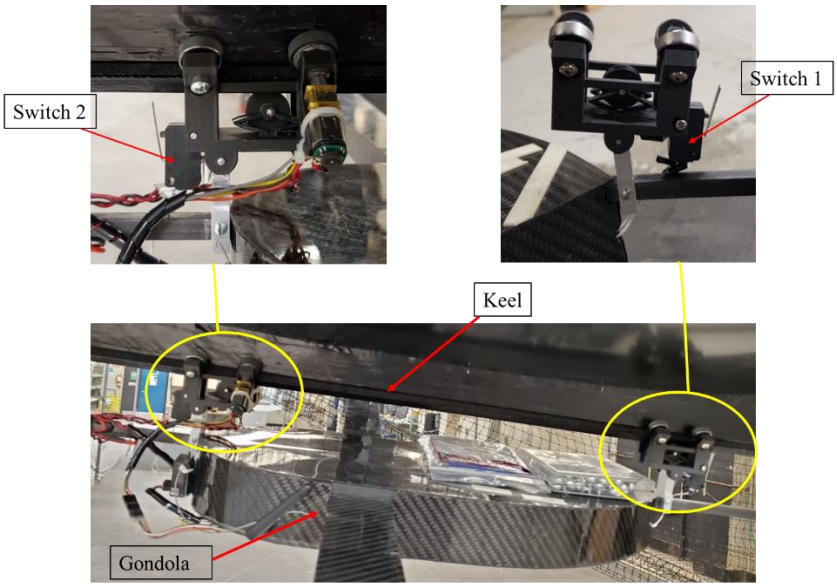


Figure 8.15 driving mechanism of the gondola with safety switch circuit breakers

The gondola's position is obtained based on wheel rotation using a magnetic encoder. The encoder is added to one of the brushed DC gearmotor with the extended back shaft. More details about the encoder can be found in Appendix C. The signal sent to the ESC to control the motor speed is calculated by the error between the encoder reading and the controller output and using PID control. In order to read the signals from the encoders properly, a high sampling frequency is required. Therefore, the best encoder's operating frequency tested with less delay that affects the system was found 100 Hz. The signal varies between 0-180 which is the maximum motor command values for forward and backward directions and the neutral value is 92. The ESC signal is calculated from the PID approach as,

$$u(t) = k_p(t)e + k_i \int_0^t e(t)\,dt + k_d \frac{de(t)}{dt}$$

where k_p is the proportional gain, k_i is the integral gain, k_d is the derivative gain, and the error e is

$$e = s_{\mathrm{d}} - s_{\mathrm{g}}$$

The signal is then normalized as,

$$u_s = (0.88\,u + 92)$$

where the constant 0.88 was calculated from the linear equation to map the ESC signal (0-180).

The PID gains used in the experiments are found by trial and error (k_p =40, k_i = 1.5, k_d=0.5).

Chapter 9: **Experimental Results and Discussion**

In this chapter, the pitch tracking performance of each of the controllers is evaluated in this chapter using the experimental setup described in the previous Chapter. The backstepping controller's experimental results are presented first in Section 8.1, followed by the INDI controller's experimental results in Section 8.2. Section 8.3 concludes with a discussion of the results.

9.1 Backstepping experimental results

The backstepping controller was implemented first in the Raspberry Pi. The test results of sinusoidal reference input and the tracking trajectory are shown in Figure 9.1. The control output response of the sinusoidal trajectory in form of the gondola's position s_g is shown in Figure 9.2. The figure shows that airship started from zero pitch angle at position of the gondola s_g= 0 and then the gondola moves to the positive direction up to 10 cm to generate a negative pitch angle of -5^o and similarly it moves to the opposite direction until it reaches about -10 cm and eventually goes back to initial position at the end of the trajectory. The gains used for this test were K_1 =5, K_2 =5. The controller was able to follow the pitch trajectory with a steady-state error of ±1^o. The blue line in Figure 9.2 is the encoder reading and the red line is controller output.

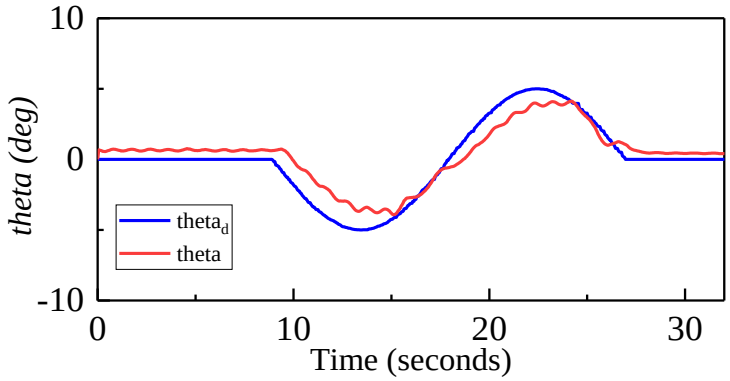


Figure 9.1 Reference and pitch angle

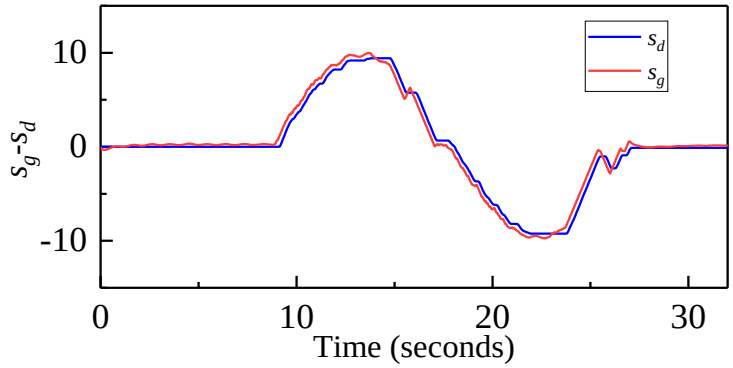


Figure 9.2 Gondola's position

The experimental results at different pitch angles are shown in Figure 9.3 through Figure 9.6. The objective of the test is to run the airship until the desired angle is tracked and maintained. In all tests, the reference signal is received approximately 2 seconds after it is transmitted from the ground station. The results show that the controller was able to track the reference pitch with a steady-state error of ±1°. However, the controller was able to effectively track the trajectory with very small oscillations. Therefore, the gains were determined iteratively and may not be optimal. The results were improved by choosing different gains for each test. Similar gains to the sinusoidal test were used in case of step reference angles up to -30°. For step reference angles greater than

-40°, gains of K_1 =1, K_2 =0.5 were found to be effective in reducing steady-state error and oscillations. The results in Figure 9.5 shows that the controller used the sensor reading of the airship state until it reaches the angle 30° which corresponding to position of the gondola at 47 cm and then used a fifth order polynomial interpolation computed from the empirical method illustrated in Figure 8.13. The reference angle of -40° was achieved when the gondola reached the position of 56 cm which is located on the curved keel. Figure 9.6 show that the reference angle of 80° was achieved when gondola traveled for 27 s to reach the position of -135 cm.

Please note experimental results of the angle -20°, -30°, -50°, -60°, -65° can be found in Appendix D. In addition, due to minor differences in results between the empirical and theoretical methods, all results obtained using the theoretical method have been moved to Appendix DD to avoid duplication.

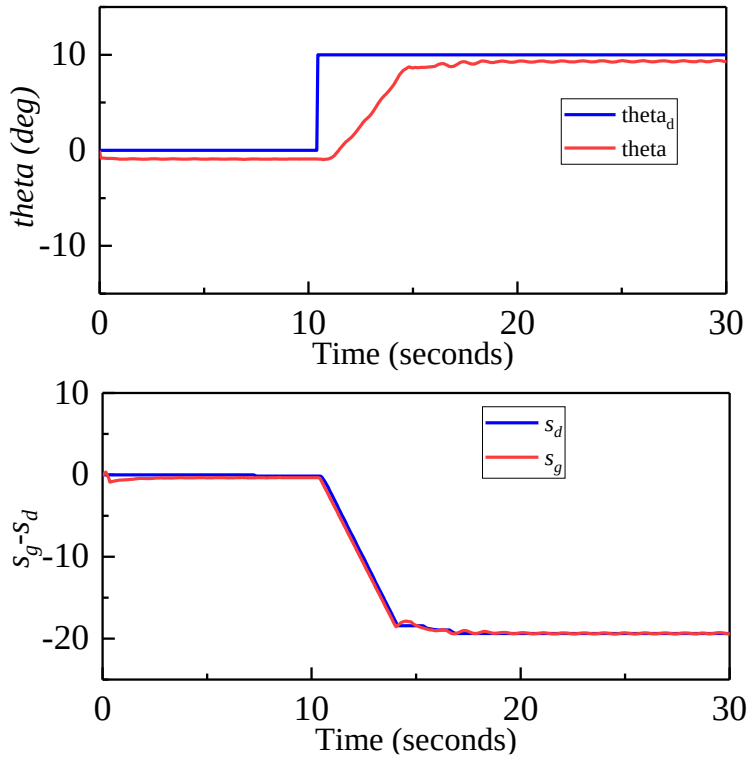


Figure 9.3 Flight test trajectory of 10° and gondola's position

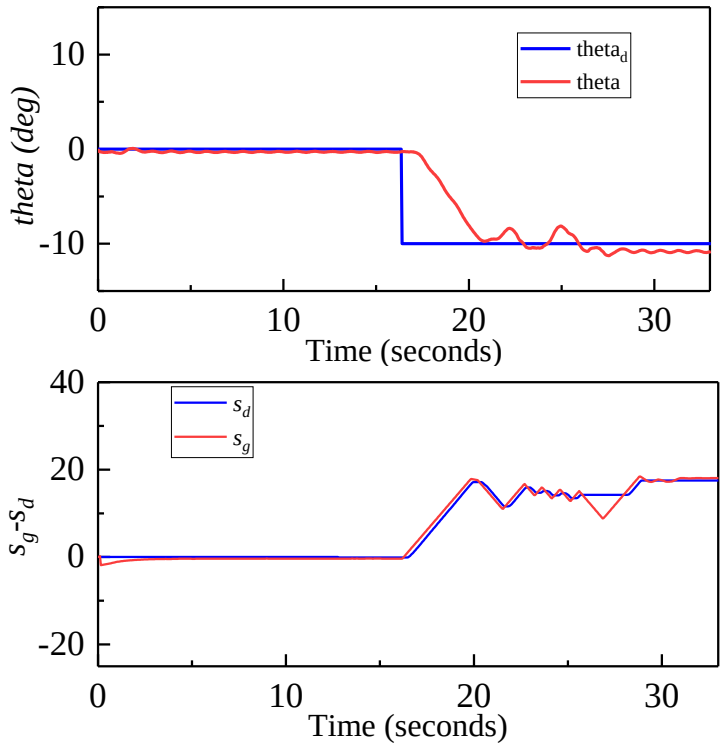


Figure 9.4 Flight test trajectory of -10° and gondola's position

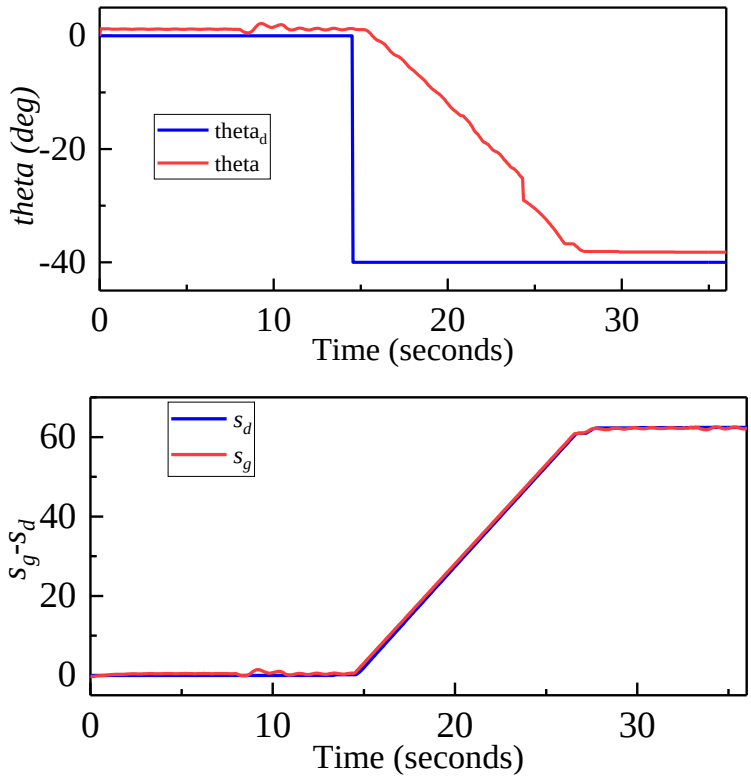


Figure 9.5 Flight test trajectory of -40° and gondola's position

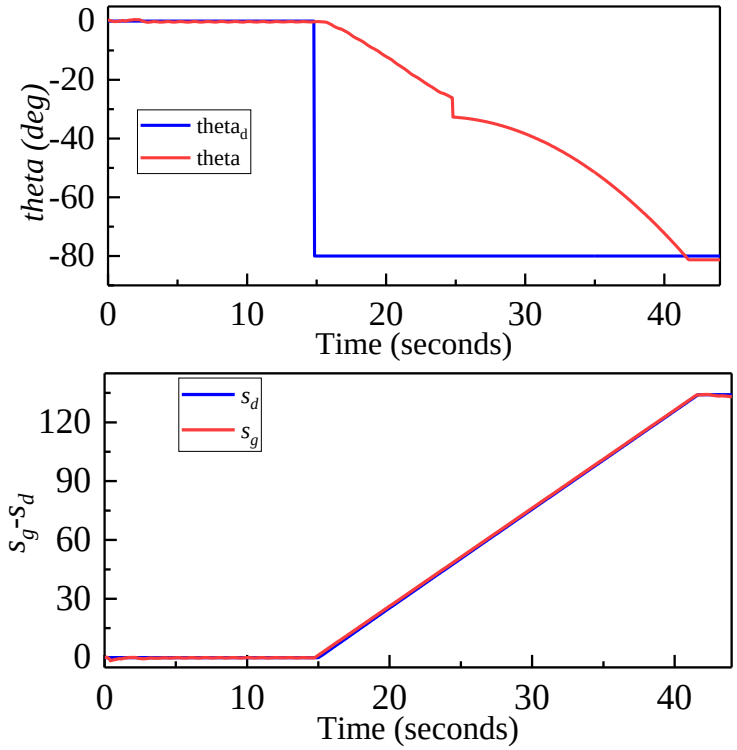


Figure 9.6 Flight test trajectory of -80^{o} and gondola's position

In order to test the controller's robustness to wind disturbances, the airship is subjected to external source of wind. It was noticed during the experimental test that horizontal wind parallel to y-axis does not affect the pitch stability therefore the wind was applied parallel to z-axis. The wind was applied by a 20" box fan on the tail of the airship to generate a pitch moment. The wind was applied for 2 s and removed for 2s and this process was repeated 3 times to generate large-amplitude oscillations. The results in Figure 9.7 and Figure 9.8 show the control response to track the pitch reference angles of -5^{o} and 5^{o} in the presence of wind. The wind is applied after the reference angle is tracked. The steady-state error is about 1.5^{o}. The experimental results show that the control effort was effective in following the reference trajectory with a maximum amplitude error of 3^{o} and quickly stabilising the airship when the external wind was removed.

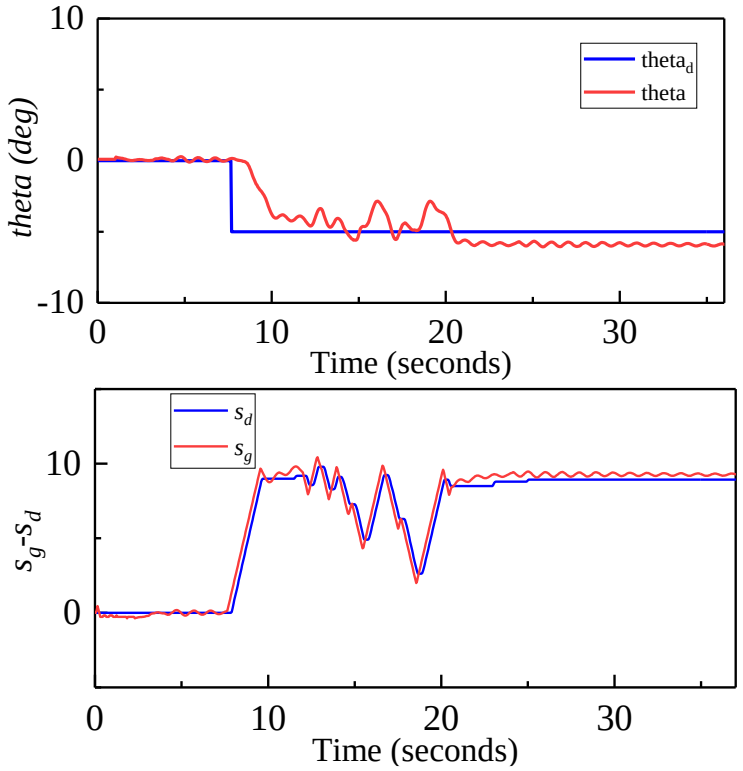


Figure 9.7 Flight test trajectory of -5° and gondola's position in the presence of wind

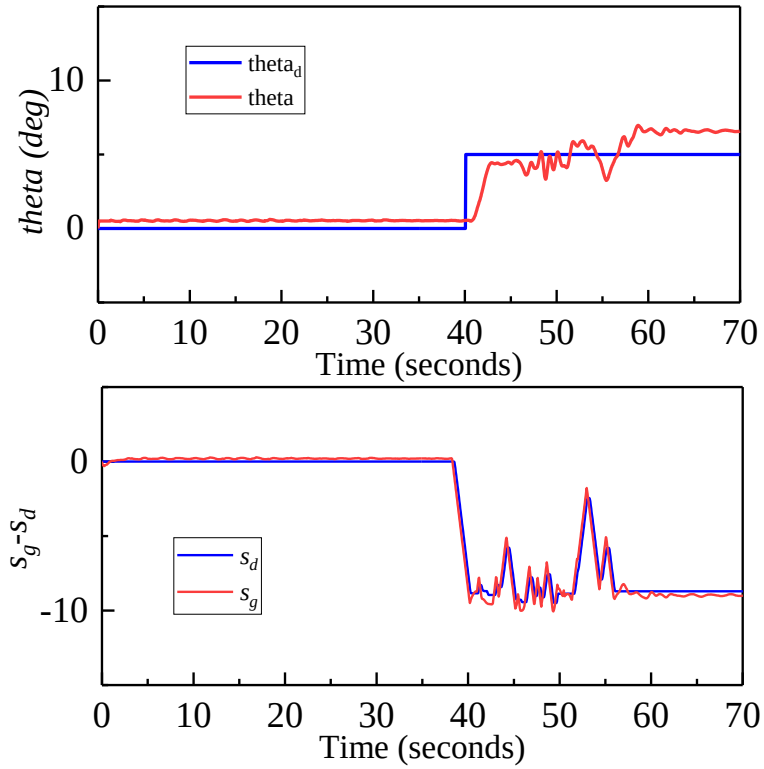


Figure 9.8 Flight test trajectory of 5° and gondola's position in the presence of wind

9.2 INDI experimental results

The INDI for pitch control only requires knowledge of the kinematics and the actuator dynamics. The INDI control requires the angular acceleration reading from the sensor and the virtual control. The angular acceleration reading from the accelerometer contains biases and noise therefore the IMU angles were measured first by a complimentary filter as discussed in section 8.2.1. The angular acceleration was then calculated by integrating the pitch angle twice. This method of obtaining the angular acceleration was found to be simple and reliable. The virtual control was found from a simple P control structure as explained in section 3.2. Figure 9.9 shows the controlled system block diagram of the experimental setup.

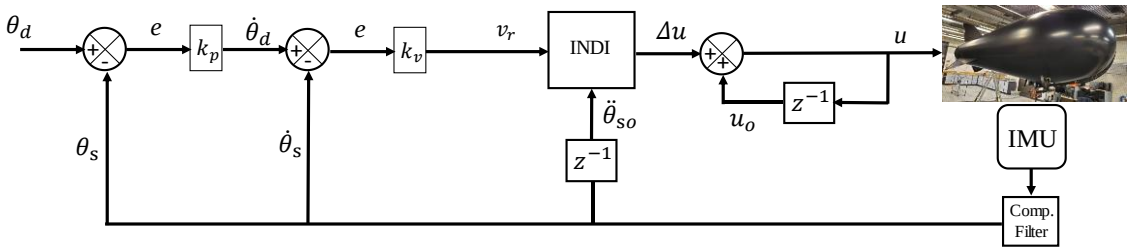


Figure 9.9 Controlled system block diagram

The experimental tests were carried out with reference trajectories that were similar to those used in the backstepping control tests. The gains used for this test were chosen by trial and error K_θ = 0.5, $K_{\dot{\theta}}$ = 0.1. Figure 9.10 shows the test results of sinusoidal reference input and the tracking trajectory as a response of the control output. The control output response of the sinusoidal trajectory in form of the gondola's position s_g is shown in Figure 9.11. The controller was able to follow the pitch trajectory finish the mission with a maximum error of ±1.5°. The steady-state error was 0.5°.

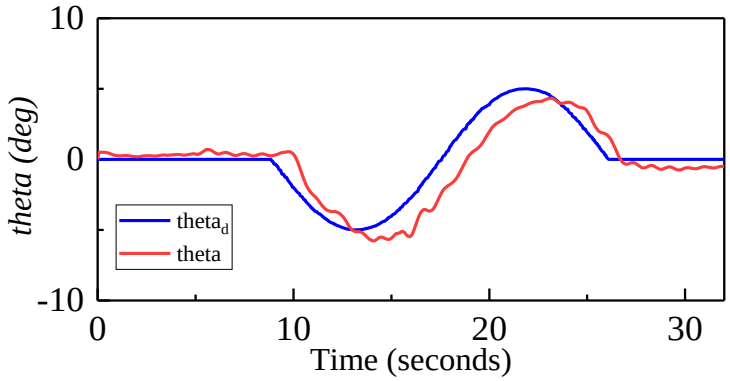


Figure 9.10 reference and pitch angle

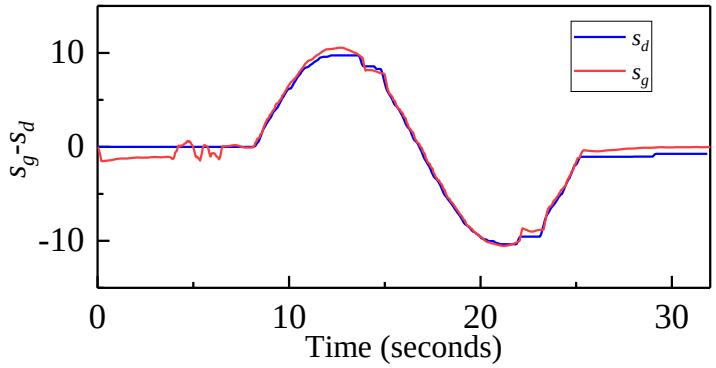


Figure 9.11 Gondola's position

The controller's output was also evaluated using the different step reference inputs. The experimental results of INDI control are shown in Figure 9.12 through Figure 9.15. The results, on the other hand, show that the controller was able to track the reference pitch with very small oscillations. The controller has a smooth tracking performance with very relatively no saturation however, a high amplitude oscillation that causes control saturation when tracking the step reference of -80^{o} is observed. The best solution for the performance degradation issue due to control saturation found in literature to be effective is in [147]. A weighting parameter η is added to the INDI controller in (3.27) as,

$$u = u_0 + \eta\, \Delta u$$

The parameter is intended to reduce the controller's sensitivity to actuation errors. As can be seen in Figure 9.15 adding the weighting parameter η = 0.4, improves the tracking result of the controller. Figure 9.15 show that the reference angle of -80^o was achieved when gondola traveled for 13 s to reach the position of -136 cm. Using the gains of K_θ= 0.1, $K_{\dot{\theta}}$ = 0.1 in case of tracking the reference angles of -65^o and -80^o, the steady-state errors were reduced to 0.5^o.

Please note experimental results of the angle -20^o, -30^o, -50^o, -65^o can be found in Appendix D. Also, all the results using the theoretical method are moved to appendix D.

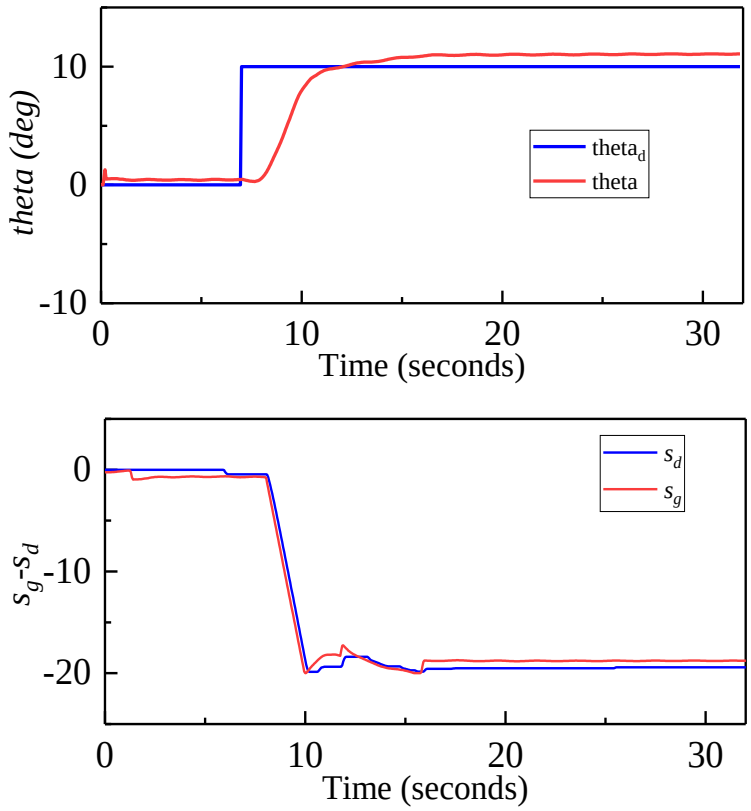


Figure 9.12 Flight test trajectory of 10^o and gondola's position

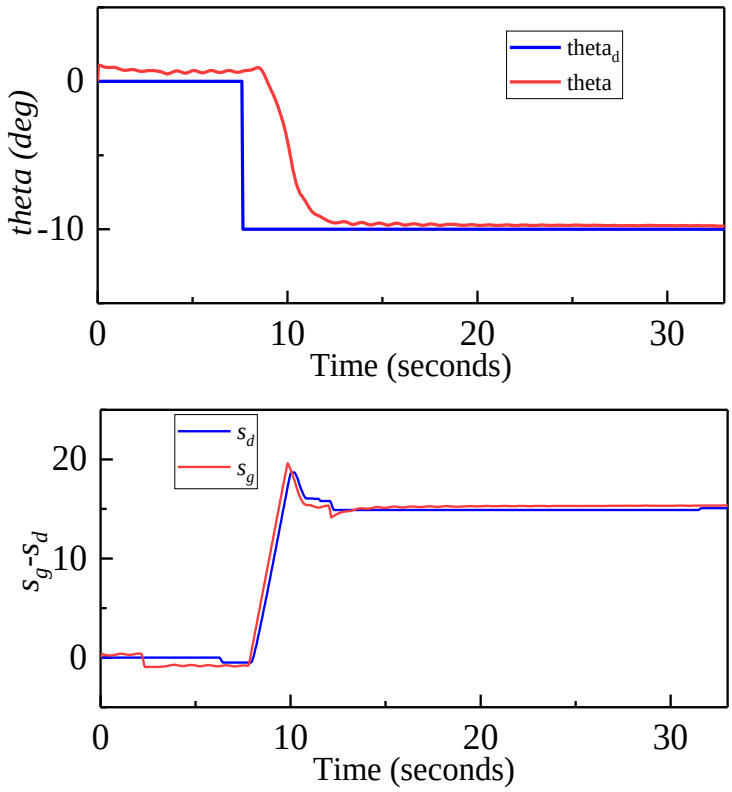


Figure 9.13 Flight test trajectory of -10^{o} and gondola's position

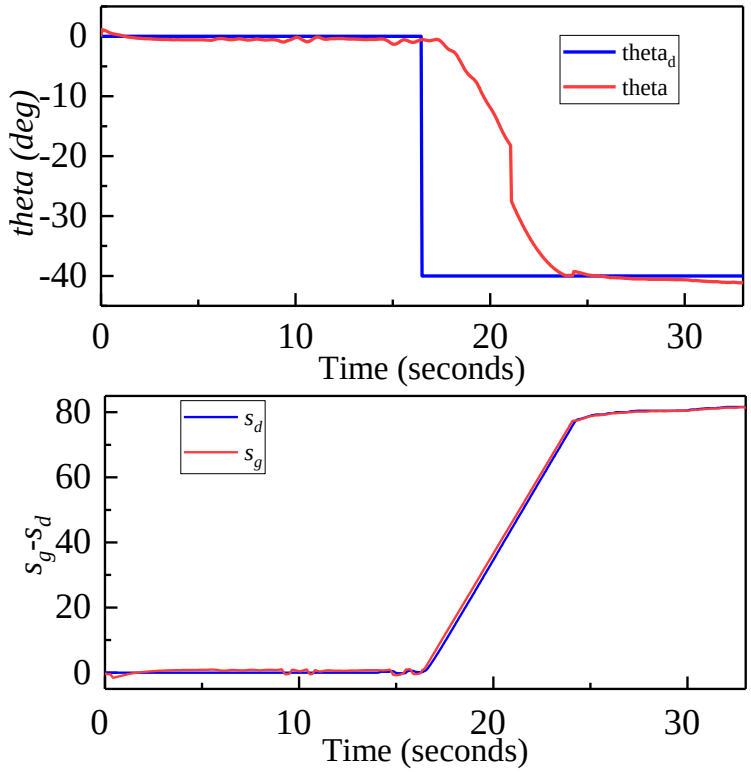


Figure 9.14 Flight test trajectory of -40^{o} and gondola's position

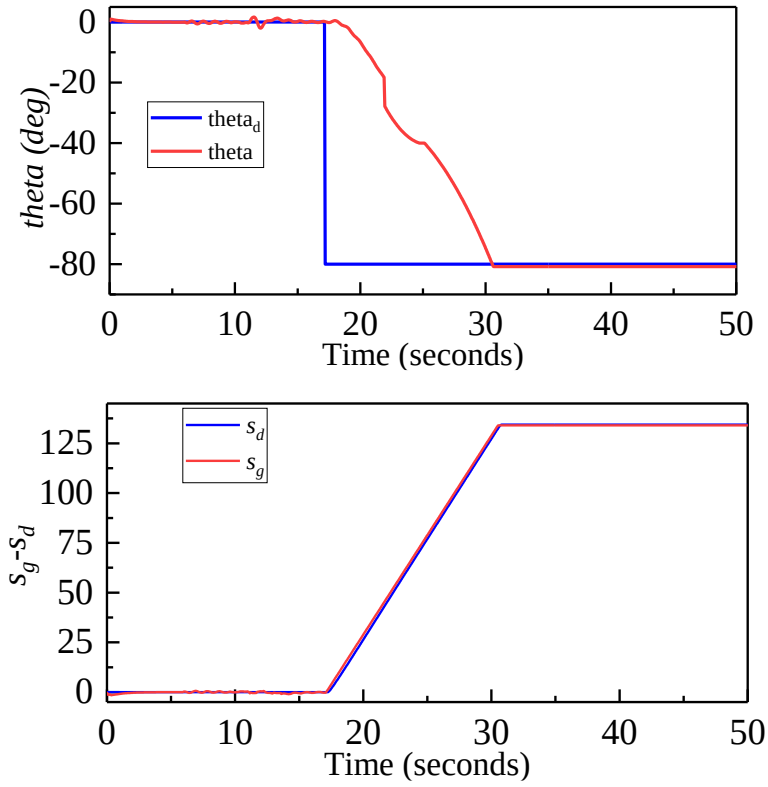


Figure 9.15 Flight test trajectory of -80^{o} and gondola's position

Similar to the backstepping controller experiments, the INDI controller was tested on the airship in the presence of wind. The results in Figure 9.16 and Figure 9.17 show the control response to track the pitch reference angles of -5^{o} and 5^{o} in the presence of the external wind. The airship reached the steady-state with an error of 1.2^{o} after the wind source was removed. The results obtained show that the proposed controller is able to track the reference commands with a maximum amplitude error of 2^{o} and quickly stabilize the system.

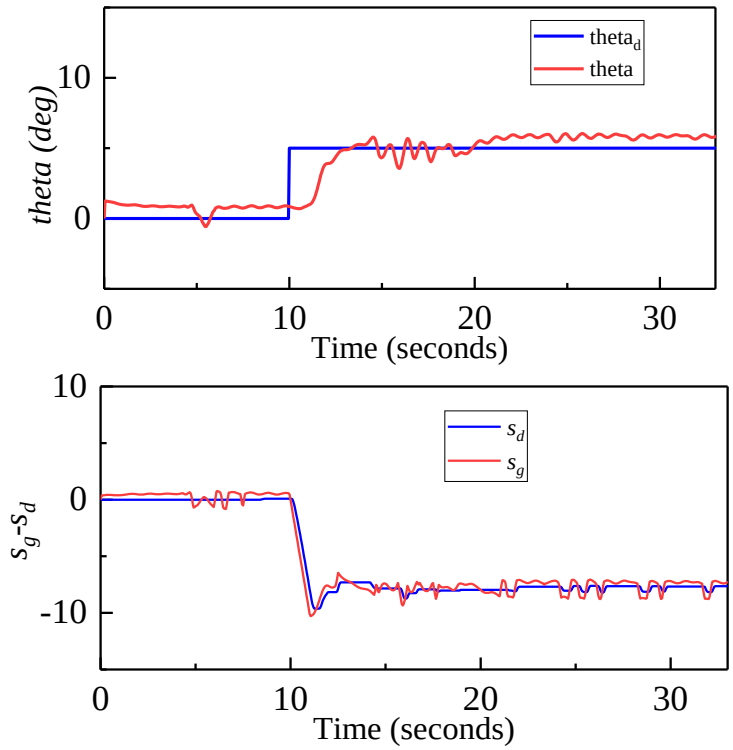


Figure 9.16 Flight test trajectory of 5° and gondola's position in presence of wind

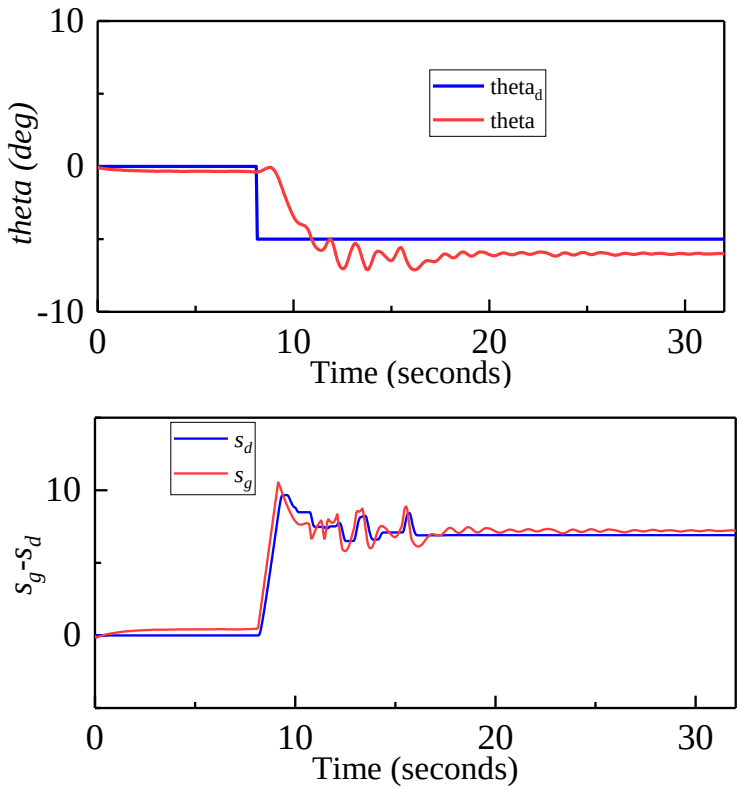


Figure 9.17 Flight test trajectory of -5° and gondola's position in presence of wind

9.3 Result Discussion

The main difference between the controllers is their dependency on the dynamic model. Backstepping is dependent on the accuracy of the dynamic model, whereas INDI is mainly dependent on sensor reading and the actuator dynamic. The controllers were evaluated and compared using similar input signals from the ground station. The reference trajectories and tracking results were presented above. Although both controllers are able to complete the mission, there is a noticeable difference in their performance.

The initial results of the backstepping controller were not satisfactory, and the airship became unstable very quickly. The instability was mainly due to the different frequencies of the onboard components, which caused excessive time delays. Hence, the initial results showed the sensitivity of the INDI control to the time delay and the control frequency. The results then improved when the time delay of both actuators and the angular rate were synchronized, and the frequency of 100 Hz was used. The tracking performance was also found to be sensitive to the controller gains that design the virtual control laws. If the control causes actuator saturation, the control commands can be accelerated or decelerated by using the suitable choice of control gains, which can speed up or slow down the system.

Compared to the backstepping controller, the INDI controller's response is fast for all step inputs. Figure 9.12 and Figure 9.13 show that the INDI controller only required 2 seconds to track the reference input of -10^{o} or 10^{o}, while backstepping required about 4.5 seconds, as shown in Figure 9.3 and Figure 9.4. In addition, the INDI required 14 seconds to track the input command of -80^{o} and backstepping required only 26 seconds to track a similar reference command. The INDI is basically tracking the angular acceleration, and when an acceleration error is detected, the INDI controller will increment the control inputs in order to rapidly renew the reference tracking.

Moreover, the selection of optimal gains leads to a faster and more stable control response. Due to the value of the gains used (K_1, K_2), the backstepping response is relatively slow. When K_1 was set to a high value, the control response was fast, but this creates instability in the system, which then oscillates and diverges. Furthermore, when K_1 is set to a low value, the speed response is too slow, resulting in an overshoot in speed and requiring a long time to reach the desired position. The system is stabilized by choosing a small value for K_2, but this causes a small position error that must be corrected. Therefore, the gains used for the experimental tests and considered to be optimal for step reference angles less than -40^o are K_1 =5, K_2 =5 and for step reference angles greater than -40^o are K_1 =1, K_2 =0.5. In the presence of external wind disturbances, both controllers show satisfactory results. However, the INDI controller quickly compensates for the wind since it uses sensor measurements. The effects of the actuator saturation can be unpredictable especially on the curved keel. In the case of the INDI control, the control effectiveness is what relates the actuator inputs to angular accelerations, and it includes the inertia of the airship. The control saturation was observed when tracking the step input of -80^o, which was improved by adding the weighting parameter η. Almost no overshoot was observed in all tracking results for the backstepping controller. The INDI controller clearly has a faster rise time and less overshoot.

As mentioned earlier, the pitch angle of the airship was calculated using an empirical method as well as a geometric method when the gondola is on the curved keel. To evaluate the difference between both methods, the results of both controllers were also obtained in the curved keel using the geometric method in (8.1) for the step inputs -40^o and up. The results of this comparison are illustrated in Appendix D. For both controllers, a similar tracking performance is observed between the geometric method and the empirical method. The sensor's pitch angle reading remains almost constant between -30^o and -40^o, as shown in Figure 8.13. This behaviour is observed in all of the

results when using the geometric method equation. At the beginning of the curved keel, the gondola moves for about 10 cm without a change in the pitch, which generates a vertical line.

In reality, when the gondola moves in that range, it does affect the airship's pitch. The angle of the airship was detected in this range and the results were less affected when using the empirical method.

In general, it is obvious that no controller performs better or worse than the other, throughout the mission. Nonetheless, the main advantage of the INDI controller is model independence, which improves the robustness of this controller to model mismatches.

Chapter 10: **Conclusion and Future Work**

Studying different concepts for a rapid altitude change is necessary for energy saving aircraft. Unlike the conventional airship which is driven by propellers, rudders and elevators, a moving mass concept is considered in this thesis. The moving mass concept has been used in different successful applications, such as, in underwater vehicles, and missiles. The concept of shifting the centre of gravity by moving mass to increase the system's manoeuvrability and generate a rapid ascent and descent without needing to add considerable number of components motivated this research to study two different airship architectures equipped with a moving gondola. The goal of this thesis was to perform a systematic analysis of optimal flight trajectories for uninhabited airships to achieve a longer aerial endurance with lower energy consuming.

The non-linear dynamic model of the proposed architectures based on the physical characteristics of each model was developed using Newton-Euler equations. Two nonlinear controllers were designed to evaluate the impact of the moving gondola and the vectored thrust to track a desired trajectory. Backstepping and incremental nonlinear dynamic inversion (INDI) controllers were chosen because of their dependency of the dynamic model. Backstepping control requires an accurate dynamic model of the complete system, whereas INDI control only requires the measurements of the dynamic model. The effectiveness of the controllers in different environmental conditions was investigated using numerical simulations. Through the simulations, pitch of the airship was mainly controlled by the moveable gondola and the altitude is controlled by the vectored thrust. The simulation results showed that both controllers were able to accomplish the entire mission successfully, but with significantly different performances.

The airships used in this study are equipped with two actuation methods (moving gondola and vectoring thrust) that can be used to preform different flight modes. The efficiency of each actuator at completing the flight missions was compared. The purpose of this comparison was to demonstrate the impact of the moving gondola used in the current platform when compared to that of the fixed gondola used in other airships with different pitch control systems (e.g., thrust vectoring). Furthermore, the effectiveness of a composite control strategy that integrates both actuators to achieve similar flight missions was also included in the comparison. Extensive simulations were conducted using the actual physical properties of prototype Model 2. Both backstepping and INDI controllers, were used to evaluate the tracking performance of the different actuators. The results obtained from both controllers demonstrate that all actuation methods were capable of successfully completing the ascent and cruise missions. However, the energy consumed when using the INDI control during the ascent mode (with the various actuation methods and in the presence of wind) was lower than the energy consumed when using the backstepping control. In addition, the results show that the most efficient control during the ascent, in the presence of wind, was the INDI control. When used with the gondola actuator, only 2.1 KWs of energy was consumed. In the case of the cruise mode, the most efficient control was the backstepping control with the vectored thrust, consuming 0.44 KWs, and the lowest consumption in the descent mode was the backstepping control with the composite configuration, consuming only 0.8 KWs of energy.The effectiveness of the composite actuation method was demonstrated by completing the flight mission using the least amount of energy, for both controllers.

The total energy consumption, as see in Equation (6.1), is proportional to the energy consumed by the on-board components (constant energy) which is approximately 60% of the total energy as well as the energy required by the propulsion system. Because these systems are running the entire

flight time, it is not surprising that travel time has a significant impact on energy consumption. The results show that as the pitch angle increases, the airship reaches the reference altitude faster and consumes less energy. That is due to reorienting the airship, using the moving gondola, to the direction of motion reduces the angle of attack, therefore reducing the drag force which in turn decreases the energy consumption. As a result, using lower-power electronic components would reduce the overall energy consumption of the current system as well as any future airships that employ a similar method.

An optimization method was developed to obtain the optimum design configuration for a minimal energy flight scenario, while respecting the design constraints. The Heuristic technique was used to ensure that the desired goal was reached within acceptable limits of flight time and energy consumption. This method used data from the results of the extensive simulations conducted to measure energy consumption during different flight scenarios, in order to develop simplified trajectory models. These models capture the general structure of the expected optimal trajectory, enabling an optimization over a small number of parameters to be performed quickly. Using this method to find the optimal trajectory improved the energy consumption of the entire flight by approximately 17%. This method was developed to quickly generate a minimum energy trajectory for onboard implementation.

This optimization method can be used to quickly compute near-optimal trajectories on other airships with similar actuators. Furthermore, if data from various flight scenarios is available, it can be applied to any airship.

The airship prototype Model 2 was manufactured and used for the experimental test to evaluate the pitch tracking performance of each of the controllers. The controllers were implemented onboard the airship platform and several experimental tests were performed. The dynamic model

of the system and the controller were developed first on Simulink and then uploaded to the onboard system (Raspberry Pi) of the airship. The AR610 receiver was connected to a laptop where a Simulink interface allowed a person on the ground station to send reference commands and receive sensor data and control measurements from the airship, in real-time.

The controllers were set to track a sinusoidal trajectory and step input at various pitch angles. The results obtained showed that the proposed controllers were able to track the reference commands successfully. However, the effects of actuator saturation can be unpredictable, especially on the curved keel. For the INDI control, the results showed a control saturation when tracking a step input of more than -65^{o}, but this was improved by adding the weight parameter. Moreover, both controllers showed satisfactory results in the presence of external wind disturbances.

The experimental results were found to be sensitive to the onboard frequency and the time delay, requiring further investigation when future work is conducted on this platform. The results of this study were improved when the time delay of both the actuators and the angular rate were synchronized and a frequency of 100 Hz was used.

The gondola's movement relies on friction between the driving wheels on the gondola and the rail surface. In the current prototype, the holding torque of the motors keeps the gondola in position. The motors used in this study are designed for high holding torque and do not require a lot of power supply voltage to hold them in position. However, we noticed a slipping between the wheels and the rail surface as the gondola moved from high to low pitch angles. This is due to the friction force is insufficient to move the gondola backward. Theoretically, the solution is to increase the friction force by using wheels or rail surfaces with a higher coefficient of friction. Increasing the friction force, on the other hand, results in a significant increase in power consumption. Furthermore, it causes a high number of driving motor failures as experienced during the

experimental test. Therefore, for future study, larger motors are required that can supply a higher friction force to overcome the weight resistance of the gondola including all the equipment used, or using a different moving system, such as rack and pinion system to replace the rail surface and the motors' wheels for high accuracy and to avoid slipping.

References

[1] T. Lutz and S. Wagner, “Drag Reduction and Shape Optimization of Airship Bodies,” *J. Aircr.*, vol. 35, no. 3, pp. 345–351, May 1998.

[2] P. González, W. Burgard, R. Sanz, and J. L. Fernández, “Developing a low-cost autonomous indoor blimp,” *J. Phys. Agents*, vol. 3, no. 1, pp. 43–52, 2009.

[3] E. C. De Paiva, J. R. Azinheira, J. G. Ramos, A. Moutinho, and S. S. Bueno, “Project AURORA: Infrastructure and flight control experiments for a robotic airship,” *J. F. Robot.*, vol. 23, no. 3–4, pp. 201–222, 2006.

[4] A. B. Moutinho, “Modeling and Nonlinear Control for Airship Autonomous Flight,” PhD thesis, Universidade Tecnica De Lisboa Instituto Superior Tecnico, 2007.

[5] J. B. Mueller, “Design and Analysis of Optimal Ascent Trajectories for Stratospheric Airships,” PhD thesis, University Of Minnesota, 2013.

[6] L. Chen, G. Zhou, X. J. Yan, and D. P. Duan, “Composite Control Strategy of Stratospheric Airships with Moving Masses,” *J. Aircr.*, vol. 49, no. 3, pp. 794–801, 2012.

[7] F. Wang, J. Zhou, and J. Miao, “Pitch handling characteristics of the stratospheric airship based on moving mass control,” *Chinese Control Conf. CCC*, vol. 2016-Augus, pp. 10915–10920, 2016.

[8] T. Petsopoulos, F. J. Regan, and J. Barlow, “Moving-Mass Roll Control System for Fixed-Trim Re-Entry Vehicle,” *J. Spacecr. Rockets*, vol. 33, no. 1, pp. 54–60, 1996.

[9] C. A. Woolsey and N. E. Leonard, “Moving mass control for underwater vehicles,” *Proc. Am. Control Conf.*, vol. 4, pp. 2824–2829, 2002.

[10] P. G. Thomasson, “Equations of Motion of a Vehicle in a Moving Fluid,” *J. Aircr.*, vol. 37, no. 4, pp. 630–639, 2000.

[11] P. K. Menon, G. D. Sweriduk, E. J. Ohlmeyer, and D. S. Malyevac, “Integrated guidance and control of moving-mass actuated kinetic warheads,” *J. Guid. Control. Dyn.*, vol. 27, no. 1, pp. 118–126, 2004.

[12] H. Di Xiao, P. P. Mi, L. Bin Liu, and S. Cao, “A new method of strengthening longitudinal static stability of airship in ascent,” *Appl. Mech. Mater.*, vol. 687–691, pp. 212–215, 2014.

[13] E. Lanteigne, W. Gueaieb, D. Robillard, and S. Recoskie, “Unmanned airship design with sliding ballast: Modeling and experimental validation,” in *2016 International Conference on Unmanned Aircraft Systems (ICUAS)*, 2016, pp. 1246–1253.

[14] E. Paiva, F. Benjovengo, S. Bueno, and P. Ferreira, “Sliding Mode Control Approaches for an Autonomous Unmanned Airship,” in *18th AIAA Lighter-Than-Air Systems Technology Conference*, 2009, p. 13082.

[15] M. T. Frye, S. M. Gammon, and C. Qian, “The 6-DOF dynamic model and simulation of the tri-turbofan remote-controlled airship,” *Proc. Am. Control Conf.*, pp. 816–821, 2007.

[16] G. Khoury and J. D. Gillett, *Airship Technology*. Cambridge,UK: CambridgeUniversity Press, 1999.

[17] C. V. Ram and R. S. Pant, “Multidisciplinary Shape Optimization of Aerostat Envelopes,” *J. Aircr.*, vol. 47, no. 3, pp. 1073–1076, 2010.

[18] J. Waishek, A. Dogan, and Y. Bestaoui, “Investigation into the Time Varying Mass Effect on Airship Dynamics Response,” in *47th AIAA Aerospace Sciences Meeting including The New Horizons Forum and Aerospace Exposition*, 2009.

[19] E. Hygounenc, I.-K. Jung, P. Souères, and S. Lacroix, “The Autonomous Blimp Project of LAAS-CNRS: Achievements in Flight Control and Terrain Mapping,” *Int. J. Rob. Res.*, vol. 23, no. 4, pp. 473–511, 2004.

[20] J. DELAURIER and D. SCHENCK, “Airship dynamic stability,” in *3rd Lighter-Than-Air Systems Technology Conference*, 1979.

[21] S. Gomes, “An Investigation of the Flight Dynamics of Airships with Application to the YEZ-2A.,” Cranfield Institute of Technology, 1990.

[22] Y. Li, M. Nahon, and I. Sharf, “Airship dynamics modeling: A literature review,” *Prog. Aerosp. Sci.*, vol. 47, no. 3, pp. 217–239, Apr. 2011.

[23] Y. Bestaoui Sebbane, *Lighter than Air Robots*, vol. 58. Dordrecht: Springer Netherlands, 2012.

[24] Yuwen Li, "Dynamics Modeling and Simulation of Flexible Airships," PhD thesis, McGill University, 2008.

[25] N. Azouz, Y. Bestaoui, and O. Lemaitre, "Dynamic analysis of airships with small deformations," in *Proceedings of the Third International Workshop on Robot Motion and Control, 2002. RoMoCo '02.*, 2002, pp. 209–215.

[26] A. B. Moutinho, "Modeling and Nonlinear Control for Airship Autonomous Flight," Universidade Tecnica De Lisboa Instituto Superior Tecnico, 2007.

[27] A. Moutinho, J. R. Azinheira, E. C. de Paiva, and S. S. Bueno, "Airship robust path-tracking: A tutorial on airship modelling and gain-scheduling control design," *Control Eng. Pract.*, vol. 50, pp. 22–36, May 2016.

[28] C. Stockbridge, A. Ceruti, and P. Marzocca, "Airship research and development in the areas of design, structures, dynamics and energy systems," *Int. J. Aeronaut. Sp. Sci.*, vol. 13, no. 2, pp. 170–187, 2012.

[29] L. Liao and I. Pasternak, "A review of airship structural research and development," *Prog. Aerosp. Sci.*, vol. 45, no. 4–5, pp. 83–96, 2009.

[30] 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)*, 1999, vol. 3, pp. 2328–2335.

[31] M. Battipede, M. Lando, and P. Gili, "Mathematical modelling of an innovative unmanned airship for its control law design," *IFIP Int. Fed. Inf. Process.*, vol. 202, pp. 31–42, 2006.

[32] X. Wu, C. H. Moog, L. A. Marquez-Martinez, and Y. Hu, "Full model of a buoyancy-driven airship and its control in the vertical plane," *Aerosp. Sci. Technol.*, vol. 26, no. 1, pp. 138–152, 2013.

[33] José, R. Azinheira, E. C. de Paiva, and S. S. Bueno, “Influence of Wind Speed on Airship Dynamics,” *J. Guid. Control. Dyn.*, vol. 25, no. 6, pp. 1116–1124, 2002.

[34] E. C. Paiva, J. R. Azinheira, and A. Moutinho, “Erratum on ‘Influence of Wind Speed on Airship Dynamics,’” *J. Guid. Control. Dyn.*, vol. 31, no. 2, pp. 443–444, 2008.

[35] S. Hima and Y. Bestaoui, “MOTION GENERATION ON TRIM TRAJECTORIES FOR AN AUTONOMOUS UNDERACTUATED AIRSHIP,” in *4th International Airship Convention and Exhibition*, 2002, pp. 1–12.

[36] S. Hima and Y. Bestaoui, “NONHOLONOMIC Motion Generation on Trim Trajectories for an Autonomous Underactuated Airship,” in *4th International Airship Convention and Exhibition*, 2002, pp. 1–12.

[37] X. Wu, Y. Wang, C. Huang, Y. Liu, and L. Lu, “Experiment and numerical simulation on the characteristics of fluid-structure interactions of non-rigid airships,” *Theor. Appl. Mech. Lett.*, vol. 5, no. 6, pp. 258–261, 2015.

[38] A. U. Haque, W. Asrar, A. A. Omar, E. Sulaeman, and J. S. M. Ali, “Assessment of engine′s power budget for hydrogen powered hybrid buoyant aircraft,” *Propuls. Power Res.*, vol. 5, no. 1, pp. 34–44, 2016.

[39] A. B. Moutinho, L. Mirisola, J. R. Azinheira, and J. Dias, “Project DIVA - Guidance and Vision Surveillance Techniques for an Autonomous Airship,” *Robot. Res. Trends*, pp. 77–120, 2008.

[40] J. Ramos J.G. *et al.*, “Autonomous flight experiment with a robotic unmanned airship,” *Proc. - IEEE Int. Conf. Robot. Autom.*, vol. 4, pp. 4152–4157, 2001.

[41] M. Z. Ashraf and M. A. Choudhry, “Dynamic modeling of the airship with Matlab using geometrical aerodynamic parameters,” *Aerosp. Sci. Technol.*, vol. 25, no. 1, pp. 56–64, Mar. 2013.

[42] Z. Ashraf and M. A. Choudhry, “Dynamic modeling of the airship using analytical aerodynamic model,” *2009 Int. Conf. Emerg. Technol. ICET 2009*, vol. 100, no. 2, pp. 188–193, 2009.

[43] 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)*, 1998, vol. 4, pp. 3462–3467.

[44] J. C. Zufferey, A. Guanella, A. Beyeler, and D. Floreano, “Flying over the reality gap: From simulated to real indoor airships,” *Auton. Robots*, vol. 21, no. 3, pp. 243–254, 2006.

[45] A. Elfes *et al.*, “Project AURORA: development of an autonomous unmanned remote monitoring robotic airship,” *Journal of the Brazilian Computer Society*, vol. 4. pp. 1–16, 1998.

[46] J. R. Azinheira, A. Moutinho, and E. C. De Paiva, “Airship Hover Stabilization Using a Backstepping Control Approach,” *J. Guid. Control. Dyn.*, vol. 29, no. 4, pp. 903–914, 2006.

[47] R. R. Arias, “Modelagem de um dirigível robótico com propulsão elétrica de quatro motores . Modeling of a robotic airship with four electric engines as thrusters,” *Univ. Camp.*, 2015.

[48] J. R. Azinheira, A. Moutinho, and J. R. Carvalho, “Lateral Control of Airship with Uncertain Dynamics using Incremental Nonlinear Dynamics Inversion,” *IFAC-PapersOnLine*, vol. 48, no. 19, pp. 69–74, 2015.

[49] M. Battipede, P. Gili, and M. Lando, “Control Allocation System for an Innovative Remotely-Piloted Airship,” in *AIAA Atmospheric Flight Mechanics Conference and Exhibit*, 2004, vol. 1, pp. 369–381.

[50] V. Singh and E. Lanteigne, “Design and shape optimization of unmanned, semi-rigid airship for rapid descent using hybrid genetic algorithm,” *2019 Int. Conf. Unmanned Aircr. Syst. ICUAS 2019*, pp. 1099–1107, 2019.

[51] J. B. Mueller, M. A. Paluszekt, and Y. Zhao, “Development of an aerodynamic model and control law design for a high altitude airship,” *Collect. Tech. Pap. - AIAA 3rd “Unmanned-Unlimited” Tech. Conf. Work. Exhib.*, vol. 1, pp. 415–431, 2004.

[52] Â. R. Azinheira *et al.*, “Lateral / directional control for an autonomous , unmanned airship,” *Aircr. Eng. Aerosp. Technol.*, vol. 73, no. 5, pp. 453–458, 2001.

[53] A. Elfes, J. Montgomery, J. Hall, S. Joshi, J. Payne, and C. Bergh, “Autonomous Flight Control for a Planetary Exploration Aerobot,” in *Space 2005*, 2005, pp. 1–9.

[54] de Paiva E. Carneiro, S. S. Bueno, and M. Bergerman, “A Robust Pitch Attitude Controller for Aurora’s Semi-Autonomous Robotic Airship,” in *American Institute of Aeronautics and Astronautics 13th Lighter-Than-Air Systems Technology Conference*, 1999, pp. 141–148.

[55] W. Adamski, P. Herman, Y. Bestaoui, and K. Kozlowski, “Control of airship in case of unpredictable environment conditions,” *Conf. Control Fault-Tolerant Syst. SysTol’10 - Final Progr. B. Abstr.*, pp. 843–848, 2010.

[56] T. Takaya, H. Kawamura, Y. Minagawa, M. Yamamoto, and A. Ohuchi, “PID landing orbit motion controller for an indoor blimp robot,” *Artif. Life Robot.*, vol. 10, no. 2, pp. 177–184, 2006.

[57] S. Bouabdallah, A. Noth, R. Siegwart, and R. Siegwan, “PID vs LQ Control Techniques Applied to an Indoor Micro Quadrotor,” *IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, vol. 3, pp. 2451–2456, 2004.

[58] S. Bouabdallah, “Design and Control of Quadrotors With Application To Autonomous Flying,” *École Polytech. Fédérale Lausanne, À La Fac. Des Sci. Tech. L’Ingénieur*, vol. 3727, no. 3727, p. 61, 2007.

[59] L. M. Alkurdi and R. B. Fisher, “Visual Control of an Autonomous Indoor Robotic Blimp,” in *Robotic Vision: Technologies for Machine Learning and Vision Applications*, J. Garcia-Rodriguez and M. A, Eds. IGI Global, 2013, pp. 352–370.

[60] S. Recoskie, “Autonomous Hybrid Powered Long Ranged Airship for Surveillance and Guidance,” PhD theis, University of Ottawa, 2014.

[61] I. Masar and E. Stöhr, “Gain-scheduled LQR-control for an autonomous airship,” *18th Int. Conf. Process Control*, pp. 197–204, 2011.

[62] M. Bergerman, O. Amidi, J. R. Miller, N. Vallidis, and T. Dudek, “Cascaded position and heading control of a robotic helicopter,” in *IEEE International Conference on Intelligent Robots and Systems*, 2007, pp. 135–140.

[63] E. Kulczycki, S. Joshi, R. Hess, and A. Elfes, “Towards Controller Design for Autonomous Airships Using SLC and LQR Methods,” in *AIAA Guidance, Navigation, and Control Conference and Exhibit*, 2006.

[64] A. Moutinho and J. R. Azinheira, “Stability and Robustness Analysis of the AURORA Airship Control System using Dynamic Inversion,” in *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, 2005, pp. 2265–2270.

[65] A. Isidori, *Nonlinear Control Systems*. London: Springer London, 1995.

[66] G. Chowdhary, E. Frazzoli, J. How, and H. Liu, “Chapter 1 Nonlinear Flight Control Techniques for Unmanned Aerial Vehicles,” 1st ed., LAP LAMBERT Academic Publishing, 2012.

[67] D. Acosta and S. Joshi, “Adaptive Nonlinear Dynamic Inversion Control of an Autonomous Airship for the Exploration of Titan,” in *AIAA Guidance, Navigation and Control Conference and Exhibit*, 2007, pp. 1–13.

[68] A. Moutinho and R. Azinheira, “Path Control of an Autonomous Airship Using Dynamic Inversion,” *5th IFAC Symp. Intell. Auton. Veh.*, 2004.

[69] R. K. Yedavalli, P. Shankar, and D. B. Doman, “Robustness study of the dynamic inversion based indirect adaptive control of flight vehicles with uncertain model data,” *Proc. 2003 Am. Control Conf. 2003.*, vol. 2, pp. 1005–1010, 2003.

[70] R. C. van’t Veld, “Incremental Nonlinear Dynamic Inversion flight control,” Delft University of Technology, 2016.

[71] B. Bacon and A. Ostroff, “Reconfigurable flight control using nonlinear dynamic inversion with a special accelerometer implementation,” in *AIAA Guidance, Navigation, and Control Conference and Exhibit*, 2000, pp. 1–5.

[72] X. Wang, E.-J. Van Kampen, Q. P. Chu, and P. Lu, “Stability Analysis for Incremental Nonlinear Dynamic Inversion Control,” in *2018 AIAA Guidance, Navigation, and Control Conference*, 2018.

[73] S. Sieberling, Q. P. Chu, and J. A. Mulder, “Robust flight control using incremental

nonlinear dynamic inversion and angular acceleration prediction," *J. Guid. Control. Dyn.*, vol. 33, no. 6, pp. 1732–1742, 2010.

[74] Z. Liu, J. Guo, M. Li, S. Tang, and X. Wang, "VTOL UAV Transition Maneuver Using Incremental Nonlinear Dynamic Inversion," *Int. J. Aerosp. Eng.*, vol. 2018, pp. 1–19, 2018.

[75] E. Tal and S. Karaman, "Accurate Tracking of Aggressive Quadrotor Trajectories Using Incremental Nonlinear Dynamic Inversion and Differential Flatness," *Proc. IEEE Conf. Decis. Control*, vol. 2018-Decem, pp. 4282–4288, 2019.

[76] P. Lu, E. J. van Kampen, C. de Visser, and Q. Chu, "Aircraft fault-tolerant trajectory control using Incremental Nonlinear Dynamic Inversion," *Control Eng. Pract.*, vol. 57, pp. 126–141, 2016.

[77] P. Smith and A. Berry, "Flight test experience of a non-linear dynamic inversion control law on the VAAC Harrier," in *Atmospheric Flight Mechanics Conference*, 2000, pp. 132–142.

[78] B. J. Bacon, A. J. Ostroff, and S. M. Joshi, "Reconfigurable NDI controller using inertial sensor failure detection & isolation," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 37, no. 4, pp. 1373–1383, 2001.

[79] E. L. Da Silva, "Incremental Nonlinear Dynamic Inversion for Quadrotor Control," Master's thesis, Delft University of Technology, 2015.

[80] R. van 't Veld, E.-J. Van Kampen, and Q. P. Chu, "Stability and Robustness Analysis and Improvements for Incremental Nonlinear Dynamic Inversion Control," in *2018 AIAA Guidance, Navigation, and Control Conference*, 2018.

[81] L. Beji * and A. Abichou, "Tracking control of trim trajectories of a blimp for ascent and descent flight manoeuvres," *Int. J. Control*, vol. 78, no. 10, pp. 706–719, 2005.

[82] J. R. Azinheira, A. Moutinho, and E. C. De Paiva, "A backstepping controller for path-tracking of an underactuated autonomous airship," *Int. J. Robust Nonlinear Control*, vol. 19, no. 4, pp. 418–441, 2009.

[83] E. Kahale, P. C. Garcia, and Y. Bestaoui, “Autonomous path tracking of a kinematic airship in presence of unknown gust,” *J. Intell. Robot. Syst. Theory Appl.*, vol. 69, no. 1–4, pp. 431–446, 2013.

[84] L. Beji, A. Abichou, and Y. Bestaoui, “Stabilization of a nonlinear underactuated autonomous airship-a combined averaging and backstepping approach,” *Proc. 3rd Int. Work. Robot Motion Control. RoMoCo 2002*, pp. 223–229, 2002.

[85] E. Repoulias, Filoktimon Papadopoulos, “Robotic Airship Trajectory Tracking Control Using a Backstepping Methodology,” *IEEE Int. Conf. Robot. Autom.*, pp. 188–193, 2008.

[86] E. Hygounenc and P. Soueres, “Automatic airship control involving backstepping techniques,” in *IEEE International Conference on Systems, Man and Cybernetics*, 2002, vol. 6, p. 6.

[87] T. Liesk, M. Nahon, and B. Boulet, “Design and experimental validation of a nonlinear low-level controller for an unmanned fin-less airship,” *IEEE Trans. Control Syst. Technol.*, vol. 21, no. 1, pp. 149–161, 2013.

[88] G. Murguía-Rendon, H. Rodríguez-Cortés, and M. Velasco-Villa, “Trajectory tracking control for the planar dynamics of a thrust vectored airship,” *Midwest Symp. Circuits Syst.*, pp. 329–332, 2009.

[89] D. Han, X. Wang, L. Chen, and D. Duan, “Command-filtered backstepping control for a multi-vectored thrust stratospheric airship,” *Trans. Inst. Meas. Control*, vol. 38, no. 1, pp. 93–104, 2015.

[90] Y. Liu, Z. Pan, D. Stirling, and F. Naghdy, “Control of autonomous airship,” *2009 IEEE Int. Conf. Robot. Biomimetics, ROBIO 2009*, pp. 2457–2462, 2009.

[91] Y. Yang, J. Wu, and W. Zheng, “Station-keeping control for a stratospheric airship platform via fuzzy adaptive backstepping approach,” *Adv. Sp. Res.*, vol. 51, no. 7, pp. 1157–1167, 2013.

[92] H. Bang, S. Lee, and H. Lee, “Nonlinear trajectory tracking using vectorial backstepping approach,” *2008 Int. Conf. Control. Autom. Syst. ICCAS 2008*, pp. 169–174, 2008.

[93] Y. Yang and Y. Yan, “Neural network gain-scheduling sliding mode control for three-dimensional trajectory tracking of robotic airships,” *Proc. Inst. Mech. Eng. Part I J. Syst. Control Eng.*, vol. 229, no. 6, pp. 529–540, Jul. 2015.

[94] Y. Yang, J. Wu, and W. Zheng, “Trajectory tracking for an autonomous airship using fuzzy adaptive sliding mode control,” *J. Zhejiang Univ. Sci. C*, vol. 13, no. 7, pp. 534–543, 2012.

[95] E. C. de Paiva, F. Benjovengo, and S. S. Bueno, *Sliding Mode Control for the Path Following of an Unmanned Airship*, vol. 40, no. 15. IFAC, 2007.

[96] Y. Yang, J. Wu, and W. Zheng, “Design, modeling and control for a stratospheric telecommunication platform,” *Acta Astronaut.*, vol. 80, pp. 181–189, 2012.

[97] a.J. Healey and D. Lienard, “Multivariable sliding mode control for autonomous diving and steering of unmanned underwater vehicles,” *Ocean. Eng. IEEE J.*, vol. 18, no. 3, pp. 327–339, 1993.

[98] K. Nonami, F. Kendoul, S. Suzuki, W. Wang, and D. Nakazawa, *Autonomous flying robots : unmanned aerial vehicles and micro aerial vehicles.* 2010.

[99] A. Moutinho and J. R. Azinheira, “A gain-scheduling approach for airship path-tracking,” *Lect. Notes Electr. Eng.*, vol. 15, pp. 263–275, 2008.

[100] J. Nakpiam, O. Daskiran, C. Elliott, and A. Dogan, “Airship Waypoint Navigation in the Presence of Wind,” in *AIAA Atmospheric Flight Mechanics Conference*, 2012, pp. 1–27.

[101] T. S. Potdaar, A. S. Sinha, and R. S. Pant, “Controller Design for an Outdoor Autonomous Airship,” in *AIAA Lighter-Than-Air Systems Technology (LTA) Conference*, 2013, pp. 1–11.

[102] W. J. Rugh and J. S. Shamma, “Research on gain scheduling,” *Automatica*, vol. 36, no. 10, pp. 1401–1425, 2000.

[103] R. C. do Valle, L. L. Menegaldo, and A. M. Simões, “Smoothly Gain-Scheduled Control of a Tri-Turbofan Airship,” *J. Guid. Control. Dyn.*, vol. 38, no. 1, pp. 53–61, 2015.

[104] C. Nielsen, “Set stabilization using transverse feedback linearization,” PhD thesis,

University of Toronto, 2009.

[105] C. Nielsen, C. Fulford, and M. Maggiore, “Path following using transverse feedback linearization: Application to a maglev positioning system,” *Automatica*, vol. 46, no. 3, pp. 585–590, 2010.

[106] B. Dadashzadeh, M. J. Mahjoob, M. Nikkhah Bahrami, and C. Macnab, “Stable active running of a planar biped robot using Poincare map control,” *Adv. Robot.*, vol. 28, no. 4, pp. 231–244, 2014.

[107] S. Wiggins, F. John, J. Marsden, L. Sirovich, M. Golubitsky, and W. J. Äger, *Introduction to Applied Nonlinear Dynamical Systems and Chaos*. 1990.

[108] A. Fradkov, P. Guzenko, and A. Pavlov, “Adaptive Control of Recurrent ´ Map Linearization of Poincar E,” *Int. J.*, vol. 10, no. 3, pp. 621–637, 2000.

[109] H. Nijmeijer and A. van der Schaft, *Nonlinear Dynamical Control Systems*. New York, NY: Springer New York, 1990.

[110] S.-R. XIE, X. I. E. Shao-rong, L. U. O. Jun, R. A. O. Jin-jun, and G. Zhen-bang, “Computer Vision-based Navigation and Prede¯ned Track Following Control of a Small Robotic Airship,” *Acta Autom. Sin.*, vol. 33, no. 3, p. 0286, 2007.

[111] Y. Street and K. District, “Attitude Control for a Station Keeping Airship Using Feedback Linearization and Fuzzy Sliding Mode Control,” *Int. J. Innov. Comput. Inf. Control*, vol. 8, no. 12, pp. 8299–8310, 2012.

[112] S. Wen and Y. Yan, “Robust adaptive fuzzy control for a class of uncertain nonlinear systems,” *Math. Probl. Eng.*, 2015.

[113] Y. Yang, J. Wu, and W. Zheng, “Positioning Control for an Autonomous Airship,” *J. Aircr.*, vol. 53, no. 6, pp. 1638–1646, 2016.

[114] H. Lamb, “The inertia coefficients of an ellipsoid moving in fluid,” 1918.

[115] M. M. Munk, “Aerodynamic Forces on Airship Hulls,” Technical Report, National Advisory Committee for Aeronantics, Washington, DC, 1924.

[116] S. P. Jones and J. D. DELAURIER, “Aerodynamic Estimation Techniques for Aerostats

and Airships," *J. Aircr.*, vol. 20, no. 2, pp. 120–126, 1982.

[117] O. Daskiran, B. Huff, and A. Dogan, "Low Speed Airship Control using Reinforcement Learning and Expert Demonstrations," in *AIAA Atmospheric Flight Mechanics Conference*, 2017, pp. 1–35.

[118] R. Beard and T. Princeton, *Small Unmanned Aircraft: Theory and Practice*, vol. 36, no. 1. 2013.

[119] A. Colozza and J. L. Dolce, "High-Altitude, Long-Endurance Airships for Coastal Surveillance," *NASA Tech. Rep.*, 2005.

[120] D. Yu and X. Lu, "Configurations analysis for high-altitude/ long-endurance airships," *Aircr. Eng. Aerosp. Technol.*, vol. 82, no. 1, pp. 48–59, 2010.

[121] S. Lubkowski, B. Jones, E. Rojas, and D. Morris, "Trade-off analysis of regenerative power source for long duration loitering airship," *2010 IEEE Syst. Inf. Eng. Des. Symp. SIEDS10*, pp. 25–30, 2010.

[122] C.-M. Tseng, C.-K. Chau, K. Elbassioni, and M. Khonji, "Flight Tour Planning with Recharging Optimization for Battery-operated Autonomous Drones," pp. 1–10, Mar. 2017.

[123] A. Ceruti and P. Marzocca, "Conceptual approach to unconventional airship design and synthesis," *J. Aerosp. Eng.*, vol. 27, no. 6, 2014.

[124] P. Liu, G. Y. Fu, L. J. Zhu, and X. L. Wang, "Aerodynamic characteristics of airship Zhiyuan-1," *J. Shanghai Jiaotong Univ.*, vol. 18, no. 6, pp. 679–687, 2013.

[125] Q. B. Wang, J. A. Chen, G. Y. Fu, and D. P. Duan, "An approach for shape optimization of stratosphere airships based on multidisciplinary design optimization," *J. Zhejiang Univ. Sci. A*, vol. 10, no. 11, pp. 1609–1616, 2009.

[126] R. P. Bajpai, U. Chandrasekhar, and A. R. Arankalle, *Innovative Design and Development Practices in Aerospace and Automotive Engineering*. Singapore: Springer Singapore, 2016.

[127] L. Sangjong, H. Lee, W. Daeyeon, and B. Hyochoong, "Backstepping Approach of

Trajectory Tracking Control for the Mid-Altitude Unmanned Airship," in *AIAA Guidance, Navigation and Control Conference and Exhibit*, 2007, pp. 1–14.

[128] Y. Huang, D. M. Pool, O. Stroosma, and Q. P. Chu, "Incremental Nonlinear Dynamic Inversion Control for Hydraulic Hexapod Flight Simulator Motion Systems," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 4294–4299, 2017.

[129] J. R. Azinheira, A. Moutinho, and J. R. Carvalho, "Lateral Control of Airship with Uncertain Dynamics using Incremental Nonlinear Dynamics Inversion," *IFAC-PapersOnLine*, vol. 48, no. 19, pp. 69–74, 2015.

[130] C. Cakiroglu, E. J. Van Kampent, and Q. Chu, "Robust incremental nonlinear dynamic inversion control using angular accelerometer feedback," *AIAA Guid. Navig. Control Conf. 2018*, no. 210039, 2018.

[131] D. Gobiha and N. K. Sinha, "Autonomous maneuvering of a stratospheric airship," in *2018 Indian Control Conference (ICC)*, 2018, pp. 318–323.

[132] M. E. Mkiramweni, C. Yang, J. Li, and Z. Han, "Game-Theoretic Approaches for Wireless Communications with Unmanned Aerial Vehicles," *IEEE Wirel. Commun.*, vol. 25, no. 6, pp. 104–112, 2018.

[133] Y. Zhao, W. Garrard, and J. Mueller, "Benefits of Trajectory Optimization in Airship Flights," in *AIAA 3rd "Unmanned Unlimited" Technical Conference, Workshop and Exhibit*, 2004.

[134] D. W. Mellinger, "Trajectory Generation and Control for Quadrotors barw.," PhD thesis, University of Pennsylvani, 2012.

[135] "Hobbyking." [Online]. Available: https://hobbyking.com/en_us/zippy-compact-8000mah-3s1p-30c-lipo-pack-with-xt90.html. [Accessed: 30-Aug-2020].

[136] "Buyapi." [Online]. Available: https://www.buyapi.ca/product/raspberry-pi-3-model-b-plus/. [Accessed: 30-May-2020].

[137] "Canadarobotix." [Online]. Available: https://www.canadarobotix.com/products/66. [Accessed: 28-Dec-2020].

[138] “Pololu.” [Online]. Available: https://www.pololu.com/product/1596. [Accessed: 05-Jul-2020].

[139] “Mikroe.” [Online]. Available: https://www.mikroe.com/counter-click. [Accessed: 06-Jun-2020].

[140] “Pololu.” [Online]. Available: https://www.pololu.com/product/4760. [Accessed: 06-Jun-2020].

[141] “Robotshop.” [Online]. Available: https://www.robotshop.com/ca/en/hs-77bb-low-profile-servo.html. [Accessed: 07-May-2020].

[142] “HobbyKing.” [Online]. Available: https://hobbyking.com/en_us/hobbykingr-tm-brushless-car-esc-45a-w-reverse.html?___store=en_us. [Accessed: 12-Dec-2020].

[143] “Zonehobbies.” [Online]. Available: https://www.zonehobbies.net/products/rimfire-400-28-30-950-outrunner-brushless-motor. [Accessed: 06-Jun-2020].

[144] “Robotshop.” [Online]. Available: https://www.robotshop.com/ca/en/mpu-6050-6-dof-gyro-accelerometer-imu.html. [Accessed: 20-Dec-2020].

[145] “Newbecca.” [Online]. Available: https://www.newbecca.com/product/37406185535. [Accessed: 28-Dec-2020].

[146] “Spektrumrc.” [Online]. Available: https://www.spektrumrc.com/Products/Default.aspx?ProdId=SPMAR610. [Accessed: 12-Dec-2020].

[147] R. Coelho, “Quadrotor Control using Incremental Nonlinear Dynamics Inversion,” Master’s thesis, TECNICO LISBOA, 2017.

[148] T. Lutz, P. Funk, A. Jakobi, and S. Wagner, “Summary of aerodynamic studies on the lotte airship,” in *4th International Airship Convention and Exhibition*, 2002.

[149] G. C. Avenant, “Autonomous Flight Control System for an Airship,” Master’s thesis, Stellenbosch University, 2010.

[150] S. H. Miller, R. Fesen, and and et Al, “Airships: A New Horizon for Science,” 2014.

[151] “Defense update,” 2016. [Online]. Available: http://defense-update.com/20120529_blue-devil-2-airship-deflated.html. [Accessed: 08-Mar-2016].

[152] Anthony Sebastian, “ars technica uk,” 2017. [Online]. Available: https://arstechnica.co.uk/information-technology/2017/05/airlander-10-largest-aircraft-airship-details-test-flight/. [Accessed: 23-Aug-2017].

[153] H. & B. Beds, “BBC News,” 2017. [Online]. Available: http://www.bbc.com/news/uk-england-beds-bucks-herts-37111527. [Accessed: 23-Aug-2017].

[154] L. Tannian, “The Hindenburg Disaster: The Final Flight.” [Online]. Available: http://livebooklet.com/publish.php?wpKey=zBtn2I0yYv40NqmhX4BKIL#page=6. [Accessed: 28-Oct-2016].

[155] “Planespotters.net.” [Online]. Available: https://www.planespotters.net/photo/138828/dWlzzfWdeutscheWzeppelinWreedereiWzeppelinWnt07W100. [Accessed: 10-Aug-2017].

[156] S. F. Hoerner, *Fluid-Dynamic Drag*. Bakersfield, USA: HOERNER FLUID DYNAMICS, 1965.

[157] “Elevation.maplogs.” [Online]. Available: https://elevation.maplogs.com/poi/ottawa_on_canada.550.html. [Accessed: 10-Aug-2018].

[158] K. Instruments, “Battery Discharge / Charge Cycling Using Keithley Series 2400 SourceMeter ® Instruments,” 2013.

[159] J. Schoemann and M. Hornung, “Modeling of Hybrid Electric Propulsion Systems for Small Unmanned Aerial Vehicles,” in *12th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference and 14th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, 2012, pp. 1–18.

[160] S. A. Erturk and A. Dogan, “Dynamic Simulation and Control of Mass-Actuated Airplane,” in *AIAA Atmospheric Flight Mechanics Conference*, 2017, pp. 1–24.

[161] M. Drela, “Second-Order DC Electric Motor Model Fundamental relations,” *MIT Aero & Astro*, pp. 1–5, 2006.

[162] O. Gur and A. Rosen, “Optimizing Electric Propulsion Systems for UAVs,” in *12th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, 2008.

[163] D. Lundström, K. Amadori, and P. Krus, “Validation of Models for Small Scale Electric Propulsion Systems,” in *48th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition*, 2010.

[164] “Electricrcaircraftguy.” [Online]. Available: https://www.electricrcaircraftguy.com/2013/09/propeller-static-dynamic-thrust-equation.html. [Accessed: 12-Sep-2019].

APPENDICES

Appendix A Historical Background

Man's dream of flight is so ancient that at any ancient civilization's art you will likely find images of winged humanoids. Archeologists have discovered such sights in prehistoric caves dating back 4,300 years. This shows that man for a long time has attempted to fly with wings. It was not until the Montgolfier brothers from Annona, France on June 5, 1783 conducted the first unmanned hot

air balloon flight. The envelope was made of linen and paper. The Montgolfier balloon flew 1.2 mile from the starting point.

In 1900, the beginning of the golden age of airships was marked after Count von Zeppelin, a German general, an aircraft manufacturer and designer of rigid airships – Zeppelins, lunched the Luftschiff Zeppelin LZ1. Von Zeppelin began his experiments with rigid airship designs in the 1890s leading to some patents and the LZ1 (1900) and the LZ2 (1906). The Zeppelin airships became the most famous aircraft of that time and some of these airships were used for surveillance, cargo transport and bombing during World War I by the German Army. The airship became the first aircraft of choice for long distance flights, when in the 1920s and 1930s the Zeppelins, operating for the DELAGE airline, provided safe and trans-Atlantic crossings between Europe and America.

The LZ 129 Hindenburg is the largest flying machine was ever built by the Zeppelin Company. After several successful flights crossing of the North Atlantic, the Hindenburg disaster took place on Thursday, May 6, 1937, as the German passenger airship LZ 129 Hindenburg caught fire and was destroyed during its attempt to dock with its mooring mast at the Lakehurst Naval Air Station, New Jersey.

Since the disaster of the Hindenburg, Hydrogen has never been used again to lift passenger airships and airship activity has been confined to the non-rigid type of craft.

Airships were used with reasonable success prior to during WWII. During the inter-war period, countries such as the United States and Britain experienced costly airship crashed, and by the late 30s, Germany was the only remaining country manufacturing them. By WWII they had been deemed near obsolete by the majority of countries around the world.

Although airships are no longer used for passenger transportation, they continued to be used for other purposes, such as advertising and sightseeing.

Since the 1980's, airships slowly began to gain popularity again. This is due to the fact that airships require less energy than conventional aircraft to stay afloat, since they use aerostatic forces instead of aerodynamic forces for vertical lift. This makes it suitable for a wide variety of modern applications, including advertising, surveillance and environmental monitoring.

Some recent airship prototypes including theses previously discussed.

The Airship "Lotte"

Scientific investigation was done at University of Stuttgart on the remotely controlled airship Lotte and this airship was chosen as reference configuration to perform scientific investigations in the fields of aerodynamics, flight mechanics and control, aeroelasticity, structural design and solar propulsion systems. This airship has a length of 16 m, a volume of 109 m^3 and a maximum payload of 15 kg. The modeling of the flight mechanical model included system identification techniques to obtain the flight dynamical characteristics of the airship. These methods ware assumed to be adequate, since this model corresponds well to actual flight data [148]

YEZ-2A Airships

Gomes [21] preformed experimental studies on the YEZ-2A airship at the Cranfield Institute of Technology. The YEZ-2A airship has a length of 129.5 m. The aerodynamic model was obtained from 8ft x 6ft wind-tunnel tests of 1:75 scale model of the college of Aeronautics. The aerodynamic model of the YEZ-2A used as the database for the flight dynamic simulation. The study is considered a good overview of the airship's behaviour in simulation.

AURORA

The Autonomous Unmanned Remote Monitoring Robotic Airship (AURORA) focuses on the development of the sensing, control, navigation, and inference technologies required for semi-autonomous operation of unmanned robotic airships for aerial inspection [45]. The AURORA is a non-rigid, 9 m long, 2.25 m in diameter, 24 m^3 airship as in Figure . The AS800 aerodynamic model was adapted from the wind tunnel database acquired to model Westinghouse's YEZ-2A airship [148]. Since it featured the same length/diameter ratio (4:1). This article reported a successful airship autonomous flight achieved through a set of pre-defined points. This flight was achieved with a PI control-based guidance strategy for the trajectory path following. The horizontal trajectory was controlled automatically by the onboard system, while altitude was controlled manually by the ground pilot [40].

Figure A.1 The AURORA I AS800 [149]

NASA/JPL Titan Aerobot Project

The arrival of the Huygens probe at Saturn's moon Titan on January 14, 2005, motivated the authors to a follow-on mission that would use a substantially autonomous aerobot to explore

Titan's surface. They discussed steps towards the development of an autonomy architecture, and concentrate on the autonomous flight control subsystem. They developed a new nonlinear robotic airship model intended for control system design and evaluation. The model gathered much of the previous airship modelling results available in the literature, and added new elements to extend the model's range of applicability. The aerodynamic model developed had the ability to simulate all four primary modes of flight (launch, cruise, hover, and landing). The prototype aerobot testbed developed at Jet Propulsion Laboratory (JPL) at the University of California was based on an Airspeed Airship AS-800B. The airship Figure had a length of 11 m, a diameter of 2.5 m and total volume of 34 m^3 and a static lift payload of approximately 12 kg [53].

Figure A.2 Autonomous flight of the JPL aerobot [53]

BD2 M1400

One of the current operational airship, the Blue Devil 2 airship, built by MAV6 for the US Air Force as in Figure , is a conventional non-rigid and 123 m long designed to fly at 20 kft for 4 to 5 days with a 2,500 lb Intelligence, Surveillance, and Reconnaissance (ISR) payload including onboard processing that makes it an aerial data fusion node. Originally scheduled for first flight in the fall of 2011, the program was cancelled in June 2012 due to unacceptable performance and

recurring failures to meet minimum operating standards. To date, the airship has not achieved the first test-flight status [150].

Figure A.3 A front view of the M1400 airship [151]

Airlander 10

Airlander 10, Figure , is the world's longest aircraft built by Hybrid Air Vehicles (HAV) at Cardington Airfield in Bedfordshire and was officially named Martha Gwyn by the duke of Kent. The Airlander 10 is a non-rigid 92-metre (302ft) blimp filled with 38,000 cubic metres of helium. It has four propellers, two at the back, one on the front left and one on the front right, that provide vectored thrust from four V8 turbo-diesel engines. The Airlander 10 has been damaged after nosediving on landing during its second test flight from Cardington Airfield in Bedfordshire on August 24, 2016. Figure shows the moment of the crash [152].

Figure A.4 the world's longest aircraft, the Airlander 10 [152]

On May 11, 2017, Airlander 10 completed its first flight successfully after been repaired since the crash. The airship flew for almost three hours and successfully landed. The purpose of building the Airlander 10 is to fulfil a wide range of communication, cargo carrying and survey roles in both the military and commercial sectors all with a significantly lower carbon footprint than other forms of air transport.

Figure A.5 the Airlander 10 damaged on landing [153]

Airship architectures

Airships fall into three main structural categories, namely, rigid airships, semi-rigid airships and non-rigid airships. These types are defined by the shape of the envelope support structure.

Rigid Airships

Rigid airships are airships that have an internal framework that sustains their shape. Unlike pressure airships, rigid airships usually tend to have gas cells to provide lift instead of a larger gasbag and the envelope is a separate cover that goes over the outside of the framework as shown in Figure . Gas cells of rigid airships increases safety and avoids sudden loss of substantial lift during an emergency. This compartmentalization of the flammable gas will help prevent the fiery catastrophies of the past. It also facilitates lift adjustment due to altitude or temperature change that can be accomplished by expansion and contraction of individual gas cells [29]. Typically, rigid airships take off with the gasbags inflated to about 95% of their total volume and expand as altitude increases. Rigid airships have a capability of being built much larger than non-rigid and semi-rigid airships because there is no chance of kinking in the hull due to moments and aerodynamic loads. Rigid airships can reach an altitude more than 74000 feet depending on the design, weight of the structure and its cargo [29].

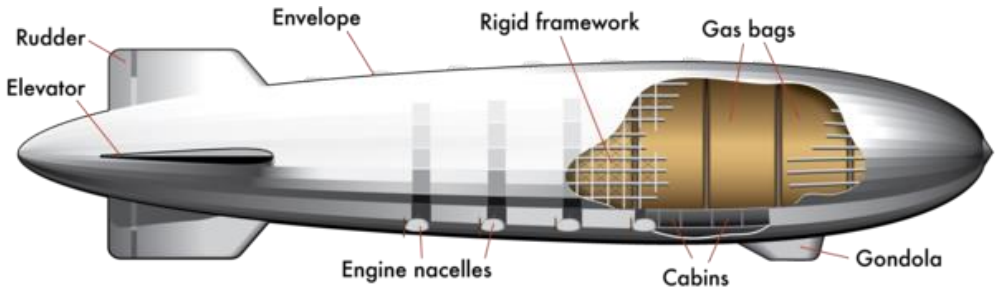


Figure A.6 a typical rigid airship [154]

Semi-rigid Airships

Semi-rigid airships are similar to non-rigid airships in their structure with an addition of a rigid keel or a truss that extends from the nose to the tail along the bottom surface of the airship. Similar to blimps, semi-rigid airships maintain their aerodynamic shape by lifting gas overpressure in the envelope. Ballonets can be placed inside the envelope to accommodate the altitude and

temperature effects. In semi-rigid airships, the keel acts like a backbone to the airship distributing the weight and providing an attachment for gondola, fins and propellers. The keel offers structural integrity during flight maneuvering [29]. Keels can be partially flexible or composed of joined sections and can be placed inside or outside the main envelope. Although semi-rigid airships weight significantly more than non-rigid airships, they are structurally more stable. A recent example of a semi-rigid airship is the Zeppelin NTW07 shown in Figure . The structure of the Zeppelin consists of three longitudinal aluminum beams divided into sections of 6 m length, and of triangular spar elements.

Figure A.7 Good Year Zeppelin NTW07 [155]

Non-Rigid Airships

Non-rigid airships, also known as blimps, are the simplest form of lighter than air (LTA) vehicles. They maintain their shape and structure by the internal pressure applied by its lifting gas. Because they have no heavy support structure, non-rigid airships have the greatest lift to weight ratio. They are mainly composed of an envelope, which contains the lifting gas and the ballonets, and a gondola. They have no rigid structure or a keel however, they lose their shape when deflated. Generally, in a non-rigid airship, the only solid parts are the gondola, tail, fins, and propellers. The thrusters are usually installed on the gondola and sometimes another one will be installed on the downward vertical fin to provide additional yaw control. Non-rigid airships are distinguished by

their lightweight and construction simplicity. Airships can also use heated gas instead of a light gas as their lifting medium and are called hot-air airships [28]. These may require additional components to accommodate to the high internal pressures induced. Figure A. shows a typical non-rigid airship

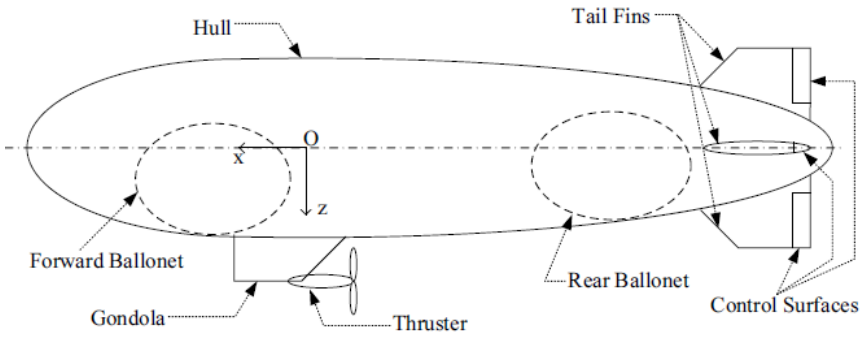


Figure A.8 a typical non-rigid airship [24]

Appendix **B Physical properties**

Table B.1 Physical properties of Model 1

Symbol	**Description**	**Value**	**Units**
m	Airship mass	1.66	kg
V	Airship volume	1.422	m^3
D	Airship diameter	0.86	m
L	Airship length	3	m
$d_{g,z}$	Distance between CV to gondola CG along z	0.5275	m
$d_{p,y}$	Distance between propeller centers to the x - z plane along y	0.533	m
$d_{p,z}$	Distance between propeller centers to gondola CG along z	0.5275	m
I_x	Moment of inertia about OX	0.1	m^2kg
I_y	Moment of inertia about OY	0.3	m^2kg
I_z	Moment of inertia about OZ	0.3	m^2kg
k_1	Lamb's inertia ratio along OX	0.1069	
k_2	Lamb's inertia ratio along OY or OZ	0.8239	
$K`$	Lamb's inertia ratio about OY or OZ	0.5155	

Table B.2 Parameters used to calculate aerodynamic forces of Model 1

Symbol	Description	Value	Units
C_{Dho}	Hull zero-incidence drag coefficient [156]	0.024	-
C_{Dfo}	Fin zero-incidence drag coefficient [156]	0.003	-
C_{Dgo}	Gondola zero-incidence drag coefficient [51]	0.01	-
C_{Dch}	Hull cross-flow drag coefficient [156]	0.32	-
C_{Dcf}	Fin cross-flow drag coefficient [13]	2	-
C_{Dcg}	Gondola cross-flow drag coefficient [51]	.25	-
$\left(\frac{\delta C_L}{\delta \alpha}\right)_f$	Derivative of fin lift-coefficient with respect to angle of attack [13]	5.73	-
$\left(\frac{\delta C_L}{\delta \delta_{E,R}}\right)_f$	Derivative of fin lift-coefficient with respect to fin angle [13]	1.24	-
S_h	Hull reference area, $V^{2/3}$	1.257	m^2
S_f	Fin reference area [16] *Calculated*	0.8	m^2
S_g	Gondola reference area [16] *Calculated*	0.05	m^2
$d_{f,x1}$	Distance between CV to GC of fins along *x* *S.W*	1.191	*m*
$d_{f,z}$	Distance between CV to AC of fins along *z* *S.W*	0.4322	*m*
$d_{c,z}$	Distance between CV to AC of gondola along *z* *S.W*	0.8239	*m*
η_f	Fin efficiency factor [116]	0.5155	-
η_h	Hull efficiency factor [116]	1.2	-
I_1	Hull integrals [51]	0	m^2kg
I_3		-1.162	m^2kg
J_1		1.919	m^2kg

J_3		0.7301	m^2kg

Table B.3 Physical properties of Model 2

Symbol	Description	Value	Units
m	Airship mass	4	kg
V	Airship volume	6.23	m^3
D	Airship diameter	1.827	m
L	Airship length	4.1	m
$d_{g,z}$	Distance between CV to gondola CG along z	0.95	m
$d_{p,y}$	Distance between propeller centers to the x - z plane along y	0.823	m
$d_{p,z}$	Distance between propeller centers to gondola CG along z	0.29	m
I_x	Moment of inertia about OX	0.54	m^2kg
I_y	Moment of inertia about OY	1.3	m^2kg
I_z	Moment of inertia about OZ	1.3	m^2kg
k_1	Lamb's inertia ratio along OX	0.2	
k_2	Lamb's inertia ratio along OY or OZ	0.75	
$K`$	Lamb's inertia ratio about OY or OZ	0.3	

Table B.4 Parameters used to calculate aerodynamic forces of Model 2

Symbol	Description	Value	Units
C_{Dho}	Hull zero-incidence drag coefficient [156]	0.024	-
C_{Dfo}	Fin zero-incidence drag coefficient [156]	0.003	-
C_{Dgo}	Gondola zero-incidence drag coefficient [51]	0.01	-
C_{Dch}	Hull cross-flow drag coefficient [156]	0.32	-
C_{Dcf}	Fin cross-flow drag coefficient [13]	2	-
C_{Dcg}	Gondola cross-flow drag coefficient [51]	.25	-
$\left(\frac{\delta C_L}{\delta \alpha}\right)_f$	Derivative of fin lift-coefficient with respect to angle of attack [13]	5.73	-
$\left(\frac{\delta C_L}{\delta \delta_{E,R}}\right)_f$	Derivative of fin lift-coefficient with respect to fin angle [13]	1.24	-
S_h	Hull reference area, $V^{2/3}$	3.386	m^2
S_f	Fin reference area [16] *Calculated*	0.2	m^2
S_g	Gondola reference area [16] *Calculated*	0.01	m^2
$d_{f,x1}$	Distance between CV to GC of fins along *x* *S.W*	1.723	*m*
$d_{f,z}$	Distance between CV to AC of fins along *z* *S.W*	0.43	*m*
η_f	Fin efficiency factor [116]	0.5	-
η_h	Hull efficiency factor [116]	1.2	-
I_1	Hull integrals [51]	0	m^2kg
I_3		-2.382	m^2kg
J_1		1.995	m^2kg
J_3		0.6104	m^2kg

Appendix **C Simulation results**

Position tracking errors of the simulation results in Chapter 5

Tracking error of backstepping controller Model 1

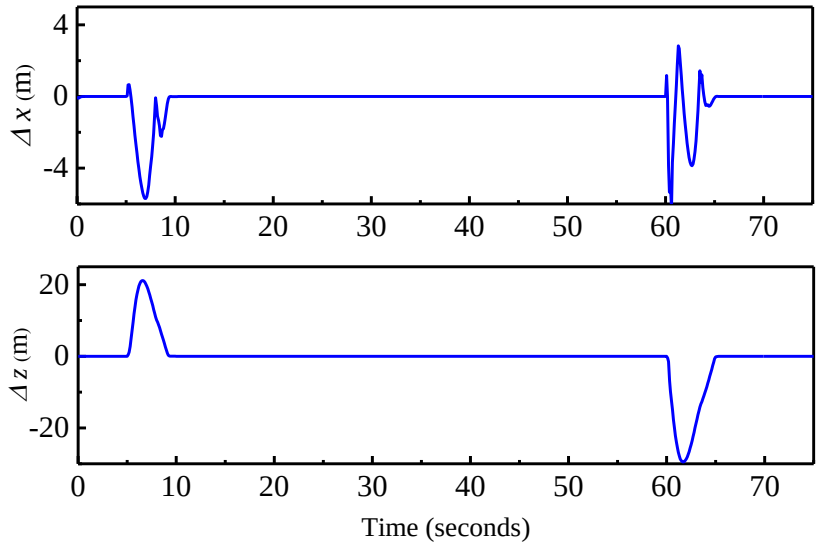


Figure B.1 Tracking error no wind

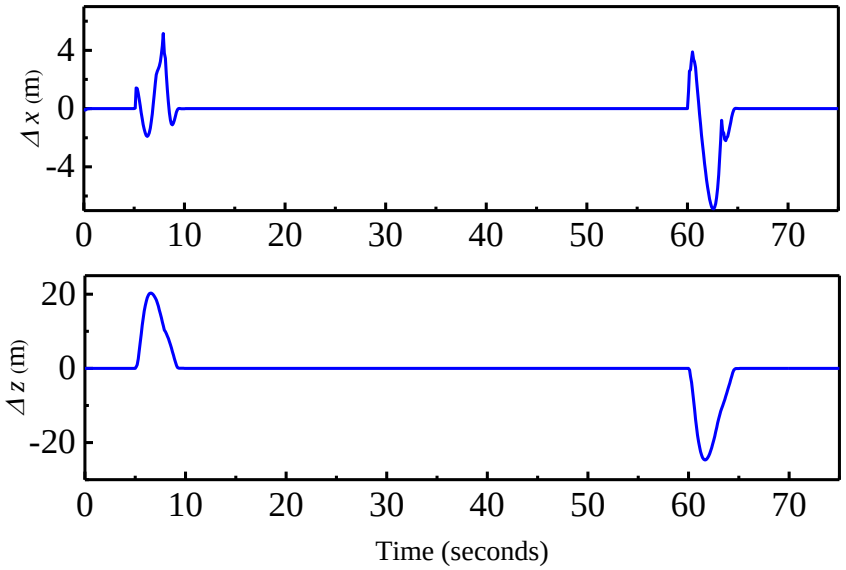


Figure B.2 Tracking error with wind disturbances

Tracking error of INDI controller Model 1

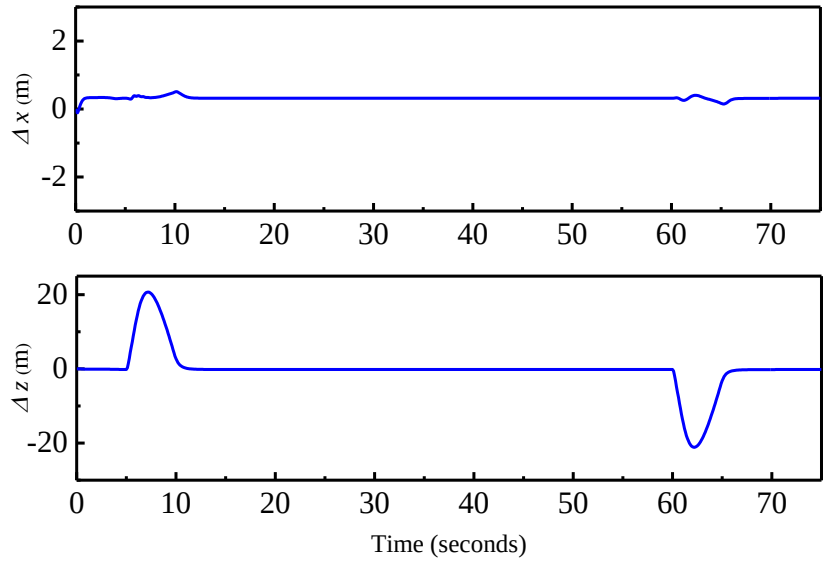


Figure B.3 Tracking error no wind

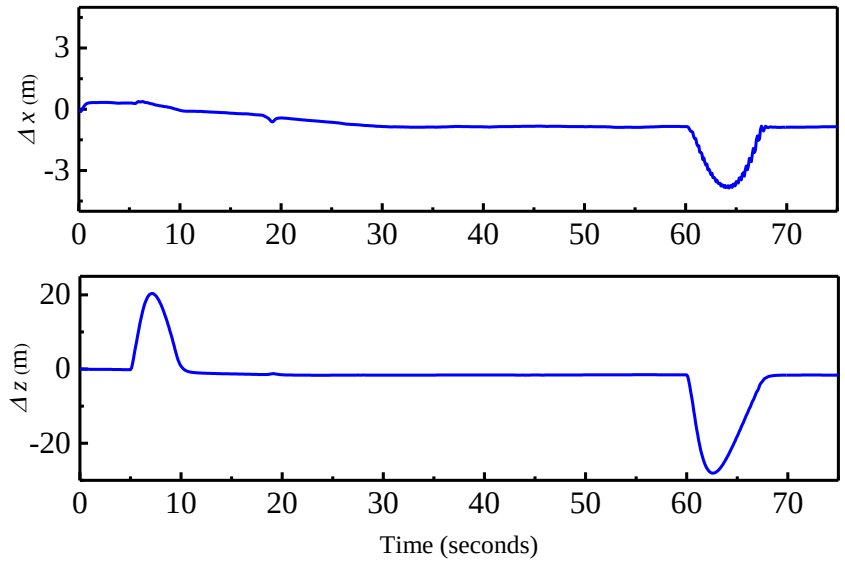


Figure B.4 Tracking error with wind disturbances

Tracking error of backstepping controller Model 2

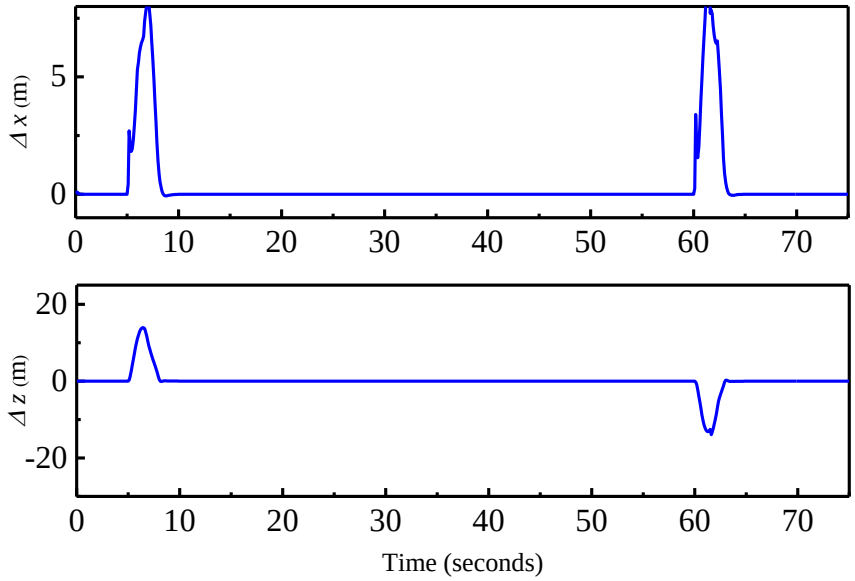


Figure B.5 Tracking error no wind

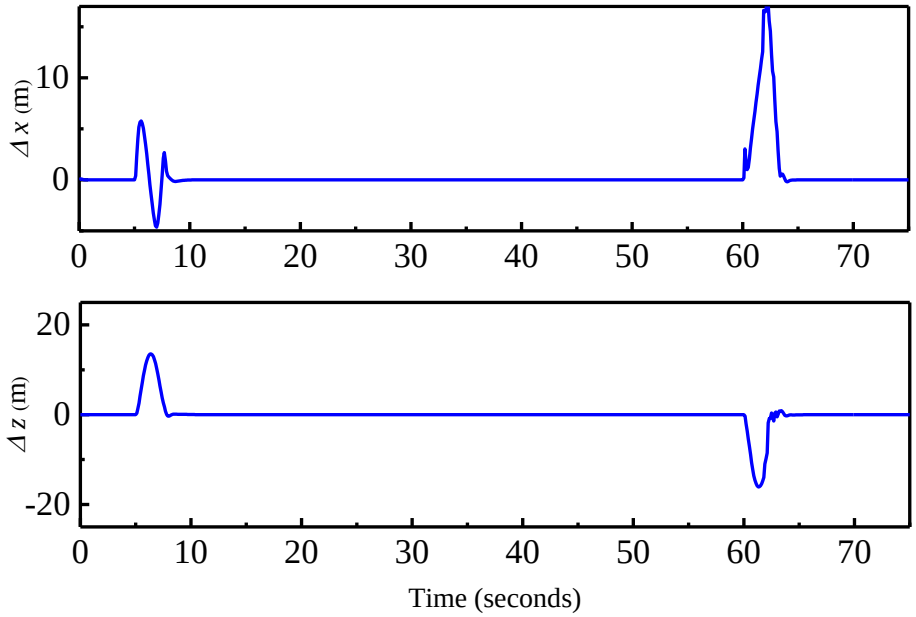


Figure B.6 Tracking error with wind condition

Tracking error of INDI controller Model 2

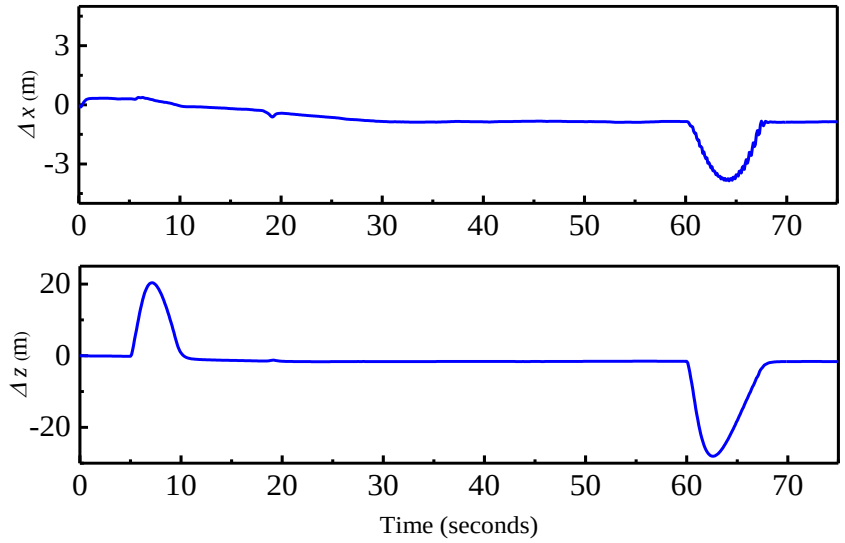


Figure B.7 Tracking error with no wind disturbances

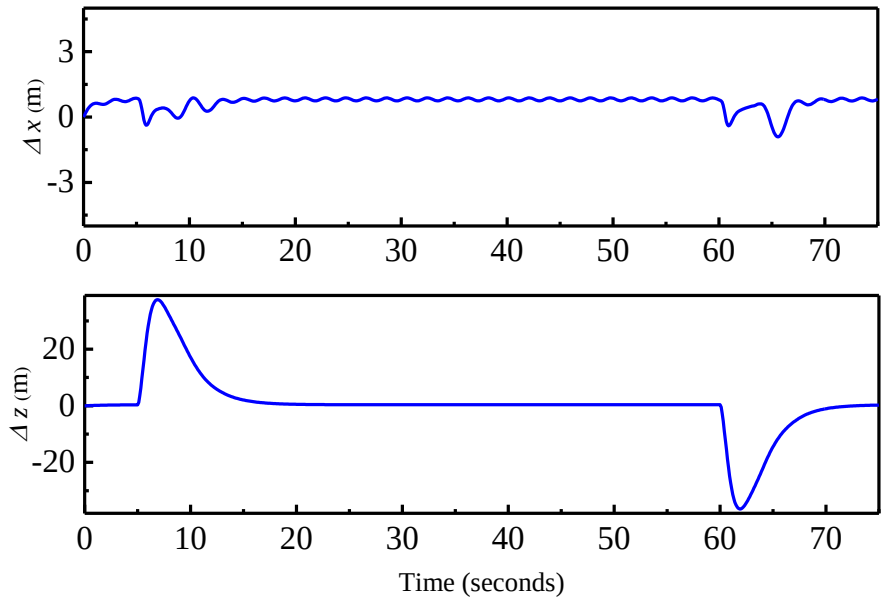


Figure B.8 Tracking error under wind disturbances

Some results from the Simulation used for calculating the energy, the gondola's position and the corresponding pitch angle results during the ascent and descent modes.

During the ascent mode

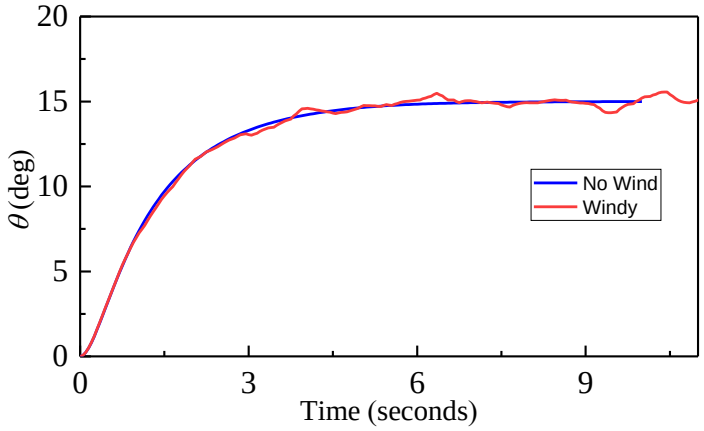


Figure B.9 The pitch angle by using backstepping during the ascent

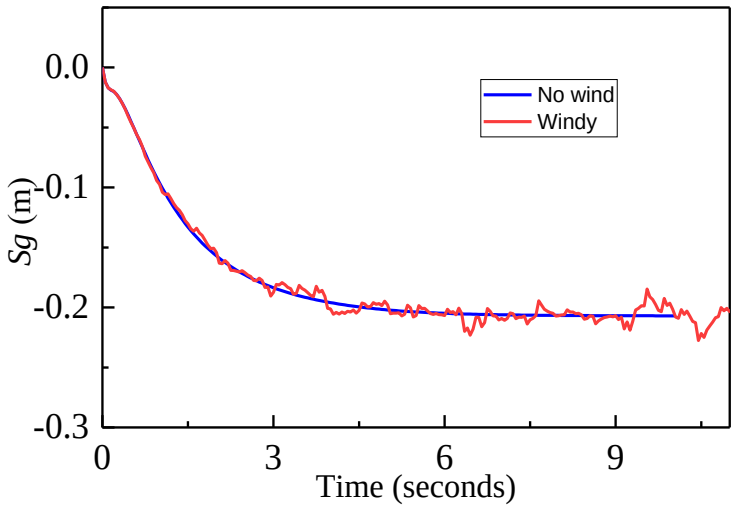


Figure B.10 backstepping control response

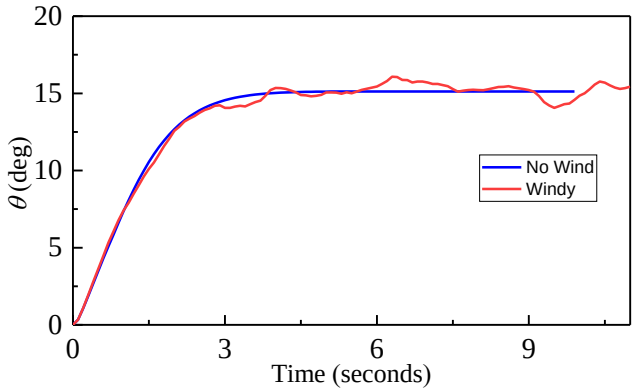


Figure B.11 The pitch angle by using INDI during the ascent

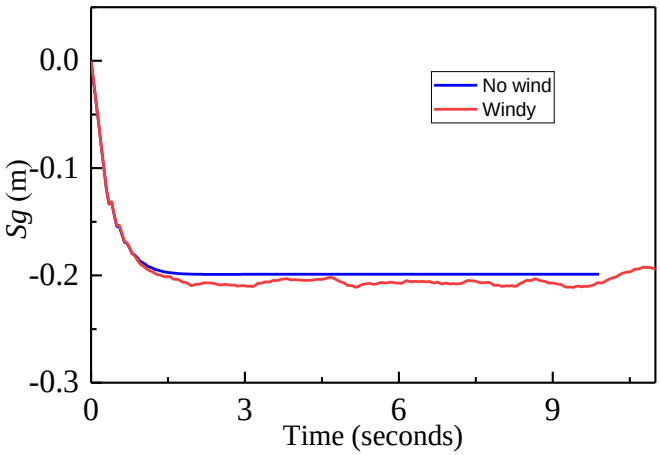


Figure B.12 INDI control response

2 During the descent mode

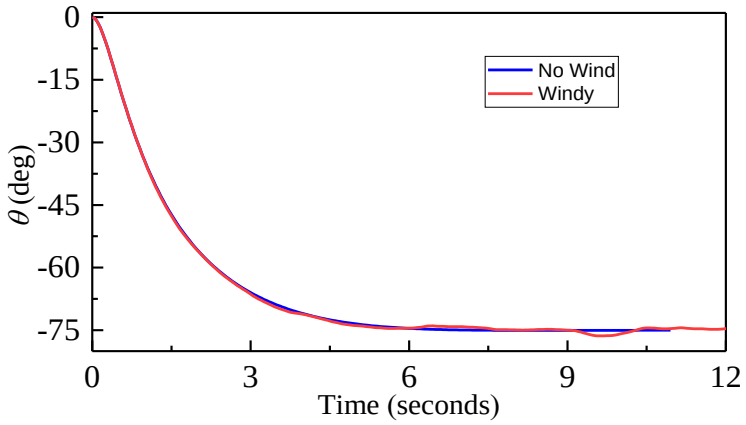


Figure B.13 The pitch angle by using backstepping during the descent

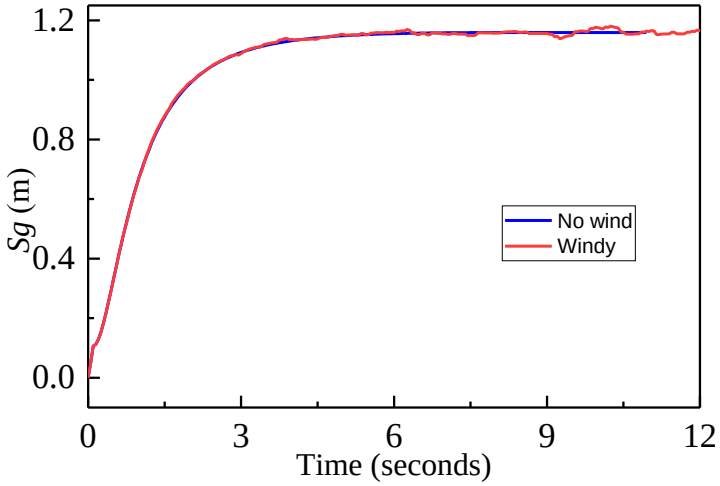


Figure B.14 backstepping control response

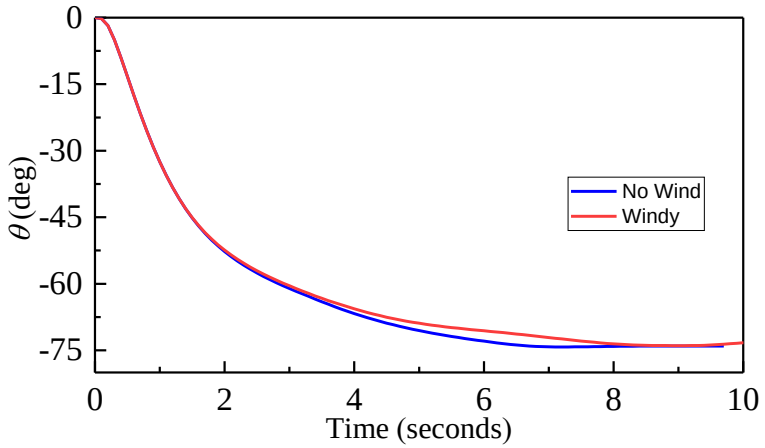


Figure B.15 The pitch angle by using INDI during the descent

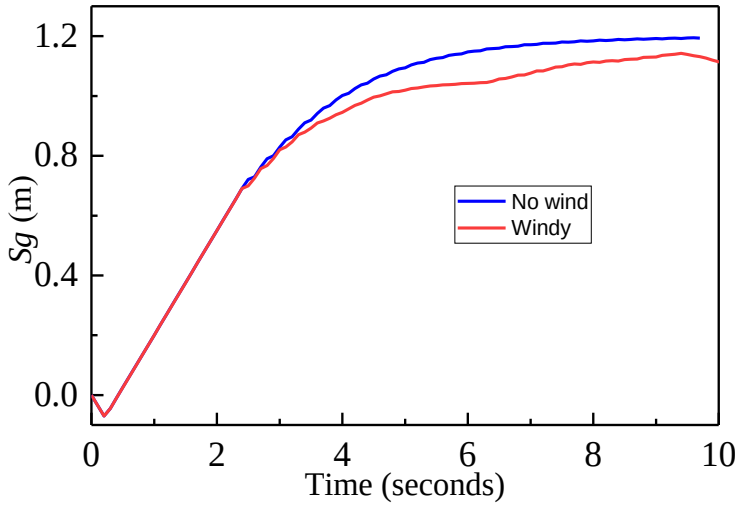


Figure B.16 INDI control response

Appendix C Experimental plate form

Battery (1)

The battery is the only source of power for all electronics on-board including the propulsion system in this study. Two 11.1 V ZIPPY Compact 3-cell battery are used, each with a capacity of 8000mAh and weight of 565 g. Two batteries are used to provides enough power to be able to sustain the entire flight mission. Figure C.1 shows the zippy battery used.

Figure C.1 Zippy compact battery

Raspberry Pi board (2)

Raspberry Pi is a small sized minicomputer that runs Linux and it is shown in Figure C.2. It can plug to monitor, keyboard and mouse. This inexpensive board has several usages like education, home automation, industrial automation and commercial product. Raspberry Pi is used in this study as a platform for implementation because it is an affordable and lightweight platform with extremely low complexity for design. In addition, it is documented extensively online and has a vibrant open source community. In this study the Raspberry Pi 3 Model B+ is used and it has the following specifications [136]:

- 1.4GHz 64-bit quad-core ARM Cortex-A53 CPU (BCM2837)
- 1GB RAM (LPDDR2 SDRAM)

- On-board wireless LAN - dual-band 802.11 b/g/n/ac (CYW43455)
- On-board Bluetooth 4.1 HS low-energy (BLE) (CYW43455)
- 4 x USB 2.0 ports
- 300Mbit/s ethernet
- 40 GPIO pins
- Full size HDMI 1.3a port
- Combined 3.5mm analog audio and composite video jack
- Camera interface (CSI)
- Display interface (DSI)
- microSD slot
- VideoCore IV multimedia/3D graphics core @ 400MHz/300MHz

The Raspberry Pi receives the data from all sensors and sends that data over Wi-Fi to the ground station. The Pi receives signals from the user indicating the desired trajectory and takes these commands to adjust the airship actuators such that the airship achieves the user's desired trajectory and stability. All data logging from sensors and signals into MAT files saved on the Raspberry Pi.

Figure C.2 Raspberry Pi 3 Model B+

TReX Jr Dual Motor Controller (3)

The TReX Jr Dual-Motor Controller is a versatile DC motor controller designed to seamlessly blend autonomous and human control of small robots and it is shown in . The TReX Jr can control two bidirectional and one unidirectional motor via three independent control interfaces. The TReX Jr operating voltage range is between 5V and 24V and in this current set up it is powered directly by the batteries 12V. TReX Jr receives the Pulse Width Modulation (PWM) signal from the Pi and distributes power to two Brushed DC motors according to the received signal.

Figure C.3 The TReX Jr Dual-Motor Controller

Brushed DC gearmotors 1000:1 (4)

The Brushed DC gearmotor is a miniature high-power, 12V with a 986.41:1 metal gear ratio, making it a great choice for applications requiring fine control at very low speeds and high torque. Two motors are used to drive the gondola along the keel by custom wheels of 18mm in diameter, mounted on the output shaft of the motors, one on each side of the keel as shown in Figure 8.3. The motors receive power from the TReX Jr controller which adjusts output frequency to allow the motor speed to be precisely controlled. The Brushed DC motor is not expensive and commercially available, and it is shown in Figure C.4.

Figure C.4 Brushed DC with extended motor shaft

Counter click (5)

Counter click carries an LS7366R quadrature counter Figure C.5. The counter click communicates with the Raspberry Pi through SPI interface (CSK, MISO, MOSI). The board can use either a 3.3V or a 5V power supply. In the current platform, it receives 5V from the Pi. The counter chip is used because using only encoder to count the steps introduces more that 10 seconds delay and misses counting some of the steps. In addition, the Raspberry Pi keeps crashing almost immediately when encoder recorded movie. Using a counter chip to count the encoder and send the data to the pi helps prevent crashing the pi and obtains higher accuracy reading.

Figure C.5 Counter click LS7366R

Magnetic Encoder (6)

The encoder is added to the brushed DC gearmotor with the extended back shaft and it is illustrated in Figure C.6. The encoder board senses the rotation of the magnetic disc and provides a resolution of 12 counts per revolution of the motor shaft when counting both edges of both channels. The gear ratio is multiplied by 12 to compute the counts per revolution of the gearbox output shaft. The encoder operates from 2.7 V to 18 V and provide digital outputs that can be connected directly to a pi or other digital circuit. One encoder is used in this set up and it receives 5V from the counter click.

Figure C.6 Magnetic Encoder

HS-77BB Servo motor (7)

Vectoring of thrust is provided by two HS-77BB Servo motors illustrated in Figure C.7, one on each side of the gondola as in **Figure 3.7**. The servo motors were limited to a tilting angle from -7^{o} to +100^{o} to respect the design limitation. The servo motor receives a PWM input from the Raspberry Pi to set the desired angle of the output shaft. the desired angle of the servo's output shaft is in degrees measured from the horizontal axis in body frame. The max voltage required by the servo motor is 6V and is rated to generate a maximum torque of 5.5 kg-cm with no load and a maximum speed of 60 degrees per 0.18 sec.

Figure C.7 HS-77BB Servo motor

Standard 45A car ESC (8)

Standard 45A car electronic speed controller (ESC) with forward reverse and a good range of features such as timing adjust and breaking. An ESC is required to control the motor when using a brushless motor. Two Standard 45A car ESCs are used in this platform to drive two brushless motors to generate the required thrust to drive the airship. Electronic speed controller (ESC) converts input PWM signals from the Raspberry Pi and drives the brushless DC motors with the correct power. The ESC receives 12V from the battery and receives a 50Hz pulse width modulated (PWM) control signal from the Pi. The ESC used in this airship is shown in Figure C.8.

Figure C.8 Standard 45A car ESC

Rimfire 400 Brushless motor (9)

Two Rimfire 400 brushless DC (BLDC) motors with two propellers installed on the shafts of the motors are used in this airship as the propulsion system, as shown in Figure C.10. The motors used is shown in Figure C.9 and have a weigh of 54g each and Input voltage of 7.4-11.1V. The propeller

thrust system produces -0.26 N < T < 2.6 N based on thruster static bench testing prototype which will be discussed in Appendix G. The ESC must first receive a PWM signal of width 0.075 ms from the Pi in order to be armed and be ready to control the motors. Stopping the beep sound is indicating that the motor is ready to run. Each motor's input is a PWM signal which corresponds to a value ranging from 0.05 for maximum reverse to 0.1 for maximum forward form the ESC.

Figure C.9 Rimfire 400 brushless DC

The APC 9 x 4.7 propellers (9 inches in diameter with pitch of 4.7 inches) are driven by the two brushless motors and propellers are shown in Figure C.10. this propeller can produce high thrust in any direction perpendicular to the rotating axis of the blade. The airship can move forward, backward, upward or downward according to the user to achieve the required trajectory. For a safe flight and to ensure that the operator can take full control of the airship, the drive system needs to be controlled by both mode, automatic control and Manual control, as discussed earlier.

Figure C.10 the propulsion system

Radio Frequency Module (11)

Two radio frequency modules are used to communicate wirelessly between the ground station and the airship. One is connected to the ground station (receiver) and the second one is connected to onboard system (transceiver). The transceiver is connected to the Pi through (Universal Asynchronous Receiver/Transmitter) UART. It is powered by 3.3V from the Pi and operated at a 57600 baud rate. Radio frequency module used in this study is Module RF 7020 v4.0 and it is shown in Figure C.11.

Figure C.11 Module RF 7020 v4.0

AR610 Aircraft Receiver (12)

The Aircraft Receiver is a six channels receiver mounted in the gondola used for manual control and it is shown in Figure C.12. The manual control is used for emergency control to override the control of the drive system. In this thesis, the manual control mode is utilized for altitude control and pitch control. The override control is engaged when the Remote Control (RC) turned ON and the disable command is activated from the ground station thus the operator can have a complete control of the drive system.

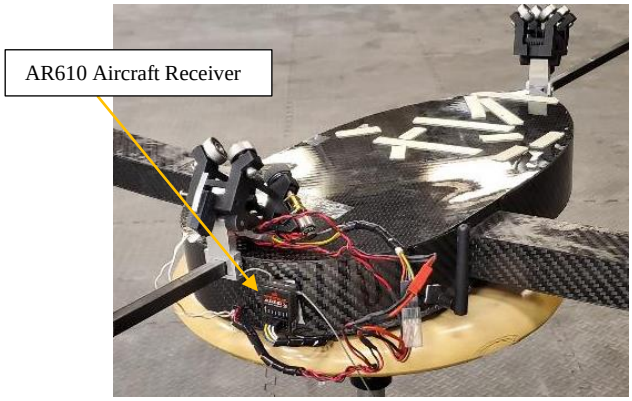


Figure C.12 AR610 Aircraft Receiver attached to the gondola

GPS (global positioning system)

The GPS is used to determine the current location (latitude and longitude) of the blimp. The Adafruit Ultimate GPS Breakout used in this study is shown in Figure C.13. This GPS was chosen because it is a high-quality GPS module that can track up to 22 satellites with 10 Hz on 66 channels and has an excellent high-sensitivity receiver and a built-in antenna. The GPS module attached directly to the Raspberry Pi using TTL adapter and communicates via a USB/UART interface. To improve the signal reception, a GPS antenna was attached to the GPS module. The GPS signal is very sensitive to the environment like indoor, or between very tall buildings and forest areas where the signals become weak or jammed. Therefore, **Kalman Filter** is used to estimate and update the airship states it in real time during the signal blockage or outage.

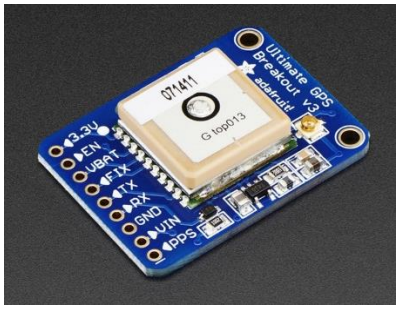


Figure C.13 Adafruit Ultimate GPS Breakout

Figure C.14 GPS Antenna

The goal of the Kalman filter is to take a probabilistic estimate of the state and update it in real time using two steps: prediction and correction (update). The position estimation is done using linear Kalman filter that can access GPS to estimate the airship's states. The position format is latitude, longitude in radians and altitude in meter.

Prediction equations:

$$X_k = X_{k+1} + K_k\,(Z_k - H_k\,X_{k+1}) \qquad \text{(C.1)}$$

$$P_k = (I - K_k\,H_k\,)\,P_{k+1} \qquad \text{(C.2)}$$

$$K_k = P_{k+1}\,H_k^T\,\text{inv}(H_k P_{k+1}\,H_k^T + R) \qquad \text{(C.3)}$$

Update equations:

$$X_{k+1} = \emptyset_k\,X_k \qquad \text{(C.4)}$$

$$P_{k+1} = \emptyset_k\,P_k\,\emptyset_k^T + Q \qquad \text{(C.5)}$$

where:

X_k *is* the state vector in t_k ,

P_k is the state covariance matrix,

Z_k is the measured vector,

K is the Kalman gain matrix which determines the weights of GPS information when status is updated,

$\emptyset_k$ is the transition matrix of X_k to X_{k+1},

$$\emptyset_k = \begin{bmatrix} \text{eye}(3,3) & \Delta t * \text{eye}(3,3) & \frac{1}{2} * \Delta t^2 * \text{eye}(3,3) \\ \text{zeros}(3,3) & \text{eye}(3,3) & \Delta t * \text{eye}(3,3) \\ \text{zeros}(3,3) & \text{zeros}(3,3) & \text{eye}(3,3) \end{bmatrix}$$

Δt is the sample time,

Q is the system noise covariance matrix, it was chosen small for better rejection of the noise and smoother estimates,

$$Q=0.2*[\text{ eye}(9,9)]$$

H is the ideal transition matrix of Z_k to X_k,

$$H= [\text{eye (3,3)} \quad \text{zeros (3,6)}]$$

R is the observation noise covariance matrix,

$$R= \text{eye (3,3)}$$

The goal is to merge the information from different sensors to produce a final estimate of some unknown state. Kalman filter is used to predict new state by using the inertial sensor measurements and the GPS data to correct that prediction of the airship's position at time *k.* The block diagram of Kalman filter is shown in Figure C.15.

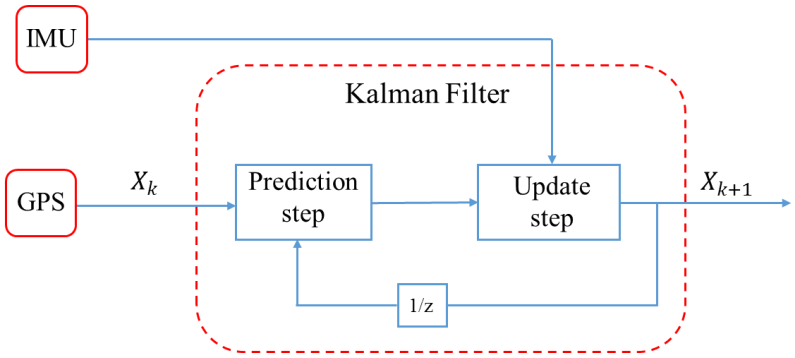


Figure C.15 block diagram of Kalman filter

Buoyancy loss test

The purpose of buoyancy testing is to maintain some controllability during the descent and avoid dangerous crashes. Also, it is useful for designing safety airship systems.

The first step in the test is to fill the blimp with helium until it is fully inflated. The weight of the gondola, as well as any additional mass, is then measured and attached to the airship. The test begins when the airship is neutrally buoyant. The weight of the gondola with all the components attached is 3.4 Kg and the total weight (gondola and added mass) at the beginning of the test was 5.3 Kg. The test preformed at the room temperature (21 ^{o}C) with air pressure varying from (101.9 kpa to 102.2 kpa) throughout the duration of test, and at the altitude of 70 m above the sea level [157]. The test started at 10:15 am until 6:45 pm. Every 30 minutes some ballast is dropped until the airship buoyancy is levelled again. The amount of ballast dropped at various time is illustrated in Figure C.16. The total ballast dropped at the end of the test time was 650 g which is about 12% of the total weight.

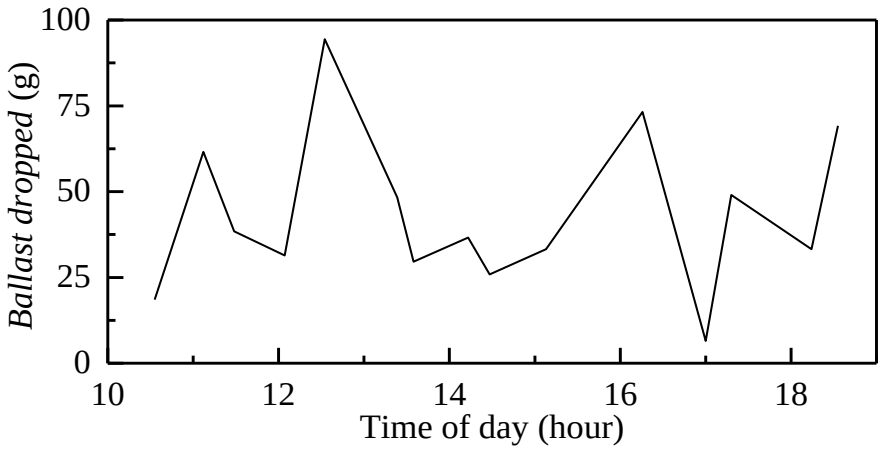


Figure C.16 Ballast dropped during the daytime

Appendix D Experimental results

The results of backstepping controller.

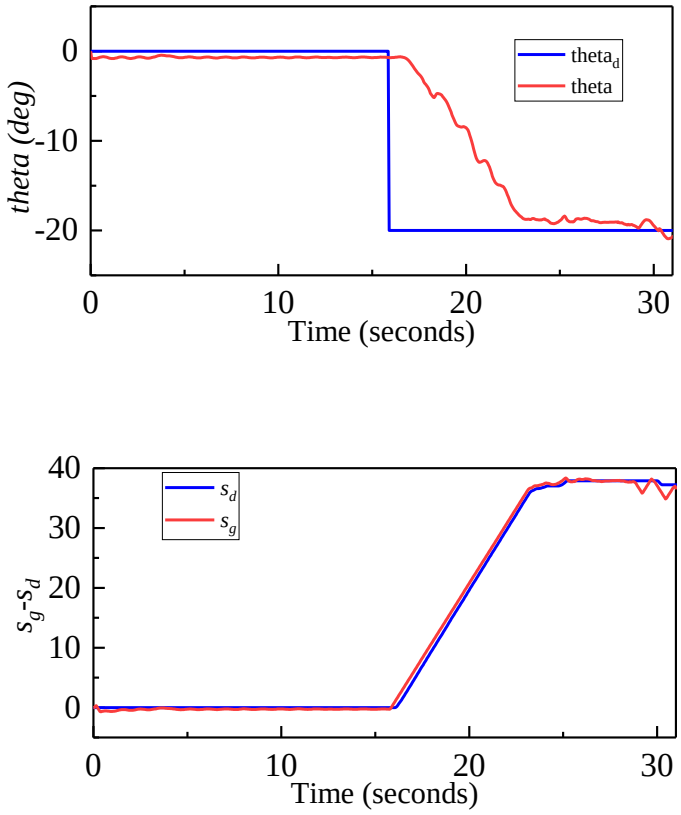


Figure D.1 Flight test trajectory of -20º and gondola's position

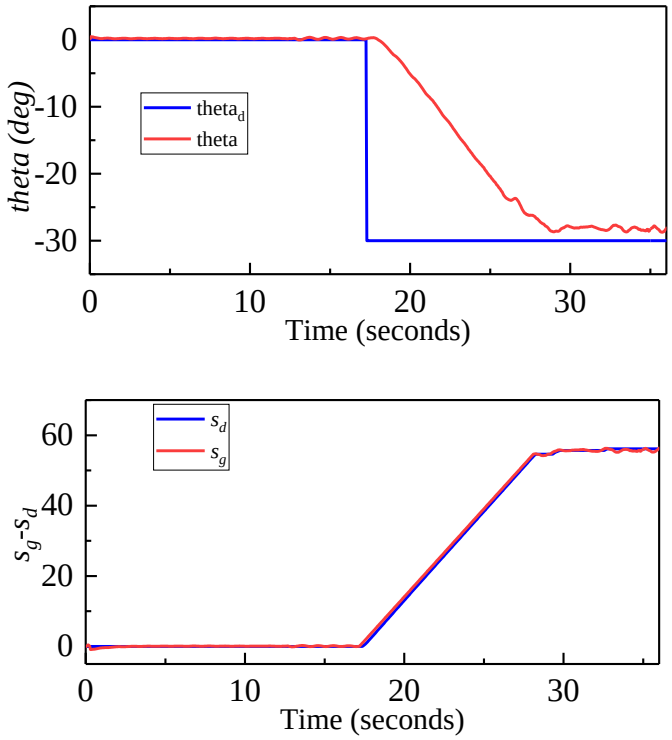


Figure D.2 Flight test trajectory of -30º and gondola's position

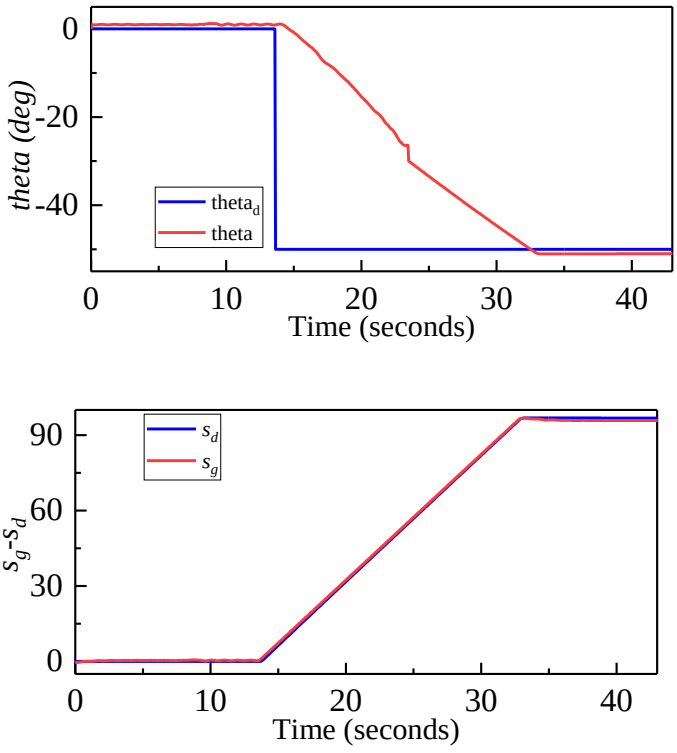


Figure D.3 Flight test trajectory of -50º and gondola's position

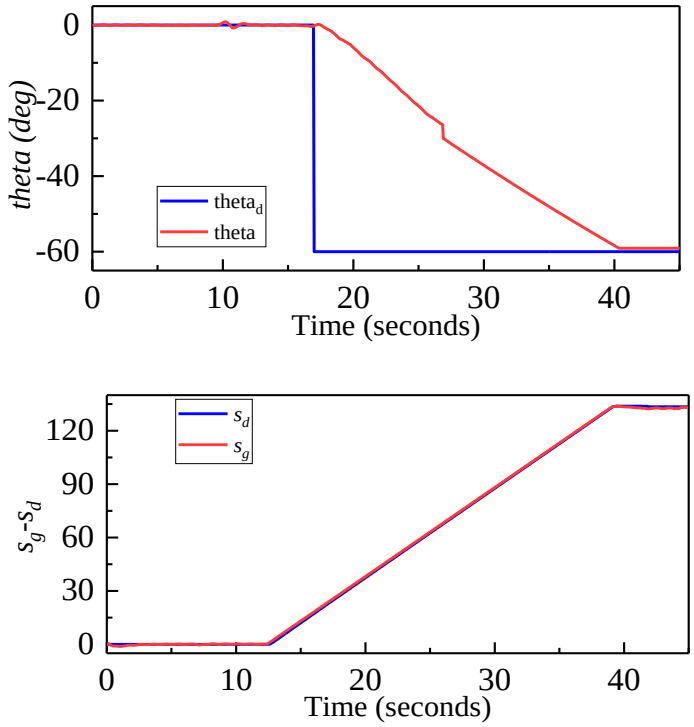


Figure D.4 Flight test trajectory of -60º and gondola's position

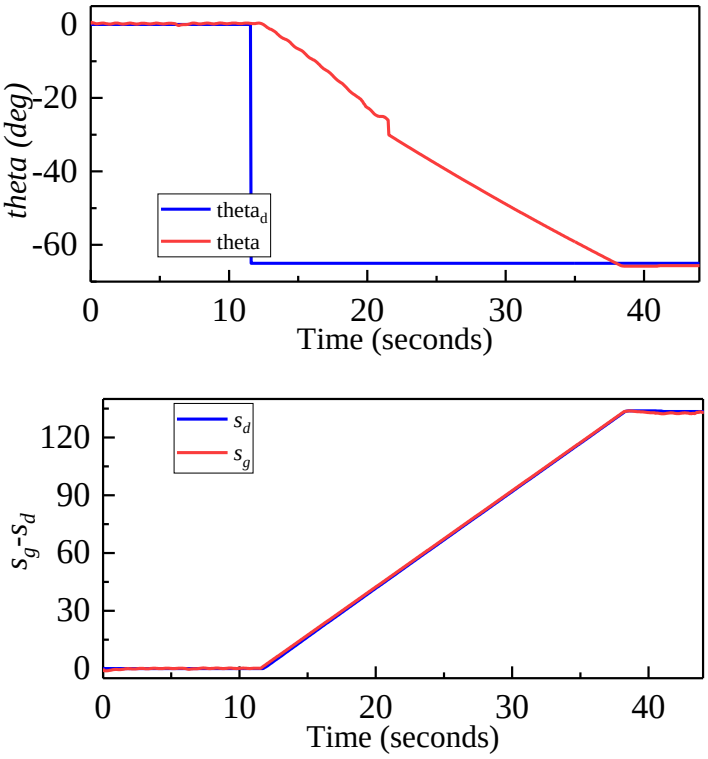


Figure D.5 Flight test trajectory of -65º and gondola's position

The experimental results of INDI

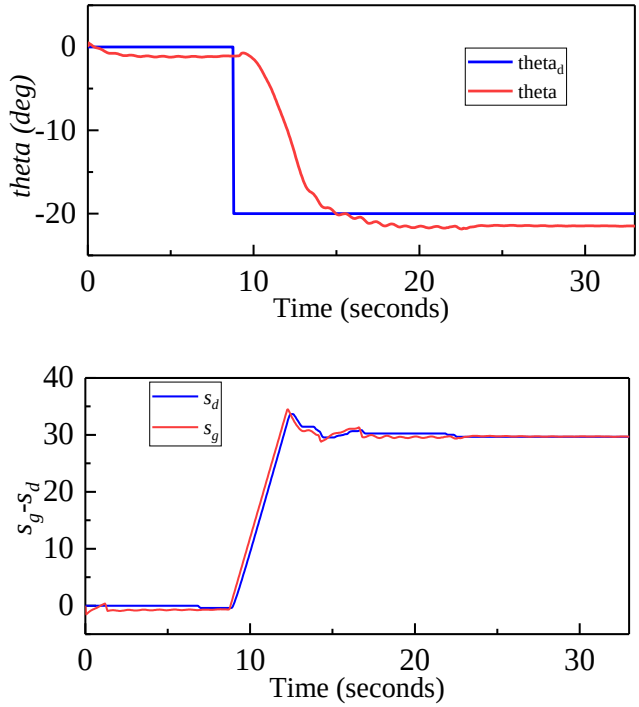


Figure D.6 Flight test trajectory of -20º and gondola's position

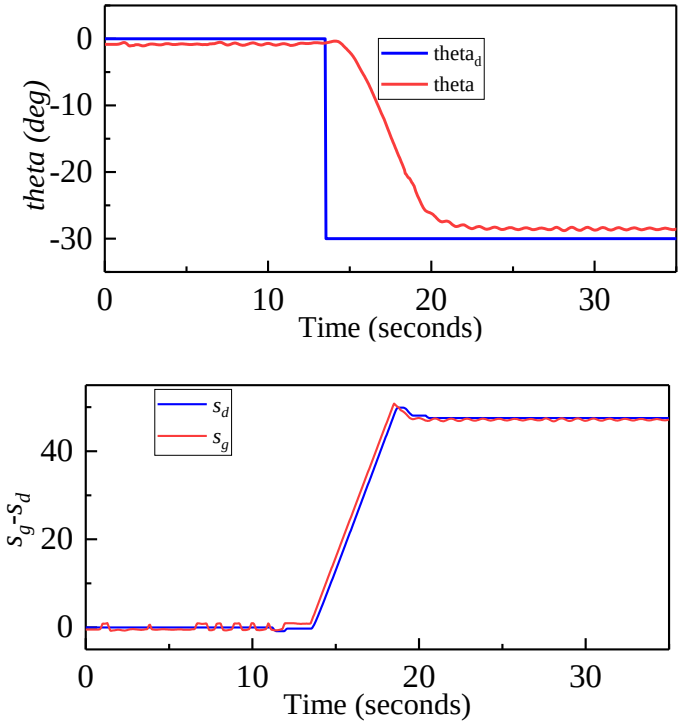
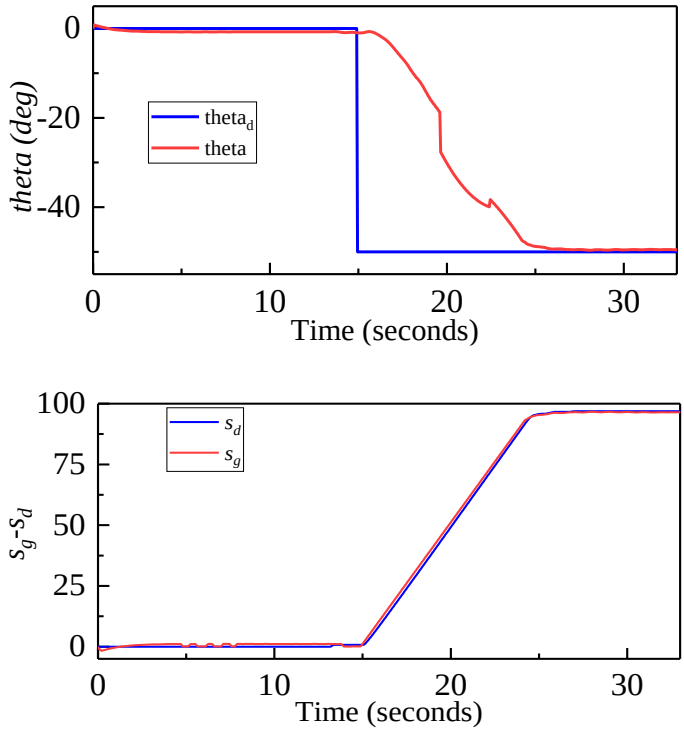


Figure D.7 Flight test trajectory of -30º and gondola's position



F Figure D.8 Flight test trajectory of -50º and gondola's position

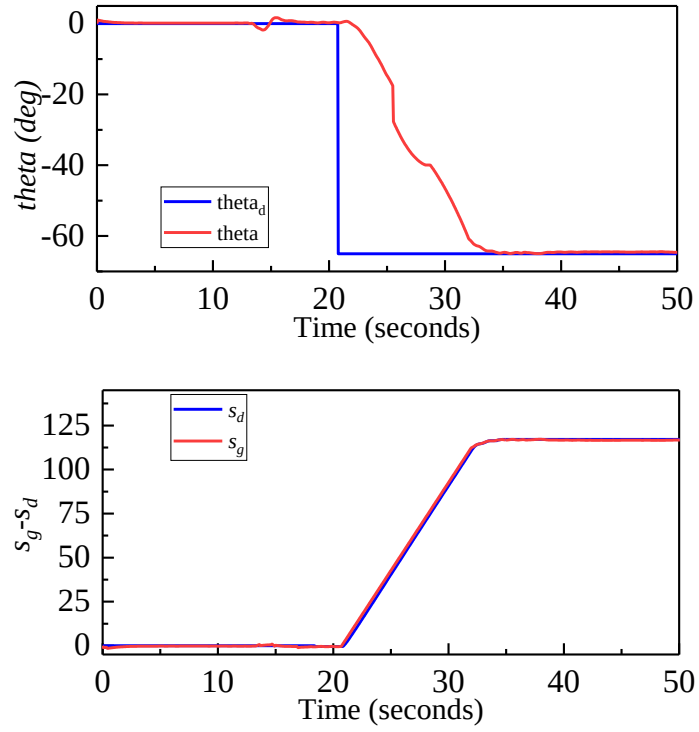


Figure D.9 Flight test trajectory of -65º and gondola's position

Results of the theoretical method

Results of Backstepping control

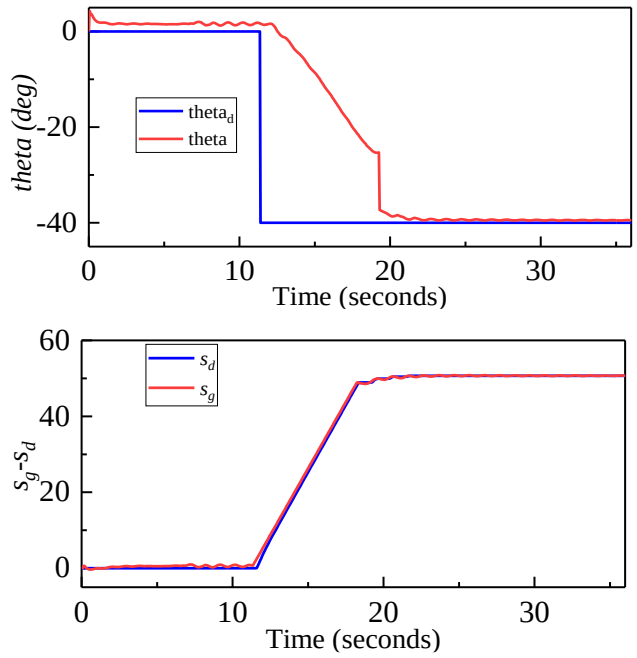


Figure D.10 Flight test trajectory of -40º and gondola's position

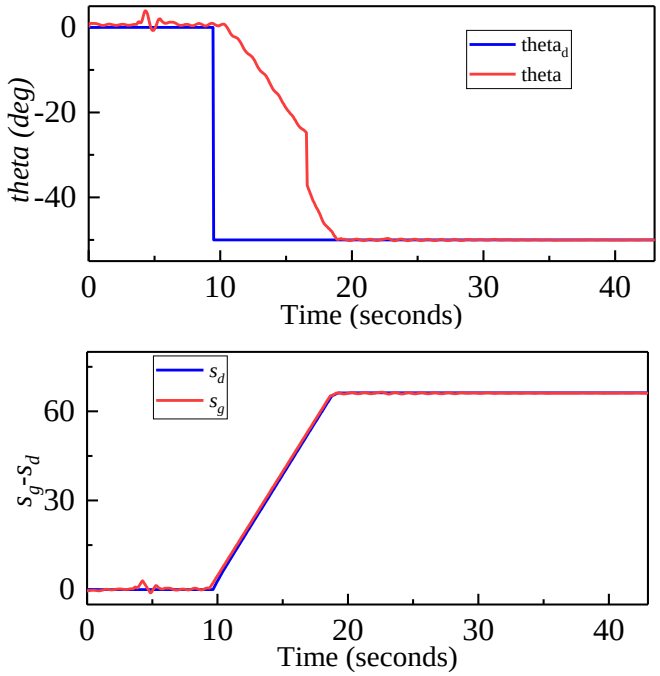


Figure D.11 Flight test trajectory of -50º and gondola's position

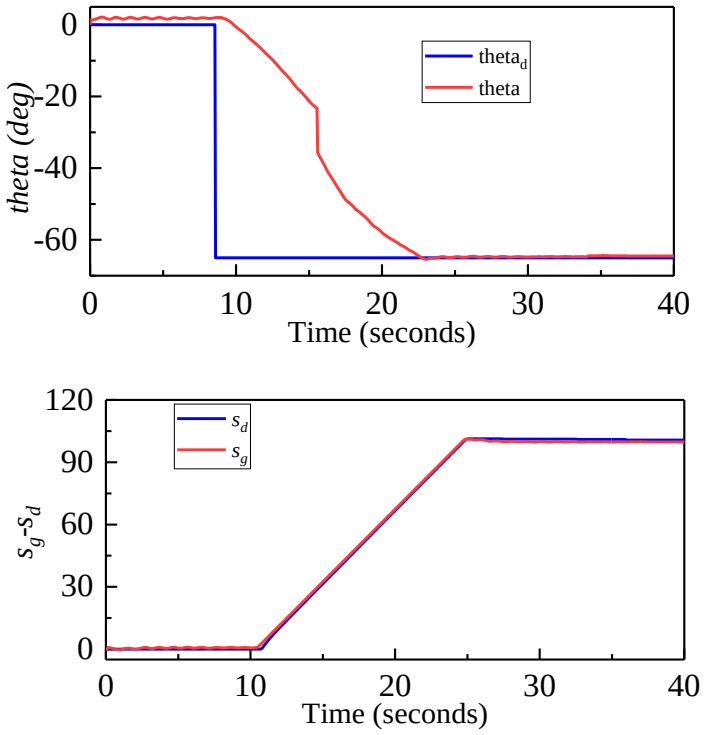


Figure D.12 Flight test trajectory of -65º and gondola's position

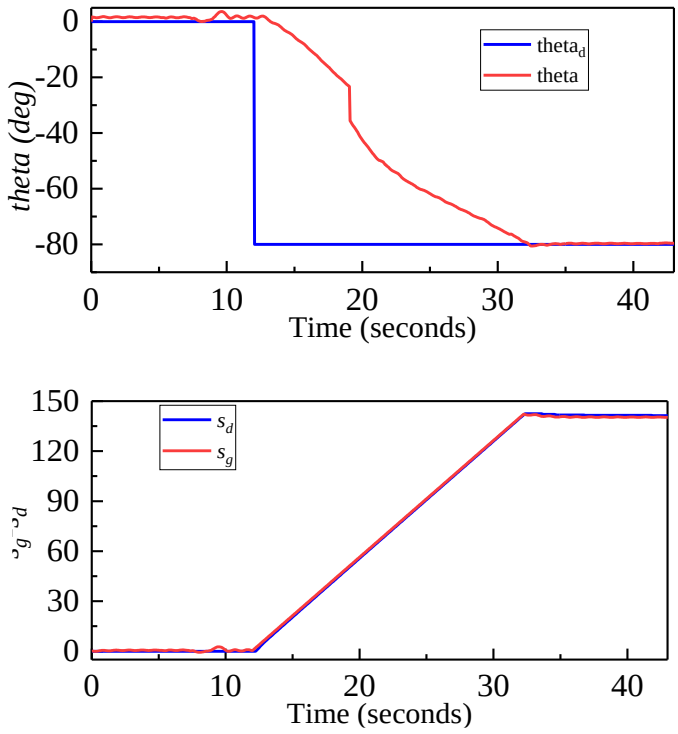


Figure D.13 Flight test trajectory of -80º and gondola's position

Results of INDI control

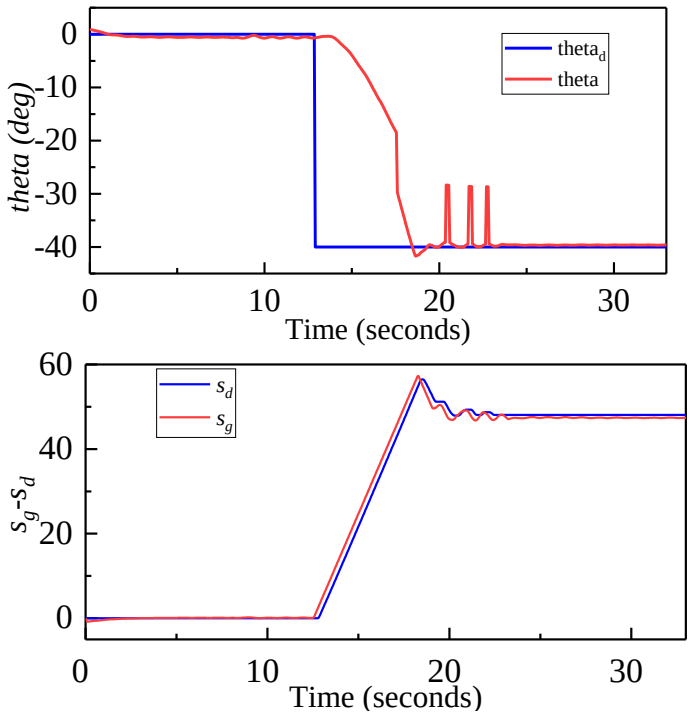


Figure D.14 Flight test trajectory of -40° and gondola's position

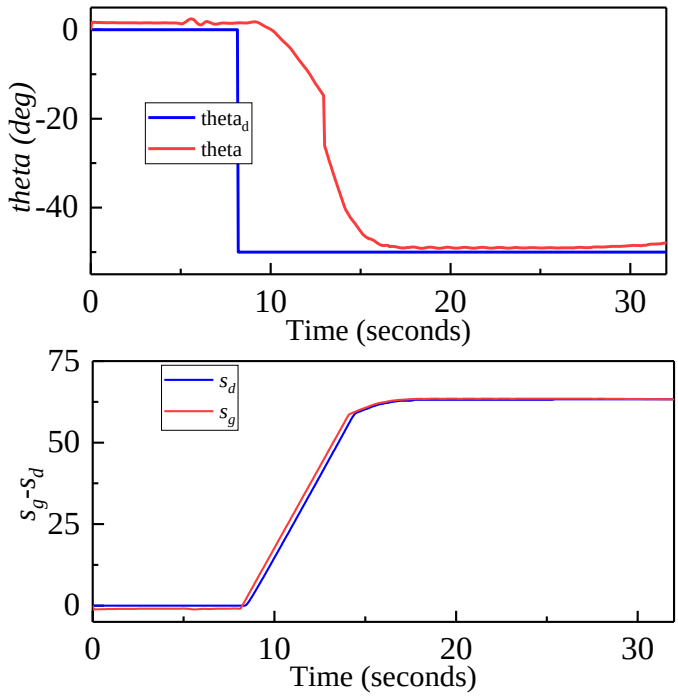


Figure D.15 Flight test trajectory of -50° and gondola's position

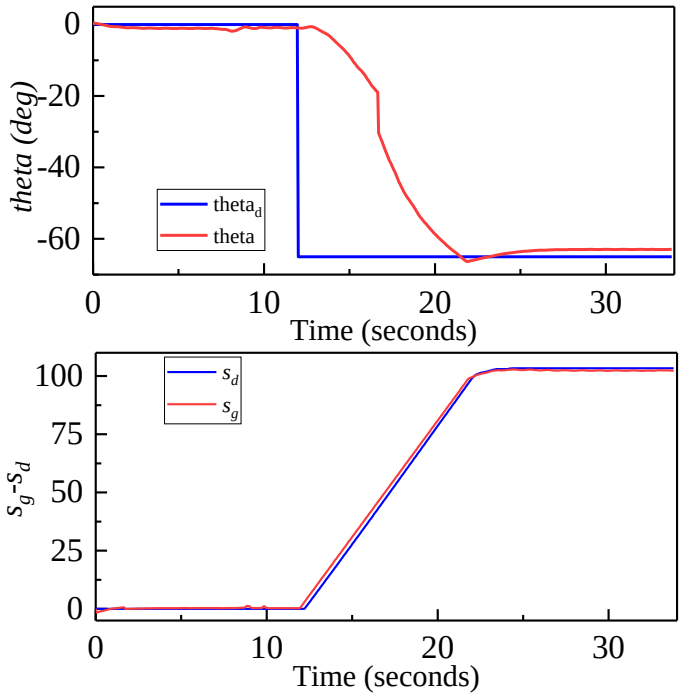


Figure D.16 Flight test trajectory of -65º and gondola's position

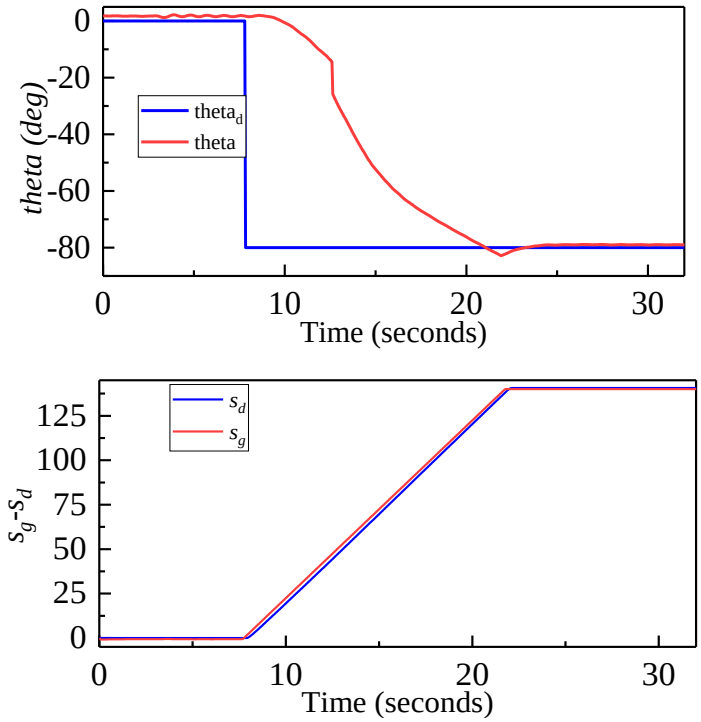


Figure D.17 Flight test trajectory of -80º and gondola's position

Appendix E Total Energy required

Backstepping control results

Table E.1 Energy required for ascent and cruise modes in calm condition using Gondola configuration

	Pitch angle									
	0º (Cruise mode)		5º		10º		15º		20º	
Velocity (m/s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)
1	1692	50	19775	577	9793	287	7186	194	5233	145
2	857	25	10676	287	6108	146	5475	96	3293	73
3	576	16.7	7614	192	6398	96	5020	64	2795	49
4	460	12.5	10457	144	5870	73	4407	48	2387	37
5	356	10	11666	116	6645	59	3364	39	2014	29

Table E.2 Energy required for ascent and cruise mode in windy condition using Gondola configuration

	Pitch angle									
	0º (Cruise mode)		5º		10º		15º		20º	
Velocity (m/s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)
1	1834	53	52856	582	32055	291	28122	197	26381	149
2	914	27	30440	290	25549	149	23106	97	21457	74
3	651	18	20189	197	18860	100	22297	66	14015	50
4	570	14	35534	148	10028	77	19380	49	13094	39
5	511	12	53254	120	14914	64	19754	41	9658	30

Table E.3 Energy required for descent mode in calm condition using Gondola configuration

	Pitch angle							
	-15º		-30º		-45º		-90º	
Velocity (m/s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)
1	6467	194	3377	100	2895	71	2015	52
2	3364	97	1773	51	1538	36	1178	27
3	2218	65	1364	34	1109	24	991	16
4	1727	49	971	25	913	18	884	13
5	1400	38	848	20	717	15	752	11

Table E.4 Energy required for descent mode in windy condition

	Pitch angle							
	-15°		-30°		-45°		-90°	
Velocity (m/s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)
1	40524	198	15081	104	9022	75	4002	54
2	14814	101	10985	56	7345	39	2621	29
3	8540	67	4955	39	5897	28	2076	19
4	5496	53	2032	29	2360	21	1425	16
5	4774	45	1320	25	1107	18	1032	14

Table E.5 Energy required for ascent and cruise mode in calm condition using vectored thrust configuration

	Pitch angle									
	0° (Cruise mode)		5°		10°		15°		20°	
Velocity (m/s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)
1	1671	50	19466	576	9641	288	7797	193	5271	146
2	849	25	9579	287	5061	144	4767	97	3953	76
3	571	16.7	6576	191	3786	96	3957	65	3045	49
4	423	12.5	4864	144	4282	72	3133	48	1828	37
5	394	10	4050	116	3167	58	1893	39	1557	30

Table E.6 Energy required for ascent and cruise mode in windy condition using vectored thrust configuration

	0° (Cruise mode)		5°		10°		15°		20°	
Velocity (m/s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)	`Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)
1	1926	52	21501	580	12569	292	9853	198	6551	150
2	1045	27	11427	291	6684	150	6261	100	4623	79
3	712	18	7848	195	4980	105	4361	69	3758	52
4	533	14	5474	148	4942	76	3852	51	2328	40
5	439	11	5030	130	3877	61	2512	45	1924	32

Table E.7 Energy required for descent mode in calm condition using vectored thrust configuration

	Pitch angle			
	-15°		Vertical descent	
Velocity (m/s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)
1	8924	198	3124	50
2	4516	97	2098	25
3	3157	65	4095	17
4	2802	50	4156	12
5	1929	39	4055	10

Table E.8 Energy required for descent mode in windy condition using vectored thrust configuration

	Pitch angle			
	-15º		Vertical descent	
Velocity (m/s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)
1	14272	200	4125	53
2	8241	99	3234	27
3	6047	68	4500	21
4	4390	52	5122	15
5	2829	43	5015	12

Table E.9 Energy required for ascent and cruise mode in calm condition using composite control configuration

	Pitch angle									
	0º (Cruise mode)		5º		10º		15º		20º	
Velocity (m/s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)
1	1824	50	19476	575	9935	287	6941	194	4942	146
2	848	25	10633	286	6431	146	5075	96	3603	73
3	622	16.7	6533	192	3473	96	4520	64	2795	49
4	438	12.5	5950	144	3042	73	3107	48	2095	38
5	363	10	6091	116	3398	59	2824	39	1874	29

Table E.10 Energy required for ascent and cruise mode in windy condition using composite control configuration

	Pitch angle									
	0º (Cruise mode)		5º		10º		15º		20º	
Velocity (m/s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)
1	1856	53	21856	578	12069	291	8525	197	5876	147
2	937	27	15436	289	10270	149	7106	97	4589	76
3	673	18	7618	195	5464	100	6442	66	4015	50
4	536	14	7460	148	3729	77	5700	49	3254	40
5	470	12	8109	120	4398	64	4854	41	2708	30

Table E.11 Energy required for descent mode in calm condition using composite control configuration

	Pitch angle							
	-15º		-30º		-45º		-90º	
Velocity (m/s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)
1	7005	194	3484	100	2635	71	1968	50
2	3584	97	1883	51	1369	36	998	25
3	2405	65	1344	34	1009	24	791	16
4	1857	49	1021	25	790	18	670	13
5	1582	38	848	20	698	14	642	10

Table E.12 Energy required for descent mode in windy condition using composite control configuration

	Pitch angle							
	-15º		-30º		-45º		-90º	
Velocity (m/s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)
1	7650	197	3831	102	2919	74	2110	52
2	3958	100	2585	55	1409	38	1327	27
3	2931	66	1972	38	1150	27	1136	18
4	2106	53	1632	28	870	20	857	15
5	1924	44	1088	24	813	17	790	13

INDI Control results

Table E.13 Energy required for ascent and cruise mode in calm condition using Gondola configuration

	Pitch angle									
	0º (Cruise mode)		5º		10º		15º		20º	
Velocity (m/s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)
1	3167	50	22501	577	18943	287	15083	194	12470	145
2	1826	25	17614	287	10111	146	8045	96	7189	73
3	810	16.7	8116	192	5302	96	3556	64	2811	49
4	612	12.5	11680	144	4274	73	2465	48	1447	37
5	570	10	12747	116	3616	59	1629	39	1083	29

Table E.14 Energy required for ascent and cruise mode in windy condition using Gondola configuration

	Pitch angle									
	0º (Cruise mode)		5º		10º		15º		20º	
Velocity (m/s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)
1	4269	53	35293	582	30241	290	26122	197	21140	149
2	2631	27	22256	290	10201	149	7591	97	6672	74
3	1467	18	19674	197	6651	100	4449	66	3681	50
4	1180	14	27709	148	5806	78	3689	49	2802	39
5	994	12	25891	119	4953	64	2112	42	2066	31

Table E.15 Energy required for descent mode in calm condition using Gondola configuration

	Pitch angle							
	-15º		-30º		-45º		-90º	
Velocity (m/s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)
1	9610	194	6847	100	3312	71	2910	50
2	7733	97	3022	51	1676	36	1801	25
3	3772	65	1759	34	1007	24	1073	16
4	2661	49	952	25	900	18	805	13
5	1793	38	812	20	729	15	668	10

Table E.16 Energy required for descent mode in windy condition using Gondola configuration

	Pitch angle							
	-15º		-30º		-45º		-90º	
Velocity (m/s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)
1	11638	198	8750	102	6562	73	3869	52
2	6887	100	6164	55	3649	38	2549	27
3	4598	67	3713	38	2393	26	2011	18
4	3723	53	2230	27	1763	20	1501	15
5	2287	44	1429	24	1380	17	1224	13

Table E.17 Energy required for ascent and cruise mode in calm condition using vectored thrust configuration

	Pitch angle									
	0º (Cruise mode)		5º		10º		15º		20º	
Velocity (m/s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)
1	2229	50	42831	577	9930	287	8037	193	6262	146
2	1496	25	18835	288	7243	144	5776	97	3644	76
3	1697	16.7	18200	195	7847	96	3547	65	3258	49
4	2552	12.5	11240	145	6158	71	4515	48	3801	37
5	2840	10	9295	115	9299	58	7867	39	5122	31

Table E.18 Energy required for ascent and cruise mode in windy condition using vectored thrust configuration

	Pitch angle									
	0º (Cruise mode)		5º		10º		15º		20º	
Velocity (m/s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)	`Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)
1	2495	52	47073	582	11610	293	10990	198	8481	150
2	1874	27	24207	292	9454	150	7967	100	5245	80
3	2212	18	22625	199	10272	105	5864	69	4058	52
4	3079	13	20944	149	12487	75	4905	52	4811	39
5	3266	11	21949	131	14145	61	10784	45	5944	34

Table E.19 Energy required for ascent and cruise mode in calm condition using composite configuration

	Pitch angle									
	0º (Cruise mode)		5º		10º		15º		20º	
Velocity (m/s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)
1	2065	50	18134	575	9092	286	5946	194	4422	145
2	1339	25	9233	286	4433	146	2914	96	2668	73
3	1123	16.7	5263	1931	3026	96	2167	64	1974	49
4	910	12.5	4565	144	2561	73	1986	48	1892	37
5	832	10	3961	116	2365	59	2096	39	1944	29

Table E.20 Energy required for ascent and cruise mode in windy condition using composite configuration

	Pitch angle									
	0º (Cruise mode)		5º		10º		15º		20º	
Velocity (m/s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)
1	2165	53	20856	579	10097	290	6945	197	5109	147
2	1412	28	11931	289	5705	149	3810	98	3544	75
3	1240	18	7029	195	3626	100	2610	66	2801	51
4	1117	14	5920	148	3212	77	2331	50	2204	39
5	1051	12	5564	120	3424	64	2361	41	2088	30

Table E.21 Energy required for descent mode in calm condition using composite configuration

	Pitch angle							
	-15º		-30º		-45º		-90º	
Velocity (m/s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)
1	6441	194	3283	100	2365	71	1786	50
2	3074	97	1840	50	1326	36	1140	25
3	2205	65	1641	34	1035	24	1054	16
4	2020	49	1503	25	965	18	980	13
5	2297	38	1630	20	1260	14	912	10

Table E.22 Energy required for descent mode in windy condition using composite configuration

	Pitch angle							
	-15º		-30º		-45º		-90º	
Velocity (m/s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)	Energy (Ws)	Travel time (s)
1	7030	197	3881	102	2774	74	2156	52
2	3785	100	2235	55	1665	38	1442	27
3	2900	66	2038	38	1450	27	1307	18
4	2770	53	1725	27	1394	20	1292	15
5	2837	44	2020	24	1306	17	1233	12

Appendix F Battery Usage

The performance of UAVs powered by on-board batteries is critically constrained by the limited battery lifetime. There is a lack of related work on optimizing the flight plans of battery-operated UAVs and the optimization of electric vehicles have been studied mainly for ground vehicles [122]. One of the goals of this study is to compare the airship performance with different actuation mechanism in terms of the electric power consumption. The battery is the only source of power for all electronics on-board including the propulsion system in this study. Therefore, the battery capacity will be studied. The battery capacity represents the maximum amount of energy that can be extracted from the battery under certain specified conditions. The most common measure of battery capacity is mAh (milliamp-hours), defined as the number of hours for which a battery can provide a current until a specified cut-of voltage is reached [158] and is expressed as:

$$C_b = \int_{\Delta t} I_{tot}\, dt \tag{0.1}$$

where C_b is the battery capacity in mAh and I_{tot} is the sum of current drawn from the battery by the motors and on-board electronics during the different operating time for each system and is calculating as:

$$I_{tot} = n\,.\,i_m + m\,.\,i_{servo} + i_{other} \tag{0.2}$$

where i_m and i_{servo} are the amounts of current drawn by the BLDC motors and the servo motors respectively, and *n* and *m* are the number of BLDC motors and servo motors respectively. And i_{other} is the amount of current drawn by the on-board equipment excluding the motors.

The amounts of current drawn by the BLDC motors are calculated based on the model used in several publication [131–135].

$$P_{in} = I_{in}\, V_{in} \tag{0.3}$$

where P_{in} is the electrical power input of the motor, I_{in} and V_{in} are the motor input current and voltage respectively.

$$i_m = I_{in} = \tau\, K_v\, \frac{2\pi}{60} + I_0 \tag{0.4}$$

$$V_{in} = \frac{\omega}{K_v \frac{2\pi}{60}} + \left(\tau\, K_v\, \frac{2\pi}{60} + I_0\right) R_{in} \tag{0.5}$$

where K_v is the motor's speed constant (rpm/V), I_0 is the no-load current, τ and ω are the required torque and rotational speed of the motor, and R_{in} is the winding resistance. The motor constants K_v, I_0 and R_{in} are usually given by all motor manufacturers.

The current drawn by the servo motor (i_{servo}) in no-load and maximum load (stall) conditions is expressed in [160] as:

$$i_{servo} = i_{nl} + \left(\frac{i_{stall} - i_{nl}}{\tau_{stall}}\right) \tau_{servo} \tag{0.6}$$

where i_{nl} is the no-load current of the servo motor, i_{stall} is the stall current, τ_{servo} and τ_{stall} are the servo torque and the stall torque of the servo motor.

Appendix G Motor-Propeller Static & Dynamic Thrust Calculation

Theoretical propeller dynamic thrust equation was developed by Staples [164] from simple momentum theory and then fit to experimental data. It is of the form:

$$T = \rho \frac{\pi d^2}{4} (V_e^2 - V_e V_o) \qquad \text{(G.1)}$$

where T is thrust in (N), d is the propeller diameter, ρ is the air density, ρ = 1.225 kg/m3, V_e is the exit velocity of the air (m/s) or the induced velocity of the air by a propeller and V_0 is the aircraft velocity (m/s).

V_e was assumed approximately equal to the pitch speed of the propeller.

$$V_{pitch}(mph) = n_{prop}(rpm) \,.\, P(in) \,.\, \frac{1\,ft}{12\,in} . \frac{1\,mile}{5280\,ft} . \frac{60\,min}{hr} \qquad \text{(G.2)}$$

where P is the propeller's pitch (in) and n is its speed (rpm). By substituting the value of V_{pitch} in (G.1), we get,

$$T = 1.225 \ \frac{\pi (0.0254 \,.\, d)^2}{4} \left[\left(\frac{n}{60} \ 0.0254 \,.\, P \right)^2 - \left(\frac{n}{60} \ 0.0254 \,.\, P \right) . V_o \right] \qquad \text{(G.3)}$$

Then Staples added a correction factor to make the thrust equation accurate, $(k_1 \frac{d}{P})^{k_2}$, the correction factor was then multiplied to (G.3). The result of the theoretical static thrust by setting V_0 = 0 m/s verses the rotational speed is shown in Figure G.2.

Experimental tests

The static test of motor-propeller combination was preformed by using the RCbenchmark motor jig and RCbenchmark software. The test setup as shown in Figure G.1, was used to measure power source voltage, current, motor mechanical speed (via the measurement of motor electrical speed), motor-propeller static thrust, and torque. The experiment was performed using a high discharge Li-Po battery with 3 cells (11.1V). The brushless motor was used in the test along with APC 9 x 4.7 propeller (9 inches in diameter with pitch of 4.7 inches). The measurements were taking at auto increment of ESC signals. The static thrust verses the ESC signals and the rotational speed (rpm) were gathered and shown in the Figure G.1.

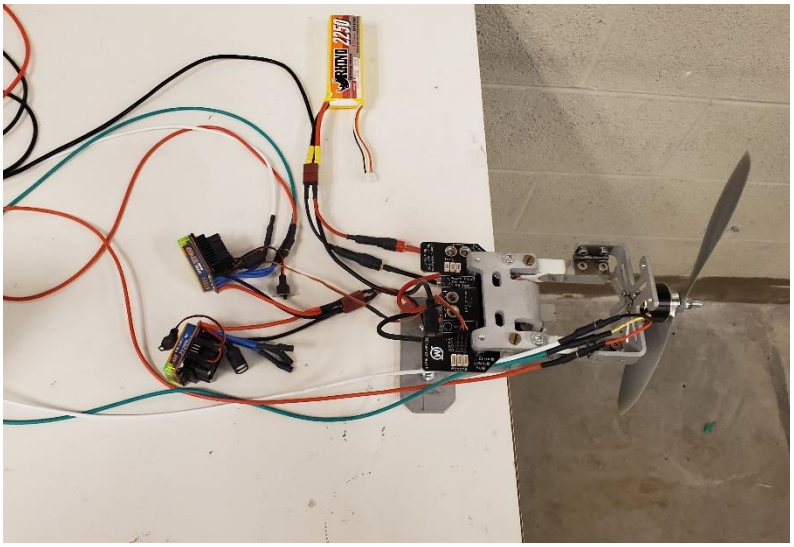


Figure G.1. Static Testing of Motor-Propeller Combinations

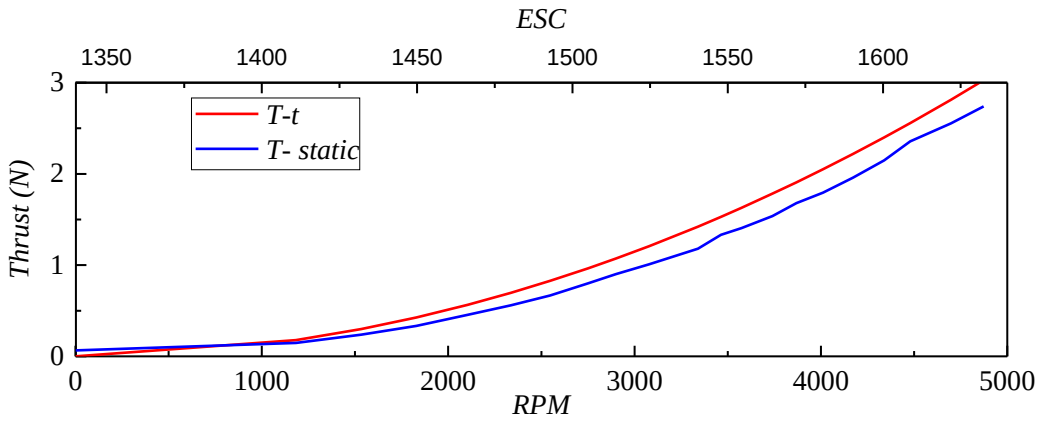


Figure G.2 the measured static thrust and theoretical thrust at different *n* (rpm)

As can be seen in the Figure G.1 the error between the two results are increasing with increasing the rotational speed. It is possible that the difference resulted between the curves are due to the assumptions Stapes made in equation (G.1). He assumed that the induced velocity is approximately equal to pitch speed ($V_e \approx V_{pitch}$). However, this is unrealistic, as the inflow velocity varies across the cross-sectional inflow area, or rotor disc area. In addition, the correction factors K_1 and K_2 were determined by trial and error and the value of the correction factor was constant. However, by checking the error in the graph, it can be noticed that the error is increased with increasing the rotational speed, therefore adding correction factor as constant values will keep the growth rate of the error increases as the time increases. Alternative method to calculate a correction function is developed here to correct and minimize the error in (G.3) in order to approximate motor and propeller performance without the hassle of measuring and testing the products in a laboratory setting. The developed function is then used to calculate the dynamic thrust. The correction function assumed here is $f(n)$ which is multiplied to (G.3) instead of the correction factor of Staples and $V_0 = 0$ m/s.

$$f(n) = \frac{T}{\left(1.225 \ \frac{\pi(0.0254 \,.\, d)^2}{4}\left[\left(\frac{n}{60} \ 0.0254 \,.\, P\right)^2\right]\right)} \qquad \text{(G.4)}$$

By using the experimental data of static thrust and the rotational speed, the correction function with respect to the rotational speed can be plotted as in Figure G.3. By using the curve fitting toolbox in Matlab, a third-order polynomial was created.

$$f(n) = -1.521 \times 10^{-12} . n^3 + 1.512 \times 10^{-8} \ . \ n^2 - 2.522 \times 10^{-5} . \ n + 0.4135 \qquad \text{(G.5)}$$

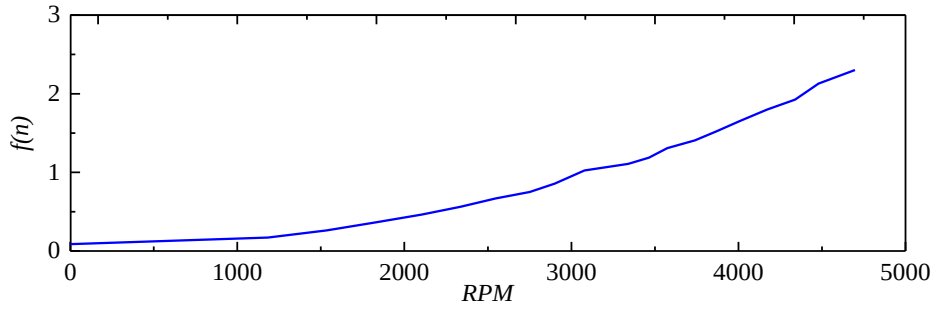


Figure G.3 the correction function at different *n* (rpm)

The equation (G.3) then was multiplied by the correction function *f* (*n*);

$$T = 1.225 \ \frac{\pi(0.0254 \,.\, d)^2}{4}\left[\left(\frac{n}{60} \ 0.0254 \,.\, P\right)^2 - \left(\frac{n}{60} \ 0.0254 \,.\, P\right) . V_o\right] . f(n) \qquad \text{(G.6)}$$

The results of the static test and the modified (G.6) in addition to the theoretical equation of Staples at V_0 = 0 m/s are shown in Figure G.4

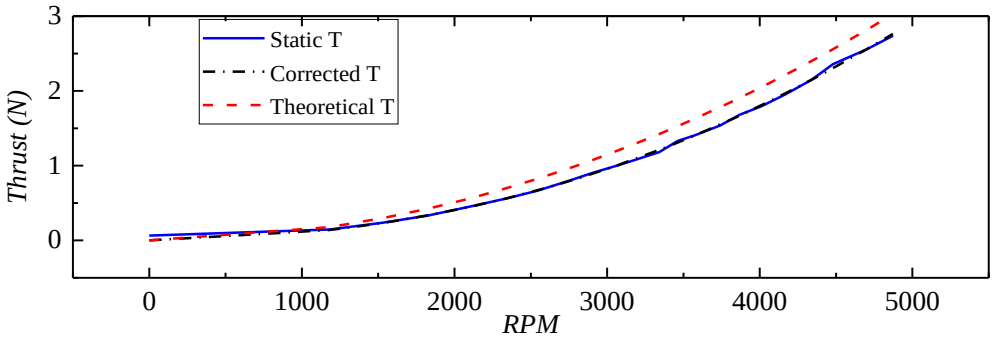


Figure G.4 the static test, theoretical and developed equation results

Dynamic thrust

The objective of the present experiment was to determine the thrust performance of the electric motor-propeller combinations at static conditions and simply estimate motor and propeller performance in dynamic conditions without the need for laboratory testing. The dynamic thrust is calculated here by using eq (G.6) with the corrected function at different aircraft velocities. Figure G.5 shows the result of the modified equation and the results of the dynamic equation of Staples.

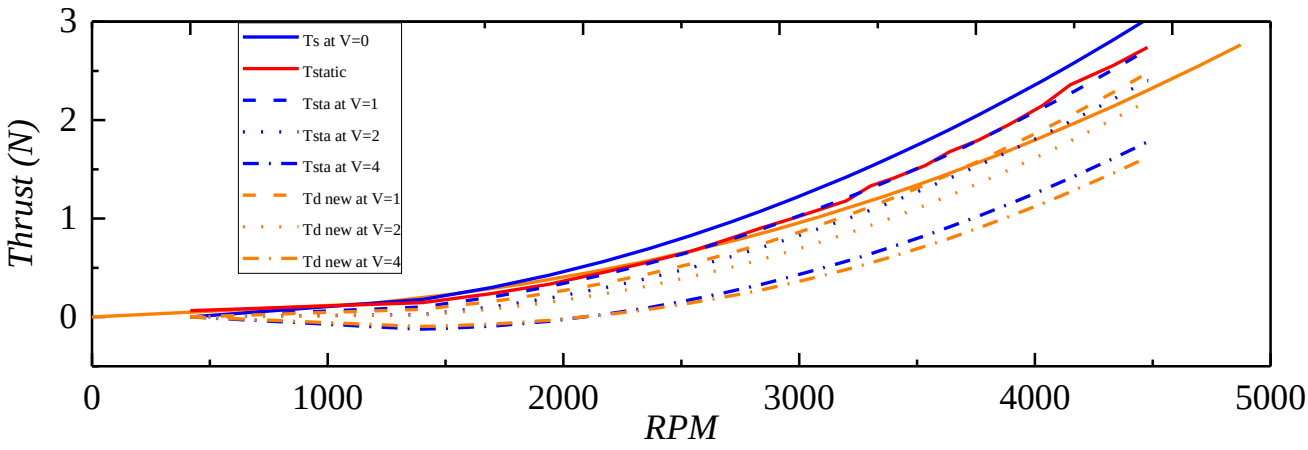


Figure G.5 the dynamic equation and developed equation at different aircraft velocity

Appendix H Manuscripts

H.1. Pitch Tracking for an Airship with Moving Gondola Using Backstepping Control

Presentation and full paper in the 6th International Conference on Mechatronics and Robotics Engineering, 2020

2020 6th International Conference on Mechatronics and Robotics Engineering

Pitch Tracking for an Airship with Moving Gondola Using Backstepping Control

Ali Mansur
Department of Mechanical Engineering
University of Ottawa
Ottawa, Canada

Eric Lanteigne
Department of Mechanical Engineering
University of Ottawa
Ottawa, Canada

***Abstract*—This paper presented a backstepping approach for trajectory tracking control of an airship using a moving gondola as a control input. The gondola and thrusters can travel from the mid-rear of helium envelop to the front via a curved rail thereby altering the location of the center of gravity (CG) with respect to the center of volume (CV). The dynamic equations of airship is derived using the Newton–Euler method, and the model was implemented and simulated in Matlab/Simulink to illustrate the effectiveness of the designed controller for tracking pitch trajectories in the presence of uncertainty and wind disturbances. The global asymptotically stability is proved by Lyapunov stability analysis. The results show the effectiveness of the CG control for pitch and altitude tracking.**

Keywords-component; nonlinear control; UAVs; simulation; stability

I. INTRODUCTION

In the past twenty years, airships have experienced a resurgence in both manned and unmanned applications. A growing safety trend is to keep humans out of danger especially in the field of Search and Rescue (SAR), military operations, and firefighting, etc. motivated researchers over the past several decades. Researchers have been working impetuously to understand the airship flight behavior, design their control systems, and optimize their flight trajectories. Amongst the published work in this field and considered a landmark in the subject even nowadays is the paper published by DeLaurier 1979 [1]. He used his formulation, an analytical model and assumptions to investigate the dynamic stability of several airship designs. This analysis has been applied to full size airships, U.S. Navy, Airships ZR-1 and ZR-4. In 2011, Li et al. [2] published a comprehensive review of airship dynamics covering many decades. In this review, different prediction approaches were reviewed for different aerodynamic effects to airships based on a rigid-body assumption. The lighter-than-air (LTA) airships were compared for their long endurance in air, high payload-to-weight ratio, and low noise level features, with fixed-wing aircrafts and rotor-wing aircrafts [3].

The dynamic modeling of conventional Heavier-Than-Air (HTA) aircraft has been broadly studied and become highly reliable. However, for a variety of reasons, limitations exist when applying these HTA dynamics models to airships. First, the airships have a lift generation mechanism, which is different from the HTA aircrafts due to the use of light lifting gas. Certain solid–fluid interaction forces can be neglected for the HTA aircraft, such as buoyancy and those related to the inertia of the surrounding air, but these forces become important for airships. Second, airships have very different shape and structure compared with HTA aircraft. Reductions of 20–30% of the energy loss caused by aerodynamic drag are possible in the future [5]. An aircraft of conventional shape must fly against the apparent wind in order to have low drag [6]. Ram et al. [7] carried out a study on aerostat envelope shapes on the basis of aerodynamic considerations, they found that the envelope shape affects the payload capacity in many ways. In their studies, it was a 20% increase in payload capacity due to decrease in fin size.

A significant amount of literature presented different approaches to improve the dynamic model of the airships. These approaches could be classified into two groups. The first one uses the Newton-Euler method and the second group uses the Lagrangian formulation. This formulation induces very complex equations of motion because of the interaction fluid-structure between the airships and their environment [4]. Unmanned aerial vehicles (UAVs) serve a wide range of applications. Many of the UAV applications require the ability for autonomous flight, involving the development of a flight control and navigation system. Several control approaches for flight control of an airship have been proposed in literature, these can be categorized in to linear control design techniques and nonlinear control techniques [5]. The classical linear control design techniques have been used for flight control problems for many years, especially when wind gusts and the other disturbances do not consider into the real system conditions. However, if the automatic control system is to cover the complete aerodynamic range from hover to cruise flight, then the control solution must be able to cope with the nonlinear and underactuated airship dynamics [6]. Several solutions in literature have been made in this field by applying different control solutions aiming at the autonomous airship goal, aerial platform positioning and path-tracking [7].

The Backstepping method is a nonlinear approach which uses Control Lyapunov Functions (CLF), and has been used in UAVs and has proven to offer robustness in the presence of parametric uncertainties [8–13]. A backstepping methodology was adapted to design a closed-loop trajectory-tracking controller for an underactuated airship [14]. The authors proposed a time-varying stabilizing controller of the position and the orientation of an underactuated autonomous airship. They found that the airship could not be stabilized to a point using continuous pure-state feedback law. However, the stabilization problem was solved with an explicit homogeneous time varying control law, based on an averaging approach and the stabilization in [15] was

978-1-7281-5739-9/20/$31.00 ©2020 IEEE

achieved by using a time-varying continuous feedback. Marin and Samson [15] used the backstepping technique to derive explicit time-varying feedbacks to ensure exponential stability of the closed loop system.

In recent years, the moving mass control has been studied and applied in spacecraft, underwater vehicles, and low-altitude airships [16–18]. Moving mass control methods such as moving ballast, attitude adjustment deputy ballonets, and the allocation strategy in conjunction with the traditional elevator have been studied, the pitch handling characteristics have not been researched in-depth. Chen et al. [19] investigated the mechanism of moving-mass control and compared the control capabilities between the moving mass and the aerodynamic control surface. The results showed that the movements of the moving mass changes the pitch and roll angles of the airship and were the primary factors affecting the attitude moment. Moving-mass control was independent of velocity. The results of maximum pitch angles when the mass was at maximum ends were -9° and +9°. Lanteigne et al. [20] used a PID controller to control the airship pitch using the gondola position. They studied a miniature-unmanned airship with a moving platform to address the limited altitude maneuverability of airships. In this study, the model developed by Lanteigne et al. was modified to reflect the new vehicle architecture and apply the new platform to evaluate the capabilities and the limitation of the design.

This paper presents the airship dynamic model and develop a backstepping controller for trajectory tracking control of an unmanned airship. Global asymptotically stability of tracking control law is proven by applying Lyapunov stability analysis. Finally, the proposed control law is simulated on full . 6-DOF nonlinear airship dynamic equations to show the performance of the controller.

II. Airship Architecture and Nonlinear Dynamic Model

A. Airship Architecture

The proposed vehicle architecture presented in this article as shown in Fig. 1 used a moving gondola to provide an alternative solution to over-actuation and ballonets for the rapid altitude changes required when landing or avoiding obstacles and thus improves the maneuverability of unmanned airship, and to reduce the weight and energy costs for the controlled flight. The pitch angle of the airship, without rudder and elevator, based on moving gondola control are analyzed. The airship in [21] features a 1.83 m helium envelope had the ballast contained within the gondola. The gondola moved along a straight rail from −0.45 m to 0.5 m with respect to the CV and had two forward facing thrusters providing a maximum thrust of 0.15 N at full throttle. The 3 m-airship proposed in this study is shown in Fig. 1. The physical properties of the airship are listed in Table 1 and the airship has;

- The thrusters remain in line with the center of volume (CV) of the airship.
- The fins are located at the bottom to facilitate fabrication.
- The new airship will have the ability to move the gondola to the front end of the airship via a curved rail therefore allowing a -90° pitch.

The gondola travels along the rail from -0.81 m to 0.96 m along the straight rail and from 0.96 m to 1.40 m along the circular rail at the bow as shown in Fig. 2. The initial position of gondola is s_g = 0 m when it is located directly below the CV.

B. Nonlinear Dynamic Model

The non-linear dynamic model for the proposed architecture is derived using the Newton-Euler method. In order to apply trajectory tracking control strategies, a complete dynamic mechanistic model of the airship is needed based on the physical characteristics of the prototype platform as well as empirical coefficients derived from similar airships.

Two frames of reference are defined to describe the airship motion; the inertial reference axis, the popular North-East-Down (NED) axis system, this system is a fixed point on the earth throughout the simulation. The airship body axis is defined at the center of volume (CV) of the airship with the x-axis along the centerline and pointing to the nose, the z-axis is positive downward and the y-axis is positive in the direction perpendicular to xz-plane as in Fig. 1. The orientation of the body axis system is represented by Euler angles (roll ϕ pitch θ and yaw ψ).

The airship was modeled as a rigid body with three translational and three rotational DOFs and hence, the dynamics model is composed of six nonlinear differential equations describing the translational and rotational dynamics. For simplicity, the non-linear dynamic model of the proposed airship was developed using Newton-Euler equations as derived by Gomes [22] with addition from Recoskie [23].

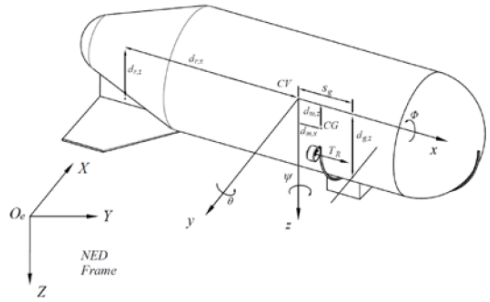


Fig. 1. Body axis convention and inertia axis of the airship

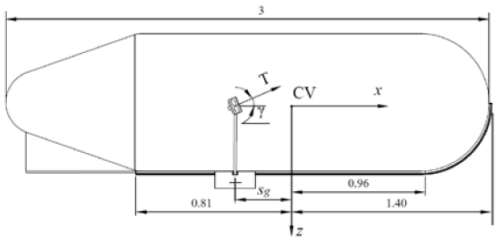


Fig. 2. Side view of the proposed airship with dimensions in (meters)

The model's reference system is shown in Fig. 1 and its equation of motion is defined as,

$$\mathbf{M}\dot{\mathbf{x}}_v + \mathbf{D}(\dot{\mathbf{x}}_v) = \mathbf{E} + \mathbf{U} \quad (1)$$

where, **M** is the 6x6 mass matrix, **D** is the 6x1 dynamics vector, **E** is the 6x1 model uncertainty vector, $\mathbf{x}_v$ is the 6x1 state vector of linear and angular velocities expressed in the moving frame located at the CV, **U** is the 6x1 external forces and moments and defined as,

$$\mathbf{U} = \mathbf{A}(\dot{\mathbf{x}}_v) + \mathbf{G}(\boldsymbol{R}_1) + \mathbf{F}_u \quad (2)$$

where, **A** is the 6x1 aerodynamic vector, **G** is the 6x1 gravitational and buoyancy vector, $\boldsymbol{R}_1$ is the directional cosine matrix and $\mathbf{F}_u$ is the 6x1 propulsive force vector.

The mass matrix contains the true mass and inertias of the airship as well as relevant added mass terms equation (3). After applying simplifications due to vehicle symmetry,

$$\mathbf{M} = \begin{bmatrix} \mathrm{M}_a & -m\mathbf{d}_{CG} \\ m\mathbf{d}_{CG} & \mathrm{J}_a \end{bmatrix} = \begin{bmatrix} m_x & 0 & 0 & 0 & md_{m,z} & 0 \\ 0 & m_y & 0 & -md_{m,z} & 0 & md_{m,x} \\ 0 & 0 & m_z & 0 & -md_{m,x} & 0 \\ 0 & -md_{m,z} & 0 & J_x & 0 & -J_{xz} \\ md_{m,z} & 0 & -md_{m,x} & 0 & J_y & 0 \\ 0 & md_{m,x} & 0 & -J_{xz} & 0 & J_z \end{bmatrix} \quad (3)$$

$$\mathbf{d}_{CG} = \begin{bmatrix} 0 & -d_{m,z} & 0 \\ d_{m,z} & 0 & -d_{m,x} \\ 0 & d_{m,x} & 0 \end{bmatrix} \quad (4)$$

where $\mathbf{d}_{CG}$ is a skew symmetric matrix containing distances between the CV and CM. M_a and J_a are the added mass and added inertia matrices and are computed from the sum of the true mass and inertia and are defined in [20]. Note that J_a and $\mathbf{d}_{CG}$ are both expressed in the body frame. The mass of the airship is equal to the sum of the mass of the gondola, the rail, the envelope, the motors and the fins $m=m_g+m_R+m_E+m_m+m_F$.

$$d_{m,x} = \frac{m_g \times s_g + (2 \times m_m \times x_m)}{m}$$
$$d_{m,z} = \frac{m_g \times z_g + m_r \times L_r + m_f \times L_f}{m} \quad (5)$$

where

$$\left.\begin{aligned} z_g &= d_{p,z} \\ x_m &= s_g \end{aligned}\right\} \quad -0.81 \le s_g \le 0.96$$

$$\left.\begin{aligned} z_g &= d_{p,z} \times cos\zeta \\ x_m &= s_g - (d_{p,z} \times sin\zeta) \end{aligned}\right\} \quad s_g > 0.96$$

where Z_g is the distance between the CV to the CM of the gondola along z, x_m is distance between the CV to CM of the propeller along x, ζ is the angle between the link and z axis when the gondola is in the curved link. L_f and L_r are the distance between the CV to the CM of the rail and fin along z respectively. The masses and the inertias of the fixed components as indicated by the subscript f represent those of the envelope and rail and were found based on the dimensions of the airship in the computer-aided drafting program SolidWorks, while virtual masses terms k_1, k_2, and $k`$ above were approximated using the works of Lamb [24] and Munk [25]. Table 1 lists the geometric and physical properties of the simulated airship.

The 6×1 column matrix that contains the dynamic terms associated with inertial linear and angular velocities is given by (6),

$$\boldsymbol{D} = \begin{bmatrix} \omega \times (\mathrm{M}_a v) - m\omega \times (\omega \times \mathrm{r}_{CG}) \\ \omega \times (\mathrm{J}_a \omega) + m\mathrm{r}_{CG} \times (\omega \times v) \end{bmatrix} \quad (6)$$

where the axis convention is defined in Fig. 1. The forces and moments in equation (6) are derived based on Newton's laws for rigid body motion about the CV in the body frame. These equations consist of both centrifugal ($\omega \times \mathrm{M}_a v$) and Coriolis ($\omega \times \mathrm{J}_a \omega$) components, which are a function of the linear and angular velocities of the vehicle.

Since the centre of rotation is considered to be offset from the centre of mass, the velocity used for these components be modified according to

$$v = v_{CG} + \omega \times r_{CG}, \quad r_{CG} = \begin{bmatrix} d_{m,x} & 0 & d_{m,z} \end{bmatrix}^{\mathrm{T}}$$

The components of the dynamics vector simplified due to the airship symmetry about the x-z plane and the complete derivation can be found in [21].

The aerodynamic model derived based on the work of both Jones and Mueller [26], [27] with additional rotational dampening added.

The axis convention is defined in Fig. 1. The components of the aerodynamic vector **A** were previously defined in [21].

All other variables and coefficients are summarized in [20], and including references to where they can be determined for any airship platform using a combination of semi empirical models and geometric functions. The gravitational and buoyancy vector **G** is given by,

$$\mathbf{G}(R_1) = \begin{bmatrix} m\mathrm{g} \\ m\mathbf{d}_{CG}\mathrm{g} \end{bmatrix} - \begin{bmatrix} \rho_a V_u \mathrm{g} \\ \rho_a V \mathbf{d}_{CG}\mathrm{g} \end{bmatrix} \quad (7)$$

where V_u is the volume of the helium envelope, $\mathbf{d}_{CG}$ is defined in equation (4), and g is the gravitational vector expressed in the body frame and R_1 is a rotation matrix (also known as the 123 or roll-pitch-yaw Euler sequence),

$$g = \begin{bmatrix} g_1 & g_2 & g_3 \end{bmatrix}^T = R_1^T \begin{bmatrix} 0 & 0 & g \end{bmatrix}^T$$

$$R_1 = \begin{bmatrix} \mathrm{c}\theta\mathrm{c}\psi & \mathrm{s}\phi\mathrm{s}\theta\mathrm{c}\psi - \mathrm{c}\phi\mathrm{s}\psi & \mathrm{c}\phi\mathrm{s}\theta\mathrm{c}\psi + \mathrm{s}\phi\mathrm{s}\psi \\ \mathrm{c}\theta\mathrm{s}\psi & \mathrm{s}\phi\mathrm{s}\theta\mathrm{s}\psi + \mathrm{c}\phi\mathrm{c}\psi & \mathrm{c}\phi\mathrm{s}\theta\mathrm{s}\psi - \mathrm{s}\phi\mathrm{c}\psi \\ -\mathrm{s}\theta & \mathrm{s}\phi\mathrm{c}\theta & \mathrm{c}\phi\mathrm{c}\theta \end{bmatrix} \quad (8)$$

The uncertainty due to several parameters (calculation based on the size and weight, inertia term, the uncertainty

arises from assumptions made about the vehicle aerodynamics and the actuator dynamics) is modelled as,

$$\boldsymbol{E} = \sigma_1(\mathbf{A}+\mathrm{F}_u)^{\mathrm{T}} \tag{9}$$

where σ_1 is a 6x1 vector of uniformly distributed random percentages within the range of ±10%. This range was suggested by Recoskie [23] based on the variation between methods to determine empirically derived constants.

The input force vector is expressed in form of 6x1 matrix, which contains the propulsive forces and moments. The input equation of the vectored thrust written as,

$$\mathbf{F}_u = \begin{bmatrix} (T_R + T_L)cos\gamma \\ 0 \\ -(T_R + T_L)sin\gamma \\ (T_L - T_R)d_{p,y}\ sin\gamma \\ -(T_R + T_L)s_g\ sin\gamma \\ (T_L - T_R)d_{p,y}\ cos\gamma \end{bmatrix} \tag{10}$$

where T_R and T_L are the right and left propeller thrust, γ the angle of the thruster as shown in Fig. 2, and $d_{p,y}$ and $d_{p,z}$ are defined in Table 1.

C. Kinematics

The position of all points of the airship body frame with respect to the inertial fixed frame can be completely defined by knowing the orientation of a body fixed frame to the airship body and the position of its origin with respect to the inertial reference frame [5]. The body fixed frame origin was chosen at center of volume (assumed to coincide with the gross center of lift). Vehicle kinematics were applied in tandem to determine the vehicle's velocity and positional changes and how they translate to the earth reference frame. Coordinates and trajectories in the earth reference frame can be determined from the equivalent states in body reference frame pre-multiplied by the Jacobian matrix,

$$\dot{\mathrm{x}}_{\mathrm{g}} = \mathrm{J}\dot{\mathrm{x}}_{\mathrm{v}} + v_{\mathrm{w}} = \begin{bmatrix} R_1 & 0_{3x3} \\ 0_{3x3} & R_2 \end{bmatrix} \dot{\mathrm{x}}_{\mathrm{v}} + v_{\mathrm{w}} \tag{11}$$

where v_w is the total wind (steady + gusts) in a 6x1 vector and, R_2 is the rotation matrix described by

$$R_2 = \begin{bmatrix} 1 & \sin\phi\tan\theta & \cos\phi\tan\theta \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi\cos\theta & \cos\phi\sec\theta \end{bmatrix} \tag{12}$$

TABLE 1 SIMULATED AIRSHIP PHYSICAL PROPERTIES

Term	Value	Term	Value
m	1.66 kg	I_x	0.1 m^2.kg
V	1.422 m^3	I_y	0.3 m^2.kg
D	0.86 m	I_z	0.3 m^2.kg
$d_{g,z}$	0.533 m	k_1	0.1069
$d_{p,y}$	0.533 m	k_2	0.8239
$d_{p,z}$	0.533 m	K`	0.5155

III. BACKSTEPPING CONTROLLER DESIGN FOR TRAJECTORY TRACKING

Backstepping approach is recursive procedure, which allows deriving control law for a nonlinear system, based on appropriate Lyapunov function candidate that guarantees the designed control system be asymptotically stable and robust to some unmatched uncertainties [28]. The complex nonlinear system is broken down into smaller subsystems and the Lyapunov function is applied for each of the subsystems. The backstepping has an advantage in case of applying to multi input multi output systems since the control law is derived in few steps [29]. The tracking controller is designed in this section using the backstepping technique.

Let us define the first backstepping variable e_1 and desired state x_d then the tracking error variable is

$$\mathrm{e}_1 = \mathrm{x}_\mathrm{g} - \mathrm{x}_\mathrm{d} \tag{13}$$

The virtual control is chosen by introducing the second backstepping variable e_2.

$$\dot{\mathrm{x}}_\mathrm{g} = J\,\dot{\mathrm{x}}_\mathrm{v} = \mathrm{e}_2 + \alpha_1 \tag{14}$$

where α_1 is a stabilizing function. The time differentiation of equation (13) and using equation (14) gives

$$\dot{\mathrm{e}}_1 = \dot{\mathrm{x}}_\mathrm{g} - \dot{\mathrm{x}}_\mathrm{d} = J\,\dot{\mathrm{x}}_\mathrm{v} - \dot{\mathrm{x}}_\mathrm{d} = \mathrm{e}_2 + \alpha_1 - \dot{\mathrm{x}}_\mathrm{g} \tag{15}$$

Let us consider the Lyapunov function candidate V_1 for the first step to be,

$$V_1 = \frac{1}{2}\mathrm{e}_1^{\mathrm{T}}\mathrm{e}_1 \tag{16}$$

and its time derivative is

$$\dot{V}_1 = \mathrm{e}_1^{\mathrm{T}}\dot{\mathrm{e}}_1 \tag{17}$$

To make $\dot{V}_1$ negative definite with respect to e_1, the stabilizing function can be selected as

$$\alpha_1 = \dot{\mathrm{x}}_\mathrm{d} - K_1\mathrm{e}_1 \tag{18}$$

Where $K_1 = K_1^T > 0$ is a 6X6 positive definite design matrix. By substituting equations (15) and (18) in equation (17)

$$\dot{V}_1 = \mathrm{e}_1^{\mathrm{T}}\mathrm{e}_2 - \mathrm{e}_1^{\mathrm{T}}K_1\mathrm{e}_1 \tag{19}$$

Equations (14) and (15) are rearranged as

$$\dot{\mathrm{e}}_1 = \mathrm{e}_2 - K_1\mathrm{e}_1 \tag{20}$$

$$\mathrm{e}_2 = J\,\dot{\mathrm{x}}_\mathrm{v} - \dot{\mathrm{x}}_\mathrm{d} + K_1\,\mathrm{e}_1 \tag{21}$$

Let us consider the second Lyapunov function candidate as

$$V_2 = V_1 + \frac{1}{2}\mathrm{e}_2^{\mathrm{T}}\mathrm{e}_2 \tag{22}$$

Its time derivative is

$$\dot{V}_2 = \mathrm{e}_1^{\mathrm{T}}\dot{\mathrm{e}}_1 - \mathrm{e}_1^{\mathrm{T}}K_1\mathrm{e}_1 + \mathrm{e}_2^{\mathrm{T}}\dot{\mathrm{e}}_2 \tag{23}$$

Taking the time derivative of e_2 and substituting $\dot{\mathrm{x}}_\mathrm{v}$ from equation (1) to equation (23)

$$\dot{\mathrm{e}}_2 = \dot{J}\,\dot{\mathrm{x}}_\mathrm{v} + J\,\ddot{\mathrm{x}}_\mathrm{v} - \ddot{\mathrm{x}}_\mathrm{d} + K_1\,\dot{\mathrm{e}}_1 \tag{24}$$

$$\dot{V}_2 = \mathrm{e}_1^{\mathrm{T}}\mathrm{e}_1 - \mathrm{e}_1^{\mathrm{T}}K_1\mathrm{e}_1 + \mathrm{e}_2^{\mathrm{T}}(\dot{J}\,\dot{\mathrm{x}}_\mathrm{v} + J\,\mathbf{M}^{-1}[-\mathbf{D}(\dot{\mathrm{x}}_\mathrm{v})] - \ddot{\mathrm{x}}_\mathrm{d} + K_1\dot{\mathrm{e}}_1) \tag{25}$$

$\dot{V}_2$ is negative definite in e_1 and e_2, if the terms between brackets are equal to $(-e_1 - K_2 e_2)$

where $K_2 = K_2^T > 0$ is another 6X6 positive definite design matrix. Finally, the desired control force, **U**, can be found from equation (25)

$$\mathbf{U} = \mathbf{M}J^{-1}(-K_2 e_2 - e_1 - \dot{J}\dot{x}_v + \ddot{x}_d - K_1 e_2 + K^2{}_1 e_1) - [-\mathbf{D}(\dot{x}_v)] \quad (26)$$

Substitution of equation (26) into (25) we get

$$\dot{e}_2 = -e_1 - K_2 e_2 \quad (27)$$

and the time derivative of second Lyapunov candidate function becomes

$$\dot{V}_2 = -e_1^T K_1 e_1 - e_2^T K_2 e_2 \quad (28)$$

Second Lyapunov function $\dot{V}_2$ can be made negative definite by choosing the positive definite matrices K_1 and K_2. Thus, according to Lyapunov stability theory, the airship system is globally asymptotically stable.

The control allocation problem with aerodynamic forces and moments can be formulated according to the control law of equation (26) and equation (2).

$$\mathbf{U} = \begin{bmatrix} F^T & M^T \end{bmatrix}^T \quad (29)$$

where the external forces $F = [u_1\ u_2\ u_3]^T$, the external moments $M = [u_4\ u_5\ u_6]^T$ and $(^T)$ is the transpose.

The real control inputs are $[s_g, T_R, T_L, \gamma]^T$. In this paper, the variables T_R, T_L, γ are considered constant in order to study the effect of the gondola position on the altitude control. In addition, the desired roll and yaw angles are chosen here to be equal to zero and the airship is symmetric about the x-z plane.

IV. Numerical Simulation

The numerical simulation of the airship was carried out to illustrate the effectiveness of the designed controller for tracking trajectories in the presence of uncertainty. The dynamic simulation was developed in the Matlab/Simulink environment to implement the nonlinear dynamics model discussed in Section II. The desired trajectory had to be generated first in order to test the performance of the controller. The desired trajectory in x-z plane is shown in Fig. 3. The results were obtained for two cases, no wind case and windy case.

The trajectory is generated with no wind, no temporal changes in pressure or temperature, and changes in altitude driven slowly by the changes in air density to illustrate the effectiveness of the designed controller for tracking trajectories in main flight trajectory modes (ascent, cruise flight, descent). The backstepping controller is used to control the airship altitude using the moving gondola as a control input.

The controller gains are selected based on a trial and error procedure after getting the desired closed-loop response ($K_1 = 10 \times I_{6x6}$, $K_2 = 22 \times I_{6x6}$). In the simulation process, the thrusters were fixed to $T_R = T_L = 0.5$ N and the vectoring thrust angle equal to zero. The initial position of CV of the airship is assumed to be at the sea level, $(x_g, y_g, z_g) = (0, 0, 0)$ m in the inertial reference frame and the initial desired roll, pitch and yaw angles are chosen to be equal to zero ($\phi = \theta = \psi = 0$). The simulation was run for 75 seconds starting at initial velocity $v_o = 2$ m/s and $s_g = 0$, when the airship is leveled, the CG is located directly below the CV.

The closed loop of the control system is shown in Fig. 4 where the real control inputs is calculated from the control command ***U***. A rate limiter and output saturation were added to the closed-loop system. A maximum gondola repositioning rate of $\dot{s}_g = 0.3$ m/s and a limit of $-0.8 < s_g < 1.4$ m are applied to respect the geometry and vehicle capability to avoid saturation. In addition, to avoid singularity in the model at $\pm \pi/2$, the quaternions are used. The 2D reference trajectory and the tracking result are shown in Fig. 3. The blue line represents the reference trajectory and the red line represents the tracking result of the proposed controller. The reference trajectory was filtered to smoothen the corners and avoid the aggressive turns similar to [30]. The Fig. 3. shows that the proposed backstepping controller follows the reference trajectory with small error. The tracking errors of altitude is shown in Fig. 7. The errors are due to aggressive change in the altitude and the time of the gondola to move from the initial position to reach the required position as shown in Fig. 6. During the ascent, the gondola moves backward to generate a positive pitch angle with the rate of 0.3 m/s and reaches $s_g = -0.8$ m after 6 seconds. The results of the pitch responses to movement of the moving gondola in degree are shown in Fig. 5. The airship ascent to the altitude of 50 m with the pitch angle about 80° when the gondola is at -0.8 m and then the controller will update the output to maintain the height of 50 m with the necessary pitch angle for about 45 seconds. The gondola moves back to $s_g = 0$ m to generate pitch angle about 0° as shown in Fig. 5 and Fig. 6. Then the airship descends vertically in the last mode of flight. In this stage, the controller changes the position of gondola forward to generate a negative pitch angle in order to follow the trajectory. The gondola moves forward to about 1.35 m on the curved rail to generate a pitch angle of -88°.

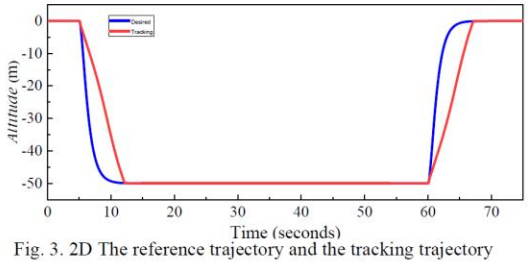


Fig. 3. 2D The reference trajectory and the tracking trajectory

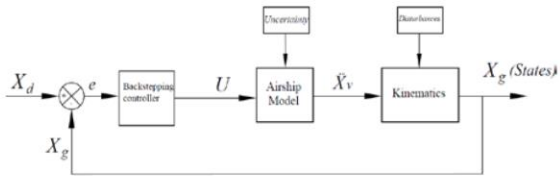


Fig. 4. Block diagram of control system

213

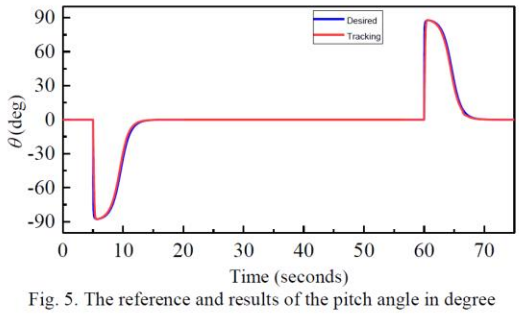


Fig. 5. The reference and results of the pitch angle in degree

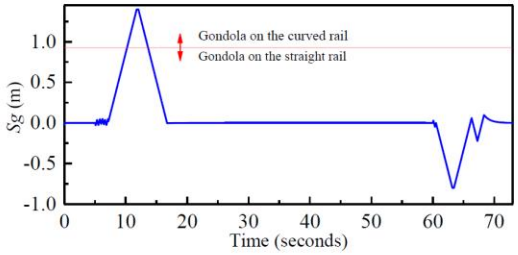


Fig. 6. Results of gondola's position

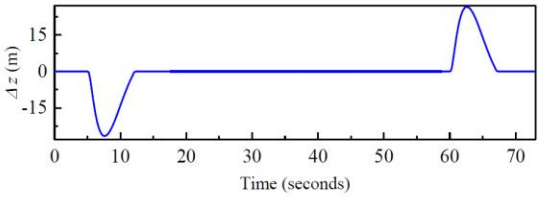


Fig. 7. Tracking error of the altitude

Fig. 6 shows a smooth transition of the gondola from the straight rail to the curved rail until the pitch attitude of -88° is reached. The results show the effectiveness of the proposed backstepping control approach to tracking control of an autonomous airship in the presence of the parametric uncertainties. The results indicated that the vehicle is capable of pitching -88° and 80° by moving the gondola position to front s_g = 1.35 m and rearmost s_g = −0.8 m positions respectively.

In order to evaluate the controller robustness in the presence of wind disturbances v_w, the model was simulated included a turbulent gust with an intensity of 2 m/s, in addition to a wind gust with gust amplitude of 2 m/s. this represent 33% of the maximum speed of the airship. The turbulence was generated by passing white noise through a filter to create Dryden model. The sum vector of wind velocity v_w was added to the simulation (equation (11)) in the velocity components in x and z directions and assuming effect of the wind in y direction was zero. The simulation results of the pitch angle of the airship in the presence of the wind disturbances is shown in Fig. 8. Fig. 9 shows the gondola's position with time in the presence of the wind disturbance, and the figure shows small oscillations during horizontal flight and changes in altitude. The oscillation generated when the airship travels under influence of wind gust. The oscillations of the pitch angle θ that can be observed have been generated during the ascent and the descent. We can observe an oscillation in the pitch angle is between -0.03 rad and + 0.14 rad. The oscillation in Fig. 9 indicate that the controller is changing the gondola's position to adjust the pitch angle in the present of the wind disturbances. The tracking trajectory agreed with the reference trajectory adequately with a small offset.

The results shows that during the airship ascent and descent, the controller output reaches its maximum limit to overcome the cross wind, and during the horizontal flight, the airship is experiencing the destabilizing Munk moment. In landing, the airship attempts to follow the vertical trajectory, however, pitching down to -88° increases the exposed surface area to the wind, which increases the drag, force. This explains the oscillations during the different flight periods.

The results show the robustness of the proposed control and the airship remains stable even in the presence of unknown wind disturbance.

The simulation results show that the displacement and the speed of the gondola are the primary factors affecting attitude control capability however, the moving-mass control is independent of velocity of the airship.

V. Conclusion

Accurate and rapid landing are one of the major limitations of airships, the increase of the pitch angles of new configuration can address these limitations. This improvement in performance justifies the development of a control strategy to preform autonomous flight and experimental validation of these results.

The backstepping approach was applied and the stability analysis for the proposed control law was carried out by using the Lyapunov theory. The simulation results show that the proposed vehicle architecture allows for large pitch variations and rapid changes in altitude at any vehicle velocity. The vehicle was capable of producing changes in the vehicle's pitch of 80° and -88° when the airship was ascending and descending, respectively. This configuration could potentially facilitate autonomous landings or payload deliveries in larger vehicles by ensuring that the gondola reaches the ground before the envelope in all vehicle orientations, and by minimizing the projected area of the lifting gas envelope near the gondola.

The future work is to perform MIMO of pitch and altitude with moving gondola and vector thrust and perform the experimental test of the controller on the actual airship.

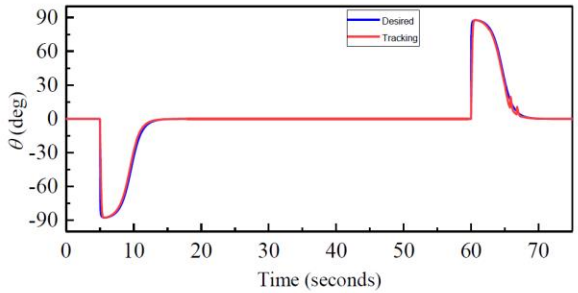


Fig. 8. The reference and results of the pitch angle in degree

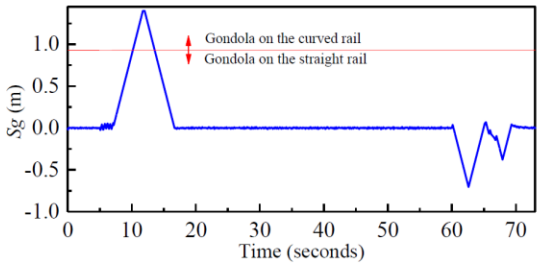


Fig. 9. Results of gondola's position in windy case

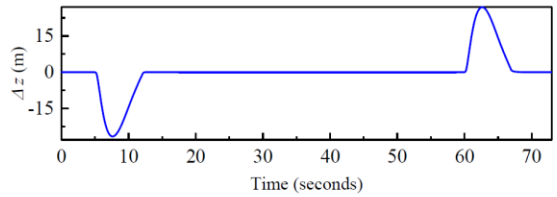


Fig. 10. Tracking error of the altitude in windy case

REFERENCES

[1] J. DELAURIER, "Airship dynamic stability," *AIAA Light. Syst. Technol. Conf. Proc.*, no. July 11-13, 1979.

[2] Y. Li, M. Nahon, and I. Sharf, "Airship dynamics modeling: A literature review," *Progress in Aerospace Sciences*. 2011.

[3] Y. Wang, G. Zheng, D. Efimov, and W. Perruquetti, "Differentiator application in altitude control for an indoor blimp robot," *Int. J. Control*, vol. 91, no. 9, pp. 2121–2130, 2018.

[4] N. Azouz, Y. Bestaoui, and O. Lemaitre, "Dynamic analysis of airships with small deformations," *Proc. 3rd Int. Work. Robot Motion Control. RoMoCo 2002*, pp. 209–215, 2002.

[5] Y. Bestaoui Sebbane, *Lighter than Air Robots*, vol. 58. Dordrecht: Springer Netherlands, 2012.

[6] A. Moutinho, J. R. Azinheira, E. C. de Paiva, and S. S. Bueno, "Airship robust path-tracking: A tutorial on airship modelling and gain-scheduling control design," *Control Eng. Pract.*, vol. 50, no. November, pp. 22–36, 2016.

[7] A. B. Moutinho, "Modeling and Nonlinear Control for Airship Autonomous Flight," *Thesis PhD*, no. August, p. 178, 2007.

[8] J. R. Azinheira, A. Moutinho, and E. C. De Paiva, "A backstepping controller for path-tracking of an underactuated autonomous airship," *Int. J. Robust Nonlinear Control*, vol. 19, no. 4, pp. 418–441, 2009.

[9] E. Kahale, P. C. Garcia, and Y. Bestaoui, "Autonomous path tracking of a kinematic airship in presence of unknown gust," *J. Intell. Robot. Syst. Theory Appl.*, vol. 69, no. 1–4, pp. 431–446, 2013.

[10] E. Repoulias, Filoktimon Papadopoulos, "Robotic Airship Trajectory Tracking Control Using a Backstepping Methodology," *IEEE Int. Conf. Robot. Autom.*, pp. 188–193, 2008.

[11] T. Liesk, M. Nahon, and B. Boulet, "Design and experimental validation of a nonlinear low-level controller for an unmanned fin-less airship," *IEEE Trans. Control Syst. Technol.*, vol. 21, no. 1, pp. 149–161, 2013.

[12] G. Murguía-Rendon, H. Rodríguez-Cortés, and M. Velasco-Villa, "Trajectory tracking control for the planar dynamics of a thrust vectored airship," *Midwest Symp. Circuits Syst.*, pp. 329–332, 2009.

[13] D. Han, X. Wang, L. Chen, and D. Duan, "Command-filtered backstepping control for a multi-vectored thrust stratospheric airship," *Trans. Inst. Meas. Control*, vol. 38, no. 1, pp. 93–104, 2015.

[14] L. Beji, A. Abichou, and Y. Bestaoui, "Stabilization of a nonlinear underactuated autonomous airship-a combined averaging and backstepping approach," *Proc. 3rd Int. Work. Robot Motion Control. RoMoCo 2002*, pp. 223–229, 2002.

[15] P. Morin and C. Samson, "Application of Backstepping Techniques to the Time-Varying Exponential Stabilisation of Chained Form Systems," *Eur. J. Control*, vol. 3, no. 1, pp. 15–36, Jan. 1997.

[16] F. Wang, J. Zhou, and J. Miao, "Pitch handling characteristics of the stratospheric airship based on moving mass control," *Chinese Control Conf. CCC*, vol. 2016-Augus, pp. 10915–10920, 2016.

[17] P. K. Menon, G. D. Sweriduk, E. J. Ohlmeyer, and D. S. Malyevac, "Integrated guidance and control of moving-mass actuated kinetic warheads," *J. Guid. Control. Dyn.*, vol. 27, no. 1, pp. 118–126, 2004.

[18] T. Petsopoulos, F. J. Regan, and J. Barlow, "Moving-Mass Roll Control System for Fixed-Trim Re-Entry Vehicle," *J. Spacecr. Rockets*, vol. 33, no. 1, pp. 54–60, 1996.

[19] L. Chen, G. Zhou, X. J. Yan, and D. P. Duan, "Composite Control of Stratospheric Airships with Moving Masses," *J. Aircr.*, vol. 49, no. 3, pp. 794–801, 2012.

[20] E. Lanteigne, A. Alsayed, D. Robillard, and S. G. Recoskie, "Modeling and Control of an Unmanned Airship with Sliding Ballast," *J. Intell. Robot. Syst. Theory Appl.*, vol. 88, no. 2–4, pp. 285–297, 2017.

[21] E. Lanteigne, W. Gueaieb, and R. S. Recoskie, "Unmanned Airship Design with Sliding Ballast : Modeling and Experimental Validation," 2016.

[22] S. Gomes, "An Investigation of the Flight Dynamics of Airships with Application to the YEZ-2A." Cranfield Institute of Technology, p. 326, 1990.

[23] S. Recoskie, "Autonomous Hybrid Powered Long Ranged Airship for Surveillance and Guidance," p. 125, 2014.

[24] H. Lamb, "The inertia coefficients of an ellipsoid moving in fluid," *Br. Aeronaut. Res. Commitee Rep. ...*, no. 623, pp. 3–5, 1918.

[25] M. M. Munk, "Aerodynamics of airships, Aerodynamic Theory." pp. 32–48, 1936.

[26] S. P. Jones and J. D. DELAURIER, "Aerodynamic Estimation Techniques for Aerostats and Airships," *J. Aircr.*, vol. 20, no. 2, pp. 120–126, 1982.

[27] J. B. Mueller, M. A. Paluszekt, and Y. Zhao, "Development of an aerodynamic model and control law design for a high altitude airship," *Collect. Tech. Pap. - AIAA 3rd "Unmanned-Unlimited" Tech. Conf. Work. Exhib.*, vol. 1, pp. 415–431, 2004.

[28] Y. Yang, J. Wu, and W. Zheng, "Positioning Control for an Autonomous Airship," *J. Aircr.*, vol. 53, no. 6, pp. 1638–1646, 2016.

[29] L. Sangjong, H. Lee, W. Daeyeon, and B. Hyochoong, "Backstepping Approach of Trajectory Tracking Control for the Mid-Altitude Unmanned Airship," *AIAA Guid. Navig. Control Conf. Exhib.*, no. August, pp. 1–14, 2007.

[30] D. Gobiha and N. K. Sinha, "Autonomous maneuvering of a stratospheric airship," *2018 Indian Control Conf. ICC 2018 - Proc.*, vol. 2018-Janua, no. Icc, pp. 318–323, 2018.

H.2. Altitude Trajectory Control of An Unmanned Airship Using Incremental Nonlinear Dynamic Inversion

Presentation and full paper in the Proceedings of the Canadian Society for Mechanical Engineering International Congress 2020

Proceedings of the Canadian Society for Mechanical Engineering International Congress 2020
CSME Congress 2020
June 21-24, 2020, Charlottetown, PE, Canada

ALTITUDE TRAJECTORY CONTROL OF AN UNMANNED AIRSHIP USING INCREMENTAL NONLINEAR DYNAMIC INVERSION

Ali Mansur[1], Eric Lanteigne[2]
Mechanical engineering Department, University of Ottawa, Ottawa, Canada

Abstract—This paper presents the application of the incremental nonlinear dynamic inversion (INDI) control approach for the trajectory tracking of an unmanned airship. The proposed vehicle architecture is composed of two main components: a helium envelop with a rigid keel and moving gondola capable of travelling along the keel. This motion significantly alters the location of the centre of gravity and thus the pitch of the airship. The dynamic equations of the airship are derived using the Newton–Euler formulation, and the model was implemented and simulated in Matlab/Simulink to illustrate the effectiveness of the designed controller for translational and longitudinal motions in the presence of wind disturbances.

Keywords-component; unmanned airship; trajectory tracking; incremental nonlinear dynamic inversion

I. INTRODUCTION

Incremental nonlinear dynamic inversion (INDI) control is a variation on the nonlinear dynamic inversion (NDI). The NDI is a nonlinear control method which eliminates the system nonlinearity by means of feedback and results into partly or entirely linearized closed-loop system dynamics, where conventional linear control techniques can then be applied. The INDI method retains the high-performance characteristics of NDI while reducing model dependency, increasing robustness and reducing computational complexity [1-2]. The INDI only requires knowledge of the system kinematics and the actuator dynamics. The dynamic model is replaced with measurements of the system dynamics [3]. The INDI controller solves the incremental form of equations of motion using acceleration feedback and generates a control law substantially reducing the controller's dependence on complex, and sometimes inaccurate, models such as those describing the aerodynamic phenomena in flight dynamics. Feedback of angular accelerations eliminates the sensitivity to model mismatch, greatly increasing the performance of the system compared with conventional nonlinear dynamic inversion [4]. The incremental nonlinear dynamic inversion method has been successfully used to control various aerospace systems and shown desirable robust performance to aerodynamic model uncertainties [4–8]. The concept of INDI was first suggested in [9]. They examined the behaviour of NDI control law on the VAAC Harrier. During flight test a poor quality angular accelerations signals were observed and filtering and differentiating the angular rates led to oscillatory closed-loop behaviour. They suggested that the preferred solution would be to use an independent angular acceleration signals. An extension of Smith's approach to reformulate the dynamic inversion to be less dependent on the onboard model was developed in [10]. They chose two different sources to get independent data of the state acceleration vector: linear accelerometers and numerically differentiated angular rates. Angular acceleration data was derived from a unique implementation of linear accelerometer measurements, in addition to the differentiated angular rates. Recently, the INDI methodology has been studied at Delft University of Technology [4]. INDI was adopted to control the attitude of a fixed-wing aircraft and, when compared to NDI, demonstrated improved robustness to model uncertainties. More recently INDI control theory was used to achieve a complete quadrotor controller in [11]. INDI showed improvements with the control of a time varying quadrotor model. The INDI controller was able to maintain the quadrotor trajectory while performing some degree of aggressive manoeuvres.

INDI control was applied to the lateral control of an under actuated airship with uncertain dynamics model [5]. The stability and robustness of the proposed sensor-based control solution was illustrated with representative simulation results, including wind disturbances and a path following loop. The INDI controller performance was evaluated by adapting the DRONI's geometry with four-propellers configuration of the AS800 of [12]. The controller was successfully able to follow the path with a maximum error of 10 m in the presence of wind disturbances.

The derivations of INDI control law in literature are based on the time scale separation principle, which considers that the controls can change significantly faster than the states when the sampling frequency is high [4–8]. The time scale separation principle uses the fact that the time constants of the inner and outer loops are different and hence the control laws for inner and outer loops can be designed independently. The closed-loop system stability of a general linear system controlled by INDI was investigated in [13]. The analytical stability analysis showed that implementing discrete-time INDI with a smaller sampling time resulted in larger stability margins regarding system characteristics and controller gains.

The reduction of the noise is essential in the development of an INDI controller. Several approaches were taken into consideration to address this issue. A first order filter used in [14] to reduce the effect of noise and time-delays. Veld proposed three filters to compensate for sensor noise: a first-order low-pass filter, a second-order low-pass filter and two low-pass filters [1]. A linear predictive filter was used to predict the angular accelerations, solve the time delay and angular acceleration availability problem [4]. Other methods include improvements in the indirect measurement of angular accelerations by using Recurrent Neural Networks [15].

INDI is also sensitive to the system sampling frequency [16–17]. It was observed that the system becomes unstable when the sampling frequency falls below 50 Hz leading to a higher overshoot, noticeable ringing and longer settling time [16]. High sampling rates can compensate for model uncertainties and simplifications. The results in [6] showed that the INDI controller was robust to uncertainties in the control matrix, with a sampling rate above 50 Hz.

Signal delay is third main challenges in INDI control. Examples of delays include the differentiation of angular rates for the computation of the angular accelerations and the filters needed to decrease noise in the linear acceleration measurements. This is particularly important in the former as differentiation amplifies the sensor noise [18]. A solution to prevent performance degradation due to the time delay are to synchronize the time delays of all signals by applying the same filter to the angular rate measurements [8]. In addition, Pseudo control hedging can be used to reduces the magnitude of the commanded signals to a level achievable by the saturated controller [13].

In this paper, the INDI methodology is applied to the trajectory tracking of the vehicle architecture shown in Fig. 1. The controller developed in [5] was modified for longitudinal motion, and the effect of the moving gondola for rapid altitude changes for ascending or landing an airship was studied.

II. Airship Architecture and Nonlinear Dynamic Model

A. Airship Architecture

The vehicle architecture is shown in Fig. 1. It is comprised of a semi-rigid helium envelope and a moving gondola. The gondola moves on a rail fixed to the keel of the helium envelope. The rail is strait along the mid-section of the vehicle and circular along the bow. This allows the thruster to remain in line with the centre of volume (CV) of the airship at all times. This vehicle configuration provides an alternative solution to over-actuation and ballonets for the rapid altitude changes required when landing or avoiding obstacles. The physical properties of the airship are listed in Table 1. The airship is 3 m long and the gondola can travel along the keel from s_g = -0.81 m to s_g = 0.96 m along the straight section and from s_g = 0.96 m to s_g = 1.40 m along the circular section, where s_g = 0 m is defined as the gondola position directly below the CV.

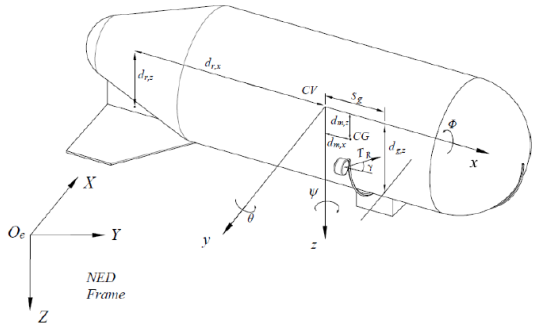


Figure 1. Body axis convention and inertia axis of the airship

B. Nonlinear Dynamic Model

The nonlinear dynamic model for the proposed architecture was derived using the Newton-Euler method similar to [19] and [20]. Two frames of reference were defined to describe the airship motion; the inertial North-East-Down (NED) reference axis system fixed on the earth and the body axis system fixed to the airship CV. The orientation of the body axis system is represented by Euler angles (roll ϕ pitch θ and yaw ψ).

The model's reference system is shown in Fig. 1 and its equation of motion is defined as,

$$\mathbf{M}\ddot{\mathbf{x}}_v + \mathbf{D}(\dot{\mathbf{x}}_v) = \mathbf{E} + \mathbf{U}, \tag{1}$$

where, **M** is the 6x6 mass matrix, **D** is the 6x1 dynamics vector, **E** is the 6x1 model uncertainty vector, $\mathbf{x}_v$ is the 6x1 state vector, the vector of external forces and moments **U** is further defined as,

$$\mathbf{U} = \mathbf{A}(\dot{\mathbf{x}}_v) + \mathbf{G}(\boldsymbol{R}_1) + \mathbf{F}_u, \tag{2}$$

where, **A** is the 6x1 aerodynamic vector, **G** is the 6x1 gravitational and buoyancy vector, $\boldsymbol{R}_1$ is the directional cosine matrix, and $\mathbf{F}_u$ is the 6x1 propulsive force vector. The complete derivation of the terms in (1) and (2) are provided in [20]. The state convention **x** is written as,

$$\mathbf{x} = \begin{bmatrix} x & y & z & \phi & \theta & \psi \end{bmatrix}^T, \quad \dot{\mathbf{x}} = \begin{bmatrix} V & \omega \end{bmatrix}^T,$$

$$V = \begin{bmatrix} \dot{x} & \dot{y} & \dot{z} \end{bmatrix}^T, \quad \omega = \begin{bmatrix} \dot{\phi} & \dot{\theta} & \dot{\psi} \end{bmatrix}^T,$$

where V is vector of linear velocities and ω is angular velocities of the vehicle defined in the moving frame of reference located at the CV of the helium envelope.

$$\boldsymbol{R}_1 = \begin{bmatrix} c\theta c\psi & s\phi s\theta c\psi - c\phi s\psi & c\phi s\theta c\psi + s\phi s\psi \\ c\theta s\psi & s\phi s\theta s\psi + c\phi c\psi & c\phi s\theta s\psi - s\phi c\psi \\ -s\theta & s\phi c\theta & c\phi c\theta \end{bmatrix} \tag{3}$$

The rotation matrix $\boldsymbol{R}_1$ (also known as the 123 or roll-pitch-yaw Euler sequence),

The mass of the airship (m) is equal to the sum of the mass of the gondola (m_g), the rail (m_r), the envelope (m_e), the motors (m_m) and the fins (m_f). The centre of mass (CM) of the airship is calculated as,

$$d_{m,x} = \frac{m_g \times s_g + (2 \times m_m \times x_m)}{m}$$
$$d_{m,z} = \frac{m_g \times z_g + m_r \times L_r + m_f \times L_f}{m} \quad (4)$$

where $d_{m,x}$ and $d_{m,z}$ are the vertical distances between CM and CV along x and z-axes respectively, and

$$\left.\begin{aligned} z_g &= d_{p,z} \\ x_m &= s_g \end{aligned}\right\} \quad -0.81 \le s_g \le 0.96$$
$$\left.\begin{aligned} z_g &= d_{p,z} \times cos\zeta \\ x_m &= s_g - \left(d_{p,z} \times sin\zeta\right) \end{aligned}\right\} \quad s_g > 0.96$$

where z_g is the distance between the CV to the CM of the gondola along z, x_m is distance between the CV to CM of the propeller along x, ζ is the angle between the link and z axis when the gondola is in the curved link. L_f and L_r are the distance between the CV to the CM of the rail and fin along z respectively. The masses and the inertias of the fixed components as indicated by the subscript f represent those of the envelope and rail and were found based on the dimensions of the prototype solid model.

All other variables and coefficients are summarized in [20], and including references to where they can be determined for any airship platform using a combination of semi empirical models and geometric functions.

C. *Kinematics*

Vehicle kinematics were applied in tandem to determine the velocity and positional changes and how they translate to the earth reference frame. Coordinates and trajectories in the earth reference frame can be determined from the following equation,

$$\dot{\mathbf{x}}_\mathbf{g} = \mathbf{J}\dot{\mathbf{x}}_\mathbf{v} + \boldsymbol{v}_\mathrm{w} = \begin{bmatrix} \boldsymbol{R_1} & 0_{3x3} \\ 0_{3x3} & \boldsymbol{R_2} \end{bmatrix} \dot{\mathbf{x}}_\mathbf{v} + \boldsymbol{v}_\mathrm{w} \quad (5)$$

where v_w is the total wind (steady + gusts) in a 6x1 vector and, $\boldsymbol{R_2}$ is the rotation matrix described by

$$\boldsymbol{R_2} = \begin{bmatrix} 1 & \sin\phi\tan\theta & \cos\phi\tan\theta \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi\cos\theta & \cos\phi\sec\theta \end{bmatrix} \quad (6)$$

III. Incremental Nonlinear Dynamics Inversion Approach

Instead of computing the total control input directly, the INDI methodology calculates the increment of the control input for every sample time based on the system states at the previous time step. This concept has been used for the purpose of coping with nonlinear control derivatives, and to reduce the impact of model mismatch [4]. Given a general nonlinear system:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{G}(\mathbf{x})\mathbf{u} \quad (7)$$

where $\mathbf{x} \in \mathbb{R}^n$ is the state vector, $\mathbf{u} \in \mathbb{R}^m$ is the input, $\mathbf{G} \in \mathbb{R}^{n \times m}$ is the control effectiveness matrix, the Taylor series expansion provides the following first-order approximation of $\dot{\mathbf{x}}$ at the previous state and control $\mathbf{x}_0$, $\mathbf{u}_0$ respectively,

$$\dot{\mathbf{x}} = \mathrm{f}(\mathbf{x}_0) + \mathrm{G}(\mathbf{x}_0)\mathbf{u}_0 + \frac{\partial}{\partial \mathrm{x}}\left[\mathrm{f}(\mathbf{x}) + \mathrm{G}(\mathbf{x})\mathbf{u}\right]\Big|_{\substack{\mathrm{x}=\mathrm{x}_0 \\ \mathrm{u}=\mathrm{u}_0}} (\mathbf{x} - \mathbf{x}_0)$$
$$+ \frac{\partial}{\partial \mathrm{u}}\left[\mathrm{G}(\mathbf{x})\mathbf{u}\right]\Big|_{\substack{\mathrm{x}=\mathrm{x}_0 \\ \mathrm{u}=\mathrm{u}_0}} (\mathbf{u} - \mathbf{u}_0) + \mathrm{H.O.T} \quad (8)$$

Assuming a high sampling frequency, the changes of state variation $\mathbf{x} \approx \mathbf{x}_0$ of equation (8) and (H.O.T) the higher order terms can be neglected. The dynamics of the previous state $\dot{\mathbf{x}}_0$ are:

$$\dot{\mathbf{x}}_0 = \mathrm{f}(\mathbf{x}_0) + \mathrm{G}(\mathbf{x}_0)\mathbf{u}_0 \quad (9)$$
$$H_u = \frac{\partial}{\partial \mathrm{u}}\left[\mathrm{G}(\mathbf{x})\mathbf{u}\right]\Big|_{\substack{\mathrm{x}=\mathrm{x}_0 \\ \mathrm{u}=\mathrm{u}_0}}$$

where H_u is the partial derivative of $[\mathrm{G}(\mathbf{x})\mathbf{u}]$ at $\mathbf{x} = \mathbf{x}_0$ and $\mathbf{u} = \mathbf{u}_0$. The incremental form of the dynamic equation is thus simplified as,

$$\dot{\mathbf{x}} = \dot{\mathbf{x}}_0 + H_u(\mathbf{u} - \mathbf{u}_0) \quad (10)$$

Equation (10) is a linear approximation of the equation of motion around $\mathbf{u}_0$ and $\mathbf{x}_0$ for small time increments, describing the changes in accelerations as a function of control increments. By setting $\dot{\mathbf{x}}$ as the virtual control variable $\boldsymbol{v}_r$, the controller can then be designed in the incremental form as,

$$\mathbf{u} = \mathbf{u}_0 + H_u^{-1}(\boldsymbol{v}_r - \dot{\mathbf{x}}_0) \quad (11)$$

where $\dot{\mathbf{x}}_0$ is assumed to be measurable and the input control matrix H_u is assumed to be invertible. The total input can be obtained by adding the previous input $\mathbf{u}_0$ to the calculated increment

$$\mathbf{u} = \mathbf{u}_0 + \Delta\mathbf{u}$$

It is important to notice that this implicit control law is not entirely independent on the model since changes in $\mathrm{f}(\mathbf{x})$ are reflected in measurements of $\dot{\mathbf{x}}_0$, improving robustness against model uncertainties contained therein. Therefore, this implicit control law is dependent on accurate measurements of the accelerations and the control input.

The acceleration reference for the INDI controller $\boldsymbol{v}_r$ is obtained as follow. The position error is calculated from the difference between the reference trajectory and the feedback of the vehicle's position, and then the desired velocity towards a goal position is calculated by a proportional controller by multiplying the position error with a gain.

$$\dot{\boldsymbol{x}}_d = (\boldsymbol{x}_d - \boldsymbol{x})\mathrm{k}_{\mathrm{px}}$$

Finally, the velocity error, resulting from subtracting the desired velocity from the feedback velocity of the vehicle, is

then multiplied with a gain to calculate the acceleration reference.

$$\boldsymbol{v}_r = (\dot{\boldsymbol{x}}_d - \dot{\boldsymbol{x}})\,k_{pv} \tag{12}$$

where k_{px}, k_{pv} are the proportional gains of the position and velocity vectors, respectively. The proportional control is used to correct for errors in position and the velocity and provides the acceleration reference for the INDI controller. Then equation (11) becomes,

$$\mathbf{u} = \mathbf{u}_0 + H_u^{-1}\left(\left[(\boldsymbol{x}_d - \boldsymbol{x})k_{px} - \dot{\boldsymbol{x}}\right]k_{pv} - \dot{\mathbf{x}}_0\right) \tag{13}$$

The equation shows that the dependency of the closed-loop system on the accurate knowledge of the airship model is largely decreased, improving robustness against model uncertainties. Therefore, the changes in $f(\mathbf{x})$ are reflected in $\dot{\mathbf{x}}_0$, and the control mainly required the measurements of $\dot{\mathbf{x}}_0$ and $\mathbf{u}_0$, making this control approach more dependent on the sensor measurements.

Different filtering approaches have been used to solve the problem of indirect measurement of the accelerations. Usually, the acceleration in $\dot{\mathbf{x}}_0$ is not measured directly and the sensors are not widely available, especially for small-unmanned aircrafts [11]. According to [8] the acceleration $\dot{\mathbf{x}}_0$ can be calculated by passing the velocity component of $\dot{\boldsymbol{x}}$ through a second order low pass filter.

A. *Translational Control Loop*

In translational control loop, the vehicle's flight position is controlled by the vectored thrust $\mathbf{F}_u$ to track the reference trajectory. The vectored thrust is defined as,

$$\mathbf{F}_u = \begin{bmatrix} F_x \\ F_z \end{bmatrix} = \begin{bmatrix} T_R\, cos\gamma_R + T_L cos\gamma_L \\ -T_R sin\gamma_R + T_L sin\gamma_L \end{bmatrix} \tag{14}$$

where F_x, F_z are the components of the thrust in body fixed frame in the x and z axes, respectively. T_R and T_L are the right and left propeller thrust, and γ_R and γ_L the right and left angles of the thrusters. The vectored thrust is considered as the control variable $\mathbf{u}_F$ for the INDI controller.

$$\mathbf{u}_F = \begin{bmatrix} F_x \\ F_z \end{bmatrix},\ g_1(\mathrm{x}) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix},\quad \text{thus} \qquad \boldsymbol{F}_u = g_1(\mathbf{x})\ \mathbf{u}_F$$

Equation (1) can be rewritten into the following control affine form of (7) for the linear motion,

$$\ddot{\mathbf{x}}_v = \mathbf{M}^{-1}[\mathbf{A}(\dot{\mathbf{x}}_v) + \mathbf{G}(\boldsymbol{R}_1) - \mathbf{D}(\dot{\mathbf{x}}_v)] + \mathbf{M}^{-1}\boldsymbol{F}_u$$

and $$f(\mathbf{x}_v) = \mathbf{M}^{-1}[\mathbf{A}(\dot{\mathbf{x}}_v) + \mathbf{G}(\boldsymbol{R}_1) - \mathbf{D}(\dot{\mathbf{x}}_v)],\quad H_{u1} = \mathbf{M}^{-1}\, g_1(\mathbf{x})$$

TABLE I. AIRSHIP PHYSICAL PROPERTIES

Term	Value	Term	Value
m	1.66 kg	$d_{p,z}$	0.533 m
V	1.422 m^3	I_x	0.1 m^2.kg
D	0.86 m	I_y	0.3 m^2.kg
$d_{g,z}$	0.533 m	I_z	0.3 m^2.kg
$d_{p,y}$	0.533 m		

The incremental form of the translational control is,

$$\mathbf{u}_F = \mathbf{u}_{F0} + \Delta\mathbf{u}_F \tag{15}$$

where $\mathbf{u}_{F0}$ is the previous input. The airship's flight path control can be calculated by using the translational equation of motion and (11). The subscript (v) represents the linear equations of motion used to distinguish between the linear and rotational equations. The linear acceleration $\dot{\mathbf{x}}_{v0}$ can be measured by onboard sensors or by passing the velocity $\dot{\boldsymbol{x}}$ through a second order low pass filter as used in this study. The term $f(\mathbf{x}_0)$ was cancelled in (9) which reduces the controller's dependence on the model accuracy. The linear acceleration reference $\boldsymbol{v}_r$ is obtained from (12), as a state error feedback with constant gains.

B. *Altitude Control Loop*

The objective of this control loop is to control the pitch angle by controlling the gondola's position along the length of the airship keel. The angular rate is controlled by the moments of gravitational and buoyancy vector **G**.

$$\mathbf{G}_M = \begin{bmatrix} G_\phi & G_\theta & G_\psi \end{bmatrix}^T \tag{16}$$

where $\mathbf{G}_M$ is the moment vector of gravitational and buoyancy, G_ϕ and G_ψ are the moment components of gravitational and buoyancy vector around the x axis and z axis and are considered here equal to zero in order to study the effects of the gondola position on the altitude control in the x-z plane. G_θ is the moment component of gravitational and buoyancy vector around the y axis. The gravitational and buoyancy forces applied to the vehicle must be premultiplied by a rotation matrix to convert them to the body reference frame. After some manipulations, the final format of G_θ is,

$$\begin{aligned} G_\theta = {} & [R(3,1)\times g(d_{m,x}(m - m_g) + m_g d_{m,z})] \\ & -[R(3,3)\times g(d_{m,x}(m - m_g) + m_g s_g)] \end{aligned} \tag{17}$$

where $d_{m,x}$ is a function of s_g. The input s_g can be calculated by substituting $d_{m,x}$ in (17) and solving for s_g, the equation (17) can be rewritten as,

$$G_\theta = g_2(\mathbf{x})\, s_g \tag{18}$$

where $g_2(\mathbf{x})$ contains the terms directly related with s_g. The equation shows that the movement of the moving mass s_g is the primary factor affecting the pitching moment. In this loop, the control variable is the gondola's position s_g.

Equation (1) can be rewritten into the following control affine form equation (7) for the angular motion,

$$\ddot{\mathbf{x}}_\omega = \mathbf{M}^{-1}[\mathbf{A}(\dot{\mathbf{x}}_\omega) + \mathbf{F}_u - \mathbf{D}(\dot{\mathbf{x}}_\omega)] + \mathbf{M}^{-1}\mathbf{G}(\boldsymbol{R}_1)$$

and $$f(\mathbf{x}_\omega) = \mathbf{M}^{-1}[\mathbf{A}(\dot{\mathbf{x}}_\omega) + \mathbf{F}_u - \mathbf{D}(\dot{\mathbf{x}}_\omega)],\qquad H_{u2} = \mathbf{M}^{-1}\, g_2(\mathbf{x})$$

The incremental form of the longitudinal control is,

$$s_g = s_{g0} + \Delta s_g \quad (19)$$

The airship's pitch angle control can be calculated by using the rotational equation of motion and (11). The subscript (ω) represents the rotational equations of motion used to distinguish between the linear and rotational equations. The angular acceleration $\dot{\mathbf{x}}_{\omega 0}$ can be measured by onboard sensors or by passing the angular velocity $\dot{\omega}$ through a second order low pass filter. The measured accelerations in both loops were filtered by a second order low pass filter to keep all signals synchronized in control equations and to prevent performance degradation of the controller due to the time delay [8]. The angular acceleration reference $\boldsymbol{\nu}_r$ is obtained from (12), as a state error feedback with constant gains.

IV. NUMERICAL SIMULATION

Numerical simulations were carried out to illustrate the altitude tracking performance of the controller altitude. The dynamic simulation was developed in the Matlab/Simulink environment. Rate limiters and output saturation were added to the closed-loop system to respect the geometry, and actuator and thruster capability and saturation. A maximum gondola repositioning rate of $\dot{s}_g$ = 0.3 m/s was adopted based on open-loop hardware tests on the gearmotors driving the gondola. The propeller thrust was limited to -0.26 N < T< 2.6 N based on thruster static bench testing prototype. The acceleration $\dot{\mathbf{x}}_0$ calculated by passing the velocity $\dot{\boldsymbol{x}}$ through a second order low pass filter with natural frequency $\omega_n = 25$ and damping $\zeta_n = 0.7$ rad/s.

The controller gains were selected heuristically to obtain a desired closed loop response. The position gains and the velocity gains of the translational control are, (K_{xp} =3, K_{zp} =2, K_{xv}=5, K_{zv}=6) and the longitudinal controller gains. (K_θ =8, $K_{\dot{\theta}}$ =2). In the simulation process, the initial position of CV of the airship is (x_g, y_g, z_g) = (0, 0, 0) m in the inertial reference frame and the initial altitude is equal to zero ($\phi = \theta = \psi$=0). T_R and T_L are assumed equal, and $\gamma_R = \gamma_L$. The simulation was run for 75 seconds starting at initial velocity v_0 = 2 m/s.

The INDI controller was evaluated in the presence of wind disturbances $\boldsymbol{\nu}_w$, the model was simulated included a turbulent gust with an intensity of 2 m/s, in addition to a wind gust with gust amplitude of 2 m/s. This represents 33% of the maximum speed of the airship. The turbulence was generated by passing white noise through a filter to create Dryden model [23]. The sum vector of wind velocity $\boldsymbol{\nu}_w$ was added to the velocity components in the x and z directions according to (5). Fig. 2 shows the tracking performance of the proposed controller and Fig. 3 shows the resulting pitch angle. The gondola's position with time is illustrated in Fig. 4. The results indicate the presence of small oscillations during horizontal flight and changes in altitude. The oscillation generated when the airship travels under influence of wind gust can be observed during longitudinal maneuvers. The control is able to limit these oscillations to ±2.9°. The oscillations in Fig. 4 at t = 7 s and t = 62 s are due to the changing of the gondola's position to adjust the pitch angle in the windy condition. The tracking error of the altitude in the presence of wind is shown in Fig. 5. The steady-state error is always produced when using only proportional control. Increasing or decreasing the proportional gain will result in a change in output however, changing the K_P value allows for the rise time and steady state error to be reduced at the cost of increasing overshoot. Therefore, adding integral term has the effect of eliminating the steady-state error. However, it may have a negative effect on the transient response of the control system.

It can be seen in the result that during the airship ascent and descent period, the lift gas is unable to compensate the airship weight and follow the reference trajectory, therefore the thrust control will used to accelerate or decelerate airship up or down result in actuator saturation. In Fig. 2, the thrust components control is saturated when the rise of velocity and position control are required. The gondola actuator reaches the rate limit in the pitch angle control as observed in Fig. 4. The controller saturated during the ascent and reached its limit s_g = –0.8 m and the controller was able follow the reference trajectory with no saturation during the descent.

The results show the robustness of the proposed control while subjected to aggressive manoeuvres and the airship remains stable even in the presence of unknown wind disturbance. The simulation results show that the displacement and the speed of the gondola are the primary factors affecting attitude control capability however, the moving-mass control is independent of velocity of the airship.

V. CONCLUSION

The incremental nonlinear dynamic inversion (INDI) approach was used for trajectory tracking control of an airship using a moving gondola and vectored thrust components as control inputs. The simulation results show that the proposed vehicle architecture allows for large pitch variations and rapid changes in altitude at any vehicle velocity. The vehicle was capable of producing changes in the vehicle's pitch of 80° and -88° when the airship was ascending and descending, respectively. This configuration could potentially facilitate autonomous landings or payload deliveries in larger vehicles by ensuring that the gondola reaches the ground before the envelope in all vehicle orientations, and by minimizing the projected area of the lifting gas envelope near the gondola. The controller was able to track longitudinal motions with uncertain dynamics model. The results show that the robustness of INDI is sensitive to the quality of state measurements.

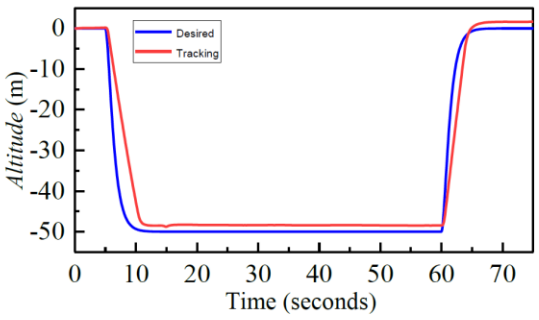


Figure 2. The reference trajectory and the tracking trajectory

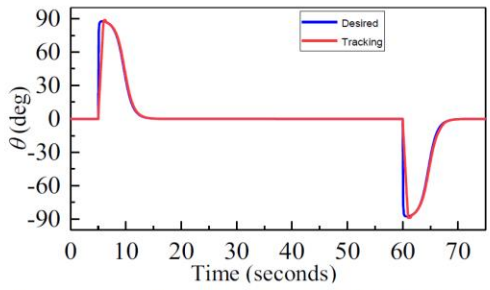


Figure 3. The reference pitch angle and simulated vehicle pitch

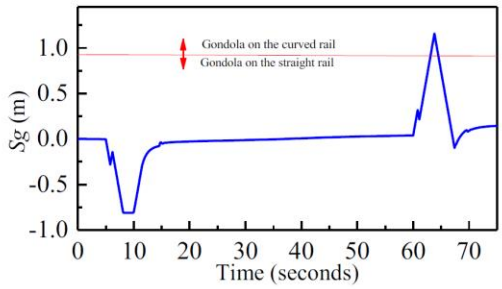


Figure 4. Gondola position

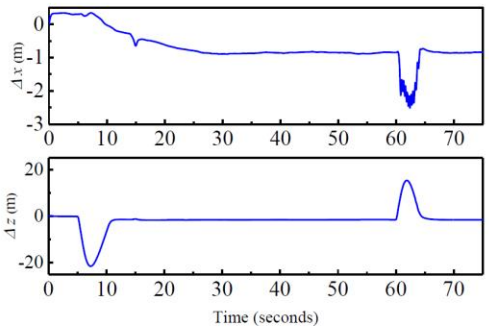


Figure 5. Tracking error

REFERENCES

[1] R. C. van't Veld, "Incremental Nonlinear Dynamic Inversion flight control," Delft University of Technology, 2016.

[2] B. Bacon and A. Ostroff, "Reconfigurable flight control using nonlinear dynamic inversion with a special accelerometer implementation," in *AIAA Guidance, Navigation, and Control Conference and Exhibit*, 2000, no. c, pp. 1–5.

[3] X. Wang, E. J. Van Kampen, Q. Chu, and P. Lu, "Stability analysis for incremental nonlinear dynamic inversion control," *AIAA Guid. Navig. Control Conf. 2018*, no. 210039, 2018.

[4] S. Sieberling, Q. P. Chu, and J. A. Mulder, "Robust flight control using incremental nonlinear dynamic inversion and angular acceleration prediction," *J. Guid. Control. Dyn.*, vol. 33, no. 6, pp. 1732–1742, 2010.

[5] J. R. Azinheira, A. Moutinho, and J. R. Carvalho, "Lateral Control of Airship with Uncertain Dynamics using Incremental Nonlinear Dynamics Inversion," *IFAC-PapersOnLine*, vol. 48, no. 19, pp. 69–74, 2015.

[6] Z. Liu, J. Guo, M. Li, S. Tang, and X. Wang, "VTOL UAV Transition Maneuver Using Incremental Nonlinear Dynamic Inversion," *Int. J. Aerosp. Eng.*, vol. 2018, pp. 1–19, 2018.

[7] E. Tal and S. Karaman, "Accurate Tracking of Aggressive Quadrotor Trajectories Using Incremental Nonlinear Dynamic Inversion and Differential Flatness," *Proc. IEEE Conf. Decis. Control*, vol. 2018-Decem, pp. 4282–4288, 2019.

[8] P. Lu, E. J. van Kampen, C. de Visser, and Q. Chu, "Aircraft fault-tolerant trajectory control using Incremental Nonlinear Dynamic Inversion," *Control Eng. Pract.*, vol. 57, pp. 126–141, 2016.

[9] P. Smith and A. Berry, "Flight test experience of a non-linear dynamic inversion control law on the VAAC Harrier," in *Atmospheric Flight Mechanics Conference*, 2000, no. c, pp. 132–142.

[10] B. J. Bacon, A. J. Ostroff, and S. M. Joshi, "Reconfigurable NDI controller using inertial sensor failure detection & isolation," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 37, no. 4, pp. 1373–1383, 2001.

[11] E. Lima and S. Da Silva, "Incremental Nonlinear Dynamic Inversion for Quadrotor Control," 2015.

[12] E. C. De Paiva, J. R. Azinheira, J. G. Ramos, A. Moutinho, and S. S. Bueno, "Project AURORA: Insfractructure and flight control experiments for a robotic airship," *J. F. Robot.*, vol. 23, no. 3–4, pp. 201–222, 2006.

[13] R. van 't Veld, E.-J. Van Kampen, and Q. P. Chu, "Stability and Robustness Analysis and Improvements for Incremental Nonlinear Dynamic Inversion Control," no. January, 2018.

[14] P. B. Acquatella, W. Falkena, E. J. van Kampen, and Q. P. Chu, "Robust nonlinear spacecraft attitude control using incremental nonlinear dynamic inversion," *AIAA Guid. Navig. Control Conf. 2012*, no. August, 2012.

[15] Y. Huang, D. M. Pool, O. Stroosma, and Q. P. Chu, "Incremental Nonlinear Dynamic Inversion Control for Hydraulic Hexapod Flight Simulator Motion Systems," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 4294–4299, 2017.

[16] R. Coelho, "Quadrotor Attitude Control using Incremental Nonlinear Dynamics Inversion," TECNICO LISBOA, 2017.

[17] Z. Lu, P. Tang, and S. Zhang, "Incremental Nonlinear Dynamic Inversion Based Control Allocation Approach for a BWB UAV," no. August, 2018.

[18] C. Cakiroglu, E. J. Van Kampent, and Q. Chu, "Robust incremental nonlinear dynamic inversion control using angular accelerometer feedback," *AIAA Guid. Navig. Control Conf. 2018*, no. 210039, 2018.

[19] S. Gomes, "An Investigation of the Flight Dynamics of Airships with Application to the YEZ-2A." Cranfield Institute of Technology, p. 326, 1990.

[20] E. Lanteigne, W. Gueaieb, and R. S. Recoskie, "Unmanned Airship Design with Sliding Ballast : Modeling and Experimental Validation," 2016.

[21] H. Lamb, "The inertia coefficients of an ellipsoid moving in fluid," *Br. Aeronaut. Res. Commitee Rep. ...*, no. 623, pp. 3–5, 1918.

[22] M. M. Munk, "Aerodynamics of airships, Aerodynamic Theory." pp. 32–48, 1936.

[23] R. Beard and T. Princeton, *Small Unmanned Aircraft: Theory and Practice*, vol. 36, no. 1. 2013.