Static and Dynamic Modeling and Simulation of the Umbilical Cable in A Tethered Unmanned Aerial System

by

Sina Doroudgar

B.Sc. (Mechanical Engineering), Shiraz University, 2010

Thesis Submitted in Partial Fulfillment of the
Requirements for the Degree of
Master of Applied Science

in the
School of Mechatronic Systems Engineering
Faculty of Applied Sciences

SIMON FRASER UNIVERSITY

Summer 2016

Approval

Name:	**Sina Doroudgar**
Degree:	**Master of Applied Science**
Title:	***Static and Dynamic Modeling and Simulation of the Umbilical Cable in A Tethered Unmanned Aerial System***
Examining Committee:	**Chair:** Krishna Vijayaraghavan Assistant Professor

Siamak Arzanpour
Senior Supervisor
Associate Professor

Carolyn Sparrey
Co-Supervisor
Assistant Professor

Kevin Oldknow
Internal Examiner
Lecturer
School of Mechatronic Systems Engineering

Date Defended/Approved: June 27, 2016

Abstract

The research at hand has been accomplished in collaboration with our industry partner, Rigid Robotics Inc. and aims to predict and suggest solutions for some of the known and unknown issues that might appear in operation of a tethered Unmanned Aerial System (to be referred to as UAS hereafter). In this work, the static and dynamic behaviour of the power cable connecting a hovering UAS to its base station is studied in different flight scenarios. The mathematical modelling of the cable is carried out using catenary equations for the static case and multi-body dynamics principles are employed for the dynamic condition. In the preliminary stages of the project, for the purpose of the cable and UAS design, a simple technique is used to estimate the maximum tension forces present in the static state of the cable as well as the cable shape function and other related parameters when UAS is in hovering mode. The derivation of the system's equations of motion is done using Lagrange's Equations by considering the cable as a discrete multi-body system. The equations of motion are derived for a system of finite segments and are solved numerically using MATLAB™ software package in order to simulate the cable's motion. The effect of wind on the dynamics of the cable is also implemented using theoretical methods and simulations. The system's dynamics is modeled in a planar motion as well as a 3D space in separate chapters.

Keywords: Mathematical Modelling; Cable Dynamics; Tethered Systems; Multi-body Dynamics; Lagrange's Equations; Umbilical Cables

To my lovely wife, Anahita.

To my beloved parents, Javad and Simin.

Acknowledgements

I would like to take this opportunity to thank everyone who helped me directly or otherwise through this process.

Firstly, I would like to express my gratitude towards my senior supervisor, Dr. Siamak Arzanpour, who provided me with this exciting opportunity to pursue my graduate studies in this program under his supervision. He helped me follow through this whole journey by his guidance and advices. His comments and suggestions helped significantly towards composition of this thesis.

Secondly, I would like to thank the members of my supervisory committee, Dr. Carolyn Sparrey, Dr. Kevin Oldknow, as well as the chair of the session, Dr. Krishna Vijayaraghavan for their valuable time kindly put into reviewing this work and making useful comments in a relatively short time.

In addition I would like to thank Natural Sciences and Engineering Research Council of Canada (NSERC) for their scholarship support in form of an Industrial Postgraduate Scholarship. Many thanks go to Mehran Motamed, president and technical director of Rigid Robotics Inc. for their financial and intellectual support during this internship. I am grateful to my friends and colleagues at Rigid Robotics, Thomas Huryn and Morteza Najafvand for their professional support during two years of working at Rigid Robotics.

This acknowledgement would not be complete without thanking my old friend Ehsan Asadi for his continued support and friendship, and my office mates at SFU, Azim Keshtkar, Vahid Zakeri, Soheil Sadeqi, and Shahab Azimi for their professional support. I hope I can be of some help to all of you in future.

I cannot emphasize enough how important the role of my beloved parents is in my professional life. I would like to thank my father and mother who have always dedicated themselves to their family in every aspect of life. I cannot thank them with words for all their efforts, financial and emotional, during every step of my undergraduate and graduate studies.

Last but definitely not the least, I would like to thank my lovely wife, Anahita without whose patience and support, I could not accomplish this work. I would like to thank her for the continued support and encouragement as well as all the challenges and difficulties she has endured during this time; even though, I will never be able to express my gratitude towards her devotions.

Table of Contents

List of Tables

List of Figures

List of Acronyms

2D	Two Dimensional
3D	Three Dimensional
AUV	Autonomous Underwater Vehicle
DOF	Degrees of Freedom
FEA	Finite Element Analysis
GPS	Global Positioning System
NSERC	Natural Sciences and Engineering Research Council of Canada
ODE	Ordinary Differential Equation
ROV	Remotely Operated Underwater Vehicle
SFU	Simon Fraser University
UAS	Unmanned Aerial System
UAV	Unmanned Aerial Vehicle
VTOL	Vertical Take Off and Lading

Nomenclature

Roman

A	Cable’s cross-sectional area, m^2
C_D	Drag Coefficient
C_L	Lift Coefficient
F	Objective/Cost function
F_D	Drag Force, N
F_L	Lift Force, N
F_x	Tangential wind force in the local segment coordinate, N
F_z	Normal wind force in the local segment coordinate, N
f_V	Wind Velocity Frequency, Hz
f_β	Wind Heading Angle Frequency, Hz
g	Gravitational acceleration, m/s^2
G	Cable’s Effective Shear Modulus, Pa
$\bar{I}_{CG}$	Mass Moment of Inertia Matrix, kgm^2
J	Tether Cross Sectional Polar Moment Of Inertia, m^4
L	Cable’s total length, m
l	Cable segment length, m
$\mathscr{L}$	Lagrangian Function, J
$\mathbf{M}$	Mass Distribution Matrix, kg
m_U	UAS’s mass, kg
Q	Generalized force, m/s^2
$\mathbf{Q}$	Generalized force matrix, m/s^2
$\mathbf{R}_{\theta_i}$	Transformation matrix
s	Cable length, m
T_0	Cable’s horizontal tension at vertex, N
T	Longitudinal cable tension, N

T	Kinetic energy, Watt
U	Potential energy, Watt
U	Horizontal Wind Velocity, m/s
V	Lateral Wind Velocity, m/s
V_{w0}	Wind Velocity Vector, m/s
$V_{w,rel}$	Relative wind velocity, m/s
δW	Virtual Work, J
W	Vertical Wind Velocity, m/s
X	Global horizontal coordinate
x	Local (cable segment) tangent coordinate
δx	Virtual Displacement
Y	Global lateral coordinate
y	Local (cable segment) lateral coordinate
Z	Global vertical coordinate
z	Local (cable segment) normal coordinate

Greek

α	Angle of Attack, rad and/or deg
β	Cable's mass per length, Kg/m
γ	UAS Segment Rotation Angle, rad and/or deg
$\delta\phi$	Virtual Angular Displacement (Virtual Rotations)
ζ	UAS State Vector
$\boldsymbol{\eta}$	Attitude Vector
θ	Pitch Angle, rad and/or deg
μ	Cable's weight per unit length, N/m
μ_V	Wind Velocity Magnitude Mean Value, m/s
μ_β	Wind Heading Angle Mean Value, deg
ρ	Cable's density, Kg/m^3

σ_V	Wind Velocity Variance, m/s
σ_β	Wind Heading Angle Variance, deg
ϕ	Roll Angle, rad and/or deg
ψ	Heading Angle, rad and/or deg

Chapter 1.

Introduction

1.1. Research Objective and Motivation

This thesis is focused on analyzing the behavior of a cable used to tether a hovering Unmanned Aerial System (to be referred to as UAS hereafter) to its stationary base station in different operation scenarios. The purpose of the cable is to power the UAS in order to increase its operation duration as well as to communicate data securely. This research was initiated in collaboration with Rigid Robotics Inc. to analyze the tether's static and dynamic effects on the UAS which they intended to design and manufacture.

Flexible cables have been in use for several years in many structures including bridges; electrical power lines, towing and mooring systems, hovering tethered aerostats, etc. For example, a cable might be used for towing an underwater vehicle using a vessel or to connect two flying aircrafts called an aerial tow system [1]. Tethering different mechanical systems are among applications which benefit significantly from the use of cables. These systems are referred to as cable-body systems and are required to satisfy certain physical requirements depending on their application and the environment in which they are operating [1].

Cables are widely used in these applications because of their flexibility and versatility. The fact that cables provide tension while not resisting compression, makes them unique and useful for certain applications. They are also able to mitigate vibrations in comparison to similar components of same mechanical properties.

To better understand the static and dynamic behavior of the system including the cable, its effect on the base station, as well as the tension that the cable exerts on the

UAS, it is important that the system is modeled so different operation scenarios can be simulated and analyzed. By having a reliable dynamic model for the cable-body system, one can investigate the cable tensions in each specific flying condition and achieve design criteria for system components, including the cable, UAS propellers and motors, cable-UAS connection joint, the winch and its motor, etc.

The effect of wind on the system and implement these considerations into the controller design procedure for both the winch controller and the UAS controller. Although the UAS autopilot mostly considers the tether tension as an external disturbance rather than a real-time estimated input, it is critical to understand the operation point which enables the control design engineers to design a more robust length controller and autopilot for the system.

1.2. Literature Review

Using cables in mechanical applications is very common; thus a wide variety of analytical and numerical research has been carried out around analysis of cable dynamics and static behaviour. Many methods have been employed to analyze the cable dynamics in literature, as noted in Choo and Cassarella's survey of different cable modelling studies [1]. These methods include continuous and discrete approaches such as method of characteristics, finite element method, linearization method, equivalent lumped-mass method, etc. Continuous approaches are mostly used to analyze a cable's oscillation modes or when the static configuration of the cable is of interest. It is common for the continuous models to be incapable of yielding a transient response for the whole system due to the complication or inefficiency of solving the equations of motion. Discrete models are used to analyze larger displacements in the cable when the transient response of the system is desired. Among these simulation approaches, method of finite-segment has been investigated properly to study the cable's behavior. Several works have been carried out by discretizing the cable into several segments and the equations were derived by writing the system equations for a multibody system of finite cable elements.

In 1972, Dominguez et al. employed a discrete approach to analyze the static and dynamic behavior of cable systems [2]. In 1976, Winget and Huston employed a non-linear

3D finite segment approach to investigate the dynamic behavior of a cable or chain. Dreyer and Van Vuuren [3] have studied static configuration of an inflexible 2D cable anchored at its endpoints in 1999 without considering external forces on the cable except gravitational forces. The solution was obtained by numerical analysis for both continuous and discrete models and results were validated experimentally.

As mentioned before, cables are used in towing aerial and marine vehicles. Kamman and Huston modeled towed and tethered cable systems with a focus on reel-in/pay-out and towing configurations for marine vehicles in 1999 [4]. Same authors also numerically simulated cables for marine applications in 2001 [5]. Dynamic analysis of cable structures as discrete systems have been studied in several works and are available widely through literature. Buckham et al. have carried out extensive research in this area, specifically for underwater vehicles including both low-tension and taut cable applications [6]–[11].

Many research papers have also been published on analysis of cable dynamics and statics for aerial applications. In 1967, Genin and Cannon investigated equilibrium configuration and tensions of flexible cables in a uniform flow field with a focus on aerial towed vehicles [12]. They considered the effect of cable weight, friction, and drag in their mathematical analyses. In 1971, Huffman and Genin in a continuation of Genin's previous work, studied the dynamical behavior of the same system [13]. Non-linear mathematical model for extensible cables were derived and computer simulations were run. This model considers large displacements as well as impact forces and drag forces while studying the stability of the system.

Quisenberry have worked on aerial tow systems by mathematical modelling of the cable as a discrete cable-body system using Lagrange's method [14]–[16]. In these works, the authors have developed a model for the cable in an aircraft tow system and predicted the dynamic response of the towed aircraft.

In a research by Hembree and Sleger [17], authors have modeled a tether using recursive rigid-body dynamics in which they estimate the tension in the tether based on a low strain model. The elements used in this work are not elastic and each cable element has only 2 DOF. Johansen et al. [18] have modeled inextensible cable dynamics using a discrete cable model and validated their results with a hanging cable.

Montano et al. [19] studied dynamics of a tethered buoy using mixed finite elements. In this work, the model considers a quasi-extensible tether attached to a buoy vessel and implemented the buoy's dynamics. Surendran and Goutam [20] investigated reduction of the dynamic amplitudes of moored cable systems. They modeled the cable as a discrete lumped mass system and modeled the static and dynamic responses of the system.

Aerostats are similar to the system we are analyzing in this work in the sense that they are aerial systems that exert tension to the upper end of the tether, with the exception that they usually have no vertical thrust and have more drag due to the wind. There are many works on tethered aerostats in literature which date back to as early as to the 70's.

Kang and Lee [21] have developed equations of motion for a tethered aerostat under atmospheric turbulence using nonlinear cable dynamics for an extensible tether. They have obtained results on tether forces due to different wind gusts. Rajani et al. [22] have studied dynamic stability of a tethered aerostat in which the cable is modeled as a discrete system of segments modeled as a "spring-mass-dashpot" system with the segment mass concentrated at the nodes.

In a similar work, Stanney and Rahn [23] developed equations of motions for a tethered aerostat wind under turbulence and obtained results on aerostat position, attitude, and tether tension due to wind disturbances. In two separate works in 1982 and 2001 by Jones and Krausman [24], and Jones and Schroder [25], respectively, the authors simulated the response of a tethered aerostat to atmospheric turbulence. They used a "Frozen-Field" turbulence model and solved for the dynamics of the system using a non-linear dynamic simulation.

Few works have been accomplished modeling a tethered aerial vehicle. Among these, Nicotra et al. [26] studied taut cable control of a tethered UAV; although the focus is mostly on controlling the UAV and less attention is paid to the cable model. The cable is considered taut and the cable tension is considered a disturbance to the UAV. Therefore no specific cable model is developed in this work. Lupashin and D'Andrea [27] accomplished a work on stabilization of a flying vehicle on a taut tether, however, the focus of this research is mostly on the control of the vehicle rather than modelling the cable itself.

Muttin [28], [29] has developed equations of motion for a flying ducted fan for oil slick surveillance and detection. In this work, the model is approximated using finite-element method and an updated Lagrangian approach to model the reeling process for the UAV as well as effect of wind on Vertical Take Off/Landing (VTOL) process. The research considers the effect of wind on the ducted fan as well as simulates the cable reeling during operation and emergency mode.

1.3. Thesis Organization

Chapter 2: Preliminary Modeling of the Cable, is dedicated to preliminary modelling of a static cable. Although all the catenary equations have been derived and used extensively in research in all applications, no organized solving algorithm for such a system has been developed. This chapter models the cable as a simple static catenary shape, however it implements the equations in a very user-friendly code.

Chapter 3: Two-Dimensional Multi-Body Dynamics Modeling and Analysis of the Tether, is allocated to mathematical modelling and simulation of the cable as a multibody dynamic system by modelling the system using Lagrange's method. This method is an energy-based method and is more capable and less labor-intensive than Newtonian Mechanics in deriving equations of motions of dynamic systems, more particularly multi-body systems.. The equations are then solved and simulated using MATAB software. Some results are presented at the end for a specific case.

Chapter 4: Three- Dimensional Multi-Body Dynamics Modeling and Analysis of the Tether, analyzes the same problem in a 3D space using the same method as Chapter 3. Simulations are run to capture the non-planar motion of the cable. Results are presented and observations are made at the end.

Finally, **Chapter 5: Conclusion**, summarizes the work done in this thesis. A few recommendations are made and some future works are proposed at the end.

1.4. Contributions

The algorithm developed in Chapter 2, although based on simple catenary equations, can help the user in analyzing a static cable efficiently without being involved in solving catenary equations. Although the purpose of this work was to analyze a captive UAS, the static model can be used in any application using a static cable when the only external forces applied to the cable are the ones at the cable ends as well as the gravitational force. The user can define mechanical properties of the cable as well as the forces at the top end, the cable length, as well as horizontal and vertical distance between both cable ends.

Although various extensive research has been carried out on dynamics of flexible cables, there is no straightforward organized approach or tool available to analyze dynamics of a cable in specific applications. The cable model derived in this work along with the computer algorithm developed, allow the user to initiate analyzing the cable's behavior efficiently. The cable model is very versatile and can be applied to a wide range of applications in several environments. Given a correct and accurate vehicle model, such as a flying vehicle, or a submarine equipment, one can integrate these models into the algorithm, allowing the software to generate results of a specific simulation. As an example, one can use the cable model to analyze the behavior of a tethered aerostat, or a towed submarine vehicle, as long as the user is familiar with the dynamics of the vehicle and is able to model it mathematically.

Chapter 2.

Preliminary Modeling of the Cable

2.1. Introduction

As mentioned in the first chapter, the main objective of this research is to predict the behaviour of the cable while the UAS is in different operation scenarios e.g. hovering above the base station, or vertical landing or take-off. As a first attempt to predict these cable shapes for the preliminary phases of the project, analysis of static behavior of the cable as a catenary shape was initiated.

Knowing certain constant parameters of the system, such as cable's mechanical properties, as well as variable parameters, one can solve for the unknowns of the problem using the algorithms offered in this chapter. In order to determine the unknown parameters of the problem, one should solve the equations for some parameters such as the cable shape function, the length of the cable, and the tension on both ends as well as the tension function along the tether. Although the latter is of less importance since the maximum tension does not occur along the cable.

The following sections will explain how calculation of some of these parameters was performed during the preliminary phases of the project in order to achieve an estimate of the cable's behaviour. The theory behind this chapter's content will be presented followed by the application to this specific problem as well as solution of the equations and results. A conclusion section will follow at the end discussing the obtained data. The MATLAB™ codes and instructions on how to use these codes for other similar case-specific applications are presented in Appendix A.

2.2. Development of the Case-Specific Cable Model

As discussed in Chapter 1, this application concerns a UAS flying at a certain altitude and is connected to a base station using a flexible power cable. This configuration is simplified assuming the cable is hanging only under effect of its own weight and is illustrated in Figure 2.1. It is assumed that the UAS is flying in the same plane as the reel i.e. *xz*-plane; therefore the problem is considered two-dimensional. In practice this is not the case since the UAS is flying in a three-dimensional space. The 3D modelling and simulation of the cable will be investigated in Chapter 4.

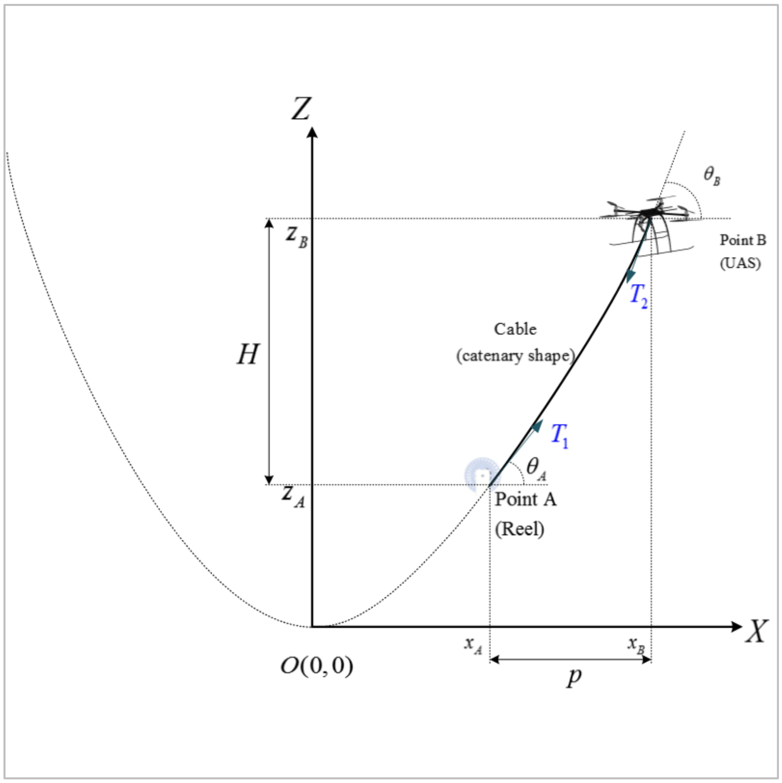

Figure 2.1. UAS – Cable Geometry Configuration

In addition, assume that there is no wind present so the variables are being acquired under a static condition. In practice, usually wind is present and affects the physics of problem. In the initial phase of this project it was necessary to estimate the relation between parameters such as UAS height and horizontal position, cable length, tensions and cable slopes in order to capture some basic working assumptions for the UAS. As a result, the analyses in this chapter are limited to a simplified case where external loads are not present. The effect of wind on the cable shape and forces will be investigated in Chapter 3.

Assuming the reel position is fixed, it will have a certain horizontal and vertical distance from the hypothetical vertex of the catenary in each different setting; named x_A and z_A respectively. Therefore the coordinate system is not fixed and is defined relative to the reel's position. That is, after solving the problem the parameters x_A and z_A represent the distance between the reel and the vertex of the virtual curve. At this stage, the origin of the virtual catenary curve is defined with respect to the base station A and is fixed for this specific combination of L, p, and H. From this point all other variables such as z_A are defined in this coordinate system.

Generally, this system has six main variables, namely, *L*, *p, H*, *x_A, z_A, and T_0*. Once these six variables are known, the system is defined and all other variables, i.e. *x_B, z_B, θ_A*, *θ_B*, *T_1* and *T_2* which are dependent on these variables and can be calculated. Since z_A is also a direct function of x_A, one can represent the system by 5 main equations and 5 variables. As an example, if some of the parameters of the problem including UAS height *H*, cable length *L*, and UAS horizontal offset from the base station, labeled *p* are known, the system can be solved.

To reduce the number of unknowns and equations, the equations are reordered by substituting some of these variables with others. Looking at catenary equations, one can express the vertical position of any point on the cable in terms of its horizontal position. The following equations hold true for both points A and B or generally any point on the cable. As stated before, points A and B, respectively represent the reel and the UAS and the difference between their vertical coordinates is the UAS height *H*.

$$z_A = \frac{T_0}{\mu}\left(\cosh\frac{\mu x_A}{T_0} - 1\right) \tag{2.1}$$

$$z_B = \frac{T_0}{\mu}\left(\cosh\frac{\mu x_B}{T_0} - 1\right) \tag{2.2}$$

Therefore, one can see that *z_A* itself is a function of *x_A and T_0* and can be considered as a dependant variable and be calculated after solving the system equations. Since the difference between these two equations represents the height of the UAS, by geometry, one can write:

$$\begin{cases} x_B = x_A + p \\ z_B = z_A + H \end{cases} \tag{2.3}$$

Equation (2.1) can be combined with (2.2) to get

$$H = \frac{T_0}{\mu}\left(\cosh\frac{\mu(x_A + p)}{T_0} - \cosh\frac{\mu x_A}{T_0}\right) \tag{2.4}$$

On the other hand, taking a derivative of the angle equation will yield the slope equation of the curve. Applying the slope equation at each point *A* and *B* will result in Equations (2.5) and (2.6).

$$\tan\theta_A = \left.\frac{dy}{dx}\right|_{x=x_A} = \sinh\frac{\mu x_A}{T_0} \tag{2.5}$$

$$\tan\theta_B = \left.\frac{dy}{dx}\right|_{x=x_B} = \sinh\frac{\mu x_B}{T_0} \tag{2.6}$$

Also, the tension at any point can be expressed in terms of its tangent angle as:

$$T = \frac{T_0}{\cos\theta} \tag{2.7}$$

that is, for any point along the cable, the longitudinal cable tension is related to the cable angle by this equation.

Figure 2.2 shows the free body diagram of the whole cable in equilibrium. Assuming the reel is locked and therefore prevented from rotation, the cable can be assumed to be suspended between two fixed points *A* and *B*. Writing the equilibrium equations in both *x* and *z* directions will yield:

$$\begin{cases} T_A \cos\theta_A = T_B \cos\theta_B = T_0 \\ T_B \sin\theta_B = T_A \sin\theta_A + \dfrac{\mu L}{T_0} \end{cases} \tag{2.8}$$

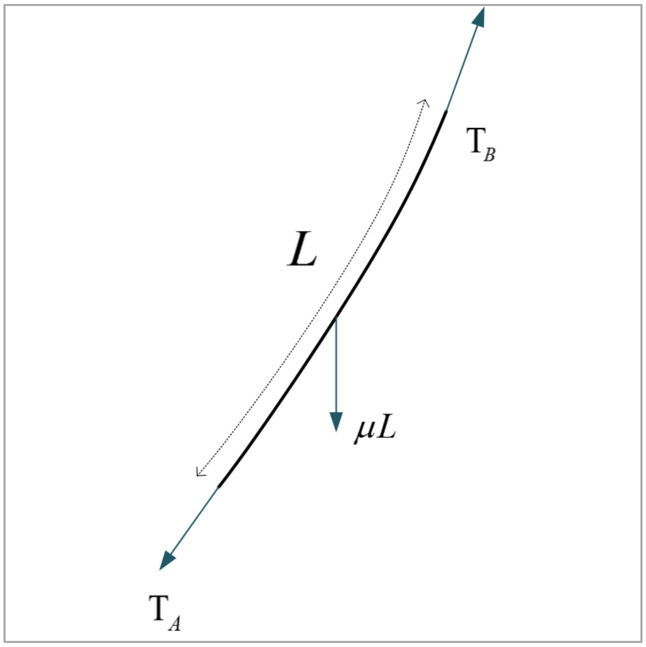

Figure 2.2. Free-body diagram of the cable in static equilibrium

By substituting *T*$_B$ from the first line in Equation (2.8) in the second equation and some simple operations one can conclude the following.

$$\tan\theta_A - \tan\theta_B = \frac{\mu L}{T_0} \tag{2.9}$$

By substituting Equations (2.5) and (2.6) into (2.9), Equation (2.9) can be expressed as:

$$\sinh\frac{\mu\left(x_A+p\right)}{T_0}-\sinh\frac{\mu x_A}{T_0}=\frac{\mu L}{T_0} \tag{2.10}$$

Overall, by substituting some of the variables and reordering equations, one can reduce the number of variables to five and the number of equations defining the system to two. Equation (2.11) shows the system of two equations that should be solved in order to determine the system's solution. The method for solving these equations will be presented in Section 2.3. At this point, there are five variables present in this system of equations which are *L*, *p, H*, x_A*, and* T_0. The equations can be solved given any three parameters of the system.

$$\begin{cases} H=\dfrac{T_0}{\mu}\left(\cosh\dfrac{\mu\left(x_A+p\right)}{T_0}-\cosh\dfrac{\mu x_A}{T_0}\right) \\ L=\dfrac{T_0}{\mu}\left(\sinh\dfrac{\mu\left(x_A+p\right)}{T_0}-\sinh\dfrac{\mu x_A}{T_0}\right) \end{cases} \tag{2.11}$$

2.3. Solving the System of Equations

Equation (2.11) is a set of two nonlinear equations which express variables L and H in terms of independent variables x_A, T_0, and *p*. Depending on the configuration of the problem, this equation can be changed to match the unknowns of the problem. As an example, in an application cable length *L*, UAS height *H* and horizontal position p might be known, therefore the only unknowns of the problem will be x_A and T_0. However this configuration will become a set of implicit equations in terms of x_A and T_0, since x_A cannot be directly expressed in terms of T_0 or vice versa. In general, Equation (2.11) represents two implicit nonlinear equations and cannot be solved analytically. As a result, a numerical method should be used to solve this equation, which is explained in Sections 2.3.1 and 2.3.2.

2.3.1. The Basic Optimization Problem

Consider a system of n equations and n unknowns as shown in Equation (2.12).

$$\begin{cases} f_1\left(x_1, x_2, \ldots, x_n\right) = 0 \\ f_2\left(x_1, x_2, \ldots, x_n\right) = 0 \\ \vdots \\ f_n\left(x_1, x_2, \ldots, x_n\right) = 0 \end{cases} \tag{2.12}$$

Now, the variables are defined as a vector called $\vec{x}$, where

$$\vec{x} = \begin{Bmatrix} x_1 \\ \vdots \\ x_n \end{Bmatrix} \tag{2.13}$$

One can then write Equation (2.12) as

$$\begin{cases} f_1\left(\vec{x}\right) = 0 \\ f_2\left(\vec{x}\right) = 0 \end{cases} \tag{2.14}$$

Equation (2.14) has a unique solution specifically for this system of equation, $\vec{x}^*$ when

$$F\left(\vec{x}\right) = \sum_{i=1}^{n} \left[f_i(\vec{x}) \right]^2 = 0 \tag{2.15}$$

where $F\left(\vec{x}\right)$ is usually referred to as the cost function or objective function [3].

In order to solve Equation (2.15), One can reduce this problem to an optimization problem of the following form

$$\begin{cases} \underset{\vec{x}}{\text{minimize}} \; F\left(\vec{x}\right) \\ \text{subject to appropriate constraints} \end{cases} \tag{2.16}$$

The solution to Equation (2.16), $\vec{x}^*$ is the minimizer of the objective function $F\left(\vec{x}\right)$; thereby satisfying the requirements of the original system Equation, (2.12). Usually a numerical search method such as Steepest-Descent method, Newton Method, or Gauss-Newton Method should be employed to solve Equation (2.16).

2.3.2. Numerical Solution to the System of Equations

As explained in Section 2.3.1, before solving the problem, it should be formulated properly. One can write Equation (2.11) in the following form to comply with the format of the optimization problem discussed in Section 2.3.1.

$$\begin{cases} H-\dfrac{T_0}{\mu}\left(\cosh\dfrac{\mu\left(x_A+p\right)}{T_0}-\cosh\dfrac{\mu x_A}{T_0}\right)=0 \\ L-\dfrac{T_0}{\mu}\left(\sinh\dfrac{\mu\left(x_A+p\right)}{T_0}-\sinh\dfrac{\mu x_A}{T_0}\right)=0 \end{cases} \tag{2.17}$$

If first and second equations are denoted by *f₁(x)* and *f₂(x)* respectively and consider T_0 and x_A as two unknown variables x_1 and x_2, one can write

$$\begin{cases} f_1\left(x_1,x_2\right)=H-\dfrac{x_2}{\mu}\left(\cosh\dfrac{\mu\left(x_1+p\right)}{x_2}-\cosh\dfrac{\mu x_1}{x_2}\right)=0 \\ f_2\left(x_1,x_2\right)=L-\dfrac{x_2}{\mu}\left(\sinh\dfrac{\mu\left(x_1+p\right)}{x_2}-\sinh\dfrac{\mu x_1}{x_2}\right)=0 \end{cases} \tag{2.18}$$

Now define a vector variable $\vec{x}$ denoting the unknown variables of the problem, where

$$\bar{x} = \begin{pmatrix} x_A \\ T_0 \end{pmatrix} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \tag{2.19}$$

One can then re-write Equation (2.18) as

$$\begin{cases} f_1\left(\bar{x}\right) = 0 \\ f_2\left(\bar{x}\right) = 0 \end{cases} \tag{2.20}$$

which can be reduced to Equation (2.21) similar to the optimization problem shown in Equation (2.16).

$$\begin{cases} \underset{\bar{x}}{\text{minimize}}\ F\left(\bar{x}\right) \\ \text{subject to} \\ x_1, x_2 > 0 \end{cases} \tag{2.21}$$

In this work, the optimization problem for Equation (2.17) is solved using MATLAB's nonlinear equation solver function “fsolve”. The programming details and the MATLAB™ codes are presented in Appendix A. This tool needs the input equations as a vector function of the following form where f_1 and f_2 are the same functions as described in Equation (2.18).

$$G = \begin{bmatrix} f_1 \\ f_2 \end{bmatrix} \tag{2.22}$$

This search algorithm is able to solve Equation (2.17) with a convergence tolerance of less than 10^{-12}. After running each algorithm, the tolerance is checked to ensure convergence of the problem. Some example results for different scenarios one can apply this tool are presented in the next section along with discussion on how this tool has helped achieve some initial working assumptions for the UAS operation. Note that the mentioned equations are set up for a specific case when the cable length, altitude and horizontal position of the UAS is known. However the equations can be set up accordingly to accommodate other scenarios. As an example imagine that the altitude and horizontal position are known and a certain tension on the cable is required. One can set up the equations in a way that the algorithm yields the required cable length to keep the desired cable tension. This would be particularly helpful if the purpose of the analyses is to design a tension controller system.

2.4. Results and Discussions

This section is dedicated to exploring some scenarios of UAS operation. The methods and algorithms discussed in previous sections are applied to some specific cases and present results and discussions about the observations. The MATLAB™ codes which solve specific problems are presented in Appendix A. It should be noted that in all cases, the constant μ, cable's weight per length should be entered at the beginning of the code. Moreover, all units used in this algorithm are in metric system. More instructions regarding running the codes to solve the problem are presented in Appendix A.

It should be emphasized again that this chapter assumes a planar motion without presence of wind or disturbance on the system's dynamics and do not consider the dynamics of the system and specifically the UAS. The UAS is considered to be able to hold its stability and position autonomously and that it introduce no disturbances on the cable. That is, a steady state equilibrium is assumed for the system. This assumption is valid only for the cases where wind is not present and there is not external forces applied to the system.

As mentioned previously, one can define the problem as they desire based on the known factors of the system. For example, one can use three inputs including L, H and p to determine the cable's shape function as well as tensions along the cable as a function of horizontal coordinate, x. Also, the codes will help the user develop some general insight about how the cable will achieve equilibrium in some scenarios. Three cases are investigated in the following three sections.

2.4.1. Case I: Known UAS Coordinates

In general, for the system developed, knowing three parameters will make the problem defined. In the event that the UAS's coordinates are known, one can solve the problem by feeding the known parameters to the algorithm written in the file named "Mtethsolve3.m". This code solves the catenary equations and determines cable variables of interest including tensions and cable angle at both ends or on any point of interest on the cable.

Generally, parameters such as altitude (*H*) and the horizontal offset (*p*) can be available through the UAS's GPS system or other systems such as an inertial navigation systems. Length of the cable (*L*) can also be known by tracking the reel's payout, although other methods might be used to determine these values. In the case considered in this work, since the project is in the research and development stage, the purpose is only to predict different scenarios that could occur during flight and gain a better understanding of the physics of the system.

To illustrate the results from this case, a sample algorithm is run for a system with parameters *p*=15, *H*=120, and *L*= 122. The cable parameters used in the simulation are presented in Table 2.1

Table 2.1. Parameters Used in Simulation

Simulation Parameter	Value	Units
Mass per length	0.14	[kg/m]
Cable Modulus of Elasticity	80	[GPa]
Cable Diameter	0.0135	[m]
Cable Length	122	[m]
UAS Height	120	[m]
UAS Offset	15	[m]

The syntax for the algorithm is in the form of

```
>>Mtethsolve3(p,H,L)
```

As an example, running a script called “Mtethsolve3” with the parameters of Table 2.1 and running the following command will yield the results.

```
>> Mtethsolve3(15,120,122)
```

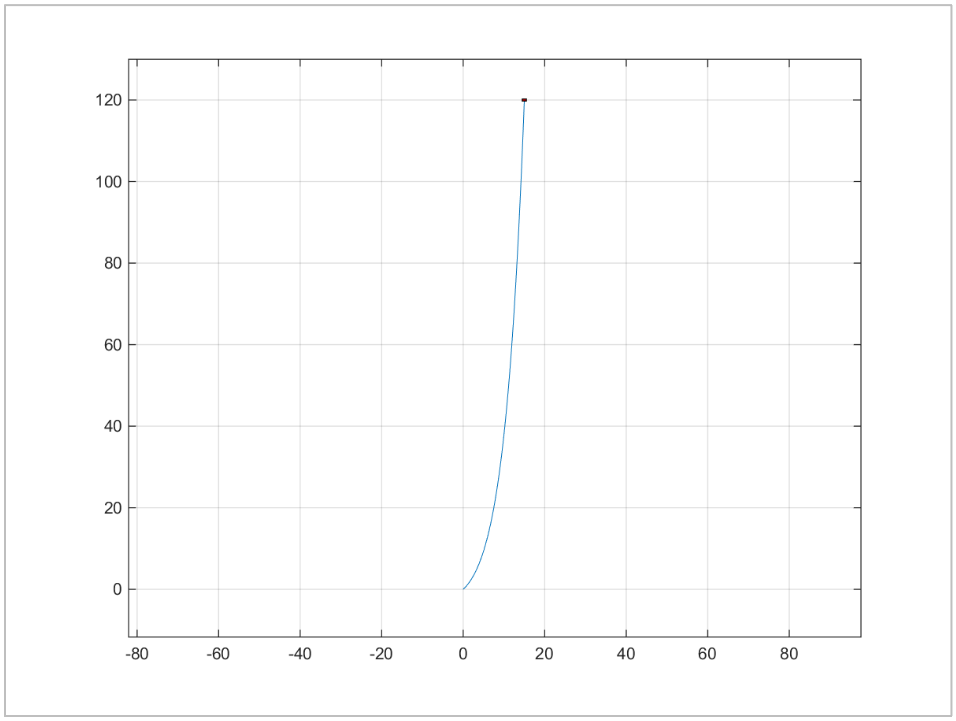

Figure 2.3. Cable shape function

Using this algorithm, once the system equations are solved, one can plot the cable shape function as well as evaluate all cable values. Using another script called “Mtethplot3”, the cable shape can be plotted in a 2D plane. Figure 2.3 shows the cable’s shape function using four different values of horizontal offset for the UAS with fixed altitude and cable length. The base station is located at the origin of the coordinate system, and the red rectangle represents the hovering UAS.

In order to see the tension ranges of the cable both at the UAS and at the winch, the algorithms to solve for the cable parameters are used and cable tensions are plotted at the winch and at the UAS along with the cable angle at both ends. This is useful specifically in understanding the ranges of cable tensions occurring in different hovering scenarios. It has helped with understanding the power ranges for the UAS based on the vertical thrust it needs to provide to maintain a specific cable tension range.

Table 2.2 shows the parameter ranges considered for the cable length as well as UAS positions according to the operation requirements. The UAS is allowed to move in a 40 meters horizontal distance from its base station while it is supposed to hold an altitude between 110 to 130 meters.

Table 2.2. UAS Operation Requirements

Simulation Parameter	Value	Units
Cable Length (L)	120.5-125	[m]
UAS Altitude (H)	110-130	[m]
UAS Horizontal Offset (p)	0-40	[m]

Figure 2.4 shows positive, zero and negative tension angles at the winch for a certain UAS height and cable length at three different offsets. Note that the angles defined are with respect to the horizontal line. For each cable length, the angles will be zero at a specific offset. This essentially means that the base station will be located at the vertex of the catenary making the tangent at the curve, i.e. cable angle zero. Negative angles technically mean that the cable is slack and is hanging below the virtual base station. In practice, if the base station is located on the ground the cable will be lying on the ground and the physics of the cable will change and this analysis will not be applicable to the system. However, assuming the cable is allowed to hang below the winch, one can use this tool to solve for the system's parameters.

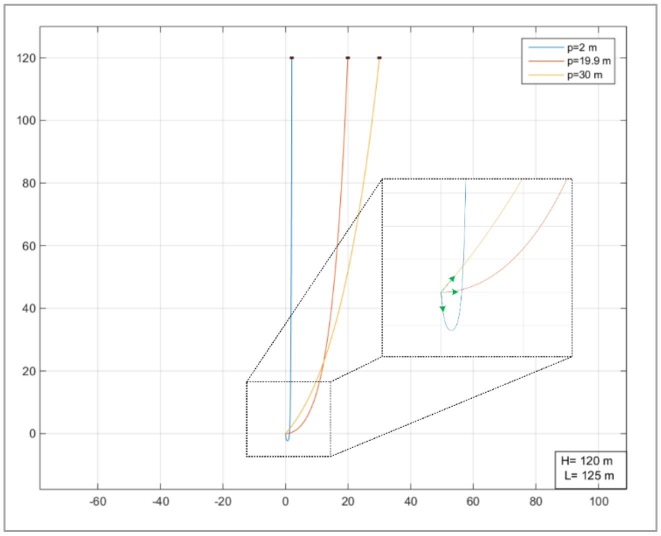

Figure 2.4. Cable Angle Definition

Figure 2.5 shows the magnitude of the cable tensions and angles at the winch versus the horizontal offset for different cable lengths for a UAS hovering at a constant height of 120 m. For each cable length, the figures are plotted against the offset to a maximum offset value as the geometry permits. For an inextensible cable, the limit of the offset is determined by the UAS height and length of the tether.

One can observe that the tension values approach small numbers as the UAS gets closer to the base station. For higher cable lengths, when the horizontal offset gets smaller, the cable will gain some slackness and the tension direction will be downwards. This can be confirmed by the cable angle curves.

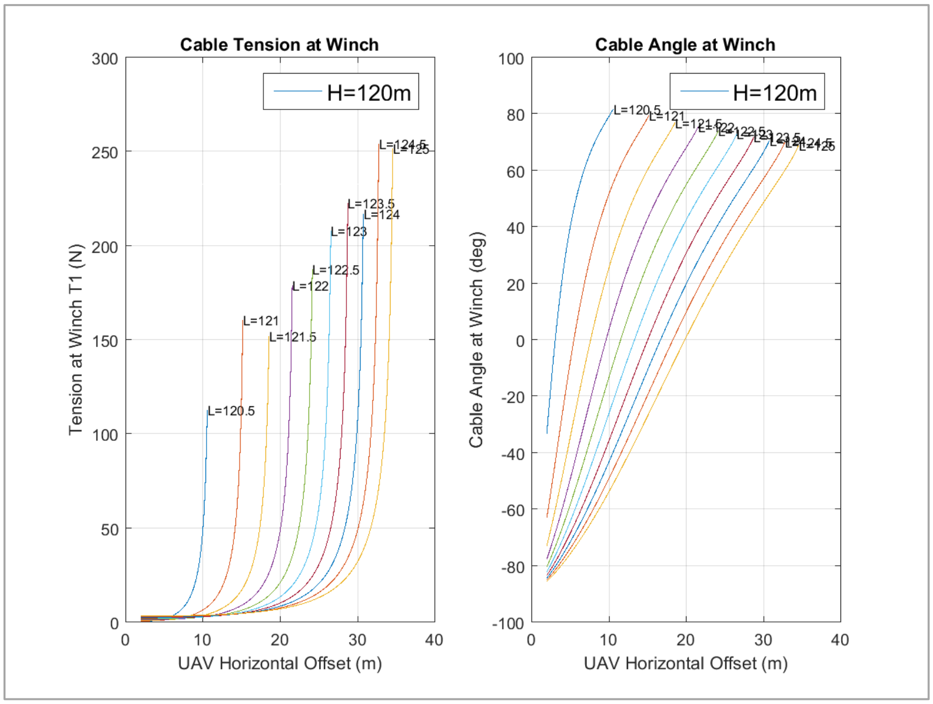

Figure 2.5. Cable Tension and Angle vs. Horizontal Offset at Winch for Various Lengths, Constant Altitude

Figure 2.6 shows the cable tensions and angles at different cable lengths plotted against the UAS horizontal offset. Again, one can observe the UAS tensions when the UAS is closer to the base station tend to converge to a certain range of numbers. These numbers are technically the weight of the cable hanging from the UAS. This could be interpreted in another way; i.e. when the UAS is closer to the base station, changing the cable length affects the tensions at the UAS less significantly than when it is horizontally positioned further, i.e. larger *p* values.

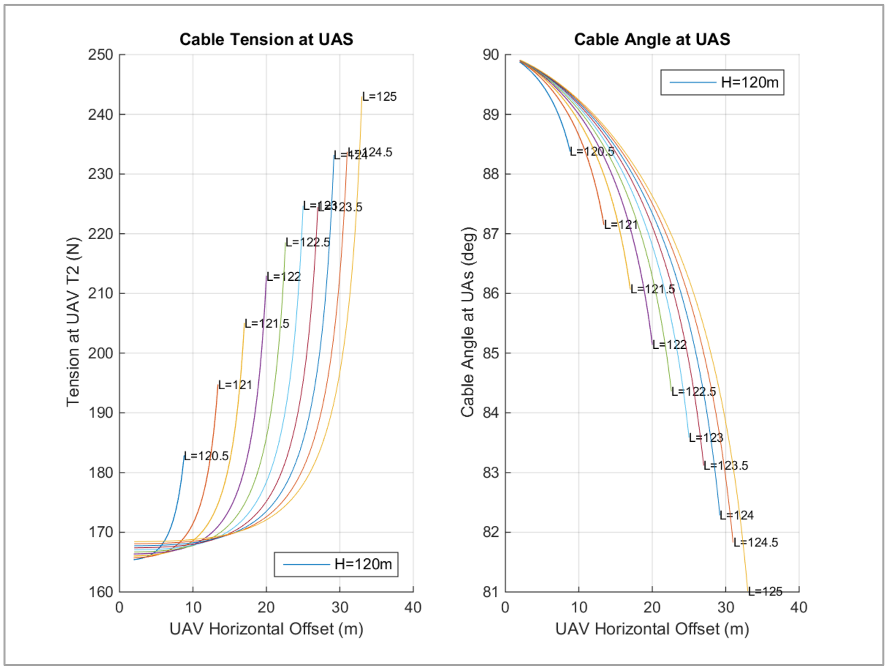

Figure 2.6. Cable Tension and Angle vs. Horizontal Offset at UAS for Various Lengths, Constant Altitude

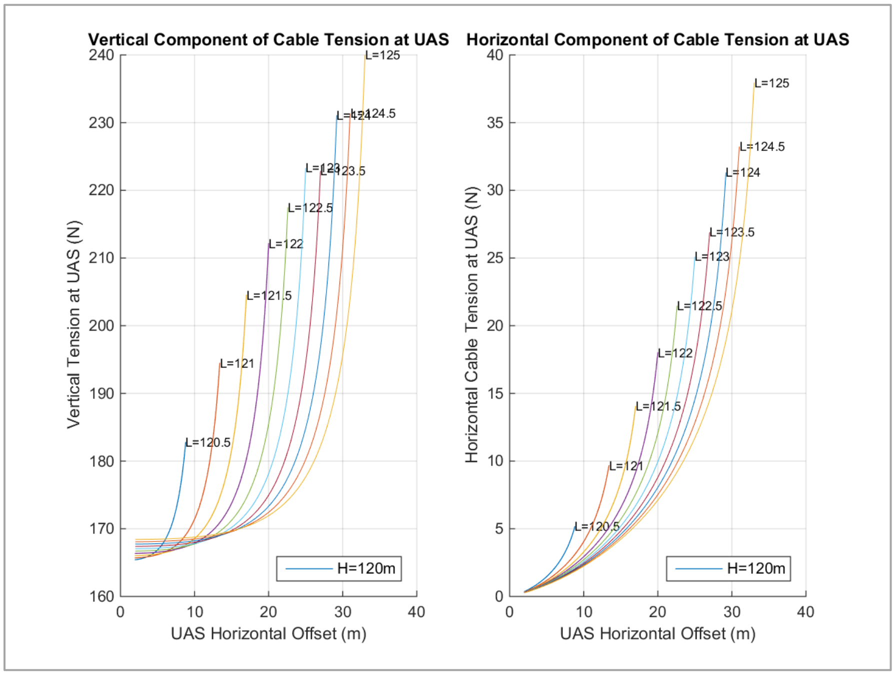

Figure 2.7. Vertical and Horizontal Components of Tether Tension at UAS for Various Lengths, Constant Altitude

Although these two figures are useful in understanding the cable tension, they do not give transparent implication of the vertical loads on the UAS. Therefore, the vertical and horizontal components of the tension at UAS are plotted and presented in Figure 2.7.

To observe the effects of the UAS altitude on the tensions, for a UAS hovering at a constant offset, Figure 2.8 and Figure 2.9 are presented below. As can be seen in Figure 2.8, the tension at the reel takes a minimum value at any certain height. That is, for this specific horizontal position, the minimum tension at the reel is 5 N. However by comparing Figure 2.8 and Figure 2.10 one can confirm that this minimum value increases as *p* increases. Figure 2.10 shows the results for the same cable lengths and altitudes at the offset of 25 meters.

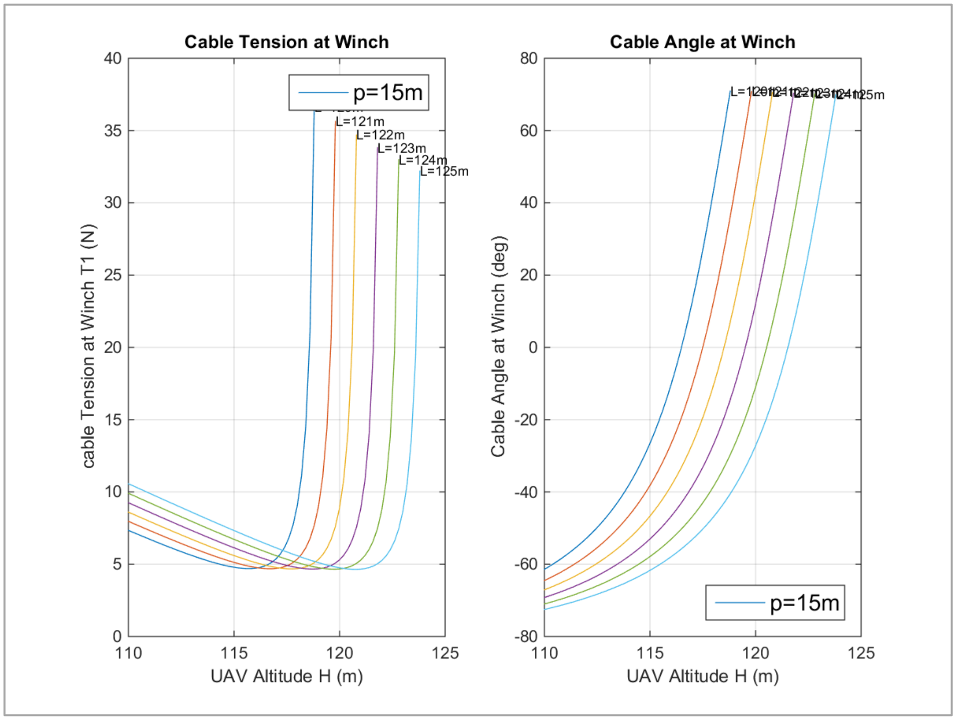

Figure 2.8. Cable Tension and Angle vs. Altitude at Winch for Various Lengths, Constant Offset

Figure 2.9 presents the cable tensions and pull angle at the UAS at different heights when the UAS holds offset and constant cable length. This figure helps one understand how much the tension increases as the UAS gains height holding other parameters constant. It is useful to know the sensitivity of the tensions both at the UAS and at the winch to height so the user can understand what to expect in different scenarios.

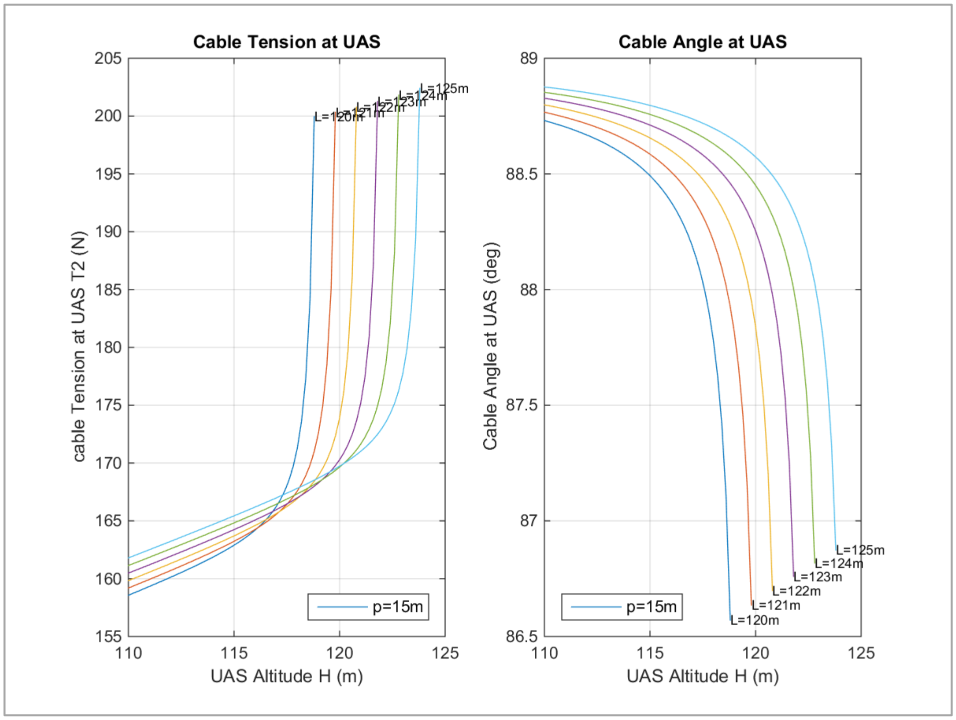

Figure 2.9. Cable Tension and Angle vs. Altitude at UAS for Various Lengths, Constant Offset

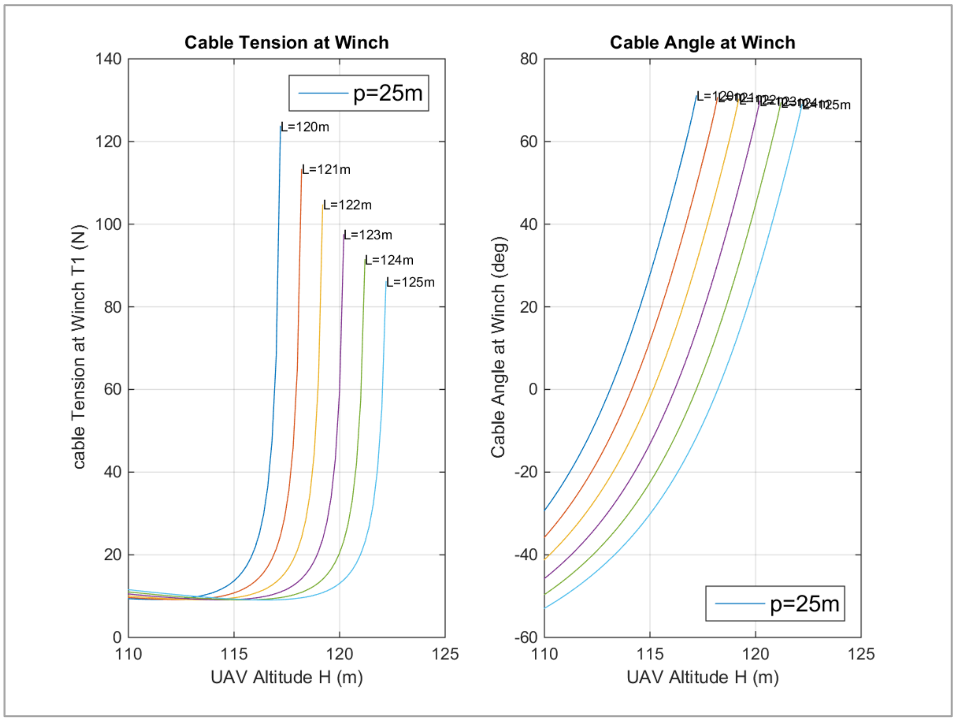

Figure 2.10. Cable Tension and Angle vs. Altitude at Winch for Various Lengths, Constant Offset of 25 m

To illustrate the vertical and horizontal components of the tension at the UAS, they are plotted against the UAS altitude *H* in Figure 2.11. Given any desired horizontal offset from the base station as well as cable length, one can plot the cable tensions at the UAS or the base station against different flying altitudes to understand how the tension changes as the UAS increases its operation height.

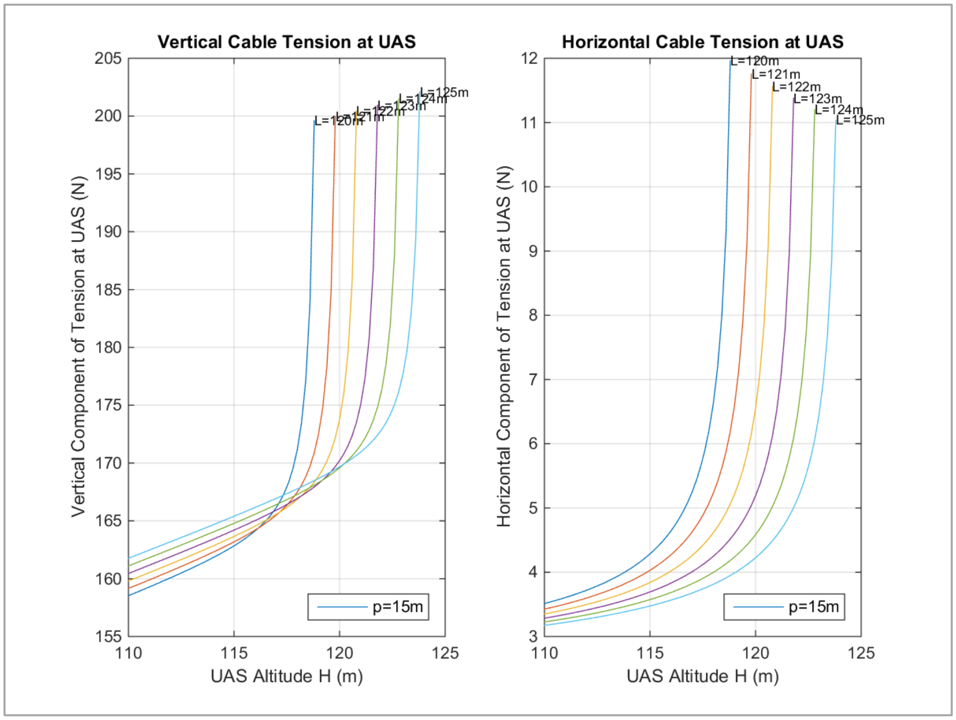

Figure 2.11. Vertical and Horizontal Components of Tether Tension at UAS for Various Offsets, Versus UAS Altitude

Considering a UAS at a certain height and distance from base station, it is worthwhile to observe the tension sensitivity to cable length changes at both cable ends. Figure 2.12 and Figure 2.13 show the tensions at winch and UAS respectively as a function of cable length when the UAS height is held constant. The results are plotted for different horizontal offsets.

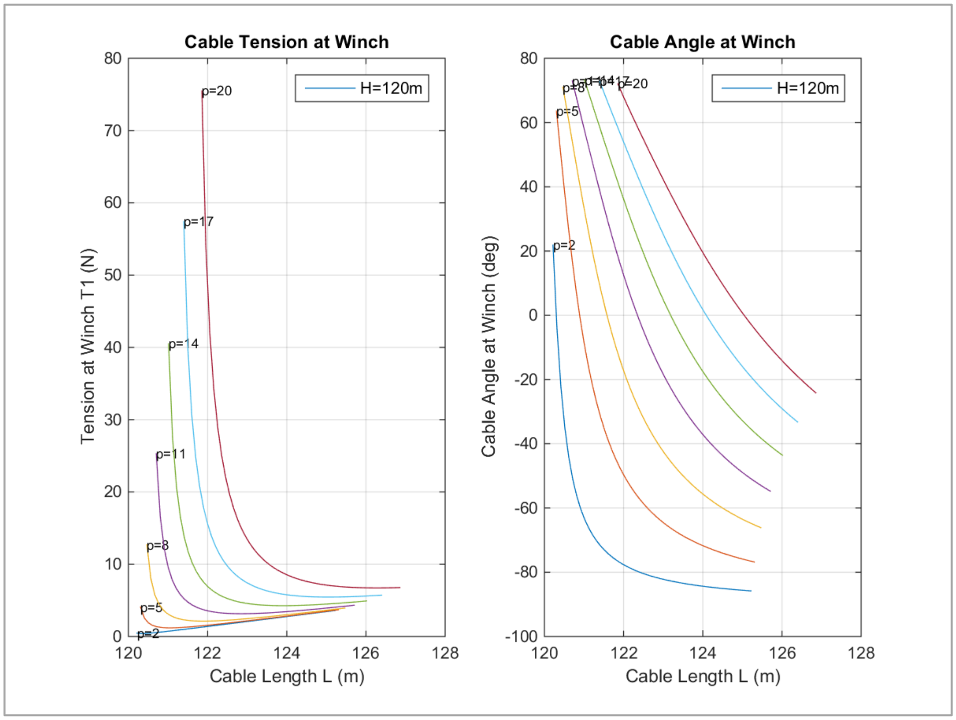

Figure 2.12. Cable Tension and Angle vs. Cable Length at Winch for Various Offsets, Constant Altitude

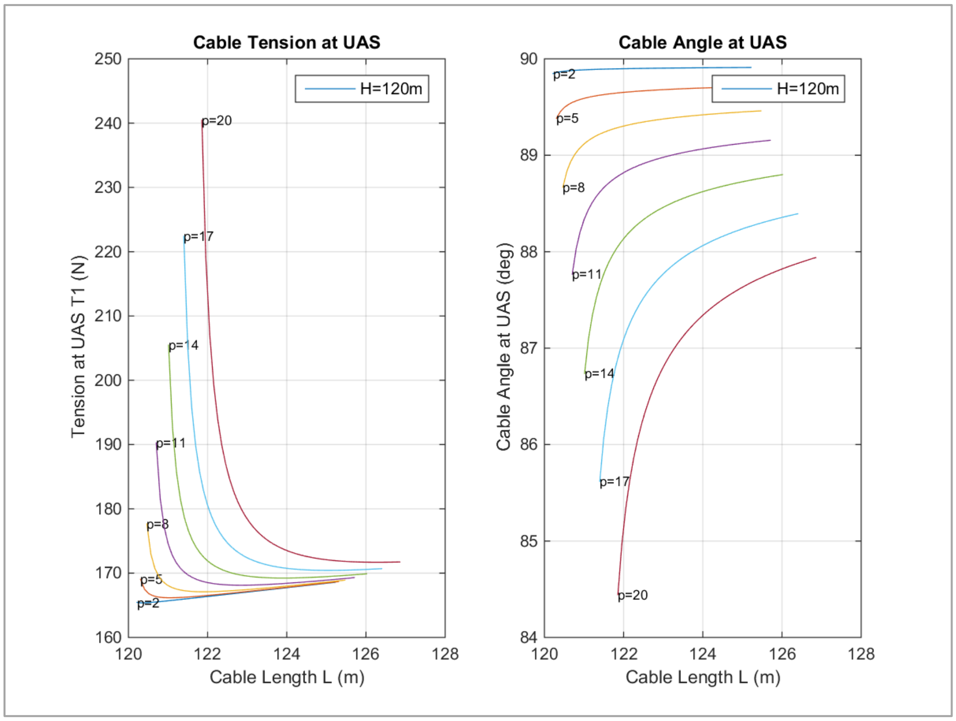

Figure 2.13. Cable Tension and Angle vs. Cable Length at UAS for Various Offsets, Constant Altitude

As mentioned before, the tension itself at the UAS might not define the need for the power requirement of UAS thrust system. Thus, Figure 2.14 has been generated showing the two horizontal and vertical component of these forces.

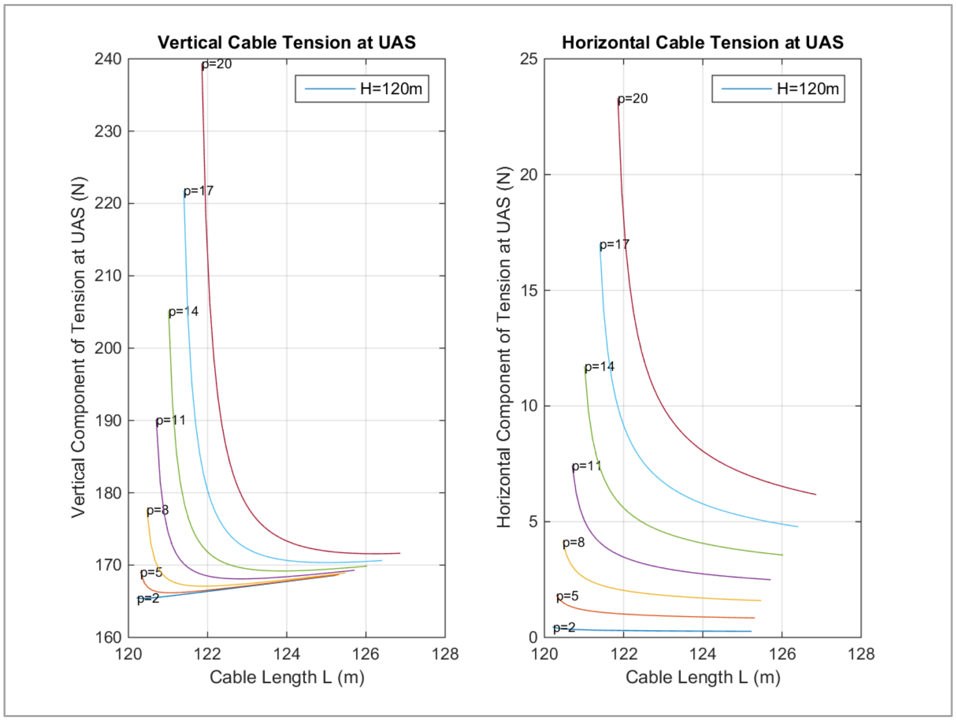

Figure 2.14. Vertical and Horizontal Components of Tether Tension at UAS for Various Offsets, Constant Altitude

The results in this section were essentially used in the first stages of research and development of the UAS system in order to estimate the winch motor's torque as well as UAS thrust production capability for a more confident conceptual design. In the next section, another approach is used and some new results are presented.

2.4.2. Case II: UAS Thrust and Pitch Effect

In order to analyze the UAS thrust effects on the cable, another algorithm was developed to which one can feed the UAS vertical and horizontal thrust. In equilibrium, the thrust force from the UAS will be equal to the tension at the upper end of the cable. Therefore, by inputting the vertical and horizontal components of the UAS thrust to the algorithm along with the cable length, the algorithm is able to solve the equations and find the parameters of the catenary including horizontal and vertical distance from the base station–called offset and altitude, respectively– as well as the vertex location $x_{A.}$

Having obtained these parameters, the system is defined and all other parameters are calculated accordingly. At this point, the cable shape function can be plotted as well as the UAS trajectory. It is worthwhile to emphasize again that UAS and the cable are under effect of no external forces and the system is considered stable and in equilibrium.

The purpose of this test is to see the effect of the length on the cable shape and UAS position while the UAS thrust and pitch remain constant. The UAS thrust's vertical and horizontal components can be defined in the form of thrust and pitch meaning that the UAS propellers produce a constant vertical thrust.

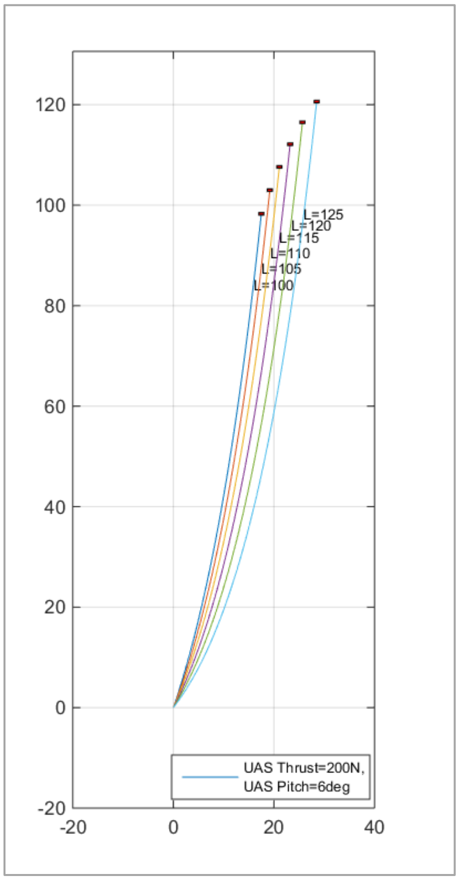

Figure 2.15. Effect of Length on UAS position, Constant Thrust and Pitch

Figure 2.15 shows the UAS position for various cable lengths when the UAS thrust and pitch are kept constant. It can be observed that by keeping all UAS parameters constant, increasing the length increases both the horizontal position and the flying altitude. This helps understand how the cable length affects the position of the UAS.

In addition, using a similar algorithm, the UAS hovering altitude and horizontal offset is plotted as a function of the cable length for the same UAS thrust and pitch in Figure 2.16.The two plots in this figure imply that both the altitude and offset change linearly with the cable length as long as the forces on the upper end of the tether remain constant. This is very important in understanding the reeling in / paying out process. Of course, this analyses can be applied to the cases where the system is in a quasi-static condition, i.e. the changes in the speed of reeling is small to the point that at each time step, one can assume the system is in equilibrium without the presence of inertial forces.

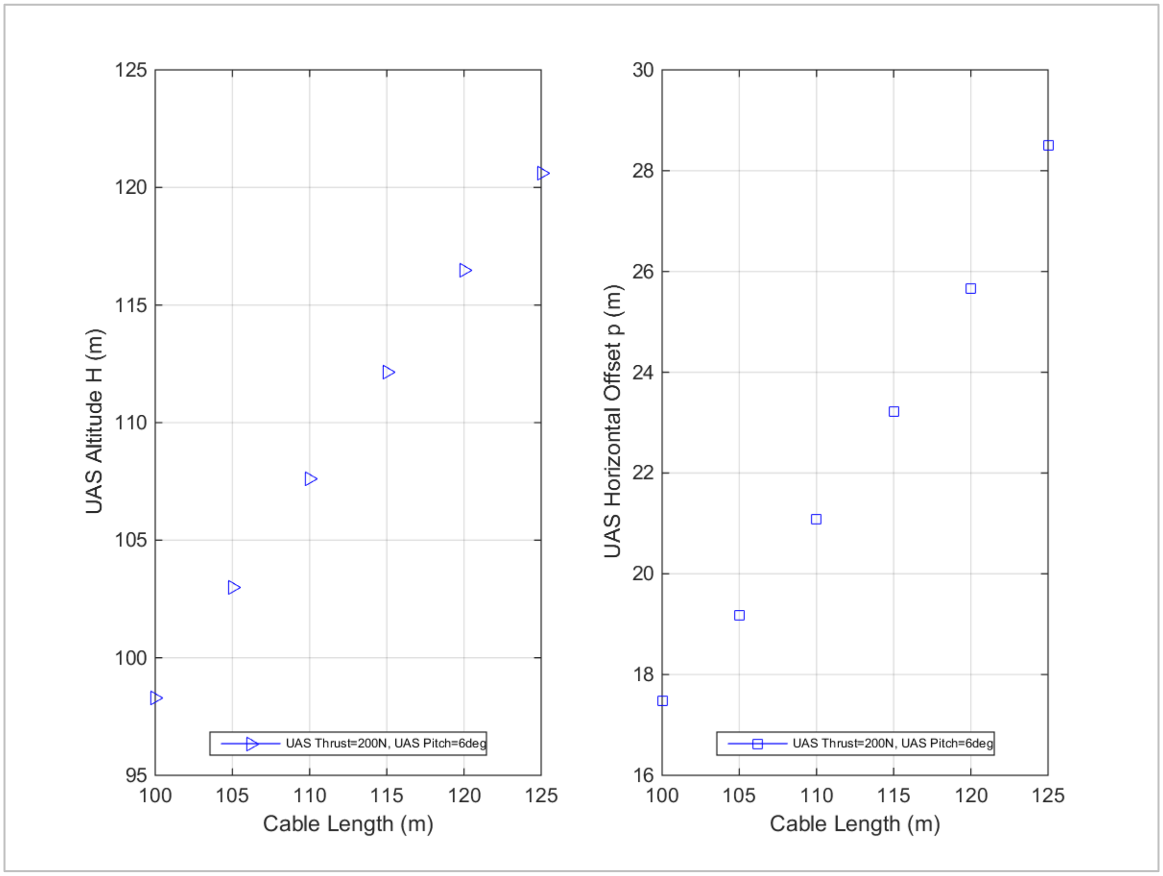

Figure 2.16. UAS Altitude and Offset vs. Cable Length at constant UAS Thrust and Pitch

2.5. Conclusion

In this chapter, a simple user-friendly tool has been developed to help any user model and solve equations of a static catenary. The focus of this chapter, however, was application of this tool to a cable which tethers an unmanned aerial system (UAS) to the ground base station. With the correct implementation of variables and equations, this tool can be applied to any static cable to achieve forces and cable shape functions in the absence of external forces.

As an example, by applying the correct weight per length for a mooring cable it can be applied to a cable connecting a floating object to the seabed. As mentioned before, the model is limited to cases where wind or water current, or generally external forces are not present and where the static condition of the cable is of interest. As another example, one can solve the equations for a certain system configuration based on different weights per length of the cable to observe the effect of cable weight on the outputs of the problem.

In this chapter, two different case studies are presented to demonstrate the capability of the developed algorithm in application to different scenarios and some results are presented. The results are used to observe the cable behavior to different configurations of the system. According to these results one can better understand the sensitivity of the tensions both at the UAS and at the winch to different parameters, including length, and UAS vertical and horizontal position. In addition, it helps understand how the UAS behaves by changing length and what pitch and thrust are required to maintain position. Another outcome of these tests is the effect of cable length on UAS position by keeping UAS thrust and pitch constant.

As mentioned in Section 2.1, the codes are limited in terms of application, in the sense that they are unable to predict the results of a dynamic system. It is also limited to the cases where external forces are absent. However, it is powerful in predicting a quasi-static system where the acceleration is almost zero and as a result, no inertial force is present. With some in-depth mathematical derivation, one may extend this model to one that can analyze external forces such as uniform wind, etc.

Chapter 3.

Two-Dimensional Multi-Body Dynamics Modeling and Analysis of the Tether

As mentioned in Chapter 1, nonlinear analysis and modeling of a flexible cable is not a straight forward task and can become quite complicated in terms of analytical solution [16]. In an effort to approach this problem, a finite segment model is developed and solved in order to investigate the cable's behavior and its interference with the UAS. This chapter is dedicated to explaining how the system is modelled and what tools and methods are used to derive the equations of the system and how they are solved. Some simulations have been run and some observations are presented at the end.

3.1. Methodology

Choo and Casarella [1] have carried out a comprehensive research review on different analytical methods for modelling and simulation of cables. Several methods are discussed along with merits and limitations of each method. The common methods include method of characteristics, finite element method, linearization method, equivalent lumped-mass method, etc. Among these, finite element method is considered the most versatile one since it can be applied almost to any application and physical system [1].

To model a flexible cable, a discretization method is employed. The cable is considered as a discrete system consisting of small segments of the same length as shown in Figure 3.1. The system is a combination of multiple rigid bodies to which multi-body dynamics rules are applied in order to model and solve for the dynamics of the system. Note that in this work, the effect of variable length dynamics is not considered.

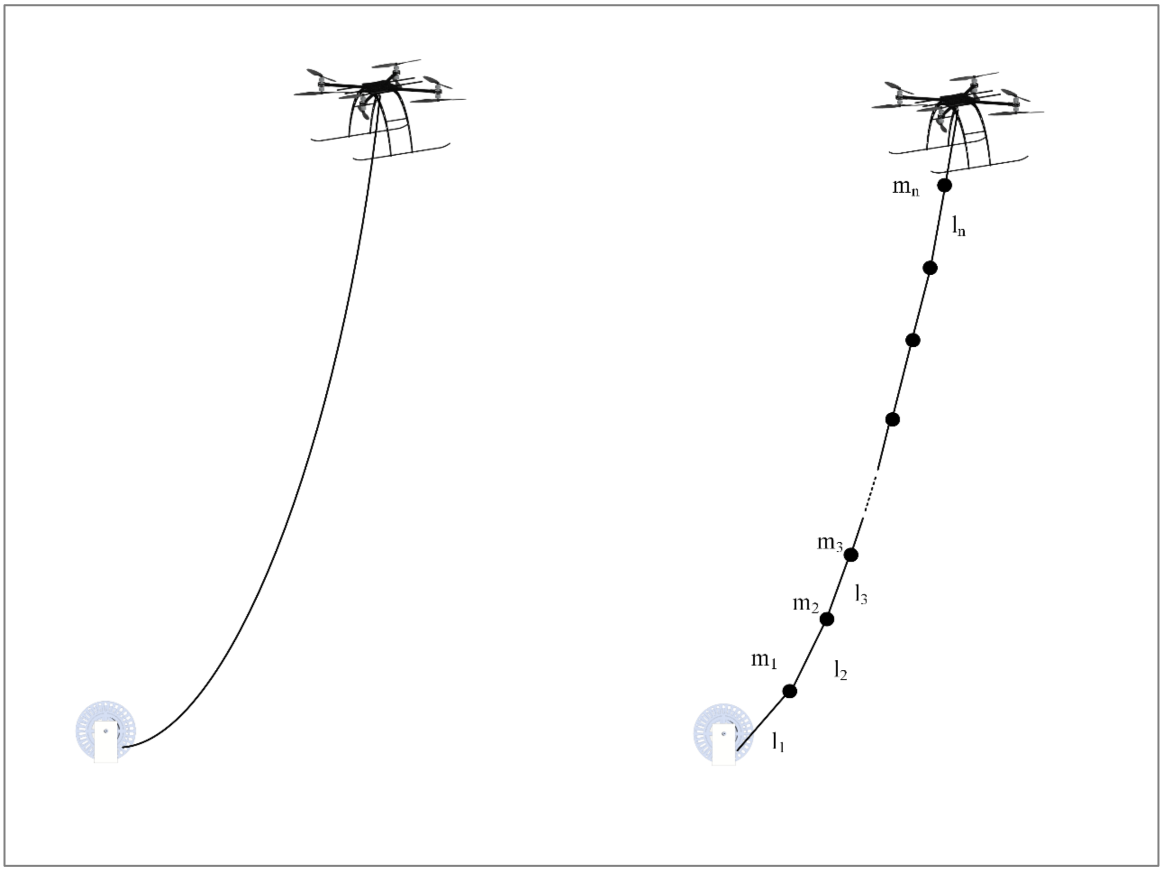

Figure 3.1. Original cable in a hovering operation vs. Discretized model of the same cable

The equation of motion could be developed using Newtonian mechanics. However, in order to develop the model using this method one would need to have full information of the external forces acting on each mass node as well as all initial conditions of the whole system. In this work, considering the fact that the system consists of several bodies in interaction, it was chosen to develop the mathematical model for the multi-body system using principles of Lagrangian Mechanics.

Dynamics of the system can also be studied using Lagrange's method. This method has two main advantages over Newtonian mechanics in deriving the systems equations. Firstly, Lagrange's method is energy-based while Newtonian mechanics is vectorial in nature [16]. Therefore using Lagrange's equations will eliminate the use of vectors in the final stages of the derivations of equations of motion since it uses scalar values rather than vectors. This can make the derivations less complicated and labourintensive. The second advantage is that unlike Newton’s laws, derivation of the equations using Lagrange’s method can be carried out through a certain set of procedures as long as the system and its constraints are well-defined [19].

The derived equations will have to be solved. However as it will be explained in following sections, the equations that are derived for this system are a set of second order ordinary differential coefficients of which include the variable and their rates which makes these equations fairly complicated to solve using common analytical methods. Hence the best options for solving the equation of motions would be numerical integration techniques such as Euler’s method, Trapezoidal method, Modified Euler’s method, Runge-Kutta methods, etc. [16]. In this work, it is chosen to use the Runge-Kutta method due to simplicity of implementation.

3.2. Cable Model

3.2.1. Cable Segment Model

The finite-element method that is used should implement a suitable cable segment model in order to satisfy the physical requirements of the problem as well as to yield an acceptable approximation of the system dynamics. Therefore, it is critical that one defines the initial and boundary conditions accordingly. As mentioned previously, there are several FEA modeling methods for a discrete flexible cable and different methods use different cable segment models. Some of the models discussed by Choo and Casarella [1] are simple pendulum model, thin rod model, spring-mass model, etc. Other models incorporating a cable segment model as several point masses connected by viscoelastic springs are also used in some works such as the research carried out by Buckham et al. [11] and also Williams et al. [31]. Figure 3.2 shows these different segment models used for FEA modeling of flexible cables.

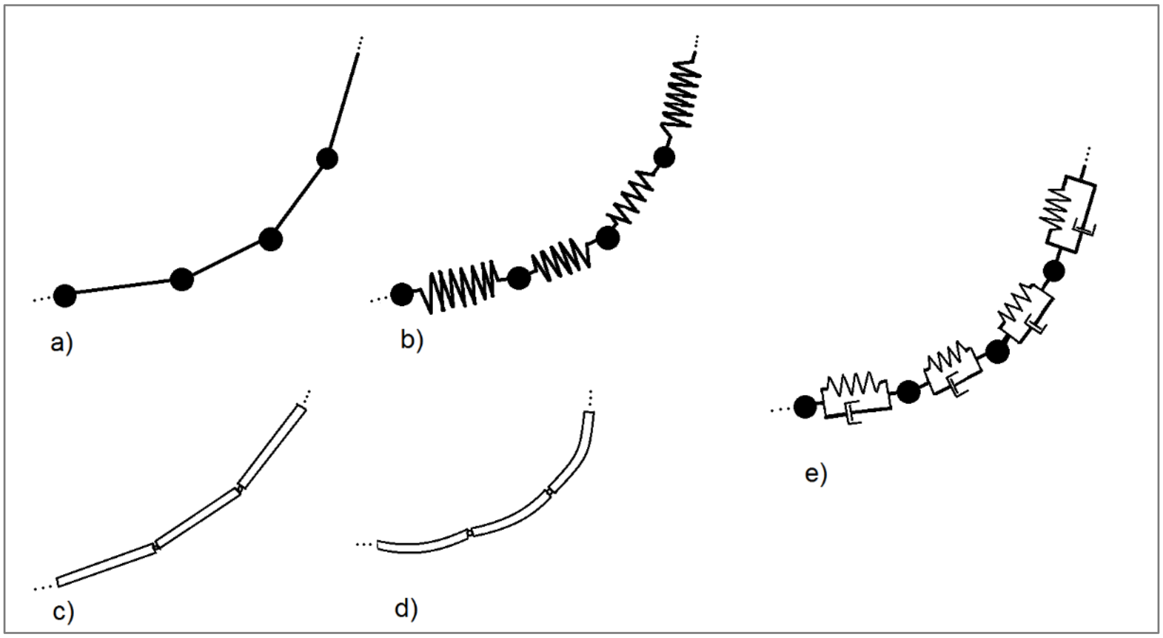

Figure 3.2. Different cable segment models used in literature
a) Simple pendulum, b) Spring-mass, c) Thin-rod [4], d) Curved beam e) Viscoelastic springs [1],[6]

For the purpose of this research, the simple pendulum model is used due to a few reasons. First of all, using this model will help effectively model the wind drag forces since straight cylinders are used. Drag forces on straight cylinders in a uniform flow are well researched and available for implementation into the equations [32]. Moreover, since the cable modeled here has a high stiffness relatively, by neglecting the effect of strain, this model yields an appropriate physical representation of the cable in different scenarios and eliminates the need for using a spring-mass model. The use of simple pendulum model for each segment is also suggested in [19] to reduce computation costs.

Model of the cable is considered as a system of several cable segments connected together in sequence as shown in Figure 3.1. Since it is assumed that the cable does not undergo any bending moment due to the large bending radii occurred in operation scenarios of this work, the connections are modeled as ideal spherical joints; i.e. there are only tension forces present at the connections. However the segments can tolerate bending since they are modeled as a simple pendulum consisting of a weightless rigid rod and a concentrated mass. The cable segment model used in this thesis is shown in Figure 3.3

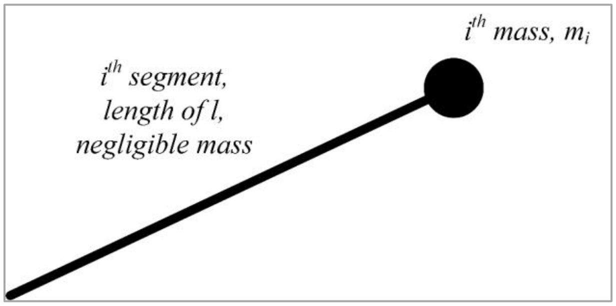

Figure 3.3. Simple Pendulum cable segment

3.2.2. Coordinate System and Model Set-up

The two dimensional model of the cable is developed in a Cartesian coordinate system labeled *XZ*-plane. *XZ* is an inertial frame of reference in which the position of each node on the cable is represented. An inertial or namely Newtonian frame of reference is defined as a reference frame which does not undergo any acceleration including rotational or translational [20]. The origin of this system is at the cable's connection to the ground station, the *X* axis is parallel to the ground and the *Z* axis is perpendicular to the ground pointing upwards to represent the height of each node. In addition to this inertial coordinate system, a local coordinate system called *xz* is used which is defined separately for each and every segment in a way that *x*-axis is always parallel to the cable segment towards the mass end of the segment, called tangent coordinate from here on, and the *z*-axis is perpendicular to the cable segment called normal coordinate hereafter. This configuration is shown in Figure 3.4.

Now, length of the i^{th} segment is labeled by l_i and the angle each segment makes with the horizon is called θ_i. Having this model in mind helps simply express the position of each mass node on the cable model as follows:

$$\begin{cases} X_i = \sum_{j=1}^{i} l_j \cos\theta_j \\ Z_i = \sum_{j=1}^{i} l_j \sin\theta_j \end{cases} \qquad (i = 1, 2, 3, \dots, n) \qquad (3.1)$$

where X_i and Z_i are the coordinates of the i^{th} mass node.

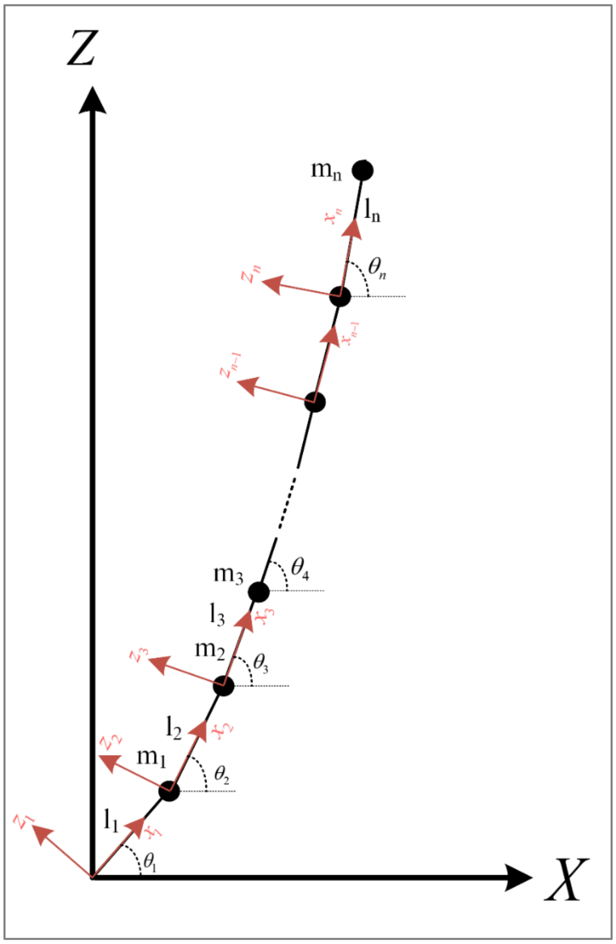

Figure 3.4. Global and local coordinate systems of the discretized cable

Since there are two different frames, one should be able to convert any vector expressed in these frames to one another using a transformation matrix. Now, consider a vector represented in the inertial reference frame by *(U,W)* and in the local *xz* coordinate system by *(u,w)*. A transformation matrix can be used to convert the vector representation to/from the inertial frame of reference. The equation below shows the transformation matrix used to convert any vector in the *xz* coordinate *(u,w)* to its *XZ* representation, i.e. *(U,W)*.

$$\begin{bmatrix} U_i \\ W_i \end{bmatrix} = \mathbf{R}_{\theta_i} \times \begin{bmatrix} u_i \\ w_i \end{bmatrix} \tag{3.2}$$

where the transformation matrix from the body frame to the inertial frame is expressed as:

$$\mathbf{R}_{\theta_i} = \begin{bmatrix} \cos\theta_i & -\sin\theta_i \\ \sin\theta_i & \cos\theta_i \end{bmatrix} \tag{3.3}$$

It is clear that the opposite of this transformation could be done using the inverse of the matrix $\mathbf{R}_{\theta_i}$.

Using Equations (3.1) - (3.3) and since each mass is located at *(l_j,0)* in its own local frame, one can write the position of each mass node in the inertial frame as:

$$\begin{bmatrix} X_i \\ Z_i \end{bmatrix} = \sum_{j=1}^{i} \mathbf{R}_{\theta_i}{}^{-1} \begin{bmatrix} x_j \\ z_j \end{bmatrix} = \sum_{j=1}^{i} \mathbf{R}_{\theta_i}{}^{-1} \begin{bmatrix} l_j \\ 0 \end{bmatrix} \qquad (i = 1, 2, 3, \ldots, n) \tag{3.4}$$

Now that the model's geometry is set up and the coordinate systems properly, different terms for Lagrange's equations can be used to derive the equations of motion as it is presented in Section 3.3.

3.3. Derivation of Equations of Motion Using Lagrange's Equations

As explained in Section 3.1, Lagrange's equations are used for this system in order to derive its equations of motion. It is important to be careful with the degrees of freedom of the system in the process of deriving the Lagrangian for a multi-body system. According to Greenwood, the number of degrees of freedom (DOF) of a system is equal to the number of independent coordinates used to specify the configuration minus the number of independent equations of constraint [34][33]. It must be noted that the DOF of a system is a characteristic of the system and does not depend on the coordinates chosen to represent its motion.

As an example, the system discussed in Section 3.2.1, can be defined by only two coordinates; examples would include *X* and *Z* in a Cartesian system or *r* and *θ* in a polar system. Consider only one segment as shown in Figure 3.5. In this segment there are two coordinates defining the position of the mass; however, this system has only one DOF since these two coordinates are not independent, i.e. they can be related to each other using a constraint equation.

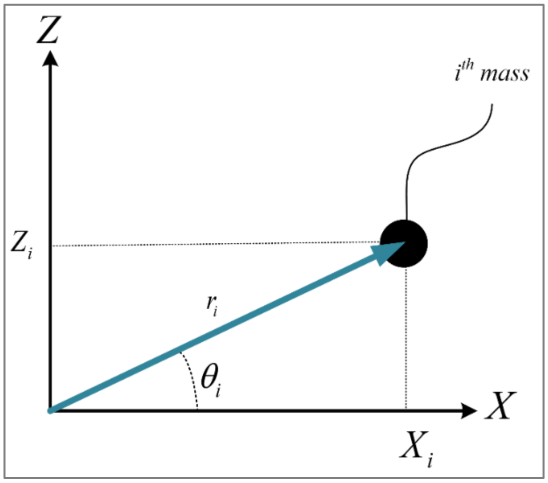

Figure 3.5. Cartesian vs. Polar coordinates for a cable segment

The equation of constraint, for a cable segment of length l_i could be written as follows which reduces the DOF to one.

$$l_i^2 = \left(X_i - X_{i-1}\right)^2 + \left(Z_i - Z_{i-1}\right)^2 \qquad (i = 1, 2, \ldots, n) \qquad (\ 3.5\ )$$

In the same way one can see that the mass could be located using only the coordinate *θ* in polar coordinates since the length of the pendulum is constant. Hence one can verify that the DOF is the same regardless of the coordinate system chosen. In the multi-body model of the cable, since *n* cable segments are used, the DOF of the system will be *n.* This could also be verified in another way. Since there are *n* segments and each segment is defined by 2 coordinates (*X* and *Z*), the system needs *2n* coordinates overall to be defined. On the other hand, there are *n* constraint equations as shown in Equation (3.5) which make the DOF equal to *2n-n*; i.e. *n* degrees of freedom. Therefore the minimum number of generalized coordinates needed for the system to be defined thoroughly is *n*.

In order to simplify the derivation of the equations using Lagrange’s equations, one should use generalized coordinates q_i in the derivations [34]. In this analysis, θ_i could be used as the generalized coordinates in the development of Lagrange’s equations.

The Lagrangian of a system is shown by $\mathscr{L}$ and is defined as [23]:

$$\mathscr{L} = T - U \tag{3.6}$$

where *T* and *U* are the system’s kinetic and potential energy, respectively.

Lagrange’s equations for a multi-body system consisting of *n* rigid bodies can be written in the following form [34]:

$$\frac{d}{dt}\left(\frac{\partial \mathscr{L}}{\partial \dot{q}_i}\right) - \frac{\partial \mathscr{L}}{\partial q_i} = Q_i' \qquad (i = 1, 2, \ldots, n) \tag{3.7}$$

where q_i’s are the generalized coordinates used to define the system and Q_i'’s are applied or generalized forces that are not derivable from the potential energy [34].

Since in this system, the potential energy is independent of velocity, one can reduce the Lagrange’s equations to the following format:

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{q}_i}\right) - \frac{\partial T}{\partial q_i} = Q_i' - \frac{\partial U}{\partial q_i} = Q_i \qquad (i = 1, 2, \ldots, n) \tag{3.8}$$

where *Q_i’s* are generalized forces which include applied forces derived from the potential energy. Therefore, by making the right selection of the segment angles as the generalized coordinates, Equation (3.8) can be expressed as:

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{\theta}_i}\right)-\frac{\partial T}{\partial \theta_i}=Q_i \qquad (i=1,2,\ldots,n) \tag{3.9}$$

Now that Lagrange’s equations are established for the system, it is critical that all the energy terms are derived correctly in order to be incorporated into the main equations.

3.3.1. Kinetic Energy and Its Derivatives

The kinetic energy for a system of *n* particles can be stated as [20]:

$$T=\frac{1}{2}\sum_{k=1}^{n} m_k \overrightarrow{V_k}^2 \tag{3.10}$$

where V_k represents the velocity vector for the k^{th} particle in the inertial reference frame.

Therefore, according to the choice of reference frame, one can express the squared velocity vector as:

$$\vec{V}_k^T \vec{V}_k = \left|\vec{V}_k\right|^2 = \dot{X}_k^{\,2}+\dot{Z}_k^{\,2} \qquad (k=1,2,\ldots,n) \tag{3.11}$$

However, this equation should be written in terms of the generalized coordinates. Differentiating and arranging Equation (3.1)yields:

$$\begin{cases} \dot{X}_i = -\sum_{j=1}^{i} l_j \dot{\theta}_j \sin\theta_j \\ \dot{Z}_i = \quad \sum_{j=1}^{i} l_j \dot{\theta}_j \cos\theta_j \end{cases} \qquad (i=1,2,3,\ldots,n) \tag{3.12}$$

By substituting this equation into the velocity vector equation, one can write:

$$\overrightarrow{V_k}^2 = \dot{X}_k^2 + \dot{Z}_k^2 = \sum_{i=1}^{k}\sum_{j=1}^{k}\left[\dot{\theta}_i\dot{\theta}_j l_i l_j\left(\sin\theta_i\sin\theta_j + \cos\theta_i\cos\theta_j\right)\right] \qquad (3.13)$$

which can then be reduced to

$$\overrightarrow{V_k}^2 = \sum_{i=1}^{k}\sum_{j=1}^{k} l_i l_j \dot{\theta}_i\dot{\theta}_j \cos(\theta_j - \theta_i) \qquad ,(k = 1, 2, \ldots, n) \qquad (3.14)$$

where l_i is the length of the i^{th} segment.

Now, having derived the velocity in terms of the generalized coordinates one can write the kinetic energy of the system as:

$$T = \frac{1}{2}\sum_{k=1}^{n}\sum_{i=1}^{k}\sum_{j=1}^{k} m_k l_i l_j \dot{\theta}_i\dot{\theta}_j \cos(\theta_j - \theta_i) \qquad (3.15)$$

where m_k is the mass of the node at the end of the i^{th} segment.

The Lagrange's equations shown in Equation (3.9) requires finding some of the derivatives of the kinetic energy. Differentiating the kinetic energy with respect to the generalized coordinate and also with respect to its time derivative respectively yields:

$$\frac{\partial T}{\partial q_i} = \frac{\partial T}{\partial \theta_i} = \sum_{k=i}^{n}\sum_{j=1}^{k} m_k l_i l_j \dot{\theta}_i\dot{\theta}_j \sin(\theta_j - \theta_i) \qquad ,(i = 1, 2, \ldots, n) \qquad (3.16)$$

$$\frac{\partial T}{\partial \dot{q}_i} = \frac{\partial T}{\partial \dot{\theta}_i} = \sum_{k=i}^{n}\sum_{j=1}^{k} m_k l_i l_j \dot{\theta}_j \cos(\theta_j - \theta_i) \qquad ,(i = 1, 2, \ldots, n) \qquad (3.17)$$

It is very critical to note that in equations (3.16) and (3.17), the summation over k, will start at the i^{th} component. The reason is that the inner summation covers the 1st to the k^{th} terms. Therefore, for all *k's* smaller than *I,* there are no θ_i or $\dot{\theta}_i$ present in the summation; hence, the derivative with respect to these terms are zero.

Following this, one can write the first term on the left hand side of the Lagrange's equation in Equation (3.9) as :

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{q}_i}\right)=\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{\theta}_i}\right)=\frac{d}{dt}\left(\sum_{k=i}^{n}\sum_{j=1}^{k} m_k l_i l_j \dot{\theta}_j \cos(\theta_j-\theta_i)\right)$$

$$=\sum_{k=i}^{n}\sum_{j=1}^{k}\Big[m_k l_i l_j\left(\ddot{\theta}_j\cos(\theta_j-\theta_i)-\dot{\theta}_j(\dot{\theta}_j-\dot{\theta}_i)\sin(\theta_j-\theta_i)\right)\Big] \qquad ,(i=1,2,\ldots,n) \qquad (3.18)$$

And finally substituting Equations (3.16) and (3.18) into (3.9) yields the following equation as the Lagrange's equation for this system:

$$\sum_{k=i}^{n}\sum_{j=1}^{k}\Big[m_k l_i l_j\left(\ddot{\theta}_j\cos(\theta_j-\theta_i)-\dot{\theta}_j^{\,2}\sin(\theta_j-\theta_i)\right)\Big]=Q_i \qquad ,(i=1,2,\ldots,n) \qquad (3.19)$$

3.3.2. Potential Energy

With the assumption of no strain or elasticity in the links, the only potential energy present in this multi-body system is of gravitational form. Gravitational potential energy is usually measured with respect to a datum line at which *U=0*. Assume a datum line on the ground, that is at *Z=0*. Then for the system shown in Figure 3.4, the potential energy could be stated as:

$$U=\sum_{j=1}^{n} m_j Z_j g \qquad (3.20)$$

3.3.3. Derivation of Generalized Forces

During the derivation of Lagrange's equations, as seen in Equation (3.19), one needs to evaluate the generalized forces associated with the system's generalized coordinates [33]. In his book, Dynamics of Multibody Systems, Shabana states that these generalized forces can be introduced by applying the principle of Virtual Work which can be used both in static and dynamic analysis of a multibody system [35]. Quisenberry has also applied this method in their thesis for an aircraft towing system [19].

Consider a system consisting of *k* rigid bodies, whose centres of mass are given by *3k* Cartesian coordinates, $x_1, x_2, \ldots, x_{3k}$. Now consider force vectors $\boldsymbol{F}_1, \boldsymbol{F}_2, \ldots, \boldsymbol{F}_k$ are applied to these bodies at these coordinates and moments $M_1, M_2, \ldots, M_k$ are applied to these rigid bodies, and the system goes through infinitesimal position and angular change of $\boldsymbol{\delta r}_1, \boldsymbol{\delta r}_2, \ldots, \boldsymbol{\delta r}_k$ and $\delta\phi_1, \delta\phi_2, \ldots, \delta\phi_k$, respectively. Then the virtual work done by these forces on the system can be written as:

$$\delta W = \sum_{j=1}^{3k} F_{x_j} \delta x_j + \sum_{i=1}^{k} M_i \delta\phi_i \qquad (3.21)$$

Using the principle of virtual work, one can obtain the generalized forces for a discretized cable in terms of the external forces. According to Quisenberry [19], for the system defined in this work, the generalized forces which cannot be derived from the potential energy can be written as:

$$Q_i' = \sum_{j=1}^{n} \left[F_{X_j} \frac{\partial X_j}{\partial \theta_i} + F_{Z_j} \frac{\partial Z_j}{\partial \theta_i} + M_{\theta_j} \frac{\partial \theta_j}{\partial \theta_i} \right] \qquad (i = 1, 2, \ldots n) \qquad (3.22)$$

Also, since

$$Q_i = Q_i' - \frac{\partial U}{\partial \theta_i} \qquad (i = 1, 2, \ldots n) \qquad (3.23)$$

by subtracting the derivative of the potential energy, one can express the generalized forces for this discrete cable system as:

$$Q_i = \sum_{j=1}^{n} \left[F_{X_j} \frac{\partial X_j}{\partial \theta_i} + \left(F_{Z_j} - m_j g \right) \frac{\partial Z_j}{\partial \theta_i} + M_{\theta_j} \frac{\partial \theta_j}{\partial \theta_i} \right] \qquad (i = 1, 2, \ldots n) \tag{3.24}$$

After calculating the derivative of the X and Z coordinates with respect to the generalized coordinates the generalized forces can be simplified to:

$$Q_i = \sum_{j=i}^{n} \left[\left(F_{Z_j} - m_j g \right) l_i \cos \theta_i - F_{X_j} l_i \sin \theta_i \right] + M_{\theta_i} \qquad (i = 1, 2, \ldots n) \tag{3.25}$$

Combining with the derived Lagrange’s equation (3.19), one can write:

$$\begin{aligned} &\sum_{k=i}^{n} \sum_{j=1}^{k} \left[m_k l_i l_j \left(\ddot{\theta}_j \cos(\theta_j - \theta_i) - \dot{\theta}_j^{\,2} \sin(\theta_j - \theta_i) \right) \right] \\ &\quad = \sum_{j=i}^{n} \left[\left(F_{Z_j} - m_j g \right) l_i \cos \theta_i - F_{X_j} l_i \sin \theta_i \right] + M_{\theta_i} \qquad , \ (i = 1, 2, \ldots, n) \end{aligned} \tag{3.26}$$

Equation (3.26) actually is a set of *n* second-order ordinary differential equations which altogether represent the equations of motion of the cable in terms of *n* unknown variables $\boldsymbol{\theta}_i$ and their two derivatives $\dot{\boldsymbol{\theta}}_i$ and $\ddot{\boldsymbol{\theta}}_i$. Defining the applied external forces as well as proper initial conditions, one can solve this set of equations for the cable segment’s angular acceleration which will fully define the dynamic behavior of the system. Note that these equations are highly nonlinear and cannot be solved using common analytical methods. Therefore, a suitable numerical technique should be employed for solving the equations of motion of the system.

Looking at the equations of motion they can be reduced to a matrix/vector format for simpler manipulation and representation. The generalized forces can be implemented in a vector of a generalized force *n×1* vector $\mathbf{Q}$ where:

$$\mathbf{Q}_i = \sum_{j=i}^{n}\left[\left(F_{Z_j} - m_j g\right) l_i \cos\theta_i - F_{X_j} l_i \sin\theta_i\right] + M_{\theta_i} \qquad (i = 1, 2, \ldots n) \tag{3.27}$$

Also, after manipulation and simplification of the summations on the left hand side of Equation (3.26), the following matrices can be defined to represent the Lagrange's equations similar to Quisenberry's work in [19]:

$$[\mathbf{A}]_{ij} = \sum_{k=\max\{i,j\}}^{n} m_k l_i l_j \cos\left(\theta_i - \theta_j\right) \qquad (i, j = 1, 2, \ldots n) \tag{3.28}$$

$$[\mathbf{B}]_{ij} = \sum_{k=\max\{i,j\}}^{n} m_k l_i l_j \sin\left(\theta_i - \theta_j\right) \qquad (i, j = 1, 2, \ldots n) \tag{3.29}$$

where matrix $\mathbf{M}$ is the mass distribution matrix for the multibody system and is defined as:

$$[\mathbf{M}]_{\mathrm{ij}} = \sum_{k=\max\{i,j\}}^{n} m_k \qquad (i, j = 1, 2, \ldots n) \tag{3.30}$$

Using these the matrices introduced in Equations (3.27) – (3.30), Equation (3.26) can be expressed as:

$$\mathbf{A}\ddot{\boldsymbol{\theta}} + \mathbf{B}diag\left\{\dot{\boldsymbol{\theta}}\right\}\dot{\boldsymbol{\theta}} = \mathbf{Q} \tag{3.31}$$

where **A** and **B** are *n* by *n* matrices and functions of θ_i's and **Q** is a vectors with *n* rows and a function of θ_i and $\dot{\theta}_i$. Also, $\dot{\boldsymbol{\theta}}$ and $\ddot{\boldsymbol{\theta}}$ are *n×1* vectors consisting of $\dot{\theta}_i$ and $\ddot{\theta}_i$ respectively. Considering these notations, calculation of the external forces will conclude the derivation of Lagrange's equations in the next section.

3.4. External Forces

In general, cables used in most applications in industry are subject to certain types of external forces. As an example, consider a mooring cable connecting a ship to a towed ROV or an aircraft towing an autonomous UAV. In all these cases as well as the case of a tethered flying vehicle for this work, external forces e.g. those caused by wind or water current may be applied to the system. This might significantly affect the cable's shape function as well as its dynamic behaviour and tensions acting on the base station and the UAS. Thus it is important to derive and implement these forces into the system's equations for the purpose of this analysis.

As Equation (3.31) suggests, the generalized force vector **Q** should be defined so that one can solve the Lagrange's equations. Equation set (3.27) shows that this generalized force vector, consists of the external forces and moments exerted on each cable segment namely F_{X_i} , F_{Z_i} and M_{θ_i} . Note that all components of the generalized forces are inherently moments rather than linear forces in this analysis since the selected generalized coordinates are angles.

In this section, with the assumption of uniform 2D wind forces being the only external load on the system, the equations for the external forces applied to each cable segment are derived to form the generalized force vector; then the results are substituted into the Lagrange's Equations.

The drag model used in this research is borrowed from the work of Hoerner [32] in which the effect of flow around circular cylinders on lift and drag coefficients in a uniform fluid flow is studied. The cross-flow principle or cosine principle – as labeled by Hoerner – is mostly valid for a straight cylinder of circular cross section in a subcritical flow [16], [32]. The principle can also be confirmed by the experimental work done by Poulin and Larsen [36]. Therefore it deems appropriate to model the simplified drag and lift forces on the cable segments of the system using this principle.

Cross-Flow Principle

For the purpose of modelling the external forces acting on the cable segments due to wind, each cable segment is considered as a straight cylindrical rod in a uniform flow field with a velocity of *V*. Although the model set up constrains the motion of the cable in a 2D plane, this wind model predicts the external forces appropriately. The rod will be subjected to lift and drag forces due to the impact of fluid on its surface. The lift forces consist of pressure lift and the drag forces consist of pressure and friction forces acting on the surface.

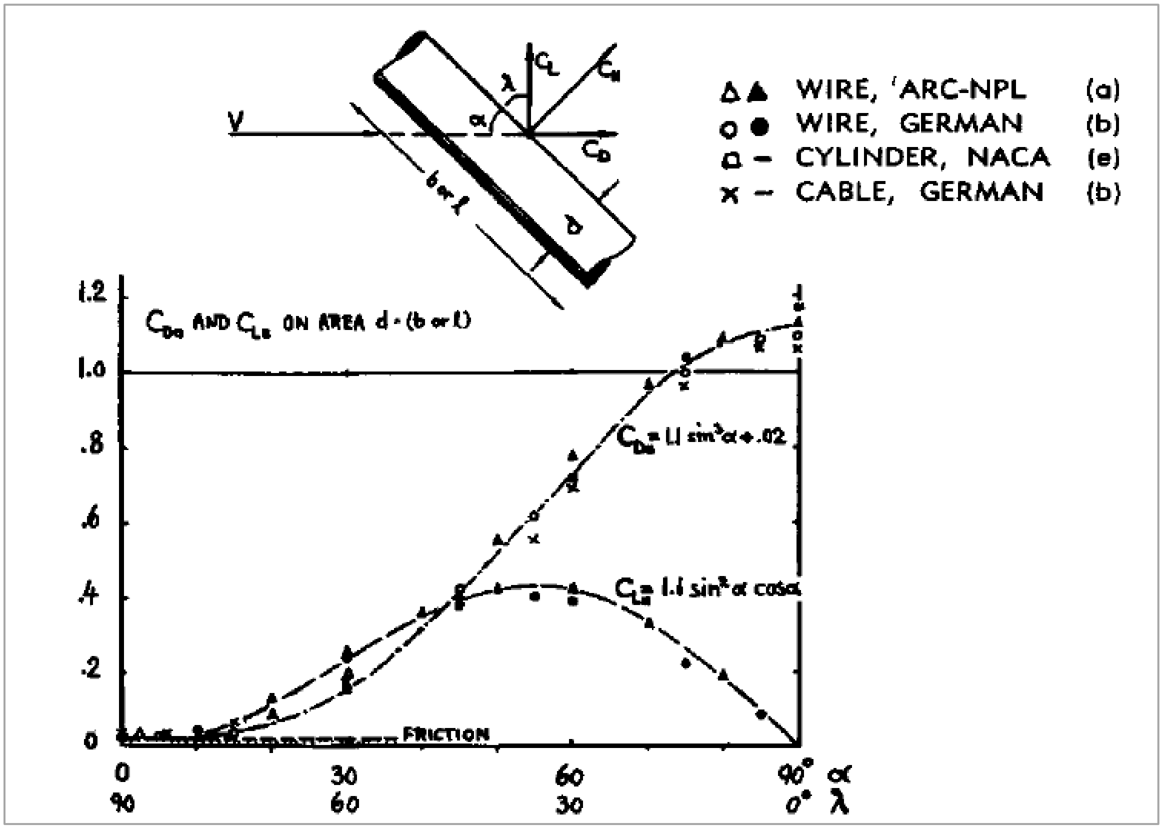

Figure 3.6. Drag and Lift coefficients of circular cylinders, wires and cables [32]
"The cable is inclined at an angle of *α* against the direction of flow - at subcritical Reynolds numbers"

Figure 3.6 shows the experimental data and the curve fit to these data. From here on, this drag/lift models will be adapted into the equations. Assume a cylinder in a uniform wind flow with velocity of *V*. Defining the angle of attack *α* as the angle between the wind velocity vector and the longitudinal axes of the cylinder as shown in Figure 3.6, assuming tangential pressure drag to be negligible, and neglecting tip vortices, Hoerner states that the lift and drag coefficients for this configuration of a slender cylinder in the uniform subcritical wind flow can be calculated as [32]:

$$\begin{cases} C_{D\square} = C_{D,basic}\left(\sin^3\alpha\right) + \pi C_f = C_{D,basic}\left(\cos^3\lambda\right) + \pi C_f \\ C_{L\square} = C_{D,basic}\left(\sin^2\alpha\cos\alpha\right) = C_{D,basic}\left(\cos^2\lambda\sin\lambda\right) \end{cases} \tag{3.32}$$

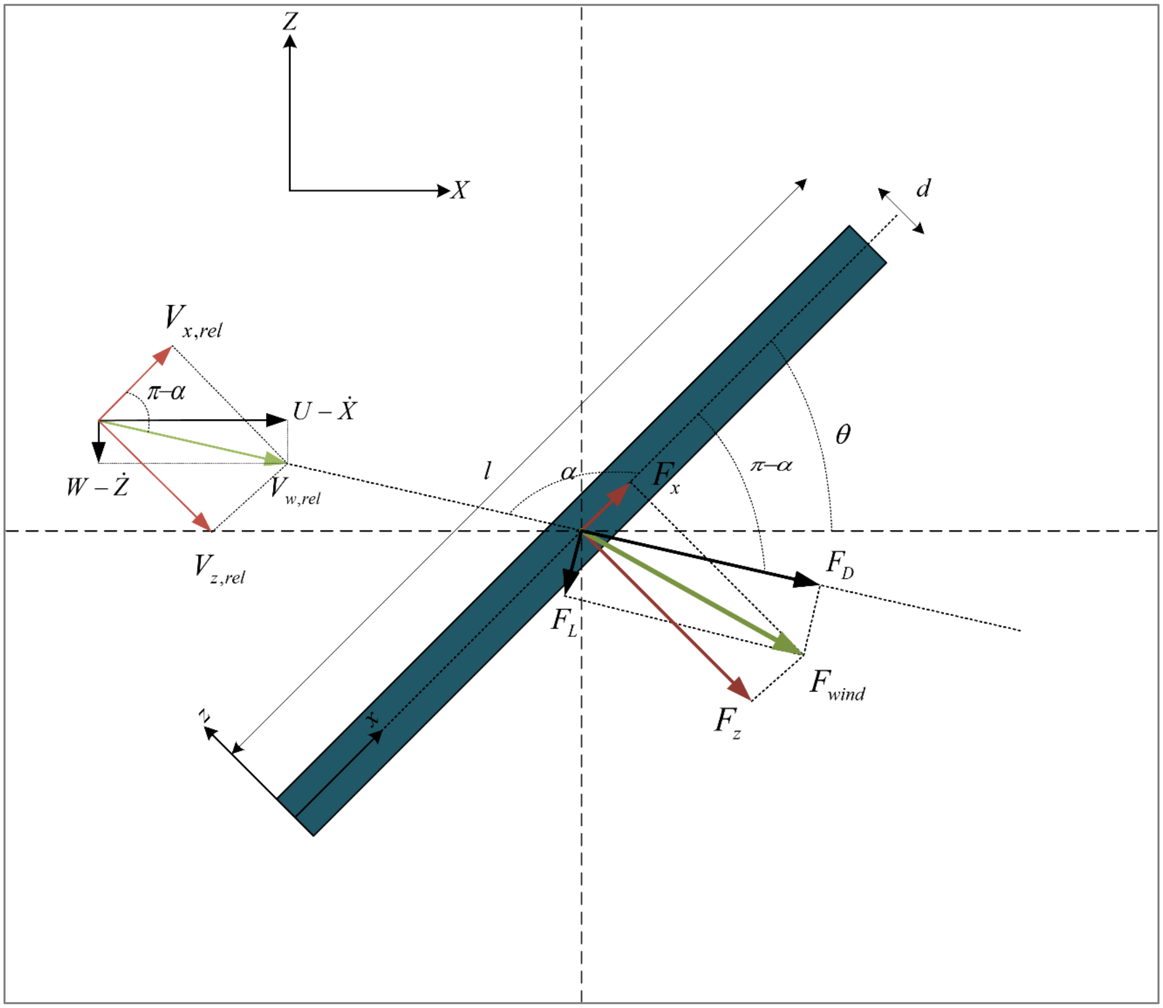

Figure 3.7. Drag model on a cable segment

A cable segment used in this work is shown in Figure 3.7. For a more general case, assume the wind is acting on the cylinder with a speed of V_{w0} with horizontal and vertical velocity components of U and W as:

$$V_{w0} = \begin{pmatrix} U \\ W \end{pmatrix} \tag{3.33}$$

Assuming the cylinder is small enough so that one can neglect the difference in translational speeds of its different points, the wind's velocity vector relative to the segment, can be defined as:

$$V_{w,rel} = \begin{pmatrix} U - \dot{X} \\ W - \dot{Z} \end{pmatrix} \tag{3.34}$$

Note that angle of attack α is defined as the angle between $V_{w,rel}$ with the positive direction of axes x as shown in Figure 3.7. The relative air velocity causes two forces on the segment; a drag force and a lift force, namely F_D and F_L. These forces can be calculated as:

$$\begin{cases} F_D = \frac{1}{2}\rho d l V_{w,rel}^2 C_D \\ F_L = \frac{1}{2}\rho d l V_{w,rel}^2 C_L \end{cases} \tag{3.35}$$

Therefore substituting the drag and lift coefficients derived by Hoerner, i.e. Equation (3.32), namely C_D and C_L, and reordering and simplifying the equations yields the forces exerted by the wind on the cable segment in the body frame as:

$$\begin{pmatrix} F_x \\ F_z \end{pmatrix} = \begin{pmatrix} \frac{1}{2}\rho d l \pi C_f V_{x,rel}\sqrt{{V_{x,rel}}^2 + {V_{z,rel}}^2} \\ \frac{1}{2}\rho d l V_{z,rel}\left(\pi C_f \sqrt{{V_{x,rel}}^2 + {V_{z,rel}}^2} + C_P \left|V_{z,rel}\right|\right) \end{pmatrix} \tag{3.36}$$

where C_p is called the pressure coefficient for the cylinder and the relative wind velocity tangential and normal components in the segment local frame, $V_{x,rel}$ and $V_{z,rel}$ can be determined by rotating the relative wind velocity in the inertial frame, i.e. $V_{w,rel}$ as shown in Equation (3.37) using the rotation matrix $\mathbf{R}_{\theta_i}$ defined in Equation (3.3).

$$\begin{pmatrix} V_{x,rel_i} \\ V_{z,rel_i} \end{pmatrix} = \mathbf{R}_{\theta_i}{}^{-1} \times \begin{pmatrix} U - \dot{X}_i \\ W - \dot{Z}_i \end{pmatrix} \tag{3.37}$$

It is also worthwhile to note that the tangential and normal forces in each segment's local frame shown in Equation **Error! Reference source not found.** can be transformed back and forth to the inertial frame using the same rotation matrix; that is:

$$\begin{pmatrix} F_{X_i} \\ F_{Z_i} \end{pmatrix} = \mathbf{R}_{\theta_i} \times \begin{pmatrix} F_{x_i} \\ F_{z_i} \end{pmatrix} \tag{3.38}$$

3.5. UAS Dynamics

The UAS's motion and dynamics can affect the dynamic of the cable significantly, since the interface between the cable and the UAS is assumed to be an ideal spherical joint which cancels moments about its axis but can exert forces and be affected by cable tension. As a result the equations of motion for both the UAS and the cable are coupled and should be derived and solved simultaneously. In this section, the UAS's dynamic equations will be derived to be implemented into the system equations which are previously derived in Equation set (3.26).

3.5.1. Implementation of UAS Segment into the Equations

To accomplish modeling the cable considering the vehicle's dynamics, the UAS can be considered as an additional cable segment with the mass of the UAS lumped at the end, which will form the *n+1*th element of the multibody system. This element is in fact a virtual element extending from the end point of the last cable segment (*n*th element), i.e. the cable connection point to the center of mass of the UAS. It will serve as another cable segment with a different mass and length as others as shown in Figure 3.8. The mass for this virtual segment will be the mass of the UAS and the length will be the distance between the connection point and the UAS center of mass. In addition, it is important to model the drag forces on this virtual element in compliance with the appropriate drag forces modeled for the UAS. This approach is the best and simplest way of implementing the UAS's equations of motions as it is also performed in a similar work for a towed aerial vehicle by Quisenberry [19]. With the assumption that the connection interface is a torque cancelling joint, such as a simple spherical joint – that is the two segments apply only forces on the neighbor segments and are free to rotate under any moment, this selection ensures that implementation of this element can be completed in consistence with the previous modeling approach. The only difference of this element with the rest of the system is in the length and mass of the segment as well as addition of the UAS mass moment of inertia.

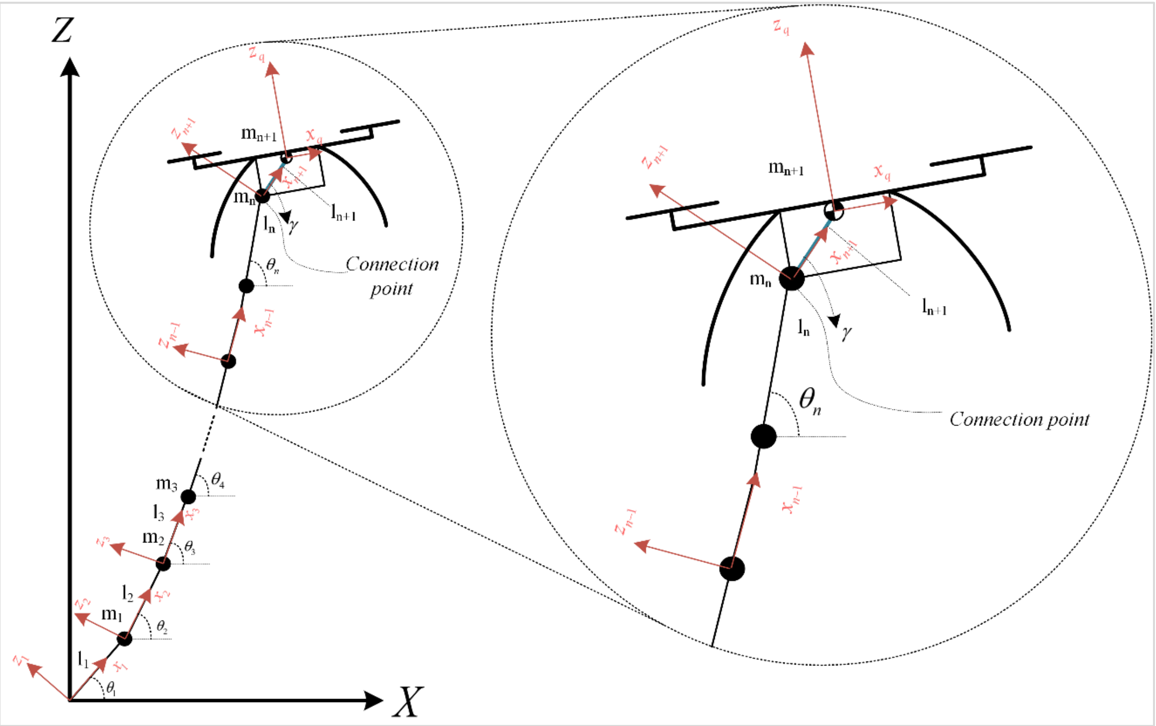

Figure 3.8. UAS Coordinate Setup
Note that the length between the connection point of the cable and the UAS's center of mass is considered as an extra segment

Moreover, to be able to define the orientation of the vehicle properly, a local reference frame $x_q z_q$ needs to be defined with the origin positioned at the center of mass of the UAS, x_q axes pointing towards the forward face of the UAS and z_q axes pointing towards the top of the UAS as illustrated in Figure 3.8. Using this configuration, vectors in the UAS segment coordinate system, i.e. $x_{n+1} z_{n+1}$ can be transformed to the $x_q z_q$ coordinate system by using a transformation matrix $\mathbf{R}_\gamma$ as defined below:

$$\mathbf{R}_\gamma = \begin{bmatrix} \cos\gamma & \sin\gamma \\ -\sin\gamma & \cos\gamma \end{bmatrix} \qquad (3.39)$$

$$\begin{bmatrix} x_{n+1} \\ z_{n+1} \end{bmatrix} = \mathbf{R}_\gamma \begin{bmatrix} x_q \\ z_q \end{bmatrix} \qquad (3.40)$$

In this system, the connection point is exactly below the center of mass, making γ 90 degrees. This causes the UAS coordinates to become equal to:

$$\begin{bmatrix} x_q \\ z_q \end{bmatrix} = \begin{bmatrix} -z_{n+1} \\ x_{n+1} \end{bmatrix} \qquad (3.41)$$

Addition of this segment will modify the derivation of the Lagrange's equation in the sense that it will add another term to the system. As this segment has moment of inertia about the origin of $x_{n+1} z_{n+1}$ unlike the cable segments, it causes a rotational kinetic energy to appear in the derivations. Since planar motion of the system are being studied, one can assume that the moment of inertia of the UAS consists of a moment of inertia about the y axes only as shown in Equation (3.42).

$$\bar{I}_{CG} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & \bar{I}_{yy} & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{3.42}$$

This matrix represents the inertia matrix of the UAS about its center of gravity, however, in the calculation of kinetic energy, the inertias should be calculated about the instantaneous center of rotation, i.e. the connection point which is also the origin of the *n+1th* segment. This requires the inertia matrix to be calculated about this point using parallel axis theorem [37], [38]. Then, inertia about the origin of the UAS segment coordinate system will be:

$$I_{yy} = \bar{I}_{yy_{CG}} + m_U l_{n+1}^2 \tag{3.43}$$

where m_U represents the UAS's mass and l_{n+1} is the length of the *n+1th* segment. Therefore the inertia matrix can be written as:

$$I = \begin{bmatrix} 0 & 0 & 0 \\ 0 & I_{yy} + m_U l_{n+1}^2 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{3.44}$$

Then the kinetic energy of the multibody system including the UAS segment which consists of a translational and a rotation term could be written as:

$$T = T_{trans.} + T_{rot.} \tag{3.45}$$

where the translational kinetic energy is calculated in the same way as in Equation **Error! Reference source not found.** with the addition of a *n+1th* segment and the rotational kinetic energy could be written as:

$$T_{rot.} = \sum_{j=1}^{n+1}\left(T_{rot.}\right)_j = \frac{1}{2}\sum_{j=1}^{n+1}\vec{\boldsymbol{\omega}}_j{}^{\mathbf{T}}\mathbf{I}_j\vec{\boldsymbol{\omega}}_j \qquad (3.46)$$

Taking the derivatives of the new kinetic energy of the system with respect to the generalized coordinate and velocity, θ_i and $\dot{\theta}_i$, respectively, and taking a time derivative of the latter, one can implement these terms into Equation (3.9) and establish the Lagrange's Equations as:

$$\frac{d}{dt}\left(\frac{\partial T_{trans.}}{\partial\dot{\theta}_i}\right)-\frac{\partial T_{trans.}}{\partial\theta_i}+I_{yy_i}\ddot{\theta}_i = Q_i \qquad (i=1,2,\ldots,n+1) \qquad (3.47)$$

Therefore, the Lagrange's Equations can be written as:

$$\begin{aligned}&\sum_{k=i}^{n+1}\left\{\sum_{j=1}^{k}\left[m_k l_i l_j\left(\ddot{\theta}_j\cos(\theta_j-\theta_i)-\dot{\theta}_j{}^2\sin(\theta_j-\theta_i)\right)\right]\right\}+I_{yy_k}\ddot{\theta}_k \\ &\quad=\sum_{j=i}^{n}\left[\left(F_{Z_j}-m_j g\right)l_i\cos\theta_i-F_{X_j}l_i\sin\theta_i\right]+M_{\theta_i} \qquad , \ (i=1,2,\ldots,n)\end{aligned} \qquad (3.48)$$

This configuration implies that the equations can be written in a similar form to Equation (3.31) with slight changes as follows:

$$\mathbf{A}'\ddot{\boldsymbol{\theta}}+\mathbf{B}diag\left\{\dot{\boldsymbol{\theta}}\right\}\dot{\boldsymbol{\theta}}=\mathbf{Q} \qquad (i=1,2,\ldots,n) \qquad (3.49)$$

where

$$\mathbf{A}'=\mathbf{A}+diag\left\{\mathbf{I}\right\} \qquad (3.50)$$

where $\mathbf{I}$ is an *(n+1)×1* vector representing inertias of system segments to be referred to as the segment inertia vector hereafter. This notation ensures that the term $I_{yy_i}\ddot{\theta}_i$ is added to the corresponding diagonal elements of matrix $\mathbf{A}$.

Equation (3.50) shows that addition of the UAS's segment to the system, affects the equations only by one term, which is $I_{yy_i}\ddot{\theta}_i$ being added to the terms on the diagonal of the matrix $\mathbf{A}'$. It should be noted that the generalized forces for the last segment, i.e. the UAS segment will change as the forces on the UAS are calculated with a different coefficient of drag and lift based on the UAS's geometry and material which will be discussed in Section 3.8. For the purpose of this chapter, however, the cable segments are modeled using a lumped mass model allowing all moments of inertias to be zero except that of the last segment. Therefore the rotational component of the kinetic energy of the system can be written as:

$$T_{rot.,U} = \frac{1}{2}\vec{\boldsymbol{\omega}}^{\mathbf{T}}\mathbf{I}'\vec{\boldsymbol{\omega}} \qquad (3.51)$$

Since the angular velocity of the system is also defined about the *y* axes only, with *x* and *z* components equal to zero, Equation (3.51) is expressed as:

$$T_{rot.,U} = \frac{1}{2}I'_{yy}\dot{\theta}^2_{n+1} \qquad (3.52)$$

Derivation of this rotational part of kinetic energy, will yield the term $I_{yy_i}\ddot{\theta}_i$. Therefore, for the lumped mass model, one can write the *i*th element of the segment inertia vector as:

$$[\mathbf{I}']_i = \begin{cases} 0 & \text{i=1},\ldots,n \\ (I'_{yy})_{UAS} & \text{i=}n+1 \end{cases} \qquad (3.53)$$

This causes the inertia term to be added to the equation of motion of the *n+1*th element only. This will conclude the derivation of equations of motions for a cable-body system consisting of a UAS and a cable considering the following assumptions:

1. The motion is planar.
2. The UAS dynamics is simplified and the UAS is considered robust without a flight controller.
3. The cable is considered inextensible.
4. The base station is considered stationary.
5. The cable is modeled as a discrete lumped-mass model.

In order to solve these equations given proper initial conditions, it is important to employ a suitable numerical technique. This will be discussed in the following sections.

3.6. Numerical Method for Solving the System's Equation of Motion

Now that the equations of the system's dynamics are completely derived and defined in Equation set (3.49), one can proceed to solving these equations using the appropriate initial conditions. Since the system's equations are highly coupled and nonlinear as well as large in terms of number of unknowns and equations, it is necessary to employ an appropriate numerical method to solve the system's dynamic equations. As a result, the Runge-Kutta scheme is chosen to solve the system's equations using MATLAB™ software package [39]. As for any numerical analysis method, Runge-Kutta integration method needs to receive the system's equation in a specific arrangement. Section 3.6.1 explains how the equations are manipulated to be solved using the fourth-order Runge-Kutta solver (RK4). Note that the name "fourth-order Runge-Kutta method" does not concern the mathematical order of the equations; but rather the truncation errors in the integration routine, details of which are beyond the scope of this work. More in-depth information on details of this method is vastly available through literature and particularly in [40].

3.6.1. Order Reduction of Differential Equations

The 4th-order Runge-Kutta method can only directly solve first-order ODE's of the form:

$$\frac{dy}{dx} = f(x, y), \qquad y(0) = y_0 \tag{3.54}$$

Therefore only first order ODE's can be solved using this technique. The system at hand consists of *n+1* nonlinear second-order differential equations in which the unknowns are the angles of the segments, i.e. θ as well as their first and second time-derivatives, $\dot{\theta}$ and $\ddot{\theta}$, respectively. In order to avoid the need to use the Taylor Expansions to solve these second-order equations, system's order will be reduced by converting these *n+1* second-order equations to *2(n+1)* first order ODE's using the following procedure.

The general form of the system's equation for each segment is presented below:

$$A(\theta,\dot{\theta})\frac{d^2\theta}{dt^2} + B(\theta,\dot{\theta})\left(\frac{d\theta}{dt}\right)^2 = f(t,\theta,\dot{\theta}) \qquad \theta(0) = \theta_0, \dot{\theta}(0) = \dot{\theta}_0 \tag{3.55}$$

where $\ddot{\theta}$, can be a function of time, θ , and $\dot{\theta}$. By assuming the following vector, one can change the variables in order to achieve a first-order ODE set with the number of equations twice as the original number of equations.

The vector $\vec{\theta}$ is called the angular position vector and is represented by:

$$\vec{\theta} = \begin{pmatrix} \theta_1 \\ \theta_2 \\ \vdots \\ \theta_{n+1} \end{pmatrix}_{(n+1)\times 1} \tag{3.56}$$

The vector $\vec{\omega}$ is called the angular velocity vector and is represented by:

$$\vec{\omega} = \dot{\vec{\theta}} = \begin{pmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ \vdots \\ \dot{\theta}_{n+1} \end{pmatrix}_{(n+1)\times 1} \tag{3.57}$$

The vector $\vec{\alpha}$ is called the angular acceleration vector and is represented by:

$$\vec{\alpha} = \dot{\vec{\omega}} = \begin{pmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \\ \vdots \\ \ddot{\theta}_{n+1} \end{pmatrix}_{(n+1)\times 1} \tag{3.58}$$

Now, define a vector $\vec{\zeta}$ consisting of $\vec{\omega}$ and $\vec{\theta}$, namely the angular state of the system as:

$$\vec{\zeta} = \begin{pmatrix} \vec{\omega} \\ \vec{\theta} \end{pmatrix} = \begin{pmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ \vdots \\ \dot{\theta}_{n+1} \\ \theta_1 \\ \theta_2 \\ \vdots \\ \theta_{n+1} \end{pmatrix}_{2(n+1)\times 1} \tag{3.59}$$

Then, one can write the time-derivative of Equation (3.59) as:

$$\dot{\vec{\zeta}} = \begin{pmatrix} \vec{\alpha} \\ \vec{\omega} \end{pmatrix} = \begin{pmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \\ \vdots \\ \ddot{\theta}_{n+1} \\ \dot{\theta}_1 \\ \dot{\theta}_2 \\ \vdots \\ \dot{\theta}_{n+1} \end{pmatrix}_{2(n+1)\times 1} \tag{3.60}$$

By comparing these equations to the system's equation of motion derived in Equation (3.49), and by multiplying the equation by inverse of matrix $\mathbf{A}'$ from the left, this equation can be expressed as:

$$\begin{aligned} \dot{\vec{\zeta}}_{2(n+1)\times 1} &= \begin{bmatrix} -\mathbf{A}'^{-1}\mathbf{B}diag\{\vec{\omega}\} & [0]_{(n+1)\times(n+1)} \\ I_{(n+1)\times(n+1)} & [0]_{(n+1)\times(n+1)} \end{bmatrix}_{2(n+1)\times 2(n+1)} \times \vec{\zeta}_{2(n+1)\times 1} \\ &+ \begin{bmatrix} \mathbf{A}'^{-1}\vec{Q} \\ [0]_{(n+1)\times 1} \end{bmatrix} \end{aligned} \tag{3.61}$$

where $\vec{Q}$ is the vector of generalized forces shown in Equation (3.27).

Equation (3.61) is a set of *2n+2* first-order ODE's which could be solved using the Runge-Kutta numerical integration routine. The codes written to implement these equation into a MATLAB™ program and to solve the problem can be found in Appendix A.

These equations are then solved numerically using appropriate initial conditions to simulate different scenarios. Using the accelerations obtained from the numerical solutions, the forces on each cable segment are calculated in order to estimate the cable tension at each node. This model is tested for correctness in Section 3.7.

3.7. Model Verification

In order to confidently use the simulations and rely on the results, it is necessary to verify the model against experiment or available similar results. As mentioned before, given the correct parameters, and initial conditions this cable model is very versatile and can be applied to a wide range of applications, from subsea to aerospace and even in space applications as long as the cable is not experiencing very low tensions and the cable's stiffness is high enough for the cable extension to be negligible. To verify the correctness of the derivations and validity of the model implemented in MATLAB™, several tests have been run the result of which will follow.

3.7.1. Simple Pendulum Test

The model has been tailored to simulate a simple pendulum and the oscillation frequency of the pendulum is verified against theoretical calculations. In theory the period of an ideal simple pendulum with a massless rod of length *L* oscillating without any wind friction is calculated by:

$$T = 2\pi\sqrt{\frac{L}{g}} \qquad (3.62)$$

A series of tests were run with the same conditions to replicate various pendulum lengths and the simulated results are compared to the calculated theoretical values. Results are compared in Figure 3.9.

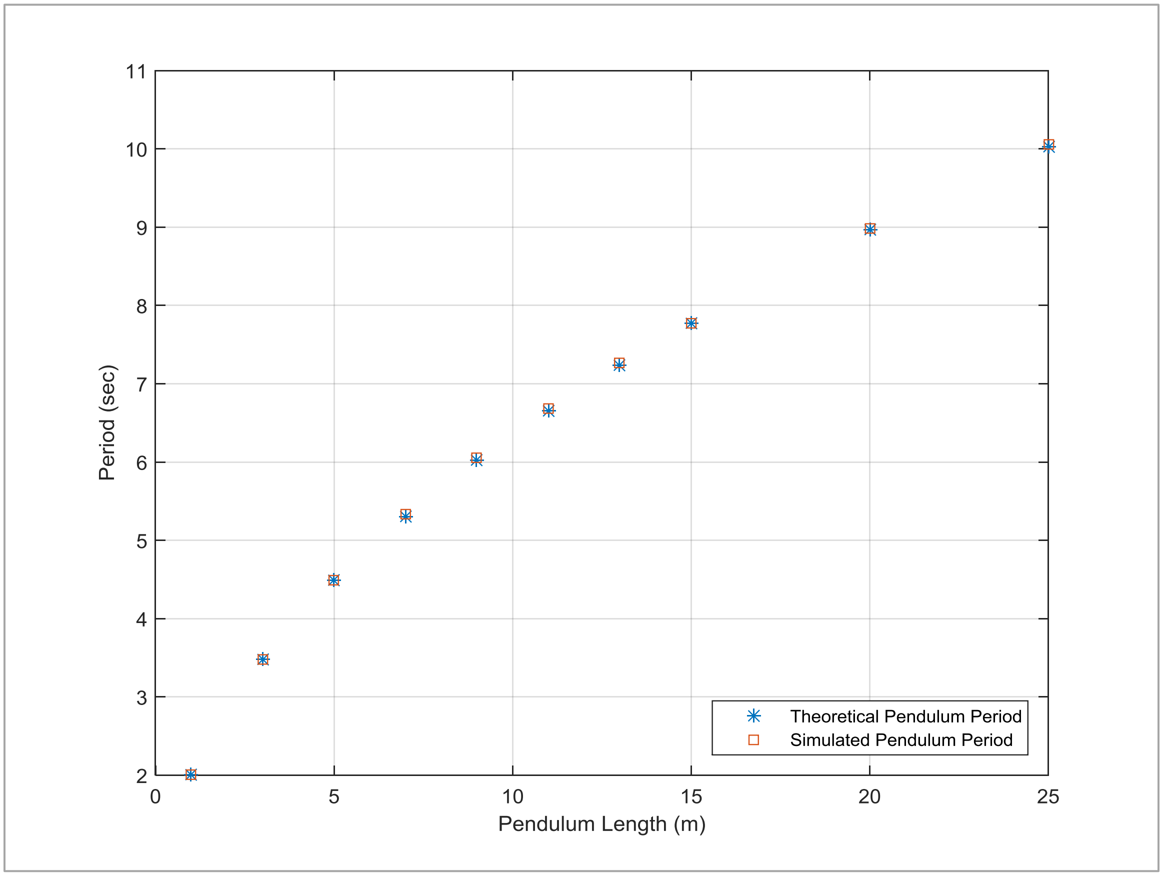

Figure 3.9. Simple Pendulum Periods, Comparison of Simulated and Theoretical Results.

The plotted results show that the model is agreement with theory in predicting the oscillation period for a simple pendulum. This approves that the equations of motion for the system were derived correctly. However more tests are needed to prove the validity of this model in terms of deriving the equations of motion for the general *n*-segment model.

3.7.2. Generalization of the Model to a Finite-Segment Cable

In previous section, it was confirmed that a simple pendulum using the developed model can replicate the oscillation of an ideal pendulum. In this set of tests, the model is taken further and the cable is modeled using *n* rigid segments. Different sets of UAS thrust and pitch are applied at the top end of the model and the cable shapes as well as UAS positions are compared to similar simulated results using the static model developed in Chapter 2. Figure 3.10 shows four different force sets applied at the top end of the cable, both simulated using the catenary model used in Chapter 2 and the discrete model developed in this chapter. Note that the simulated models in this case do not consider the weight and inertia of the UAS since the purpose of these tests are merely validation of the cable shape regardless of the UAS weight.

One can observe that the discrete model is able to replicate the shape function using 31 segments. Even with this number of segments, there is an error of up to 8% which is acceptable in most applications. In addition, the number of segments can affect the convergence and accuracy of the model, hence there is an error present due to the model approximation.

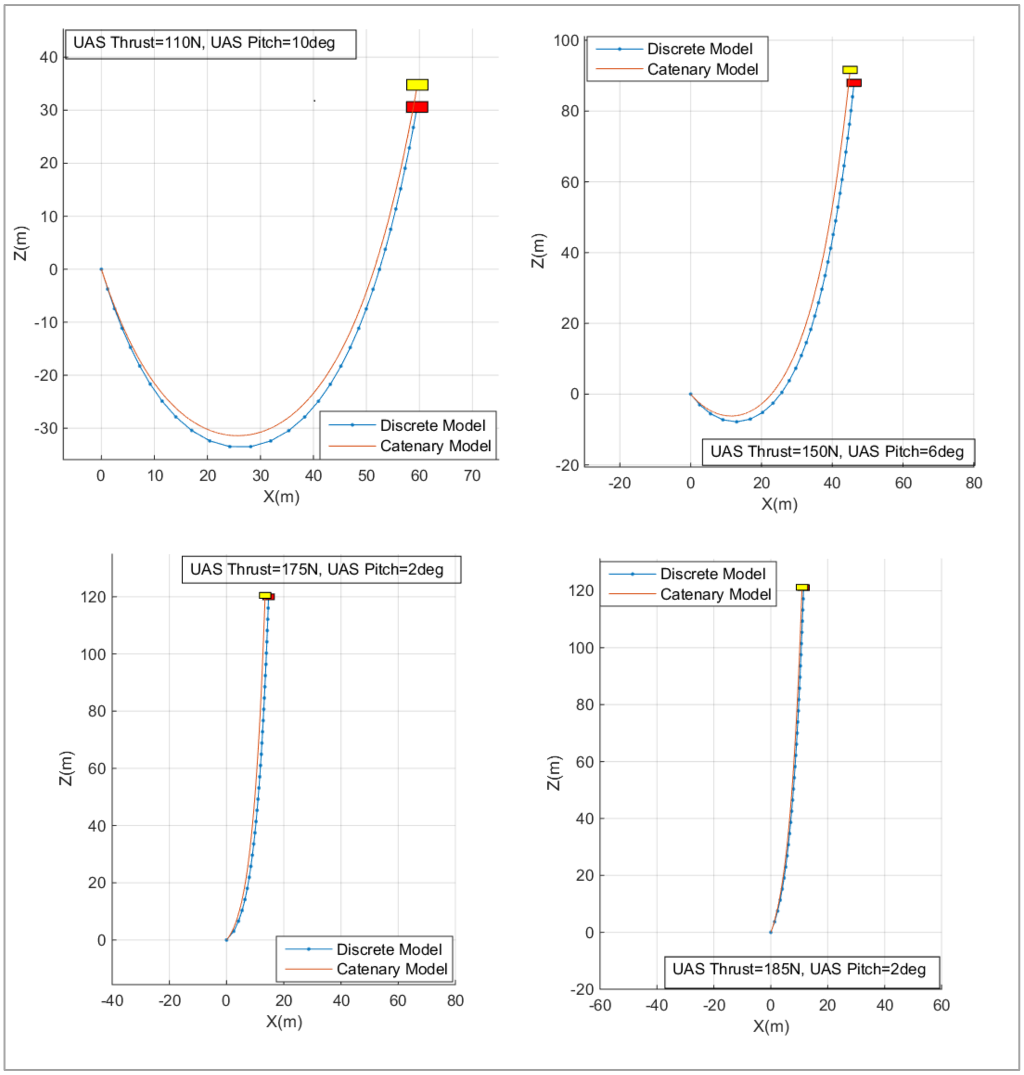

Figure 3.10. Comparison of static cable shapes; catenary model vs. discrete model

A verification of the forces in the tension concludes the validity of the model. The simulation is run with the parameters defined in Table 3.1 and are compared to the results of the same conditions using the model used in Chapter 2.

Figure 3.11 shows different stages of the cable when the UAS is taking off under the effect of UAS thrust and pitch shown in Table 3.1. The cable is initially lying on a horizontal line when at time 0, the forces are applied at its top end moving the UAS upwards until the cable reaches a steady state. The cable tension calculated using both models are then compared in Figure 3.12.

Table 3.1. Forces applied at UAS for Validation

Simulation Parameter	Value	Units
UAS Thrust	180	[N]
UAS Pitch	4	[deg]
UAS Weight	0	[-]

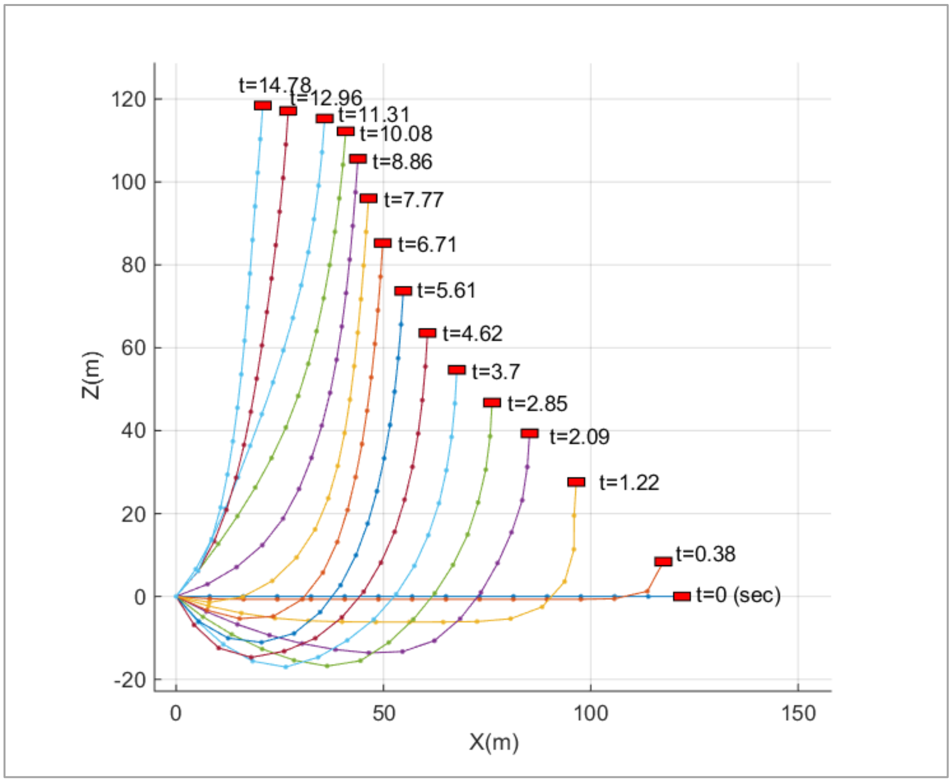

Figure 3.11. Cable shapes in the Take-Off process

The results show that the cable model can predict the tension in the cable to a close approximation. However further from the base station, the error increases. The reason behind this is mainly the error included in the end point position of the UAS. As you can see the final point on Figure 3.12 represents the UAS endpoint and while the tensions are equal, the horizontal position of nodes are not the same. As a result, these difference in the values occur. However in the same figure, it can be seen that a model with 40 segments behaves very closely to the catenary model. This concludes the equations have been derived correctly.

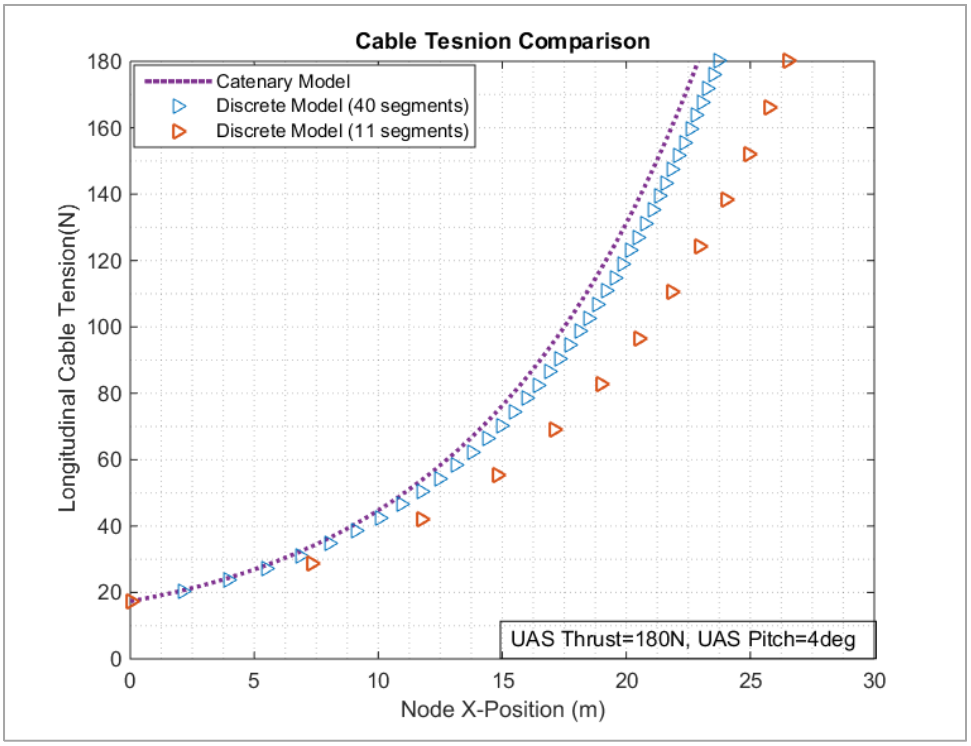

Figure 3.12. Comparison of longitudinal tension in cable

3.7.3. Model's Convergence

Since the developed model is a finite-segment model, its accuracy can be dependent on the number of segments used in the simulation as it was observed in Figure 3.12. It is clear that modeling the cable with 11 segments yield less accurate results than the same model with 40 segments. In order to achieve a better approximation of the system's behavior, it is necessary to understand the model's convergence. As a result, the simulation is run under same conditions of Table 3.1 for various cable segment numbers and the results are presented in Figure 3.13.

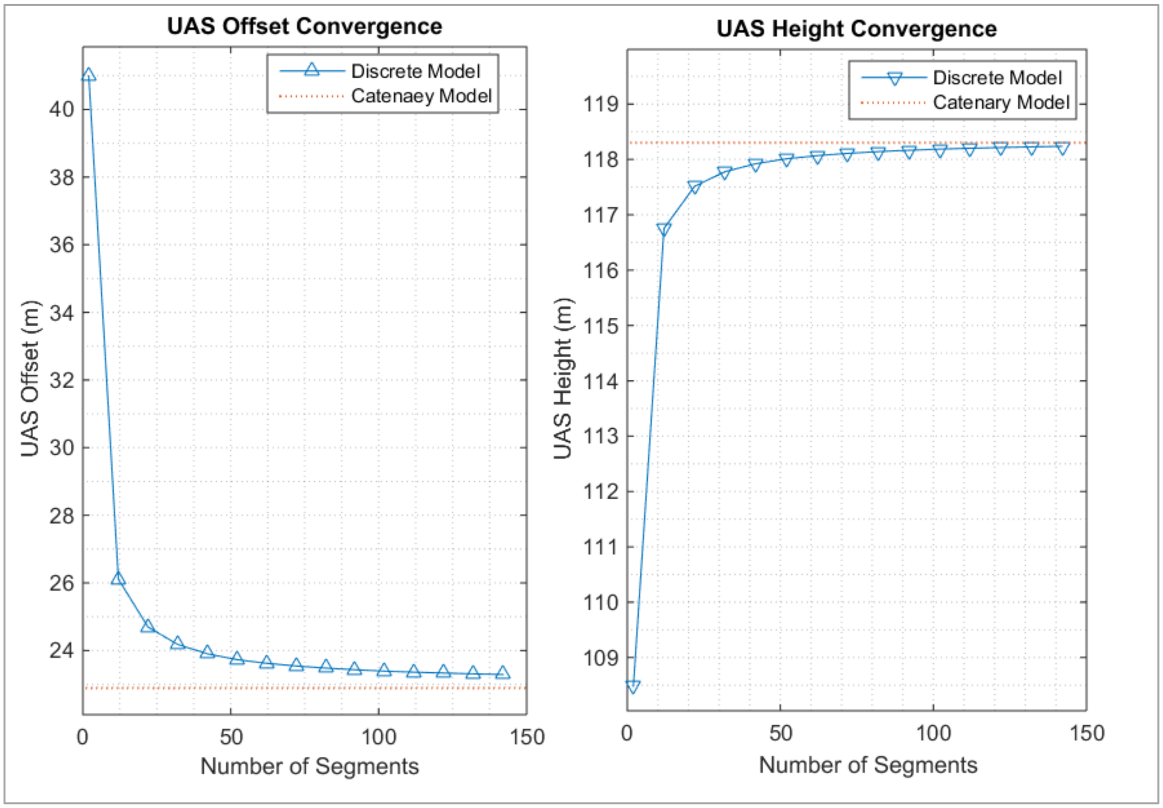

Figure 3.13. Convergence of UAS position for various number of cable segments

According to Figure 3.13, one can see that the model convergence with an error of less than 2% with 80 cable segments. Obviously adding more cable segments will increase the computational load for the solving process. Figure 3.14 shows how the duration for solving the system's dynamics changes with the number of segments for simulating a 120 second flight. One can see that the time needed for the algorithm to solve for the accelerations as well as calculating the positions along with all other parameters changes significantly with the number of segments; however, discussing the computational complexity of the algorithm is beyond the scope of this work.

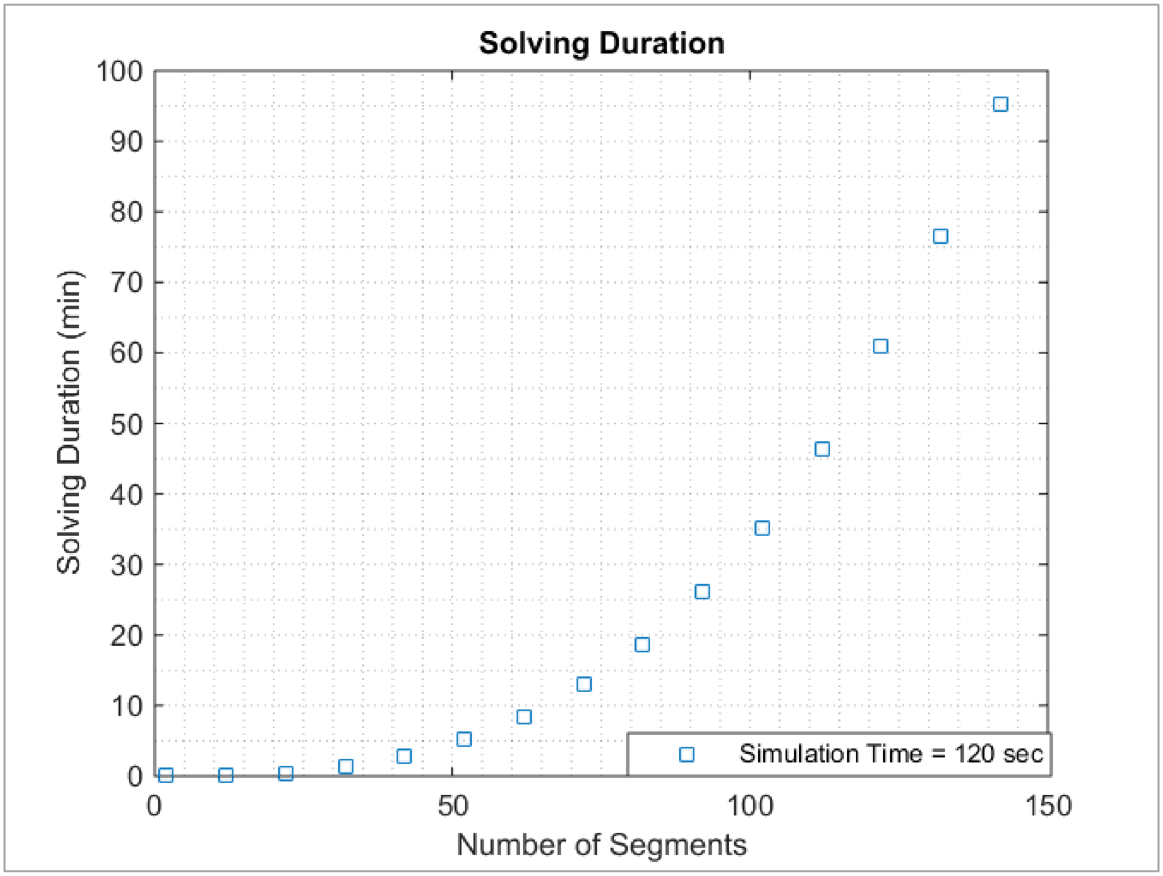

Figure 3.14. Solving duration for different number of cable segments

Overall, based on the observations discussed above, one can conclude that selecting 80 segments for a cable of length 122 m could be an appropriate option in terms of accuracy as well as computational load. Note that as the cable length changes, the required number of nodes change directly as well; i.e. for a longer cable, larger number of cable elements will be needed for the same accuracy and vice versa.

3.8. Results and Discussions

The model developed here can be used to approximate the cable shape and tensions in different flight scenarios. Although, it was shown before that 80 segments will provide an optimum solution in terms of solving time and accuracy, in these case studies, lower number of segments are used in order to reduce the processing time since the purpose of these results are illustration of the behavior of the system rather than exact calculation of variable values.

Few case studies are investigated considering the UAS's weight and moment of inertia as shown in Table 3.2 and the results are discussed in the following sections. The UAS thrust is modeled as a vectored thrust always perpendicular to the UAS horizontal axis. The thrust then is a vector consisting of a horizontal and vertical component in the inertial coordinate system which both depend on the UAS pitch; i.e. the angle with respect to the horizontal axis *X*. It is assumed the UAS thrust is held exactly constant in magnitude unless stated otherwise. In addition the normal and frictional (tangential) drag coefficients used for the cable are 1.5 and 0.02 respectively as suggested in Hoerner's book [32].

Table 3.2. UAS Parameters

Simulation Parameter	Value	Units
UAS Weight	10.56	[kg]
UAS-Cable Connection Link Length	0.1	[m]
UAS Moment of Inertia (I_{xx})$_{CG}$	1.22	[kg.m^2]
UAS Moment of Inertia (I_{yy})$_{CG}$	1.22	[kg.m^2]
UAS Moment of Inertia (at connection point) (I'_{yy})	1.3256	[kg.m^2]
UAS Frame Rod Length	1.012	[m]
UAS Frame Rod Diameter	0.025	[m]
UAS Vertical Thrust	283.59	[N]

3.8.1. Effect of Wind Speed

This section is dedicated to illustrating effect of horizontal wind speed on the cable shape and UAS location. The cable is positioned in a vertical shape in a way that the UAS is hovering exactly above the base station. For each test, a certain wind speed is applied to the system at *t=1 sec* and the final states of the system is illustrated after 150 seconds. The steady state cable shapes for different wind speeds ranging from 0 to 27 m/s are illustrated in Figure 3.15.

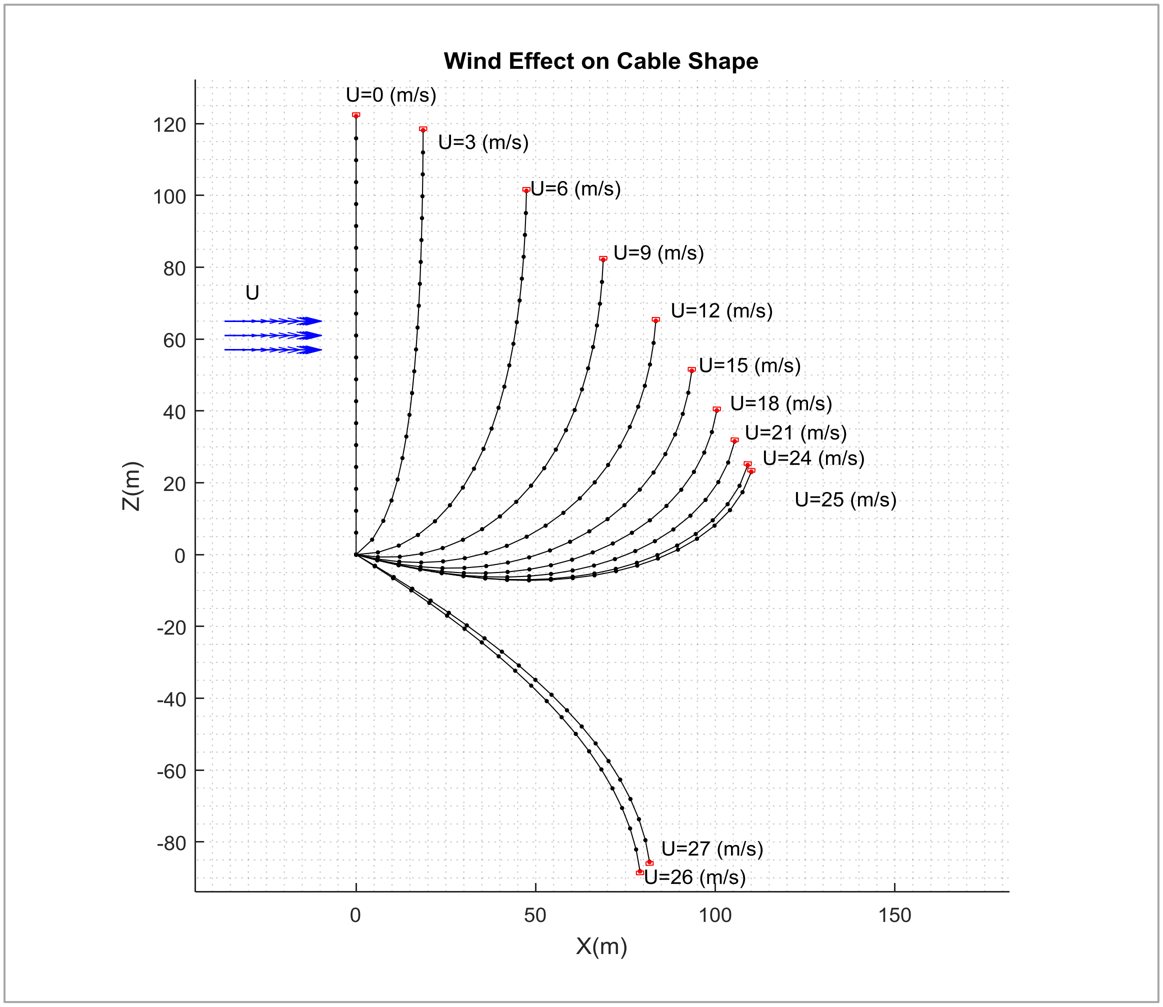

Figure 3.15. Final Cable Shapes at Different Horizontal Wind Speeds

It could be observed from this figure that the UAS becomes unstable for wind velocities larger than 25 m/s. In order to keep the UAS stable, the UAS autopilot should be able to compensate the wind force by aligning the thrust vector so that the horizontal component of the thrust can overcome the wind force while the vertical thrust can maintain altitude of the UAS.

Figure 3.16 shows the UAS horizontal offset and height as well as its pitch versus time for the same set of tests and wind conditions. From these plots, one can conclude that given this amount of thrust, the UAS is not able to stabilize itself as soon as the wind speed increases to 26 m/s.

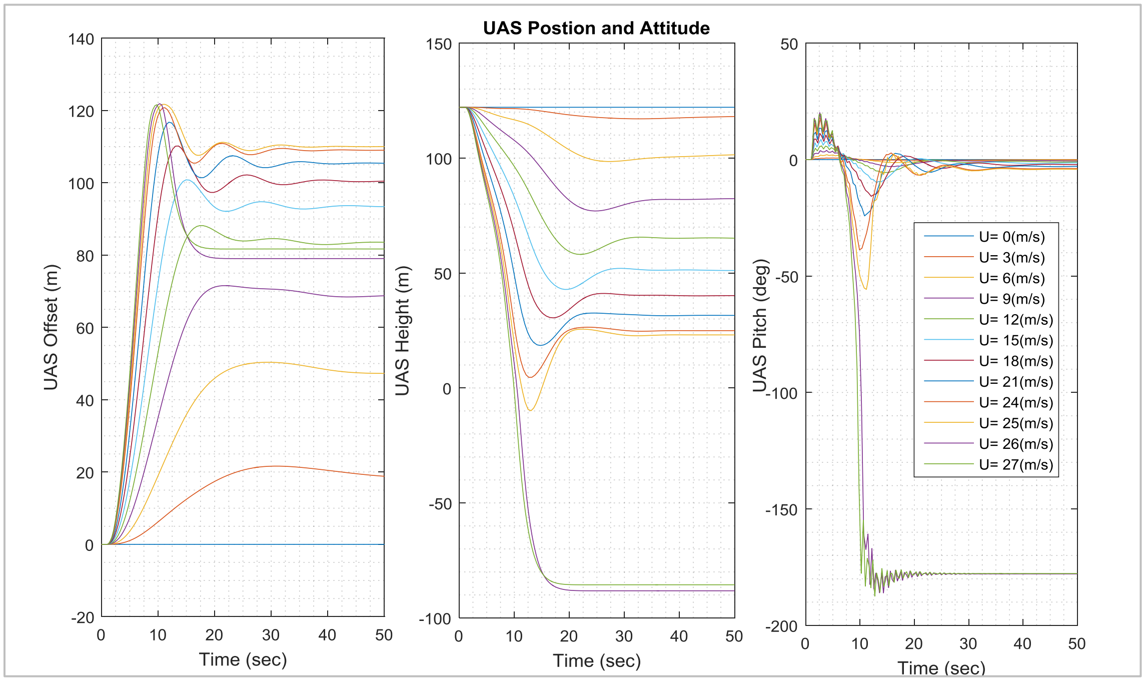

Figure 3.16. UAS position and altitude time history under different wind speeds

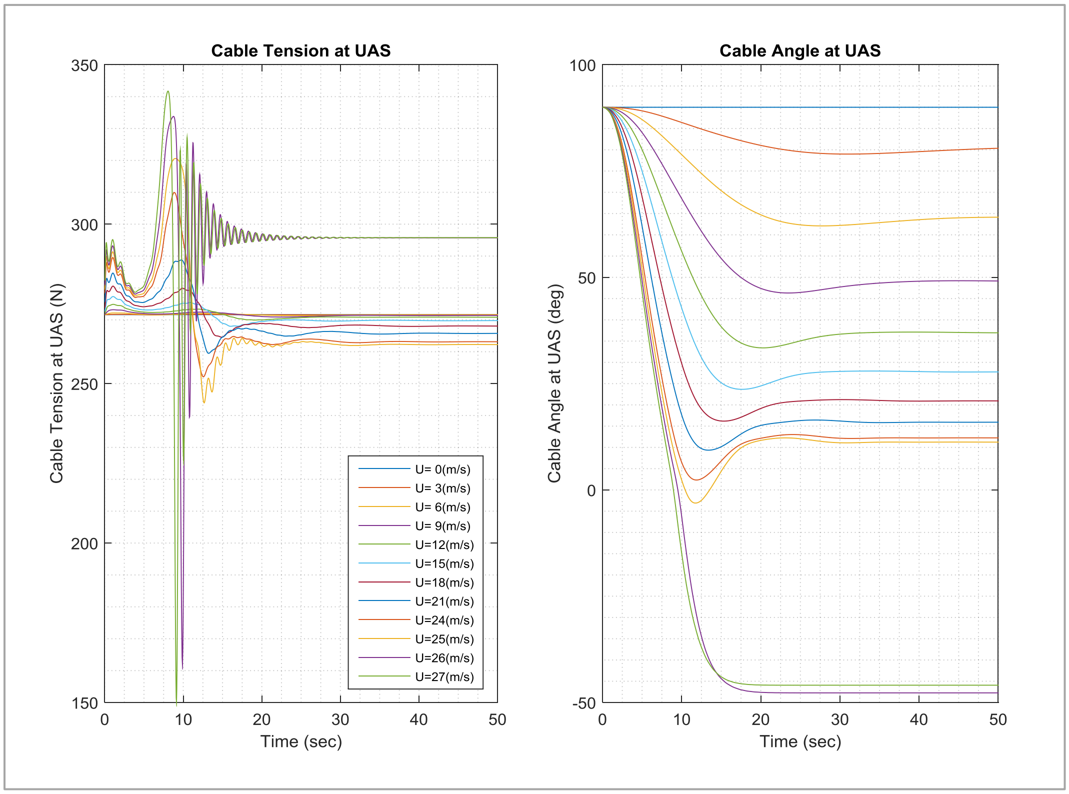

Figure 3.17. Cable tension and angle at UAS connection point

It could be also observed that the cable tensions at the UAS connection point along with the cable angle in Figure 3.17. Also, for all the cases where the UAS is stable, that is wind speeds lower than 25 m/s, the cable tension reaches a constant value after stabilization. It is observable that for higher wind speeds, the tension at the UAS decreases significantly at 11 seconds. This means that the cable becomes a little slack. This causes a sudden change in the tension which can cause a phenomenon called snap loading. This essentially means that can pull the UAS down suddenly which is a hazard to the system.

Another important parameter that should be considered, is the tension at the base station. This helps better understand the forces and therefore torques acting on the winch motor and the requirements for design. Figure 3.18 shows how the cable tension and the cable angle changes at the base station when the UAS is flying in a windy condition. One can see that almost for all wind cases, when the wind starts being applied instantly, the cable tension at base increases at once significantly almost by 600%.

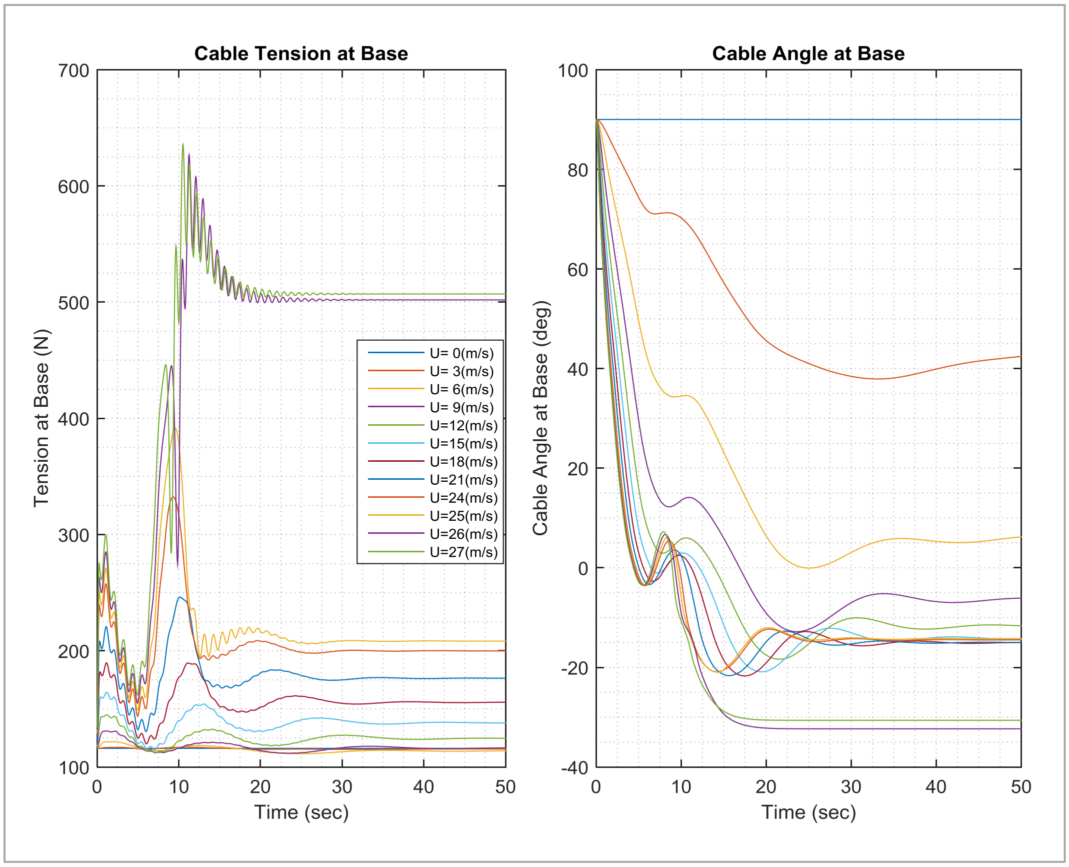

Figure 3.18. Cable tension and angle at the winch

3.8.2. Take-Off in High-Velocity Wind

In this case study, the UAS's behavior during take-off is investigated when the UAS rises from a height of zero in a high-velocity windy condition. The effect of thrust on the maximum achieved flying altitude is investigated as a guide towards the UAS design. Not considering the reeling process, and with focus on thrust effects, the UAS is simulated to takes off from the ground in a wind velocity of 28 m/s with a thrust of 283.59 mentioned in Table 3.2. The thrust then starts to rise at time t=20 sec, linearly with time and the UAS height is observed. Figure 3.19 shows the cable shapes during the increase in the UAS thrust for different wind speeds.

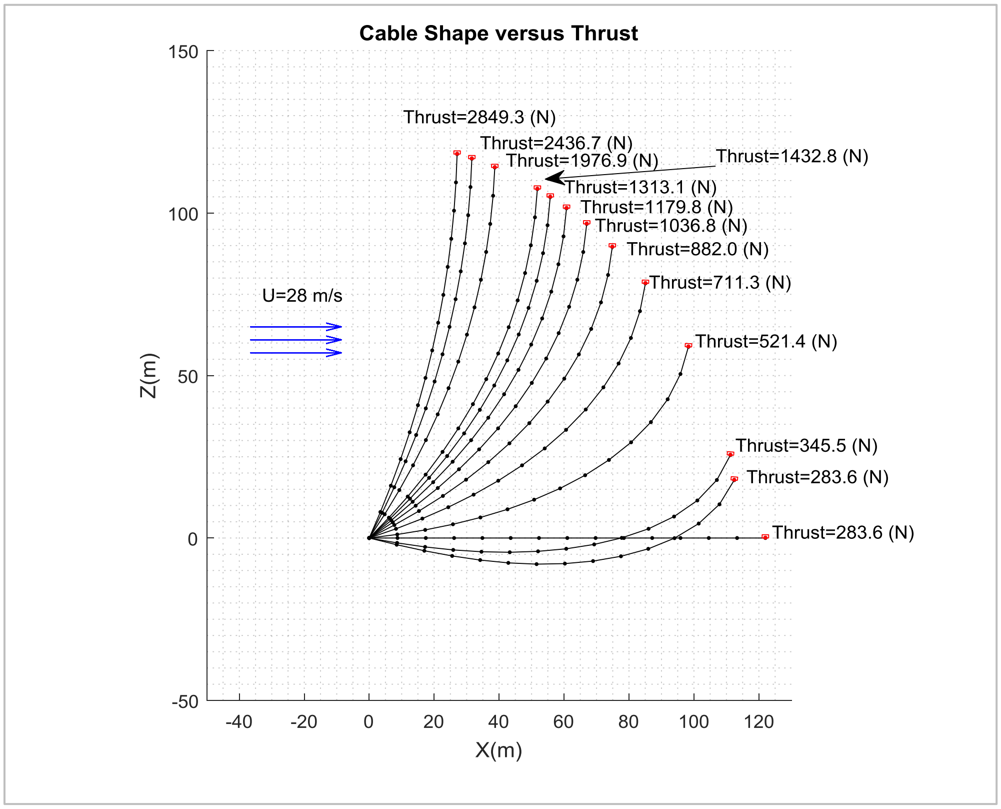

Figure 3.19. Cable shape during take-off under the effect of increasing thrust

The change in UAS position and attitude at various wind speeds under the effect of increasing thrust are also illustrated in Figure 3.20. One can see that in all cases the UAS can achieve the height of almost 122 meters as long as a large thrust is provided. Since this is not the case in reality, and since generating a larger thrust means a larger and heavier UAS, one can conclude the limitations of the system and wind operation condition from this analysis.

Figure 3.21 presents the UAS thrust versus time along with UAS altitude due to this thrust. Based on this graph, one can conclude that with the assumption the UAS can provide up to 400 N of thrust, at no wind speed larger than 4 m/s can the UAS achieve its target altitude of 122 meters.

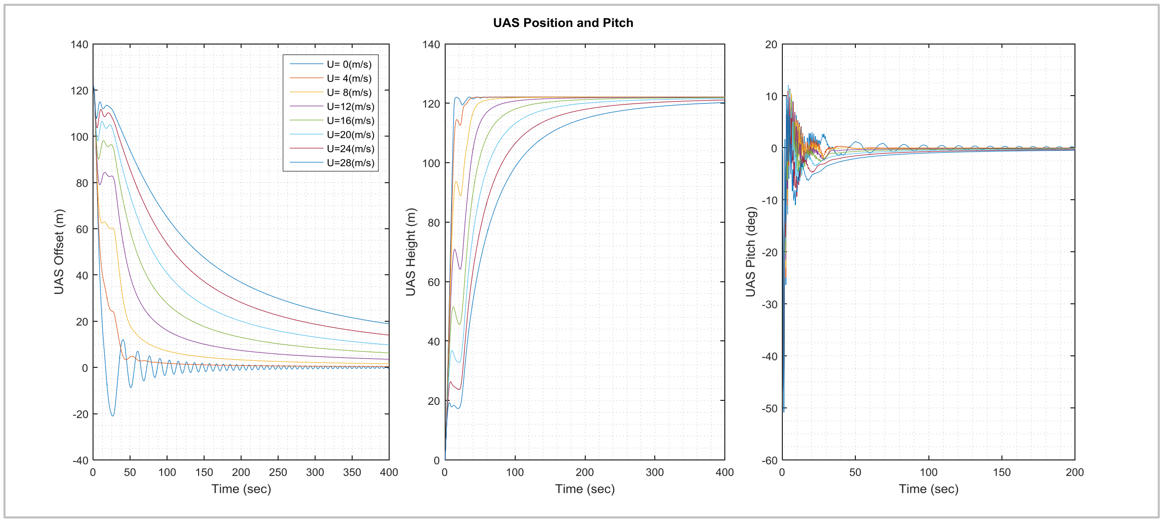

Figure 3.20. UAS position and pitch under the effect of increasing thrust

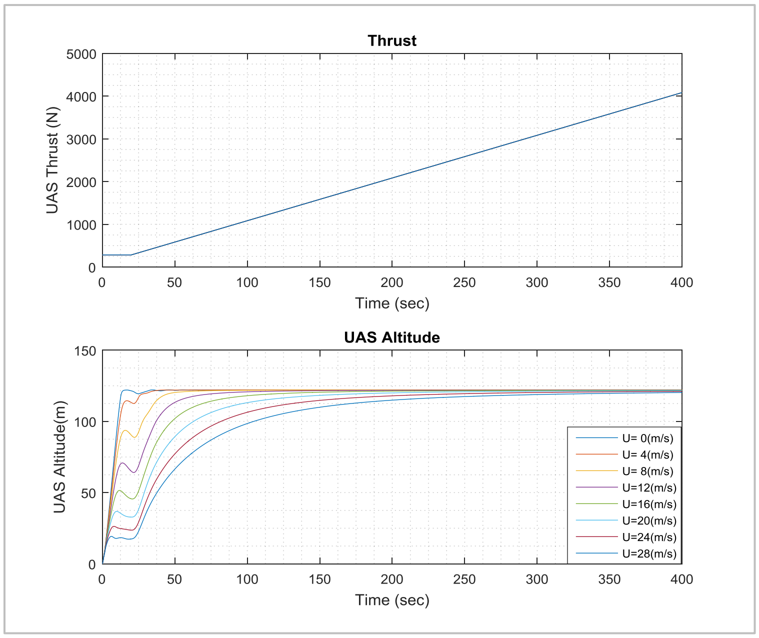

Figure 3.21. UAS altitude and the thrust change

3.8.3. Effect of UAS Thrust Fluctuations

Another usage of this simulator is to learn more about the effects of thrust fluctuations on the cable and UAS dynamics. In this example, the effect of thrust changes on the UAS's position as well as cable tension at both the UAS and the base station is investigated. The thrust is modeled as a simple sine wave of the form:

$$Thrust = T_m + T_f \left(\sin \frac{2\pi}{f} t + \phi \right) \tag{3.63}$$

where the constant ϕ does not affect the results. The variable thrust parameters are defined in Table 3.3.

Table 3.3. UAS Thrust with fluctuations

Parameter	Label	Value	Units
UAS Mean Thrust	T_m	283.59	[N]
Fluctuation Amplitude	T_f	10	[N]
Fluctuation Frequency	f	1	[Hz]

Figure 3.22 - Figure 3.24 show the UAS position and attitude as well as cable tensions at the base and at the UAS due to this UAS thrust fluctuations. It could be observed that the cable tensions vary with the same frequency of 1 Hz at both the UAS and the winch; however the thrust change does not propagate to the base with the same value, but rather with an amplitude of 4 N in comparison to 10 N.

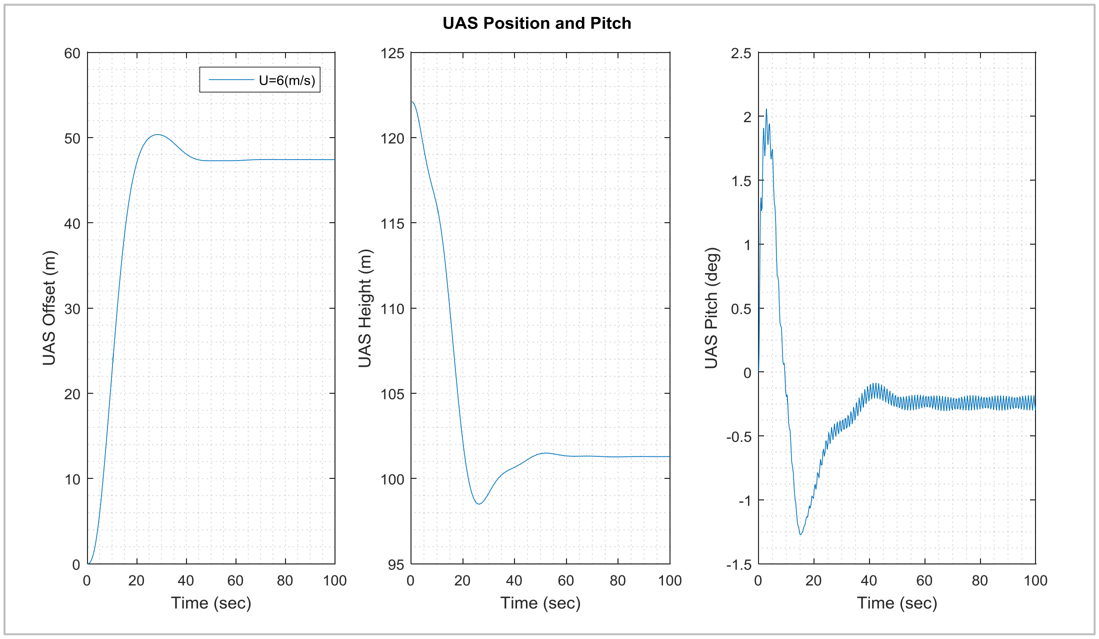

Figure 3.22. UAS position and attitude under thrust fluctuations

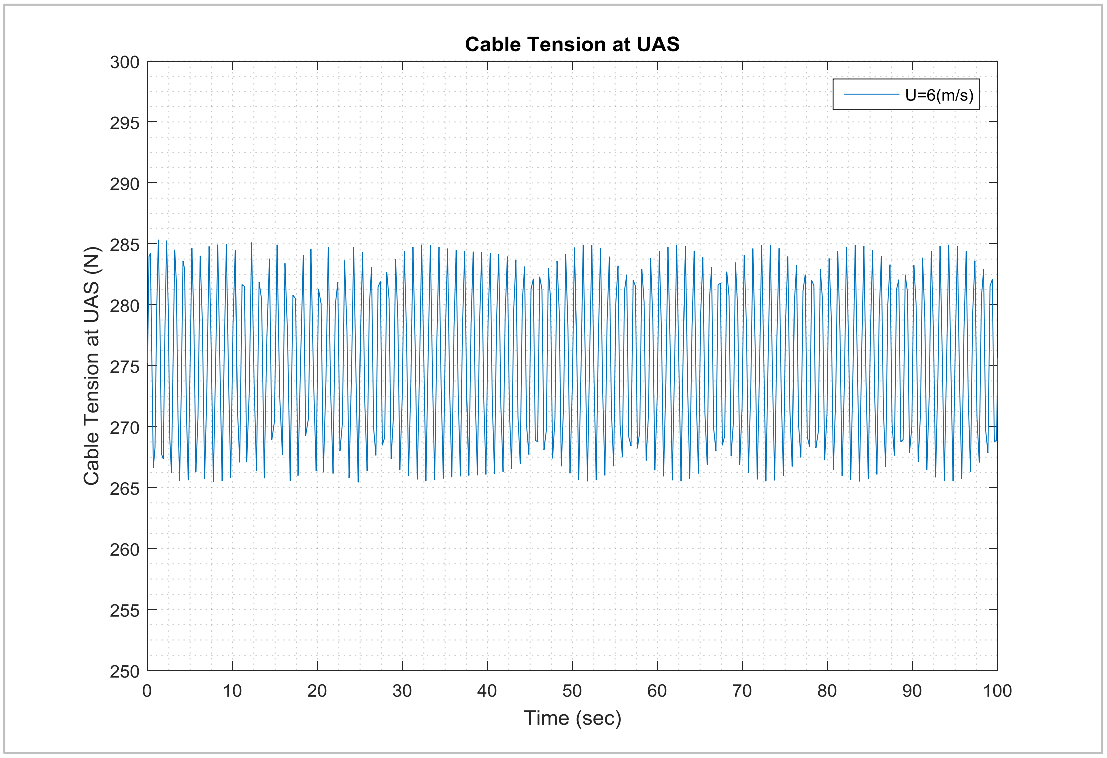

Figure 3.23. Cable tension at UAS under thrust fluctuations

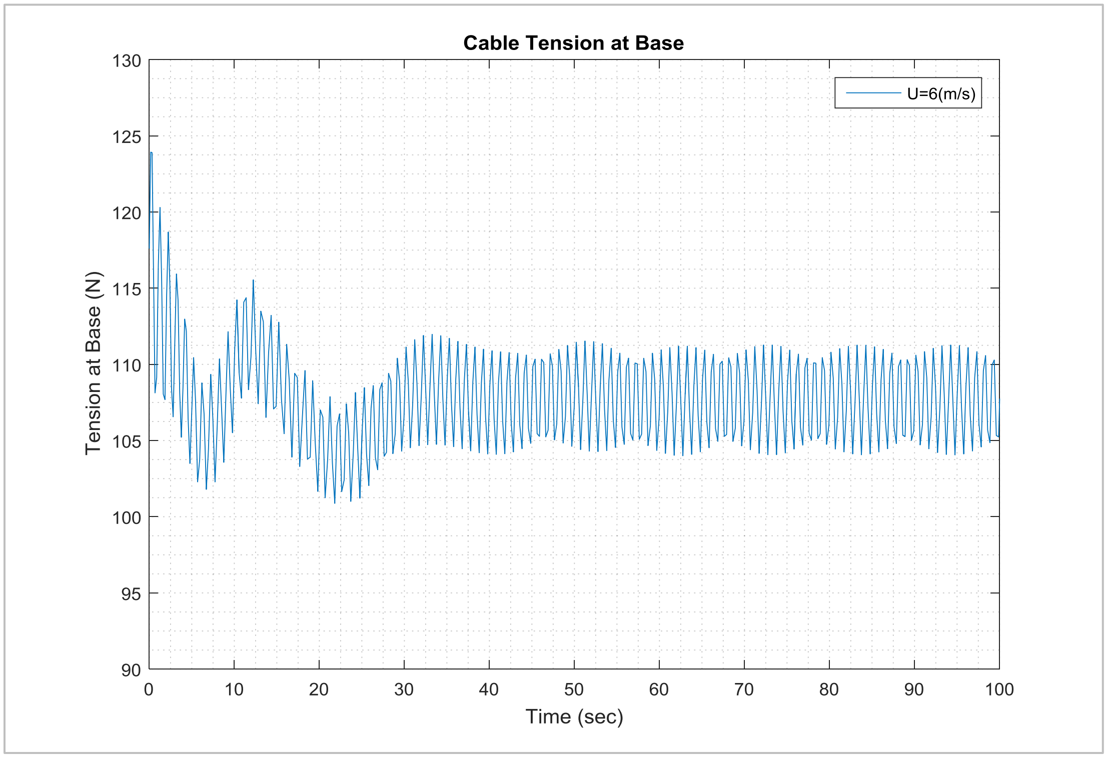

Figure 3.24. Cable tension at base under thrust fluctuations

3.9. Conclusion

In this chapter, a 2D model is developed in order to predict the planar motion of a flexible cable. The cable is considered as a discrete system of finite rigid bodies connected with frictionless joints. The equations of motions are derived using Lagrange's Method for multi body systems. The equations are then solved numerically given proper initial conditions. Wind forces are also modeled using a cable drag model adapted from literature.

The model is validated against theory in terms of oscillation periods as well as against results of Chapter 2 in terms of cable shape function, longitudinal tension, and model convergence due to number of elements. The results show close approximation and reliability of the discrete cable model. The results show that the model is capable of simulating the cable's shape function with a close approximation. This cable model is very versatile and can be applied to a wide range of applications, using proper initial conditions and system configurations.

A few case studies are investigated and results are generated using MATLAB™ software. The effect of UAS thrust on the maximum achieved height is studied. Moreover, effect of wind speed on the cable shape, tension and UAS position and attitude are investigated and presented. In addition, effects of UAS thrust fluctuations on these parameters are illustrated in another section.

In summary, the cable dynamics can be simulated for a planar motion using the 2D model developed and implemented in this chapter. This algorithm can be used to analyze a wide range of properties of the system including but not limited to: effect of cable diameter, cable mass per length, wind conditions, UAS size and weight, etc. on the dynamic and steady state behaviour of the system.

It should be noted that this model does not consider the effect of cable strain as well as bending in the cable. A more thorough modeling could be carried out to implement and investigate these parameters. In addition, a more in-depth modelling of the UAS could yield in more accurate and reliable results.

Chapter 4.

Three- Dimensional Multi-Body Dynamics Modeling and Analysis of the Tether

The results of Chapter 3, although valuable and important to consider in the preliminary stages of an analysis, cannot be fully extended to every system to reflect a realistic representation of the system's dynamic behavior. This is due to the fact that most real-world systems will mostly experience a motion in all 3 directions as opposed to the 2D planar assumption of the previous chapter. Therefore, those effects are investigated by modelling the whole system in a 3D space. Presented in following sections are the derivations and application of the system's equation as well as the simulation results for the cable system moving in the 3D space.

4.1. Methodology

The concept behind this chapter is similar to the one used in Chapter 3. The cable is discretized to a set of small rigid segments instead of a continuous non-rigid member. Each segment will then be assigned an appropriate coordinate system. Using these individual local coordinate systems, Lagrange's Equations are employed for the whole system which then concludes the equations of motion for each segment. The system's equations are then solved accordingly to capture the behavior of the cable under different flight scenarios. Similar to the previous chapter, the effect of variable length dynamics as well as cable extension are not considered.

4.2. Cable Model

4.2.1. Cable Segment Model

For simplicity and to avoid complexity of the system of equations, the segment model used in this chapter is a lumped-mass model as it was used previously in Chapter 3. However the factor complicating equations of motion for a 3D system is the fact that any rigid body such as one cable segment in a 3D space must be defined by six independent coordinates, meaning that each cable segment has 6 degrees of freedom as opposed to the planar case where each segment could be defined by two coordinates. This is due to the fact that, in order to define the position and orientation of each segment, one should specify the position of the mass node with three coordinates in addition to its orientation with three angular coordinates. However, since there is a length constraint on each node, only the orientation of the segment would define the position of the mass node eliminating the need for 6 degrees of freedom. Thus, the size of the system's equations will be three times larger. This adds to the complexity of analytical derivation of the system's equations as well as computational time during solving the equations of motion.

In this chapter, the effect of longitudinal strain is neglected since the steel cable used in this application has relatively high young's modulus. Should the cable in use have low stiffness, such as a very elastic material, it is definitely recommended to use the spring mass model to model the strain effect. This will complicate derivation of equations as well as computational load of the simulations. Due to the scope of work, this study is left for future work.

4.2.2. Coordinate System and Model Set-up

An important key to the modeling of a rigid body in 3D space is the concept of Euler Angles. This is extensively discussed in literature and text books, however the main reference for this concept for the purpose of this study is derived from the valuable Advanced Dynamic book by Greenwood [33].

Consider a rigid body in the 3D space, the center of mass of which in the inertial reference frame XYZ is defined by (x_c,y_c,z_c); To fully define the location and orientation of this object 6 coordinates are needed, 3 of which are the location of the center of mass.

Now consider a local coordinate system xyz attached to the rigid body with the origin located at an arbitrary point. The relationship between the orientation of the local coordinate system xyz and the inertial coordinate XYZ is essentially what defines the orientation of the rigid body. This relationship, technically a rotation matrix, is what determines the Euler Angles and vice versa. Therefore, any vector in the xyz local coordinate system can be represented in the inertial coordinate system, XYZ by utilizing this transformation matrix. In essence, each transformation consists of three successive rotations where each rotation is done about the latest body axis. The type of Euler Angles used for this chapter is called type I or Aircraft Euler Angles.

The first rotation is about the Z axis an angle ψ to achieve the new rotated coordinate system $x'y'z'$. Then this coordinate system is rotated about y' axes by an angle of θ and coordinate system $x''y''z''$ will result. At the end this coordinate system is rotated about the x'' axis through an angle of ϕ and finally the required local coordinate system xyz is located.

The rotation matrices for each of the rotations can be presented as [33]:

$$\mathbf{R}_{\psi} = \begin{bmatrix} \cos\psi & \sin\psi & 0 \\ -\sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{4.1}$$

$$\mathbf{R}_{\theta} = \begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{bmatrix} \tag{4.2}$$

$$\mathbf{R}_{\phi} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & \sin\phi \\ 0 & -\sin\phi & \cos\phi \end{bmatrix} \tag{4.3}$$

Note that the whole rotation from the inertial frame to the local body frame is actually a combination of these 3 successive rotations and can be written as:

$$\mathbf{R}_{\phi\theta\psi} = \mathbf{R}_{\phi}\mathbf{R}_{\theta}\mathbf{R}_{\psi} \tag{4.4}$$

In this work, the local frame should be rotated to the inertial coordinate system. This can be completed by using the inverse of matrix $\mathbf{R}_{\phi\theta\psi}$ since each of the three rotations is independent of the other. Therefore, one can obtain any vector in the body frame by:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \mathbf{R}_{\phi\theta\psi}{}^{-1} \begin{bmatrix} x \\ y \\ z \end{bmatrix} \tag{4.5}$$

Matrix $\mathbf{R}_{\phi\theta\psi}$ is an orthogonal matrix, therefore its inverse can be obtained by transposing the matrix; and then it could be expressed as:

$$\mathbf{R}_{\phi\theta\psi}{}^{-1} = \begin{bmatrix} \cos\psi\cos\theta & (-\sin\psi\cos\phi + \cos\psi\sin\theta\sin\phi) & (\sin\psi\sin\phi + \cos\psi\sin\theta\cos\phi) \\ \sin\psi\cos\theta & (\cos\psi\cos\phi + \sin\psi\sin\theta\sin\phi) & (-\cos\psi\sin\phi + \sin\psi\sin\theta\cos\phi) \\ -\sin\theta & \cos\theta\sin\phi & \cos\theta\cos\phi \end{bmatrix} \tag{4.6}$$

The reader is referred to [33] for more explanation regarding Euler Angles.

Error! Reference source not found. shows the inertial and local coordinate systems for the cable model in 3D space. Length of the i^{th} segment is labeled by l_i as before. For each segment, the angles ϕ, ψ, and θ, namely roll (bank), heading, and pitch angles, are what defines the local coordinate system with respect to the inertial frame XYZ. One can now simply express each node's location in terms of the Euler Angles as:

$$\begin{bmatrix} X_i \\ Y_i \\ Z_i \end{bmatrix} = \sum_{j=1}^{i} \mathbf{R}^{-1}_{\phi_j\theta_j\psi_j} \begin{bmatrix} x_j \\ y_j \\ z_j \end{bmatrix} \qquad (i = 1, 2, 3, \ldots, n) \tag{4.7}$$

where X_i, Y_i and Z_i are the coordinates of the i^{th} mass node.

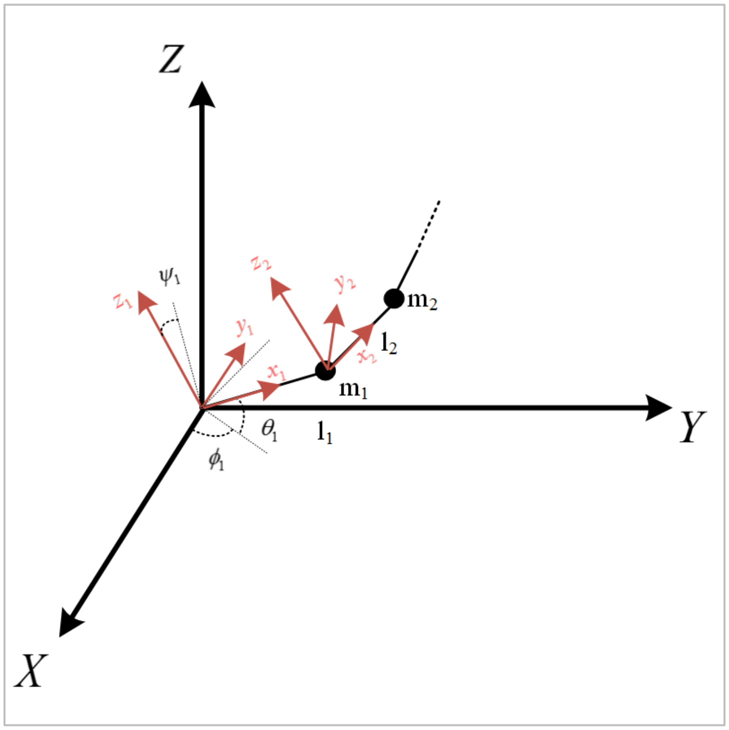

Figure 4.1**. Global and local coordinate systems of the cable**

Since each node can be located at in its local coordinate by the vector *(l$_i$,0,0),one* can define the position of each node as:

$$\begin{bmatrix} X_i \\ Y_i \\ Z_i \end{bmatrix} = \sum_{j=1}^{i} R^{-1}{}_{\phi_j\theta_j\psi_j} \begin{bmatrix} l_j \\ 0 \\ 0 \end{bmatrix} \qquad (i = 1, 2, 3, \ldots, n) \qquad (4.8)$$

Therefore, the coordinates will be defined as:

$$\begin{cases} X_i = \sum_{j=1}^{i} l_j \cos\psi_j \cos\theta_j \\ Y_i = \sum_{j=1}^{i} l_j \sin\psi_j \cos\theta_j \\ Z_i = \sum_{j=1}^{i} l_j \left(-\sin\theta_j\right) \end{cases} \qquad (i = 1, 2, 3, \ldots, n) \qquad (4.9)$$

It can be observed that in this coordinate configuration, each node location will be independent of the bank angle ϕ since each node location is not dependant on torsion of the segment due to the fact that a simple lumped-mass model is used and each node is considered a particle of mass *m*. Each node will be fully defined by 4 coordinates *X, Y, Z* as well as the segment's bank angle, in general; however the system is constrained by the cable segment length. That is:

$$l_i^2 = \left(X_i - X_{i-1}\right)^2 + \left(Y_i - Y_{i-1}\right)^2 + \left(Z_i - Z_{i-1}\right)^2 \qquad (i = 1, 2, \ldots, n) \qquad (\ 4.10\ )$$

Presence of this constraint reduces the number of system DOF by one for each segment. Therefore, in fact the system has a DOF of *3n*.

Having set up the coordinate systems properly, equations of motion for the system can now be developed using Lagrange's Equations.

4.3. Derivation of Equations of Motion Using Lagrange's Equations

As discussed before, derivation of the equations of motion using Lagrange's Equations, requires selecting, defining and using generalized coordinates for the system. In this case, the Euler Angles can be considered as generalized coordinates for the discrete cable model since Euler Angles are orthogonal and independent of each other and the configuration described here can fully define the node positions for the system.

Similar to the previous chapter, the generalized coordinates, q_i are used in the derivations for the 3D model. As used before, the Lagrangian of the system is shown by $\mathscr{L}$ and is defined as [23]:

$$\mathscr{L} = T - U \qquad (\ 4.11\ )$$

Similar to the derivation of Lagrange's equations for a multi-body system consisting of *n* rigid bodies in Chapter 3, the kinetic and potential energies are derived as well as their derivatives to substitute in the Lagrange's Equations of the form:

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{q}_i}\right)-\frac{\partial T}{\partial q_i}=Q_i \qquad (i=1,2,\ldots,3n) \tag{4.12}$$

where q_i's are the generalized coordinates used to define the system and Q_i*'s* are generalized forces which include applied forces and inertial forces derived from the potential energy. The kinetic and potential energies of the system will be derived in order to complete the derivation of Lagrange's Equations in the following sections

4.3.1. Kinetic and Potential Energy and Their Derivatives

The squared velocity vector of each node on the cable with a mass m_i can be expressed as:

$$\vec{V}_k^T \vec{V}_k = \left|\vec{V}_k\right|^2 = \dot{X}_k^{\ 2} + \dot{Y}_k^{\ 2} + \dot{Z}_k^{\ 2} \qquad (k=1,2,\ldots,n) \tag{4.13}$$

Also, taking derivative of Equation **Error! Reference source not found.** with respect to time, one can define the velocity components for each node in terms of Euler Angles as stated in Equation (4.14).

$$\begin{cases} \dot{X}_k = \sum_{j=1}^{k} l_j \left(-\dot{\psi}_j \sin\psi_j \cos\theta_j - \dot{\theta}_j \cos\psi_j \sin\theta_j\right) \\ \dot{Y}_k = \sum_{j=1}^{k} l_j \left(\dot{\psi}_j \cos\psi_j \cos\theta_j - \dot{\theta}_j \sin\psi_j \sin\theta_j\right) \\ \dot{Z}_k = \sum_{j=1}^{k} l_j \left(-\dot{\theta}_j \cos\theta_j\right) \end{cases} \qquad (k=1,2,3,\ldots,n) \tag{4.14}$$

Therefore, by taking each component to the power of two one can write:

$$
\begin{cases}
\dot{X}_k^{\;2} = \sum_{i=1}^{k}\sum_{j=1}^{k} l_i l_j \begin{bmatrix} \dot{\psi}_i\dot{\psi}_j \sin\psi_i \sin\psi_j \cos\theta_i \cos\theta_j + \ldots \\ \dot{\theta}_i\dot{\theta}_j \cos\psi_i \cos\psi_j \sin\theta_i \sin\theta_j + \ldots \\ 2\dot{\psi}_i\dot{\theta}_j \sin\psi_i \cos\theta_i \cos\psi_j \sin\theta_j \end{bmatrix} \\
\dot{Y}_k^{\;2} = \sum_{i=1}^{k}\sum_{j=1}^{k} l_i l_j \begin{bmatrix} \dot{\psi}_i\dot{\psi}_j \cos\psi_i \cos\psi_j \cos\theta_i \cos\theta_j + \ldots \\ \dot{\theta}_i\dot{\theta}_j \sin\psi_i \sin\psi_j \sin\theta_i \sin\theta_j - \ldots \\ 2\dot{\psi}_i\dot{\theta}_j \cos\psi_i \cos\theta_i \sin\psi_j \sin\theta_j \end{bmatrix} \qquad ,(k = 1,2,3,\ldots,n) \\
\dot{Z}_k^{\;2} = \sum_{i=1}^{k}\sum_{j=1}^{k} l_i l_j \left[\dot{\theta}_i\dot{\theta}_j \cos\theta_i \cos\theta_j \right]
\end{cases}
\qquad (\ 4.15\ )
$$

Substituting into Equation (4.13) to achieve the squared velocity vector in terms of the Euler Angles, one can write:

$$
\begin{aligned}
\overrightarrow{V_k}^{2} = {} & \sum_{i=1}^{k}\sum_{j=1}^{k} l_i l_j \Big\{ \dot{\psi}_i\dot{\psi}_j \left(\sin\psi_i \sin\psi_j \cos\theta_i \cos\theta_j + \cos\psi_i \cos\psi_j \cos\theta_i \cos\theta_j \right) + \\
& + \dot{\theta}_i\dot{\theta}_j \left(\cos\psi_i \cos\psi_j \sin\theta_i \sin\theta_j + \sin\psi_i \sin\psi_j \sin\theta_i \sin\theta_j \right) + \\
& 2\dot{\psi}_i\dot{\theta}_j \left(\sin\psi_i \cos\theta_i \cos\psi_j \sin\theta_j - \cos\psi_i \cos\theta_i \sin\psi_j \sin\theta_j \right) + \dot{\theta}_i\dot{\theta}_j \cos\theta_i \cos\theta_j \Big\}
\end{aligned}
\qquad (\ 4.16\ )
$$

which can further be reduced to:

$$
\begin{aligned}
\overrightarrow{V_k}^{2} = {} & \sum_{i=1}^{k}\sum_{j=1}^{k} l_i l_j \Big\{ \dot{\psi}_i\dot{\psi}_j \cos\theta_i \cos\theta_j \cos(\psi_i - \psi_j) + \\
& \dot{\theta}_i\dot{\theta}_j \left(\sin\theta_i \sin\theta_j \cos(\psi_i - \psi_j) + \cos\theta_i \cos\theta_j \right) + \\
& 2\dot{\psi}_i\dot{\theta}_j \cos\theta_i \sin\theta_j \sin\left(\psi_i - \psi_j\right) \Big\} \qquad (k = 1,2,\ldots,n)
\end{aligned}
\qquad (\ 4.17\ )
$$

Now, using the velocity vector derived for each mass, the kinetic energy of the system can be represented as:

$$\begin{aligned}T=&\frac{1}{2}\sum_{k=1}^{n}\sum_{i=1}^{k}\sum_{j=1}^{k}m_k l_i l_j\left\{\dot{\psi}_i\dot{\psi}_j\cos\theta_i\cos\theta_j\cos(\psi_i-\psi_j)+\right.\\&\dot{\theta}_i\dot{\theta}_j\left(\sin\theta_i\sin\theta_j\cos(\psi_i-\psi_j)+\cos\theta_i\cos\theta_j\right)+\\&\left.2\dot{\psi}_i\dot{\theta}_j\cos\theta_i\sin\theta_j\sin\left(\psi_i-\psi_j\right)\right\}\end{aligned}\tag{4.18}$$

In order to complete the derivation of Lagrange's equations, the derivatives of this kinetic energy with respect to the Euler Angles and their time rates should be derived.

$$\begin{aligned}T=&\frac{1}{2}\sum_{k=1}^{n}\sum_{i=1}^{k}\sum_{j=1}^{k}m_k l_i l_j\left\{\right.\\&\dot{\psi}_i\left[\dot{\psi}_j\cos\theta_i\cos\theta_j\cos(\psi_i-\psi_j)+\dot{\theta}_j\cos\theta_i\sin\theta_j\sin\left(\psi_i-\psi_j\right)\right]+\\&\left.\dot{\theta}_i\left[-\dot{\psi}_j\cos\theta_j\sin\theta_i\sin\left(\psi_i-\psi_j\right)+\dot{\theta}_j\left(\sin\theta_i\sin\theta_j\cos(\psi_i-\psi_j)+\cos\theta_i\cos\theta_j\right)\right]\right\}\end{aligned}\tag{4.19}$$

Differentiating the kinetic energy with respect to the heading angle ψ and its rate $\dot{\psi}$, yields:

$$\begin{aligned}\frac{\partial T}{\partial\psi_i}=&\sum_{k=i}^{n}\sum_{j=1}^{k}m_k l_i l_j\left\{\right.\\&\dot{\psi}_i\left[-\dot{\psi}_j\cos\theta_i\cos\theta_j\sin(\psi_i-\psi_j)+\dot{\theta}_j\cos\theta_i\sin\theta_j\cos\left(\psi_i-\psi_j\right)\right]+\\&\left.\dot{\theta}_i\left[-\dot{\theta}_j\sin\theta_i\sin\theta_j\sin(\psi_i-\psi_j)-\dot{\psi}_j\cos\theta_j\sin\theta_i\cos\left(\psi_i-\psi_j\right)\right]\right\}\\&,(i=1,2,\ldots,n)\end{aligned}\tag{4.20}$$

$$\begin{aligned}\frac{\partial T}{\partial\dot{\psi}_i}=&\sum_{k=i}^{n}\sum_{j=1}^{k}m_k l_i l_j\left\{\right.\\&\left.\dot{\psi}_j\cos\theta_i\cos\theta_j\cos(\psi_i-\psi_j)+\dot{\theta}_j\cos\theta_i\sin\theta_j\sin\left(\psi_i-\psi_j\right)\right\}\quad,(i=1,2,\ldots,n)\end{aligned}\tag{4.21}$$

The next term needed in the Lagrange's Equations is the time-derivative of Equation (4.21). Taking derivative of Equation (4.21) with respect to time yields:

$$\begin{aligned}
&\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{\psi}_i}\right)=\sum_{k=i}^{n}\sum_{j=1}^{k} m_k l_i l_j \{ \\
&\ddot{\psi}_j\left(\cos\theta_i \cos\theta_j \cos(\psi_i-\psi_j)\right)+\ddot{\theta}_j\left(\cos\theta_i \sin\theta_j \sin(\psi_i-\psi_j)\right)+ \\
&\dot{\psi}_j^{\,2}\left(\cos\theta_i \cos\theta_j \sin(\psi_i-\psi_j)\right)+\dot{\theta}_j^{\,2}\left(\cos\theta_i \cos\theta_j \sin(\psi_i-\psi_j)\right)+ \\
&\dot{\psi}_i\dot{\psi}_j\left(-\cos\theta_i \cos\theta_j \sin(\psi_i-\psi_j)\right)+\dot{\theta}_i\dot{\theta}_j\left(-\sin\theta_i \sin\theta_j \sin(\psi_i-\psi_j)\right)+ \\
&\dot{\psi}_j\dot{\theta}_j\left(-2\cos\theta_i \sin\theta_j \cos(\psi_i-\psi_j)\right)+\dot{\psi}_i\dot{\theta}_j\left(\cos_i \sin\theta_j \cos(\psi_i-\psi_j)\right)+ \\
&\dot{\theta}_i\dot{\psi}_j\left(-\sin\theta_i \cos\theta_j \cos(\psi_i-\psi_j)\right)\} \qquad ,(i=1,2,\ldots,n)
\end{aligned} \tag{4.22}$$

Following the same logic as in section 3.3.3 for derivation of generalized forces, and assuming there is no strain or elasticity in the links, the only potential energy the system preserves is of gravitational form. Selecting the datum line at which *U=0,* to be on the ground, i.e. *Z=0,* the generalized forces for this generalized coordinates can be expressed as:

$$Q_{\psi_i}=\sum_{j=1}^{n}\left[F_{X_j}\frac{\partial X_j}{\partial \psi_i}+F_{Y_j}\frac{\partial Y_j}{\partial \psi_i}+\left(F_{Z_j}-m_j g\right)\frac{\partial Z_j}{\partial \psi_i}+M_{\psi_j}\frac{\partial \psi_j}{\partial \psi_i}\right] \quad ,(i=1,2,\ldots,n) \tag{4.23}$$

Based on Equation (4.9)**Error! Reference source not found.**, deriving the rate of global coordinates with respect to the heading angle yields:

$$\begin{cases}
\dfrac{\partial X_j}{\partial \psi_i}=-l_i \sin\psi_i \cos\theta_i \\
\dfrac{\partial Y_j}{\partial \psi_i}=l_i \cos\psi_i \cos\theta_i \qquad (i=1,2,3,\ldots,n),\ j\ge i \\
\dfrac{\partial Z_j}{\partial \psi_i}=0
\end{cases} \tag{4.24}$$

Substituting back into Equation (4.23), the generalized forces in terms of the external forces and the segment mass could be written as:

$$Q_{\psi_i} = \sum_{j=i}^{n} \left[F_{X_j}(-l_i \sin\psi_i \cos\theta_i) + F_{Y_j}(l_i \cos\psi_i \cos\theta_i) \right] + M_{\psi_i} \qquad ,(i=1,2,\ldots,n) \qquad (4.25)$$

Therefore Lagrange's Equations for the heading angles as one set of the system's equations, can be stated as follows.

$$\begin{gathered}
\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{\psi}_i}\right) - \frac{\partial T}{\partial \psi_i} = \sum_{k=i}^{n}\sum_{j=1}^{k} m_k l_i l_j \{ \\
\ddot{\psi}_j \left(\cos\theta_i \cos\theta_j \cos(\psi_i - \psi_j)\right) + \ddot{\theta}_j \left(\cos\theta_i \sin\theta_j \sin(\psi_i - \psi_j)\right) + \\
\dot{\psi}_j^{\,2} \left(\cos\theta_i \cos\theta_j \sin(\psi_i - \psi_j)\right) + \dot{\theta}_j^{\,2} \left(\cos\theta_i \cos\theta_j \sin(\psi_i - \psi_j)\right) + \\
\dot{\psi}_j \dot{\theta}_j \left(-2\cos\theta_i \sin\theta_j \cos(\psi_i - \psi_j)\right)\} = \\
\sum_{j=i}^{n} \left[F_{X_j}(-l_i \sin\psi_i \cos\theta_i) + F_{Y_j}(l_i \cos\psi_i \cos\theta_i) \right] + M_{\psi_i} \qquad ,(i=1,2,\ldots,n)
\end{gathered} \qquad (4.26)$$

This concludes the derivation of the system's equation of motion in terms of the heading angles of the segments.

Following the same path and deriving the equations in terms of the pitch angles θ , derivative of the kinetic energy with respect to θ and $\dot{\theta}$ can be written as:

$$
\begin{aligned}
&\frac{\partial T}{\partial \theta_i} = \sum_{k=i}^{n} \sum_{j=1}^{k} m_k l_i l_j \Big\{ \\
&\dot{\psi}_i \left[-\dot{\psi}_j \sin\theta_i \cos\theta_j \cos(\psi_i - \psi_j) - \dot{\theta}_j \sin\theta_i \sin\theta_j \sin\left(\psi_i - \psi_j\right) \right] + \\
&\dot{\theta}_i \left[\dot{\theta}_j \left(\cos\theta_i \sin\theta_j \cos(\psi_i - \psi_j) - \sin\theta_i \cos\theta_j \right) - \dot{\psi}_j \cos\theta_i \cos\theta_j \sin\left(\psi_i - \psi_j\right) \right] \Big\} \\
&\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad , (i = 1, 2, \ldots, n)
\end{aligned}
\tag{4.27}
$$

$$
\begin{aligned}
&\frac{\partial T}{\partial \dot{\theta}_i} = \sum_{k=i}^{n} \sum_{j=1}^{k} m_k l_i l_j \Big\{ \\
&\dot{\theta}_j \left(\sin\theta_i \sin\theta_j \cos\left(\psi_i - \psi_j\right) + \cos\theta_i \cos\theta_j \right) - \\
&\dot{\psi}_j \sin\theta_i \cos\theta_j \sin(\psi_i - \psi_j) \Big\} \qquad , (i = 1, 2, \ldots, n)
\end{aligned}
\tag{4.28}
$$

Therefore, after taking the time-derivative of Equation (4.28), it can be written as:

$$
\begin{aligned}
&\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{\theta}_i} \right) = \sum_{k=i}^{n} \sum_{j=1}^{k} m_k l_i l_j \Big\{ \\
&\ddot{\psi}_j \left(-\sin\theta_i \cos\theta_j \sin(\psi_i - \psi_j) \right) + \ddot{\theta}_j \left(\sin\theta_i \sin\theta_j \cos(\psi_i - \psi_j) + \cos\theta_i \cos\theta_j \right) + \\
&\dot{\psi}_j^{\,2} \left(\sin\theta_i \cos\theta_j \cos(\psi_i - \psi_j) \right) + \dot{\theta}_j^{\,2} \left(\sin\theta_i \cos\theta_j \cos(\psi_i - \psi_j) - \cos\theta_i \sin\theta_j \right) + \\
&\dot{\psi}_i \dot{\psi}_j \left(-\sin\theta_i \cos\theta_j \cos(\psi_i - \psi_j) \right) + \dot{\theta}_i \dot{\theta}_j \left(\cos\theta_i \sin\theta_j \cos(\psi_i - \psi_j) - \sin\theta_i \cos\theta_j \right) + \\
&\dot{\psi}_j \dot{\theta}_j \left(2 \sin\theta_i \sin\theta_j \sin(\psi_i - \psi_j) \right) + \dot{\psi}_i \dot{\theta}_j \left(-\sin\theta_i \sin\theta_j \sin(\psi_i - \psi_j) \right) + \\
&\dot{\theta}_i \dot{\psi}_j \left(-\cos\theta_i \cos\theta_j \sin(\psi_i - \psi_j) \right) \Big\} \qquad\qquad\qquad\qquad , (i = 1, 2, \ldots, n)
\end{aligned}
\tag{4.29}
$$

Derivatives of the global coordinates in terms of the pitch angle can be written as follows:

$$\begin{cases} \dfrac{\partial X_j}{\partial \theta_i} = -l_i \cos\psi_i \sin\theta_i \\ \dfrac{\partial Y_j}{\partial \theta_i} = -l_i \sin\psi_i \sin\theta_i \\ \dfrac{\partial Z_j}{\partial \theta_i} = -l_i \cos\theta_i \end{cases} \qquad (i = 1,2,3,\ldots,n),\ j \geq i \tag{4.30}$$

Then, these equations are substituted into the following equation, to define the generalized forces for the pitch angle generalized coordinates.

$$Q_{\theta_i} = \sum_{j=1}^{n} \left[F_{X_j} \frac{\partial X_j}{\partial \theta_i} + F_{Y_j} \frac{\partial Y_j}{\partial \theta_i} + \left(F_{Z_j} - m_j g \right) \frac{\partial Z_j}{\partial \theta_i} + M_{\theta_j} \frac{\partial \theta_j}{\partial \theta_i} \right] \quad ,(i = 1,2,\ldots,n) \tag{4.31}$$

Therefore, the generalized forces are reduced to:

$$Q_{\theta_i} = \sum_{j=i}^{n} \left[F_{X_j} (-l_i \cos\psi_i \sin\theta_i) + F_{Y_j} (-l_i \sin\psi_i \sin\theta_i) + \left(F_{Z_j} - m_j g \right) (-l_i \cos\theta_i) \right] + M_{\theta_i} \quad ,(i = 1,2,\ldots,n) \tag{4.32}$$

Therefore Lagrange’s Equations for the pitch angles can be expressed as follows:

$$
\begin{aligned}
&\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{\theta}_i}\right)-\frac{\partial T}{\partial \theta_i}=\sum_{k=i}^{n}\sum_{j=1}^{k} m_k l_i l_j \{ \\
&\ddot{\psi}_j\left(-\sin\theta_i\cos\theta_j\sin(\psi_i-\psi_j)\right)+\ddot{\theta}_j\left(\sin\theta_i\sin\theta_j\cos(\psi_i-\psi_j)+\cos\theta_i\cos\theta_j\right)+ \\
&\dot{\psi}_j{}^2\left(\sin\theta_i\cos\theta_j\cos(\psi_i-\psi_j)\right)+\dot{\theta}_j{}^2\left(\sin\theta_i\cos\theta_j\cos(\psi_i-\psi_j)-\cos\theta_i\sin\theta_j\right)+ \\
&\dot{\psi}_j\dot{\theta}_j\left(2\sin\theta_i\sin\theta_j\sin(\psi_i-\psi_j)\right)\Big\}= \\
&\sum_{j=i}^{n}\Big[F_{X_j}(-l_i\cos\psi_i\sin\theta_i)+F_{Y_j}(-l_i\sin\psi_i\sin\theta_i)+\left(F_{Z_j}-m_j g\right)(-l_i\cos\theta_i)\Big]+M_{\theta_i} \\
&\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad ,(i=1,2,\ldots,n)
\end{aligned}
\qquad (\ 4.33\ )
$$

Equation (4.33) shows the equations of motion based on the pitch angle coordinate using Lagrange’s Equations which should be solved in conjunction with all other equation sets derived. Please note that Equations (4.26) and (4.33) are in fact two sets of equations each with *n* equations in terms of the corresponding generalized coordinates. Therefore these two equations represent *2n* equations for the *2n*-DOF system which can be solved numerically.

The last set of generalized coordinates to consider for the Lagrange’s Equations is the bank or roll angle ϕ. Looking at the equations of kinetic energy, it can be confirmed that both the translational kinetic energy as well as the generalized forces are independent of this coordinate. Therefore, derivation of Lagrange’s Equations based on the bank angles is inconclusive at this point. However, it will be shown later in Section 4.5.1 that implementation of the rotational kinetic energies will add bank angle’s equations of motion to the system.

To this point, all the equations of motion for the cable considering zero twist has been derived and should be solved numerically after substituting the proper external forces into the generalized force equations, i.e. Equations (4.25) and (4.32).

4.3.2. Assembling Equations in a Matrix Form

All the *2n* equations of motion derived in the previous section should be solved in parallel to yield the solutions for the system dynamics. All the equations should be assembled in an easy-to-use matrix form to be plugged into a similar algorithm developed in Chapter 3. However, in order to comply with the equations for the UAS dynamics derived later, one needs to implement these equations into a general *3n×3n* matrix format.

Based on Equations (4.26) and (4.33), the system's equations of motion can be assembled in the following form:

$$\mathbf{A}\ddot{\boldsymbol{\eta}} + \mathbf{B}diag\left\{\dot{\boldsymbol{\eta}}\right\}\dot{\boldsymbol{\eta}} + \mathbf{C}diag\begin{Bmatrix}\dot{\boldsymbol{\theta}}\\ \dot{\boldsymbol{\phi}}\\ \dot{\boldsymbol{\psi}}\end{Bmatrix}\begin{bmatrix}\dot{\boldsymbol{\psi}}\\ \dot{\boldsymbol{\theta}}\\ \dot{\boldsymbol{\phi}}\end{bmatrix} = \mathbf{Q} \qquad (4.34)$$

in which, $\boldsymbol{\psi}$ and $\boldsymbol{\theta}$, and $\boldsymbol{\phi}$ are vectors representing heading and pitch angles of all elements. Vector $\boldsymbol{\eta}$ is also called attitude vector representing the Euler Angles in a vector of *3n×1* elements and is represented by:

$$\boldsymbol{\eta} = \begin{bmatrix}\boldsymbol{\psi}\\ \boldsymbol{\theta}\\ \boldsymbol{\phi}\end{bmatrix} \qquad (4.35)$$

Therefore, we can write Equation (3.26) as:

$$\begin{bmatrix}\mathbf{A}_{11} & \mathbf{A}_{12} & \mathbf{A}_{13}\\ \mathbf{A}_{21} & \mathbf{A}_{22} & \mathbf{A}_{23}\\ \mathbf{A}_{31} & \mathbf{A}_{32} & \mathbf{A}_{33}\end{bmatrix}\begin{bmatrix}\ddot{\boldsymbol{\psi}}\\ \ddot{\boldsymbol{\theta}}\\ \ddot{\boldsymbol{\phi}}\end{bmatrix} + \begin{bmatrix}\mathbf{B}_{11} & \mathbf{B}_{12} & \mathbf{B}_{13}\\ \mathbf{B}_{21} & \mathbf{B}_{22} & \mathbf{B}_{23}\\ \mathbf{B}_{31} & \mathbf{B}_{32} & \mathbf{B}_{33}\end{bmatrix} diag\begin{Bmatrix}\dot{\boldsymbol{\psi}}\\ \dot{\boldsymbol{\theta}}\\ \dot{\boldsymbol{\phi}}\end{Bmatrix}\begin{bmatrix}\dot{\boldsymbol{\psi}}\\ \dot{\boldsymbol{\theta}}\\ \dot{\boldsymbol{\phi}}\end{bmatrix} + \begin{bmatrix}\mathbf{C}_{11} & \mathbf{C}_{12} & \mathbf{C}_{13}\\ \mathbf{C}_{21} & \mathbf{C}_{22} & \mathbf{C}_{23}\\ \mathbf{C}_{31} & \mathbf{C}_{32} & \mathbf{C}_{33}\end{bmatrix}\begin{bmatrix}\dot{\boldsymbol{\theta}}\dot{\boldsymbol{\psi}}\\ \dot{\boldsymbol{\phi}}\dot{\boldsymbol{\theta}}\\ \dot{\boldsymbol{\psi}}\dot{\boldsymbol{\phi}}\end{bmatrix} = \begin{bmatrix}\mathbf{Q}_{\psi}\\ \mathbf{Q}_{\theta}\\ \mathbf{Q}_{\phi}\end{bmatrix} \qquad (4.36)$$

Although, since there is no equation in terms of the roll angle in this case, and all terms of the Lagrange's Equations are zero, the previous equation can be expressed as:

$$\begin{bmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} & \underline{0} \\ \mathbf{A}_{21} & \mathbf{A}_{22} & \underline{0} \\ \underline{0} & \underline{0} & \underline{0} \end{bmatrix} \begin{bmatrix} \ddot{\boldsymbol{\psi}} \\ \ddot{\boldsymbol{\theta}} \\ \ddot{\boldsymbol{\phi}} \end{bmatrix} + \begin{bmatrix} \mathbf{B}_{11} & \mathbf{B}_{12} & \underline{0} \\ \mathbf{B}_{21} & \mathbf{B}_{22} & \underline{0} \\ \underline{0} & \underline{0} & \underline{0} \end{bmatrix} diag \begin{Bmatrix} \dot{\boldsymbol{\psi}} \\ \dot{\boldsymbol{\theta}} \\ \dot{\boldsymbol{\phi}} \end{Bmatrix} \begin{bmatrix} \dot{\boldsymbol{\psi}} \\ \dot{\boldsymbol{\theta}} \\ \dot{\boldsymbol{\phi}} \end{bmatrix} + \begin{bmatrix} \mathbf{C}_{11} & \underline{0} & \underline{0} \\ \mathbf{C}_{21} & \underline{0} & \underline{0} \\ \underline{0} & \underline{0} & \underline{0} \end{bmatrix} \begin{bmatrix} \dot{\boldsymbol{\theta}}\dot{\boldsymbol{\psi}} \\ \dot{\boldsymbol{\phi}}\dot{\boldsymbol{\theta}} \\ \dot{\boldsymbol{\phi}}\dot{\boldsymbol{\psi}} \end{bmatrix} = \begin{bmatrix} \mathbf{Q}_{\psi} \\ \mathbf{Q}_{\theta} \\ [0] \end{bmatrix} \qquad (4.37)$$

where $\underline{0}$ represents a zero matrix of size *n*×*n* and sub-matrix components of matrix $\mathbf{A}$ are defined as:

$$\begin{cases} [\mathbf{A}_{11}]_{i,j} = \mathbf{M}_{\mathrm{ij}} l_i l_j \left(\cos\theta_i \cos\theta_j \cos(\psi_i - \psi_j)\right) \\ [\mathbf{A}_{12}]_{i,j} = \mathbf{M}_{\mathrm{ij}} l_i l_j \left(\cos\theta_i \sin\theta_j \sin(\psi_i - \psi_j)\right) \\ [\mathbf{A}_{21}]_{i,j} = \mathbf{M}_{\mathrm{ij}} l_i l_j \left(-\sin\theta_i \cos\theta_j \sin(\psi_i - \psi_j)\right) \\ [\mathbf{A}_{22}]_{i,j} = \mathbf{M}_{\mathrm{ij}} l_i l_j \left(\sin\theta_i \sin\theta_j \cos(\psi_i - \psi_j) + \cos\theta_i \cos\theta_j\right) \end{cases} \qquad (i, j = 1, 2, \ldots, n) \qquad (4.38)$$

and sub-matrices of matrix $\mathbf{B}$ are:

$$\begin{cases} [\mathbf{B}_{11}]_{i,j} = \mathbf{M}_{\mathrm{ij}} l_i l_j \left(\cos\theta_i \cos\theta_j \sin(\psi_i - \psi_j)\right) \\ [\mathbf{B}_{12}]_{i,j} = \mathbf{M}_{\mathrm{ij}} l_i l_j \left(\cos\theta_i \cos\theta_j \sin(\psi_i - \psi_j)\right) \\ [\mathbf{B}_{21}]_{i,j} = \mathbf{M}_{\mathrm{ij}} l_i l_j \left(\sin\theta_i \cos\theta_j \cos(\psi_i - \psi_j)\right) \\ [\mathbf{B}_{22}]_{i,j} = \mathbf{M}_{\mathrm{ij}} l_i l_j \left(\sin\theta_i \cos\theta_j \cos(\psi_i - \psi_j) - \cos\theta_i \sin\theta_j\right) \end{cases} \qquad (i, j = 1, 2, \ldots, n) \qquad (4.39)$$

and finally, matrix $\mathbf{C}$ consists of two sub-matrices:

$$\begin{cases} [\mathbf{C}_{11}]_{i,j} = \mathbf{M}_{\mathrm{ij}} l_i l_j \left(-2\cos\theta_i \sin\theta_j \cos(\psi_i - \psi_j)\right) \\ [\mathbf{C}_{21}]_{i,j} = \mathbf{M}_{\mathrm{ij}} l_i l_j \left(2\sin\theta_i \sin\theta_j \sin(\psi_i - \psi_j)\right) \end{cases} \qquad (i, j = 1, 2, \ldots, n) \qquad (4.40)$$

based on equations of motion derived in (4.26) and (4.33). Matrix $\mathbf{M}$ is also called the mass distribution matrix by Quisenberry [16] and is represented as:

$$\mathbf{M}_{\mathrm{ij}} = \sum_{k=\max\{i,j\}}^{n} m_k \qquad (i, j = 1, 2, \ldots n) \tag{4.41}$$

By implementing the generalized forces in terms of external forces in Equation (4.36), the equations can be solved numerically using appropriate initial conditions.

4.4. External Forces

For the case of the 3D model, the cross flow principle introduced by Hoerner [32] is used to calculate the external forces due to wind. Equations (4.25) and (4.32) suggest that generalized force vectors $\mathbf{Q}_{\psi}$ and $\mathbf{Q}_{\theta}$ require the external forces and moments applied to the system to be derived and entered in terms of the Euler Angles in order to solve the equations. It should be emphasized once again that the so-called generalized forces are in fact moments in these equations, i.e. the dimension is actually *N.m* and not *N*. This is due to the fact that the choice of generalized coordinates are angles rather than linear positions. This ensures that the equations derived which are energy-based are consistent in terms of dimension.

It is assumed that the flow field is uniform in each direction and the wind profile in a certain direction is constant with respect to the other 2 global coordinates; that is, wind speed in the *X* direction is constant in terms of *Y* and *Z*, and so on. However; the model can be simply extended to a variable profile wind condition with correct implementation of the wind profile values in the solver algorithm developed here.

Cross-Flow Principle

According to Section 3.4, one can extend the cross flow principle to a 3D cable segment model. Thus having the rod model illustrated in Section 3.4, consider the wind velocity is defined as the following vector:

$$V_{w0} = \begin{pmatrix} U \\ V \\ W \end{pmatrix} \tag{4.42}$$

Therefore one can express the wind velocity relative to the segment's speed as:

$$V_{w,rel} = \begin{pmatrix} U - \dot{X} \\ V - \dot{Y} \\ W - \dot{Z} \end{pmatrix} \tag{4.43}$$

Defining the angle of attack *α* as the angle between the wind velocity vector and the longitudinal axes of the cylinder as shown in Figure 3.6, and assuming tangential pressure drag to be negligible, and neglecting tip vortices, and employing Hoerner's cross flow principle for a slender cylinder in the uniform subcritical wind flow [32], the forces exerted by the wind on a cable segment can be expressed as:

$$\begin{pmatrix} F_x \\ F_y \\ F_z \end{pmatrix} = \begin{pmatrix} \frac{1}{2}\rho dl\pi C_f V_{x,rel}\sqrt{V_{x,rel}{}^2 + V_{y,rel}{}^2 + V_{z,rel}{}^2} \\ \frac{1}{2}\rho dl V_{y,rel}\left(\pi C_f\sqrt{V_{x,rel}{}^2 + V_{y,rel}{}^2 + V_{z,rel}{}^2} + C_D\sqrt{V_{y,rel}{}^2 + V_{z,rel}{}^2}\right) \\ \frac{1}{2}\rho dl V_{z,rel}\left(\pi C_f\sqrt{V_{x,rel}{}^2 + V_{y,rel}{}^2 + V_{z,rel}{}^2} + C_D\sqrt{V_{y,rel}{}^2 + V_{z,rel}{}^2}\right) \end{pmatrix} \tag{4.44}$$

in which the relative wind velocity normal components in the segment local frame, $V_{x,rel}$ $V_{y,rel}$,and $V_{z,rel}$ can be determined by transforming the relative wind velocity in the inertial frame to each body coordinate system using Euler Angle transformation matrix $\mathbf{R}_{\phi\theta\psi}$ defined in Equation **Error! Reference source not found.**.

$$\begin{pmatrix} V_{x,rel_i} \\ V_{y,rel_i} \\ V_{z,rel_i} \end{pmatrix} = \mathbf{R}_{\phi\theta\psi} \times \begin{pmatrix} U - \dot{X} \\ V - \dot{Y} \\ W - \dot{Z} \end{pmatrix} \qquad (4.45)$$

It is also worthwhile to note that the tangential and normal forces in each segment's local frame shown in Equation **Error! Reference source not found.** can be transformed back and forth to the inertial frame using the same rotation matrix; that is:

$$\begin{pmatrix} F_{X_i} \\ F_{Y_i} \\ F_{Z_i} \end{pmatrix} = \mathbf{R}_{\phi\theta\psi}{}^{-1} \times \begin{pmatrix} F_{x_i} \\ F_{y_i} \\ F_{z_i} \end{pmatrix} \qquad (4.46)$$

These forces will be used in deriving the generalized forces for the purpose of solving the equations of motion.

4.5. UAS Dynamics

It is important to consider the UAS dynamics in the modelling of the system in order to capture the dynamic and static behavior of the system with a closer approximation. The same procedure used in Chapter 3 is followed in order to implement the dynamics of the UAS into the equations of motion.

4.5.1. Implementation of UAS Segment into the Equations

The UAS is considered to be another extra segment added to the *n*-segments modeled before and derive the Lagrange's Equations for this *n+1*th segment. Since each segment is considered to have 2 degrees of freedom, there are two equation derived for the UAS which are then added to the matrix equations derived before. It is assumed this particular segment expands from the last mass node which happens to be on the cable-UAS connection point to the center of gravity of the UAS so that it complies with the lumped mass model. In essence, the UAS, is reduced to a cable segment with a mass equal to the UAS's and mass moment of inertia according to that of the UAS.

A new axis is defined as the UAS's principal axis and is labeled as $x_q y_q z_q$ to be able to define the orientation of the vehicle with respect to the inertial frame in terms of Euler Angles. The origin of this coordinate system is located at the center of mass of the UAS, x_q axes pointing towards the forward face of the UAS and z_q axes pointing towards the top of the UAS while the y_q axes point towards port as shown in Figure 4.2. In this work, the cable/UAS connection is constrained not to rotate about its longitudinal axes. That is, it can rotate about x_q and y_q axis but is not able to rotate freely about the z_q axes. Thus, the UAS $x_q z_q$ plane will always be in the same plane as the virtual segment plane $x_{n+1} z_{n+1}$. Using this configuration, vectors in the UAS coordinate system, $x_q y_q z_q$, can be transformed to the UAS virtual segment coordinate system, i.e. $x_{n+1} y_{n+1} z_{n+1}$ through rotation with an angle γ by using the following rotation matrix:

$$\mathbf{R}_\gamma = \begin{bmatrix} \cos\gamma & 0 & \sin\gamma \\ 0 & 1 & 0 \\ -\sin\gamma & 0 & \cos\gamma \end{bmatrix} \tag{4.47}$$

In this system, the connection point is exactly below the center of mass, which makes the angle $\gamma = 90^\circ$. This causes the UAS coordinates to become equal to:

$$\begin{bmatrix} x_{n+1} \\ y_{n+1} \\ z_{n+1} \end{bmatrix} = \begin{bmatrix} z_q \\ y_q \\ -x_q \end{bmatrix} \tag{4.48}$$

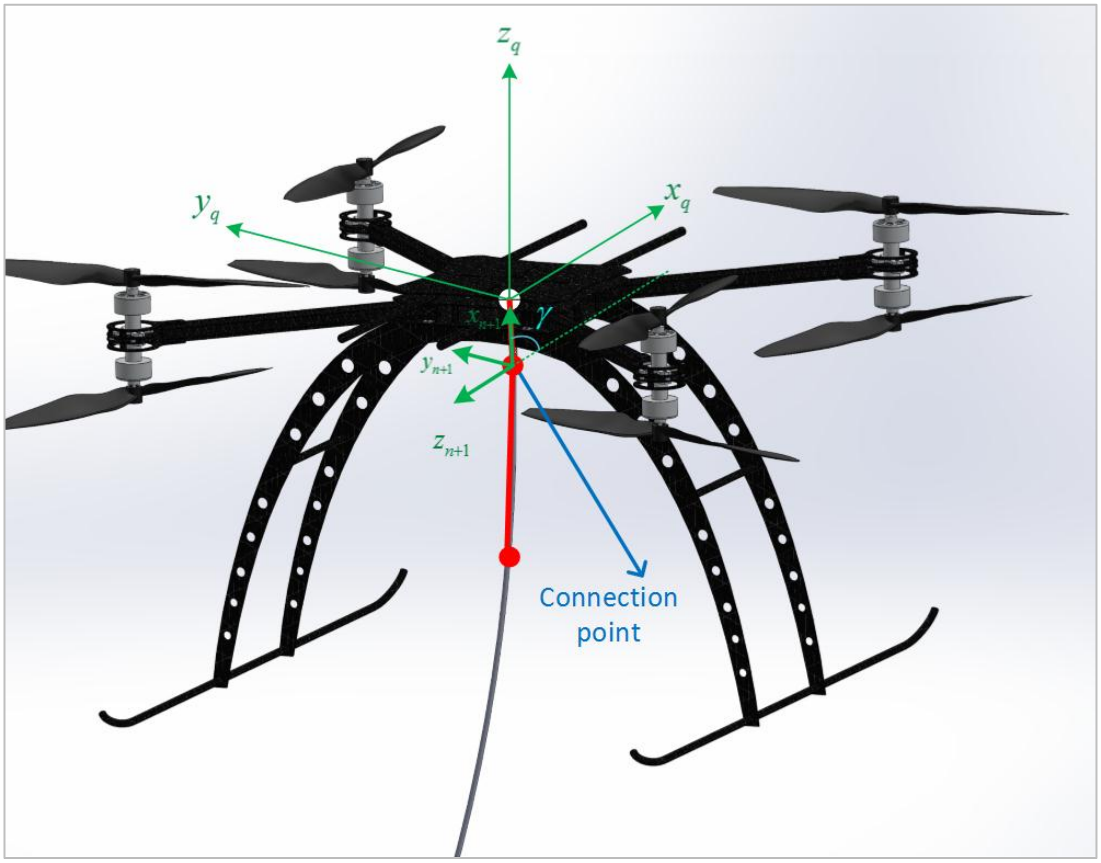

Figure 4.2. UAS 3D Coordinate Setup
Note that the length between the connection point of the cable and the UAS's center of mass is considered as an extra virtual segment

Consider that the UAS has the mass moment of inertia I_{CG} at its center of gravity and about its principal axis $x_q y_q z_q$.

$$\bar{I}_{CG} = \begin{bmatrix} \bar{I}_{xx} & \bar{I}_{xy} & \bar{I}_{xz} \\ \bar{I}_{yx} & \bar{I}_{yy} & \bar{I}_{yz} \\ \bar{I}_{zx} & \bar{I}_{zy} & \bar{I}_{zz} \end{bmatrix} \tag{4.49}$$

in which the off-diagonal elements $\bar{I}_{xy}$, $\bar{I}_{xz}$,and $\bar{I}_{zy}$ are considered to be very close to zero and therefore neglected due to symmetry of the UAS. This matrix of mass moment of inertia should be transformed to a new matrix representing the inertia at the origin of the cable segment and aligned with the virtual segment axis $x_{n+1}y_{n+1}z_{n+1}$.

In order to transform the inertia matrix, one needs to align it with the $x_{n+1}y_{n+1}z_{n+1}$ coordinate system by using the rotation matrix $\mathbf{R}_\gamma$ and then translate it to reflect the inertia in the desired coordinate system. Define the inertia matrix about the center of mass aligned with the UAS segment axis $x_{n+1}y_{n+1}z_{n+1}$ to be called I_{CG}. Then we can state this inertia matrix in terms of the principal inertias as:

$$I_{CG} = \mathbf{R}_\gamma \bar{I}_{CG} \mathbf{R}_\gamma^{-1} \qquad (4.50)$$

This matrix is then needed to be transferred to be calculated about the $x_{n+1}y_{n+1}z_{n+1}$ axis using parallel axes theorem [37], [38], [41]. Therefore:

$$I_{n+1} = \begin{bmatrix} \cos\gamma & 0 & \sin\gamma \\ 0 & 1 & 0 \\ -\sin\gamma & 0 & \cos\gamma \end{bmatrix} \begin{bmatrix} \bar{I}_{xx} & 0 & 0 \\ 0 & \bar{I}_{yy} & 0 \\ 0 & 0 & \bar{I}_{zz} \end{bmatrix} \begin{bmatrix} \cos\gamma & 0 & -\sin\gamma \\ 0 & 1 & 0 \\ \sin\gamma & 0 & \cos\gamma \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ 0 & m_U l_{n+1}^2 & 0 \\ 0 & 0 & m_U l_{n+1}^2 \end{bmatrix} \qquad (4.51)$$

where m_U is the UAS mass and l_{n+1} is the length of the *n+1*th segment, i.e. the distance between the connection point and the center of mass. One can finally write the inertia about the origin of this segment as:

$$I_{n+1} = \begin{bmatrix} I_{x_{n+1}} & 0 & 0 \\ 0 & I_{y_{n+1}} & 0 \\ 0 & 0 & I_{z_{n+1}} \end{bmatrix} = \begin{bmatrix} \bar{I}_{zz} & 0 & 0 \\ 0 & \bar{I}_{yy} + m_U l_{n+1}^2 & 0 \\ 0 & 0 & \bar{I}_{xx} + m_U l_{n+1}^2 \end{bmatrix} \qquad (4.52)$$

The kinetic energy of the multibody system including the UAS segment which consists of a translational and a rotation term could be written as:

$$T = T_{trans.} + T_{rot.} \tag{4.53}$$

In calculation of the rotational part of the kinetic energy of the UAS, one needs to use the angular velocities of the segment about each axis in the inertial coordinate system. Note that the rate of change of the Euler angles, i.e. $\dot{\phi}$, $\dot{\theta}$, and $\dot{\psi}$ are not orthogonal and one needs to use the components of the angular velocity in terms of these three rates. For any segment in the system, including the virtual UAS segment, the angular velocity vector can be stated as [33]:

$$\vec{\boldsymbol{\omega}}_i = \begin{bmatrix} \omega_{x_i} \\ \omega_{y_i} \\ \omega_{z_i} \end{bmatrix} = \begin{bmatrix} \dot{\phi}_i - \dot{\psi}_i \sin\theta_i \\ \dot{\psi}_i \cos\theta_i \sin\phi_i + \dot{\theta}_i \cos\phi_i \\ \dot{\psi}_i \cos\theta_i \cos\phi_i - \dot{\theta}_i \sin\phi_i \end{bmatrix} \tag{4.54}$$

One can calculate the rotational kinetic energy using the angular velocity of each segment as well as inertia matrix as:

$$T_{rot.} = \sum_{j=1}^{n+1} \left(T_{rot.}\right)_j = \frac{1}{2} \sum_{j=1}^{n+1} \vec{\boldsymbol{\omega}}_j^{\mathbf{T}} \mathbf{I}_j \vec{\boldsymbol{\omega}}_j \tag{4.55}$$

Then, substituting the components of the angular velocity vector, the rotational kinetic energy is written as:

$$T_{rot.} = \frac{1}{2} \sum_{j=1}^{n+1} \left(\begin{bmatrix} \omega_{x_j} & \omega_{y_j} & \omega_{z_j} \end{bmatrix} \begin{bmatrix} I_{x_j} & 0 & 0 \\ 0 & I_{y_j} & 0 \\ 0 & 0 & I_{z_j} \end{bmatrix} \begin{bmatrix} \omega_{x_j} \\ \omega_{y_j} \\ \omega_{z_j} \end{bmatrix} \right) \tag{4.56}$$

which can be written as the following summation:

$$T_{rot.} = \frac{1}{2}\sum_{j=1}^{n+1}\left(I_{x_j}\omega_{x_j}^2 + I_{y_j}\omega_{y_j}^2 + I_{z_j}\omega_{z_j}^2 \right) \qquad (4.57)$$

In which, the angular velocity terms can be substituted from Equation (4.54) to achieve the following:

$$\begin{aligned} T_{rot.} = \frac{1}{2}\sum_{j=1}^{n+1}\Big\{ & I_{x_j}\left(\dot{\phi}_j^{\,2} + \dot{\psi}_j^2 \sin^2\theta_j - 2\dot{\phi}_j\dot{\psi}_j \sin\theta_j \right) + \\ & I_{y_j}\left(\dot{\psi}_j^2 \cos^2\theta_j \sin^2\phi_j + \dot{\theta}_j^2 \cos^2\phi_j + 2\dot{\psi}_j\dot{\theta}_j \cos\theta_j \sin\phi_j \cos\phi_j \right) + \\ & I_{z_j}\left(\dot{\psi}_j^2 \cos^2\theta_j \cos^2\phi_j + \dot{\theta}_j^2 \sin^2\phi_j - 2\dot{\psi}_j\dot{\theta}_j \cos\theta_j \cos\phi_j \sin\phi_j \right)\Big\} \end{aligned} \qquad (4.58)$$

The translational part of the Kinetic energy will be in the exact same format as before and in line with representation of kinetic energy for all other segments.

Taking the derivatives of the new kinetic energy of the system with respect to the heading angle ψ_i and its rate yields:

$$\frac{\partial T_{rot.}}{\partial \psi_i} = 0 \qquad (i = 1, 2, \ldots, n+1) \qquad (4.59)$$

$$\begin{aligned} \frac{\partial T_{rot.}}{\partial \dot{\psi}_i} = & I_{x_i}\left(\dot{\psi}_i \sin^2\theta_i - \dot{\phi}_i \sin\theta_i \right) + \\ & I_{y_i}\left(\dot{\psi}_i \cos^2\theta_i \sin^2\phi_i + \dot{\theta}_i \cos\theta_i \sin\phi_i \cos\phi_i \right) + \\ & I_{z_i}\left(\dot{\psi}_i \cos^2\theta_i \cos^2\phi_i - \dot{\theta}_i \cos\theta_i \cos\phi_i \sin\phi_i \right) \qquad (i = 1, 2, \ldots, n+1) \end{aligned} \qquad (4.60)$$

After taking a time-derivative of Equation (4.60), it can be stated as:

$$
\begin{aligned}
&\frac{d}{dt}\left(\frac{\partial T_{rot.}}{\partial \dot{\psi}_i}\right)=I_{x_i}\left[\ddot{\psi}_i \sin^2\theta_i+2\dot{\psi}_i\dot{\theta}_i\cos\theta_i\sin\theta_i-\ddot{\phi}_i\sin\theta_i-\dot{\phi}_i\dot{\theta}_i\cos\theta_i\right]+\\
&I_{y_i}\left[\ddot{\psi}_i\cos^2\theta_i\sin^2\phi_i-2\dot{\psi}_i\dot{\theta}_i\sin\theta_i\cos\theta_i\sin^2\phi_i+2\dot{\psi}_i\dot{\phi}_i\cos^2\theta_i\sin\phi_i\cos\phi_i+\right.\\
&\left.\ddot{\theta}_i\cos\theta_i\sin\phi_i\cos\phi_i-\dot{\theta}_i^2\sin\theta_i\sin\phi_i\cos\phi_i+\dot{\theta}_i\dot{\phi}_i\cos\theta_i\left(\cos^2\phi_i-\sin^2\phi_i\right)\right]+\\
&I_{z_i}\left[\ddot{\psi}_i\cos^2\theta_i\cos^2\phi_i-2\dot{\psi}_i\dot{\theta}_i\cos\theta_i\sin\theta_i\cos^2\phi_i-2\dot{\psi}_i\dot{\phi}_i\cos^2\theta_i\cos\phi_i\sin\phi_i-\right.\\
&\left.\ddot{\theta}_i\cos\theta_i\cos\phi_i\sin\phi_i+\dot{\theta}_i^2\sin\theta_i\cos\phi_i\sin\phi_i-\dot{\theta}_i\dot{\phi}_i\cos\theta_i\left(\cos^2\phi_i-\sin^2\phi_i\right)\right]\\
&\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad(i=1,2,\ldots,n+1)
\end{aligned}
\tag{4.61}
$$

Then, the addition of rotational kinetic energy to the left hand side of the Lagrange's Equations in terms of the heading angle will be:

$$
\begin{aligned}
&\frac{d}{dt}\left(\frac{\partial T_{rot.}}{\partial \dot{\psi}_i}\right)-\frac{\partial T_{rot.}}{\partial \psi_i}=I_{x_i}\left[\ddot{\psi}_i \sin^2\theta_i+2\dot{\psi}_i\dot{\theta}_i\cos\theta_i\sin\theta_i-\ddot{\phi}_i\sin\theta_i-\dot{\phi}_i\dot{\theta}_i\cos\theta_i\right]+\\
&I_{y_i}\left[\ddot{\psi}_i\cos^2\theta_i\sin^2\phi_i-2\dot{\psi}_i\dot{\theta}_i\sin\theta_i\cos\theta_i\sin^2\phi_i+2\dot{\psi}_i\dot{\phi}_i\cos^2\theta_i\sin\phi_i\cos\phi_i+\right.\\
&\left.\ddot{\theta}_i\cos\theta_i\sin\phi_i\cos\phi_i-\dot{\theta}_i^2\sin\theta_i\sin\phi_i\cos\phi_i+\dot{\theta}_i\dot{\phi}_i\cos\theta_i\left(\cos^2\phi_i-\sin^2\phi_i\right)\right]+\\
&I_{z_i}\left[\ddot{\psi}_i\cos^2\theta_i\cos^2\phi_i-2\dot{\psi}_i\dot{\theta}_i\cos\theta_i\sin\theta_i\cos^2\phi_i-2\dot{\psi}_i\dot{\phi}_i\cos^2\theta_i\cos\phi_i\sin\phi_i-\right.\\
&\left.\ddot{\theta}_i\cos\theta_i\cos\phi_i\sin\phi_i+\dot{\theta}_i^2\sin\theta_i\cos\phi_i\sin\phi_i-\dot{\theta}_i\dot{\phi}_i\cos\theta_i\left(\cos^2\phi_i-\sin^2\phi_i\right)\right]\\
&\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad(i=1,2,\ldots,n+1)
\end{aligned}
\tag{4.62}
$$

The derivation of the Equations for the pitch angle is done by taking derivatives of the kinetic energy with respect to the pitch angle and its rate as shown below:

$$
\begin{aligned}
&\frac{\partial T_{rot.}}{\partial \theta_i}=I_{x_i}\left(\dot{\psi}_i^2\sin\theta_i\cos\theta_i-\dot{\phi}_i\dot{\psi}_i\cos\theta_i\right)+\\
&I_{y_i}\left(-\dot{\psi}_i^2\cos\theta_i\sin\theta_i\sin^2\phi_i-\dot{\psi}_i\dot{\theta}_i\sin\theta_i\sin\phi_i\cos\phi_i\right)+\\
&I_{z_i}\left(-\dot{\psi}_i^2\cos\theta_i\sin\theta_i\cos^2\phi_i+\dot{\psi}_i\dot{\theta}_i\sin\theta_i\cos\phi_i\sin\phi_i\right)\qquad(i=1,2,\ldots,n+1)
\end{aligned}
\tag{4.63}
$$

$$\begin{aligned}\frac{\partial T_{rot.}}{\partial \dot{\theta}_i} =& I_{y_i}\left(\dot{\theta}_i \cos^2\phi_i + \dot{\psi}_i \cos\theta_i \sin\phi_i \cos\phi_i\right)+\\ & I_{z_i}\left(\dot{\theta}_i \sin^2\phi_i - \dot{\psi}_i \cos\theta_i \cos\phi_i \sin\phi_i\right)\Big\} \qquad (i=1,2,\ldots,n+1)\end{aligned} \tag{4.64}$$

And therefore the time derivative of Equation (4.64) could be expressed as:

$$\begin{aligned}\frac{d}{dt}\left(\frac{\partial T_{rot.}}{\partial \dot{\theta}_i}\right) =& I_{y_i}\Big[\ddot{\theta}_i \cos^2\phi_i - 2\dot{\theta}_i\dot{\phi}_i \cos\phi_i \sin\phi_i + \ddot{\psi}_i \cos\theta_i \sin\phi_i \cos\phi_i -\\ & \dot{\psi}_i\dot{\theta}_i \sin\theta_i \sin\phi_i \cos\phi_i + \dot{\psi}_i\dot{\phi}_i \cos\theta_i\left(\cos^2\phi_i - \sin^2\phi_i\right)\Big]+\\ & I_{z_i}\Big[\ddot{\theta}_i \sin^2\phi_j + 2\dot{\theta}_i\dot{\phi}_i \sin\phi_i \cos\phi_i - \ddot{\psi}_i \cos\theta_i \cos\phi_i \sin\phi_i +\\ & \dot{\psi}_i\dot{\theta}_i \sin\theta_i \cos\phi_i \sin\phi_i - \dot{\psi}_i\dot{\phi}_i \cos\theta_i\left(\cos^2\phi_i - \sin^2\phi_i\right)\Big] \qquad (i=1,2,\ldots,n+1)\end{aligned} \tag{4.65}$$

The left hand side of the Lagrange’s Equations for the UAS rotational kinetic energy in terms of the pitch angle is now written in the following form:

$$\begin{aligned}\frac{d}{dt}\left(\frac{\partial T_{rot.}}{\partial \dot{\theta}_i}\right) - \frac{\partial T_{rot.}}{\partial \theta_i} =& I_{x_i}\Big[-\dot{\psi}_i^2 \sin\theta_i \cos\theta_i + \dot{\phi}_i\dot{\psi}_i \cos\theta_i\Big]+\\ & I_{y_i}\Big[\ddot{\theta}_i \cos^2\phi_i - 2\dot{\theta}_i\dot{\phi}_i \cos\phi_i \sin\phi_i + \ddot{\psi}_i \cos\theta_i \sin\phi_i \cos\phi_i +\\ & +\dot{\psi}_i\dot{\phi}_i \cos\theta_i\left(\cos^2\phi_i - \sin^2\phi_i\right) + \dot{\psi}_i^2 \cos\theta_i \sin\theta_i \sin^2\phi_i\Big]+\\ & I_{z_i}\Big[\ddot{\theta}_i \sin^2\phi_j + 2\dot{\theta}_i\dot{\phi}_i \sin\phi_i \cos\phi_i - \ddot{\psi}_i \cos\theta_i \cos\phi_i \sin\phi_i -\\ & \dot{\psi}_i\dot{\phi}_i \cos\theta_i\left(\cos^2\phi_i - \sin^2\phi_i\right) + \dot{\psi}_i^2 \cos\theta_i \sin\theta_i \cos^2\phi_i\Big] \qquad (i=1,2,\ldots,n+1)\end{aligned} \tag{4.66}$$

Continuing to the Lagrange’s Equations based on the rotational kinetic energy of the UAS for the roll angle and its rate, ϕ and $\dot{\phi}$, respectively, one can write:

$$\begin{aligned}\frac{\partial T_{rot.}}{\partial \phi_i} =& I_{y_i}\left[\dot{\psi}_i^2 \cos^2\theta_i \sin\phi_i \cos\phi_i - \dot{\theta}_i^2 \cos\phi_i \sin\phi_i + \right.\\ & \left.\dot{\psi}_i \dot{\theta}_i \cos\theta_i \left(\cos^2\phi_i - \sin^2\phi_i\right)\right] + \\ & I_{z_i}\left[-\dot{\psi}_i^2 \cos^2\theta_i \cos\phi_i \sin\phi_i + \dot{\theta}_i^2 \sin\phi_i \cos\phi_i - \right.\\ & \left.\dot{\psi}_i \dot{\theta}_i \cos\theta_i \left(\cos^2\phi_i - \sin^2\phi_i\right)\right] \qquad (i = 1, 2, \ldots, n+1)\end{aligned} \tag{4.67}$$

$$\frac{\partial T_{rot.}}{\partial \dot{\phi}_i} = I_{x_j}\left(\dot{\phi}_i - \dot{\psi}_i \sin\theta_i\right) \qquad (i = 1, 2, \ldots, n+1) \tag{4.68}$$

Derivative of Equation (4.68) is taken with respect to time to get the first term of Lagrange’s Equation as:

$$\frac{d}{dt}\left(\frac{\partial T_{rot.}}{\partial \dot{\phi}_i}\right) = I_{x_j}\left[\ddot{\phi}_i - \ddot{\psi}_i \sin\theta_i - \dot{\psi}_i \dot{\theta}_i \cos\theta_i\right] \qquad (i = 1, 2, \ldots, n+1) \tag{4.69}$$

Therefore, the left hand side of the Lagrange’s Equations for all segments in terms of the roll angle based on the UAS rotational kinetic energy can be stated as follows:

$$\begin{aligned}\frac{d}{dt}\left(\frac{\partial T_{rot.}}{\partial \dot{\phi}_i}\right) - \frac{\partial T_{rot.}}{\partial \phi_i} =& I_{x_i}\left[\ddot{\phi}_i - \ddot{\psi}_i \sin\theta_i - \dot{\psi}_i \dot{\theta}_i \cos\theta_i\right] + \\ & I_{y_i}\left[-\dot{\psi}_i^2 \cos^2\theta_i \sin\phi_i \cos\phi_i + \dot{\theta}_i^2 \cos\phi_i \sin\phi_i - \right.\\ & \left.\dot{\psi}_i \dot{\theta}_i \cos\theta_i \left(\cos^2\phi_i - \sin^2\phi_i\right)\right] + \\ & I_{z_i}\left[\dot{\psi}_i^2 \cos^2\theta_i \cos\phi_i \sin\phi_i - \dot{\theta}_i^2 \sin\phi_i \cos\phi_i + \right.\\ & \left.\dot{\psi}_i \dot{\theta}_i \cos\theta_i \left(\cos^2\phi_i - \sin^2\phi_i\right)\right] \qquad (i = 1, 2, \ldots, n+1)\end{aligned} \tag{4.70}$$

These equations derived based on rotational kinetic energy can now be combined in a matrix form. The equations of motion based on rotational kinetic energy can then be added to the Equation (4.36) which has been derive based on translational kinetic energy of the system. It should be noted that Equation (4.36) consists of *3n* equations which should be extended to *3(n+1)* equations after adding the UAS segment.

Arranging Equations (4.62), (4.66), (4.70) to a matrix form, one can write them in the following form which represents the additional terms to be added to the left hand side of Equation (4.37) to reflect the dynamics of the UAS into the equations of motion:

$$\begin{bmatrix} \mathbf{A}_{11_{\text{rot.}}} & \mathbf{A}_{12_{\text{rot.}}} & \mathbf{A}_{13_{\text{rot.}}} \\ \mathbf{A}_{21_{\text{rot.}}} & \mathbf{A}_{22_{\text{rot.}}} & \mathbf{A}_{23_{\text{rot.}}} \\ \mathbf{A}_{31_{\text{rot.}}} & \mathbf{A}_{32_{\text{rot.}}} & \mathbf{A}_{33_{\text{rot.}}} \end{bmatrix} \begin{bmatrix} \ddot{\boldsymbol{\psi}} \\ \ddot{\boldsymbol{\theta}} \\ \ddot{\boldsymbol{\phi}} \end{bmatrix} + \begin{bmatrix} \mathbf{B}_{11_{\text{rot.}}} & \mathbf{B}_{12_{\text{rot.}}} & \mathbf{B}_{13_{\text{rot.}}} \\ \mathbf{B}_{21_{\text{rot.}}} & \mathbf{B}_{22_{\text{rot.}}} & \mathbf{B}_{23_{\text{rot.}}} \\ \mathbf{B}_{31_{\text{rot.}}} & \mathbf{B}_{32_{\text{rot.}}} & \mathbf{B}_{33_{\text{rot.}}} \end{bmatrix} \begin{bmatrix} \dot{\boldsymbol{\psi}}^2 \\ \dot{\boldsymbol{\theta}}^2 \\ \dot{\boldsymbol{\phi}}^2 \end{bmatrix} + \begin{bmatrix} \mathbf{C}_{11_{\text{rot.}}} & \mathbf{C}_{12_{\text{rot.}}} & \mathbf{C}_{13_{\text{rot.}}} \\ \mathbf{C}_{21_{\text{rot.}}} & \mathbf{C}_{22_{\text{rot.}}} & \mathbf{C}_{23_{\text{rot.}}} \\ \mathbf{C}_{31_{\text{rot.}}} & \mathbf{C}_{32_{\text{rot.}}} & \mathbf{C}_{33_{\text{rot.}}} \end{bmatrix} \begin{bmatrix} \dot{\boldsymbol{\theta}}\dot{\boldsymbol{\psi}} \\ \dot{\boldsymbol{\phi}}\dot{\boldsymbol{\theta}} \\ \dot{\boldsymbol{\phi}}\dot{\boldsymbol{\psi}} \end{bmatrix} \qquad (\ 4.71\ )$$

where each sub matrix of $\mathbf{A}$ is a diagonal matrix with the elements stated below:

$$\begin{cases} \left[\mathbf{A}_{11_{\text{rot.}}}\right]_{i,i} = I_{x_i}\sin^2\theta_i + \left(I_{y_i}\sin^2\phi_i + I_{z_i}\cos^2\phi_i\right)\cos^2\theta_i \\ \left[\mathbf{A}_{12_{\text{rot.}}}\right]_{i,i} = \left[\mathbf{A}_{21_{\text{rot.}}}\right]_{i,i} = \left(I_{y_i} - I_{z_i}\right)\cos\theta_i\cos\phi_i\sin\phi_i \\ \left[\mathbf{A}_{13_{\text{rot.}}}\right]_{i,i} = \left[\mathbf{A}_{31_{\text{rot.}}}\right]_{i,i} = -I_{x_i}\sin\theta_i \\ \left[\mathbf{A}_{22_{\text{rot.}}}\right]_{i,i} = \left(I_{y_i}\cos^2\phi_i + I_{z_i}\sin^2\phi_i\right) \\ \left[\mathbf{A}_{23_{\text{rot.}}}\right]_{i,i} = \left[\mathbf{A}_{32_{\text{rot.}}}\right]_{i,i} = 0 \\ \left[\mathbf{A}_{33_{\text{rot.}}}\right]_{i,i} = I_{x_i} \end{cases} \qquad (i = 1, 2, \ldots, n+1) \qquad (\ 4.72\ )$$

and sub-matrices of matrix $\mathbf{B}$ are:

$$\begin{cases} \left[\mathbf{B}_{\mathbf{11}_{\mathbf{rot.}}}\right]_{i,i} = 0 \\ \left[\mathbf{B}_{\mathbf{12}_{\mathbf{rot.}}}\right]_{i,i} = \left(I_{z_i} - I_{y_i}\right)\sin\theta_i \cos\phi_i \sin\phi_i \\ \left[\mathbf{B}_{\mathbf{13}_{\mathbf{rot.}}}\right]_{i,i} = 0 \\ \left[\mathbf{B}_{\mathbf{21}_{\mathbf{rot.}}}\right]_{i,i} = \left(-I_{x_i} + I_{y_i}\sin^2\phi_i + I_{z_i}\cos^2\phi_i\right)\sin\theta_i\cos\theta_i \\ \left[\mathbf{B}_{\mathbf{22}_{\mathbf{rot.}}}\right]_{i,i} = 0 \qquad\qquad\qquad\qquad (i = 1, 2, \ldots, n+1) \\ \left[\mathbf{B}_{\mathbf{23}_{\mathbf{rot.}}}\right]_{i,i} = 0 \\ \left[\mathbf{B}_{\mathbf{31}_{\mathbf{rot.}}}\right]_{i,i} = \left(I_{z_i} - I_{y_i}\right)\cos^2\theta_i \cos\phi_i \sin\phi_i \\ \left[\mathbf{B}_{\mathbf{32}_{\mathbf{rot.}}}\right]_{i,i} = \left(I_{y_i} - I_{z_i}\right)\cos\phi_i \sin\phi_i \\ \left[\mathbf{B}_{33_{rot.}}\right]_{i,i} = 0 \end{cases} \tag{4.73}$$

and finally, matrix $\mathbf{C}$ consists of two sub-matrices:

$$\begin{cases} \left[\mathbf{C}_{\mathbf{11}_{\mathbf{rot.}}}\right]_{i,i} = 2\cos\theta_i \sin\theta_i \left(I_{x_i} - I_{y_i}\sin^2\phi_i - I_{z_i}\cos^2\phi_i\right) \\ \left[\mathbf{C}_{\mathbf{12}_{\mathbf{rot.}}}\right]_{i,i} = \cos\theta_i \left(-I_{x_i} + \left(I_{y_i} - I_{z_i}\right)\left(\cos^2\phi_i - \sin^2\phi_i\right)\right) \\ \left[\mathbf{C}_{\mathbf{13}_{\mathbf{rot.}}}\right]_{i,i} = 2\cos^2\theta_i \sin\phi_i \cos\phi_i \left(I_{y_i} - I_{z_i}\right) \\ \left[\mathbf{C}_{\mathbf{21}_{\mathbf{rot.}}}\right]_{i,i} = 0 \\ \left[\mathbf{C}_{\mathbf{22}_{\mathbf{rot.}}}\right]_{i,i} = 2\sin\phi_i \cos\phi_i \left(I_{z_i} - I_{y_i}\right) \qquad\qquad (i = 1, 2, \ldots, n+1) \\ \left[\mathbf{C}_{\mathbf{23}_{\mathbf{rot.}}}\right]_{i,i} = \cos\theta_i \left(I_{x_i} + \left(I_{y_i} - I_{z_i}\right)\left(\cos^2\phi_i - \sin^2\phi_i\right)\right) \\ \left[\mathbf{C}_{\mathbf{31}_{\mathbf{rot.}}}\right]_{i,i} = \cos\theta_i \left(-I_{x_i} + \left(I_{z_i} - I_{y_i}\right)\left(\cos^2\phi_i - \sin^2\phi_i\right)\right) \\ \left[\mathbf{C}_{\mathbf{32}_{\mathbf{rot.}}}\right]_{i,i} = 0 \\ \left[\mathbf{C}_{\mathbf{33}_{\mathbf{rot.}}}\right]_{i,i} = 0 \end{cases} \tag{4.74}$$

It is very important to note that Equation (4.71) includes the rotational kinetic energy for all cable segments. That is it can implement the effect of inertias of cable segments into the equations. It is also worthwhile to mention that unless the inertias of the segments are included as thin rods, matrix $\mathbf{A}$ becomes irreversible causing the system to be singular and not solvable. Therefore, the rotational kinetic energy for all segments is added in all directions. This means that the model has migrated from a lumped mass model to a thin rod model at this point. Although the mass is assumed to be located at the end of the segment, the mass moments of inertia are calculated and implemented in matrix $\mathbf{A}$ for the purpose of solving the problems.

Addition of Equation (4.71) to the left hand side of Equation (4.37) and adding the generalized force component for the UAS, will finalize the derivation of equations of motion for the UAS-cable system. This set of *3(n+1)* equations can be expressed as follows and will be solved through a similar expanded algorithm developed in Chapter 3.

$$\begin{bmatrix} \mathbf{A}'_{11} & \mathbf{A}'_{12} & \mathbf{A}'_{13} \\ \mathbf{A}'_{21} & \mathbf{A}'_{22} & \mathbf{A}'_{23} \\ \mathbf{A}'_{31} & \mathbf{A}'_{32} & \mathbf{A}'_{33} \end{bmatrix} \begin{bmatrix} \ddot{\boldsymbol{\psi}} \\ \ddot{\boldsymbol{\theta}} \\ \ddot{\boldsymbol{\phi}} \end{bmatrix} + \begin{bmatrix} \mathbf{B}'_{11} & \mathbf{B}'_{12} & \mathbf{B}'_{13} \\ \mathbf{B}'_{21} & \mathbf{B}'_{22} & \mathbf{B}'_{23} \\ \mathbf{B}'_{31} & \mathbf{B}'_{32} & \mathbf{B}'_{33} \end{bmatrix} diag \begin{Bmatrix} \dot{\boldsymbol{\psi}} \\ \dot{\boldsymbol{\theta}} \\ \dot{\boldsymbol{\phi}} \end{Bmatrix} \begin{bmatrix} \dot{\boldsymbol{\psi}} \\ \dot{\boldsymbol{\theta}} \\ \dot{\boldsymbol{\phi}} \end{bmatrix} + \begin{bmatrix} \mathbf{C}'_{11} & \mathbf{C}'_{12} & \mathbf{C}'_{13} \\ \mathbf{C}'_{21} & \mathbf{C}'_{22} & \mathbf{C}'_{23} \\ \mathbf{C}'_{31} & \mathbf{C}'_{32} & \mathbf{C}'_{33} \end{bmatrix} \begin{bmatrix} \dot{\boldsymbol{\theta}}\dot{\boldsymbol{\psi}} \\ \dot{\boldsymbol{\phi}}\dot{\boldsymbol{\theta}} \\ \dot{\boldsymbol{\psi}}\dot{\boldsymbol{\phi}} \end{bmatrix} = \begin{bmatrix} \mathbf{Q}_{\psi} \\ \mathbf{Q}_{\theta} \\ \mathbf{Q}_{\phi} \end{bmatrix} \qquad (4.75)$$

where any primed matrix conveys the addition of rotational and translational coefficient matrices added together.

Since the cable model used in this work is the lumped-mass model, there is no inertia involved in the equations of motion for the cable segments. Thus the only inertia added to the equations is the UAS inertia. As seen before, Equation (4.37) in fact consists of *2(n)* equations, which after addition of the UAS segments will be augmented to reflect *3(n+1)* equations. Since there is no inertia involved in the cable model, the only term added to the translational equations is at the *n+1*th row of submatrices of matrices $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{C}$. This causes the submatrices of $\mathbf{C}$ to remain zero except at the *n+1*th element which is a coefficient of the UAS's bank angle. This will be observed in the results section.

4.6. Numerical Method for Solving the System's Equation of Motion

The equations derived in this chapter are similar to the equations derived in Chapter 3 in terms of mathematical order. That is the equation set is still a coupled set of several highly non-linear second-order ordinary differential equations with a larger variable number. In Chapter 3, the main variables were a set of *n+1* generalized coordinates each of which represented the attitude angle of each cable segment. In this chapter, there are mainly three sets of attitude angles which consist of heading, pitch and roll angles. Each of these attitude angles consist of *n+1* angles in compliance with the number of segments used for the model in addition to one cable segment.

For the purpose of solving this set of equations, the same tool in MATLAB™ software package is used which solves the non-linear ODE's using the Runge-Kutta numerical integration method.

After adding the rotational terms and implementing the UAS dynamic equations to Equation (4.37), it will have *3(n+1)* equations in which the unknowns are the Euler Angles of the segments as well as their first and second time derivatives, i.e.,ψ ,θ ,ϕ, $\dot{\psi}$,$\dot{\theta}$,$\dot{\phi}$, $\ddot{\psi}$,$\ddot{\theta}$,$\ddot{\phi}$ respectively. In order to use the "ode45" solver in MATLAB™, the equation set is reduced by order and enlarged by equation size using the following procedure. This will convert the *3(n+1)* second-order ODE's to *6(n+1)* first-order ODE's which can be solved using proper initial conditions.

4.6.1. Order Reduction of Differential Equations

The general form of the system's equation for each segment is presented below:

$$
\begin{cases}
A_1(\psi,\dot{\psi},\theta,\dot{\theta},\phi,\dot{\phi})\dfrac{d^2\theta}{dt^2}+A_2(\psi,\dot{\psi},\theta,\dot{\theta},\phi,\dot{\phi})\dfrac{d^2\phi}{dt^2}+A_3(\psi,\dot{\psi},\theta,\dot{\theta},\phi,\dot{\phi})\dfrac{d^2\psi}{dt^2}+ \\
B_1(\psi,\dot{\psi},\theta,\dot{\theta},\phi,\dot{\phi})\left(\dfrac{d\theta}{dt}\right)^2+B_2(\psi,\dot{\psi},\theta,\dot{\theta},\phi,\dot{\phi})\left(\dfrac{d\phi}{dt}\right)^2+B_3(\psi,\dot{\psi},\theta,\dot{\theta},\phi,\dot{\phi})\left(\dfrac{d\psi}{dt}\right)^2+ \\
C_1(\psi,\dot{\psi},\theta,\dot{\theta},\phi,\dot{\phi})\left(\dfrac{d\theta}{dt}\dfrac{d\psi}{dt}\right)+C_2(\psi,\dot{\psi},\theta,\dot{\theta},\phi,\dot{\phi})\left(\dfrac{d\phi}{dt}\dfrac{d\theta}{dt}\right)+C_3(\psi,\dot{\psi},\theta,\dot{\theta},\phi,\dot{\phi})\left(\dfrac{d\psi}{dt}\dfrac{d\theta}{dt}\right) \\
\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad = f(t,\psi,\dot{\psi},\theta,\dot{\theta},\phi,\dot{\phi})
\end{cases}
\tag{4.76}
$$

$$\psi(0)=\psi_0,\dot{\psi}(0)=\dot{\psi}_0,\theta(0)=\theta_0,\dot{\theta}(0)=\dot{\theta}_0,\phi(0)=\phi_0,\dot{\phi}(0)=\dot{\phi}_0$$

By assuming the following vectors as attitude vectors, one can change the order of the system as follows.

The vector $\vec{\theta}$ is called the angular position vector and is represented by:

$$\vec{\psi}=\begin{pmatrix}\psi_1\\ \psi_2\\ \vdots\\ \psi_{n+1}\end{pmatrix}_{(n+1)\times 1},\vec{\theta}=\begin{pmatrix}\theta_1\\ \theta_2\\ \vdots\\ \theta_{n+1}\end{pmatrix}_{(n+1)\times 1},\vec{\phi}=\begin{pmatrix}\phi_1\\ \phi_2\\ \vdots\\ \phi_{n+1}\end{pmatrix}_{(n+1)\times 1} \tag{4.77}$$

The vector $\vec{\omega}$ is called the Euler Angle rate vector and is represented by:

$$\vec{\dot{\psi}}=\begin{pmatrix}\dot{\psi}_1\\ \dot{\psi}_2\\ \vdots\\ \dot{\psi}_{n+1}\end{pmatrix}_{(n+1)\times 1},\vec{\dot{\theta}}=\begin{pmatrix}\dot{\theta}_1\\ \dot{\theta}_2\\ \vdots\\ \dot{\theta}_{n+1}\end{pmatrix}_{(n+1)\times 1},\vec{\dot{\phi}}=\begin{pmatrix}\dot{\phi}_1\\ \dot{\phi}_2\\ \vdots\\ \dot{\phi}_{n+1}\end{pmatrix}_{(n+1)\times 1} \tag{4.78}$$

The vector $\vec{\alpha}$ is the vector consisting of second time-derivative of Euler Angles and is represented by:

$$\ddot{\vec{\psi}} = \begin{pmatrix} \ddot{\psi}_1 \\ \ddot{\psi}_2 \\ \vdots \\ \ddot{\psi}_{n+1} \end{pmatrix}_{(n+1)\times 1} , \ddot{\vec{\theta}} = \begin{pmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \\ \vdots \\ \ddot{\theta}_{n+1} \end{pmatrix}_{(n+1)\times 1} , \ddot{\vec{\phi}} = \begin{pmatrix} \ddot{\phi}_1 \\ \ddot{\phi}_2 \\ \vdots \\ \ddot{\phi}_{n+1} \end{pmatrix}_{(n+1)\times 1} \qquad (4.79)$$

Now, define a vector $\vec{\zeta}$ consisting of vectors defined in Equations (4.77) and (4.78), named the angular state of the system as:

$$\vec{\zeta} = \begin{pmatrix} \dot{\vec{\eta}} \\ \vec{\eta} \end{pmatrix} = \begin{pmatrix} \dot{\vec{\psi}} \\ \dot{\vec{\theta}} \\ \dot{\vec{\phi}} \\ \vec{\psi} \\ \vec{\theta} \\ \vec{\phi} \end{pmatrix}_{6(n+1)\times 1} \qquad (4.80)$$

where $\vec{\eta}$ is defined in Equation (4.35).Then, the time-derivative of Equation (4.80) can be written as:

$$\dot{\vec{\zeta}} = \begin{pmatrix} \ddot{\vec{\eta}} \\ \dot{\vec{\eta}} \end{pmatrix} = \begin{pmatrix} \ddot{\vec{\psi}} \\ \ddot{\vec{\theta}} \\ \ddot{\vec{\phi}} \\ \dot{\vec{\psi}} \\ \dot{\vec{\theta}} \\ \dot{\vec{\phi}} \end{pmatrix}_{6(n+1)\times 1} \qquad (4.81)$$

Manipulating the system's equation of motion and substituting the variables by the system state vector, one can express the reduced order system of equations in the form of:

$$
\begin{aligned}
\dot{\vec{\zeta}}_{6(n+1)\times 1} = {} & \begin{bmatrix} -\mathbf{A}'^{-1}\mathbf{B}'diag\left\{\dot{\vec{\eta}}\right\} & \underline{0}_{3(n+1)\times 3(n+1)} \\ I_{3(n+1)\times 3(n+1)} & \underline{0}_{3(n+1)\times 3(n+1)} \end{bmatrix}_{6(n+1)\times 6(n+1)} \times \vec{\zeta}_{6(n+1)\times 1} \\
& + \begin{bmatrix} \mathbf{A}'^{-1}\vec{Q} \\ \underline{0}_{3(n+1)\times 1} \end{bmatrix} + \begin{bmatrix} -\mathbf{A}'^{-1}\mathbf{C}' & \underline{0}_{3(n+1)\times 3(n+1)} \\ \underline{0}_{3(n+1)\times 3(n+1)} & \underline{0}_{3(n+1)\times 3(n+1)} \end{bmatrix}_{6(n+1)\times 6(n+1)} \times diag\left\{ \begin{matrix} \dot{\boldsymbol{\theta}} \\ \dot{\boldsymbol{\phi}} \\ \dot{\boldsymbol{\psi}} \\ \underline{0}_{3(n+1)\times 1} \\ \underline{0}_{3(n+1)\times 1} \\ \underline{0}_{3(n+1)\times 1} \end{matrix} \right\} \begin{bmatrix} \dot{\boldsymbol{\psi}} \\ \dot{\boldsymbol{\theta}} \\ \dot{\boldsymbol{\phi}} \\ \underline{0}_{3(n+1)\times 1} \\ \underline{0}_{3(n+1)\times 1} \\ \underline{0}_{3(n+1)\times 1} \end{bmatrix}
\end{aligned}
\qquad (\ 4.82\ )
$$

where $\mathbf{A}'$, $\mathbf{B}'$ and $\mathbf{C}'$ are coefficient matrices of Equation (4.37) added to the rotational matrices of Equation (4.71) and $\vec{Q}$ is the vector of generalized forces defined as:

$$
\vec{Q} = \begin{bmatrix} \mathbf{Q}_{\psi} \\ \mathbf{Q}_{\theta} \\ \mathbf{Q}_{\phi} \end{bmatrix} \qquad (\ 4.83\ )
$$

Equation (4.82) is a set of *6n+6* first-order ODE’s which will be solved using the Runge-Kutta numerical integration algorithm. The algorithms and MATLAB™ programs are presented in Appendix A.

These equations are then solved numerically using appropriate initial conditions to simulate different scenarios. Using the accelerations obtained from the numerical solutions, the forces on each cable segment are calculated in order to estimate the cable tension at each node. This model is then verified for correctness in the following section.

4.7. Model Verification

The derivation of the Lagrange’s Equations for the 3D multibody discrete cable system is a detail-oriented task. Thus it is very likely to make mistakes while deriving or implementing the equations into the software. Therefore, it is deemed necessary to test the validity of the 3D model as well to ensure it is derived and implemented correctly. The next few sections present some simple tests ensuring the model's validity.

4.7.1. Simple Pendulum Test

Similar to Chapter 3, as an initial test, the model is tested to ensure it is acting reasonably with only one segment; that is when it is modeling an ideal simple pendulum with a mass located at the end, and no friction present. In addition, to test this model against a simple pendulum, it is important to set the inertias to zero to reflect the behavior of the pendulum with a pointed mass and a massless rod. The comparison between the theoretical oscillation periods and simulated periods are presented in Figure 4.3.

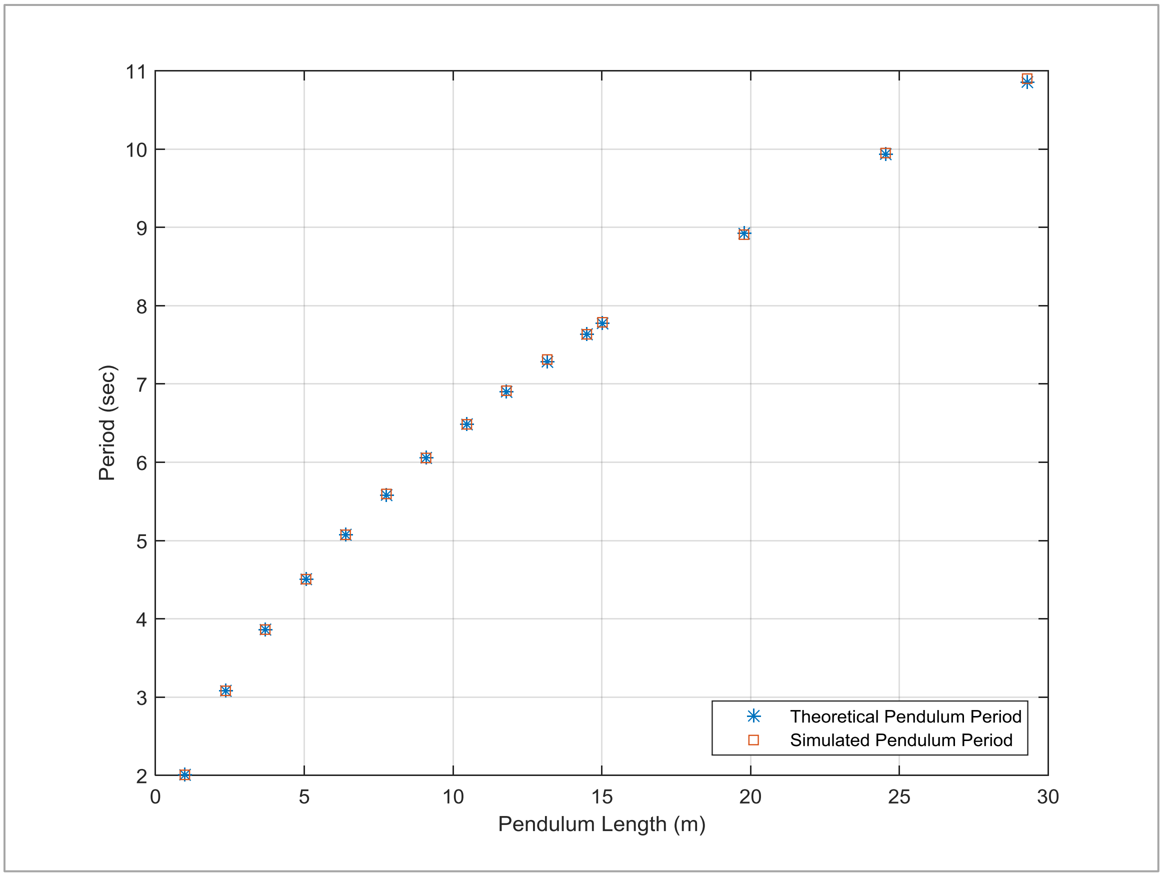

Figure 4.3. Simple Pendulum Periods, Comparison of Simulated and Theoretical Results.

According to the results of a few tests, one can observe that the model is capable of predicting the dynamics of an ideal pendulum with a high accuracy. As you can see in Figure 4.3, in these tests, different set of pendulum lengths are chosen in comparison to Chapter 3. It should be noted that these results simulate the pendulum oscillating in the YZ plane where heading angle, ψ_i is 90 degrees for the segment. Other sets of tests have been run with the pendulum oscillating in other vertical planes– that is with other values of heading angles–to ensure the model's accuracy.

4.7.2. Generalization of the Model to a Finite-Segment Cable

To confirm that the model simulates a finite segment cable model properly and correctly, some tests similar to Section 3.7.2 are run and compared in this section. It should be noted that since this section is merely studying the cable's behavior, the UAS's properties is neglected in these set of simulations. Four different combinations of thrust and pitch are tested for each of the lumped mass model as well as the lumped mass model where the segment inertias are added to the cable model. The results are presented in Figure 4.4. It can be seen in these plots that the cable shape generated using the 3D model agrees to the ones generated using the 2D models.

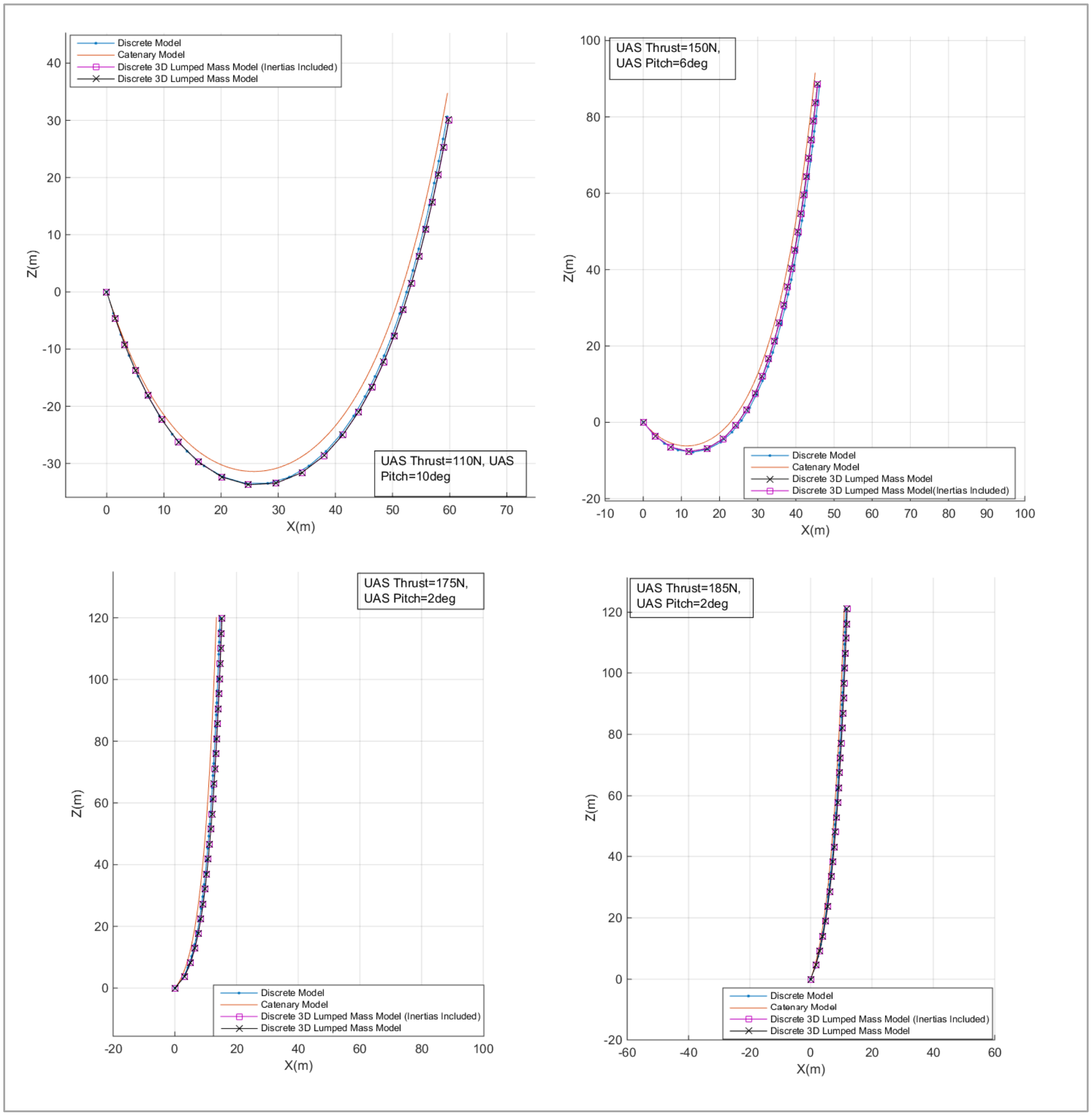

Figure 4.4. Comparison of static cable shapes; catenary model vs. 2D and 3D discrete models

4.8. Results and Discussions

Results from the modeling and simulation of the 3D cable-body system in a few case studies are demonstrated in the following sections. Table 4.1 shows the UAS properties as well as the simulation parameters used in these set of tests. Again the normal and frictional drag coefficients are chosen to be 1.5 and 0.02 respectively to follow the guidelines from Hoerner [32]. Similar to the 2D case, the UAS thrust is modeled as a vectored thrust always perpendicular to the UAS horizontal axis; that is the thrust T is always in the x direction of the UAS virtual segment. The UAS thrust is assumed to be held exactly constant in magnitude unless stated otherwise. In general, the thrust components in the global coordinate is defined as:

$$\begin{pmatrix} T_X \\ T_Y \\ T_Z \end{pmatrix} = R^{-1}{}_{\phi_{n+1}\theta_{n+1}\psi_{n+1}} \begin{pmatrix} T \\ 0 \\ 0 \end{pmatrix} \tag{4.84}$$

Then the thrust vector can be written as:

$$\begin{pmatrix} T_X \\ T_Y \\ T_Z \end{pmatrix} = \begin{pmatrix} T\cos\left(\psi_{n+1}\right)\cos\left(\theta_{n+1}\right) \\ T\sin\left(\psi_{n+1}\right)\cos\left(\theta_{n+1}\right) \\ -T\sin\left(\theta_{n+1}\right) \end{pmatrix} \tag{4.85}$$

It is worthwhile to mention that in this configuration of Euler Angle's, positive values of the pitch angles θ_i represent downward pitch since the angles follow right-hand rule.

Table 4.1. UAS Parameters

Simulation Parameter	Value	Units
UAS Weight	10.56	[kg]
UAS-Cable Connection Link Length	0.1	[m]
Cable Effective Shear Modulus	56	[MPa]
UAS Moment of Inertia (at connection point) (I'$_{xx}$)	0.94	[kg.m^2]
UAS Moment of Inertia (at connection point) (I'$_{yy}$)	1.33	[kg.m^2]
UAS Moment of Inertia (at connection point) (I'$_{zz}$)	1.33	[kg.m^2]
UAS Frame Rod Length	1.01	[m]
UAS Frame Rod Diameter	0.025	[m]
UAS Vertical Thrust	283.59	[N]

A very important fact about the 3D modelling of any rigid body using Euler Angles is the concept of gimbal lock. Using Type I Euler Angles, this condition occurs “when two of the Euler angle rates represent rotations about the same axis in space” [33]; that is when $\theta = \pm\frac{\pi}{2}$. This causes a singularity in the problem making the angles ψ and ϕ not defined. It is important to be aware of this limitation in the model and avoid situations that this phenomenon might happen.

4.8.1. Effect of Wind on Lateral Motion

In this section the UAS and cable’s dynamic behavior is investigate when the UAS is operating in a windy condition. According to Chapter 3, and due to the inherent limitation of the 2D model, the UAS and cable were always moving in the XZ plane. However, in reality this is not the case and the UAS will have lateral motion, called sway, hereafter, in addition to drift or offset motion in the X direction.

The UAS is considered to have reached steady state under the effect of a 4 m/s wind in the X direction when a wind with the velocity of 8 m/s starts blowing on the system at time *t=1* seconds in the lateral direction for 10 seconds. The UAS is then allowed to reach steady state condition and the position as well as cable tensions are plotted. Table 4.2 shows the wind parameters used for simulation.

Table 4.2. Wind Velocity Vector

Parameter	Label	Value (m/s)	Application Time (sec)
Wind Velocity in X direction	U	4	0:00-120:00
Wind Velocity in Y direction	V	8	1:00-11:00
Wind Velocity in Z direction	W	0	N/A

Figure 4.5 shows how the position of UAS center of mass changes in all horizontal and vertical planes. One can see the inherent stability of the UAS in the sense that it can achieves the same position after some time. The wind stops blowing at *t=11s* and by *t=80s* the UAS has achieved the initial stable position.

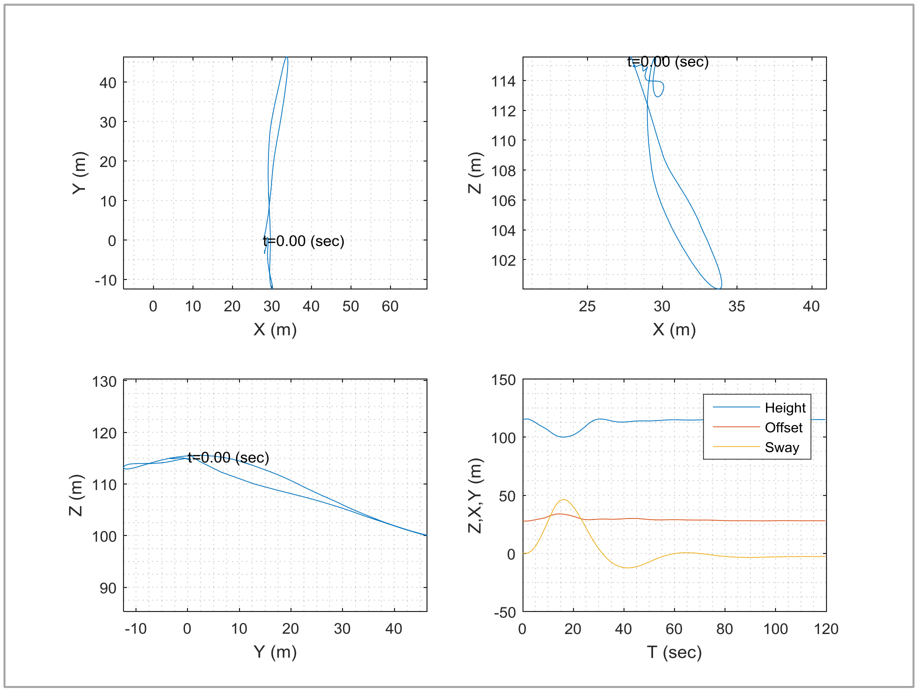

Figure 4.5. UAS position in different planes, Lateral Wind Effect

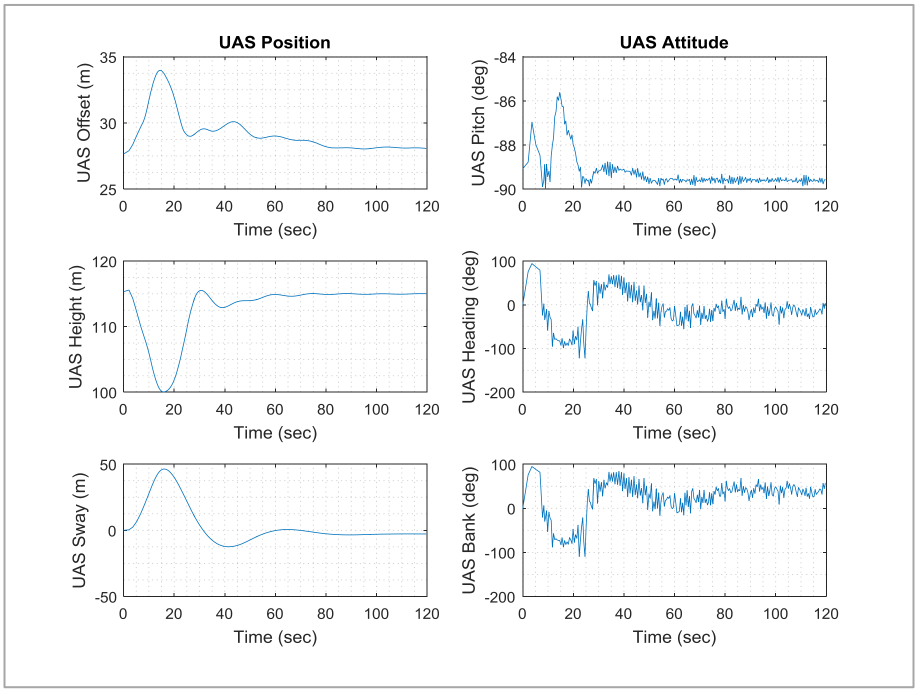

Figure 4.6. UAS Position and Attitude, Lateral Wind Effect

Figure 4.6 presents the UAS position as well as its attitude plotted against time. Generally the UAS holds a relatively stable heading and pitch and it can settle to that value after the lateral wind stops blowing. Also it can be inferred that with this lateral wind blowing the UAS height drops about 15 meters in about 17 seconds. This is something to consider seriously when designing the control for the flying aerial system. Also, this wind pulls the UAS sideways by 50 meters which is a significant amount. Again this is something to be aware of when specifying the operating point for the UAS.

4.8.2. Variable Wind Model

To better model the wind effect on stability of the cable-body system, a more realistic wind model can be introduced in order to study the effect of wind speed and direction variations on the system's static and dynamic stability. As an example, the effect of a certain wind variation on the system's states is simulated here.

Consider that vector $\vec{V}_w$ represents the wind velocity vector parallel to the ground; i.e. the XY plane as a function of time. For simplicity, assume there is no cross-flow wind present in the vertical direction. Now, the variables U and V which represent the wind velocity in X and Y direction respectively are also functions of time. By employing a simple wind model used to model an aerostat in Fourie's work [42], the wind velocity vector is defined as:

$$\vec{V}_w(t) = \begin{pmatrix} U(t) \\ V(t) \end{pmatrix} = \begin{pmatrix} \left|\vec{V}_w(t)\right| \cos\beta(t) \\ \left|\vec{V}_w(t)\right| \sin\beta(t) \end{pmatrix} \tag{4.86}$$

where the magnitude which varies with a frequency of 0.5 *Hz,* mean wind speed of 3 *m/s* and a variance of 1 *m/s*. The wind heading angle β also has a mean value of 45 degrees and a variance of 15 degrees which fluctuates every 10 seconds. The wind parameters are then defined as shown in **Error! Reference source not found.**. Figure 4.7 shows the wind speed and heading variations during a 100 seconds of simulation. The UAS attitude and position due to application of this fluctuating wind velocity are presented in Figure 4.9.

Table 4.3. Wind Parameters

Parameter	Label	Value	Units
Wind Velocity Magnitude Mean Value	μ_V	3	[m/s]
Wind Velocity Variance	σ_V	2	[m/s]
Wind Velocity Frequency	f_V	0.5	[Hz]
Wind Heading Angle Mean Value	μ_β	0	[deg]
Wind Heading Angle Variance	σ_β	10	[deg]
Wind Heading Angle Frequency	f_β	0.1	[Hz]

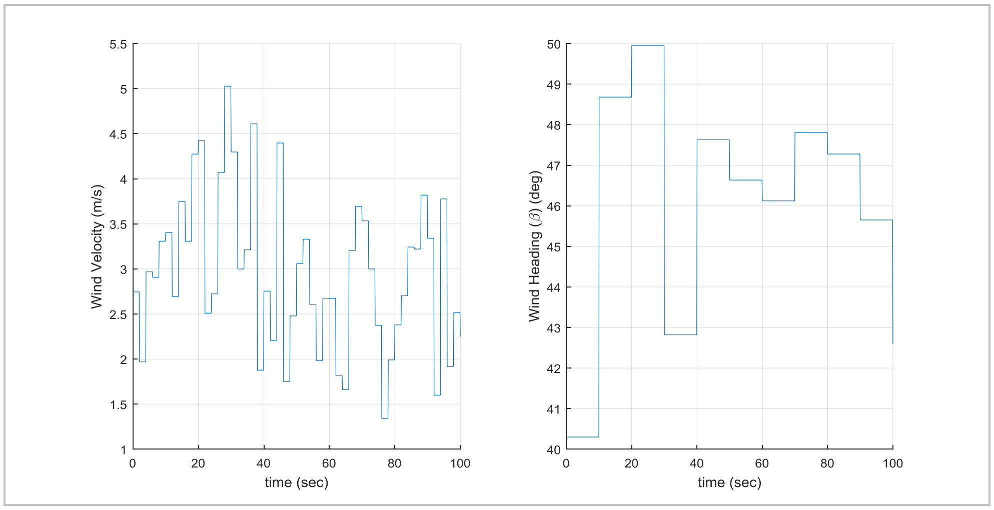

Figure 4.7. Wind Variations; left: speed variations; right: heading variations

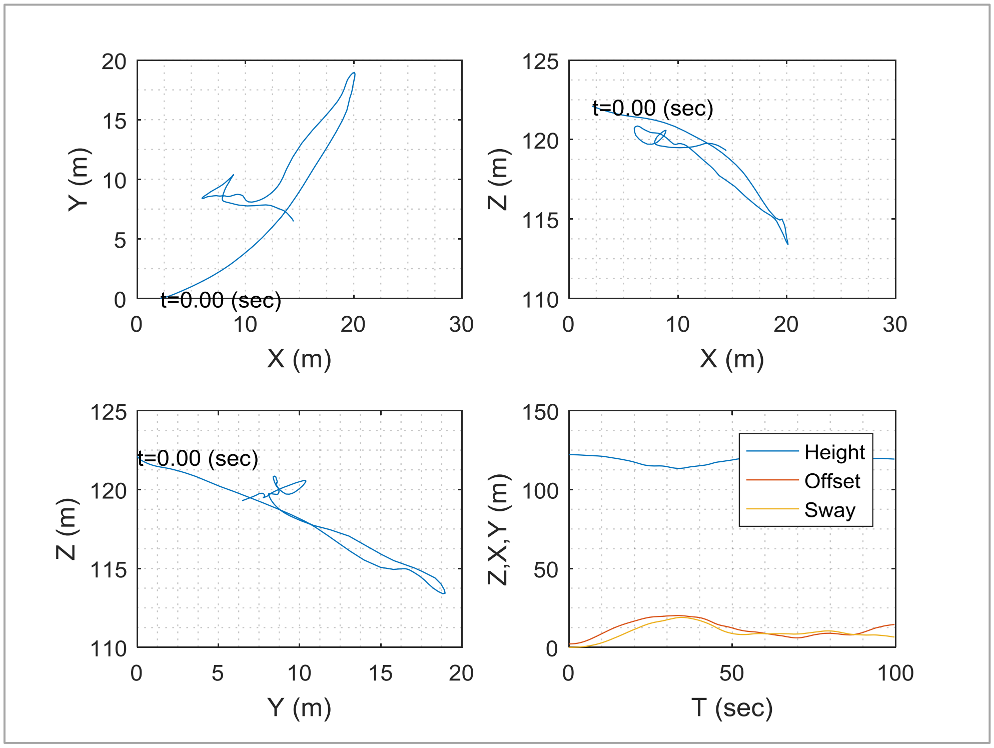

Figure 4.8. UAS position in different planes, Lateral Wind Effect

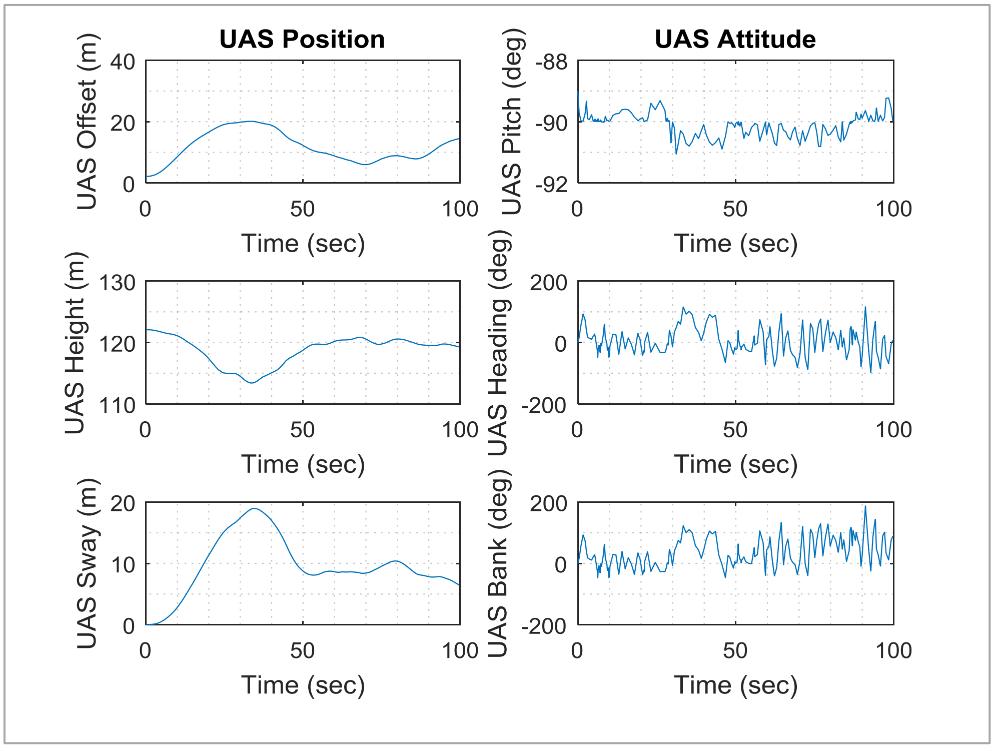

Figure 4.9. UAS Position and Attitude, Lateral Wind Effect

4.8.3. Effect of UAS Thrust Fluctuations on Lateral Motion

Having reliable data regarding thrust fluctuations in a specific flying vehicle, effect of this force fluctuations on the vehicles stability and operating point can be simulated. As a case study and in order to demonstrate the simulation, consider a simple harmonic thrust fluctuation in which the thrust magnitude is modeled in the exact same way as Section 3.8.3:

$$Thrust = T_m + T_f \left(\sin \frac{2\pi}{f} t + \varphi \right) \qquad (4.87)$$

where the initial phase φ does not affect the result. The variable thrust parameters are defined in Table 4.4. The UAS and the cable segments are initially positioned at 89 degrees with respect to the horizon to avoid singularity in the Euler Angle calculations. Then the system simulation starts with this thrust fluctuation. Since there is no initial positon or force in Y direction, the system states in the lateral direction remain zero and the motion is planar.

Table 4.4. UAS Thrust with fluctuations

Parameter	Label	Value	Units
UAS Mean Thrust	T_m	283.59	[N]
Fluctuation Amplitude	T_f	50	[N]
Fluctuation Frequency	f	1	[Hz]

Figure 4.10 shows the UAS position on horizontal and vertical planes. The time response for the UAS position and attitude can also be seen in Figure 4.11. According to this figure, it can be observed that a *1 Hz* thrust fluctuation with an amplitude of *50 N*, causes a decaying oscillation with a period of *57 sec* in the UAS offset. The thrust fluctuation causes an oscillation in the UAS altitude with a frequency of *1 Hz* and a peak-to-peak amplitude of 0.15 meters; that is the vertical UAS motion is 15 centimeters at most. However the UAS pitch changes every two seconds in comparison to the thrust frequency of *1 Hz* with a peak-to-peak amplitude of 1 degrees.

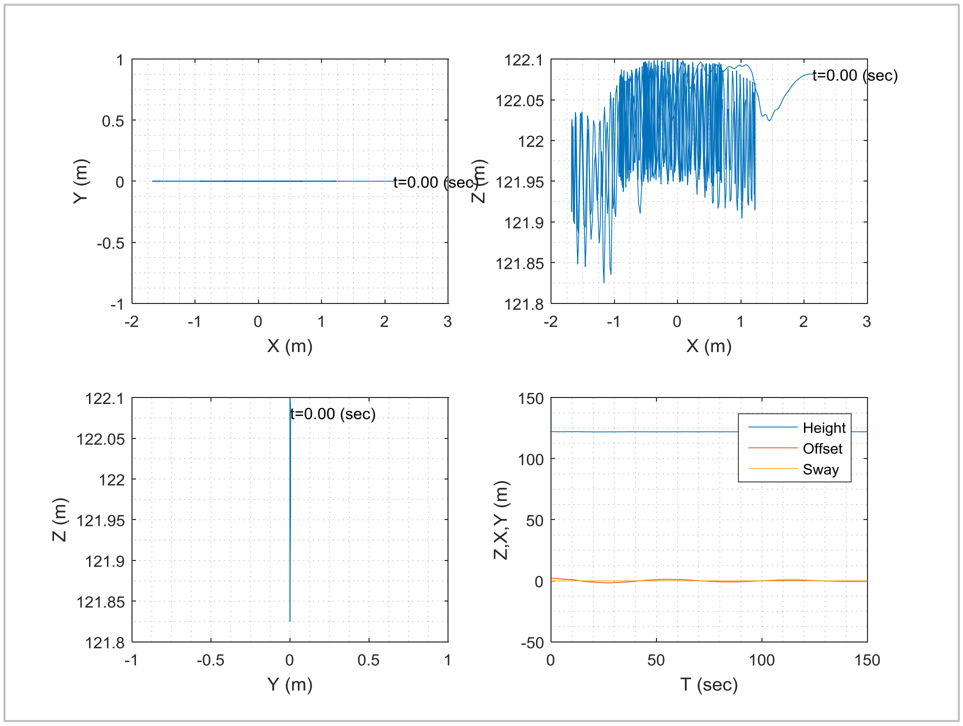

Figure 4.10. UAS position results due to thrust fluctuations

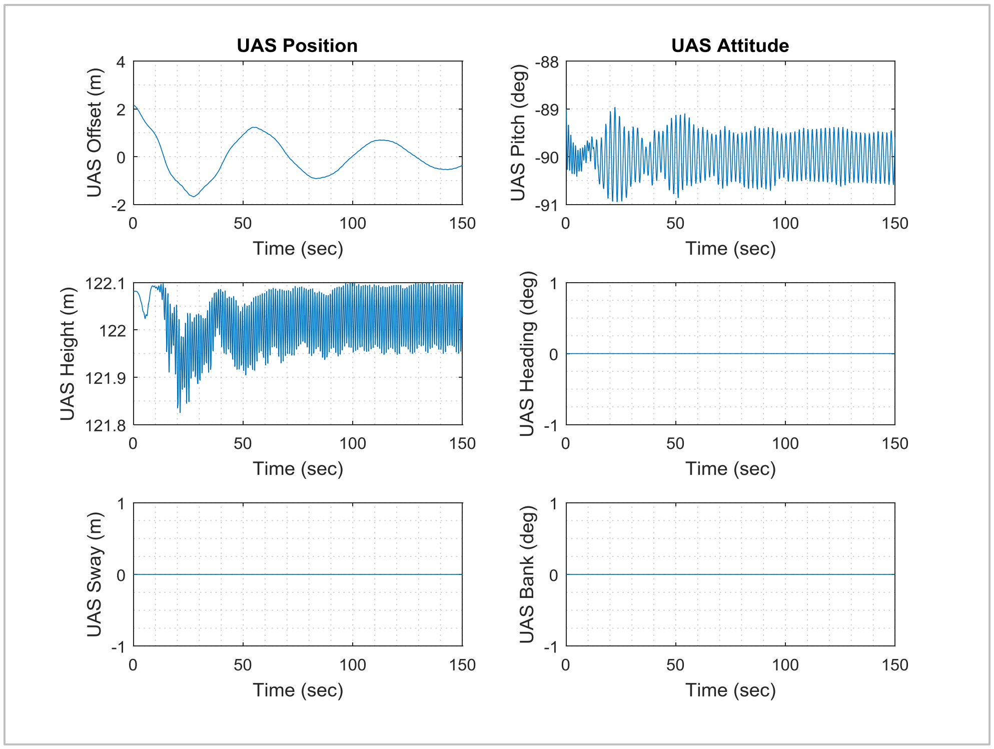

Figure 4.11. UAS position and attitude under thrust fluctuations

4.9. Conclusions

This chapter was dedicated to 3D modelling and simulation of the cable which is used to tether a captive UAS to its base station. Modelling of the system was accomplished by discretizing the cable into finite rigid body segments using a lumped-mass model. An Euler Angle representation were used to specify the coordinates of each segment after which equations of motion were derived using Lagrange’s Method for multibody systems. The UAS itself was modeled as an additional cable segment with specific length, mass and inertia matrix. The lumped-mass model found to be insufficient in terms of solving the problem numerically as it caused the coefficient matrices to be singular. Therefore, a thin rod model including inertia matrices of the cable segments were used instead.

The equations of motion were then manipulated in order to reduce the mathematical order of the differential equations, enabling the author to numerically solve them using the 4th order Runge-Kutta integration routine in MATLAB software package. A torsion stiffness model was added since the model showed unreasonable results due to the limitation of the torsion-free model. The results were generated considering the torsion stiffness in the cable and were compared.

Effect of lateral wind on the dynamic of the cable and the UAS was also presented to demonstrate stability of the system in such conditions. Another Effect that is investigated in this chapter, is the effect of variable wind on the system’s stability. A wind model is used in which wind velocity and heading angle changes randomly with a normal distribution. In addition, UAS take off in a windy condition is investigated along with effect of UAS thrust fluctuation on the UAS states.

Chapter 5.

Conclusions and Future Work

In this work, a captive tether/UAS system was modeled mathematically and the results were simulated numerically. A summary of the contributions, recommendations and potential future work is given in this chapter.

5.1. Summary

Firstly, in Chapter 2, a computer algorithm was developed in order to predict the static behavior of the tether/UAS system under different loading conditions. This model uses catenary equations as the basis of modelling and was able to predict different parameters of the system given certain flying conditions.

Two different case studies were used to demonstrate the capability of the developed algorithm in application to different scenarios and some results were illustrated. Using this algorithm and results, one can understand the sensitivity of the tensions both at the UAS and at the winch to different parameters, including length, and UAS vertical and horizontal position. The results show the effect of cable length on UAS behavior as well as UAS pitch and thrust requirements to maintain vertical and horizontal position.

Dynamic behavior as well as the effect of wind was neglected in this chapter. The study is also limited to the cases where external forces are absent except at the endpoint. However, it is possible to implement external forces into the algorithm with more complicated mathematical derivations. The algorithm is powerful in predicting a quasi-static system where the acceleration is almost zero and as a result, no inertial forces are present. With more detailed mathematical derivation, one may extend this model to one that can analyze external forces such as uniform wind, etc.

In Chapter 3, the cable/UAS system was modeled by writing the Lagrange's Equations for the multibody system of discretized cable. The cable was modelled as a multibody system of finite rigid segments. The equations of motion were derived and solved numerically. Three different case studies were investigated including effect of wind speed on UAS stability, takeoff in extreme wind, and effect of thrust fluctuations on the dynamics of the system.

The model is limited in the sense that it does not consider the effect of cable strain as well as bending in the cable. A more thorough modeling could be carried out to implement and investigate these parameters. In addition, a more in-depth modelling of the UAS could yield in more accurate and reliable results.

In Chapter 4, a 3D model was developed for the same system. A lumped-mass model was used to represent the discretized cable model. Euler Angles were used to represent each segments orientation in space. The Lagrange's Equations were used in order to derive equations of motion for such a multi-body cable/UAS system. The problem was solved numerically using 4th order Runge-Kutta numerical integration method. A few case studies were simulated including torsion stiffness effect, effect of lateral wind on the dynamic of the cable and the UAS, and effect of randomly varying wind speed. In addition, the effect of UAS thrust fluctuations were investigated when the UAS is in hovering mode.

5.2. Future Work

Modelling flexible media is a vast field and there have been several areas identified during this work that the author would like to explore in future. By considering the following effects in the models, one will be able to generate a more accurate model for a flexible cable.

5.2.1. Bending Stiffness

It would make the model more accurate if one implements the effect of bending stiffness in the cable segment joints. Cables do not behave exactly similar to chains in practice and usually a cable shows resistance towards bending. This is due to the resistance that the cable shows if the bending radius gets smaller than the bending radius limit of the cable which depends on the material, wire arrangement, diameter, etc. By including this stiffness into the model the cable is being limited in extreme scenarios which enables the model to replicate the real physics of the problem with a better accuracy.

5.2.2. Longitudinal Strain

Adding the effect of longitudinal strain in each cable segment allows the model to simulate the change in length of the tether due to tension. In most applications, cables have a relatively low stiffness allowing the cable to stretch throughout its length. Addition of this factor could improve the reliability and validity of the model significantly.

5.2.3. Reeling Process

In many marine, submarine, aerial, and space applications, the cable is usually used to deploy a system or a vehicle. In most of these applications the vehicle or the mechanical system is deployed using a winch and a tether by paying out the cable Thus, modelling the reeling process could be a worthwhile addition to this study. The author hopes to be able to model this process using a method called Updated Lagrangian Method which is employed in a similar work by Muttin [29].

5.2.4. Contact Modelling

During each operation, it might occur that the cable comes in contact with its surroundings [9]. As an example, the cable might have to be laid on the ground during landing off-base due to an emergency. Modeling these types of contacts can be a helpful improvement to the model.

5.2.5. Sophisticated UAS Modelling

With implementation of a sophisticated UAS model, one can model different flight scenarios more realistically. This can include a flight controller based on position feedback as well as more accurate aerodynamic modelling of the UAS based on its geometry. In addition, detailed modelling of the UAS thrusters based on their relative position on the

UAS with respect to the center of mass could be very helpful in predicting the system's behavior.

5.2.6. Base Station Motions

The cable-body system studied in this work can be used in a lot of applications including oil spill detection in the ocean [28], [29], as well as civil or border surveillance. It could be also used in mining sites for surveillance of the operation. In some of these cases, it is required to mount the base station on a moving vehicle; either a ground vehicle or a vessel in the case of the marine operations. Thus, in the scope of development of this cable model further, it is expected to extend the model to add the effect of base station motions or oscillations.

5.3. Recommendations

In this work the coordinate system used to represent each cable segment will cause the segment pitch angle to be close to 90 degrees. It is recommended that the choice of Euler Angles be different according to the common operating range of the system in order to leave a bigger margin between the operating angles and the gimbal lock angles of the system. In spite of this fact, this model tends to be able to predict the behavior at most of the cases without any singularity in the system.

References

[1] Y. Choo and M. J. Casarella, “A Survey of Analytical Methods for Dynamic Simulation of Cable-Body Systems.,” *J. Hydronautics*, vol. 7, no. 4, pp. 137–144, 1973.

[2] R. F. Dominguez, A. M. ASCE, and C. E. Smith, “Dynamic Analysis of Cable Systems,” *J. Struct. Div.*, vol. 92, no. 8, pp. 1817–1834, 1972.

[3] T. P. Dreyer and J. H. Van Vuuren, “A comparison between continuous and discrete modelling of cables with bending stiffness,” vol. 23, pp. 527–541, 1999.

[4] J. W. Kamman and R. L. Huston, “Modeling of Variable Length Towed and Tethered Cable Systems,” *J. Guid. Control. Dyn.*, vol. 22, no. 4, pp. 602–608, Jul. 1999.

[5] J. W. Kamman and R. L. Huston, “Multibody Dynamics Modeling of Variable Length Cable Systems,” pp. 211–221, 2001.

[6] B. Buckham, M. Nahon, M. Seto, X. Zhao, and C. Lambert, “Dynamics and control of a towed underwater vehicle system , part I : model development,” vol. 30, pp. 453–470, 2003.

[7] C. Lambert, M. Nahon, B. Buckham, and M. Seto, “Dynamics and control of towed underwater vehicle system , part II : model validation and turn maneuver optimization,” vol. 30, pp. 471–485, 2003.

[8] S. Prabhakar and B. Buckham, “Dynamics Modeling and Control of a Variable Length Remotely Operated Vehicle Tether,” *Proc. Ocean. 2005 MTS/IEEE*, pp. 1–8.

[9] B. J. Buckham, “Dynamics modelling of low-tension tethers for submerged remotely operated vehicles,” University of Victoria, 2003.

[10] B. Buckham and M. Nahon, “Dynamics simulation of low tension tethers,” *Ocean. ’99. MTS/IEEE. Rid. Crest into 21st Century. Conf. Exhib. Conf. Proc. (IEEE Cat. No.99CH37008)*, vol. 2, pp. 757–766, 1999.

[11] B. Buckham, F. R. Driscoll, and M. Nahon, “Development of a Finite Element Cable Model for Use in Low-Tension Dynamics,” vol. 71, pp. 476–485, 2004.

[12] J. Genin and T. C. Cannon, “Equilibrium Configuration and Tensions of a Flexible Cable in a Uniform Flowfield,” *J. Aircr.*, vol. 4, no. 3, pp. 200–202, 1967.

[13] R. R. Huffman and J. Genin, “The Dynamical Behavior of a Flexible Cable in Uniform Flow Field,” *Aeronaut. Q.*, no. 22, pp. 183–195, 1971.

[14] J. E. Quisenberry and A. S. Arena, “Discrete Cable Modeling and Dynamic Analysis,” in *AIAA Aerospace Sciences Meeting and Exhibit*, 2006.

[15] J. E. Quisenberry and A. S. Arena, “Dynamic simulation of low altitude aerial tow system,” in *AIAA Atmospheric Flight Mechanics Conference and Exhibit*, 2004, p. 4813.

[16] J. E. Quisenberry Jr., “Dynamic Simulation of Low Altitude Aerial Tow Systems,” 2005.

[17] B. Hembree and N. Slegers, “Efficient tether dynamic model formulation using recursive rigid-body dynamics,” *Proc. Inst. Mech. Eng. Part K J. Multi-body Dyn.*, vol. 224, no. 4, pp. 353–363, Dec. 2010.

[18] V. Johansen, S. Ersdal, A. J. Sørensen, and B. Leira, “Modelling of inextensible cable dynamics with experiments,” *Int. J. Non. Linear. Mech.*, vol. 41, no. 4, pp. 543–555, May 2006.

[19] A. Montano, M. Restelli, and R. Sacco, “Numerical simulation of tethered buoy dynamics using mixed finite elements q,” vol. 196, pp. 4117–4129, 2007.

[20] S. Surendran and M. Goutam, “Reduction in the dynamic amplitudes of moored cable systems,” *Ships Offshore Struct.*, vol. 4, no. 2, pp. 145–163, May 2009.

[21] W. Kang and I. Lee, “Analysis of Tethered Aerostat Response Under Atmospheric Turbulence Considering Nonlinear Cable Dynamics,” *J. Aircr.*, vol. 46, no. 1, pp. 343–348, Jan. 2009.

[22] A. Rajani, R. S. Pant, and K. Sudhakar, “Dynamic Stability Analysis of a Tethered Aerostat,” vol. 47, no. 5, 2010.

[23] K. a. Stanney and C. D. Rahn, “Response of a tethered aerostat to simulated turbulence,” *Commun. Nonlinear Sci. Numer. Simul.*, vol. 11, no. 6, pp. 759–776, Sep. 2006.

[24] S. P. Jones and L. D. Schroeder, “Nonlinear Dynamic Simulation of a Tethered Aerostat: A Fidelity Study,” *J. Aircr.*, vol. 38, no. 1, pp. 64–68, Jan. 2001.

[25] S. P. Jones and J. A. Krausman, “Nonlinear dynamic simulation of a tethered aerostat,” *J. Aircr.*, vol. 19, no. 8, pp. 679–686, 1982.

[26] M. M. Nicotra, R. Naldi, and E. Garone, “Taut Cable Control of a Tethered UAV,” pp. 3190–3195, 2014.

[27] S. Lupashin and R. D'Andrea, "Stabilization of a flying vehicle on a taut tether using inertial sensing," *2013 IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, pp. 2432–2438, Nov. 2013.

[28] F. Muttin, "Modelling of a captive unmanned aerial system teledetecting oil pollution on sea surface," *arXiv Prepr. arXiv1302.3774*, 2013.

[29] F. Muttin, "Umbilical deployment modeling for tethered UAV detecting oil pollution from ship," *Appl. Ocean Res.*, vol. 33, no. 4, pp. 332–343, Oct. 2011.

[30] A. Antoniou and W.-S. Lu, *Practical Optimization: Algorithms and Engineering Applications*. Springer Science & Business Media, 2007.

[31] P. Williams and P. Trivailo, "Cable deployment control for towed aerial-cable payload pick-up and delivery system," in *Proceedings of the Land Warfare Conference*, 2004, pp. 313–329.

[32] S. F. Hoerner, *Fluid-Dynamic Drag: Practical Information on Aerodynamic Drag and Hydrodynamic Resistance*. Hoerner Fluid Dynamics Midland Park, NJ, 1965.

[33] D. T. Greenwood, *Advanced dynamics*. Cambridge University Press, 2006.

[34] D. T. Greenwood, *Principles of dynamics*. Prentice-Hall Englewood Cliffs, NJ, 1988.

[35] A. A. Shabana, *Dynamics of multibody systems*. Cambridge university press, 2013.

[36] S. Poulin and A. Larsen, "Drag Loading of Circular Cylinders Inclined in The Along-wind direction," *J. Wind Eng. Ind. Aerodyn.*, vol. 95, no. 9–11, pp. 1350–1363, 2007.

[37] S. T. Thornton and J. B. Marion, "Classical Dynamics of Particles and Systems," 2004.

[38] J. L. Meriam and L. G. Kraige, *Engineering Mechanics: Dynamics*, vol. 2. John Wiley & Sons, 2012.

[39] M. Documentation, "Matlab documentation," *Matlab*. p. R2012b, 2012.

[40] J. C. Butcher, *The Numerical Analysis of Ordinary Differential Equations: Runge-Kutta and General Linear Methods*. J. Wiley, 1987.

[41] J. L. Meriam and L. G. Kraige, *Engineering Mechanics: Statics*, vol. 1. John Wiley & Sons, 2006.

[42] D. A. Fourie, “The Dynamic Modelling and Control System of a Tethered Aerostat for Remote Sensing Applications by,” University of Stellenbosch, 2009.

Appendix A.

MATLAB Program Codes

A.1. Chapter 2

Below are presented a few sample MATLAB codes as used in the static simulation of Chapter 2.

Function “MTeth8” defines the function of unknowns for the cable.

```
% Implementation of formula derivations for Catenary cables
% Derived and Implemented by: Sina Doroudgar, Rigid Robotics Inc.,

% Description:
%Three variable function to be solved for the static Tether
%variables are: x=[x1 p H]',
%p, the offset from the base station, H, the height, and T1 the cable
tension at winch should be entered.

function G=MTeth8(x,L,T2ver,T2hor);
g=9.81;            % gravitational acceleration
mu=0.14*g;       % cable's weight per unit length
Th2Radian=atan(T2ver/T2hor)
% p=p
% H=H
% L=L
T0=T2hor;
G=[x(3)-(T0/mu).*(cosh(mu.*(x(1)+x(2))/T0)-cosh(mu.*x(1)/T0));
%function G defines a vector of two functions to be solved
    L-(T0/mu).*(sinh(mu.*(x(1)+x(2))/T0)-sinh(mu.*x(1)/T0));
%     tan(Th2Radian)-sinh(mu.*(x(1)+x(2))/T0)];
    T2ver/T2hor-sinh(mu.*(x(1)+x(2))/T0)];
% sqrt(T2ver.^2+T2hor.^2)-T0*cosh(mu.*(x(1)+x(2))/T0)];
% T0-T2.*(cos(atan(sinh(mu.*(x(1)+x(2))/T0))))];
%F=G'*G;
% global p
% global H
% global L
end
```

The script “Mtethsolve8.m” solves the system using the inputs as defined in the codes.

```
% Implementation of formula derivations for Catenary cables
% Derived and Implemented by: Sina Doroudgar, Rigid Robotics Inc.,
% Description: Solves the Catenary equation for known parameters:
offset of p, height
% of H and cable length of L
% The variable is implemented as vector x=[x1;T0]



function F = Mtethsolve8(L,T2ver,T2hor);
format long
x0=[122;25;120];    % iteration start point
g=9.81;             % gravitational acceleration
mu=0.14*g;        % cable's weight per unit length

 [x,fval] = fsolve(@(x) MTeth8(x,L,T2ver,T2hor), x0)

Th2Radian=atan(T2ver/T2hor);
T0=T2hor;

Th1Radian=atan(sinh(x(1)*mu/T0));
Th2Radian=atan(sinh((x(1)+x(2))*mu/T0));

y1=T0/mu.*(cosh(mu.*x(1)/T0)-1);
y2=T0/mu.*(cosh(mu.*(x(1)+x(2))/T0)-1);




if abs(fval) > 1e-3;
    display ('Function cannot calculate forces accurately;')
    display('The offset is too small and the error of the code is high
at this point.')
    display('The force on the UAV is approximated by the cable weight
which is approximately equal to')
    T2=mu.*x(3)
    display('and the force at the base station could be approximated by
zero.')
    T1=0
end
fval
if abs (fval) <= 1e-3;
    format short
    display 'at altitude(m)='
    H=x(3)
    display 'with a tether length of (m)='
    L=L
    display 'at offset position (m)='
    p=x(2)
    display 'Horizontal Force (N) is:'
    T0= T0
    display 'angle at winch (degrees) is:'
    Th1 =Th1Radian*180/pi
    display 'tension force at winch (N) is:'
    T1=T0/cos(Th1Radian)
    display 'angle at UAV (degrees) is:'
    Th2 =Th2Radian*180/pi
    display 'tension force at UAV (N) is:'
    T2=T0/cos(Th2Radian)

end


xt=[x(1):.01:x(1)+p];
TTT=T0*cosh(mu.*xt./T0);
plot(xt-x(1),TTT,':')
hold on

end
```

The script “Mtethplote8.m” solves and plots the cable shape for the system using the inputs as defined in the codes.

```
% Implementation of formula derivations for Catenary cables
% Derived and Implemented by: Sina Doroudgar, Rigid Robotics Inc.,

% Description: Plots the hyperbolic shape function of the catenary
tether

function [y,H,p] = Mtethplot8(L, T2ver,T2hor);
x0=[2;25;120];      % iteration start point
g=9.81;            % gravitational acceleration
mu=0.14*g;       % cable's weight per unit length

% global p
% global H
% global L

% [x,fval]=fsolve(@Teth3,[x0 p H L]);
% if (L<=sqrt(p.^2+H.^2))
%   display ('ERROR: Cable length is insufficient for this altitude.
Please enter a cable length larger than sqrt(p.^2+H.^2)')
%     return
% else

[x,fval] = fsolve(@(x) MTeth8(x, L,T2ver,T2hor), x0);


Th2Radian=atan(T2ver/T2hor);
T0=T2hor;
```

```
Th1Radian=atan(sinh(x(1)*mu/T0));
Th2Radian=atan(sinh((x(1)+x(2))*mu/T0));

y1=T0/mu.*(cosh(mu.*x(1)/T0)-1);
y2=T0/mu.*(cosh(mu.*(x(1)+x(2))/T0)-1);

p=x(2);
H=x(3);

xt=[x(1):.01:x(1)+p];      % generates points to graph the cable shape
y=T0/mu.*(cosh(mu.*xt/T0)-1);      % representation of the cable graph
in terms of variable x (called xt here)
plot (xt-x(1),y-y(1));
% axis([x(1)-.5 x(1)+p+.5 -y(1)-5 H+10]);
axis([-2 p+2 -20 H+10]);
grid on;
hold on;
axis equal;
rectangle('Position',[p-2,H-1,4,2],'FaceColor','y');
% plot(plot::Circle2d(1, [0, 1])
% plot:Circle2d(1, <[0, 1]>)
% filledCircle([0,1],1,1000,'b');
% end
text(xt(round(end/1.1))-x(1),y(round(end/1.1))-
y(1),['L=',num2str(L)],'fontsize',6);

end
```

A.2. Chapter 3

The following codes are used in simulation and solution of the 2D cable-body system’s dynamics:

The script “simParams.m” loads the geometrical and mechanical system parameters.

```
%% simParams


global segLength
global lengthVector
global massVector
global n
global massMatrix
global m
global k
global windVelocity
global airDensity
global segArea
global Cf
global Cd
global L
global UASInertia
global inertias
global g
global T2hor
global T2ver
global UASTHRUST
global UASArea
global MaxU
global MaxW
global UTrig
global WTrig




UASArea=[];

segNo=20;
n=segNo;
k=2*n+2;

MaxU=6; %wind horizontal velocity_mps
UTrig=1; %wind horizontal velocity application time
MaxW=0; %wind vertical velocity_mps
WTrig=10000; %wind vertical velocity application time
Cd=1.2;    % slender cylinder drag coefficient
Cf=0.05;   % slender cylinder friction coefficient
L=122;%-122/(n+1);  % cable length
g=9.81; %gravitational acceleration
airDensity=1.225; %kg/m^3
massPerLength=0.14; %cable mass per length (kg/km)
cableDia=0.0135; %Cable diameter (m)
E=0.4*200e9; %cable modulus of elasticity (Pa)
windVelocity=zeros(2*n+2,1);

n=k/2-1; %number of cable segments
segLength=L/n;
m=segLength*massPerLength;
massUAS=10.56;%35; % UAS mass(kg)
lUAS=0.1;%0.2; %UAS segment length(m)
UASInertia=1.3256;   %kgm^2
massMatrix=zeros(n+1,n+1);
lengthVector=zeros(n+1,1);
lengthVector(1:n)=segLength;
lengthVector(n+1)=lUAS;
segArea=lengthVector*cableDia;   %characteristic length of cable
segment
segArea(n+1)=(1.012*0.025);
% UASArea(1)=;
% UASArea(2)=;
% moments=zeros(n+1,1); % External Moments Vector
```

```
massVector=zeros(n+1,1);
massVector(1:n)=m;
massVector(n+1)=massUAS;


%% Inertias
inertias=zeros(n+1,1);
inertias(n+1)=UASInertia;   %Deactivated for validation
%%

for i=1:n+1;
    for j=1:n+1;
%         massMatrix(i,j)=(n+1-max(i,j))*m+massUAS;
        massMatrix(i,j)=sum(massVector(max(i,j):end));

    end
end
```

The function “ndimrigid.m” defines the equations of motion completely including the coefficient matrices, etc.

```
%ndimrigid

function dQ = ndimrigid(t,Q);
global segLength
global lengthVector
global massVector
global n
global k
global massMatrix
global windVelocity
global airDensity
global segArea
global Cf
global Cd
global inertias
global g
global T2hor
global T2ver
global UASTHRUST
global MaxU
global MaxW
global UTrig
global WTrig
UASTHRUST =283.5936+10.*sin(2.*pi./1.*t);
U=MaxU*(heaviside(t-UTrig));
W=MaxW*(heaviside(t-WTrig));
windVelocity(1:2:end,1)=U;
windVelocity(2:2:end,1)=W;
temp=0;
velocity=zeros(2*n+2,1);
% windVelocity=zeros(2*n+2,1);
```

```
rotationBigMatrix=[0];
forceSegment=[0];
% for i=1:n+1;
% windVelocity(1:2:end,1)=U;
% windVelocity(2:2:end,1)=W;
% end
%%
dQ=zeros(k,1);
moments=zeros(n+1,1); % External Moments Vector

A=zeros(1);
B=zeros(1);
AA=[];
BB=[];


for i=1:n+1;
    for j=1:n+1;


A(i,j)=massMatrix(i,j)*lengthVector(i)*lengthVector(j)*cos(Q(n+1+i)-
Q(n+1+j))+(i==j)*inertias(i);%+(i==j)*(sum(AA(i:end,1))+inertias(i)+1/4
*massVector(i)*lengthVector(i).^2);
    end
end

for i=1:n+1;
    for j=1:n+1;
%         for i1=1:n+1;

B(i,j)=massMatrix(i,j)*lengthVector(i)*lengthVector(j)*sin(Q(n+1+i)-
Q(n+1+j));%-(i==j)*(sum(BB(i:end,1)));
    end
end

% newA=[-inv(A)*B*diag(Q(1:n+1))  zeros(n+1);
    newA=[-A\(B*diag(Q(1:n+1)))  zeros(n+1);

      eye(n+1)                  zeros(n+1)];
%   newC=[-inv(A)*(Q(1:n+1))  zeros(n+1);...
%       zeros(n+1)                  zeros(n+1)];

%   dQ=newA*Q+0*newC;

%% Rotation Matrix for Velocity

for i=1:n+1;
    temp=[0;0];
      for j=1:i;
          RDotThet=[-sin(Q(j+n+1))  -cos(Q(j+n+1));
              cos(Q(j+n+1))       -sin(Q(j+n+1))];
          velocity(2*i-1:2*i,1)=
temp+Q(j)*RDotThet*[lengthVector(j);0];
          temp=velocity(2*i-1:2*i,1);
      end
end


relativeVelocity=windVelocity-velocity;


for i=1:n+1;
    rotationBigMatrix(2*i-1:2*i,2*i-1:2*i)=[cos(Q(n+1+i)) -
sin(Q(n+1+i)) ;sin(Q(n+1+i)) cos(Q(n+1+i))];
end

% relVelSegment=inv(rotationBigMatrix)*relativeVelocity;
relVelSegment=(rotationBigMatrix)\relativeVelocity;

% Wind Forces on Cable
for i = 1:n;
    forceSegment(2*i-
1,1)=0.5*airDensity*segArea(i)*3.14*Cf*relVelSegment(2*i-
1,1)*sqrt(relVelSegment(2*i-1,1).^2+relVelSegment(2*i,1).^2);

forceSegment(2*i,1)=0.5*airDensity*segArea(i)*relVelSegment(2*i,1)*(Cf*
3.14*sqrt(relVelSegment(2*i-
1,1).^2+relVelSegment(2*i,1).^2)+Cd*abs(relVelSegment(2*i,1)));
end

%% Wind forces on UAS
for i = n+1;
forceSegment(2*i-1,1)=0.5*airDensity*segArea(i)*relVelSegment(2*i-
1,1)*(Cf*3.14*sqrt(relVelSegment(2*i-
1,1).^2+relVelSegment(2*i,1).^2)+2*Cd*abs(relVelSegment(2*i-1,1)));
forceSegment(2*i,1)=0.5*airDensity*segArea(i)*relVelSegment(2*i,1)*(3.1
4*Cf*sqrt(relVelSegment(2*i-
1,1).^2+relVelSegment(2*i,1).^2)+Cd*abs(relVelSegment(2*i,1)));
end

% forceSegment(2*n+1) = T2ver;
% forceSegment(2*n+2) = T2hor;

% forceSegment(2*n-1)=200*(heaviside(t-10)-heaviside(t-10.6));
% forceSegment(2*n+2)=20;
forceWld=rotationBigMatrix*forceSegment;
%
forceWld(2*n+1:2*n+2,1)=[0*sin(2*pi*t/1);1*(massPerLength*L+massUAS)*g+
0*1.5*sin(2*pi*t/0.1)];

forceWld(2*n-1) = 1*forceWld(2*n-1)+UASTHRUST*(1)*cos(Q(2*n+2));
forceWld(2*n+2) = 1*forceWld(2*n+2)+UASTHRUST*(1)*sin(Q(2*n+2));

%%Impact Force
```

```
%
forceWld(3*round((n+1)/4)+1)=forceWld(3*round((n+1)/4)+1)+40.*(heavisid
e(t-60)&heaviside(61-t));
%%
% forceWld(2*n+1)=5*(heaviside(t-5)-heaviside(t-5.1));
%% generalized force (Q) in equations
generalizedForce=zeros(n+1,1);
generalizedForcei=[];
moments(1)=0;
moments(n+1)=0;

for i=1:n;
%     moments(i)=forceWld(2*i-1)*.5*lengthVector(i)*sin(Q(n+1+i))-
forceWld(2*i)*.5*lengthVector(i)*cos(Q(n+1+i));
    tempq=[0];
        for j=i:n+1;
%        if j==i;
%         tempq=[0.5*((forceWld(2*j)-
massVector(j))*lengthVector(i)*cos(Q(n+1+i))-forceWld(2*j-
1)*lengthVector(i)*sin(Q(n+1+i)))];
%        end
        generalizedForcei(i,1)=tempq+(forceWld(2*j)-
massVector(j)*g)*1*lengthVector(i)*cos(Q(n+1+i))-forceWld(2*j-
1)*lengthVector(i)*sin(Q(n+1+i));%-
lengthVector(i)/2*cos(Q(n+1+i));%+moments(i);
        tempq=generalizedForcei(i,1);
    end
    generalizedForce(i,1)=generalizedForcei(i,1)+moments(i);
end

for i=n+1;
%     moments(i)=forceWld(2*i-1)*.5*lengthVector(i)*sin(Q(n+1+i))-
forceWld(2*i)*.5*lengthVector(i)*cos(Q(n+1+i));
    tempqq=[0];
        for j=i:n+1;
%        if j==i;
%         tempq=[0.5*((forceWld(2*j)-
massVector(j))*lengthVector(i)*cos(Q(n+1+i))-forceWld(2*j-
1)*lengthVector(i)*sin(Q(n+1+i)))];
%        end
        generalizedForcei(i,1)=tempqq+(forceWld(2*j)-
massVector(j)*g)*1*lengthVector(i)*cos(Q(n+1+i))-forceWld(2*j-
1)*lengthVector(i)*sin(Q(n+1+i));%-
lengthVector(i)/2*cos(Q(n+1+i));%+moments(i);
        tempqq=generalizedForcei(i,1);
    end
    generalizedForce(i,1)=generalizedForcei(i,1)+moments(i);
end




generalizedForce;
```

```
% generalizedForce=diag(forceSegment)*lengthVector/2
t
% dQ=newA*Q+[inv(A)*generalizedForce; zeros(n+1,1)];
dQ=newA*Q+[A\generalizedForce; zeros(n+1,1)];
```

____

Script “solveSystem.m” solves the equations of motion given the initial conditions defined in the same file in the form of segment angles.

```
%%solveSystem

tic

simParams

thet0=[zeros(1,n+1) ones(1,n+1)*(1)*pi/2]';  % define initial
conditions
thet0(end)=pi/2;
options = odeset('RelTol',1e-4*eye(1,size(thet0,2)),'AbsTol',1e-
6*eye(1,size(thet0,2)));
global segLength
global m
global lengthVector
global massVector
global n

dimTol=0;

timeEnd = 100;      % define simulation time

[T,Y] = ode45(@ndimrigid,[0 timeEnd],thet0,options);
solvingTime=toc/60
disp('minutes')

plotSystem
forceCalc
plottest
```

____

Script “plotSystem.m” calculates and plots the results in terms of acceleration, position and velocities.

```
%% plot system

tic

global segLength
global lengthVector
global massVector
global n
global massMatrix
global m
global k
global windVelocity
global airDensity
global segArea
global Cf
global Cd

limlim = (n+1)*segLength*3;

 %Plotting the columns of the returned array Y versus T shows the
solution
XX=[];
ZZ=[];
X=[];
Z=[];
TT=T(1:20:end,:);
YY=Y(1:20:end,:);
accel = 0*(YY);
for j=1:size(TT);
%     clf
    X(1,j)=lengthVector(1)*cos(YY(j,n+1+1));%-
lengthVector(1)/2*cos(YY(j,n+1+1));
    Z(1,j)=lengthVector(1)*sin(YY(j,n+1+1));%-
lengthVector(1)/2*sin(YY(j,n+1+1));
    for i=2:n+1;
        X(i,j)=X(i-1,j)+lengthVector(i)*cos(YY(j,n+1+i));
        Z(i,j)=Z(i-1,j)+lengthVector(i)*sin(YY(j,n+1+i));
%
%         sumx=X(i);
%         sumz=Z(i);
    end


    filename = 'testnew51.gif';

[accel(j,:)]=ndimrigid(TT(j), YY(j,:)');
acceleration=accel(:,1:n+1);
velocity=accel(:,n+2:end);


 figure(1)
 subplot(312)
    plot(TT(:,1),Z(end,:),'-')
    xlabel('Time (sec)')
    ylabel('UAS Height (m)')
    hold on
    grid minor
    subplot(311)
    plot(TT(:,1),X(end,:),'-')
    xlabel('Time (sec)')
    ylabel('UAS Offset (m)')
    hold on
    grid minor

    subplot(313)
    pitch=-pi/2*ones(size(YY(:,2*n+2),1),1)+YY(:,2*n+2);
    pitch=pitch*180/pi;
    plot(TT(:,1),pitch,'-')
    xlabel('Time (sec)')
    ylabel('UAS Pitch (deg)')
    hold on
    grid minor
% legend('WindSpeed=')
   print('positions','-dmeta')
   print('positions','-dpng')
   saveas(gcf,'positions.fig')
    %}

        %%
    figure(2)
    angle=atan2(Z(1,:),X(1,:));
    angle=angle*180/pi+360;
    plot(TT(:,1),angle,'-');
    hold on
    legend on
    plot(TT(:,1),YY(:,n+2)*180./pi-180,'-k')
    legend('Cable Angle from calculation','Cable Angle from the
results(measured from -X)')
    xlabel('Time (sec)')
    ylabel('Cable Angle (deg)')

    print('angles','-dmeta')
   print('angles','-dpng')
   saveas(gcf,'angles.fig')

    figure(3)

    plot(TT(:,1),acceleration(:,end),'-');
    hold on
    legend on
    legend('Angular Acceleration')
    xlabel('Time (sec)')
    ylabel('Angular Acceleration(rad/sec^2)')

    print('Angular Acceleration','-dmeta')
   print('Angular Acceleration','-dpng')
   saveas(gcf,'Angular Acceleration.fig')

    figure(4)
    plot(TT(:,1),velocity(:,end),'-');
    hold on
    legend on
    legend('Angular Velocity')
    xlabel('Time (sec)')
    ylabel('Angular Velocity(rad/sec)')

     print('Angular Velocity','-dmeta')
   print('Angular Velocity','-dpng')
   saveas(gcf,'Angular Velocity.fig')



plottingTime=toc/60
disp('minutes')
```

The script “forceCalc.m” calculates and plots the forces in the cable against time.

```
%% forces
global g
global massVector
global lengthVector
global T2hor
global T2ver
global UASTHRUST


interForce=0*YY;
baseForce=0*YY(:,1:2);
linAccel=0*YY;

for jj=1:size(TT);


for i=1:n+1;
    tempa=[0;0];
      for j=1:i;
          RDotThet=[-sin(YY(jj,j+n+1))  -cos(YY(jj,j+n+1));
              cos(YY(jj,j+n+1))       -sin(YY(jj,j+n+1))];

          RDotDotThet=[-cos(YY(jj,j+n+1))  sin(YY(jj,j+n+1));
              -sin(YY(jj,j+n+1))       -cos(YY(jj,j+n+1))];

          linAccel(jj,2*i-1:2*i)=
tempa+YY(jj,j).^2*RDotDotThet*[lengthVector(j);0]+acceleration(jj,j)*RD
otThet*[lengthVector(j);0];
%
          tempa=linAccel(jj,2*i-1:2*i);
          tempa=tempa';
%           tempa'
```

```
      end
%
end


hold on
%% Force Calculations




T2hor = UASTHRUST*(1)*cos(YY(jj,2*n+2));
T2ver = UASTHRUST*(1)*sin(YY(jj,2*n+2));

for i1 = n+1
interForce(jj,2*i1-1:2*i1)=[T2hor T2ver];%-massVector(i1-1)*[0 g];
end
for i2=n:-1:2;
interForce(jj,2*i2-1:2*i2)=[interForce(jj,2*i2+1:2*i2+2)]-
massVector(i2)*[linAccel(jj,2*i2+1) linAccel(jj,2*i2+2)+g];%-
massVector(i2-1)*[0 g];
end
for i3=1;
interForce(jj,2*i3-1:2*i3)=[interForce(jj,2*i3+1:2*i3+2)]-
massVector(i3)*[linAccel(jj,2*i3+1) linAccel(jj,2*i3+2)+g];
end

baseForce(jj,1:2)=[interForce(jj,1:2)]-massVector(1)*[linAccel(jj,1)
linAccel(jj,2)+g];




end


figure (5)
subplot(211)
plot(TT(:),linAccel(:,1:2:end),'-')
xlabel('Time (sec)')
ylabel('Node Horizontal Accelerations (m/s^2)')
grid minor
title('Node Accelarations')

subplot(212)
plot(TT(:),linAccel(:,2:2:end),'-')
xlabel('Time (sec)')
ylabel('Node Vertical Accelerations (m/s^2)')
grid minor

 print('Node Accel','-dmeta')
   print('Node Accel','-dpng')
   saveas(gcf,'Node Accel.fig')
```

```
figure (6)
subplot(211)
plot(TT(:),interForce(:,1:2:end))
xlabel('Time (sec)')
ylabel('Node Horizontal Forces (N)')
grid minor
title('Node Forces')

subplot(212)
plot(TT(:),interForce(:,2:2:end))
xlabel('Time (sec)')
ylabel('Node Vertical Forces (N)')
grid minor
legend([num2str((2:2:size(interForce,2))')])
% text(TT(end-
30),interForce,['L=',num2str(L),'m'],'fontsize',8);%,'Color',colours(k)
);

 print('Node Forces','-dmeta')
   print('Node Forces','-dpng')
   saveas(gcf,'Node Forces.fig')


figure (7)
subplot(211)
plot(TT(:),baseForce(:,1))
xlabel('Time (sec)')
ylabel('Base Horizontal Forces (N)')
grid minor
title('Base Forces')

subplot(212)
plot(TT(:),baseForce(:,2))
xlabel('Time (sec)')
ylabel('Base Vertical Forces (N)')
grid minor

 print('Base Forces','-dmeta')
   print('Base Forces','-dpng')
   saveas(gcf,'Base Forces.fig')

tension1=[];
tension=[];
for j4=1:n+1;
tension1(j4,1)=norm(interForce(end,2*j4-1:2*j4));
end
tension=[norm(baseForce(end,:));...
    tension1];

figure(8)
Xs=[zeros(1,size(X,2)); X(:,:)];
plot(Xs(:,end),tension,'>')
```

```
xlabel('Node X-Position (m)')
ylabel('Longitudinal Cable Tension(N)')
grid minor
title('Cable Tesnion Comparison')
legend('Discrete Model')
hold on

print('Cable Tesnion','-dmeta')
   print('Cable Tesnion','-dpng')
   saveas(gcf,'Cable Tesnion.fig')
```

A.3. Chapter 4

The following codes are used in simulation and solution of the 3D cable-body system's dynamics:

The script "simParams.m" loads the geometrical and mechanical system parameters.

```
%% simParams

global segLength
global lengthVector
global massVector
global n
global massMatrix
global m
global k
global windVelocity
global airDensity
global segArea
global Cf
global Cd
global L
global G
global Jp
global UASInertia
global inertias
global g
global TX
global TY
global TZ
global UASTHRUST
global UASArea
global MaxU
global MaxV
global MaxW
global UTrig
global VTrig
global WTrig
global BETA UU jk ts
```

```
UU=[];
BETA=[];
ts=0:0.1:150;
for tt=0:2:150;
idx=find(ts==tt);
idx=idx(1);
UU(1,idx:size(ts,2))=3+sqrt(50)*randn(1,1);
end


for tt=0:10:150;
idx=find(ts==tt);
idx=idx(1);
BETA(1,idx:size(ts,2))=45+sqrt(10)*randn(1,1);
end

figure(123)
subplot(121)
hold on
plot(ts,UU);
xlabel('time (sec)')
ylabel('Wind Velocity (m/s)')
grid on
subplot(122)
hold on
plot(ts,BETA)
xlabel('time (sec)')
ylabel('Wind Heading (\beta) (deg)')
grid on
print('wind','-dmeta')
   print('wind','-dpng')
   saveas(gcf,'wind.fig')

figure(12314)
subplot(121)
hold on
plot(ts,UU.*cosd(BETA));
xlabel('time (sec)')
ylabel('Wind Velocity in X direction (m/s)')
grid on
subplot(122)
hold on
plot(ts,UU.*sind(BETA))
xlabel('time (sec)')
ylabel('Wind Velocity in Y Direction(m/s)')
grid on
print('windUV','-dmeta')
   print('windUV','-dpng')
   saveas(gcf,'windUV.fig')
```

```
UASArea=[];

segNo=14;   %define no. of segments
n=segNo;
k=6*n+6;





%% Parameters
UTrig=0;
VTrig=0;
MaxW=0;
WTrig=1000;
Cd=1.2;    % slender cylinder drag coefficient
Cf=0.05;   % slender cylinder friction coefficient
L=122;%-122/(n+1);  % cable length
g=9.81; %gravitational acceleration
airDensity=1.225; %kg/m^3
massPerLength=0.14; %cable mass per length (kg/km)
cableDia=0.0135; %Cable diameter (m)
E=0.4*200e9; %cable modulus of elasticity (Pa)
G=56e6; %effective shear modulus (Pa)
Jp=0.5*pi*(cableDia/2)^4;
windVelocity=zeros(3*n+3,1);

% n=k/2-1; %number of cable segments
segLength=L/n;
m=segLength*massPerLength;
massUAS=10.56;%35; % UAS mass(kg)
lUAS=0.1;%0.2; %UAS segment length(m)
% UASInertia=1.3256;   %kgm^2
UASInertia=[0.94 1.3256 1.3256];   %kgm^2
massMatrix=zeros(n+1,n+1);
lengthVector=zeros(n+1,1);
lengthVector(1:n)=segLength;
lengthVector(n+1)=lUAS;
segArea=lengthVector*cableDia;   %characteristic length of cable
segment
segArea(n+1)=(1.012*0.025);
% UASArea(1)=;
% UASArea(2)=;
% moments=zeros(n+1,1); % External Moments Vector
massVector=zeros(n+1,1);
massVector(1:n)=m;
massVector(n+1)=massUAS;


%% Inertias
inertias=zeros(n+1,3);
% for i=1:n+1
%     inertias(i)=massVector(i).*lengthVector(i).^2./3;
% end
```

```
 for i=1:n
    inertias(i,1)=massVector(i).*(cableDia/2).^2;
    inertias(i,2)=massVector(i).*lengthVector(i).^2;
    inertias(i,3)=massVector(i).*lengthVector(i).^2;
 end
inertias(n+1,:)=UASInertia;   %Deactivated for validation
%%

for i=1:n+1;
    for j=1:n+1;
%         massMatrix(i,j)=(n+1-max(i,j))*m+massUAS;
        massMatrix(i,j)=sum(massVector(max(i,j):end));

    end
end
```

The function “ndimrigid.m” defines the equations of motion completely including the coefficient matrices, etc.

```
%rigid

function [dQ,FORCEWORLD] = ndimrigid(t,Q);
global segLength
global lengthVector
global massVector
global n
global k
global massMatrix
global windVelocity
global airDensity
global segArea
global Cf
global Cd
global G
global Jp
global inertias
global g
global TX
global TY
global TZ
global UASTHRUST
global MaxU
global MaxV
global MaxW
global UTrig
global VTrig
global WTrig
global UU BETA jk ts
% Wind VTOL
PsiRate=Q(1:n+1);ThetaRate=Q(n+2:2*n+2);PhiRate=Q(2*n+3:3*n+3);
Psi=Q(3*n+4:4*n+4);Theta=Q(4*n+5:5*n+5);Phi=Q(5*n+6:6*n+6);
% UASTHRUST =283.5936;%+10.*(t>20).*(t-20);
```

```
UASTHRUST =283.5936;%+10.*(t>5).*(t-5);

Uinf=interp1(ts,UU,t);
beta=interp1(ts,BETA,t);


U=Uinf*cosd(beta);%+MaxU./10*sin(0.5./50*t*t);%0.2*sin(2*pi/.05*t)*(hea
viside(t-UTrig));
V=Uinf*sind(beta);%*(heaviside(t-VTrig));
W=MaxW;%*(heaviside(t-WTrig));
windVelocity(1:3:end,1)=U;
windVelocity(2:3:end,1)=V;
windVelocity(3:3:end,1)=W;

temp=0;
velocity=zeros(3*n+3,1);
% windVelocity=zeros(2*n+2,1);
rotationBigMatrix=[0];
forceSegment=[0];
% for i=1:n+1;
% end
%%
dQ=zeros(k,1);
moments=zeros(n+1,1); % External Moments Vector
%%
A=[];B=[];C=[];
A11=zeros(n+1,n+1);A12=zeros(n+1,n+1);A13=zeros(n+1,n+1);A21=zeros(n+1,
n+1);A22=zeros(n+1,n+1);A23=zeros(n+1,n+1);A31=zeros(n+1,n+1);A32=zeros
(n+1,n+1);A33=zeros(n+1,n+1);
B11=zeros(n+1,n+1);B12=zeros(n+1,n+1);B13=zeros(n+1,n+1);B21=zeros(n+1,
n+1);B22=zeros(n+1,n+1);B23=zeros(n+1,n+1);B31=zeros(n+1,n+1);B32=zeros
(n+1,n+1);B33=zeros(n+1,n+1);
C11=zeros(n+1,n+1);C12=zeros(n+1,n+1);C13=zeros(n+1,n+1);C21=zeros(n+1,
n+1);C22=zeros(n+1,n+1);C23=zeros(n+1,n+1);C31=zeros(n+1,n+1);C32=zeros
(n+1,n+1);C33=zeros(n+1,n+1);
A11rot=zeros(n+1,n+1);A12rot=zeros(n+1,n+1);A13rot=zeros(n+1,n+1);A21ro
t=zeros(n+1,n+1);A22rot=zeros(n+1,n+1);A23rot=zeros(n+1,n+1);A31rot=zer
os(n+1,n+1);A32rot=zeros(n+1,n+1);A33rot=zeros(n+1,n+1);
B11rot=zeros(n+1,n+1);B12rot=zeros(n+1,n+1);B13rot=zeros(n+1,n+1);B21ro
t=zeros(n+1,n+1);B22rot=zeros(n+1,n+1);B23rot=zeros(n+1,n+1);B31rot=zer
os(n+1,n+1);B32rot=zeros(n+1,n+1);B33rot=zeros(n+1,n+1);
C11rot=zeros(n+1,n+1);C12rot=zeros(n+1,n+1);C13rot=zeros(n+1,n+1);C21ro
t=zeros(n+1,n+1);C22rot=zeros(n+1,n+1);C23rot=zeros(n+1,n+1);C31rot=zer
os(n+1,n+1);C32rot=zeros(n+1,n+1);C33rot=zeros(n+1,n+1);
AA=[];BB=[];CC=[];
%%

for i=1:n+1;
    for j=1:n+1;
        %          for i1=1:n+1;
        %

A11(i,j)=massMatrix(i,j)*lengthVector(i)*lengthVector(j)*(cos(Theta(i))
*cos(Theta(j))*cos(Psi(i)-
Psi(j)));%+(i==j)*inertias(i);%+(i==j)*(sum(AA(i:end,1))+inertias(i)+1/
4*massVector(i)*lengthVector(i).^2);

A12(i,j)=massMatrix(i,j)*lengthVector(i)*lengthVector(j)*(cos(Theta(i))
*sin(Theta(j))*sin(Psi(i)-Psi(j)));
        A21(i,j)=massMatrix(i,j)*lengthVector(i)*lengthVector(j)*(-
sin(Theta(i))*cos(Theta(j))*sin(Psi(i)-Psi(j)));

A22(i,j)=massMatrix(i,j)*lengthVector(i)*lengthVector(j)*(sin(Theta(i))
*sin(Theta(j))*cos(Psi(i)-Psi(j))+cos(Theta(i))*cos(Theta(j)));


A11rot(i,i)=inertias(i,1)*(sin(Theta(i))).^2+(inertias(i,2)*(sin(Phi(i)
)).^2+inertias(i,3)*(cos(Phi(i))).^2)*(cos(Theta(i))).^2;
        A12rot(i,i)=(inertias(i,2)-
inertias(i,3))*cos(Phi(i))*sin(Phi(i))*cos(Theta(i));
        A21rot=A12rot;

A22rot(i,i)=(inertias(i,2)*(cos(Phi(i))).^2+inertias(i,3)*(sin(Phi(i)))
.^2);
        A13rot(i,i)=(-inertias(i,1)*sin(Theta(i)));
        A31rot=A13rot;
        A23rot(i,i)=0;
        A32rot=A23rot;
        A33rot(i,i)=inertias(i,1);
    end
end
A=[A11+A11rot A12+A12rot A13+A13rot;A21+A21rot A22+A22rot
A23+A23rot;A31+A31rot A32+A32rot A33+A33rot];

for i=1:n+1;
    for j=1:n+1;
        %         for i1=1:n+1;
        %

B11(i,j)=massMatrix(i,j)*lengthVector(i)*lengthVector(j)*(cos(Theta(i))
*cos(Theta(j))*sin(Psi(i)-
Psi(j)));%+(i==j)*inertias(i);%+(i==j)*(sum(AA(i:end,1))+inertias(i)+1/
4*massVector(i)*lengthVector(i).^2);

B12(i,j)=massMatrix(i,j)*lengthVector(i)*lengthVector(j)*(cos(Theta(i))
*cos(Theta(j))*sin(Psi(i)-Psi(j)));

B21(i,j)=massMatrix(i,j)*lengthVector(i)*lengthVector(j)*(sin(Theta(i))
*cos(Theta(j))*cos(Psi(i)-Psi(j)));

B22(i,j)=massMatrix(i,j)*lengthVector(i)*lengthVector(j)*(sin(Theta(i))
*cos(Theta(j))*cos(Psi(i)-Psi(j))-cos(Theta(i))*sin(Theta(j)));

        B11rot(i,i)=0;
        B13rot=B11rot;
        B22rot=B11rot;
        B23rot=B11rot;
        B33rot=B11rot;
        B12rot(i,i)=(inertias(i,3)-
inertias(i,2))*cos(Phi(i))*sin(Phi(i))*sin(Theta(i));
        B21rot(i,i)=(-
inertias(i,1)+inertias(i,2)*(sin(Phi(i))).^2+inertias(i,3)*(cos(Phi(i))
).^2)*sin(Theta(i))*cos(Theta(i));
        B31rot(i,i)=(inertias(i,3)-
inertias(i,2))*cos(Phi(i))*sin(Phi(i))*(cos(Theta(i))).^2;
        B32rot(i,i)=(inertias(i,2)-
inertias(i,3))*cos(Phi(i))*sin(Phi(i));



    end
end
B=[B11+B11rot B12+B12rot B13+B13rot;B21+B21rot B22+B22rot
B23+B23rot;B31+B31rot B32+B32rot B33+B33rot];


for i=1:n+1;
    for j=1:n+1;

        C11(i,j)=massMatrix(i,j)*lengthVector(i)*lengthVector(j)*(-
2*cos(Theta(i))*sin(Theta(j))*cos(Psi(i)-
Psi(j)));%+(i==j)*inertias(i);%+(i==j)*(sum(AA(i:end,1))+inertias(i)+1/
4*massVector(i)*lengthVector(i).^2);

C21(i,j)=massMatrix(i,j)*lengthVector(i)*lengthVector(j)*(2*sin(Theta(i
))*sin(Theta(j))*sin(Psi(i)-Psi(j)));

        C11rot(i,i)=2*cos(Theta(i))*sin(Theta(i))*(inertias(i,1)-
inertias(i,2)*(sin(Phi(i))).^2-inertias(i,3)*(cos(Phi(i))).^2);
        C12rot(i,i)=cos(Theta(i))*(-inertias(i,1)+(inertias(i,2)-
inertias(i,3))*((cos(Phi(i))).^2-(sin(Phi(i))).^2));

C13rot(i,i)=2*(cos(Theta(i))).^2*sin(Phi(i))*cos(Phi(i))*(inertias(i,2)
-inertias(i,3));
        C22rot(i,i)=2*sin(Phi(i))*cos(Phi(i))*(inertias(i,3)-
inertias(i,2));
%         C22rot(i,i)=2*sin(Phi(i))*cos(Phi(i))*(inertias(i,3)-
inertias(i,2));
        C21rot(i,i)=0;
        C32rot=C21rot;
        C33rot=C21rot;
        C23rot(i,i)=cos(Theta(i))*(inertias(i,1)+(inertias(i,2)-
inertias(i,3))*((cos(Phi(i))).^2-(sin(Phi(i))).^2));
        C31rot(i,i)=cos(Theta(i))*(-inertias(i,1)+(inertias(i,3)-
inertias(i,2))*((cos(Phi(i))).^2-(sin(Phi(i))).^2));


    end
end
```

```
C=[C11+C11rot C12+C12rot C13+C13rot;C21+C21rot C22+C22rot
C23+C23rot;C31+C31rot C32+C32rot C33+C33rot];


% newA=[-inv(A)*B*diag(Q(1:n+1))  zeros(n+1);
newA=[-A\(B*diag(Q(1:3*n+3)))  zeros(3*n+3);

eye(3*n+3)                  zeros(3*n+3)];

%% Rotation Matrix for Velocity

for i=1:n+1;
    temp=[0;0;0];
    for j=1:i;
        RDotThet=[-sin(Psi(j))*cos(Theta(j))  -
cos(Psi(j))*sin(Theta(j)) 0;
            cos(Psi(j))*cos(Theta(j))  -sin(Psi(j))*sin(Theta(j)) 0;
            0                       -cos(Theta(j))                0];

        velocity(3*i-2:3*i,1)=
temp+(RDotThet*[PsiRate(j);ThetaRate(j);PhiRate(j)])*lengthVector(j);
        temp=velocity(3*i-2:3*i,1);
    end
    %     velocity(2*i-1:2*i,1)=velocity(2*i-
1:2*i,1)+0.5*lengthVector(i).*[Q(i)*sin(Q(n+1+i));-Q(i)*cos(Q(n+1+i))];
end


relativeVelocity=windVelocity-velocity;


for i=1:n+1;
    rotationBigMatrix(3*i-2:3*i,3*i-2:3*i)=[cos(Psi(i))*cos(Theta(i))
sin(Psi(i))*cos(Theta(i))                                              -
sin(Theta(i));...
                              -
sin(Psi(i))*cos(Phi(i))+cos(Psi(i))*sin(Theta(i))*sin(Phi(i))
cos(Psi(i))*cos(Phi(i))+sin(Psi(i))*sin(Theta(i))*sin(Phi(i))
sin(Phi(i))*cos(Theta(i));  ...

sin(Psi(i))*sin(Phi(i))+cos(Psi(i))*sin(Theta(i))*cos(Phi(i))
-cos(Psi(i))*sin(Phi(i))+sin(Psi(i))*sin(Theta(i))*cos(Phi(i))
cos(Phi(i))*cos(Theta(i))];

end

% relVelSegment=inv(rotationBigMatrix)*relativeVelocity;
relVelSegment=(rotationBigMatrix)\relativeVelocity;

% Wind Forces on Cable
for i = 1:n ;
```

```
    Vall=sqrt(relVelSegment(3*i-2,1).^2+relVelSegment(3*i-
1,1).^2+relVelSegment(3*i,1).^2);
    forceSegment(3*i-
2,1)=0.5*airDensity*segArea(i)*3.14*Cf*relVelSegment(3*i-2,1)*Vall;
    forceSegment(3*i-1,1)=0.5*airDensity*segArea(i)*relVelSegment(3*i-
1,1)*(Cf*3.14*Vall+Cd*sqrt(relVelSegment(3*i-
1,1).^2+relVelSegment(3*i,1).^2));

forceSegment(3*i,1)=0.5*airDensity*segArea(i)*relVelSegment(3*i,1)*(Cf*
3.14*Vall+Cd*sqrt(relVelSegment(3*i-1,1).^2+relVelSegment(3*i,1).^2));
end

%% Wind forces on UAS
for i = n+1;
    Valluas=sqrt(relVelSegment(3*i-2,1).^2+relVelSegment(3*i-
1,1).^2+relVelSegment(3*i,1).^2);
    forceSegment(3*i-2,1)=0.5*airDensity*segArea(i)*relVelSegment(3*i-
2,1)*(3.14*Cf*Valluas+2*Cd*abs(relVelSegment(3*i-2,1)));
    forceSegment(3*i-1,1)=0.5*airDensity*segArea(i)*relVelSegment(3*i-
1,1)*(3.14*Cf*Valluas+Cd*sqrt(relVelSegment(3*i-
1,1).^2+relVelSegment(3*i,1).^2)+Cd*abs(relVelSegment(3*i-1,1)));

forceSegment(3*i,1)=0.5*airDensity*segArea(i)*relVelSegment(3*i,1)*(3.1
4*Cf*Valluas+Cd*sqrt(relVelSegment(3*i-
1,1).^2+relVelSegment(3*i,1).^2)+Cd*abs(relVelSegment(3*i,1)));
end

forceWld=rotationBigMatrix*forceSegment;
%

% forceWld(3*n-2) =30*(heaviside(t-3));
forceWld(3*n-2) = 1*forceWld(3*n-
2)+UASTHRUST*(cos(Psi(end))*cos(Theta(end)));
forceWld(3*n-1) = 1*forceWld(3*n-
1)+UASTHRUST*(sin(Psi(end))*cos(Theta(end)));
forceWld(3*n+3) = 1*forceWld(3*n+3)-UASTHRUST*sin(Theta(end));



%%Impact Force
%
forceWld(3*round((n+1)/4)+1)=forceWld(3*round((n+1)/4)+1)+40.*(heavisid
e(t-60)&heaviside(61-t));
%%
% forceWld(2*n+1)=5*(heaviside(t-5)-heaviside(t-5.1));
FORCEWORLD=forceWld;
%% generalized force (Q) in equations
generalizedForce=zeros(3*n+3,1);
generalizedForcePsi=zeros(n+1,1);
generalizedForceTheta=zeros(n+1,1);
generalizedForcePhi=zeros(n+1,1);

generalizedForcei=[];
generalizedForceiPsi=[];
generalizedForceiTheta=[];
```

```
generalizedForceiPhi=[];
moments(1)=0;
moments(n+1)=0;%forceWld(3*n+1)*0.5*cos(Theta(n+1));
%
for i=1;
    %     moments(i)=forceWld(2*i-1)*.5*lengthVector(i)*sin(Q(n+1+i))-
forceWld(2*i)*.5*lengthVector(i)*cos(Q(n+1+i));
    tempqPsi=[0];
    tempqTheta=[0];
    tempqPhi=[0];
    for j=i:n+1;

        %        end
        generalizedForceiPsi(i,1)=tempqPsi+forceWld(3*j-2)*(-
lengthVector(i)*sin(Psi(i))*cos(Theta(i)))+forceWld(3*j-
1)*lengthVector(i)*cos(Psi(i))*cos(Theta(i));%-
lengthVector(i)/2*cos(Q(n+1+i));%+moments(i);
        generalizedForceiTheta(i,1)=tempqTheta+forceWld(3*j-2)*(-
lengthVector(i)*cos(Psi(i))*sin(Theta(i)))+forceWld(3*j-1)*(-
lengthVector(i)*sin(Psi(i))*sin(Theta(i)))+(forceWld(3*j)-
massVector(j)*g)*(-lengthVector(i)*cos(Theta(i)));%-
lengthVector(i)/2*cos(Q(n+1+i));%+moments(i);
        tempqPsi=generalizedForceiPsi(i,1);
        tempqTheta=generalizedForceiTheta(i,1);
%         tempqPhi=generalizedForceiPhi(i,1);
    end
            generalizedForceiPhi(i,1)=-
1*G*(Jp).*(Phi(i)./lengthVector(i)+(Phi(i)-
Phi(i+1))./lengthVector(i+1))+moments(i);%-
lengthVector(i)/2*cos(Q(n+1+i));%+moments(i);

end
%}
for i=n+1;

    tempqPsi=[0];
    tempqTheta=[0];
    tempqPhi=[0];
    for j=i:n+1;
        %
        generalizedForceiPsi(i,1)=tempqPsi+forceWld(3*j-2)*(-
lengthVector(i)*sin(Psi(i))*cos(Theta(i)))+forceWld(3*j-
1)*lengthVector(i)*cos(Psi(i))*cos(Theta(i));%-
lengthVector(i)/2*cos(Q(n+1+i));%+moments(i);
        generalizedForceiTheta(i,1)=tempqTheta+forceWld(3*j-2)*(-
lengthVector(i)*cos(Psi(i))*sin(Theta(i)))+forceWld(3*j-1)*(-
lengthVector(i)*sin(Psi(i))*sin(Theta(i)))+(forceWld(3*j)-
massVector(j)*g)*(-lengthVector(i)*cos(Theta(i)));%-
lengthVector(i)/2*cos(Q(n+1+i));%+moments(i);
        tempqPsi=generalizedForceiPsi(i,1);
        tempqTheta=generalizedForceiTheta(i,1);
%         tempqPhi=generalizedForceiPhi(i,1);
    end
```

```
            generalizedForceiPhi(i,1)=-1*G*(Jp)./lengthVector(i-
1).*(Phi(i)-Phi(i-1))+moments(i);%-

end
%}

for i=2:n;

    tempqPsi=[0];
    tempqTheta=[0];
    tempqPhi=[0];
    for j=i:n+1;

        generalizedForceiPsi(i,1)=tempqPsi+forceWld(3*j-2)*(-
lengthVector(i)*sin(Psi(i))*cos(Theta(i)))+forceWld(3*j-
1)*lengthVector(i)*cos(Psi(i))*cos(Theta(i));%-
lengthVector(i)/2*cos(Q(n+1+i));%+moments(i);
        generalizedForceiTheta(i,1)=tempqTheta+forceWld(3*j-2)*(-
lengthVector(i)*cos(Psi(i))*sin(Theta(i)))+forceWld(3*j-1)*(-
lengthVector(i)*sin(Psi(i))*sin(Theta(i)))+(forceWld(3*j)-
massVector(j)*g)*(-lengthVector(i)*cos(Theta(i)));%-
lengthVector(i)/2*cos(Q(n+1+i));%+moments(i);
        tempqPsi=generalizedForceiPsi(i,1);
        tempqTheta=generalizedForceiTheta(i,1);
%          tempqPhi=generalizedForceiPhi(i,1);
    end
            generalizedForceiPhi(i,1)=-1*G*(Jp).*((Phi(i)-Phi(i-
1))./lengthVector(i)+(Phi(i)-Phi(i+1))./lengthVector(i))+moments(i);

end
generalizedForce=[generalizedForceiPsi;generalizedForceiTheta;generaliz
edForceiPhi];

generalizedForce;


t

dQ=newA*Q+[A\generalizedForce; zeros(3*n+3,1)]+[-
A\(C*diag([ThetaRate;PhiRate;PsiRate])) zeros(3*n+3);zeros(3*n+3)
zeros(3*n+3)]*[PsiRate;ThetaRate;PhiRate;zeros(3*n+3,1)];
```

Script “solveSystem.m” solves the equations of motion given the initial conditions defined in the same file in the form of segment angles.

```
%%solveSystem
```

```
tic

simParams

Psi0=ones(1,n+1)*(0)*pi/180;Theta0=ones(1,n+1)*(-
89)*pi/180;Phi0=ones(1,n+1)*(0)*pi/180;
PsiRate0=ones(1,n+1)*0;ThetaRate0=ones(1,n+1)*0;PhiRate0=ones(1,n+1)*0;
Theta0(end)=(-89)*pi/180;
% Psi0(end)=0*pi/2;
thet0=[PsiRate0 ThetaRate0 PhiRate0 Psi0 Theta0 Phi0]';
options = odeset('RelTol',1e-2*eye(1,size(thet0,2)),'AbsTol',1e-
3*eye(1,size(thet0,2)));
global segLength
global m
global lengthVector
global massVector
global n

% timeStep=.5;
dimTol=0;

timeEnd =20;
[T,SOLUT] = ode23(@ndimrigid,[0 timeEnd],thet0,options);
solvingTime=toc/60
disp('minutes')
%
plotSystem
forceCalc
save(['workspace_',num2str(MaxU),'_',num2str(MaxV),'_',num2str(MaxW),'_
',num2str(round(UASTHRUST))])
% plottest
```

Script “plotSystem.m” calculates and plots the results in terms of acceleration, position and velocities.

```
%% plot system

tic

global segLength
global lengthVector
global massVector
global n
global massMatrix
global m
global k
global windVelocity
global airDensity
global segArea
global Cf
global Cd
```

```
samplingRate=2*(round(size(T,1)/500)+1);
% timeStep=.5;
limlim = (n+1)*segLength*3;


%Plotting the columns of the returned array Y versus T shows the
solution
XX=[];
YY=[];
ZZ=[];
X=[];
Y=[];
Z=[];
% TT=T(1:10:end,:);
% YY=Y(1:10:end,:);
TT=T(1:samplingRate:end,:);
SOLUTSAMP=SOLUT(1:samplingRate:end,:);
PsiRate=[];ThetaRate=[];PhiRate=[];Psi=[];Theta=[];Phi=[];
PsiRate=SOLUTSAMP(:,1:n+1);ThetaRate=SOLUTSAMP(:,n+2:2*n+2);PhiRate=SOL
UTSAMP(:,2*n+3:3*n+3);
Psi=SOLUTSAMP(:,3*n+4:4*n+4);Theta=SOLUTSAMP(:,4*n+5:5*n+5);Phi=SOLUTSA
MP(:,5*n+6:6*n+6);

accel = 0*(SOLUTSAMP);
FORCEWORLD=zeros(size(SOLUTSAMP,1),3*(n+1));
for j=1:size(TT);
    X(1,j)=lengthVector(1)*cos(Psi(j,1))*cos(Theta(j,1));%-
lengthVector(1)/2*cos(YY(j,n+1+1));
    Y(1,j)=lengthVector(1)*sin(Psi(j,1))*cos(Theta(j,1));%-
lengthVector(1)/2*sin(YY(j,n+1+1));
    Z(1,j)=lengthVector(1)*(-sin(Theta(j,1)));%-
lengthVector(1)/2*sin(YY(j,n+1+1));
    for i=2:n+1;
        X(i,j)=X(i-1,j)+lengthVector(i)*cos(Psi(j,i))*cos(Theta(j,i));
        Y(i,j)=Y(i-1,j)+lengthVector(i)*sin(Psi(j,i))*cos(Theta(j,i));
        Z(i,j)=Z(i-1,j)+lengthVector(i)*(-sin(Theta(j,i)));
%
%         sumx=X(i);
%         sumz=Z(i);
    end

    %         figure (2)

    filename = 'testnew51.gif';

[accel(j,:),FORCEWORLD(j,:)]=ndimrigid(TT(j), SOLUTSAMP(j,:)');
Psi2dot=accel(:,1:n+1);
Theta2dot=accel(:,n+2:2*n+2);
Phi2dot=accel(:,2*n+3:3*n+3);


%%
 figure(2221)
 subplot(321)
 plot(TT(:,1),X(end,:),'-')
 xlabel('Time (sec)')
 ylabel('UAS Offset (m)')
 hold on
 grid minor
 title('UAS Position')

 subplot(323)
 plot(TT(:,1),Z(end,:),'-')
 xlabel('Time (sec)')
 ylabel('UAS Height (m)')
 hold on
 grid minor

 subplot(325)

 plot(TT(:,1),Y(end,:),'-')
 xlabel('Time (sec)')
 ylabel('UAS Sway (m)')
 hold on
 grid minor
% legend('WindSpeed=')
   print('longitudinal','-dmeta')
   print('longitudinal','-dpng')
   saveas(gcf,'longitudinal.fig')
    %}

%   figure(2222)
  subplot(322)
   pitch=Theta(:,end);
 pitch=pitch*180/pi;
 plot(TT(:,1),pitch,'-')
 xlabel('Time (sec)')
 ylabel('UAS Pitch (deg)')
 hold on
 grid minor
  title('UAS Attitude')


subplot(324)

 heading=Phi(:,end);
 heading=heading*180/pi;
 plot(TT(:,1),heading,'-')
   xlabel('Time (sec)')
 ylabel('UAS Heading (deg)')
 hold on
 grid minor
 subplot(326)
 bank=-Psi(:,end);
 bank=bank*180/pi;
 plot(TT(:,1),bank,'-')
 xlabel('Time (sec)')
 ylabel('UAS Bank (deg)')
 hold on
 grid minor


% legend('WindSpeed=')
   print('lateral','-dmeta')
   print('lateral','-dpng')
   saveas(gcf,'lateral.fig')
    %}


    figure(22221)
    subplot(221)
    plot(X(end,:),Y(end,:),'-')
text(X(end,1),Y(end,1),sprintf('t=%.2f (sec)',TT(1)))
 xlabel('X (m)')
 ylabel('Y (m)')
 hold on
 grid minor

 subplot(222)
 plot(X(end,:),Z(end,:),'-')
 text(X(end,1),Z(end,1),sprintf('t=%.2f (sec)',TT(1)))
 xlabel('X (m)')
 ylabel('Z (m)')
 hold on
 grid minor

 subplot(223)
 plot(Y(end,:),Z(end,:),'-')
  text(Y(end,1),Z(end,1),sprintf('t=%.2f (sec)',TT(1)))
 xlabel('Y (m)')
 ylabel('Z (m)')
 hold on
 grid minor

 subplot(224)
 plot(TT(:,1),Z(end,:),'-')
 hold on
 plot(TT(:,1),X(end,:),'-')
 hold on
 plot(TT(:,1),Y(end,:),'-')
 legend(['Height'], ['Offset'], ['Sway'])
  hold on
 grid minor
 xlabel('T (sec)')
 ylabel('Z,X,Y (m)')
```

```
    print('all','-dmeta')
   print('all','-dpng')
   saveas(gcf,'all.fig')


plottingTime=toc/60
disp('minutes')
```

The script “forceCalc.m” calculates and plots the forces in the cable against time.

```
%% forces
global g
global massVector
global lengthVector
global TX
global TY
global TZ
global UASTHRUST


interForce=0*SOLUTSAMP;
baseForce=0*SOLUTSAMP(:,1:2);
linAccel=0*SOLUTSAMP;

for jj=1:size(TT);


for i=1:n+1;
    tempa=[0;0;0];
      for j=1:i;
          RDotDot=[-sin(Psi(jj,j))*cos(Theta(jj,j))   -
cos(Psi(jj,j))*sin(Theta(jj,j))      0;
                    cos(Psi(jj,j))*cos(Theta(jj,j))   -
sin(Psi(jj,j))*sin(Theta(jj,j))      0;
                       0                              -cos(Theta(jj,j))
0];

          RDot2=[-cos(Psi(jj,j))*cos(Theta(jj,j))   -
cos(Psi(jj,j))*cos(Theta(jj,j))      0;
                  -sin(Psi(jj,j))*cos(Theta(jj,j))   -
sin(Psi(jj,j))*cos(Theta(jj,j))      0;
                       0                             sin(Theta(jj,j))
0];

          RDotRDot=[2*sin(Psi(jj,j))*sin(Theta(jj,j))  0  0;
                   -2*cos(Psi(jj,j))*sin(Theta(jj,j))  0  0;
                       0                               0  0];

          linAccel(jj,3*i-2:3*i)=
tempa+RDotDot*[Psi2dot(jj,j)';Theta2dot(jj,j)';Phi2dot(jj,j)']+...
```

```
RDot2*diag([PsiRate(jj,j)',ThetaRate(jj,j)',PhiRate(jj,j)'])*[PsiRate(j
j,j)';ThetaRate(jj,j)';PhiRate(jj,j)']+...

RDotRDot*[PsiRate(jj,j)'.*ThetaRate(jj,j)';PhiRate(jj,j)'.*ThetaRate(jj
,j)';PsiRate(jj,j)'.*PhiRate(jj,j)'];
          tempa=linAccel(jj,3*i-2:3*i);
          tempa=tempa';
%           tempa'

      end
%
end


hold on
%% Force Calculations


TX = UASTHRUST*(cos(Psi(jj,end))*cos(Theta(jj,end)));
TY = UASTHRUST*(sin(Psi(jj,end))*cos(Theta(jj,end)));
TZ = -UASTHRUST*sin(Theta(jj,end));



FORCEWORLD;

for i1 = n+1
% interForce(jj,3*i1-2:3*i1)=[TX TY TZ]+FORCEWORLD(jj,3*i1-2:3*i1);%-
massVector(i1-1)*[0 g];
interForce(jj,3*i1-2:3*i1)=FORCEWORLD(jj,3*i1-2:3*i1);%-massVector(i1-
1)*[0 g];
end
for i2=n:-1:1;
interForce(jj,3*i2-2:3*i2)=FORCEWORLD(jj,3*i2-
2:3*i2)+[interForce(jj,3*i2+1:3*i2+3)]-
massVector(i2)*[linAccel(jj,3*i2-2) linAccel(jj,3*i2-1)
linAccel(jj,3*i2)+g];%-massVector(i2-1)*[0 g];
end
for i3=1;
interForce(jj,3*i3-2:3*i3)=FORCEWORLD(jj,3*i3-
2:3*i3)+[interForce(jj,3*i3+1:3*i3+3)]-
massVector(i3)*[linAccel(jj,3*i3-2) linAccel(jj,3*i2-1)
linAccel(jj,3*i3)+g];
end

baseForce(jj,1:3)=[interForce(jj,4:6)]-massVector(1)*[linAccel(jj,1)
linAccel(jj,2) linAccel(jj,3)+g];




end
```

```
figure (5)
subplot(311)
plot(TT(:),linAccel(:,1:3:end),'-')
xlabel('Time (sec)')
ylabel('Node X Accelerations (m/s^2)')
grid minor
title('Node Accelarations')

subplot(312)
plot(TT(:),linAccel(:,2:3:end),'-')
xlabel('Time (sec)')
ylabel('Node Y Accelerations (m/s^2)')
grid minor

subplot(313)
plot(TT(:),linAccel(:,3:3:end),'-')
xlabel('Time (sec)')
ylabel('Node Vertical(Z) Accelerations (m/s^2)')
grid minor

 print('Node Accel','-dmeta')
   print('Node Accel','-dpng')
   saveas(gcf,'Node Accel.fig')

figure (6)
subplot(311)
plot(TT(:),interForce(:,1:3:end))
xlabel('Time (sec)')
ylabel('Node X Forces (N)')
grid minor
title('Node Forces')

subplot(312)
plot(TT(:),interForce(:,2:3:end))
xlabel('Time (sec)')
ylabel('Node Y Forces (N)')
grid minor

subplot(313)
plot(TT(:),interForce(:,3:3:end))
xlabel('Time (sec)')
ylabel('Node Vertical(Z) Forces (N)')
grid minor

 print('Node Forces','-dmeta')
   print('Node Forces','-dpng')
   saveas(gcf,'Node Forces.fig')


figure (7)
subplot(311)
plot(TT(:),baseForce(:,1))
```

```
xlabel('Time (sec)')
ylabel('Base X Forces (N)')
grid minor
title('Base Forces')

subplot(312)
plot(TT(:),baseForce(:,2))
xlabel('Time (sec)')
ylabel('Base Y Forces (N)')
grid minor

subplot(313)
plot(TT(:),baseForce(:,3))
xlabel('Time (sec)')
ylabel('Base Vertical(Z) Forces (N)')
grid minor

 print('Base Forces','-dmeta')
   print('Base Forces','-dpng')
   saveas(gcf,'Base Forces.fig')

tension1=[];
tension=[];
for j4=1:n+1;
tension1(j4,1)=norm(interForce(end,3*j4-2:3*j4));
end
tension=[norm(baseForce(end,:));...
    tension1];

figure(8)
Xs=[zeros(1,size(X,2)); X(:,:)];
plot(Xs(:,end),tension,'>')
xlabel('Node X-Position (m)')
ylabel('Longitudinal Cable Tension(N)')
grid minor
title('Cable Tesnion Comparison')
legend('Discrete 3D Model')
hold on

print('Cable Tesnion','-dmeta')
   print('Cable Tesnion','-dpng')
   saveas(gcf,'Cable Tesnion.fig')
```