DEVELOPMENT OF OPTIMIZED PATH PLANNING AND AUTONOMOUS CONTROL FOR RETURN-TO-POINT VEHICLE OF HIGH ALTITUDE BALLOONING

By

SEONG-JIN LEE

Bachelor of Science in Mechanical &Aerospace engineering
Korea Aerospace University
Gyunggi, South Korea
2002

Master of Engineering in Mechanical &Aerospace engineering
Cornell University
Ithaca, New York
2007

Submitted to the Faculty of the
GraduateCollege of the
OklahomaStateUniversity
in partial fulfillment of
the requirements for
the Degree of
DOCTOR OF PHILOSOPHY
July, 2014

DEVELOPMENT OF OPTIMIZED PATH PLANNING AND AUTONOMOUS CONTROL FOR RETURN-TO-POINT VEHICLE OF HIGH ALTITUDE BALLOONING

Dissertation Approved:

Dr. A.S. Arena

Dissertation Adviser

Dr. J. Jacob

Dr. G. Young

Dr. R. Rhinehart

Dr. J. Conner

Acknowledgements

I would like to thank Dr. Andrew. S. Arena, for giving me the opportunity to work on this project. His guidance and support helped make this project an amazing learning experience for me. I am especially grateful to Dr. Joseph Conner Jr. Without his support, this project would not have been possible. I also wish thank to Dr. AfshinJ.Ghajar for giving me many chance to work as teaching assistant, and to Dr. Gary E.Young, Dr. Jamey D. Jacob, and Dr. Russell Rhinehart for their precious advice and support in serving as committee members.

With love, my lovely wife Nada and my adorable babies Grace Haeun and Kate Sieun.

"Acknowledgements reflect the views of the author and are not endorsed by committee members or Oklahoma State University."

Name: SEONG-JIN LEE

Date of Degree: JULY, 2014

Title of Study: DEVELOPMENT OF OPTIMIZED PATH PLANNING AND AUTONOMOUS CONTROL FOR RETURN-TO-POINT VEHICLE OF HIGH ALTITUDE BALLOONING

Major Field: MECHANICAL & AEROSPACE ENGINEERING

Abstract:

In 2004, The Atmospheric and Space Threshold Research Oklahoma (ASTRO) program was launched to provide access to the near space environment for both educational and research purposes. Mainly, this ASTRO vehicle consisted of four parts: Sounding weather balloon to produce buoy force during the ascent phase, circular parachute to produce drag force during the descent phase, tracking gear with GPS (Global Positioning System) to check the position from the ground, and experimental payloads. The descent phase utilizes a circular parachute, and as such, there are no means of controlling the landing location of the vehicle and payloads. Without control, the direction the parachute takes is dependent upon the winds aloft which can allow payloads to land in undesirable locations, such as rivers, lakes, or the middle of vast forests. At times, the flight must be cancelled before it even begins if the risks of a long or difficult recovery are predicted.As the ASTRO project has evolved, the necessity of control of the payloads over the descent phase has also become obvious. In order to address this need, a study of a Return-to-Point Vehicle (RPV) has been started.

Parafoil vehicles which are used for RPVs have had demonstrated usefulness in many situations. Once the RPV has reached the desired altitude, generallyaround100,000 ft, it is released from the balloon. When the RPV has been released, it will follow a trajectory which is programmed on Autopilot to direct it a desired landing zone. Some researchers tried to analyze and test the parafoil, but there are no reported uses of the parafoil for dropping the payload from high altitudes; in addition, there are not commercial products to be matched with the ASTRO research as well. Therefore, thepurpose of this research is to design the RPV and to develop the optimal trajectory to satisfy the requirements of the ASTRO project.

In this research, the most important objective is to develop a cost-effective, simple, and reliable autopilot system which can be applied to the payload used in the ASTRO project.

TABLE OF CONTENTS

Chapter Page

Chapter Page

Chapter Page

LIST OF TABLES

Table Page

LIST OF FIGURES

Figure Page

Figure Page

Figure Page

Figure Page

NOMENCLATURE

$\alpha, \alpha_{L=0}$	Angle of attack, angle of attack at which $C_L = 0$
β	Side-slip angle
δ_f, δ_a	Asymmetric flap deflection
Γ	Anhedral angle
γ	Glide slope angle
ω	Angular velocity vector
ρ	Air density
θ_r	Rigging angle
a	Lift curve slope
AR	Aspect ratio
$A_{0,R}, A_{0,A}$	Mass contributions to 6x6 inertia matrix
a_{po}	Distance from pitch center to origin
a_{ro}	Distance from roll center to origin
a_{pc}	Distance from the pitch center to the confluence point
a_{rc}	Distance from the roll center to the confluence point
b	Parafoil span
c	Chord length
C_D	Parafoil overall drag coefficients
$C_{D,p}, C_{D,\delta f}$	Drag stability derivatives
$C_M, C_{Mq}, C_{Mc/4}$	Moment coefficients
C_L	Parafoil lift coefficient
$C_Y, C_{Y\beta}, C_{\delta a}$	Side force contribution due to side-slip angle
$C_l, C_{lp}, C_{l\delta a}, C_{lr}, C_{l\beta}$	Roll moment stability derivatives
$C_n, C_{n\delta a}, C_{np}, C_{nr}, C_{n\beta}$	Yaw moment stability derivatives

h_0	Angular momentum
I_{11}, I_{22}, I_{33}	Rotational inertia about roll, pitch, yaw centers
I_{f11}	Mass constant
$J_{o,R}$	Rotational matrix about 0
k	Small aspect ratio correction factor
k_1	Nonlinear low aspect ratio correction
L	Overall lift
M_R	Real mass
p, q, r	Roll, pitch, yaw rate
P	Linear momentum
R_{CG}	Distance between the system CG and the canopy CG
R_{SG}	Distance between the payload CG and the system CG
S	Parafoil canopy surface area
v_0	Velocity vector at point 0
W	Parafoil system weight

CHAPTER I

INTRODUCTION

In 2004, The Atmospheric and Space Threshold Research Oklahoma (ASTRO) program was launched to provide access to the near space environment for both educational and research purposes. The purpose of the ASTRO program is collecting the temperature, pressure, and radioactivity data, and also taking pictures from the space with helium filled sounding balloons at high altitudes. Mainly, this ASTRO vehicle consisted of four parts: Sounding weather balloon to produce buoy force during the ascent phase, circular parachute to produce drag force during the descent phase, tracking gear with GPS (Global Positioning System) to check the position from the ground, and experimental payloads.

High altitude balloons have several major advantages in providing access to this near-space environment when compared to aircraft or sounding rockets. High altitude balloons routinely carry payloads to high altitudes above 99.5% of the Earth atmosphere. At such high altitudes, the payload can be exposed to environmental conditions similar to a low orbital flight, with the exception of still experiencing 1-G. While sounding rockets can also allow access to the near-space environment, they are not able to endure in such environments as long as balloons can. Altitudes achievable by typical balloons exceed common aircraft operational ceilings and are far less costly when compared to the operational costs of an aircraft. For this reason, weather balloon was selected to carry the payloads to high altitude.

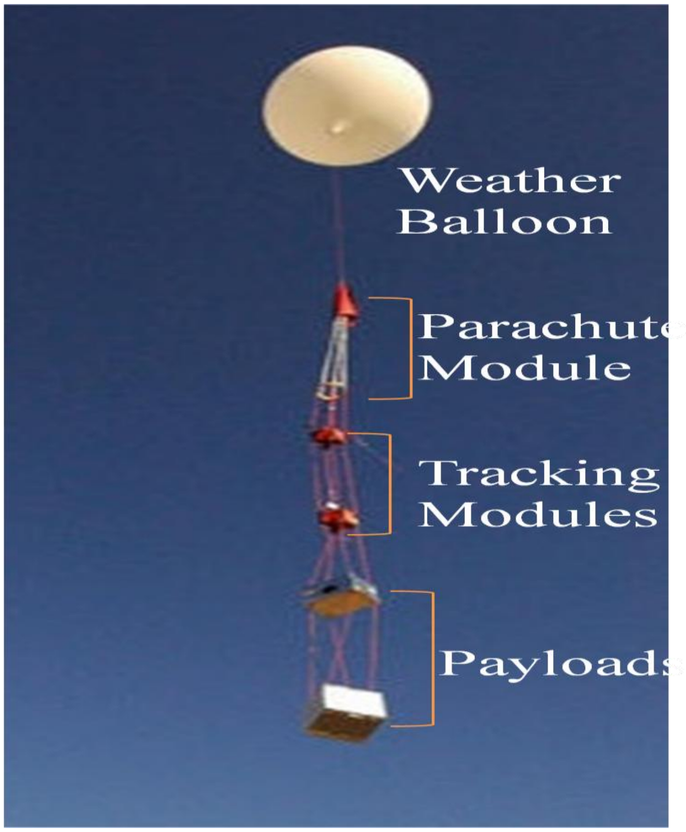


Figure 1ASTRO Balloon Launch

As mentioned, the descent phase utilizes a circular parachute, and as such, there are no means of controlling the landing location of the vehicle and payloads. Without control, the direction the parachute takes is dependent upon the winds aloft which can allow payloads to land in undesirable locations, such as rivers, lakes, or the middle of vast forests. At times, the flight must be cancelled before it even begins if the risks of a long or difficult recovery are predicted.As the ASTRO project has evolved, the necessity of control of the payloads over the descent phase has also become obvious. In order to address this need, a study of a Return-to-Point Vehicle (RPV) has been started.

Various vehicle types were initially considered for the RPV; however, the parafoil configurations were eventually chosen as the most practical, controllable vehicle during the descent phase. Table

1 lists the major advantages and disadvantages of each concept vehicle for the RPV. Mechanically, the parafoil system is relatively simple to design, and can also be easily and precisely controlled.

Table 1RPV Concepts

	Advantages	Disadvantages
Parafoil	Glide Slope Maneuverability Slow descent rate	Long flight duration Susceptibility to wind
Rotorcraft	Slow descent rate	Cost and complexity Heavy weight
Fixed Wing	Maneuverability Long Range	Cost and complexity Heavy weight

Parafoil vehicles which are used for RPVs have proven to be useful in many situations from the previous research. Their stable configuration suits autonomous guidance systems, and a low descent rate makes them an attractive choice to safely deliver the payloads to the ground. The parafoil can also glide all the way to the target without concern about the parafoil’s altitude. Maneuvering in parafoil is normally achieved by deflecting flaps or trailing edges of the canopy asymmetrically, while some degree of glide slope control is offered through symmetric flap deflections. Dropped from a sufficiently high altitude, a parafoil-controlled vehicle is capable of reaching a much vaster distance with higher accuracy than the parachute which is currently being used in The ASTRO project.

Once the RPV has reached the desired altitude, generallyaround100,000 ft, it is released from the balloon. When the RPV has been released, it will follow a trajectory which is programmed on Autopilot to direct it a desired landing zone. Some researchers tried to analyze and test the parafoil, but there are no reported uses of the parafoil for dropping the payload from high altitudes; in addition, there are not commercial products to be matched with the ASTRO research as well. Therefore, thepurpose of this research is to design the RPV and to develop the optimal trajectory to satisfy the requirements of The ASTRO project.

1.1 Previous Research on Parafoil

The parafoil has been employed in many related aerospace fields, ranging from leisure to more sophisticated aerial recoveries, because of its low speed handling qualities and versatility of applications for precision aerial delivery and recovery of payloads. Much research has been done at NASA Johnson Space Center [22, 32, 47] to describe the dynamic behavior of the parafoil, and also to develop guidance and control algorithms using wind tunnel tests, ground tow tests and actual aerial drop tests. In Europe, the Institute of Flight Research of the German Aerospace Center (DLR) [8, 13, 19] has conducted research to identify the dynamic behavior of a parafoil-payload system and to investigate Guidance Navigation and Control (GNC) concepts. The researchers used three degrees-of freedom (DOF) and four-DOF models in their own parafoil-payload system, ALEX (Autonomous Landing Experiment)-I and –II, to apply system identification algorithms and GNC designs. The ALEX system was dropped from a helicopter to acquire flight data at 10 Hz. The flight data were reconstructed with constant wind assumption; the data were also used for system identification based on 3-DOF and 4-DOF model in which the relative motion was not accounted for. Recently, Slegers and Costello [14] studied the dynamic modeling of a parafoil with 9-DOF, including the three inertial positions of the joint, as well as the three Euler angles of the parafoil and payload. In this study, both the parafoil and the payload were free to rotate around the joint but were constrained by the force and the twisting torque due to the relative yaw angle which was only present at the joint. They used a reduced state linear model based on a nonlinear 6-DOF parafoil and payload model to apply to the predictive control design. In this study, the parafoil and the payload were modeled as a single rigid body without relative motion.

1.2Research Objectives

The main goal of the project is to investigate the feasibility of an autonomous return vehicle for balloon payloads. The design objective most important to achieving the goal is to develop a cost-effective, simple and reliable autopilot system that can be applied to the payloads used in the ASTRO project. Most of the research mentioned above has been focused on the dynamics and guidance performance problems of the total system in which the parafoil and the payload are regarded as one unique body, thus neglecting the relative motion between them. The limitations of the research on modeling and flight tests seem to have been caused by the difficulty in acquiring the parameters of the parafoil itself. Moreover, because of analytical limitations, some parameters will be assumed by the typical fixed wing airplane's design simulation program.

Another important objective is optimized path finding. The parafoil vehicle is not able to perform pitch movements. Thus, identifying the optimal waypoints requires calculation of the parafoil's performance limitations in 2D and 3D. Finally, the performance of the parafoil and the application of the parafoil to the study of the RPV will be the principal focus in this project.

In this proposal, the dynamics of the RPV are modeled with 6-DOF. In Chapter 2, a dynamic model of the parafoil system is derived. The parafoil and the vehicle are considered as rigid bodies, and the elasticity of the risers and suspension lines are ignored in modeling. Additionally, the described autopilot system was chosen to perform for 'Roll' because of the limitations of payload weight and cost of the autopilot system.

CHAPTER II

PARAFOIL

The stability and controllability of a parafoil offer a great deal of flexibility to a Return-to-Point vehicle. The payload can also be released farther away from the target, which may be desirable for missions over hostile areas. In addition, the longer flight duration of the parafoil would allow for the landing site to be surveyed during the descent phase. Next, the relative simplicity of the parafoil system considerably increases the compatibility with parachute system.

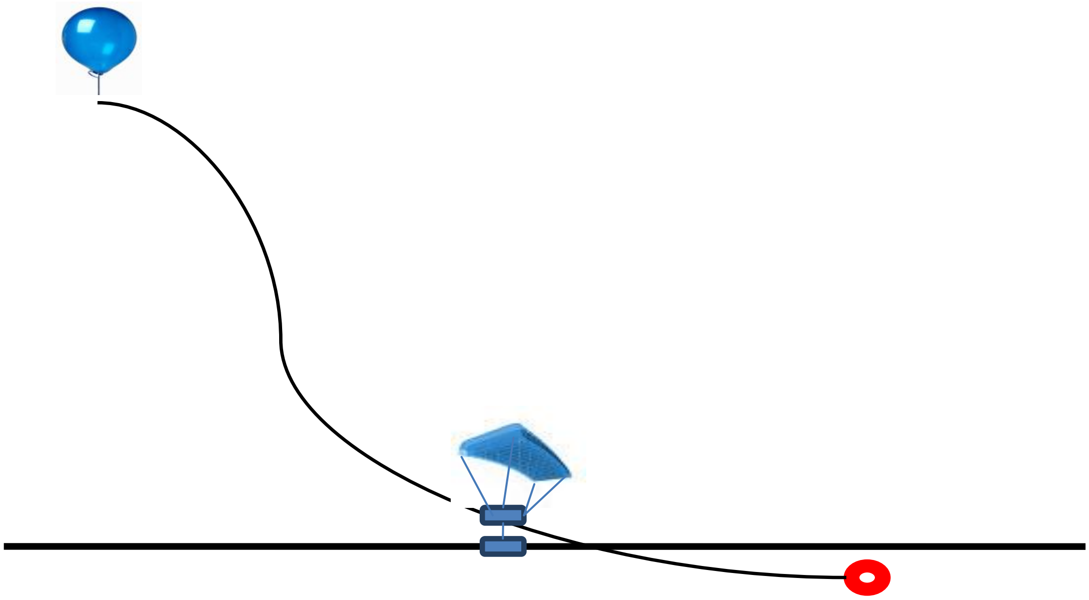


Figure 2Parafoil Return Concept

Although it brings in many advantages, the parafoil method is not a new one – there are such systems in use today. For the purpose of our system development, GPS alone can offer the required 1 mile landing accuracy. Under normal circumstances, absolute positions can only be estimated in a 5-mile radius range. For a gliding parafoil system, the landing errors are further exacerbated since a 15 mile error in altitude corresponds to a 100,000 ft horizontal error, assuming a glide ratio of 3:1. In spite of the errors, the controllability of the parafoil provides the prediction and helps achieve the movement to the desired point. The method is described in this chapter.

2.1 Parafoil Dynamics

2.1.1 Introduction

A 6 DOF dynamical model of the parafoil was developed to assist the development of the Return-to-Point vehicle project. This model is a critical component of the parafoil simulation. It is also essential to the creation of the control and guidance system, the building of an autonomous system, and finally the testing of the vehicle. This chapter describes the development of the aerodynamic model, along with its limitations.

Generally, the parafoils carry aspects of both airfoils and parachutes. Contrary to parachutes that only produce drag, a parafoil also generates lift, thus allowing for useful gliding performance. High-end parafoils can achieve glide ratios of about 3:1. Most canopies which function to produce the lift in the parafoils are ram-air chutes – their leading edges are open to allow for inflation as they move forward. The shuts are designed with individually inflatable cells to help maintain structural rigidity, and to also add stability during flight. Suspension lines are rigged to give the canopy a shape. Maneuvering is then achieved by deflecting the outer trailing edges of the canopy – asymmetric deflections for turns, symmetric deflections for flaring and velocity control.

2.1.2 Longitudinal Dynamics

We first examine the longitudinal dynamics of the parafoil. The lift curve slope, based on lifting line theory, is given as

$$a = \frac{\pi a_0 AR}{\pi AR + a_0(1+\tau)} \tag{2.1}$$

This equation is a descent phase's estimation for wings with an aspect ratio above 5, but in our case with the parafoil, the aspect ratios are approximately 2.5. A correction has been suggested by [10]:

$$a_0' = a_0 k \tag{2.2}$$

$$\text{where } k = \frac{\pi AR}{a_0}\tanh\left(\frac{a_0}{\pi AR}\right)$$

The lift coefficient in terms of angle of attack can be written as:

$$C_L = a(\alpha - \alpha_{L=0}) + k_1 \sin^2(\alpha - \alpha_{L=0})\cos(\alpha - \alpha_{L=0}) \tag{2.3}$$

Also, a non-linear component of lift is present in addition to the lift curve slope. This behavior has been observed in previous experiments, and it is believed to exist due to drag effects.

$$C_D = C_{D_0} + \frac{C_L^2}{\pi AR}(1+\delta) + k_1 \sin^3(\alpha - \alpha_{L=0}) \tag{2.4}$$

k_1 is identical to the term used in the equation for the lift coefficient. The parasite drag, C_{D0}, can be estimated by adding up the various contributions of drag in the system.

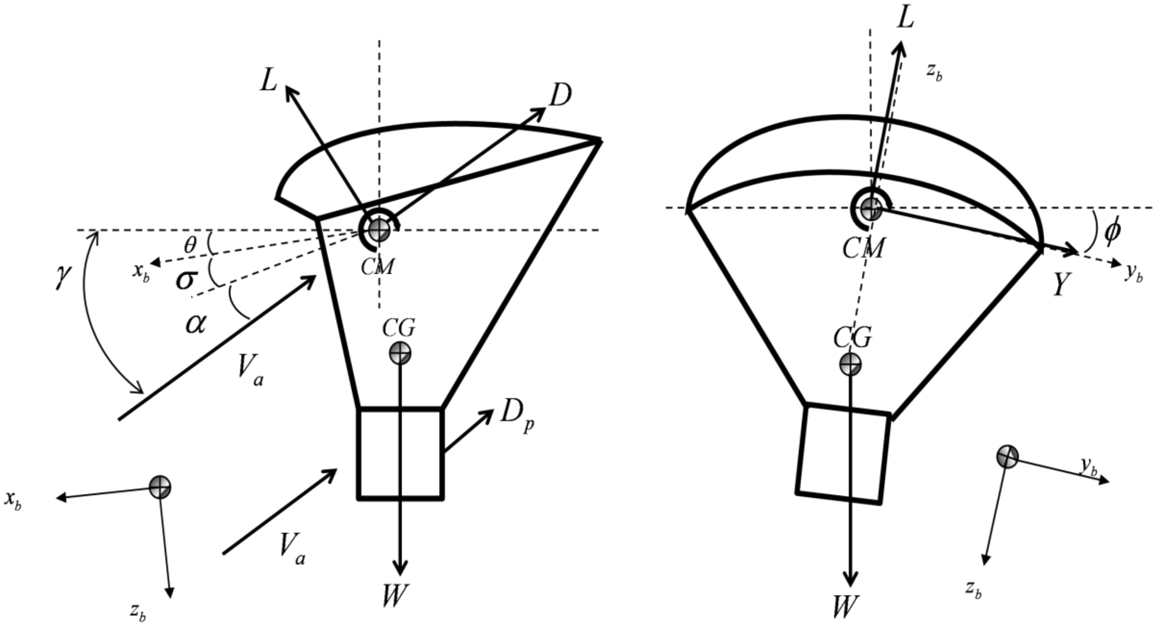


Figure 3Free body diagram of parafoil and payload

The equation for the pitching moment about the Center of Gravity (CG) of the parafoil-payload system is:

$$C_m = C_{m,c/4_0} + \frac{R_{cg}}{c}[C_D \cos(\alpha+\theta_r) - C_L \sin(\alpha+\theta_r)] - \frac{R_{sg}}{c} C_{D,p} \cos(\alpha+\theta_r) \tag{2.5}$$

The equilibrium incidence angle (where the pitching moment is zero) is a function of the rigging angle. In terms of gliding performance, there exists an optimal angle of attack at acting on the normal velocity component of the flow. Based on the experimental data for low aspect ratio wings, the constant, k_1, can be approximated from the equation:

$$k_1\text{=3.33-1.33AR} \tag{2.6}$$

We now examine the drag coefficient for ram air parachutes. Going back to the lifting line theory,

$$C_D = C_{D0} + \frac{C_L^2}{\pi AR}(1+\delta) \tag{2.7}$$

Similar to the non-linear behavior observed with the lift coefficient, there exists an additional component of drag. Including this components, the drag coefficient equation is able to be eq.(2.7) to operate. Therefore, the rigging angle is usually chosen to put the equilibrium point at the angle.

The static stability relations developed thus far provide an understanding of the parafoil’s trim performance. Given that all the forces and moments are balanced during a steady glide, the trim velocity can be obtained from,

$$V = \left(\frac{2}{\rho} \cdot \frac{W}{S} \cdot \frac{\cos\gamma}{C_L} \right)^{1/2} \tag{2.8}$$

Where the glide slope angle, $_{c}$, is related to the lift to drag ratio by

$$\gamma = \arctan\left(\frac{1}{L/D}\right) \tag{2.9}$$

2.1.3 Lateral Directional Dynamics

The lateral-directional dynamics are more complicated than the longitudinal dynamics since the roll and yaw terms are couples. Previous parafoil experiments of which the author is aware relied almost entirely on flight test data to determine the lateral-directional behavior. Since, at present, only limited flight observations have been made in our project, we use the rough estimates of the lateral-directional stability derivatives expecting that they will be tuned in the future.

A parafoil's lateral-directional control comes from the asymmetric deflections of the outer trailing edges. The difference in drag between the two sides produces a yawing moment, and a constant turn rate is then achieved when the yawing moment is balanced by damping terms.

Assuming that the yaw moment, due to the asymmetric flap deflections, comes from the additional drag, we approximate its value in eq.2.9. When calculating the moments, the resultant drag force is assumed to act halfway between the outside edge and centerline of the parafoil:

$$C_{N,\delta a} \approx \frac{1}{2}[C_{D,\delta f}\cos(\alpha + \theta_r)] \tag{2.10}$$

As mentioned above, the vehicle's turn rate depends on the yaw damping terms. That is, the larger the damping, the slower the steady-state turn will be. Simulated flight tests produced turn rates of approximately 0.6 rad/s, completing a full circle in about 10 seconds, with about 50% of asymmetric deflection. Using this turn rate for our model, and if we initially only consider the yaw due to the asymmetric deflection and the damping term, we can now balance the moments to approximate the value for $C_{N,r}$:

$$C_{N,Steady} \approx C_{N,\delta a}\delta_a - C_{N,r}\frac{b \cdot r}{2V_T} = 0 \tag{2.11}$$

The other lateral-directional terms, roll and side force, are more difficult to approximate without prior extensive flight testing. Thus, it was decided to use the aerodynamic derivatives observed in the other parafoil projects until our own flight data could become available. The only data that could be accessed came from a project dealing with a larger parafoil [30,35]. As an attempt to better fit the faster dynamics of our small parafoil, the lateral-directional stability terms were scaled so as to be in proportion to the yaw moment and the previously estimated yaw damping terms. Although this is a somewhat crude estimate for the dynamics, it does produce steady-state turn rates and bank angles that are in rough agreement with the observations made from previousflight tests by the other researchers.

The complete lateral-directional equations used in the simulations are given below. The moments are with respect to the system CG location:

$$C_\gamma = C_{\gamma,\beta}\beta + C_{\gamma,\delta a}\delta_a \tag{2.12}$$

$$C_l = C_{l,\beta}\beta + C_{l,\delta a}\delta_a + \frac{b \cdot p}{2V_T}C_{l,p} + \frac{b \cdot r}{2V_T}C_{l,r} \tag{2.13}$$

$$C_n = C_{n,\beta}\beta + C_{n,\delta a}\delta_a + \frac{b \cdot p}{2V_T}C_{n,p} + \frac{b \cdot r}{2V_T}C_{n,r} \tag{2.14}$$

2.1.4 Moments with Inertia Terms

A ram-air parachute’s structure is very light, which means that it is heavily influenced by the air passing over and around it. The pendulum motion of a parafoil system is similarly affected since the point of rotation moves away from the CG of the system and towards the parafoil.

Furthermore, because the magnitude of the ‘moments’ effect depends on the axis of rotation, there may exist multiple points of rotation for the parafoil system.

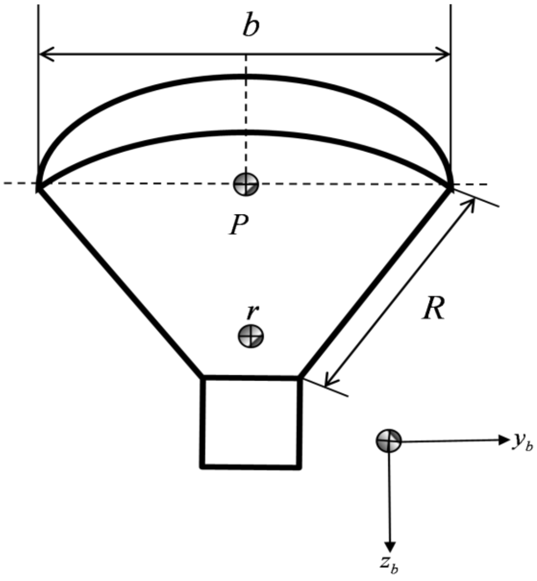


Figure 4Parafoil moments geometry, center of location

$$M_B = \begin{bmatrix} L_{aero} \\ M_{aero} \\ N_{aero} \end{bmatrix}_B = \begin{bmatrix} C_l b \\ C_m \bar{c} \\ C_n b \end{bmatrix}_B \frac{1}{2}\rho V_a^{\ 2} S \tag{2.15}$$

Three different moments terms exist in general, since the moments effect is different depending on the direction of the airflow. For instance, a parafoil moving forward will displace a greater amount of air than if it were moving in a sideslip direction. Values for the moments parameters were obtained using the flat plate approximation methods described in [7]. They are based on the assumption that the moments are related to the cross-sectional area, perpendicular to the direction of the flow.

2.1.5 Equation of Motion

Putting together the forces and moments with the inertia terms, we can now write the complete equations of motion.

The forces equation is:

$$F_B = F_A + F_W \tag{2.16}$$

$$F_A = \begin{bmatrix} X_{aero} \\ Y_{aero} \\ Z_{aero} \end{bmatrix}_B = \begin{bmatrix} C_x \\ C_Y \\ C_Z \end{bmatrix}_B \frac{1}{2}\rho {V_a}^2 S = H_B^A \begin{bmatrix} C_{Do} + C_{D\alpha}\alpha + C_{Dp} \\ C_{Y\beta}\beta \\ C_{L0} + C_{L\alpha}\alpha \end{bmatrix} \bar{q}S = \begin{bmatrix} -C_D \cos(\alpha) + C_L \sin(\alpha) \\ C_{Y\beta}\beta \\ -C_D \sin(\alpha) - C_L \cos(\alpha) \end{bmatrix} \bar{q}S \tag{2.17}$$

$$F_W = \begin{bmatrix} -\sin\theta \\ \sin\phi\cos\theta \\ \cos\phi\cos\theta \end{bmatrix} mg \tag{2.18}$$

where H_B^A is the transformation from the aerodynamic axis to body frame:

$$H_B^I = \begin{bmatrix} \cos\theta\cos\psi & \cos\theta\sin\psi & -\sin\theta \\ -\cos\phi\sin\psi + \sin\phi\sin\theta\cos\psi & \cos\phi\cos\psi + \sin\phi\sin\theta\sin\psi & \sin\phi\cos\theta \\ \sin\phi\sin\psi + \cos\phi\sin\theta\cos\psi & -\sin\phi\cos\psi + \cos\phi\sin\theta\sin\psi & \cos\phi\cos\theta \end{bmatrix} \tag{2.19}$$

And the moment of body is:

$$M_B = \begin{bmatrix} L_{aero} \\ M_{aero} \\ N_{aero} \end{bmatrix}_B = \begin{bmatrix} C_l b \\ C_m \bar{c} \\ C_n b \end{bmatrix}_B \frac{1}{2}\rho {V_a}^2 S = \begin{bmatrix} C_l b \\ C_m \bar{c} \\ C_n b \end{bmatrix}_B \bar{q}S = \begin{bmatrix} \left(C_{l\beta}\beta + \frac{b}{2V_a} C_{lp} p + \frac{b}{2V_a} C_{lr} r + C_{l\delta a}\delta_a \right) b \\ \left(C_{m0} + \frac{\bar{c}}{2V_a} C_{m\dot{\alpha}}\dot{\alpha} + \frac{\bar{c}}{2V_a} C_{mq} q \right) \bar{c} \\ \left(C_{n\beta}\beta + \frac{b}{2V_a} C_{np} p + \frac{b}{2V_a} C_{nr} r + C_{n\delta a}\delta_a \right) b \end{bmatrix} \bar{q}S \tag{2.20}$$

After put together both Forces and Moment terms, the equation of motion is now:

Rate of change of Translational Position

$$\dot{x} = (\cos\theta\cos\psi)u + (-\cos\phi\sin\psi + \sin\phi\sin\theta\cos\psi)v + (\sin\phi\sin\psi + \cos\phi\sin\theta\cos\psi)w \tag{2.21}$$

$$\dot{y} = (\cos\theta\sin\psi)u + (\cos\phi\cos\psi + \sin\phi\sin\theta\sin\psi)v + (-\sin\phi\cos\psi + \cos\phi\sin\theta\sin\psi)w \tag{2.22}$$

$$\dot{z} = (-\sin\theta)u + (\sin\phi\cos\theta)v + (\cos\phi\cos\theta)w \tag{2.23}$$

Rate of change of Translational Velocity

$$\dot{u} = X_b / m - g\sin\theta + rv - qw \tag{2.24}$$

$$\dot{v} = Y_b / m + g\sin\phi\cos\theta - ru + pw \tag{2.25}$$

$$\dot{w} = Z_b / m + g\cos\phi\cos\theta + qu - pv \tag{2.26}$$

Rate of change of angular Position

$$\dot{\phi} = p + (q\sin\phi + r\cos\phi)\tan\theta \tag{2.27}$$

$$\dot{\theta} = q\cos\phi - r\sin\phi \tag{2.28}$$

$$\dot{\psi} = (q\sin\phi + r\cos\phi)\sec\theta \tag{2.29}$$

Rate of change of Angular Velocity

$$\dot{p} = (I_{zz}L_b + I_{xz}N_b - \{I_{xz}(I_{yy} - I_{xx} - I_{zz})p + [I_{xz}^2 + I_{zz}(I_{zz} - I_{yy})]r\}q)/(I_{xx}I_{zz} - I_{xz}^2) \tag{2.30}$$

$$\dot{q} = \frac{1}{I_{yy}}[M_b - (I_{xx} - I_{zz})pr - I_{xz}(p^2 - r^2)] \tag{2.31}$$

$$\dot{r} = (I_{xz}L_b + I_{xx}N_b - \{I_{xz}(I_{yy} - I_{xx} - I_{zz})r + [I_{xz}^2 + I_{xx}(I_{xx} - I_{yy})]p\}q)/(I_{xx}I_{zz} - I_{xz}^2) \tag{2.32}$$

with $\begin{bmatrix} x & y & z & \phi & \theta & \psi & u & v & w & p & q & r \end{bmatrix}^T$

where $X_b = \frac{C_x \bar{q} S}{m}$ $\quad Y_b = \frac{C_y \bar{q} S}{m}$ $\quad Z_b = \frac{C_z \bar{q} S}{m}$ $L_b = C_l \bar{q} S b$ $\quad M_b = C_m \bar{q} S c$

$N_b = C_n \bar{q} S b$

So these equations can be:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \\ \dot{u} \\ \dot{v} \\ \dot{w} \\ \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} u(\cos\theta\cos\psi) + v(-\cos\phi\sin\psi + \sin\phi\sin\theta\cos\psi) + w(\sin\phi\sin\psi + \cos\phi\sin\theta\cos\psi) \\ u(\cos\theta\sin\psi) + v(\cos\phi\cos\psi + \sin\phi\sin\theta\sin\psi) + w(-\sin\phi\cos\psi + \cos\phi\sin\theta\sin\psi) \\ u(-\sin\theta) + v(\sin\phi\cos\theta) + w(\cos\phi\cos\theta) \\ p + (q\sin\phi + r\cos\phi)\tan\theta \\ q\cos\phi - r\sin\phi \\ (q\sin\phi + r\cos\phi)\sec\theta \\ X_b - g\sin\theta + rv - qw \\ Y_b + g\sin\phi\cos\theta - ru + pw \\ Z_b + g\cos\phi\cos\theta + qu - pv \\ \frac{I_{zz}L_b + I_{xz}N_b - \{I_{xz}(I_{yy} - I_{xx} - I_{zz})p + [I_{xz}^2 + I_{zz}(I_{zz} - I_{yy})]r\}q}{I_{xx}I_{zz} - I_{xz}^2} \\ \frac{M_b - (I_{xx} - I_{zz})pr - I_{xz}(p^2 - r^2)}{I_{yy}} \\ \frac{I_{xz}L_b + I_{xx}N_b + \{I_{xz}(I_{yy} - I_{xx} - I_{zz})r + [I_{xz}^2 + I_{xx}(I_{xx} - I_{yy})]p\}q}{I_{xx}I_{zz} - I_{xz}^2} \end{bmatrix} \tag{2.33}$$

We can leave the equations in their present form if we note that setting the other mass and force terms in the inertia matrices to zero removes the unwanted terms. Also, the equations need to be studied before application to the control.

2.2Quaternion's Rotation

The difficulty with Euler angles is that singularities can be encountered in their propagation. It occurred a number of times in flight simulations runs. Thus, Quaternion form is used in Matlab simulation program, instead of Euler rotation. In Quaternion form, a solid-body rotation from on attitude to another can be characterized as a Euler rotation, a single rotation about some axis in the reference frame. The rotation can be represented by four parameters: The three direction cosines of a unit vector n aligned with the rotational axis and the magnitude of the rotation angle χ . The transformation matrix H_B^I can be expressed by:

$$H_B^I = \left(\cos^2\frac{\chi}{2} - \sin^2\frac{\chi}{2}\right) I_3 + 2\sin^2\frac{\chi}{2} nn^T + 2\cos\frac{\chi}{2}\sin\frac{\chi}{2}\tilde{n} \tag{2.34}$$

Suggesting that the four-element quaternion vector e be defined as:

$$e = \begin{bmatrix} \cos\frac{\chi}{2} \\ \sin\frac{\chi}{2} n \end{bmatrix} \tag{2.35}$$

Then, H_I^B can be written as:

$$H_I^B = \begin{bmatrix} (e_1^2 - e_2^2 - e_3^2 + e_4^2) & 2(e_1e_2 + e_3e_4) & 2(e_2e_4 - e_1e_3) \\ 2(e_3e_4 - e_1e_2) & (e_1^2 - e_2^2 + e_3^2 - e_4^2) & 2(e_2e_3 + e_1e_4) \\ 2(e_1e_3 + e_2e_4) & 2(e_2e_3 - e_1e_4) & (e_1^2 + e_2^2 - e_3^2 - e_4^2) \end{bmatrix} \tag{2.36}$$

Because four components are used to express a three-dimensional rotation, the quaternion elements must satisfy an equation of constraint. The normality condition constraint follows, which guarantees that the vector magnitude is 1:

$$e_1^2 + e_2^2 + e_3^2 + e_4^2 = 1 \tag{2.37}$$

The quaternion elements can be propagated by the following differential equation, in which a (4x4) skew-symmetric matrix of the angular rates is employed:

$$\begin{bmatrix} \dot{e}_1 \\ \dot{e}_2 \\ \dot{e}_3 \\ \dot{e}_4 \end{bmatrix} = \begin{bmatrix} 0 & -r & -q & -p \\ r & 0 & -p & q \\ q & p & 0 & r \\ p & -q & r & 0 \end{bmatrix} \begin{bmatrix} e_1 \\ e_2 \\ e_3 \\ e_4 \end{bmatrix} \tag{2.38}$$

2.3Nine Degree of Freedom of Dynamic Modeling

Six degree-of-freedom (DOF) models have considered a parafoil-payload system with a rigid body, while higher degree-of-freedom models have considered a flexible joint between the parafoil and payload. Some of these joints allow for pitching, rolling and yawing of the payload around the confluence point, therefore adding 3 additional DOF.

$$\begin{bmatrix} \dot{x}_c \\ \dot{y}_c \\ \dot{z}_c \end{bmatrix} = \begin{Bmatrix} u_c \\ v_c \\ w_c \end{Bmatrix} \tag{2.39}$$

$$\begin{bmatrix} \dot{\phi}_b \\ \dot{\theta}_b \\ \dot{\psi}_b \end{bmatrix} = \begin{bmatrix} 1 & \sin\phi_b \tan\theta_b & \cos\phi_b \tan\theta_b \\ 0 & \cos\phi_b & \sin\phi_b \cos\theta_b \\ 0 & \sin\phi_b \sec\theta_b & \cos\phi_b \sec\theta_b \end{bmatrix} \begin{Bmatrix} p_b \\ q_b \\ r_b \end{Bmatrix} \text{ on parafoil} \tag{2.40}$$

$$\begin{bmatrix} \dot{\phi}_p \\ \dot{\theta}_p \\ \dot{\psi}_p \end{bmatrix} = \begin{bmatrix} 1 & \sin\phi_p \tan\theta_p & \cos\phi_p \tan\theta_p \\ 0 & \cos\phi_p & \sin\phi_p \cos\theta_p \\ 0 & \sin\phi_p \sec\theta_p & \cos\phi_p \sec\theta_p \end{bmatrix} \begin{Bmatrix} p_p \\ q_p \\ r_p \end{Bmatrix} \text{ on payload} \tag{2.41}$$

The actual ASTRO system connects the parafoil and payload using strings that can be twisted and stretched, thus allowing the bodies to be at different relative orientations and positions.

In the current study, a nine DOF model has been used to consider the changes in relative orientation, as proposed by Nathan Slerger& Mark Costello [14]. This modeling can be expressed using two six DOF models (parafoil and payload) that are constrained with each other through three constraint equations. The constraint equations are about the relative locations of both the parafoil and payload.

This model produces reasonable transition behavior and integral performance, but the nine DOF model has the limitation of implementing on real systems due to the complex algorism for small Autopilot with minimal benefits. For this reason, the researchers who are working on autonomous parafoils prefer the six DOF model. Moreover, they commonly develop simplified 6DOF models for their systems.

As a result, nine DOF models provide more accuracy, but simplified six DOF models are better models to use on real autonomous systems. Consequently, only the simple six DOF dynamic model is considered in the ASTRO project.

2.4Simulated Parafoil Model

The parafoil's parameters are needed to be decided in order to perform flight simulations in Matlab. Unfortunately, obtaining accurate estimates of aerodynamic coefficients is an entirely different field of study. Theoretical and experimental studies have been conducted to determine these parameters. Analytical approach to the parafoil's shape helped determine the model of the inflated parafoil Figures [5, 6]. The expected shape of the parafoil is generated with light weight wood frame in the air.

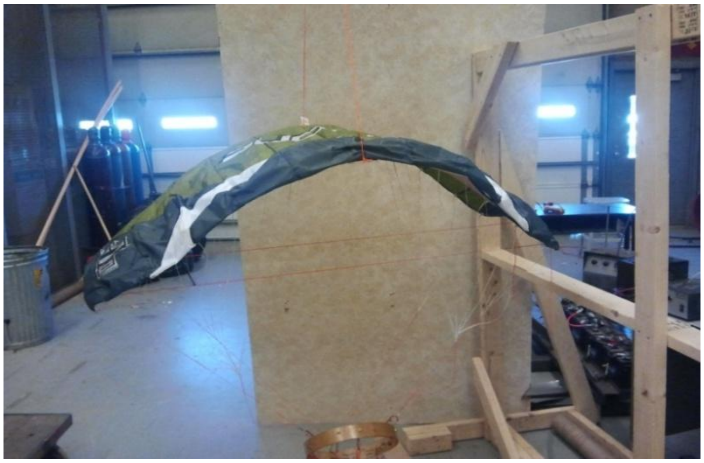


Figure 5Shape of parafoil

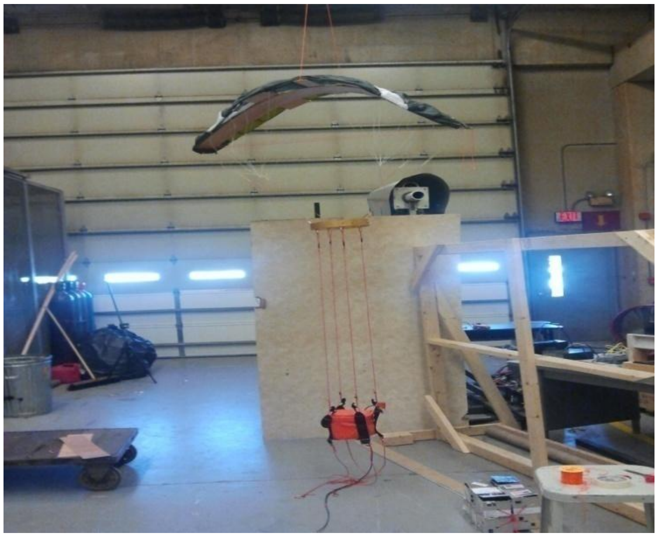
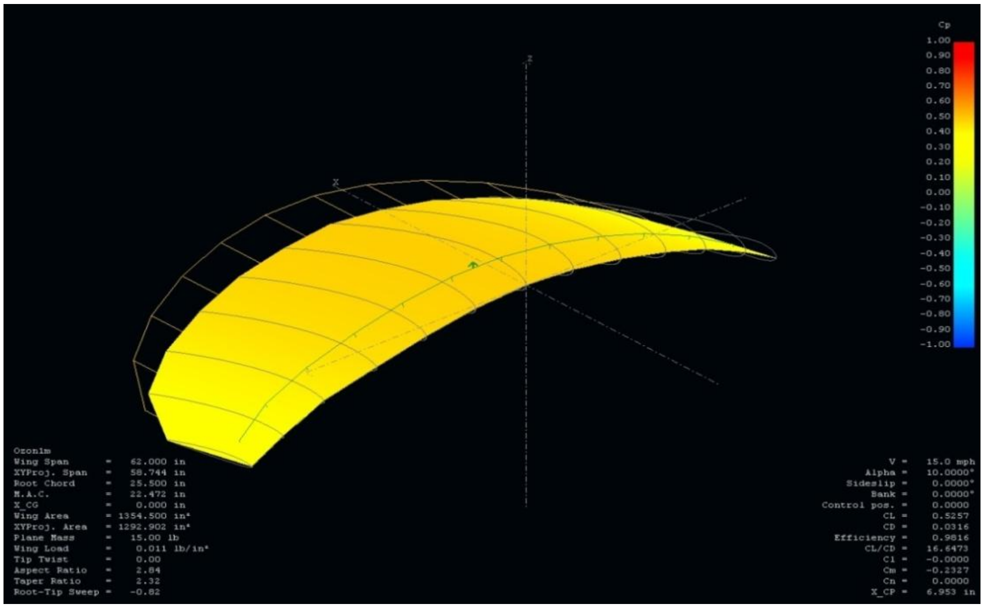


Figure 6Shape of parafoil with control box

This shape of parafoil is generated on airfoil and airplane analysis program, XFLR5. The model is simulated with 15 mph wind, and inviscid conditions and results are on Figure [7, 8]



```
Ozonim
Wing Span         =   62.000 in
XYProj. Span      =   58.744 in
Root Chord        =   25.500 in
M.A.C.            =   22.472 in
X_CG              =    0.000 in
Wing Area         = 1354.500 in²
XYProj. Area      = 1292.902 in²
Plane Mass        =   15.00 lb
Wing Load         =    0.011 lb/in²
Tip Twist         =    0.00
Aspect Ratio      =    2.84
Taper Ratio       =    2.32
Root-Tip Sweep    =   -0.82
```

```
            V =   15.0 mph
        Alpha =   10.0000°
     Sideslip =    0.0000°
         Bank =    0.0000°
 Control pos. =    0.0000
           CL =    0.5257
           CD =    0.0316
   Efficiency =    0.9816
        CL/CD =   16.6473
           Cl =   -0.0000
           Cm =   -0.2327
           Cn =    0.0000
         X_CP =   6.953 in
```

Figure 7Wing simulation at XFLR5

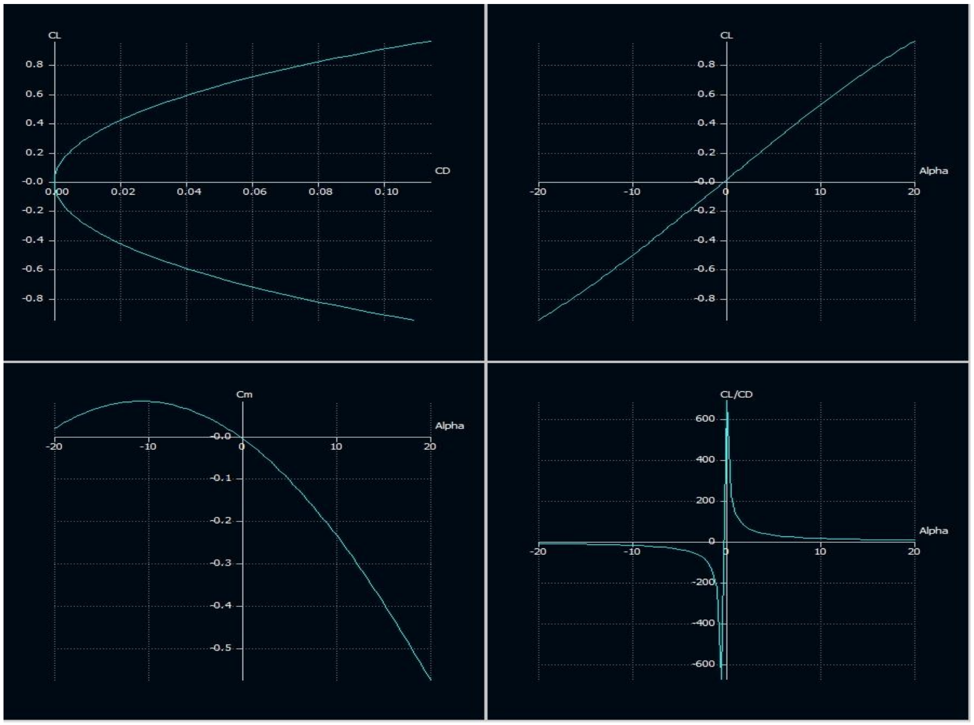


Figure 8Parafoil coefficients data

After running the analysis tool, XFLR5, some of aerodynamic coefficients about the wing are acquired. However, it is difficult to obtain coefficients about the control surfaces. Thus, the parafoil is simulated in AVL Editor as well, with two different cases of control surface: Roll only control mode, and roll and yaw control mode.

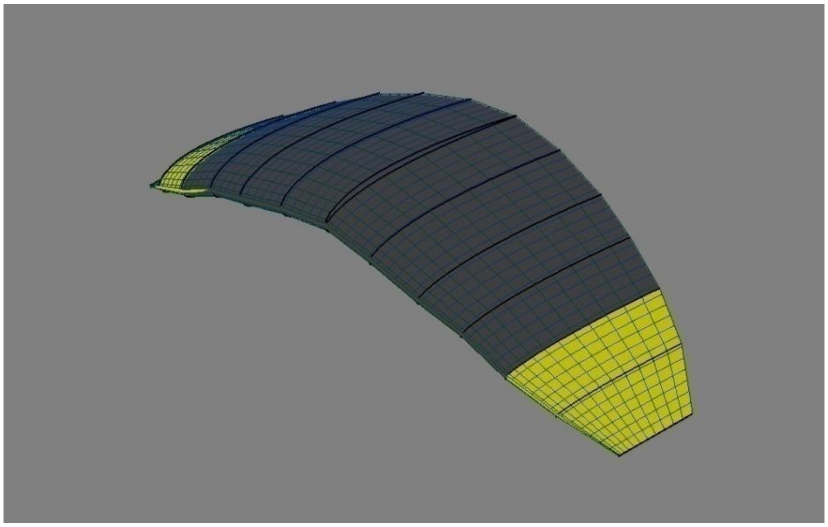


Figure 9Wing simulation on AVL Editor with roll control surface

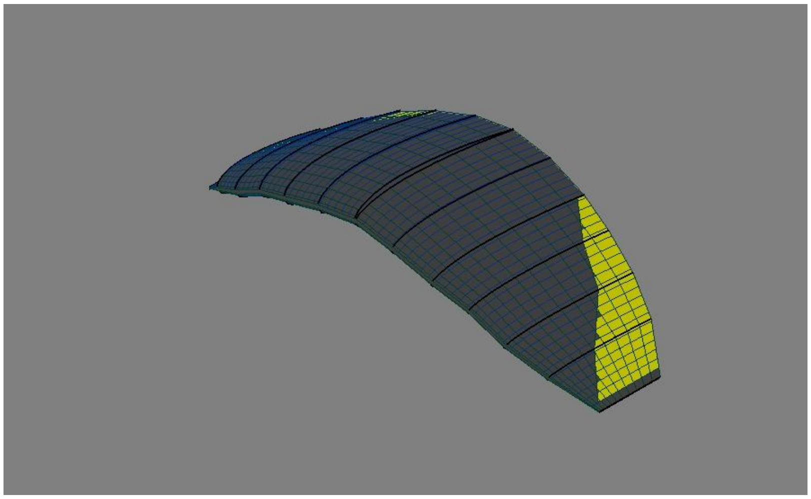


Figure 10Wing simulation on AVL Editor with roll and yaw control surface

The data from both analysis tools are applied to the flight simulation program of Matlab, which is discussed further in Chap. 7. Also, the coefficients table of each angle of attack is available in Appendix B.

CHAPTER III

OPTIMIZED PATH-FINIDING

The main purpose of path-finding is to improve the vehicle's control reliability, flexibility and accuracy in varying wind conditions, the biggest factor affecting the vehicle's movement. In this research, the system equations, the performance index and the constraints were not simple, and there were limitations on control surfaces and axes of adjustment. Therefore, numerical methods were needed to solve optimal path-finding and control problems. All numerical methods for the solution of our problems necessarily involve the optimized method. Among several methods, dynamic programming and constraints optimization were considered to this research. Dynamic programming, as applied to two-point boundary-value problems, can be described as a process of generating many solutions satisfying the specified boundary conditions at one end, using the unspecified boundary conditions as parameters [3, 4]. Constrained optimization involves calculating all possible uncertain condition and variables and identifying the possibilities within them. As a result, the optimized correct range of parameters is chosen, as some of the solutions will pass through the desired boundary conditions at the final target point. This method is used in the path-planning scheme for the RPV. This method refers to our very large class of flight-path algorithms and is meant to break a large flight range problem down (if possible) into incremental steps, so that, at any given stage, optimal solutions to sub-problems are known.

3.1 Dynamic Programming

Dynamic Programming (DP) is an optimization method applicable to the systems in which the variables and objective functions can be of various forms. The big advantage of this technique is the ability to determine an optimum sequence of decisions, or the entire system's optimal path from the starting set point to the final set point. The path determined by DP provides all possible waypoints.

The concept of DP is dividing the applications into stages and then conducting analysis on each stage. For example in DP, stage one is optimized, then the next stage is optimized with respect to the results obtained from the first. This process then continues until all the states have been analyzed. Suppose that the vehicle is flying from A to D, as shown in Fig. 11, and we ask what the shortest time path is:

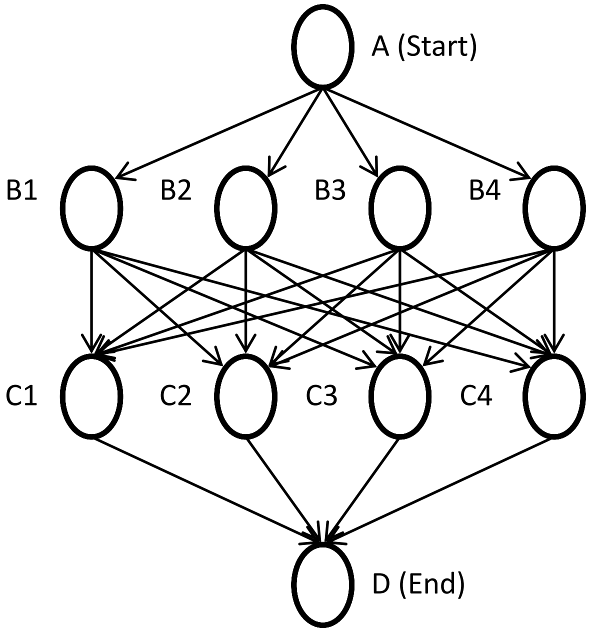


Figure 11Simple example of Dynamic Programming

The dynamic programming approach has three steps. First, from each C point, the best path to D point is selected. Second, each B point observes each C transition to D, and chooses the transition that best permits the optimal C to D transition. Third, the program computes the distance for each of the possible paths from A to B, constrained by optimal paths from the B points to the C. Thus, the combined path with the shortest distance is the ultimate solution presented.

Theoretically, dynamic programming is an effective method to find out the optimized path in typical flight conditions; however, without pitch control, the flight distance does not provide any benefits. Therefore, this dynamic programming approach is only available in case of Roll and Yaw control surface cases from Chapter 2.

To satisfy, both Roll & Yaw and Roll control cases, a constrained optimization was applied.

3.2 Constrained Optimization

Constrained optimization is a method applicable to an objective function with respect to some possible variables and uncertainties in the presence of constraints on those variables. The objective function is either a cost function, which is to be minimized, or an energy function, which is to be maximized. Constraints can be either hard constraints, which set conditions for the variables that are required to be satisfied, or soft constraints, which have some variable values that are penalized in the objective function if, and based on the extent that, the conditions on the variables are not satisfied. Many unconstrained optimization algorithms can be adapted to the constrained case, often via the use of a penalty method. However, search steps taken by the unconstrained method may be unacceptable for the constrained problem, leading to a lack of convergence.

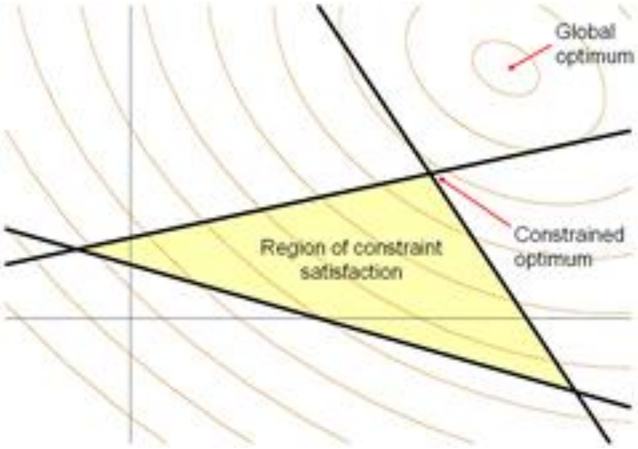


Figure 12 Example of Constrained Optimization

In this research, this method used to calculate the feasible area in which the parafoil vehicle is able to maneuver under the uncertain wind condition, the varying starting points and the parafoil's performance constraints. As mentioned earlier, the purpose of using this technique is to find an optimized path to the destination. The path is only for longitude and latitude. The reason is that the vehicle has only one control input, roll. Therefore, the parafoil is theoretically gliding with a constant rate of descent speed. In other words, the heading direction with related longitude and latitude is controllable. To identify the best feasible area in different wind speeds, the steps are divided based upon wind speed from given weather data.

After the calculation of feasible gliding range in 2D, the method can be extended3D, as in Figure 13.

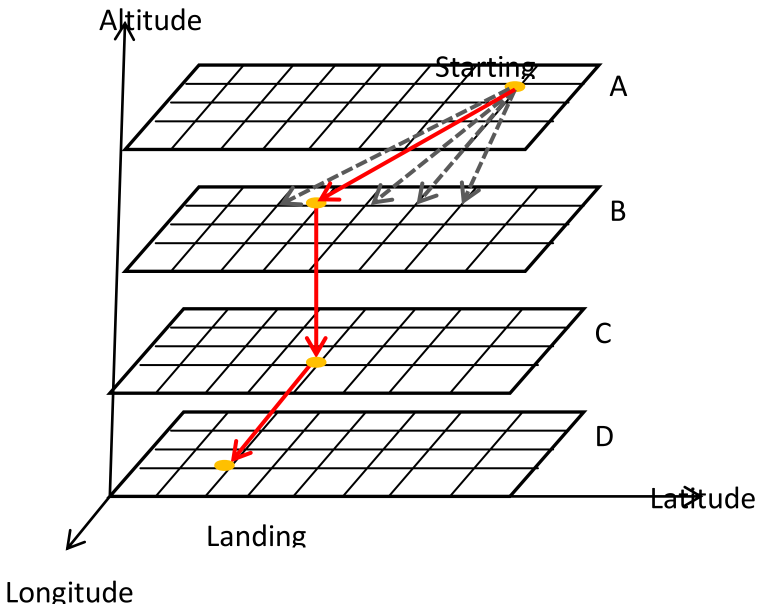


Figure 13Applied the path on three dimension

From this idea, the specific equations and calculations are derived.

3.3 Feasible gliding range calculation

3.3.1 Two dimension plane

Before applying DP, the constraints which come from the performance of the parafoil, such as wind conditions and other uncertain parameters, should be calculated to find the values of decision parameters. First of all, feasible gliding range was simulated in two dimensions with one wind direction and varying wind speed by altitude under the following assumptions:

Assumptions

- Parafoil velocity: 15mph
- Wind speed : [8,15,20,25,40,60,55,45,30,18];(mph)

  (low to high altitude)(1000ft steps)
- Descent speed of each step= 4.4(ft/s) = (3 mph)
- Step = 1000ft
- Flight time of each step = Step/descent speed
- Pitch angle is fixed
- No vertical wind (on Y axis)

For all the steps, the appropriate MATLAB/Simulink Simulation programs were developed.

Step 1.Gliding range from starting point to ground

In 2D, only two heading directions—heading into the wind and heading with the wind—were considered, called “Forward” (Heading into wind) and “Backward” (Heading with wind). Calculations were performed at each step as follows:

Forward speed = Wind speed – Parafoil velocity

Backward speed = Wind speed + Parafoil velocity

Forward distance = (Forward speed x Flight time) x 0.00019 (mile)

Backward distance = (Backward speed x Flight time) x 0.00019 (mile)

At most altitudes, wind speed is faster than the parafoil's maximum speed, so the parafoil flying against the wind did not make much headway, and the one heading with wind flew further, from its staring latitude. Forward distance and backward distance were calculated at each step, and the distance was added to the accumulated distance of the previous steps. In other words, if stage 1 had 10 miles forward distance, stage 2 started the distance calculation from 10 miles and added stage 2's result. The result is Fig.14with assumed wind speed.

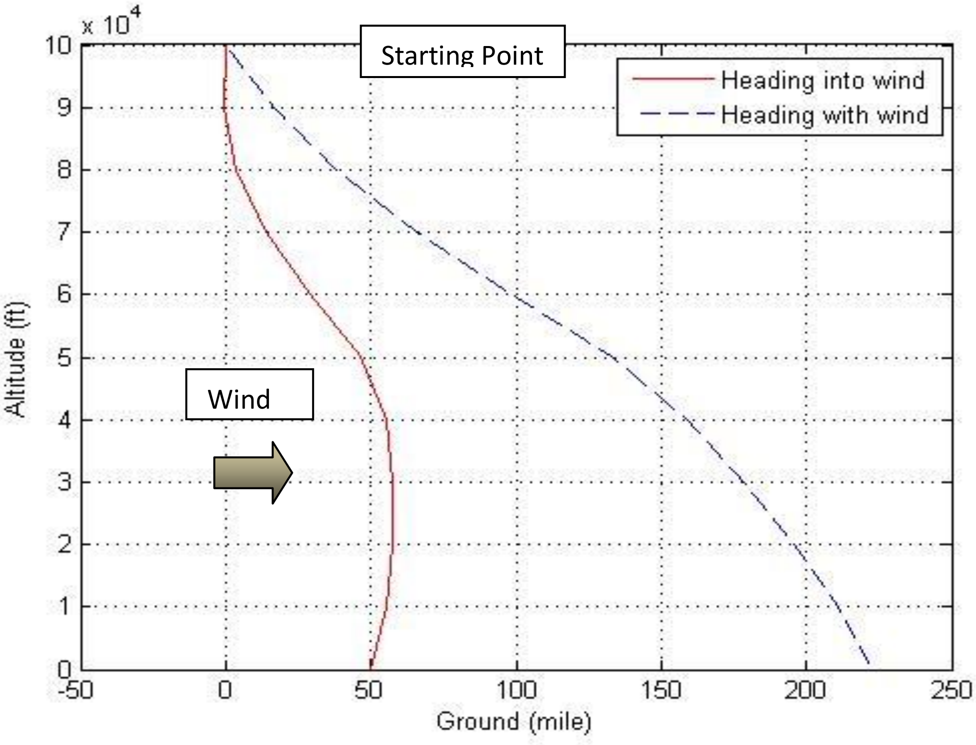


Figure 14: Feasible gliding range from air to ground

Step 2.Gliding range from desired landing point to air

From Step 1, the feasible ground range was calculated and the desired landing point was able to be identified within the feasible landing range. However, the problem was that only limited points from 1000ft. were able to reach our desired point. In other words, the point had a feasible range to reach the landing point because of the parafoil's performance limitations.

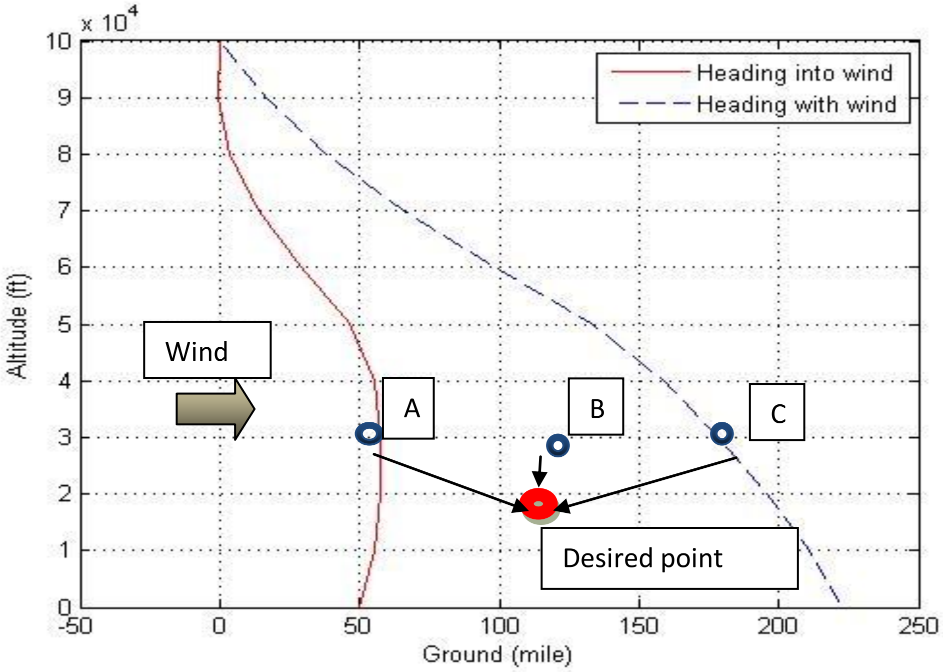


Figure 15: Feasible gliding range with possible waypoints

Fig. 15 is an example. The desired point was decided on as 120 and it was located within the feasible range of Step 1. If the three points, A, B, and C were considered as waypoints at 1000ft, A and C could not reach the desired point even though they were within the feasible range of Step 1; they were outside the range to the target landing point imposed by the parafoil's performance limitations. However, the parafoilwas able to reach the desired point from point B because it was within those performance limitations. Therefore, a feasible starting range calculation was required to determine limitations on the feasible waypoints for reaching our target point. The first thing to do for this calculation was to change the starting altitude at each stage and ensuring the desired point should fall within the feasible range of each step, as in Fig. 16.

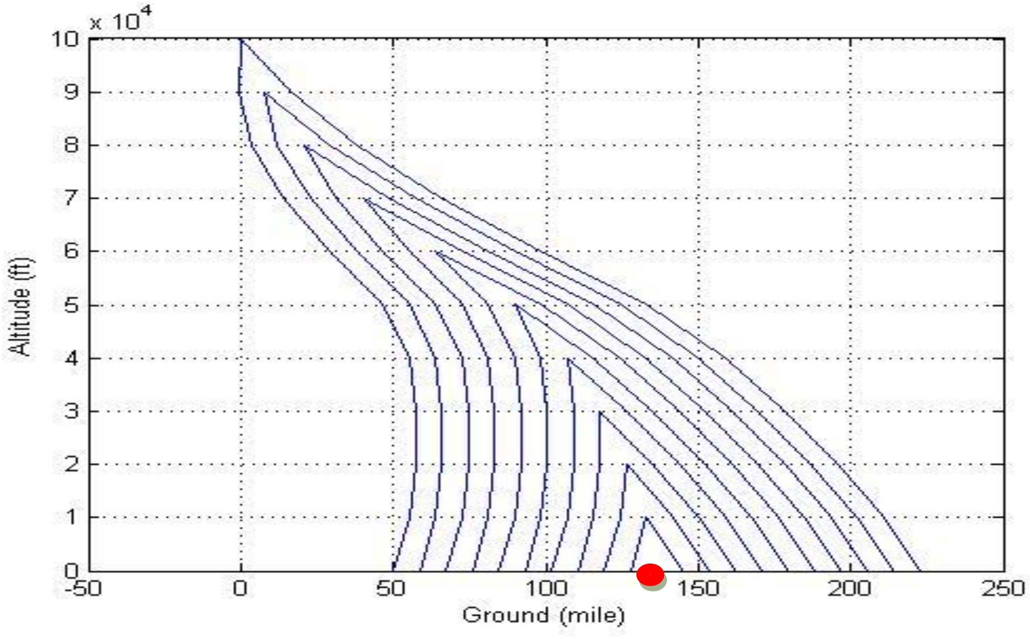


Figure 16: Feasible range from each step

For convenience, only the last stage was considered as an example to calculate the feasible starting range to reach our desired point, as in Fig. 17:

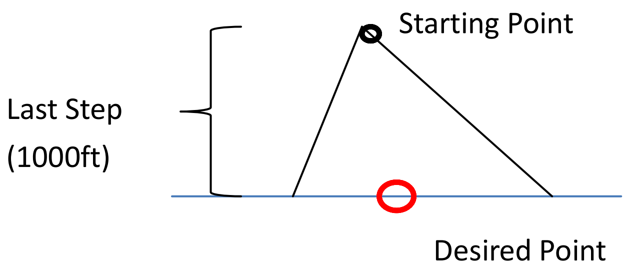


Figure 17: Feasible range of last step

At the last stage, the triangle (feasible range from the last stage) should contain the desired point.

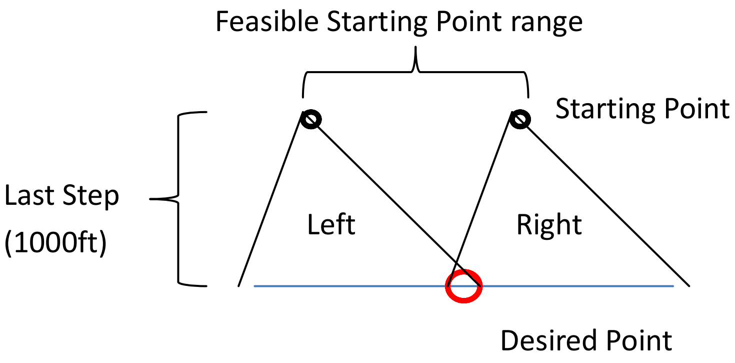


Figure 18: Feasible starting point range to last step

The cases in which the triangle contains the desired points are those in which the Forward and Backward triangles touch the near and far sides of their feasible range to the desired point. If the parafoil was located in the feasible starting point range, it was able to reach the desired point. This method was expanded to all stages, as follows:

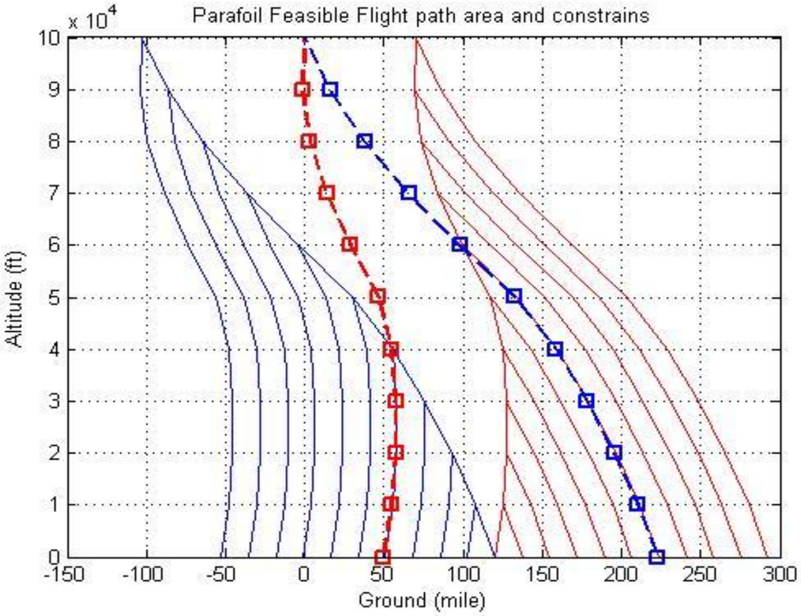


Figure 19: Feasible starting range of each step

The bold lines were Step1's calculation, and the other lines were the feasible range of each stage, from which the triangle shifted to left and right could reach the desired point. From the plot, feasible range was calculated, as below in Fig.20. This showed that if the parafoil wants to reach the desired landing point, the parafoilshould be made to glide between the red and blue line. This is the Step 2 feasible range.

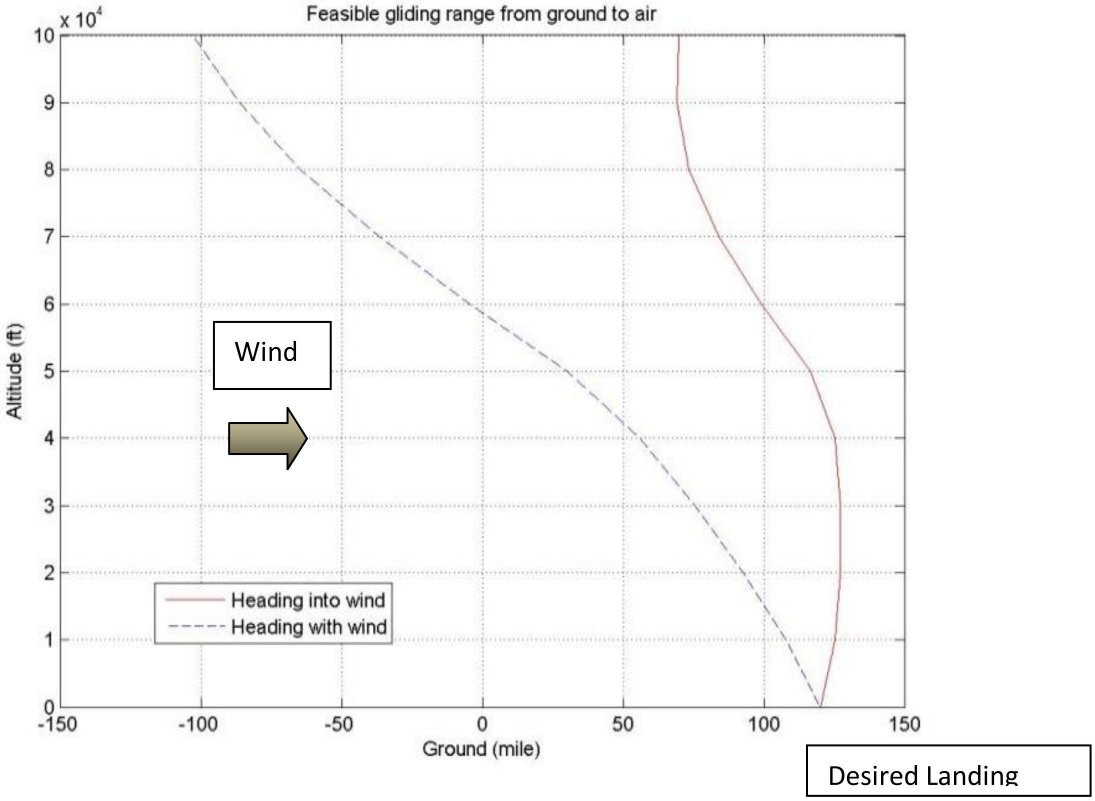


Figure 20: Feasible gliding range from ground to air

Step 3.Sum of Step 1 and Step 2

Basically, Step 3 is finding the overlapped area between Step 1 and Step 2, as in Fig. 21, below:

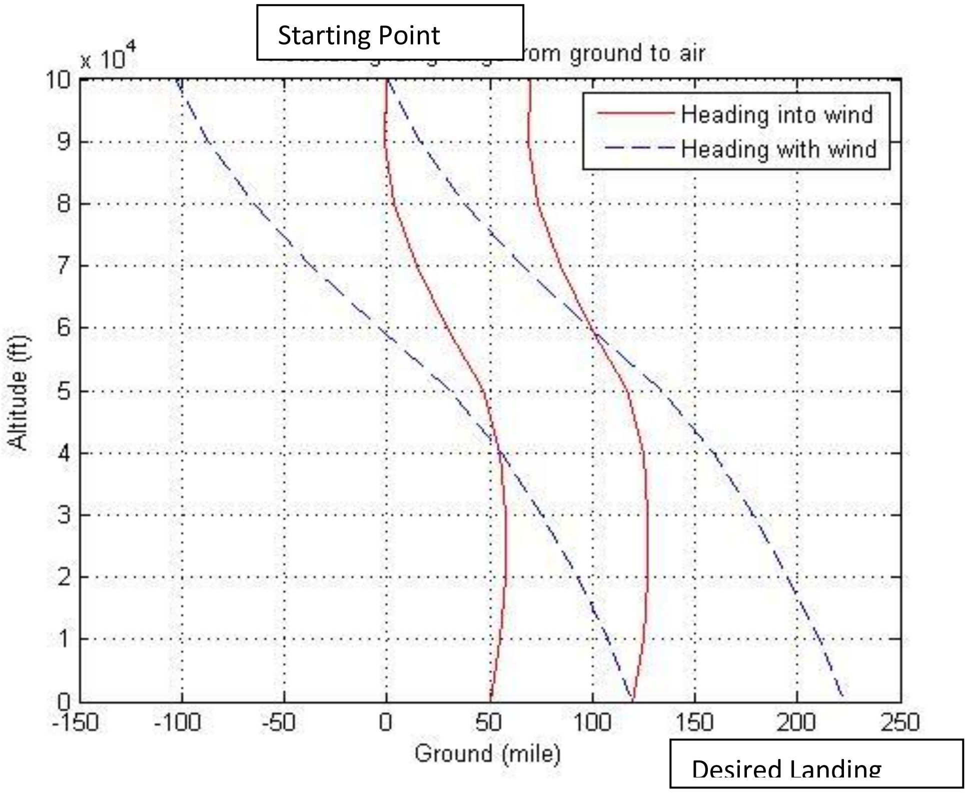


Figure 21: Sum of Step 1 and Step 2

The straight line indicates the flight path heading into the wind, and the dotted line is the flight path heading with the wind of from the starting point to ground and the ground to starting point.

In Fig.21, the area to satisfy both Step 1 and Step 2 was determined. In this area, the parafoil was able to reach the desired point from starting point with its heading controls. For example of the left flight path of overlapped area, if the parafoil heads into the wind and then turns its heading to align with the wind at the crossing point of both flight paths, the vehicle is able to land on our desired point.

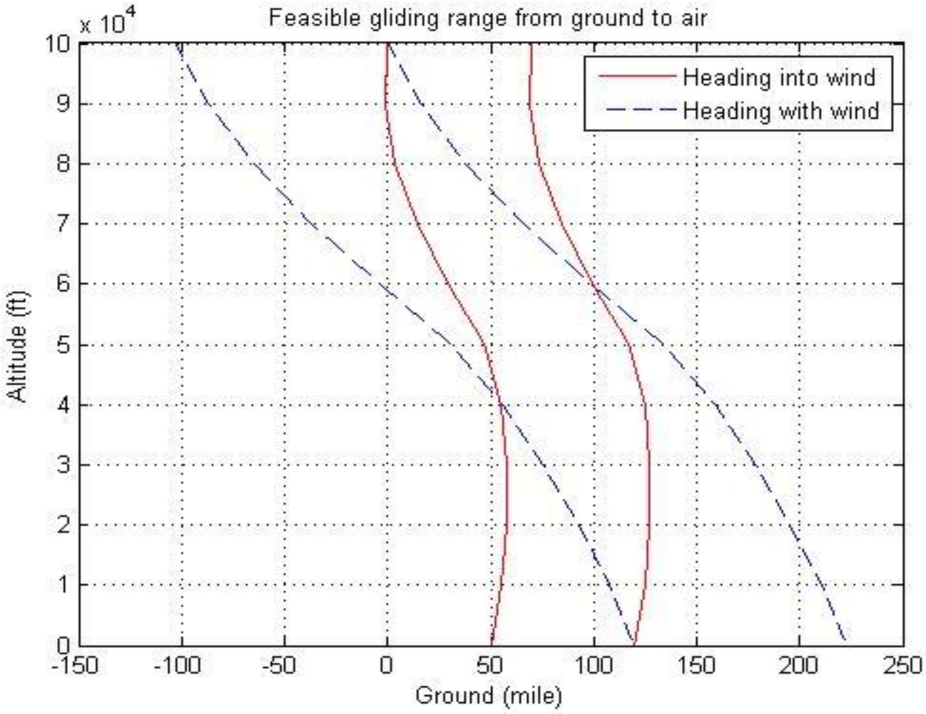


Figure 22Conflicted area calculations of Step 1 and Step 2

Step 4.Feasible gliding range

From Fig.22, feasible range was calculated as in Fig.23, calculating where the two lines cross and using the Maximum and Minimum method, which collects the largest and smallest value that a function takes at a point either within a given neighborhood or on the function domain in its entirety. By using this method, the maximum values between the red line from starting point and the blue line from the ground and the minimum values between the blue line from starting point and the red line from the ground were calculated at each step of altitude [10].

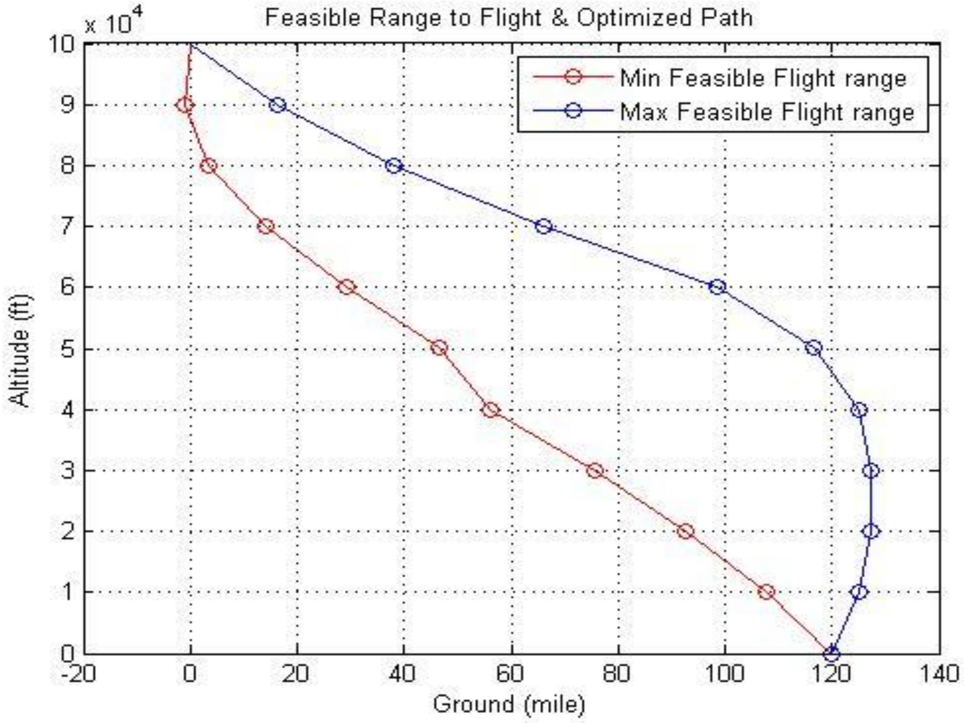


Figure 23 Feasible range of flight

Step 5.Feasible gliding range with robustness (Safety factor)

One more constraint term needed to be added to the result of Step 4: robustness (safety factor). Theoretically, the feasible area of Step 4 satisfies the all required assumptions. However, the line of the flight range was a critical limitation of the requirements. Therefore, robustness terms helped to ensure the reliability of maneuvering, and robustness was added to Step 4's result. From each path, the points that have possibly maximum waypoints were calculated and connected in the same way as step 4. So the waypoints on each step need to be within each step's feasible area. The robustness calculation was performed as follows:

With only one stage considered, one starting point has a feasible gliding range and waypoints, as in Fig.24. The waypoints are counted only as real numbers of position for convenience.

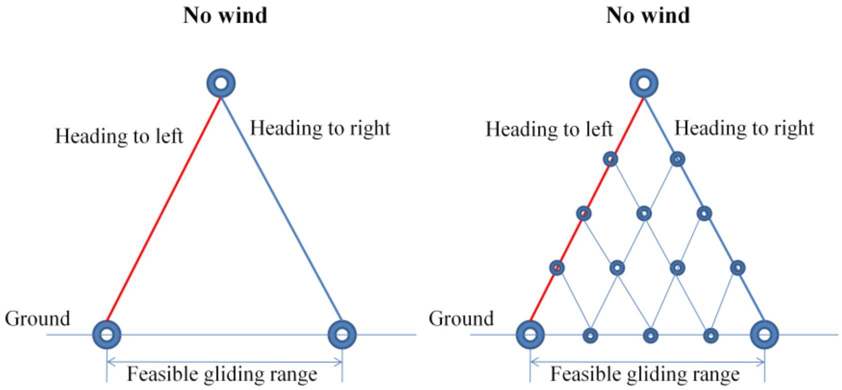


Figure 24 Feasible range with and without waypoints under No wind

When this idea is expanded to the previous calculation result, each stage has a feasible gliding range from the previous step to the next step, as in Fig. 25.

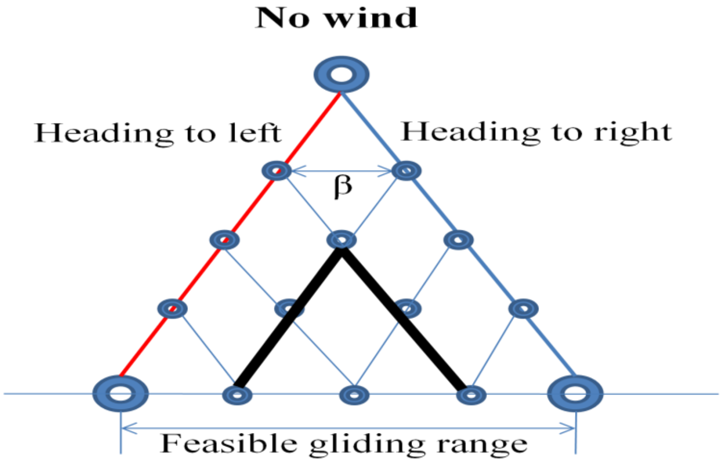


Figure 25 NPW calculation for different starting points

In Fig. 25, several starting points are able to be selected from the previous step. They all have their own gliding range. However, the critical lines have less robustness on left and right boundary. Therefore, the safety factor was applied to contain the maximum robustness inside of the feasible range.

For the robustness calculation, the global robustness method was applied. The method considers the simple abstract robust optimization problem:

$$\max_{x \in X} \left\{ f(x) : g(x,u) \le b, \forall u \in U \right\} \tag{3.1}$$

where U is the set of all possible values of u under consideration.

From Eq.3.1, the safety factor was calculated and the starting range with the same range was determined. After the numerical approach, the robust starting range was defined by subtracting the distance of maximum NPW from both left and right critical lines.

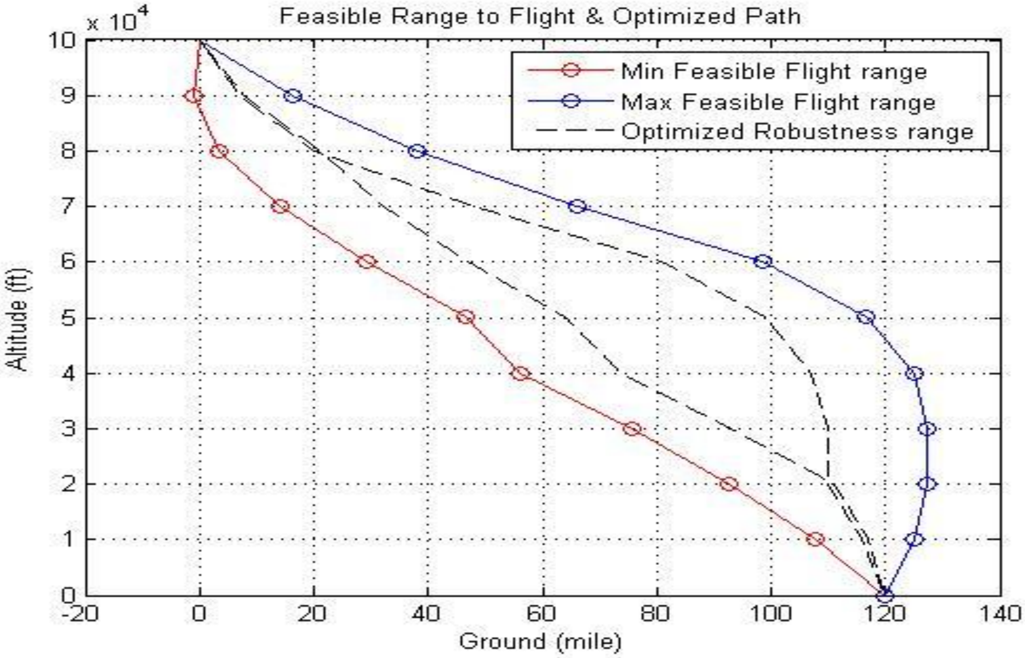
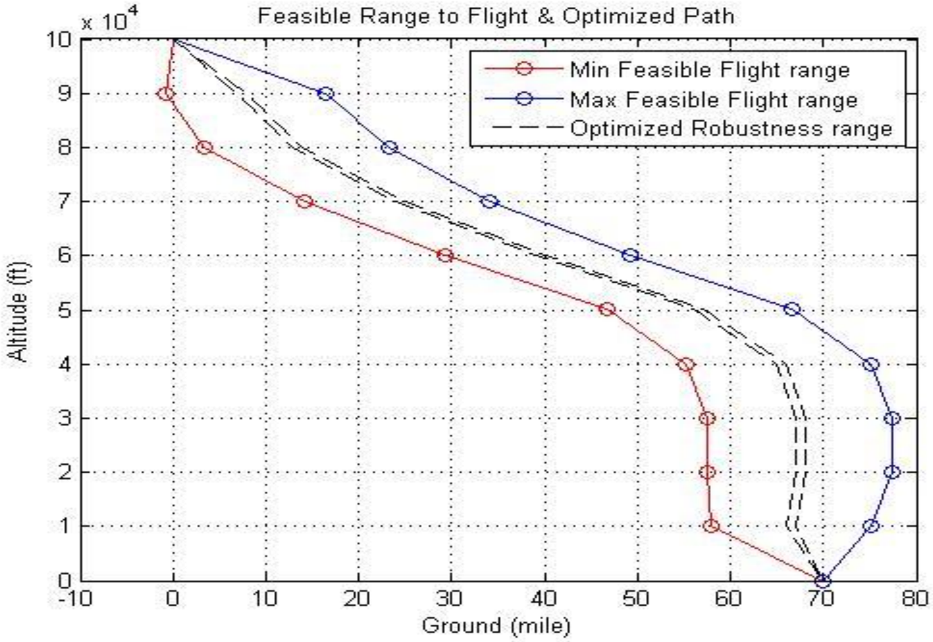


Figure 26: Feasible range with robustness

As a result, Step 4's result was narrowed down in Step 5 to optimize robustness. Obviously, the feasible range of optimized robust range will vary depending on landing position. If a different landing point, close to the gliding limitation of Step 1 was selected, the range grew narroweras in Fig.27. There is thus a relatively limited area in which we can choose waypoints. In this case, the desired landing points are 70 and 180.



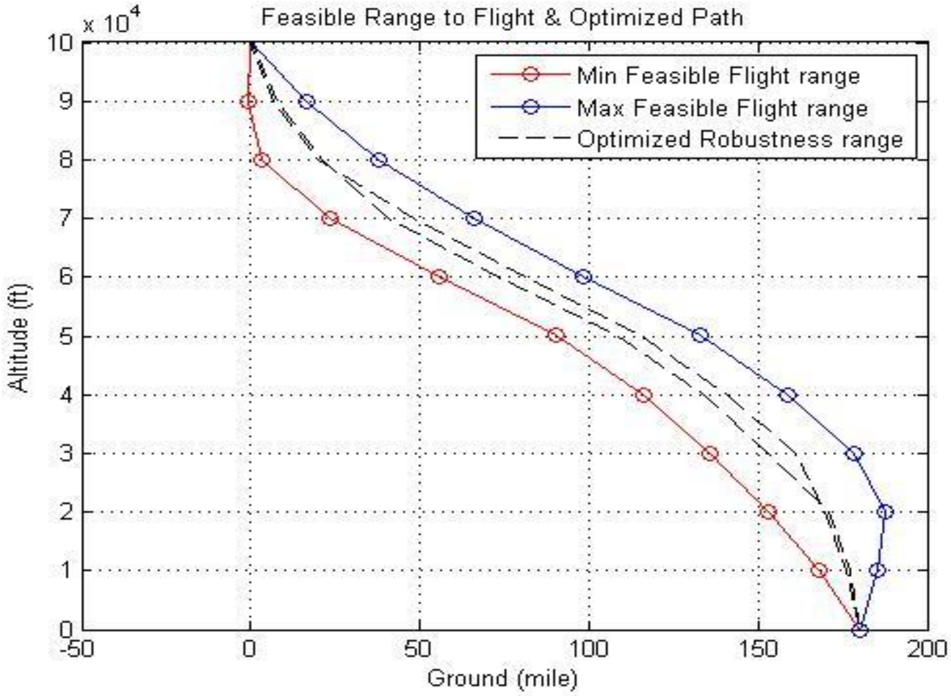


Figure 27: Feasible range with robustness at different landing points: 70 and 180

Step 6.Waypoints decision

As shown in Step 5, each point had a feasible gliding range, even when it was located with in the robustness range. Therefore, the same idea as Step 2 was applied to this step.

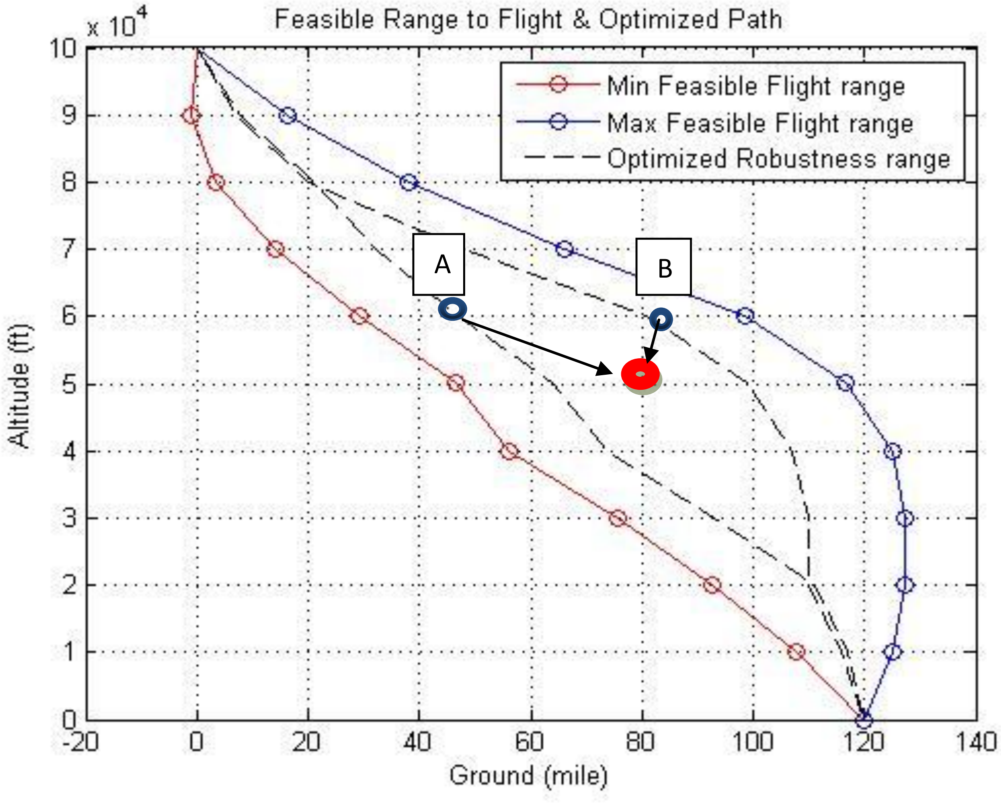


Figure 28Waypoint calculation in feasible range with robustness

If the parafoil was located at point A, it was not able to go the desired waypoint, but it was able to do so from B. That means each successive waypoint will be dependent on previous waypoints. From the parafoil's flight limit calculations, we need to find out the range and identify the waypoints inside the optimized range.

One more optimal objective, flexibility, was added in this step. The idea came from the probability of the possible path from the final destination to starting point as in Fig. 29

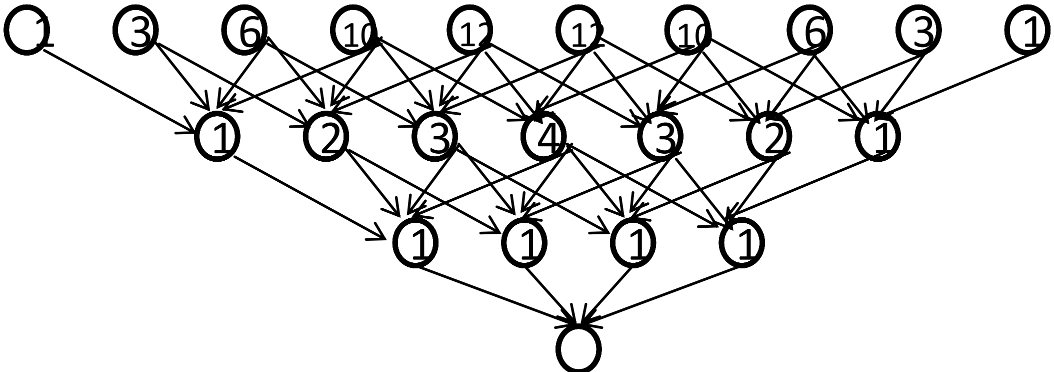


Figure 29 Flow diagram of the probability of the possible path

The numbers are the probability of the path to final destination. After this probability calculation from the destination to the starting points was conducted, the most flexible points at each stage were chosen; they were almost identical to the left line of optimized robustness range (indicated by the red dots in Fig. 29). Therefore, the vehicle was able to obtain more controllability and flexibility on the left line.

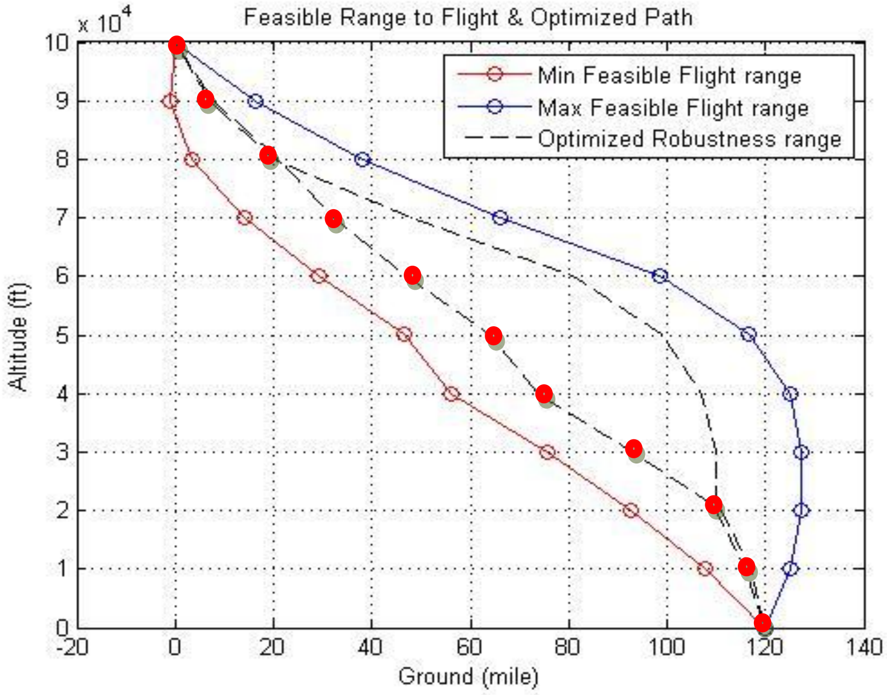


Figure 30 Waypoints in the feasible range with robustness

3.3.2Calculation with wind prediction error

In this section, wind data error was applied to previous calculations; the optimized feasible gliding range was simulated with ±5 mph wind error. The wind data were updated a few hours before the real flight, and is updated as the vehicle ascends collecting wind data, but there will be variations. As a result, the wind error was applied to optimize the feasible range. This is the same as previous calculations, except that the calculation was performed three times with different wind conditions. The given wind data was simulated the same way as in previous steps before adding 5mph wind to the given data and subtracting 5mph from the wind data at each step. As a result, there are three forward paths and three backward paths, plotted below in Figure 30.

Assumptions

- Parafoil velocity: 15mph
- Wind speed error: ±5mph
- Wind speed : [8,15,20,25,40,60,55,45,30,18];(mph)
  (low to high altitude)(1000ft steps)
- Descent speed of each step= 4.4(ft/s) = (3 mph)
- Step = 1000ft
- Flight time of each step = Step/descent speed
- Pitch angle is fixed
- No vertical wind (on Y axis)

Steps 1& 2.Gliding range from starting point to ground and to air

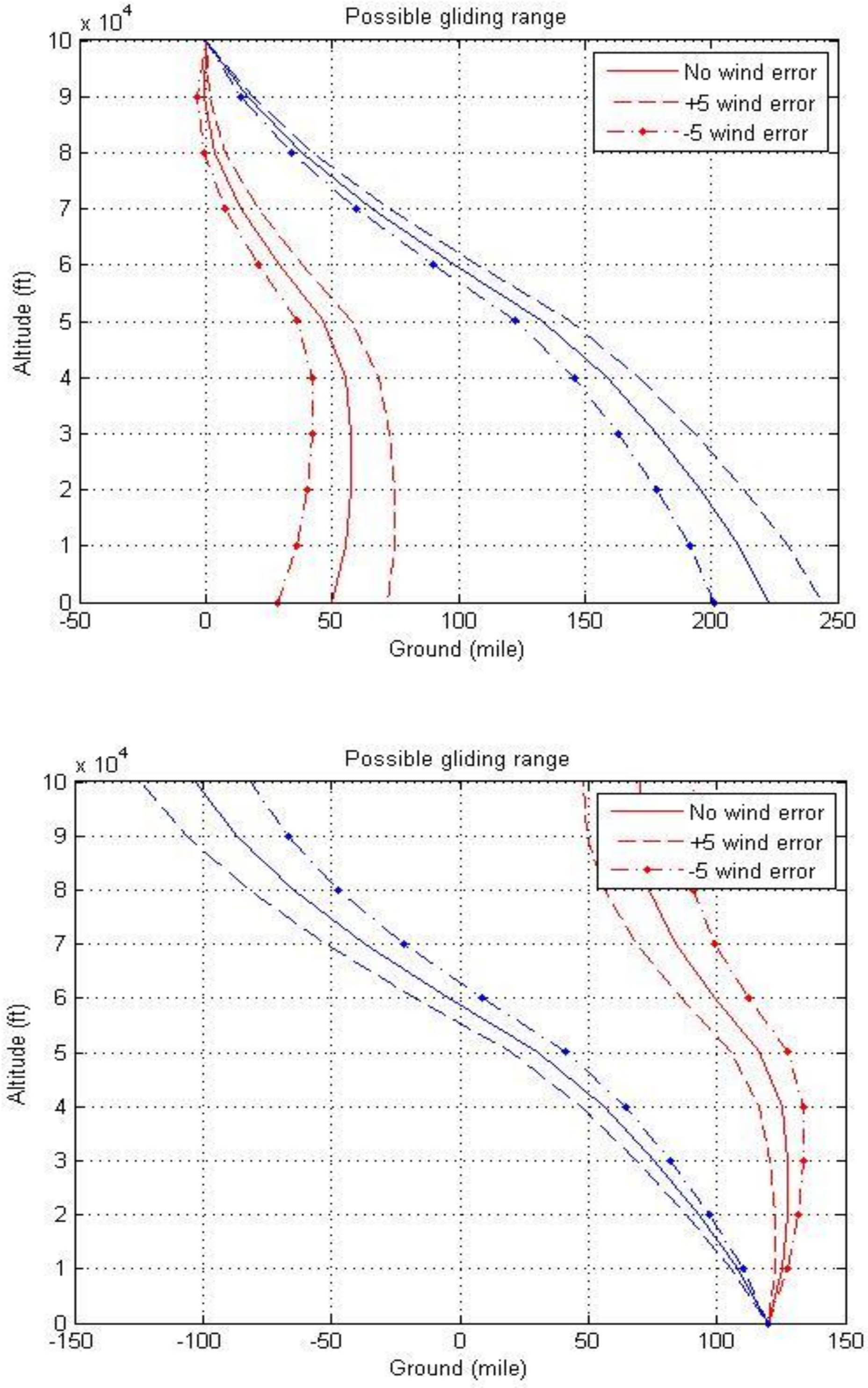
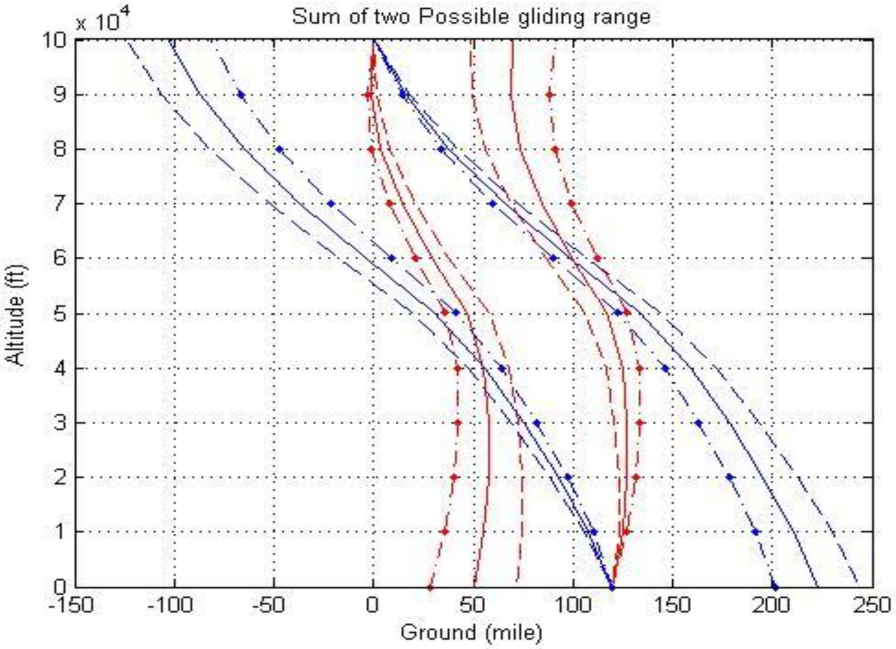


Figure 31 Feasible gliding ranges, air to ground and ground to air, with wind error

Steps 3& 4.Sum of Step 1 and Step 2& feasible gliding range



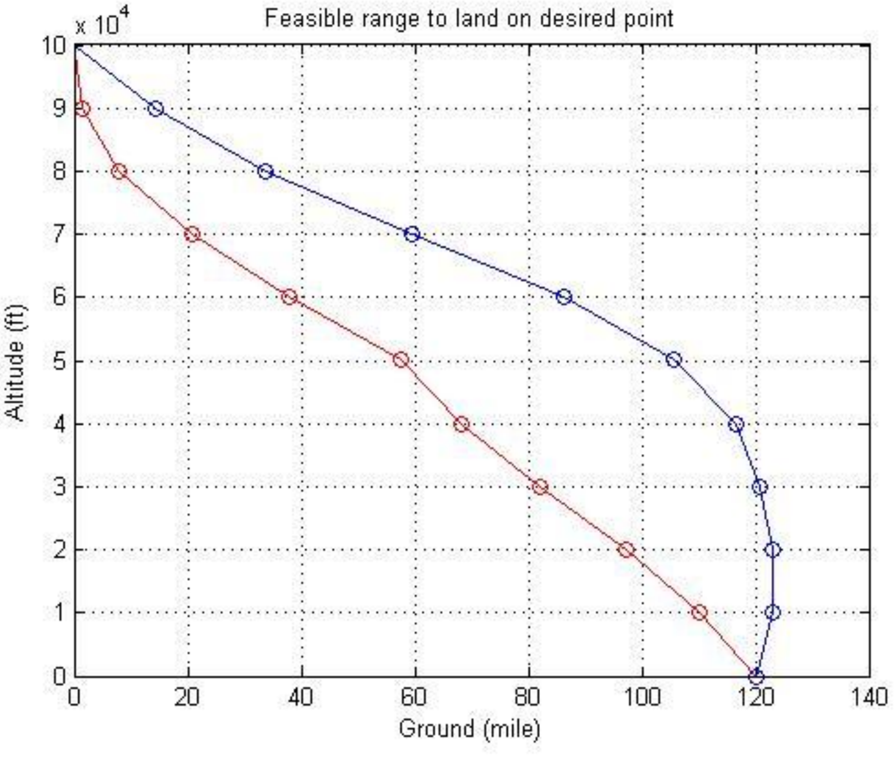


Figure 32 Sum of Steps 1 and 2 and feasible gliding range with wind error

Step 5.Feasible gliding range with robustness

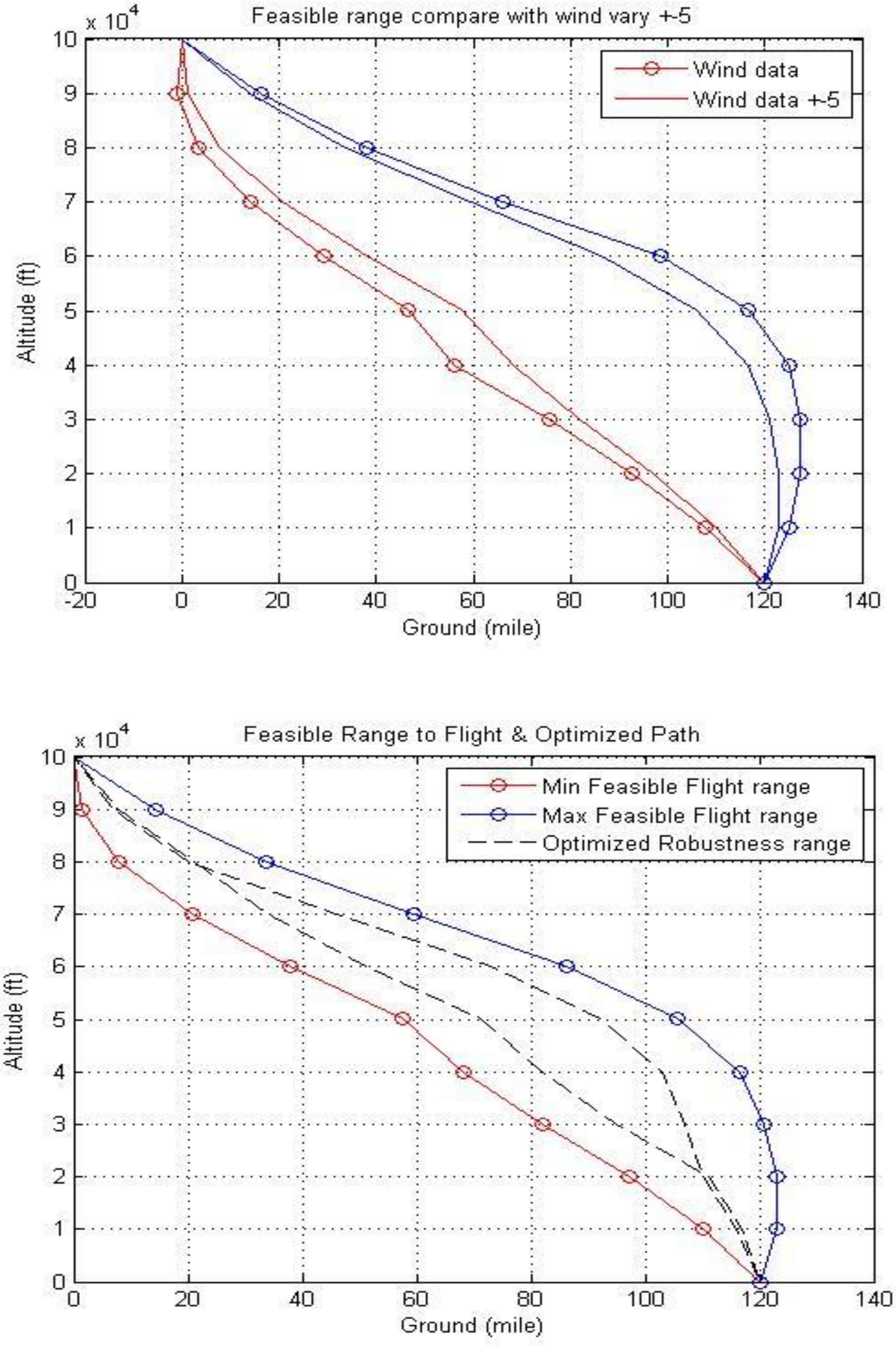


Figure 33 Comparison of feasible ranges to flight and feasible range with robustness

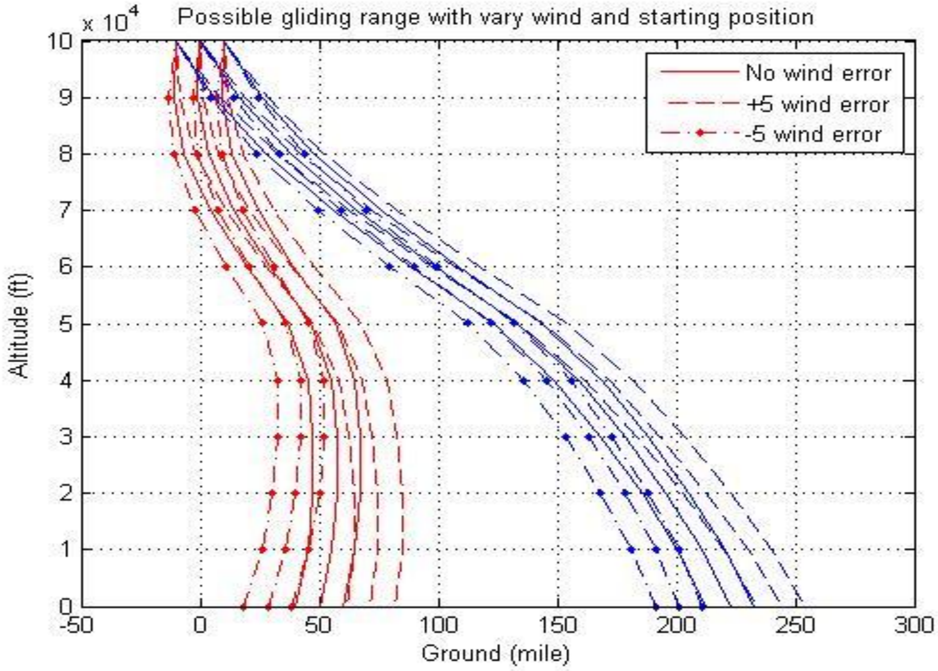
3.3.3Calculation with the wind prediction error and starting point (burst point) error

Our predicted starting point may not align with actual starting points (burst points). Therefore, starting point error compensation was applied in addition to compensation for wind error. Starting error conditions were was applied to the wind error compensation plot above, with the results shown in Figure 33, below. Basically, the wind conditions were not adjusted; only the starting point was changed. In other words, the wind profile was shifted 10 miles to left and right to account for burst point error.

Assumptions

- Parafoil velocity: 15mph
- Wind speed error:±5mph
- Starting point error: ±10 mile
- Wind speed : [8,15,20,25,40,60,55,45,30,18];(mph)
  (low to high altitude)(1000ft steps)
- Descent speed of each step= 4.4(ft/s) = (3 mph)
- Step = 1000ft
- Flight time of each step = Step/descent speed
- Pitch angle is fixed
- No vertical wind (on Y axis)

Steps 1& 2.Gliding range from starting point to ground to air



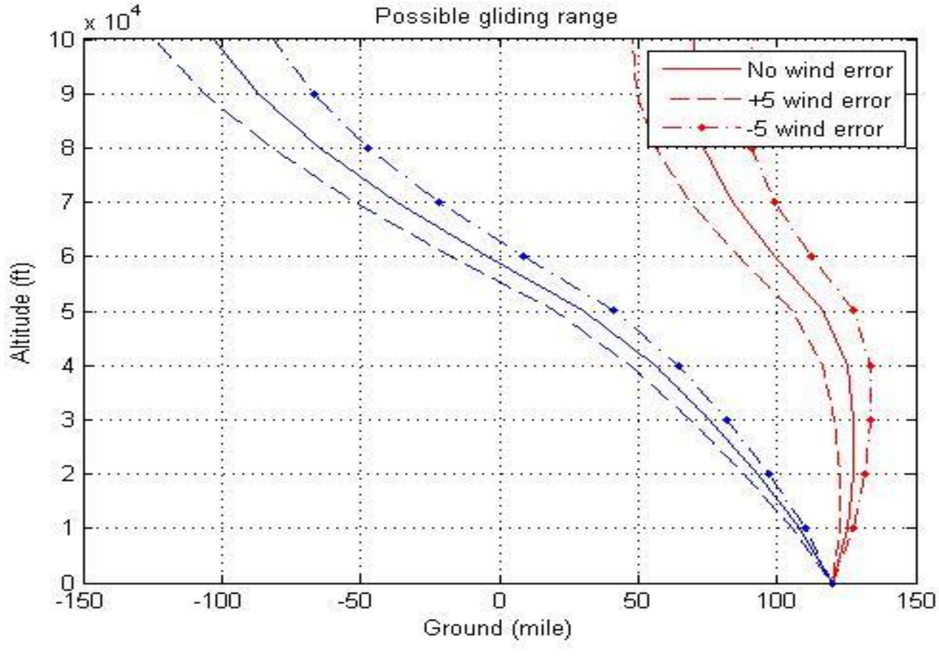
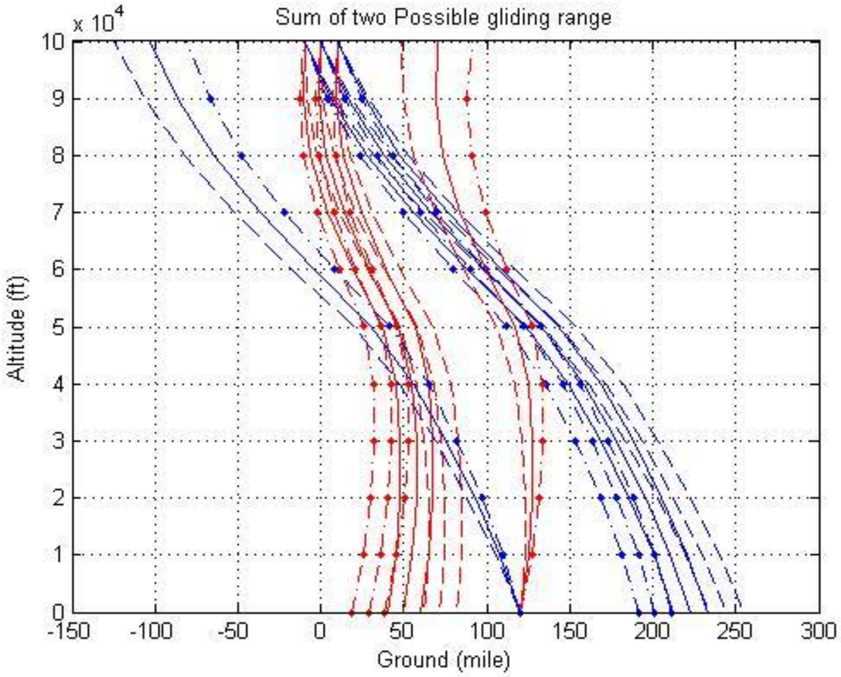


Figure 34 Feasible gliding range, air to ground and ground to air, with wind and burst point errors

Steps 3& 4.Sum of Step 1 and Step 2& feasible gliding range



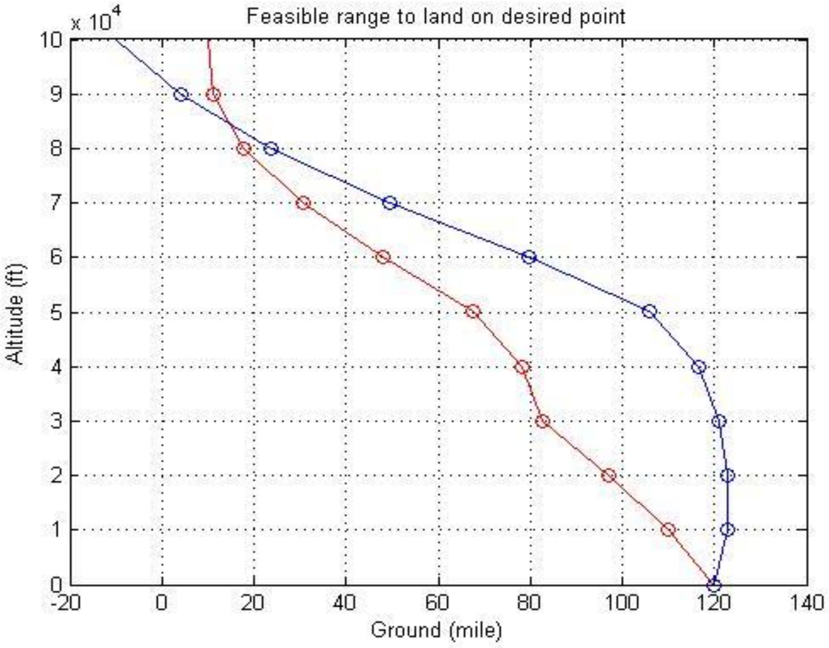
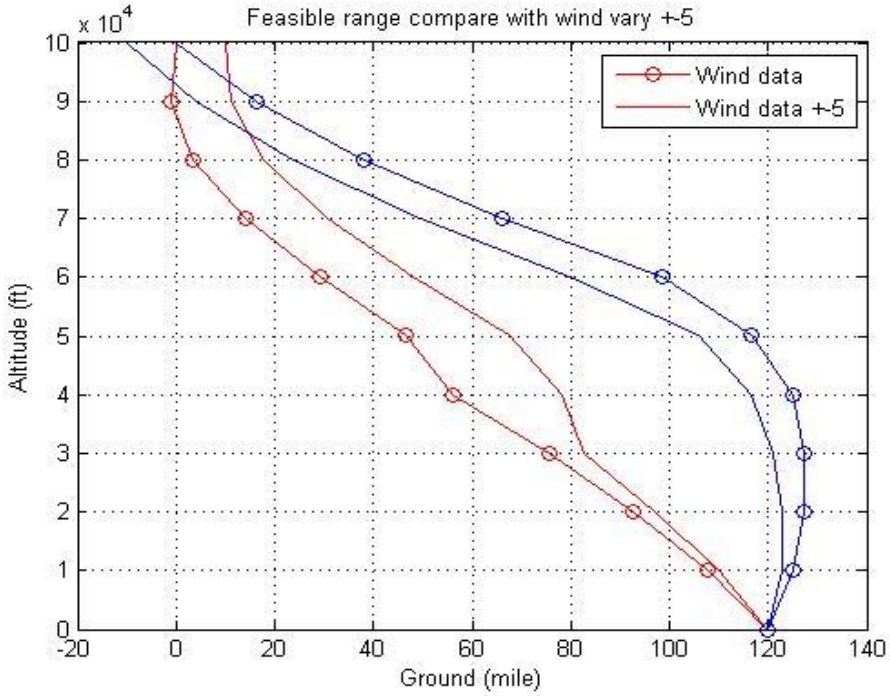


Figure 35 Sum of Steps 1 and 2 and feasible gliding ranges, with wind and burst point errors

Step 5.Feasible gliding range with robustness



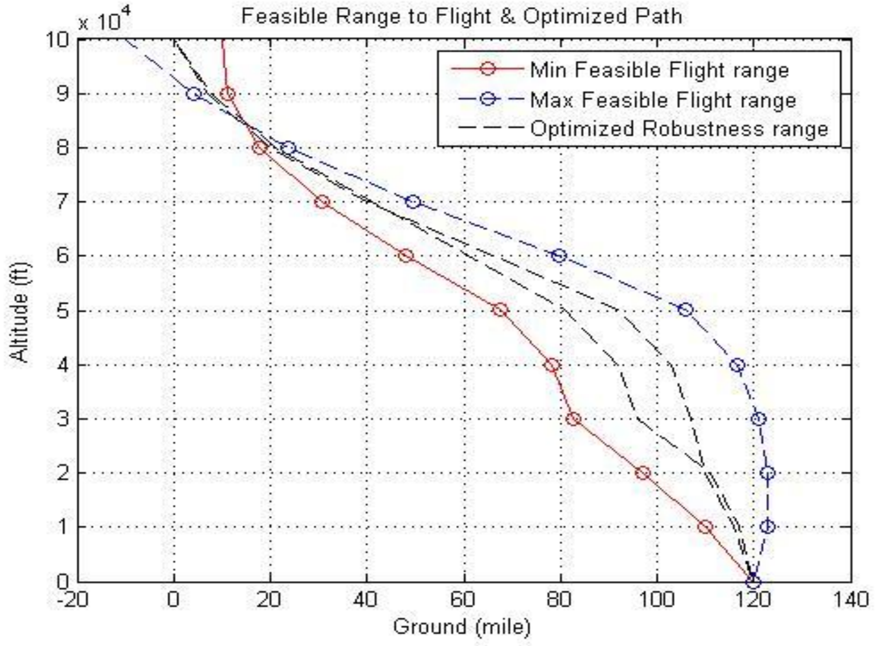


Figure 36 Comparison of feasible ranges to flight and feasible range with robustness, with wind and burst point errors

3.3.4Calculation with the wind prediction error, starting point (burst point) error, and varying parafoil speed compensation

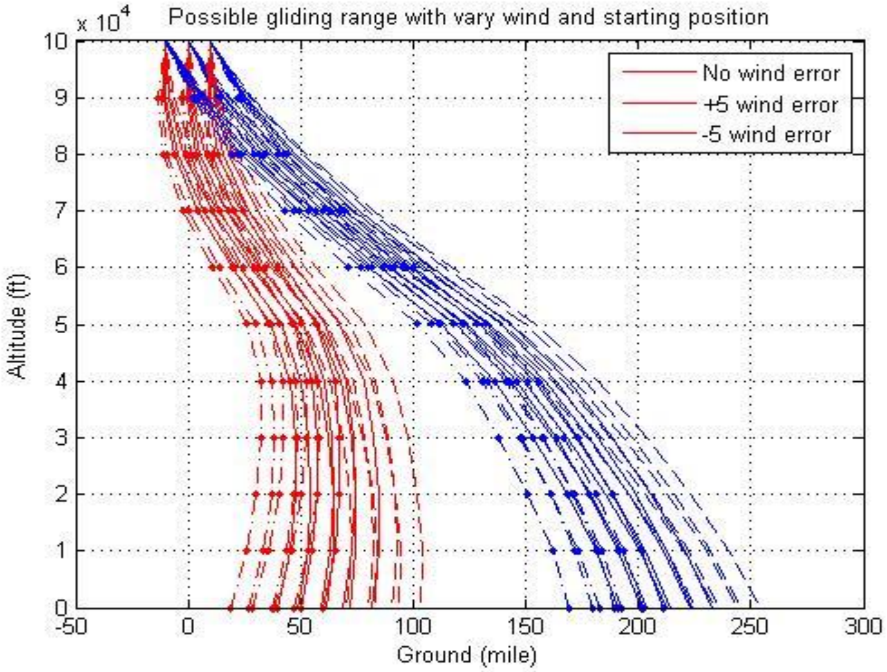
Optimized feasible gliding range was simulated with wind error compensation.Also, the predicted burst point could differ, so compensation for burst point error was applied along with wind error compensation.One more factor that should be considered is parafoil speed; it can vary in several conditions. Therefore, the velocity variation was assumed to be ±5mph and simulated. This is the same as in previous calculations, save that the calculations were performed three times with different wind conditions and twice more for different parafoil velocities (-3mph and -5mph). The parafoil velocity was dependent on the angle of attack and there was less possibility to speed up the maximum parafoil velocity, so only two lesser-speed cases were considered. The given wind data was simulated with same way as in previous steps, including adding 5mph of wind to the given data and subtracting 5mph from the wind data on each step. As a result, there are three forward paths and 3 backward paths, plotted below in Fig. 36.

Assumptions

- Parafoil velocity: 15mph
- Wind speed error: ±5mph
- Starting point error: ±10mile
- Parafoil velocity error: ±5mph
- Wind speed : [8,15,20,25,40,60,55,45,30,18];(mph)

  (low to high altitude)(1000ft steps)
- Descent speed of each step= 4.4(ft/s) = (3 mph)
- Step = 1000ft
- Flight time of each step = Step/descent speed

Steps 1& 2.Gliding range from starting point to groundto air



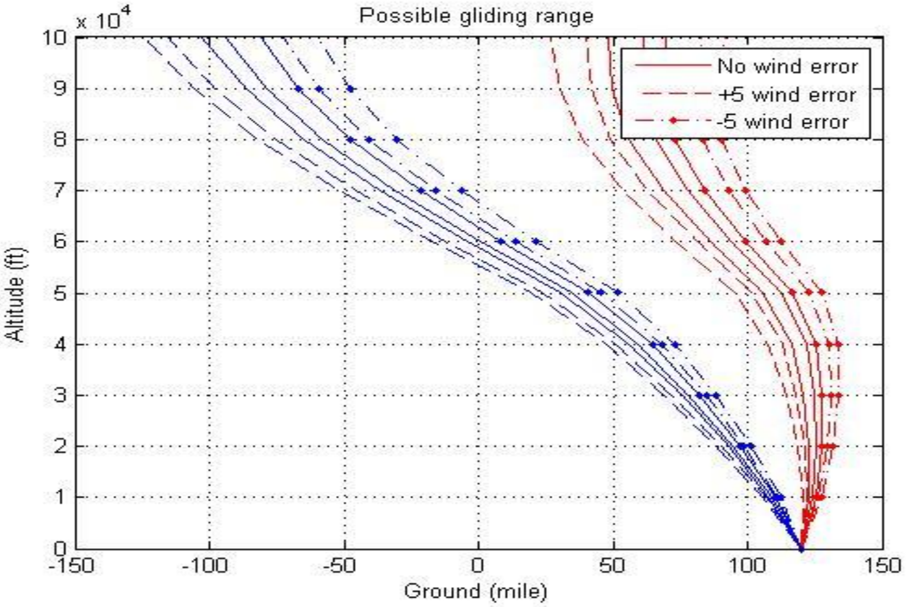
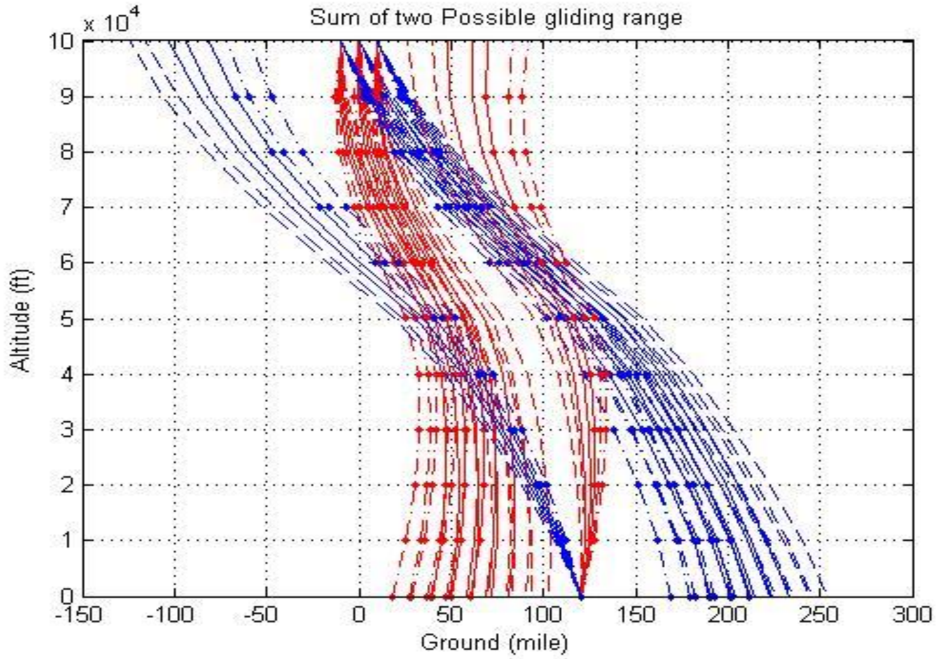


Figure 37 Feasible gliding ranges, air to ground and ground to air, with wind speed, burst point and velocity error compensation

Steps3& 4.Sum of Step 1 and Step 2& feasible gliding range



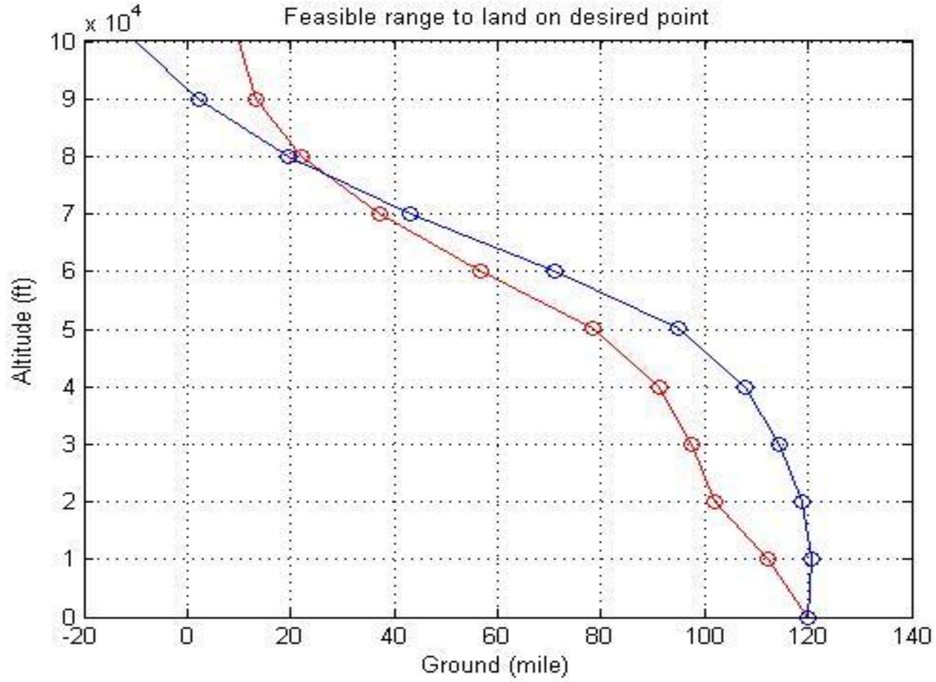


Figure 38 Sum of Steps 1 and 2 and feasible gliding ranges, with wind speed, burst point and velocity error compensation

Step 5.Feasible gliding range with robustness

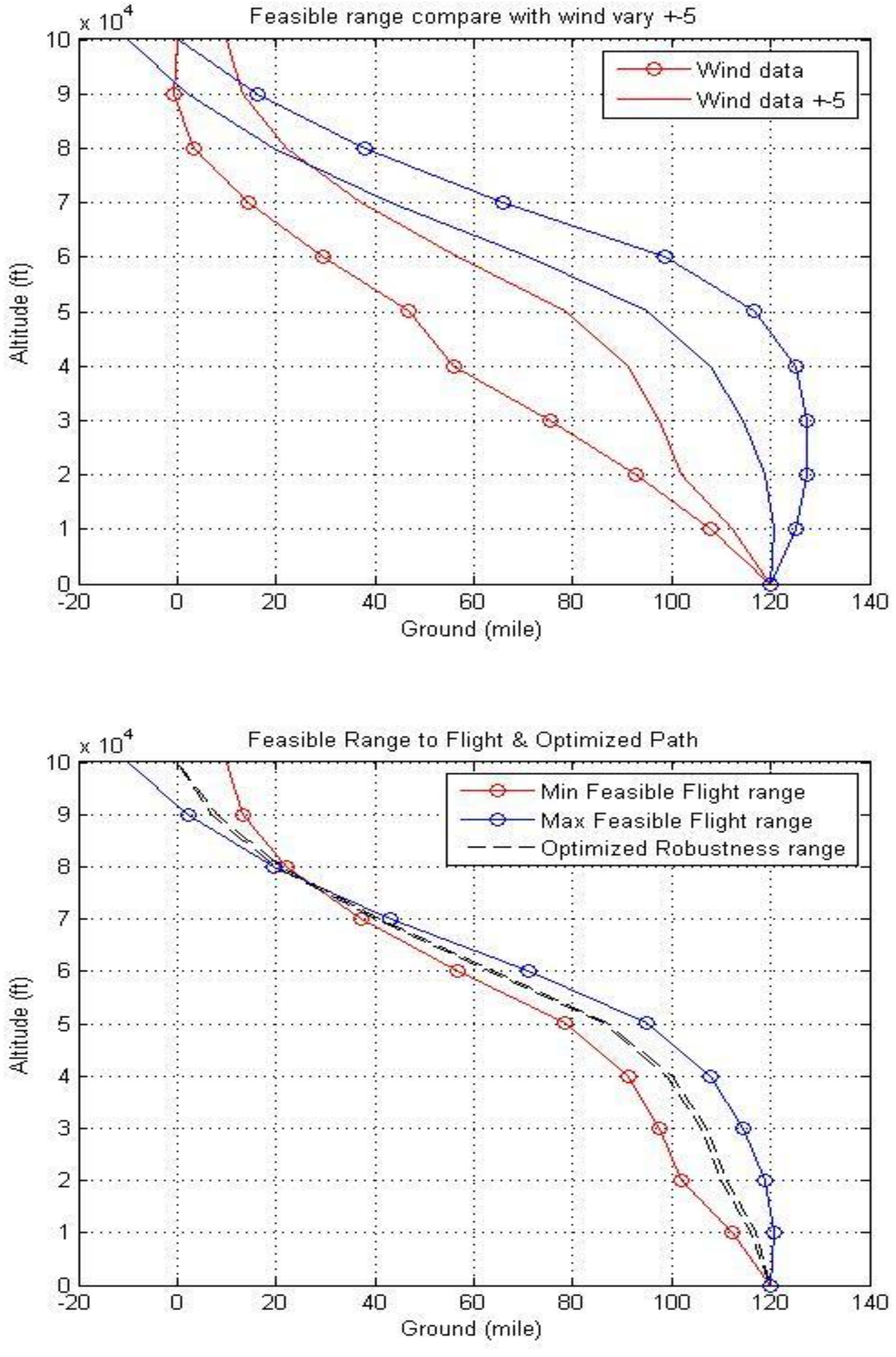


Figure 39 Comparison of feasible ranges to flight and feasible range with robustness, with wind speed, burst point and velocity error compensation

3.4 Feasible gliding range calculation in three dimensions

3.4.1 One wind direction (from west to east)

In this section, the method of 2D optimized feasible gliding range calculation was expandedinto three dimensionsunder real flight condition. The concepts and method are same as the 2D case, but 3Doptimized feasible gliding range was simulated with 6to 9different heading angles under one wind direction and varying wind speed condition. The 6 angles are 0, 45, 135, 180, 225, and 315 degrees with; the wind direction was set to 0 degrees.

Step 1.Gliding range from Starting point to ground

The same basic method used in 2D was applied to determine the feasible gliding range, extended to include 4 more headings for the vehicle to check the Y-axis range.After checkingit, the desired X and Y landing points (latitude and longitude)wereable to be chosen within the ground range.

Assumptions

- Parafoil velocity: 15mph
- Starting point: (0, 0, 10000)
- Desired landing point: (120, 20, 0)
- Wind speed: [8,15,20,25,40,60,55,45,30,18](mph) (low to high alt.)(1000ft steps)
- Descent speed of each step:4.4ft/s = (3 mph)
- Step= 1000ft
- Flight time of each step= Step/descent speed
- Pitch angle is fixed

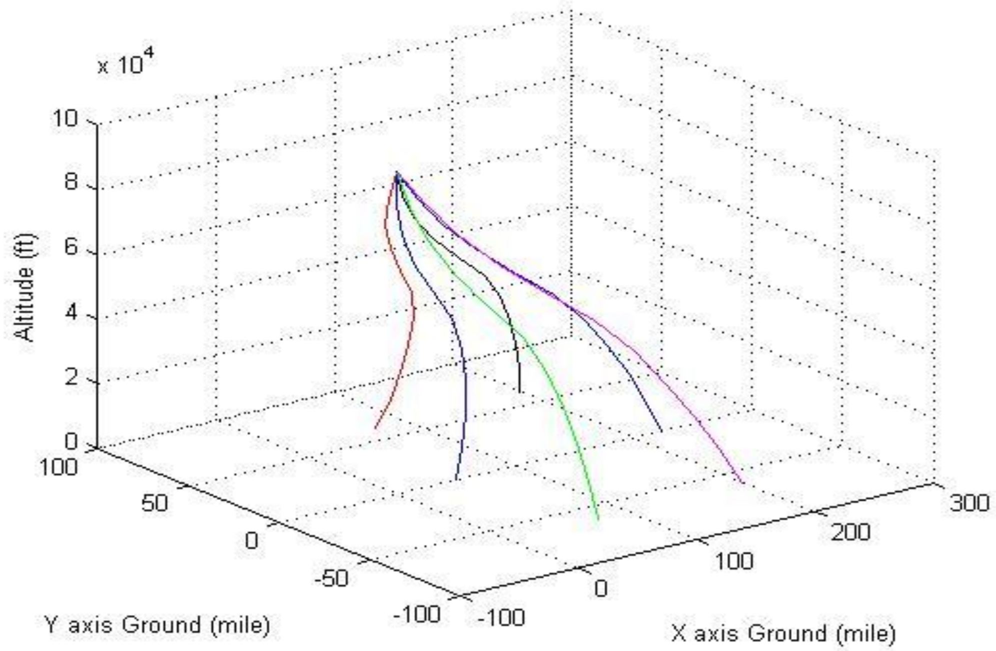


Figure 40 Feasible gliding range, air to ground

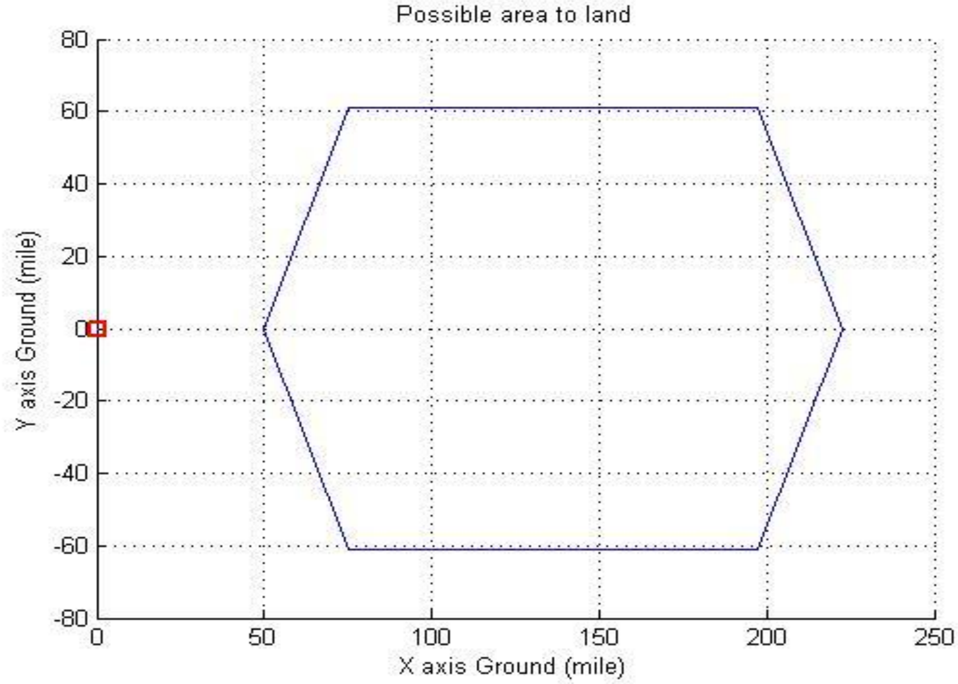


Figure 41 Feasible landing area

Step 2.Gliding range from desired landing point to sky

With the desired landing points determined, the inverse feasible range check, ground to sky,wasperformed to find out the limited area allowing the vehicle to land desired points from. This wasalso checked for 6 different heading angles, as in Step 1.

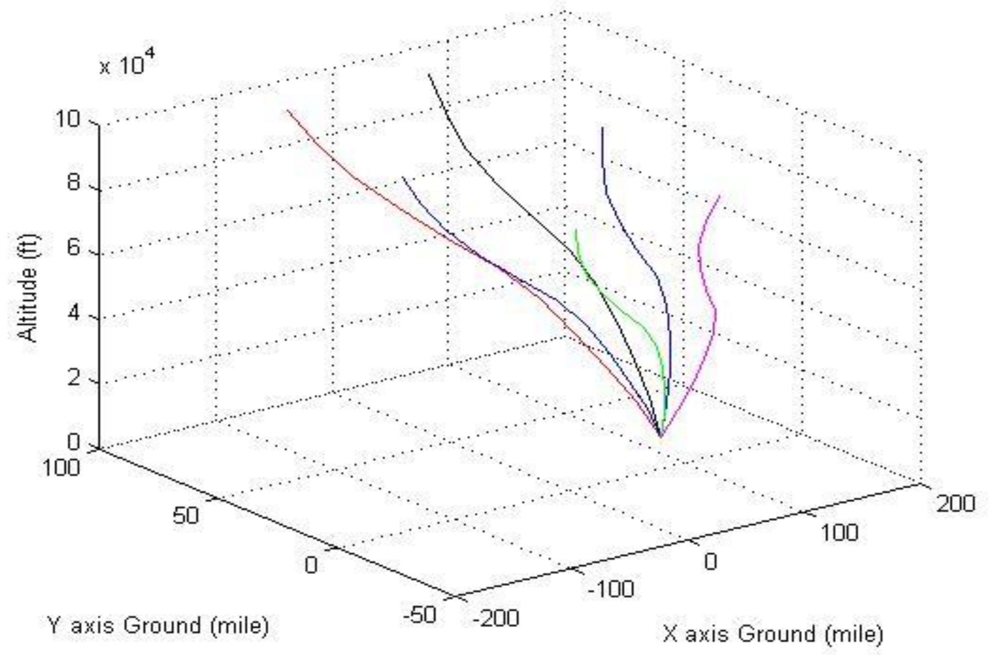


Figure 42 Feasible gliding range from ground to air

Step 3.Sum of Steps 1 and 2

Finding the overlapping area between Steps 1 and 2 is important. In2D, this was not so difficult, but in3D, consideration of the Y axis was added and generatedconfusion because the 0 and 180 degree paths share noarea withinthe range, and the other paths do not intersect. The method to find the overlapped area will be discussed in Step 4.

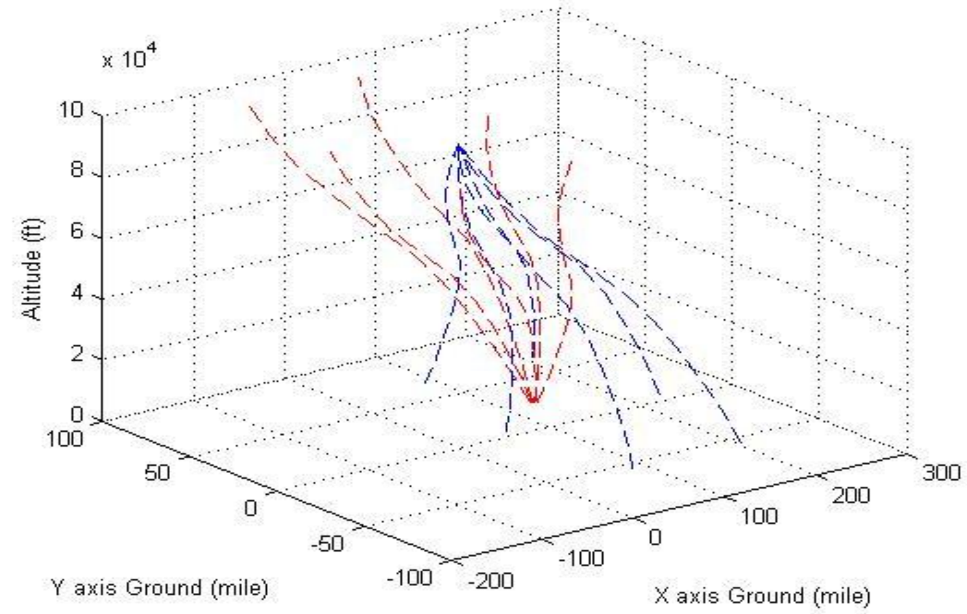


Figure 43 Feasible gliding range from ground to air

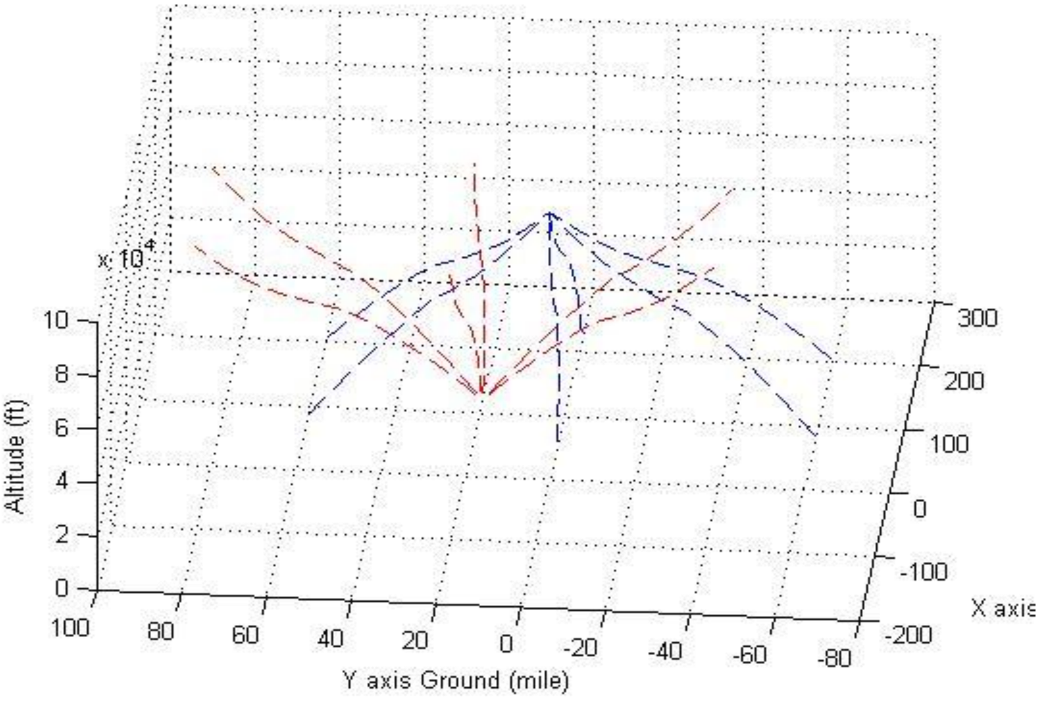


Figure 44 Feasible gliding range from ground to air (Front view)

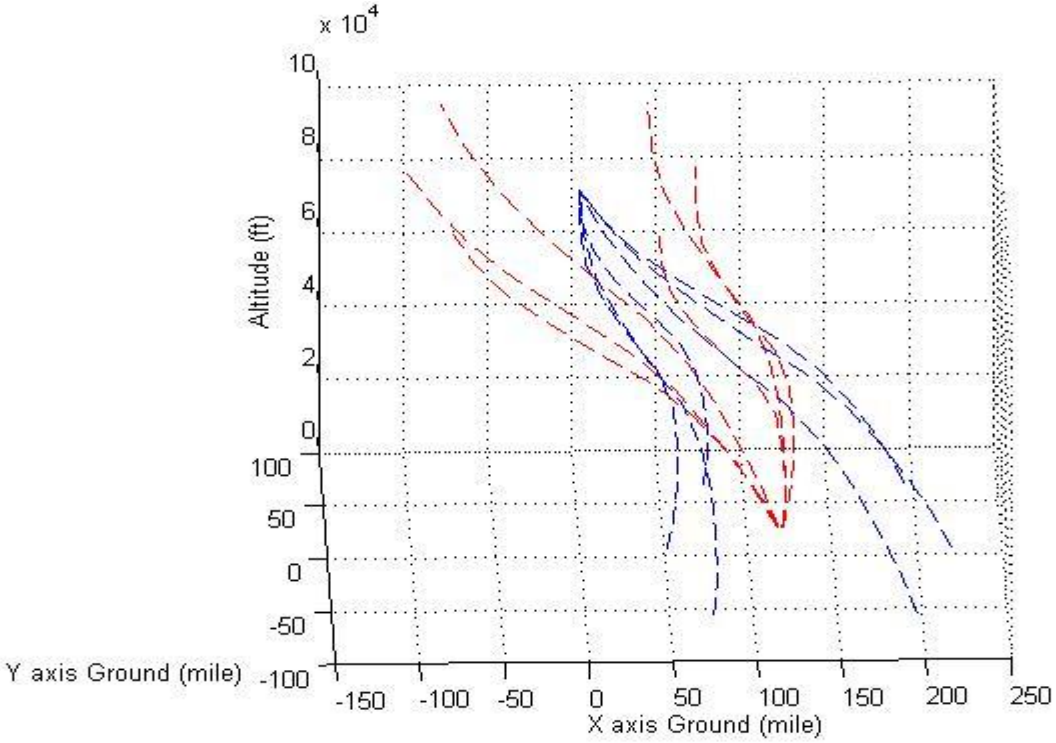


Figure 45 Feasible gliding range from ground to air (Side view)

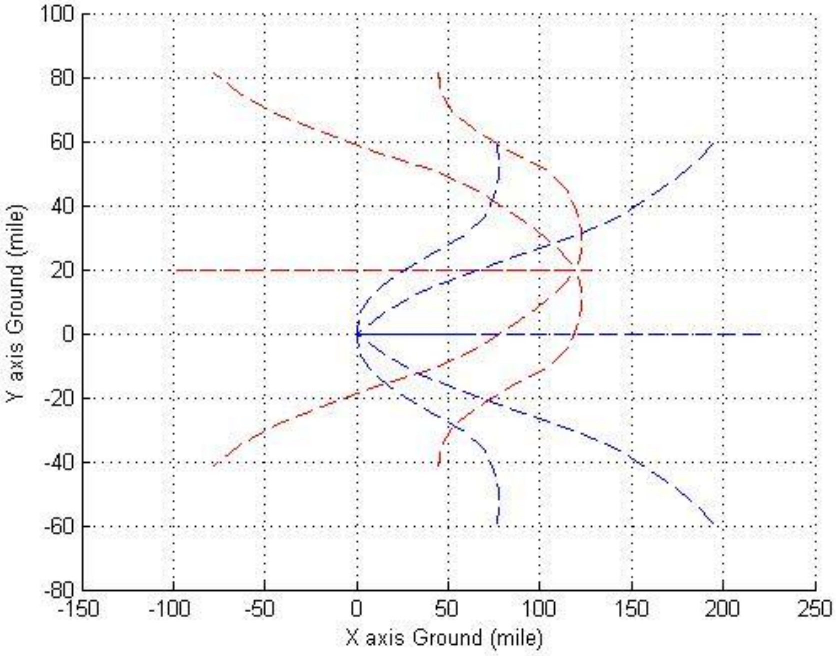


Figure 46 Feasible gliding range from ground to air (Top view)

Step 4. Finding feasible gliding range

Because of the difficultiesin finding the intersection between Steps 1 and 2, two different methods to determine range were employed. The interpolation method was used on the 0 and 180 degree headings to find the track from starting point to landing point, and the Maximum and Minimum method was used on the other headings. The result showeda reasonable plot of range and satisfied all numerical verifications.

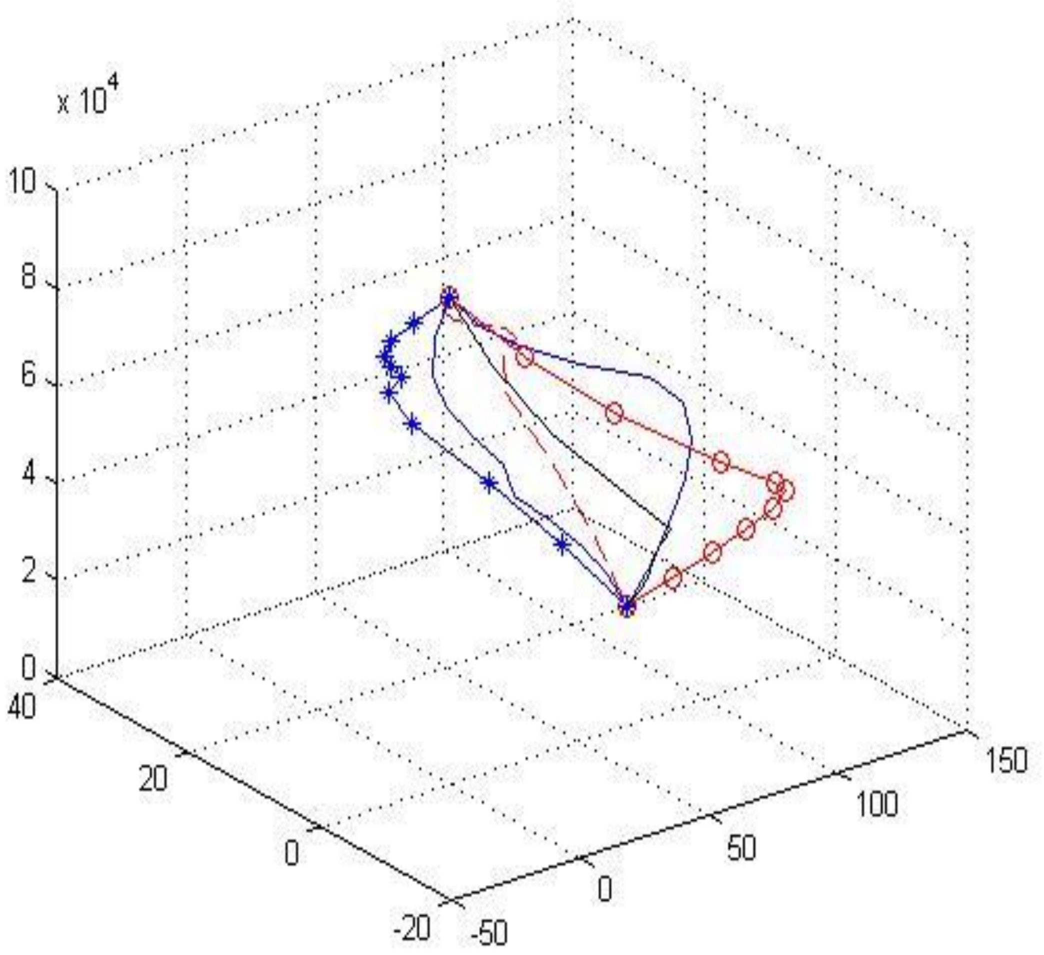


Figure 47 Feasible range of flight

Step 5.Feasible gliding range ateach step of altitude

In this step, each path was plotted to show the feasible waypoint area foreach step. On the bottom of the plot (X-Y plane), the area was drawn with different colors,and a 3D plot identifiedhow much area the vehicle has to maneuver.

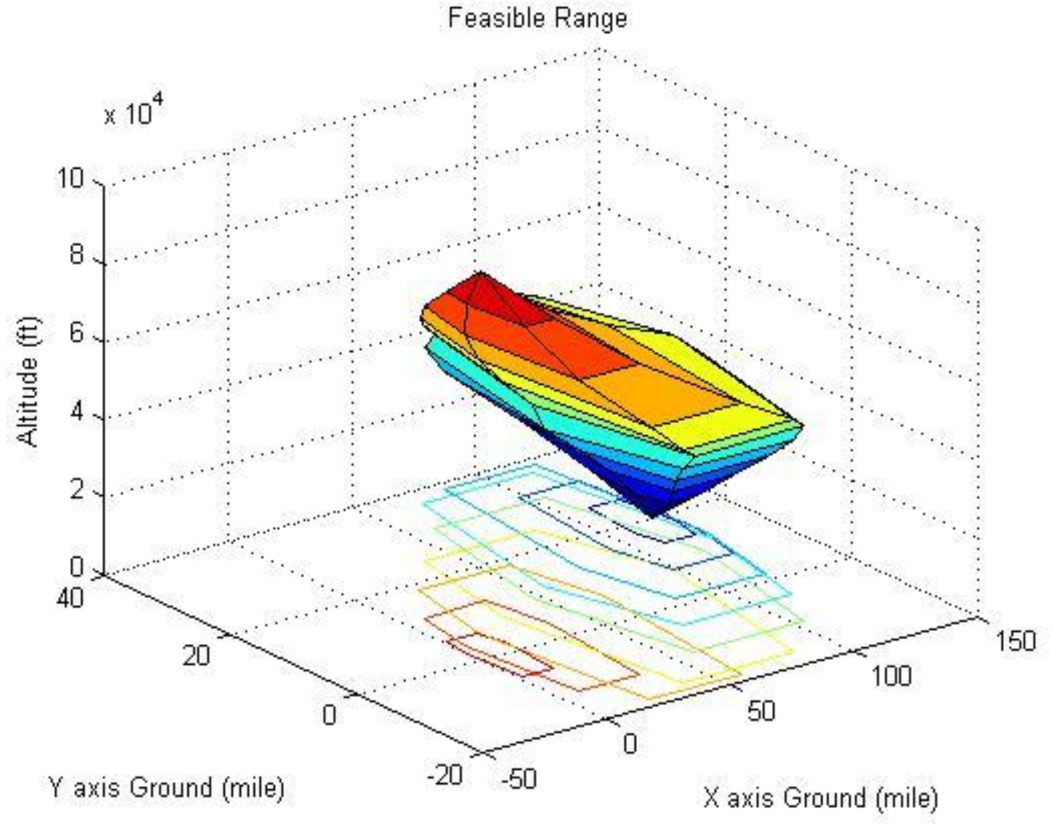


Figure 48 Feasible range of flight

Step 6.Optimized feasible gliding range

From Step 5's result, the optimized technique was added. From each couple ofpaths(0 and180, 45 and 215, 135 and 315), the points that have maximum NPW were calculated and connected in thesame way asStep 5.So the way points foreach step of altitude need to be determined in that step’s area.

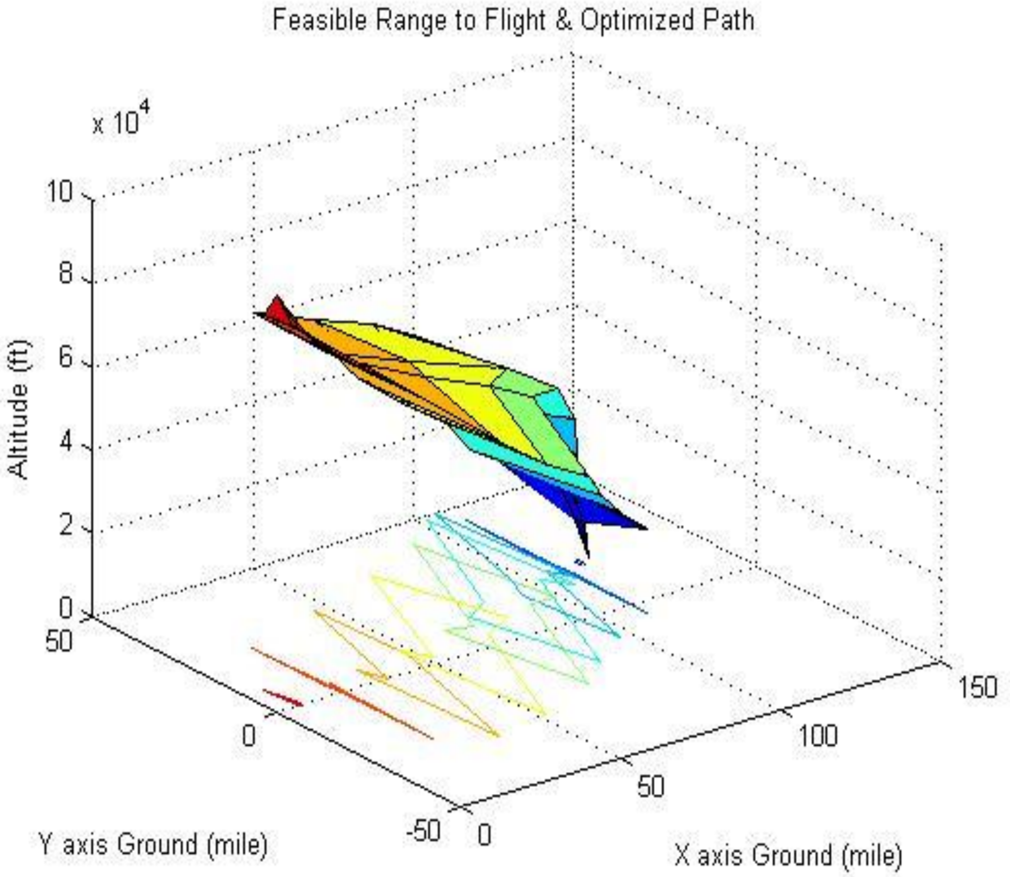


Figure 49 Feasible range of flight with robustness

3.4.2 Variable wind direction

The most important and different characteristic between 2D and 3D case was the wind direction. In real flight situations, there are varying wind directions depending on altitude and position. Therefore, this wind direction should be seriously considered in 3D cases. The wind data which was obtained from the Air Pressure Lab (APL) provides only 2D wind direction at each altitude, so the vertical direction wind was ignored in this calculation. Basically, thecalculation was based on previous calculation methods and thevarying wind direction term was added at each step of altitude. Figures 49 through 61, below, display the results.

Assumptions

- Parafoilvelocity: 15mph
- Starting point: (0, 0, 10000)
- Desired landing point: (120, 20, 0)
- Wind speed: [8,15,20,25,40,60,55,45,30,18]mph (low to high alt.)(1000ft steps)
- Winddirection= [280,290,270,300,300,330,270,280,270,300]degrees(low to high alt.)
- Descent speed of each step:4.4ft/s = 3 mph
- Step= 1000ft
- Flight time of each step= Step/descentspeed
- Pitch angle is fixed
- No vertical wind (on Y axis)

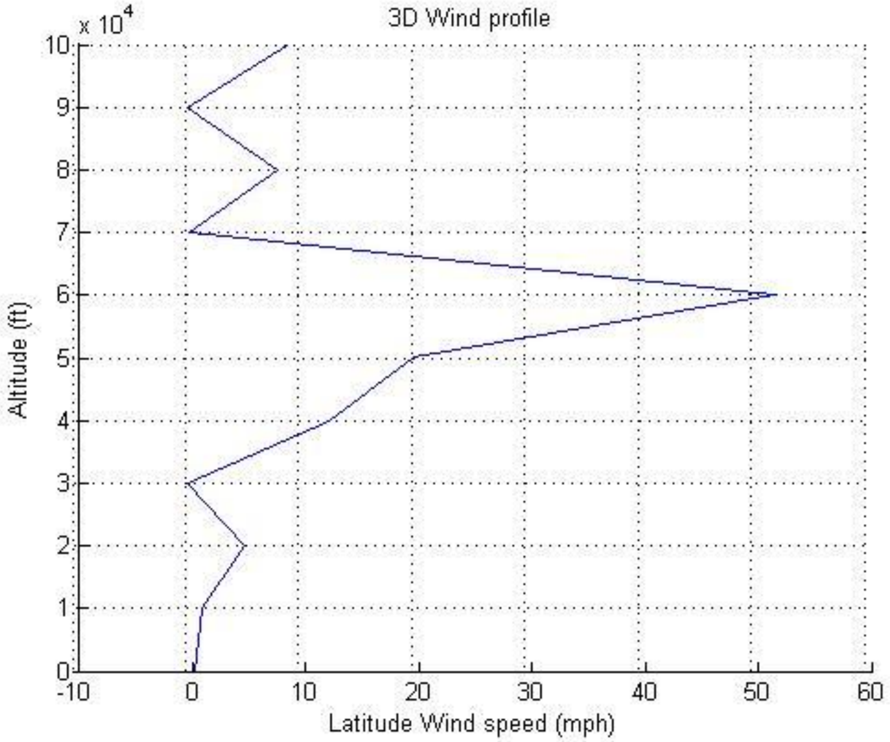


Figure 50 Wind profile (Side)

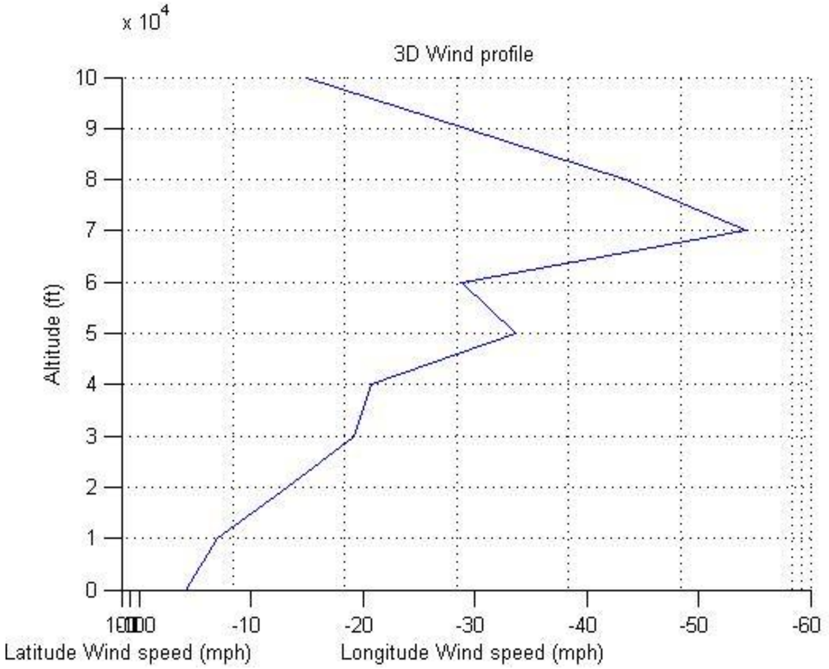


Figure 51 Wind profile (Side)

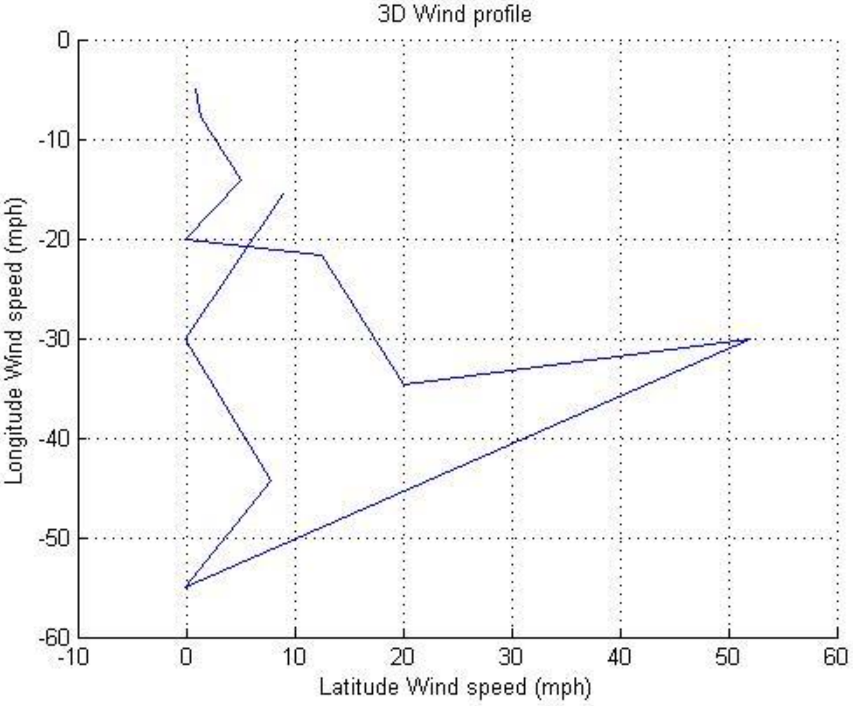


Figure 52 Wind profile (Top)

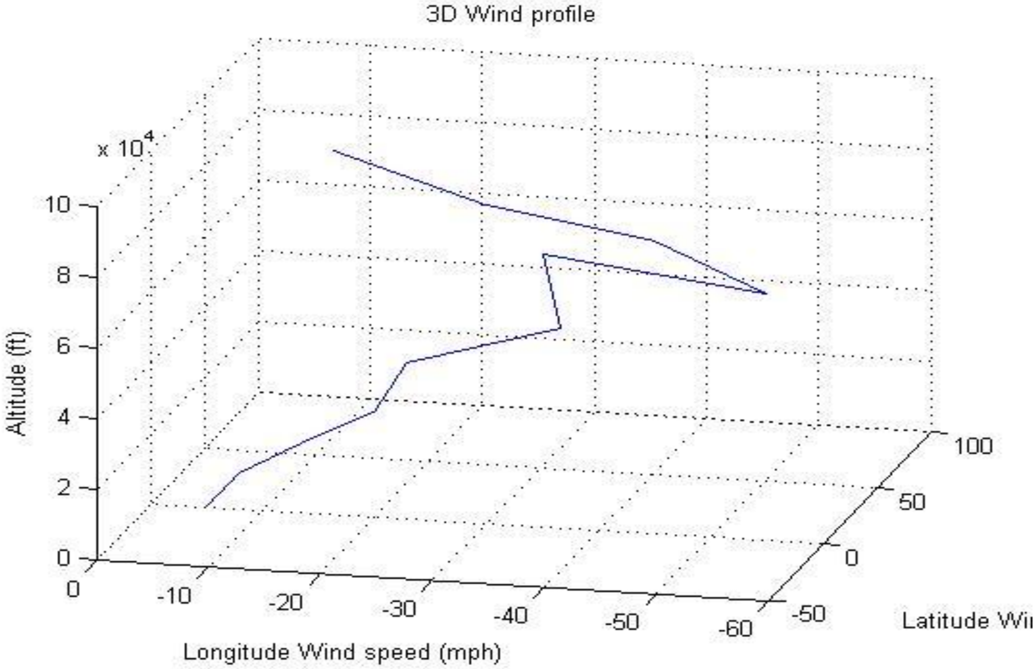


Figure 53 Wind profile (3D)

Step 1.Gliding range from starting point to ground

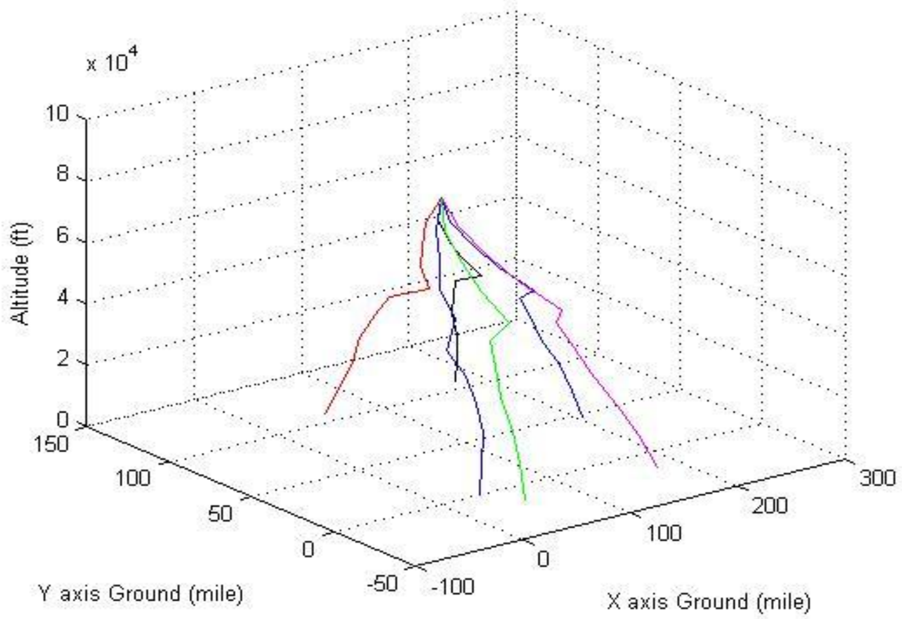


Figure 54 Feasible gliding ranges, air to ground with vary wind direction

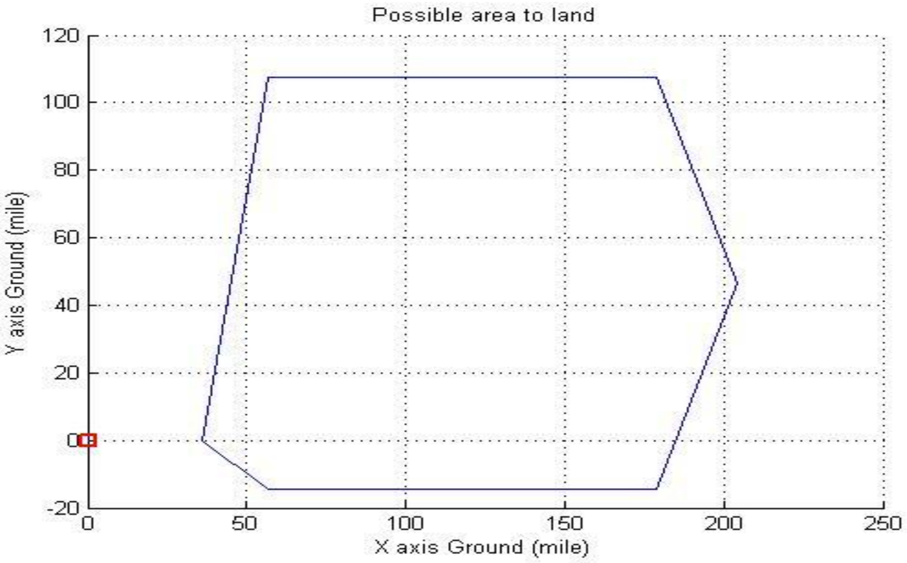


Figure 55 Feasible landing area with vary wind direction

Step 2.Gliding range from desired landing point to sky

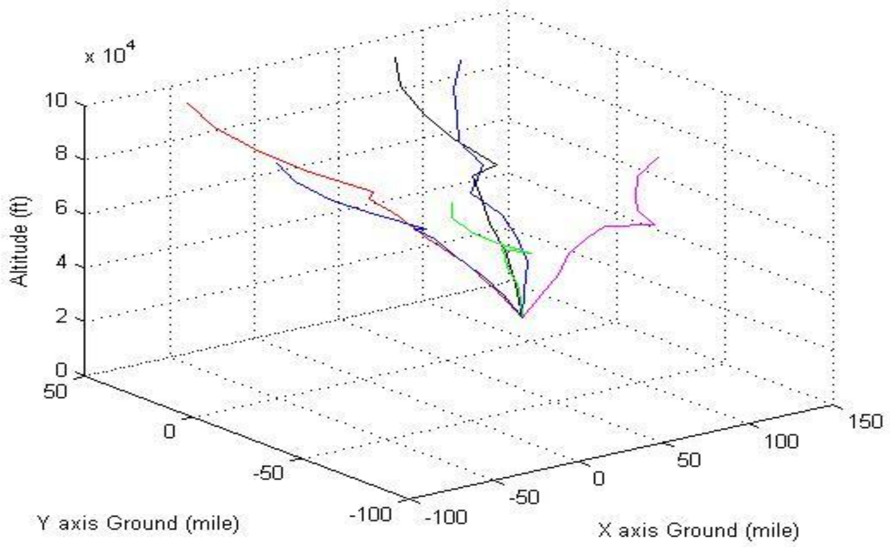


Figure 56 Feasible gliding ranges, ground to air with vary wind direction

Step 3.Sum of Steps 1 and 2

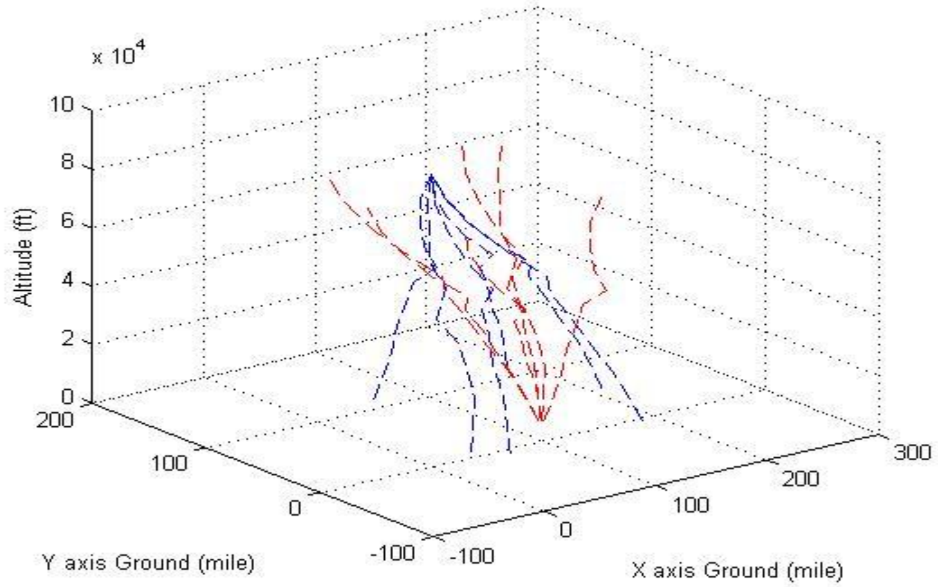


Figure 57 Sum of Step 1 and Step 2 with vary wind direction (3D)

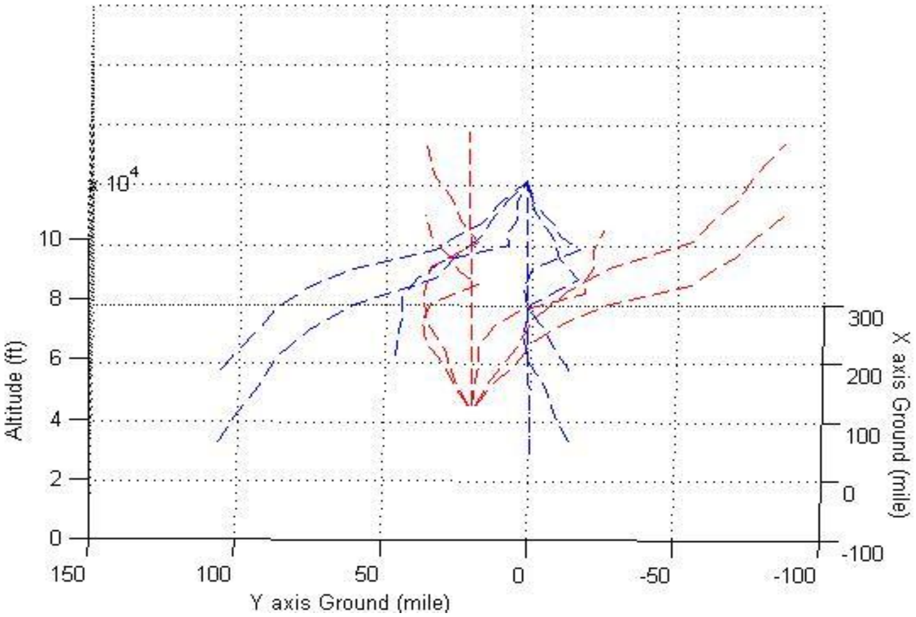


Figure 58 Sum of Step 1 and Step 2 with vary wind direction (Front)

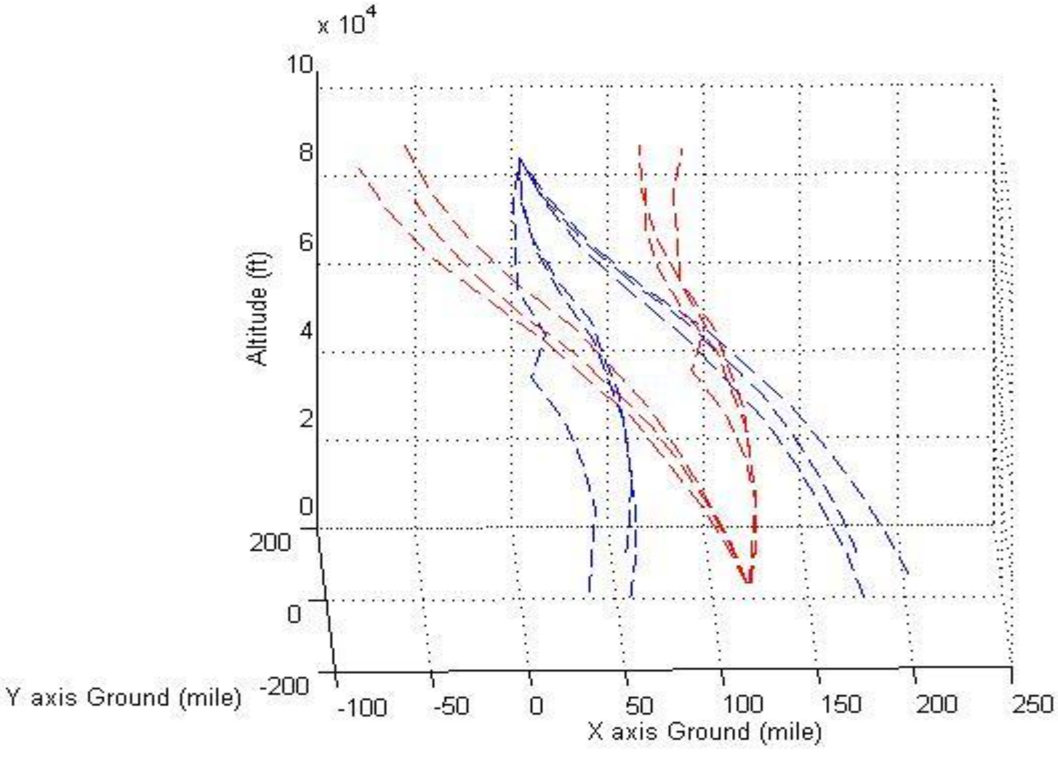


Figure 59 Sum of Step 1 and Step 2 with vary wind direction (Side)

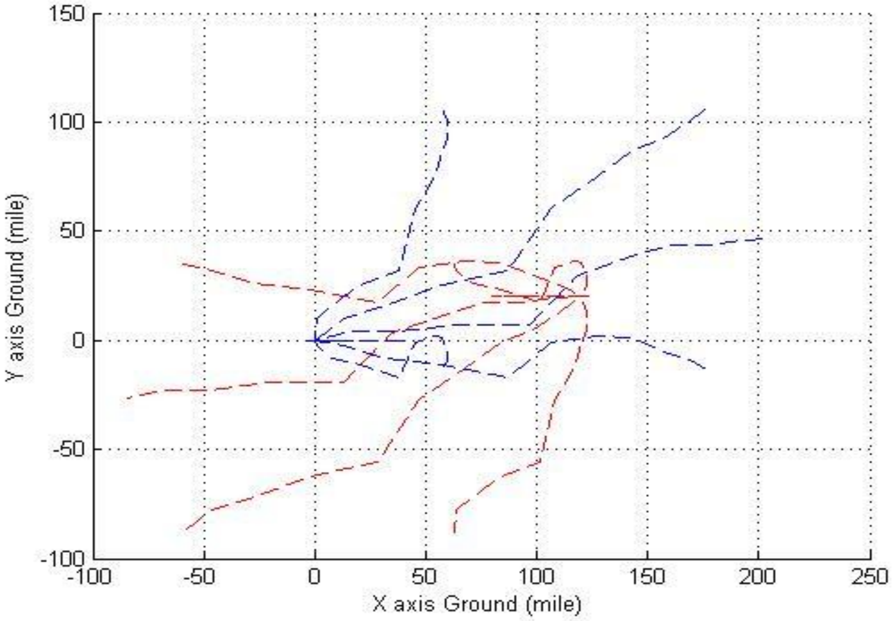


Figure 60 Sum of Step 1 and Step 2 with vary wind direction (Top)

Step 4.Feasible gliding range

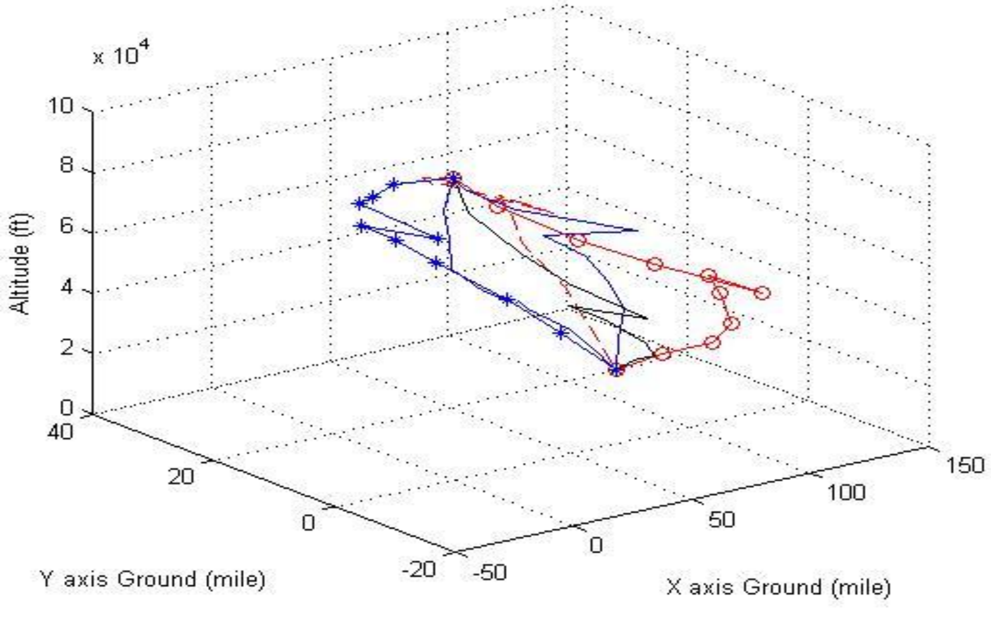


Figure 61 Feasible range of flight with vary wind direction

Step 5.Feasible gliding range ateach step of altitude

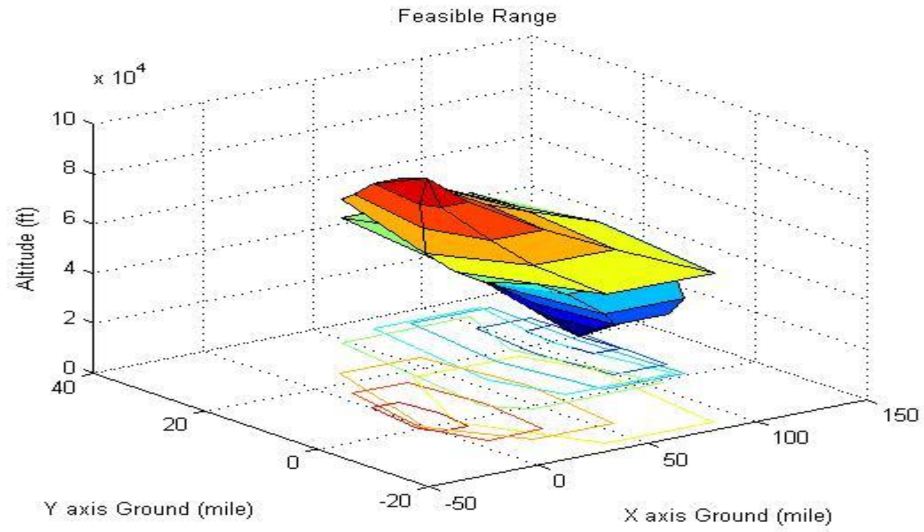


Figure 62 Feasible range of flight with vary wind direction

Step 6.Optimized feasible gliding range

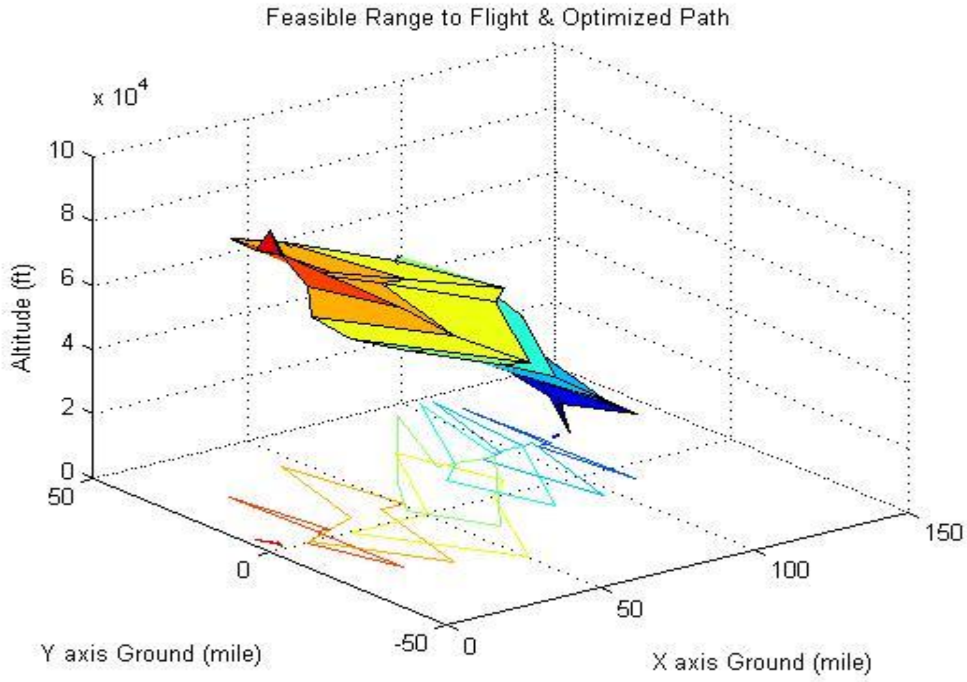


Figure 63 Feasible range of flight with robustness and vary wind direction

3.4.3 3D withvariable wind direction and starting point error

Next, starting point errorswere added to previous calculations. As noted above, the errors were imposed on Step 1 in a 2D case. The difference from a 2D case was that X and Y axis should be considered for the error calculation. Therefore, ±10 miles from the predicted starting point on both X and Y axes were set as an error range. As a result of the calculation, there were several different feasible landing areas from Step 1, and the desired landing point needed to be determined within the overlapping area of *all* landing areas. Figures 63 through 70, below, show the results.

Assumptions

- Parafoil velocity: 15mph
- Starting point: (0, 0, 10000)
- Desired landing point: (120, 20, 0)
- Starting point error: +/-10miles
- Wind speed: [8,15,20,25,40,60,55,45,30,18]mph (low to high alt.)(1000ft steps)
- Winddirection: [280,290,270,300,300,330,270,280,270,300](degrees)(low to high alt.)
- Decent speed of each step:4.4ft/s= 3 mph
- Step= 1000ft
- Flight time of each step= Step/Decent speed
- Pitch angle is fixed
- No vertical wind (on Y axis)

Step 1.Gliding range from starting point to ground

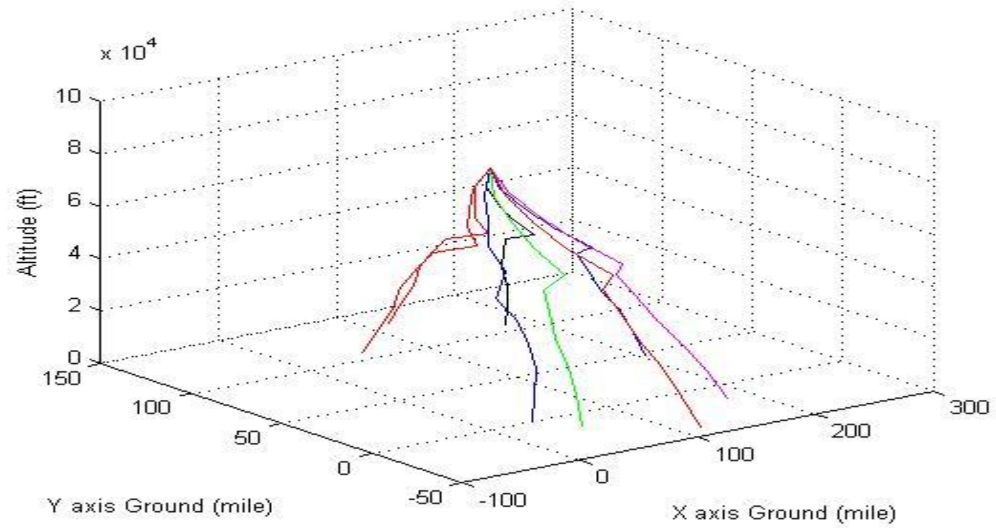


Figure 64 Feasible gliding range from air to ground with vary wind direction

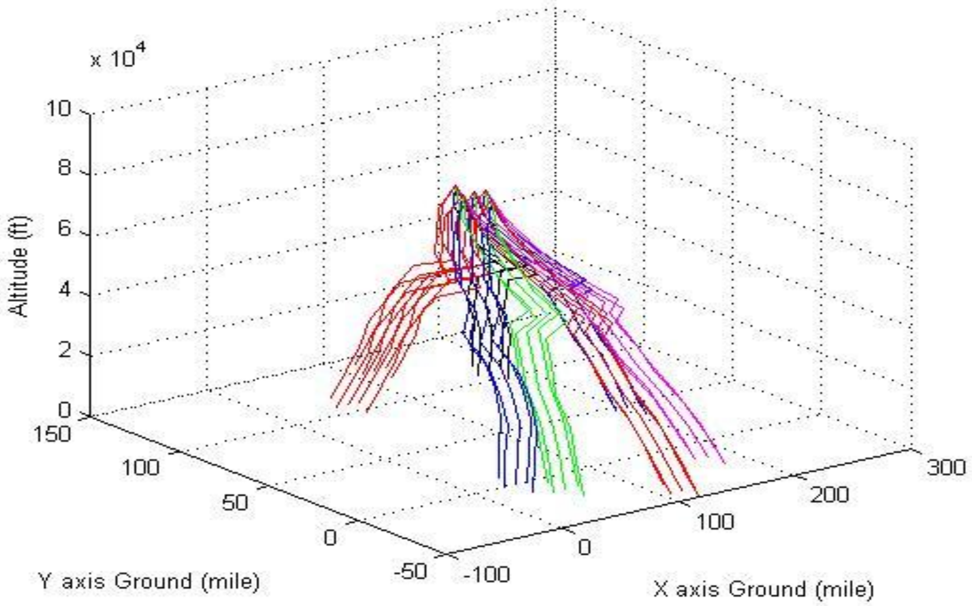


Figure 65 Feasible gliding range from air to ground with vary wind direction and starting point error

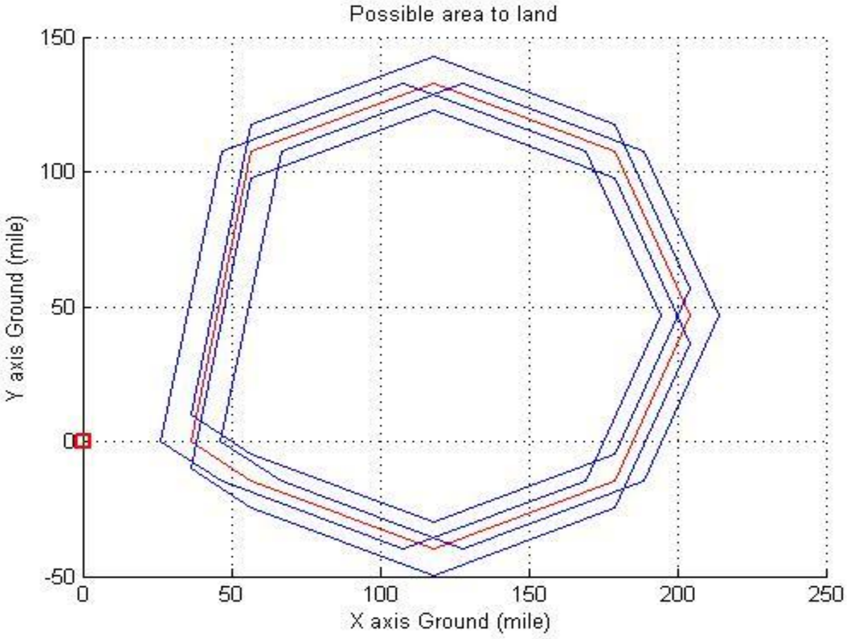


Figure 66 Feasible landing point with vary wind direction and starting point error

Step 2.Gliding range from desired landing point to sky

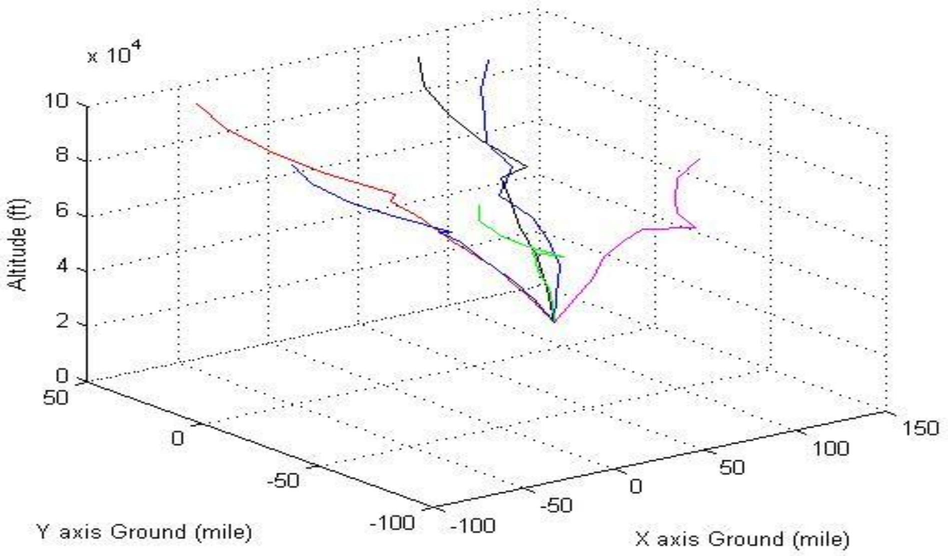


Figure 67 Feasible gliding range from ground to air with vary wind direction and starting point error

Step 3.Sum of Steps 1 and 2

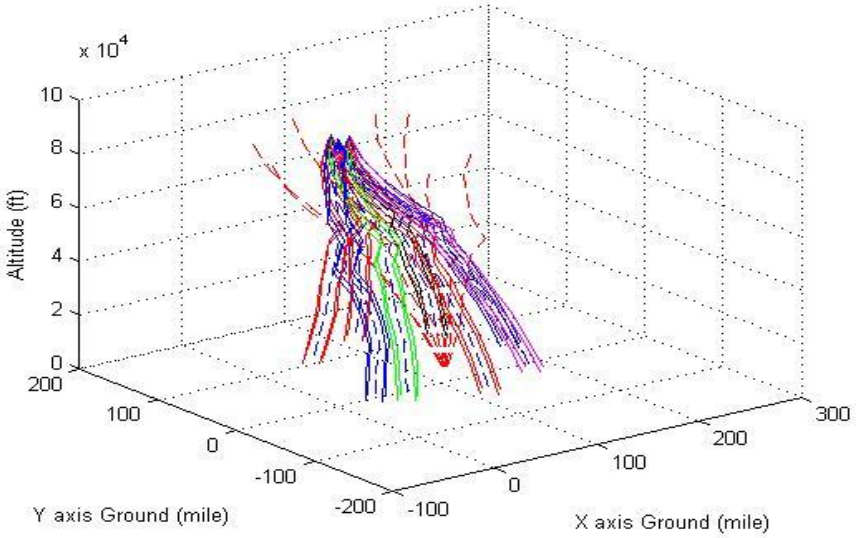


Figure 68 Sum of Step 1 and Step 2 with vary wind direction and starting point error

Step 4.Feasible gliding range

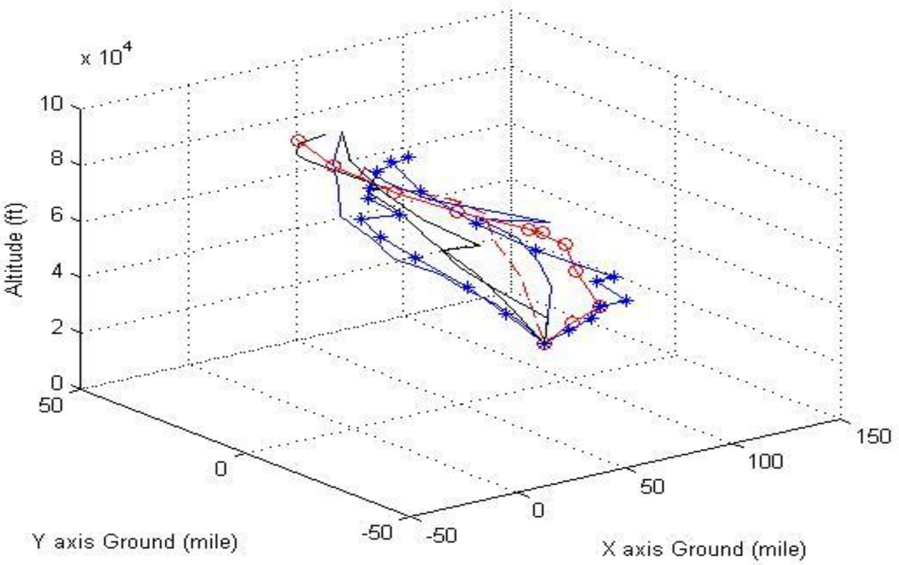


Figure 69 Feasible range of flight with vary wind direction and starting point error

Step 5.Feasible gliding range ateach step

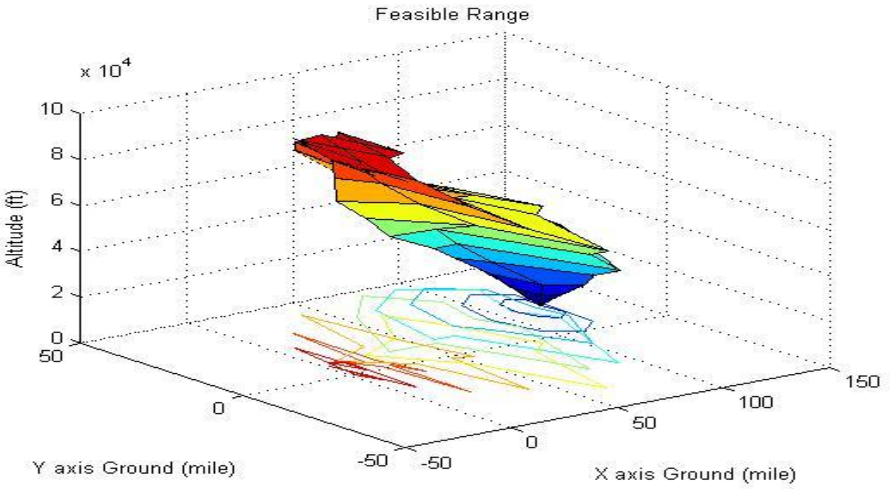


Figure 70 Feasible range of flight with vary wind direction and starting point error

Step 6.Optimized feasible gliding range

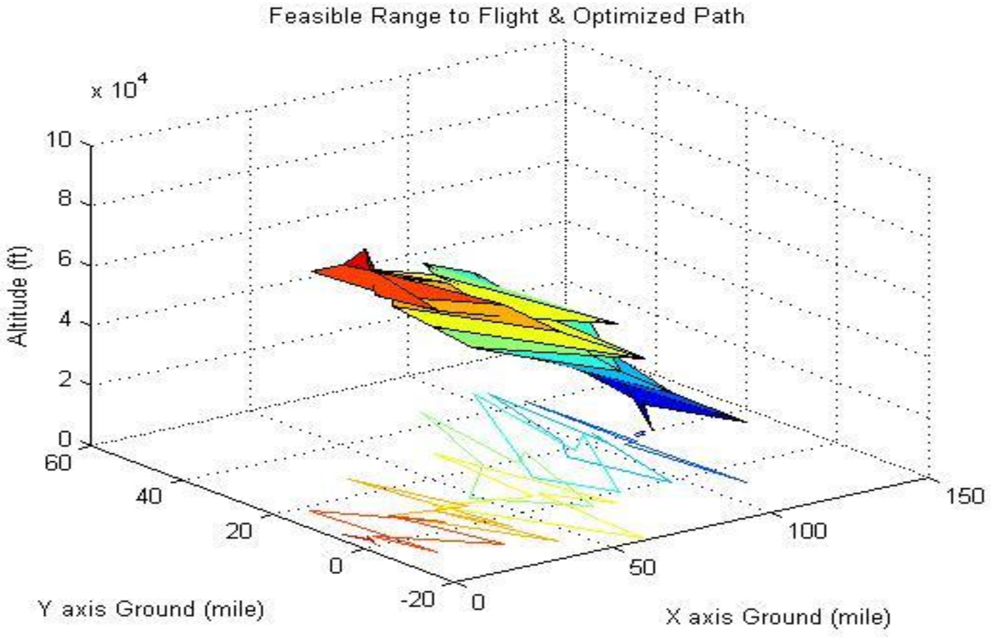


Figure 71 Feasible range of flight with robustness, vary wind direction and starting point error

3.4.4 3D Variable wind direction, starting point errors and wind direction errors

As mentioned above, the chief differences between 2D and 3D cases were the wind direction. This wind direction was obtained fromAPL data before the launch; in other words, the data is possiblydifferent in real flight condition with real weather condition. Thus,wind direction errors needed to be considered as well. A ±10 degree of wind direction error term was added tosame procedure used inprevious calculations. This joined wind speed and starting point errors. Figures through 78, below, show the results.

Assumptions

- Parafoil velocity: 15mph
- Starting point: (0, 0, 10000)
- Desired landing point: (120, 20, 0)
- Starting point error:±10miles
- Wind direction error = ±10 degrees
- Wind speed: [8,15,20,25,40,60,55,45,30,18]mph (low to high alt.)(1000ft steps)
- Winddirection: [280,290,270,300,300,330,270,280,270,300]degrees(low to high alt.)
- Descent speed ateach step:4.4ft/s= 3 mph
- Step = 1000ft
- Flight time of each step= Step/descentspeed
- Pitch angle is fixed

Step 1.Gliding range from starting point to ground

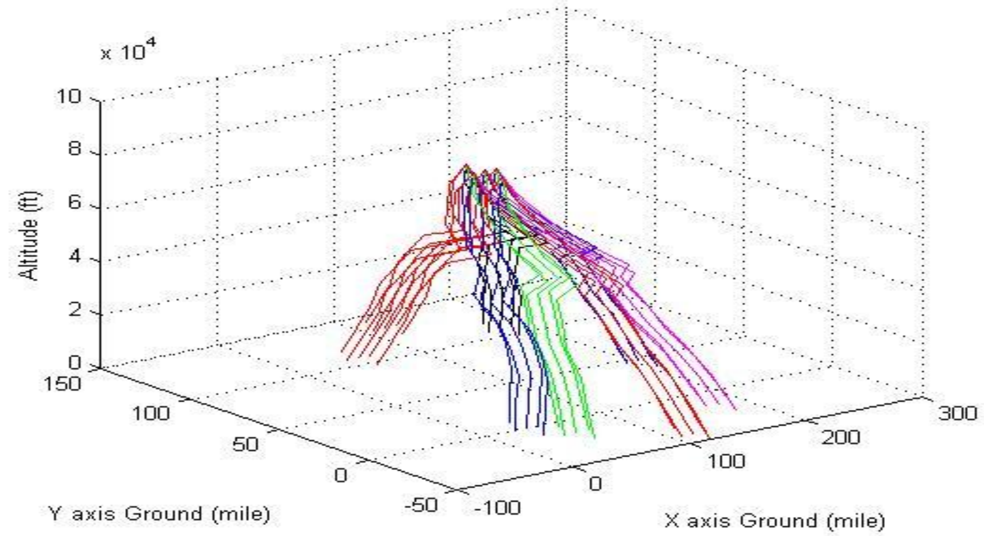


Figure 72 Feasible gliding range from air to ground with vary wind direction and starting point error

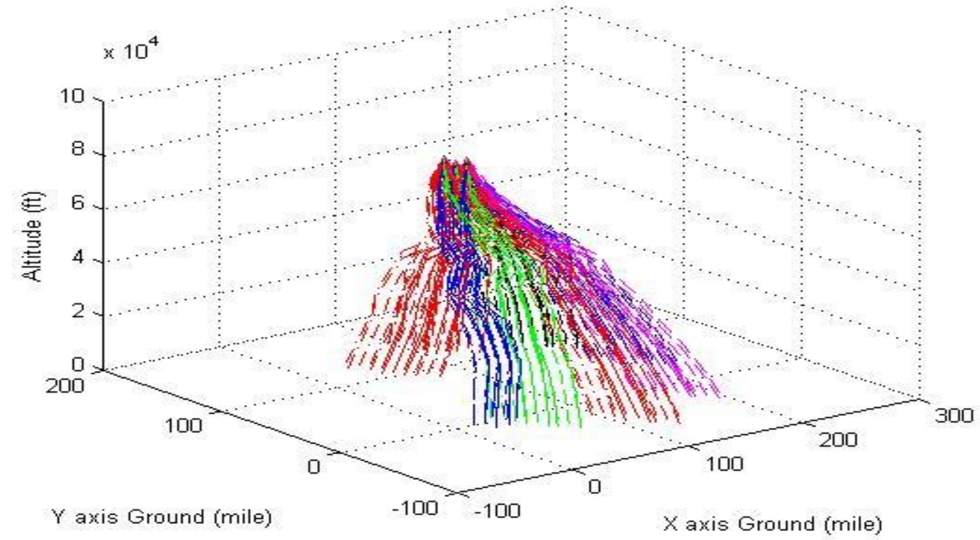


Figure 73 Feasible gliding range from air to ground with vary wind direction, starting point error, wind speed error and wind direction error

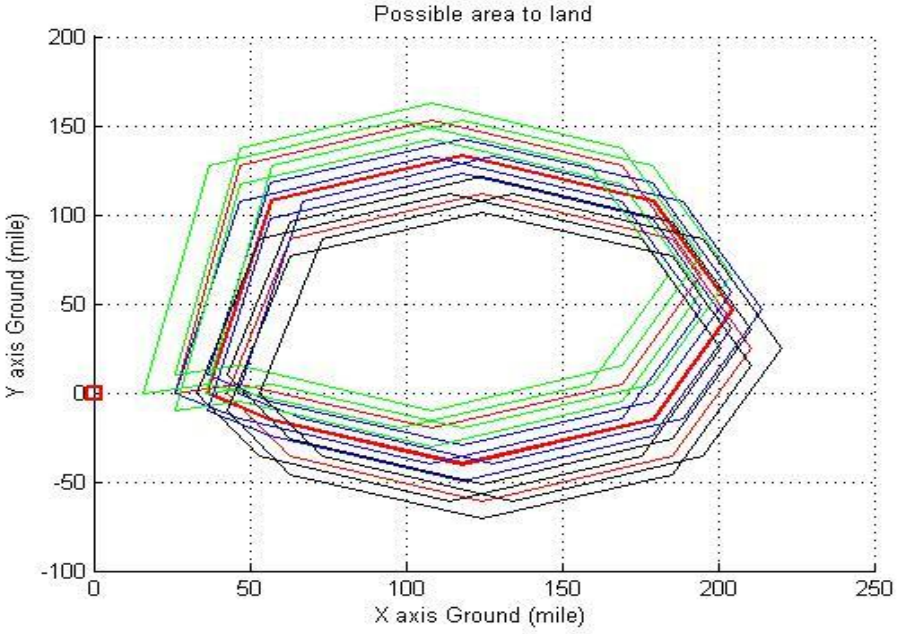


Figure 74 Feasible landing area with vary wind direction, starting point error, wind speed error and wind direction error

Step 2.Gliding range from desired landing point to sky

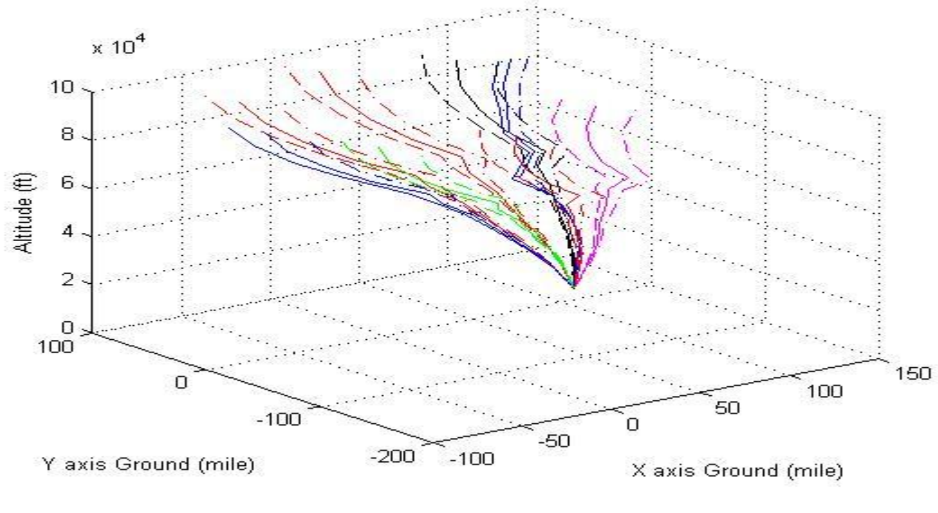


Figure 75 Feasible gliding range from ground to air with vary wind direction, starting point error, wind speed error and wind direction error

Step 3.Sum of Steps 1 and 2

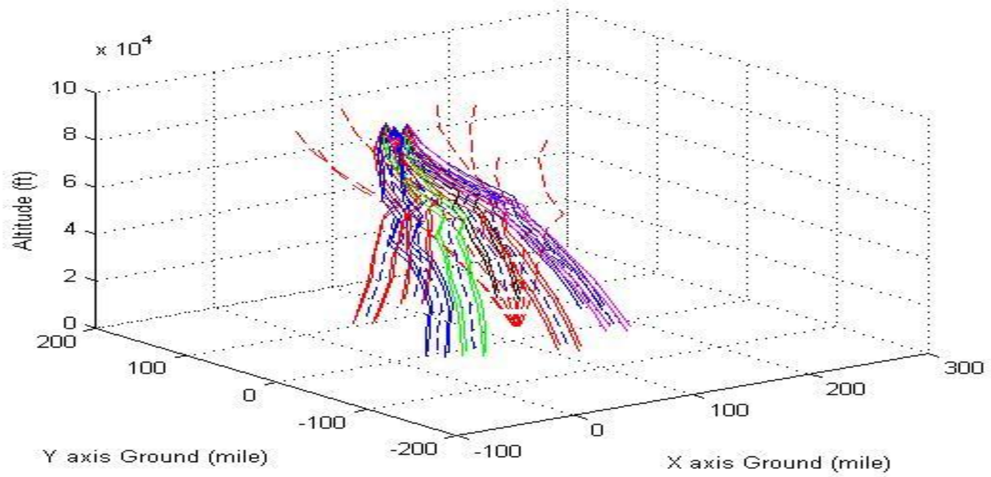


Figure 76 Sum of Step 1 and Step 2 with vary wind direction, starting point error, wind speed error and wind direction error

Step 4.Feasible gliding range

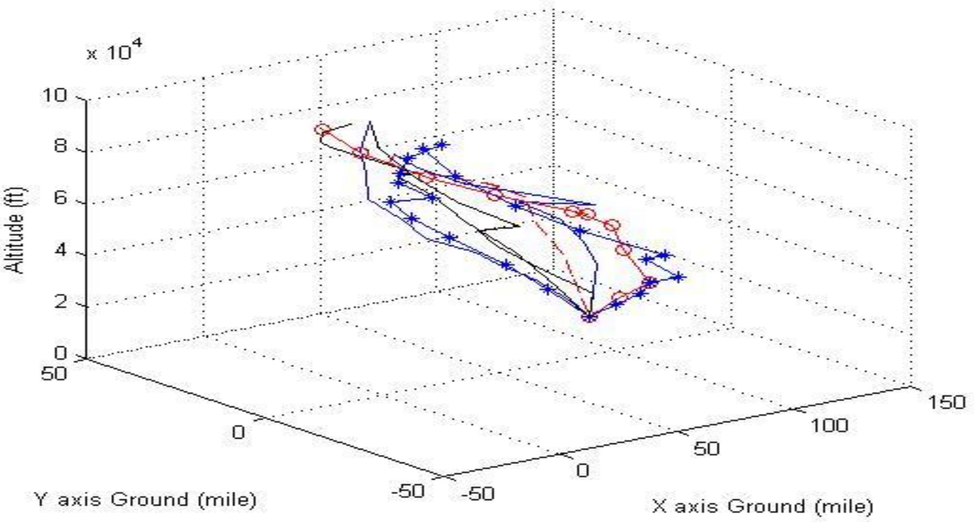


Figure 77 Feasible range of flight with vary wind direction, starting point error, wind speed error and wind direction error

Step 5.Feasible gliding range ateach step

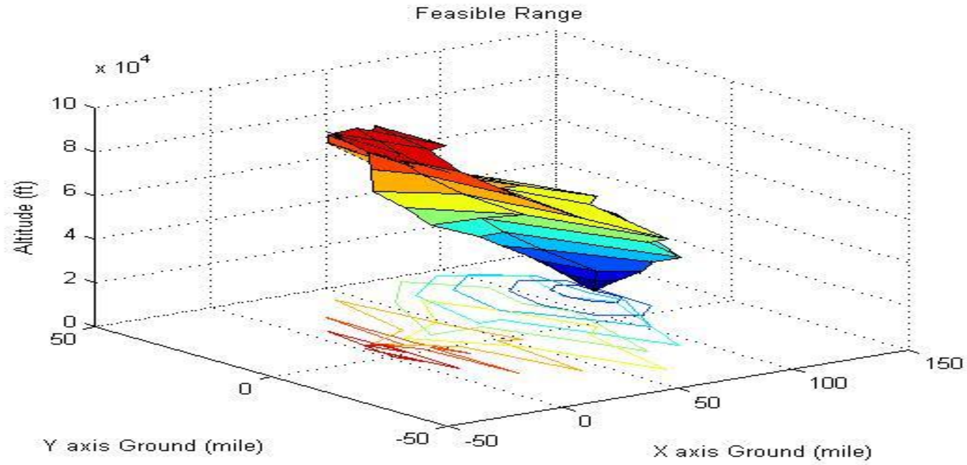


Figure 78 Feasible range of flight with vary wind direction, starting point error, wind speed error and wind direction error

Step 6.Optimized feasible gliding range

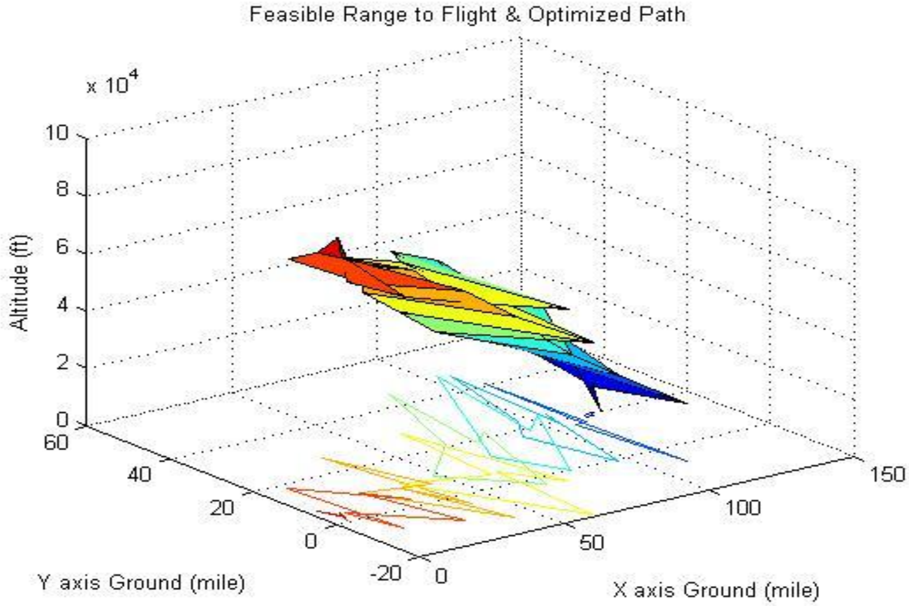


Figure 79 Feasible range of flight with robustness, vary wind direction, starting point error, wind speed error and wind direction error

3.4.5 Comparisons between with-errors calculation and without-errors calculation

A comparison of calculations with and without errors was conducted to verify the effects of the errors. For the comparison, the feasible gliding area of each step was drawn in 2D. Obviously, the feasible gliding area with errors was narrower than that without errors, and one interesting item was that the upper half of the with-errors altitude figures was narrow due to the starting point error compensation. The error in fact generated one added intersection inside of the feasible range.Andfrom the intersection, the feasible gliding range grew wider, so there were small feasible areasin the high-altitude region. Figures 79 through 86, below, show the results of the comparisons.

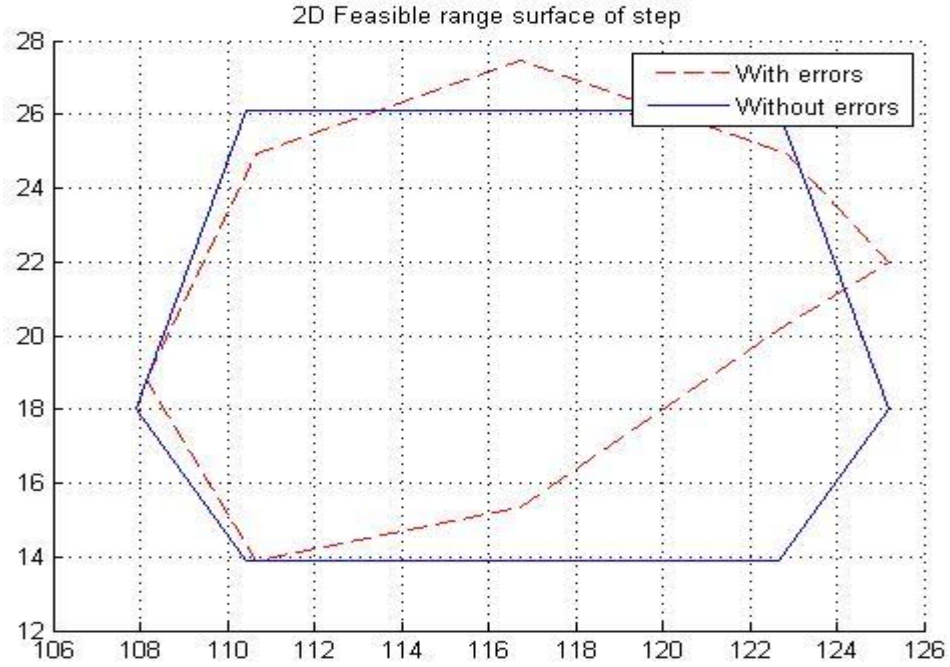


Figure 80 Feasible area of flight (10,000ft)

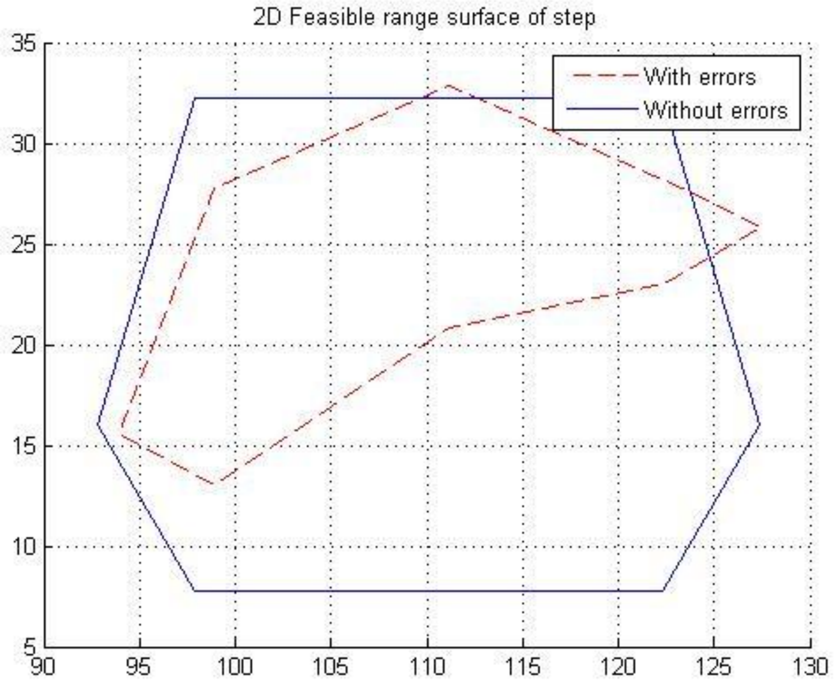


Figure 81 Feasible area of flight (20,000ft)

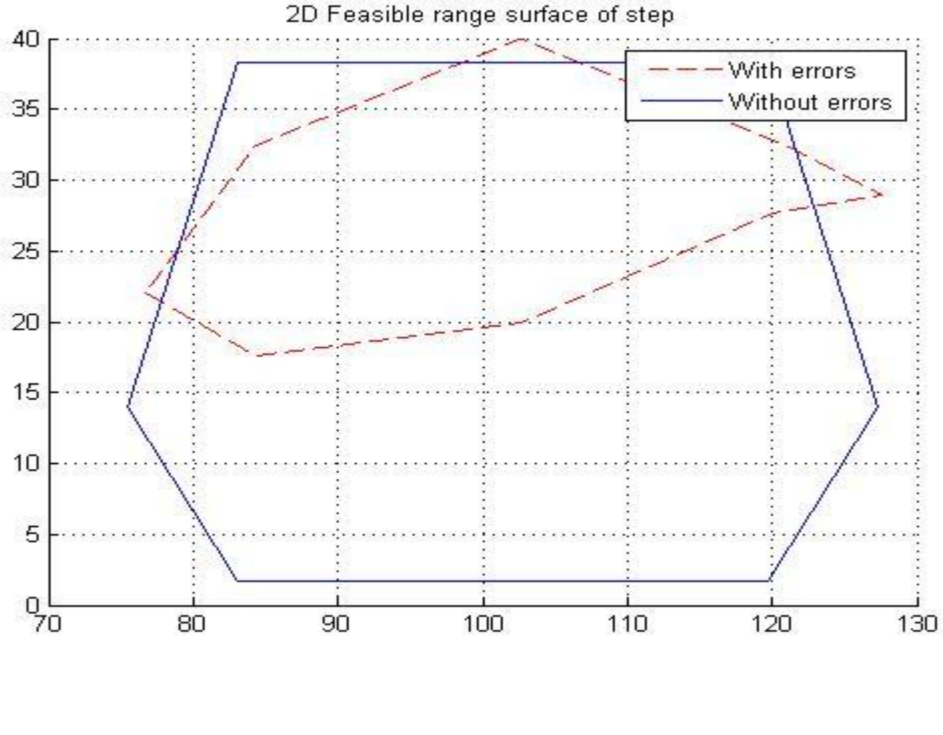


Figure 82 Feasible area of flight (30,000ft)

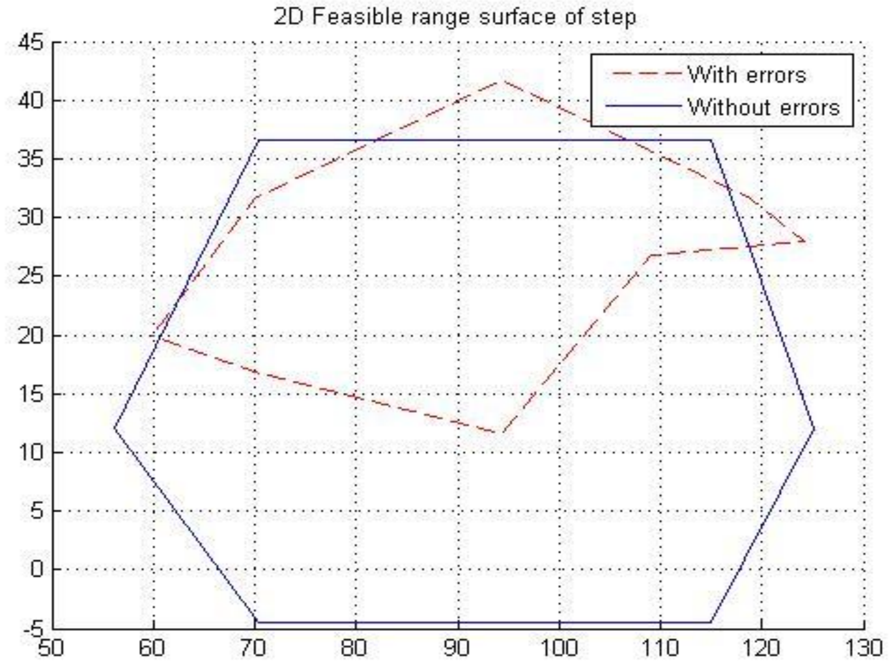


Figure 83 Feasible area of flight (40,000ft)

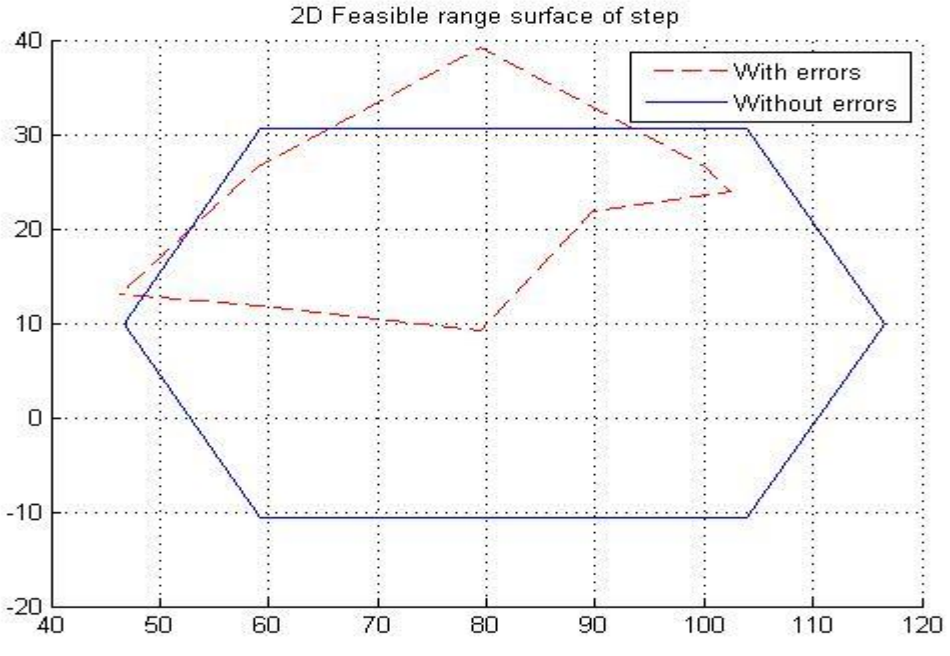


Figure 84 Feasible area of flight (50,000ft)

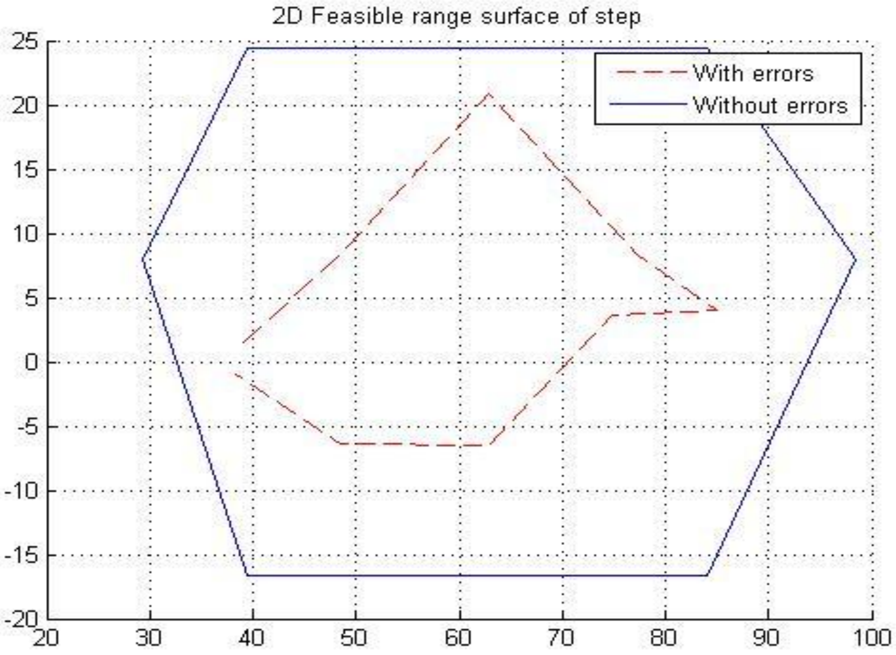


Figure 85 Feasible area of flight (60,000ft)

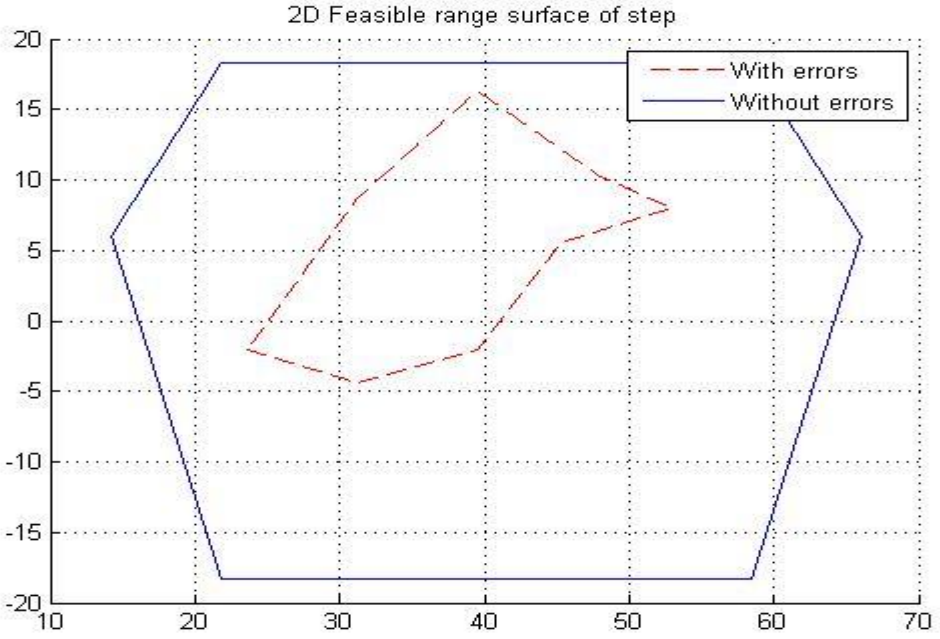


Figure 86Feasible area of flight (70,000ft)

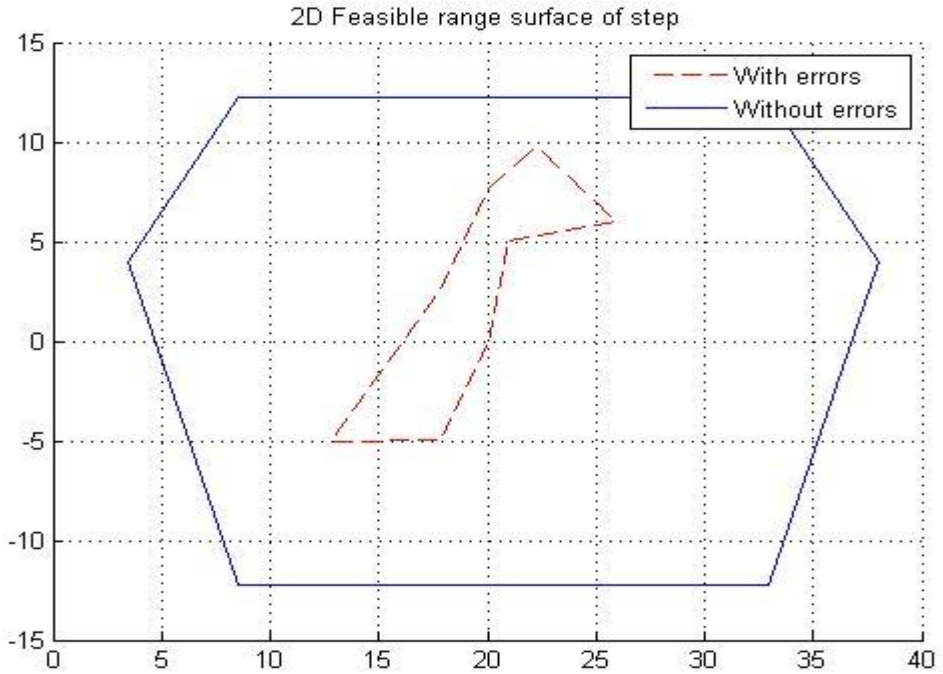


Figure 87 Feasible area of flight (80,000ft)

3.4.6 Simulation fromreal flight conditions

To verify the effectivenessof this method, a simulation was performed with ASTRO 15's flight data. All error compensations were applied and all other assumptions were the same as in the above cases except the descent speed of the parafoil.

Assumptions

- Parafoilvelocity: 15mph
- Starting point: (0, 0, 10000)
- Desired landing point: (120, 20, 0)
- Wind speed: [10.73, 34, 55.47, 46.97, 56.14, 46.3, 33.5, 15.6, 13.8, 11.1]mph (low to high alt.)(1000ft steps)
- Winddirection:[294.6, 294.4, 276.8, 289.5, 271.9, 264.4, 262.8, 206.3, 187.6, 53.4]degrees(low to high alt.)
- Descent speed ateach step: 9ft/s
- Step= 1000ft
- Flight time of each step= Step/descentspeed
- Pitch angle is fixed
- No vertical wind (on Y axis)

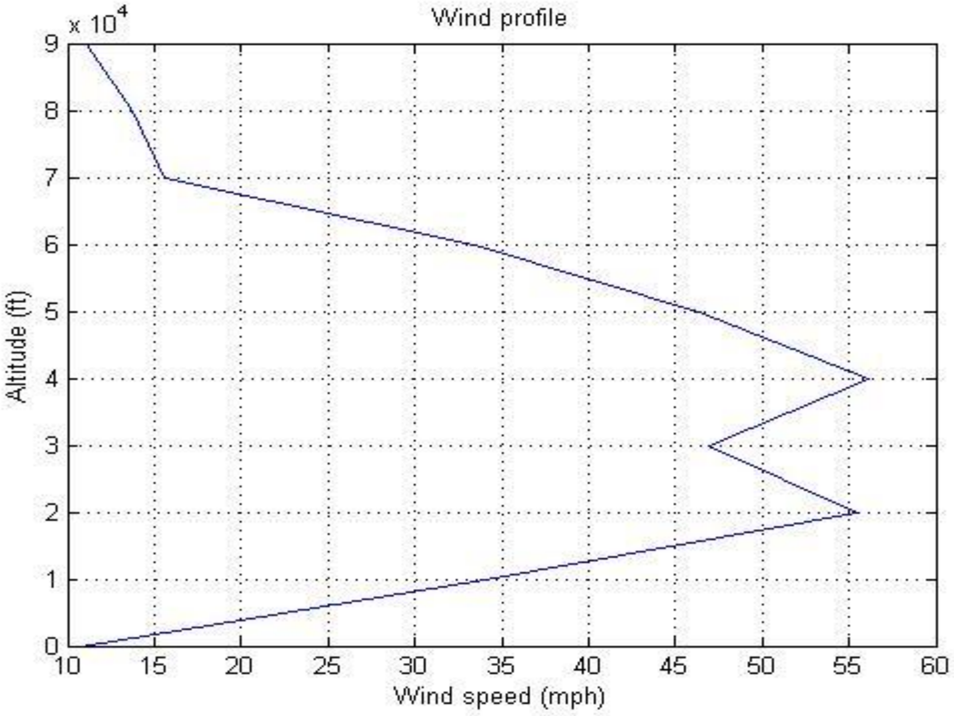


Figure 88 ASTRO 15 wind profile

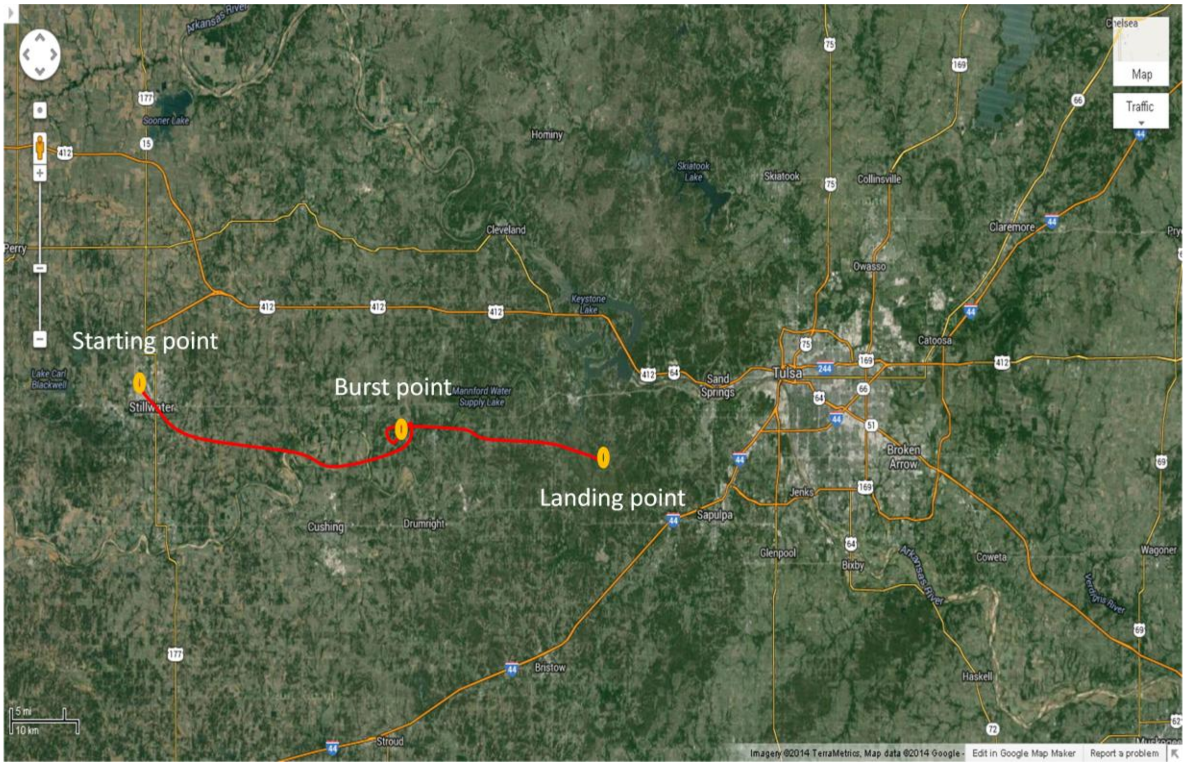


Figure 89 ASTRO 15 parachute flight path

Step 1.Gliding range from starting point to ground

The feasible gliding range was calculated from burst point to the ground. On ASTRO 15, the burst point was at a height of about 100,000ft.and about 10 miles to the north of Drumright, so the starting point was set at that point and the possible landing area was calculated. The parafoil has lift and gliding ability andtherefore glided further than a parachute. It had a wide feasible range of about 60 miles west to east and 40 miles north to south, which coversthe Tulsa area to 40 miles east of Tulsa.

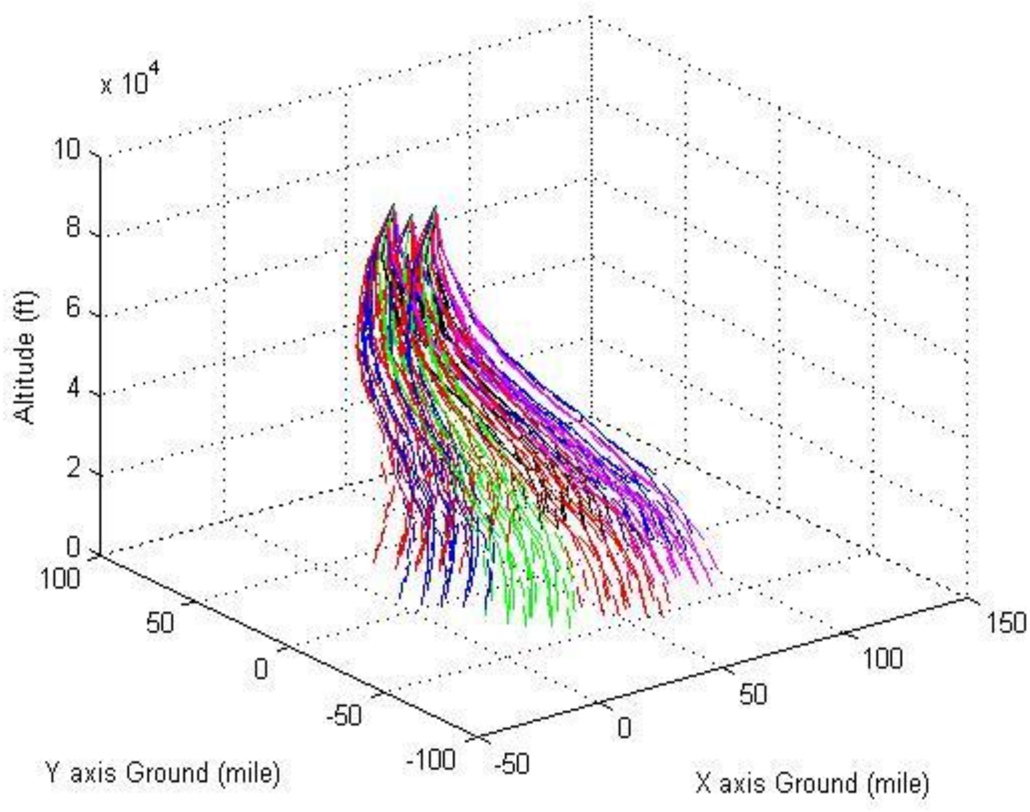


Figure 90 Feasible range, air to ground

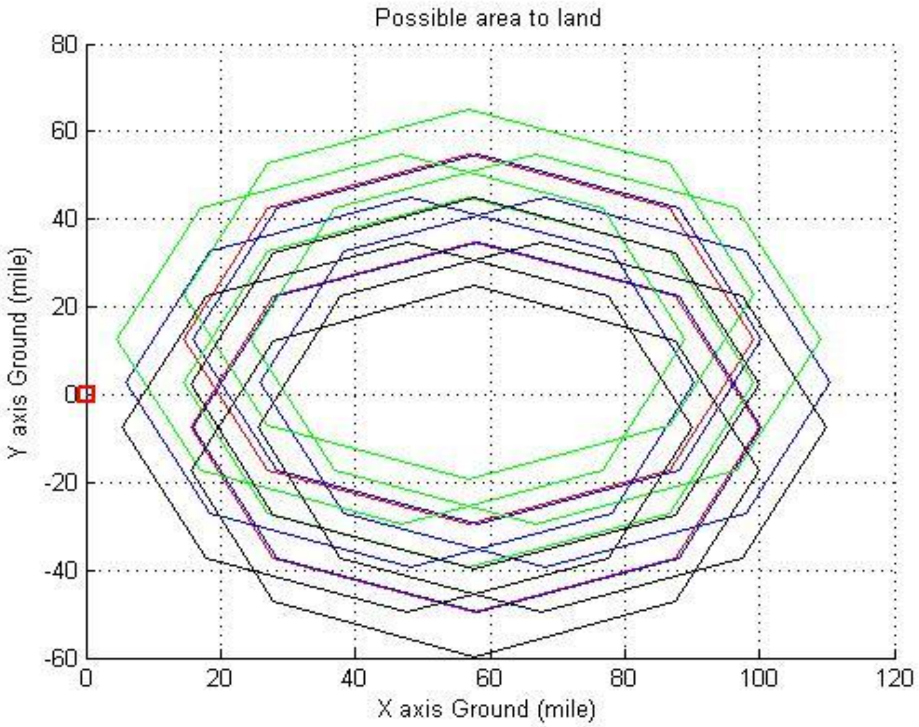


Figure 91Feasible landing area

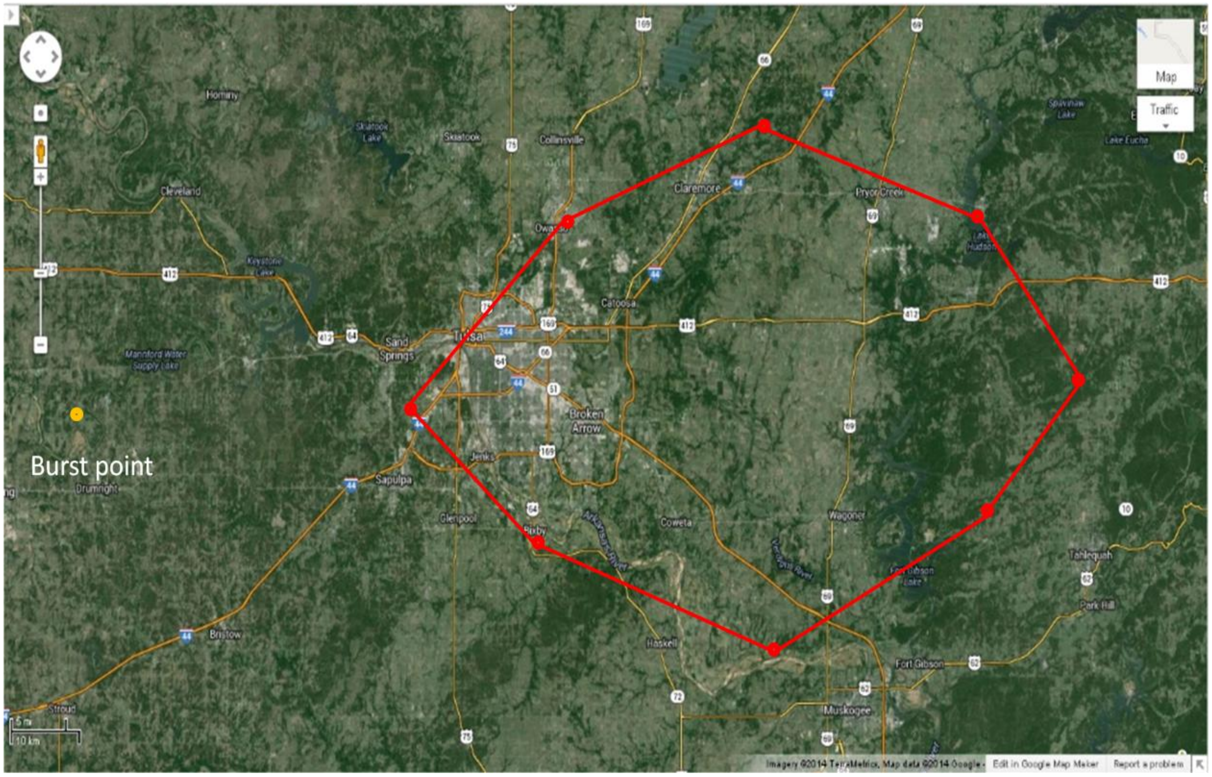


Figure 92 Feasible landing area, mapped

Step 2.Gliding range from desired landing point to sky

After checking the feasible landing area, the desired landing point was identified within the range. The west side of the area was an urban site, so the desired landing point was placed near Coweta, which is a small town located southeast of Tulsa. The objective of this project is a target accuracy of within a1 mile diameter circle, so the area circled in Figure 92, below, was selected as the target landing point; it also has easy road access and no trees. After setting the desired landing point, the Step 2 calculation was performed to check the feasible gliding range from ground to air.

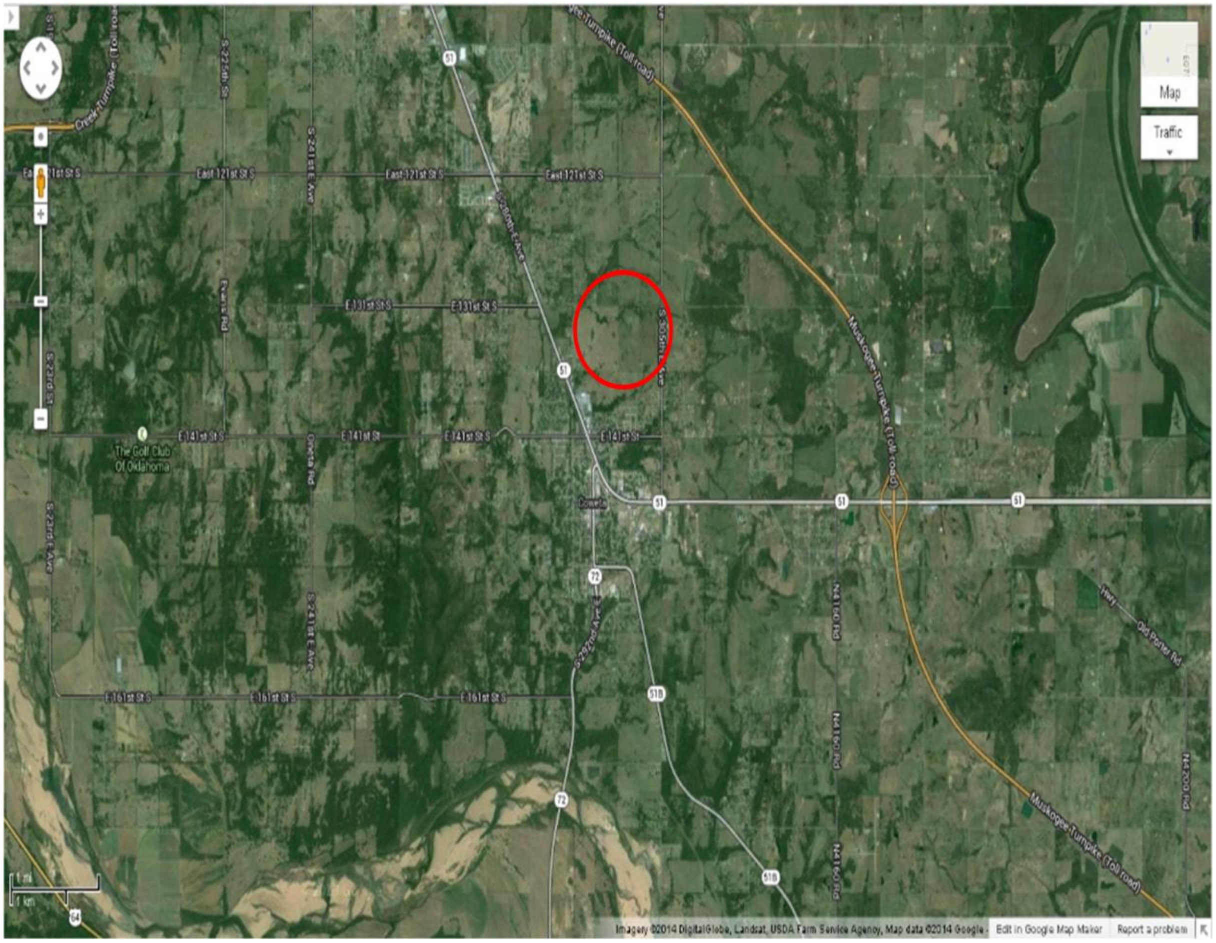


Figure 93 Desired landing point, mapped

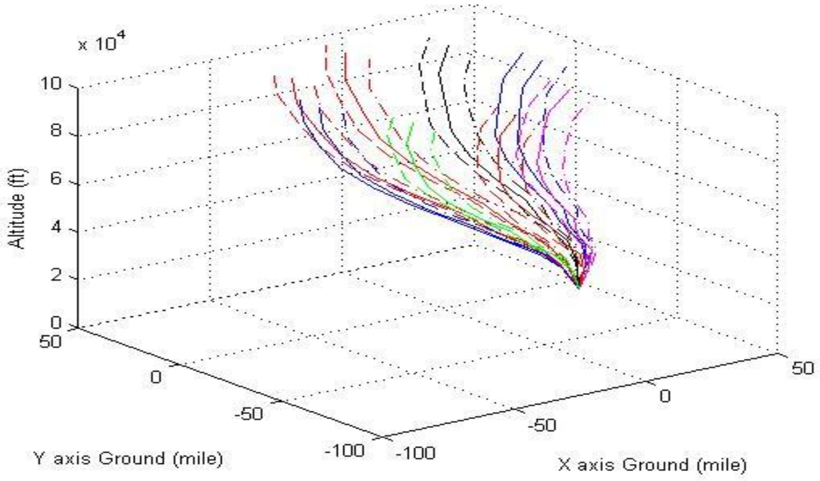


Figure 94 Feasible range, ground to air

Step 5.Feasible gliding range ateach step

After finishing the calculation for Step 2, Steps 3, 4 and 5 were performed to check the overlapping areas of Steps 1 and 2. As a result, the feasible range was obtained and plotted on the map. The wind direction was west to east at most altitudes, so the feasible range displayed similar sizes and shapes at each step. Each waypointwas thus able to be set within the feasible range.

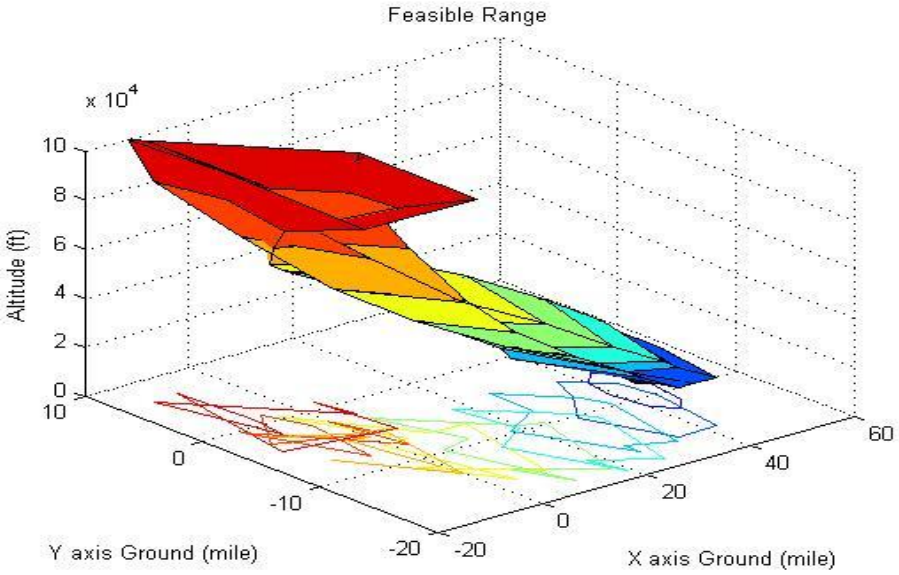


Figure 95Feasible flight range

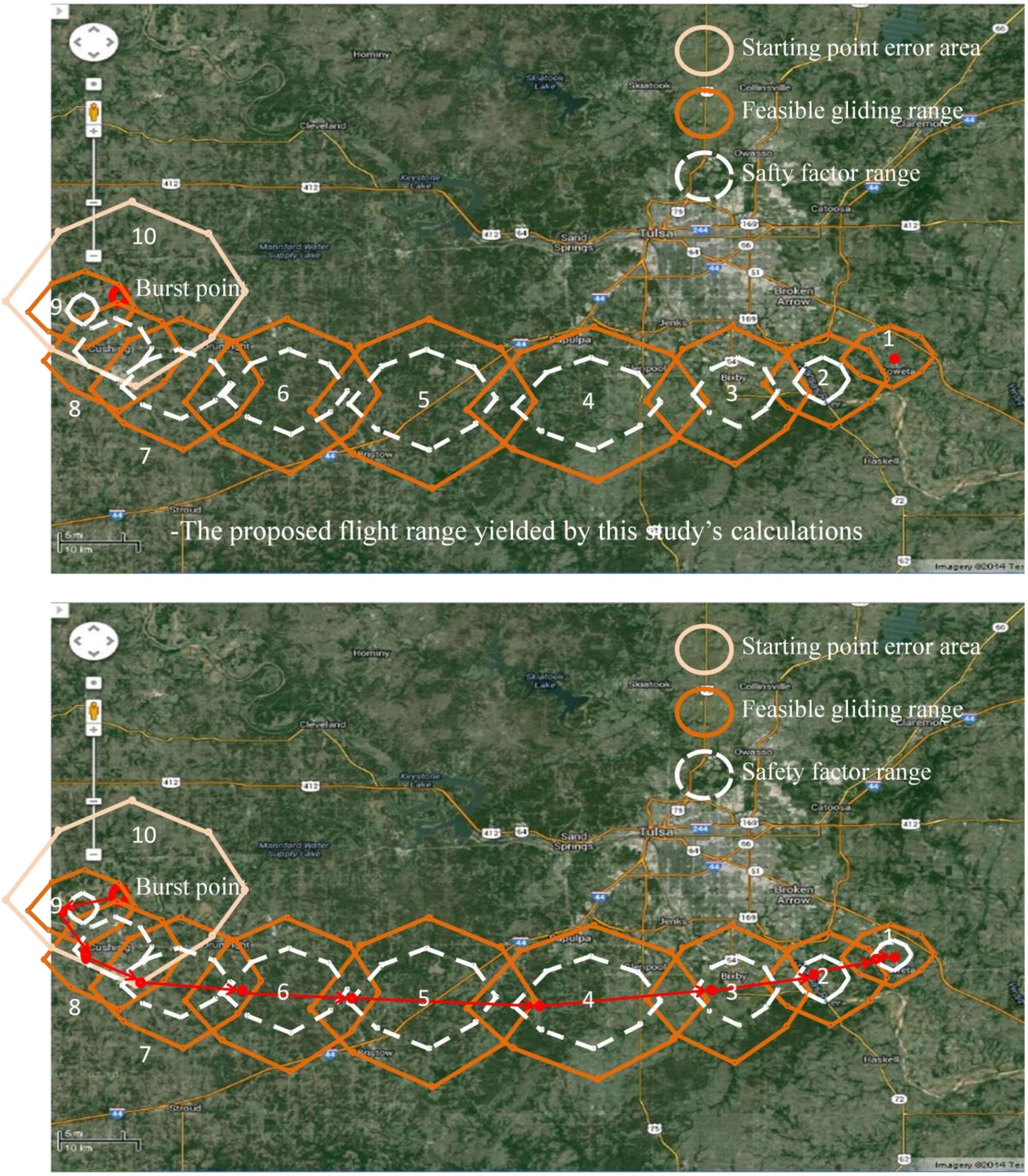


Figure 96Feasible flight range, mapped

CHAPTER IV

GUIDANCE, NAVIGATION AND CONTROL

The guidance, navigation and control systems are designed to force the parafoil to follow the optimized path as accurately as possible. The relationship between these systems is shown in Figure 96.

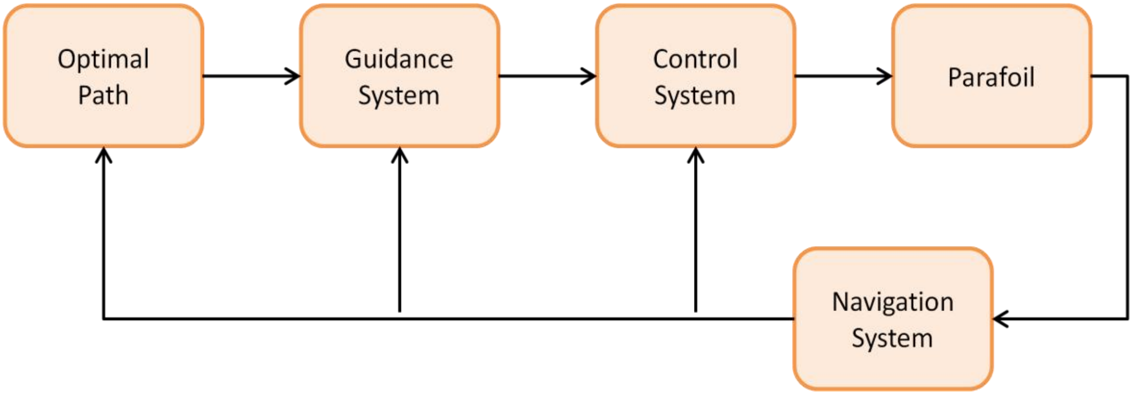


Figure 97Guidance, Navigation and Control System

The navigation system measures and estimates all the relevant states of the vehicle from the output, and the optimized path the vehicle should follow is determined from Dynamic programming. The guidance system uses the path data to generate a reference signal and the control system is also used to track that reference signal by applying a control input to the system.

4.1 Guidance System

The guidance system is responsible for generating the control input based on a desired trajectory which is either pre-programmed or generated in Autopilot. The important part of the system is the fundamental rules of the parafoil. The first rule is that the RPV is always falling down to the ground. The second rule is that the vehicle has a limitation to the rolling angle to maneuvering.

The optimized path will be suggested by a single trajectory. The considered autopilot system is a single-axis, which controls the parafoilon the roll axis only; such autopilots are also colloquially known as “wing levelers”, a name which reflects its limitations. Once the single-axis system is developed and proves to work well, the autopilot system can be extended to two-axis, which controls a vehicle in the pitch, as well as roll axis. This will provide slightly more than a “wing leveler” with a limited pitch-oscillation-correcting ability.

The autopilot in RPV reads its position and altitude from an on-board GPS and calculates the difference between the current and target points. In the programmed control system, the error is corrected. Figure 97 shows the scheme of the autopilot system:

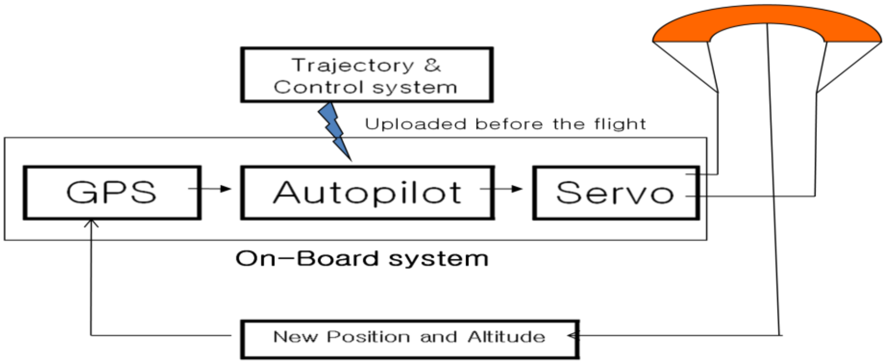


Figure 98Autopilot system flow diagram

4.2Navigation System

The approach to the waypoint navigation is to calculate a bearing, which points directly towards the target waypoint from the vehicles' current position.

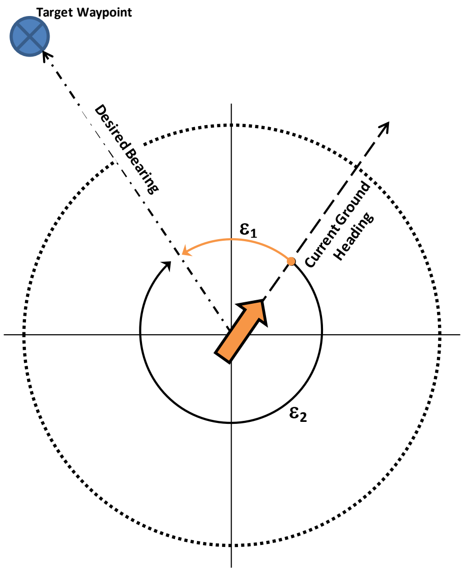


Figure 99Path Planning Scheme

Since it is safe to assume that the vehicle will have access to its current location and the location of the desired waypoint, the autopilot can compute the desired bearing to the target by using the FAI bearing calculation as follows:

$$\varphi_{bearing} = atan\left[\frac{cos(lat_0) * sin(lat_1\ )| \ sin(lat_0) * cos(lat_1) * cos(lat_1 - lat_0)}{sin(lon_1 - lon_0) * cos(lat_1)}\right] \tag{4.1}$$

Once the bearing is known, the difference between the current ground heading and desired bearing is found by:

$$\epsilon_{\varphi} = \varphi_{bearing} - \varphi_{heading_{GND}} \tag{4.2}$$

The heading error indicates the number of degrees the system would need to turn, in order to correct its current heading to that of the desired bearing. Unfortunately, this solution does not guarantee that the vehicle is heading by turning in the direction of the minimum angle. To determine the minimum angle for the vehicle to turn to avoid this, the error is corrected by either subtracting 360 degrees from the heading error, if it is greater than 180 degrees, or adding 360 degrees to the heading error, if it is less than -180 degrees. A negative heading error will produce a left-turn command to correct the vehicle, while a positive heading error produces a right turn.

One problem with this method is that as the distance to the target point becomes smaller, considerable changes in heading error might occur for a small offset in position, with respect to the target. A simple and robust correction involves defining a circle of a specified radius around the waypoint. As soon as the vehicle enters this circle, its behavior can be modified to switch from a waypoint seeking mode to a circular orbit.

Typically, the commanded turn angle and the heading error cannot be the same. There are several reasons for this. First, the system may not be able to be physically commanded to turn such a tight angle, and second, even if the system can command a large turn rate, the system will start to overshoot. In order to determine the steering angle of the vehicle, a simple proportional gain controller is used. This controller takes on the following form:

$$\textbf{Steering Angle} = \epsilon_{\varphi} * K_p \tag{4.3}$$

However, the steering command may still need to be limited to:

$$\textbf{Steering Angle} = \left[\begin{array}{c} \geq 10^{\circ} \text{ then Steering Angle} = 10 \\ \leq 10^{\circ} \text{ then Steering Angle} = -10 \\ \text{else Steering Angle} \end{array} \right. \tag{4.4}$$

Once a steering angle was found, a servo command was generated by the code to drive the overboard servo hardware found on the ArduPilot. The following was the basic pseudo code used to create the autopilot:

```
1. Initialize the system:
    a. During this stage the onboard microcontroller checks the status of the
       GPS and waits until a lock is obtained.
    b. Determine 'Return to Home' location
    c. Load waypoint(s) into the active memory
2. Wait until the Autopilot start is given;
3. Decode GPS message to determine the current states;
4. Determine the turn angle:
    a. Determine the bearing to the current waypoint
    b. Determine the current heading of the system
    c. Determine the error between the bearing and heading
    d. Calculate the servo turn angle
5. Send servo angle command to the controller hardware;
6. Wait for the updates of the GPS status;
7. Repeat Steps 2 through 6.
```

4.3Control System

The guidance of our vehicle to designated path was a challenging issue, especially from a high altitude. An accurate, simpleand flexible control system was required to navigate throughout the descent phase.

To satisfy the requirements, the proposed control system is a proportional-integral-derivative controller (the PID controller) which has been used in the previous flight tests in Appendix (###). This is a generic control loop feedback mechanism widely used in control systems – a PID is the most commonly used feedback controller. Basically, a PID controller calculates an "error" value as the difference between a measured process variable and a desired setpoint. The controller attempts to minimize the error by adjusting the process control inputs. Linear-Quadratic Regulator (LQR) and Model Predictive Control (MPC) control algorithm were also considered, but Ardupilot (Main Calculation board) was not able to calculate those complicated calculations and slowed down the vehicle’s whole system. In the knowledge of the underlying processes, the PID controllers are the best for the purposes of this study.

The PID Controller involves three separate parameters: The proportional, the integral, and the derivative values, denoted as P, I, and D. The derivative is not used in our system with the same reason as the LQR and MPC controller.

In the previous ground test, the integral value was very small because (deletion) the parafoil glided so slowly and the GPS had a little time delay to send the position signal.

Therefore, the higher accumulation value of the past errors, the integral value, caused the compensation of the response to the small errors too quickly.

The present errors, the proportional value, are mainly effective values to provide control action for the autopilot’s requirements:

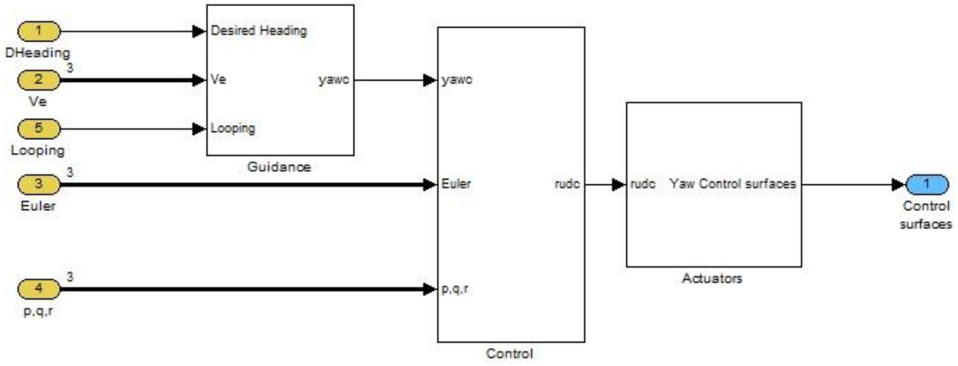


Figure 100Guidance, Control and Actuators module

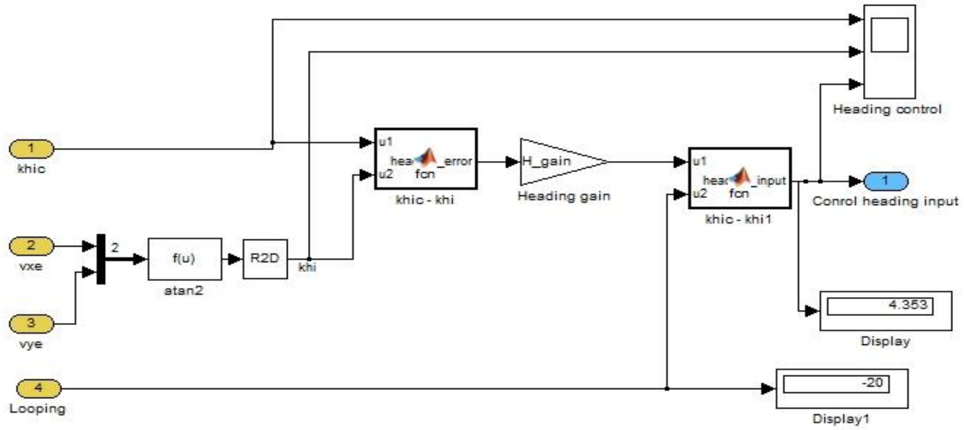


Figure 101Heading Control block

This PI algorithm for controlling the vehicle does not guarantee the optimal control. However, this has been the most suitable controller for this project so far. Possibly, other optimal controllers might also be applied and tested in near future.

CHAPTER V

AVIONICS, HARDWARE AND SOFTWARE

The main purpose of the RPV's hardware is to receive position data, calculate the path, control the servos, and record the vehicle's data. To satisfy the ASTRO project's goal, all components are of low-cost, low-weight, and high efficiency parts. One digital servo for the roll control and one winch servo for the full deflection control are planning to be used.

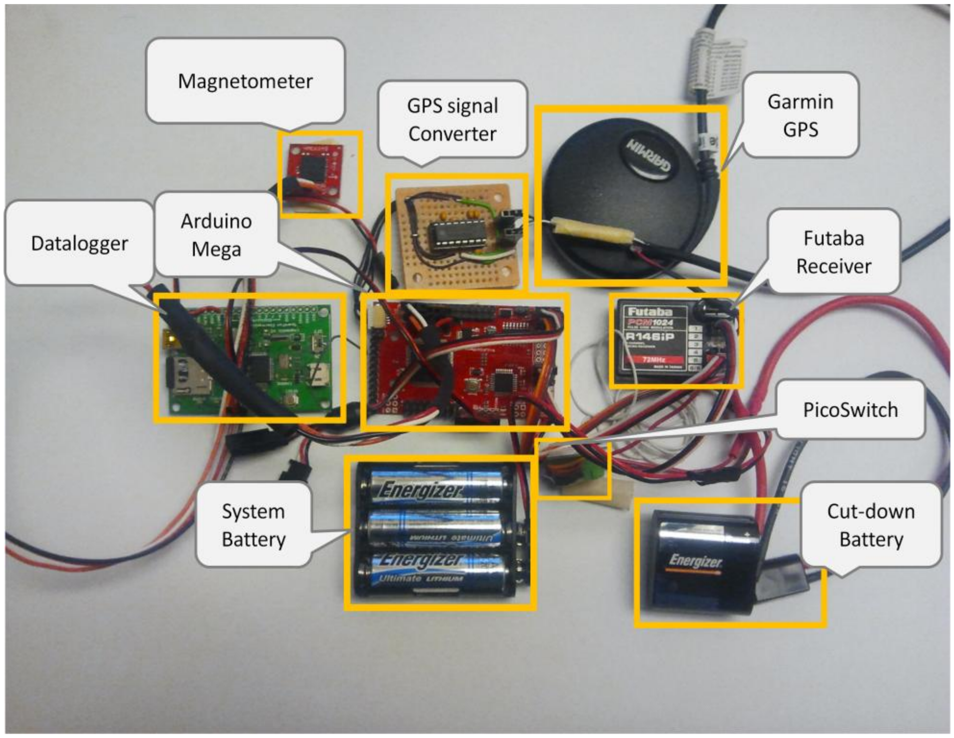


Figure 102Overview of Autopilot hardware

5.1 Autopilot

The Autopilot board was changed from “ArduPilot” to “ArduPilot Mega”. The ArduPilot Mega is an inexpensive, easy to use open source autopilot platform created by Chris Anderson and Jordi Munoz of DIY Drones. The new hardware consists of the core autopilot board (ATMega1280) which provides more memory capability and multi-serial connections for the GPS, sensors and various accessories to add to its functionality. This is a fully programmable autopilot that requires a GPS module and sensors to create a functioning Unmanned Aerial Vehicle (UAV). It also handles stabilization and navigation, and also supports a “Fly-by-Wire” mode that can stabilize an aircraft when flying manually under RC control. It is based on a 16MHz ATMega1280 processor and the total onboard processing power of approximately 32MIPS. The board has 16 spare analog inputs and 40 spare digital input/outputs to add to additional sensors. Furthermore, it provides 8 RC channels that can drive the servo control through the autopilot.

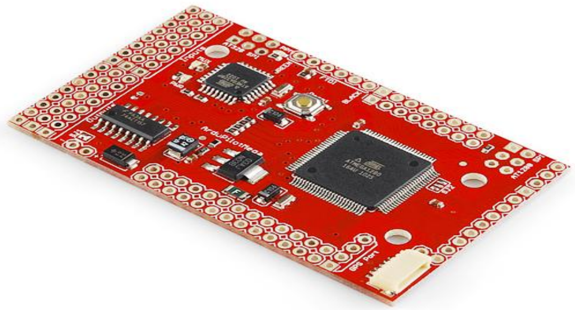


Figure 103Autopilot Board

5.2ATmega1280

The ATmega1280 is a high-performance, low-power Atmel 8-bit AVR RISC-based microcontroller that combines 128KB ISP flash memory, 8KB SRAM, 4KB EEPROM, 86 general purpose I/O lines, 32 general purpose working registers, a real time counter, six flexible timer/ counters with compare modes, PWM, 4 USARTs, byte oriented 2-wire serial interface, 16-channel 10-bit A/D converter, and a JTAG interface for on-chip debugging. The device achieves a throughput of 16 MIPS at 16 MHz and operates between 2.7-5.5 volts.

By executing powerful instructions in a single clock cycle, the device achieves a throughput approaching 1 MIPS per MHz, which balances power consumption and processing speed.

5.3FTDI Basic Breakout

This is a basic breakout board for the FTDI FT232RL USB to serial IC. The pinout of this board matches the FTDI cable to work with the ArduPilot platform. This allows the programming to read and writing the program code to the autopilot board without having to hit the reset button.

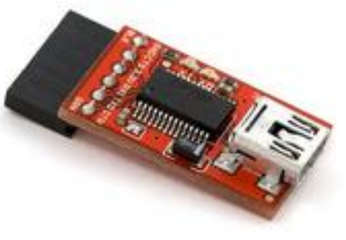


Figure 104FTDI Board

5.4GPS Module

At first, the EM-406A GPS module was tested on ASTRO 15, but during whole flight, the altitude limitation of this module was found at 18,000 meters (60,000ft). It worked until 23,735m (77,850ft); however, above this level, the same altitude of 23,735m was recorded every one second, instead of recording a higher altitude. Therefore, Garmin GPS 18 was used in our system. The new GPS is the same model being used on ASTRO primary tracking gear, so it was guaranteed to work around our desired altitude range.

The GPS 18 includes an embedded receiver and an antenna. Based on the proven technology, it tracks up to 12 satellites at a time while providing fast time-to-first-fix, precise navigation updates, and low power consumption. This GPS adds to the capability of the FAA Wide Area Augmentation System (WAAS) Differential GPS.

This GPS module also provides following features: Main power input of 4.0V ~ 5.5V DC, power consumption 60mA, 0V ~ 4.5V CMOS Serial Output voltage level, -165 dBW minimum receiver sensitivity, 4,800 bps baud rate and NMEA 0183 GGA, GSA, GSV, RMC, VTG, GLL output message, less than 15 m on position accuracy at GPS standard positioning Service, and less than 3m on position accuracy at WAAS..

Figure 105 Garmin GPS Module

The problem of using Garmin GPS was that of the compatibility with Ardupilot. The board was specialized for aEM-406A GPS module, so it supported only Entry level TTL signals. The RS-232 signal from Garmin GPS had to be converted through a signal converter. To this end, SP3232E transceiver was selected and installed between Garmin GPS and Ardupilot. The SP3232E is basically an RS-232 transceiver solution intended for portable or handheld applications, such as notebook or GPS with a high-efficiency, charge-pump power supply that requires only small capacitors.

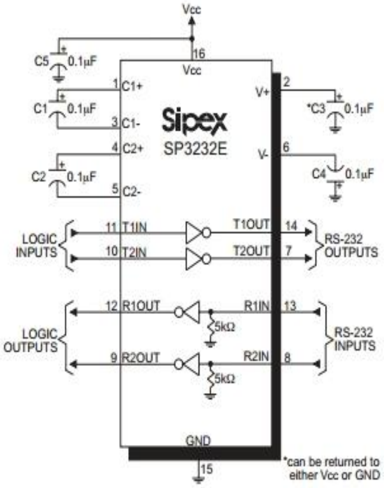


Figure 106Diagram of GPS signal converter

Figure 106 is a test result of the new GPS with Ardupilot with the signal converter:

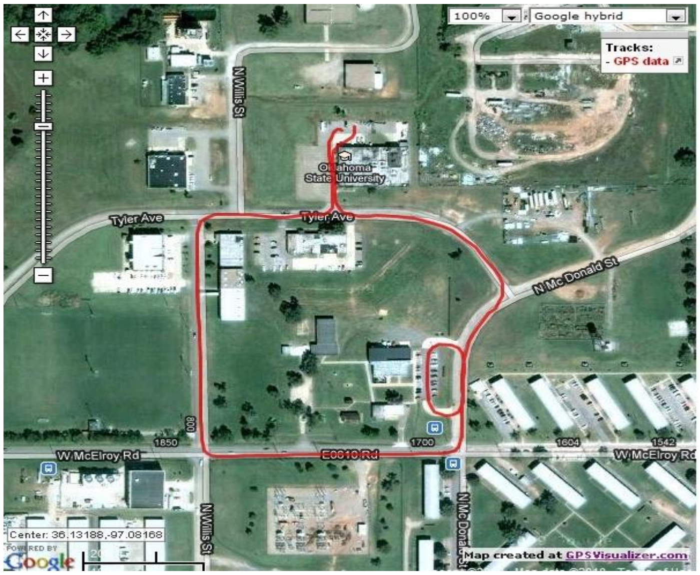


Figure 107New GPS & signal converter field test

As a result, the autopilot can recognize the new GPS signal and transfer the data with the same efficiency.

5.5HMC6343

The HMC6343 is a solid-state compass module with tilt compensation from Honeywell. This device has three axes of magneto, three axes of accelerometer, and a PIC core running all the calculations. The output is a compass heading over an I2Csignal that stays the same even when the board is tilted. Solid-state compasses fail badly when not held flat. The tilt compensated HMC6343 is helpful to indicate the heading direction and the attitude of the parafoil for our autonomous system.

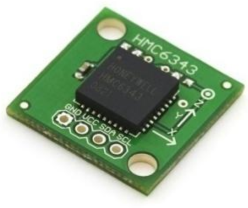


Figure 108HMC6343 magnetometer

5.6Logomatic V2

The Logomatic V2 is an easy to use, adaptable logging device that incorporates the LPC2148 with USB, battery charging, and micro SD support. It uses a USB mass storage stack to appear under any operating system as a flash drive, and logs are created in FAT16 format on the micro SD media. The created data can also be downloaded quickly over a USB connection by dragging and dropping the text files from the device. The micro SD card can also be removed and inserted into a card reader to download the logs.

The board comes with a JST connector to be powered from a line-up of LiPo batteries or other power sources up to 7.5V DC.

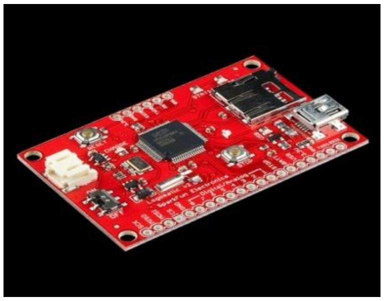


Figure 109Logomatic V2 datalogger

5.7Pico Switch

Pico Switch is a relay switch that can be toggled via a radio controller. In the ASTRO project, the switch was used for Auto Cut-down system that is described. This device can activate applications with voltage levels as high as 125VAC. This plugs into Ardupilot Mega as easily as a servo does and connects a hot wire and battery. The relay is a single pole single throw switch, so possibly two switches are going to be installed. One is for separating the RPV with the balloon, and second one is for the deployment of the parafoil.

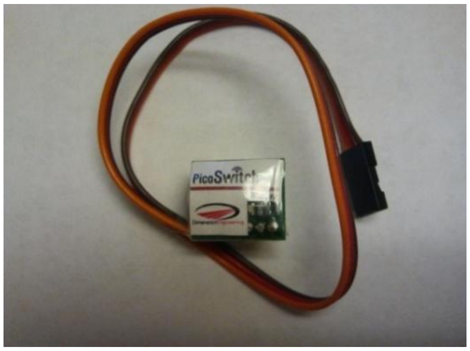
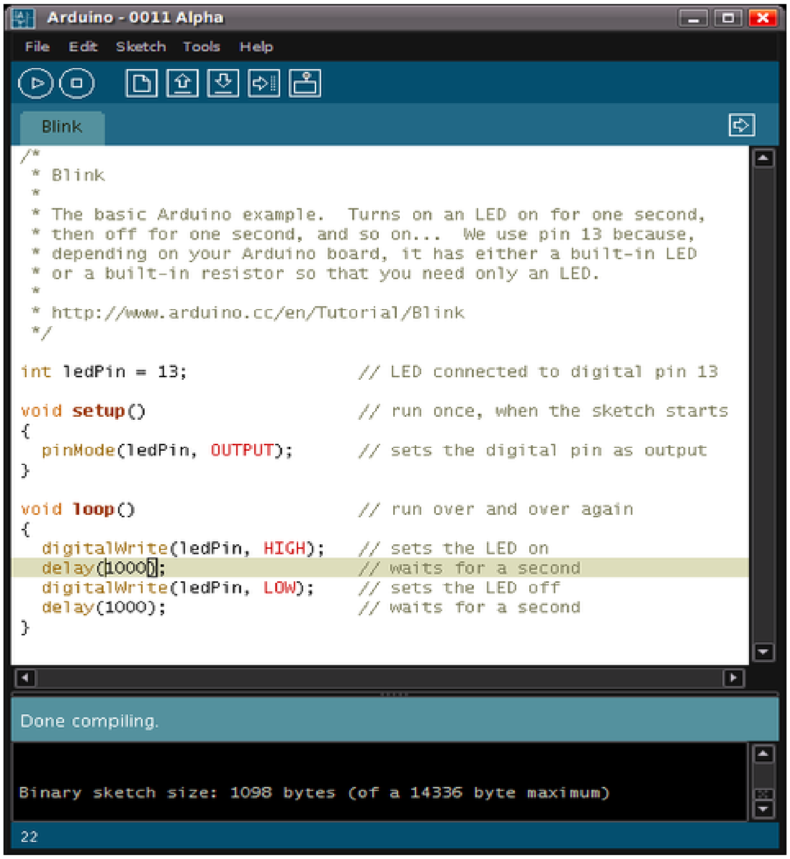


Figure 110Pico Switch for Auto cut-down system

5.8Arduino

Arduino is an embedded system platform based on a simple open hardware design for a single-board microcontroller with embedded I/O support and a standard programming language. The Arduino programming language is based on Wiring, and essentially C/C++.

Programming is usually carried out through the Arduino IDE based on the processing project. Arduino’s programming language itself is based on Wiring and resembles C. Within the Arduino IDE, this is first cross-compiled then downloaded to the attached Arduino board.



```
/*
 * Blink
 *
 * The basic Arduino example.  Turns on an LED on for one second,
 * then off for one second, and so on...  We use pin 13 because,
 * depending on your Arduino board, it has either a built-in LED
 * or a built-in resistor so that you need only an LED.
 *
 * http://www.arduino.cc/en/Tutorial/Blink
 */

int ledPin = 13;                // LED connected to digital pin 13

void setup()                    // run once, when the sketch starts
{
  pinMode(ledPin, OUTPUT);      // sets the digital pin as output
}

void loop()                     // run over and over again
{
  digitalWrite(ledPin, HIGH);   // sets the LED on
  delay(1000);                  // waits for a second
  digitalWrite(ledPin, LOW);    // sets the LED off
  delay(1000);                  // waits for a second
}
```

```
Done compiling.

Binary sketch size: 1098 bytes (of a 14336 byte maximum)
```

Figure 111Arduino compiler

CHAPTER VI

FLIGHT SIMULATION

The six degree of freedom simulations were performed on Matlab. The files implemented the equations discussed in Chapter 2 with parameters in Appendix. All of the equations and parameters seemed to be similar to those found in the literature references [8, 12, 14]. However, not all the previous research includes the non-linear apparent mass terms, even while performing a non-linear simulation, such as Slegers[14]. The reason for this variation was not found during the course of this project. Removing the apparent mass part of the equation resulted in the generation of the expected results; however, this is indicative that further investigation is needed for precise dynamic simulations.

As mentioned in Chapter 3, this parafoil flight requires several steps and different simulation programs for various conditions. First of all, the expected flight paths and burst points should be defined at the Pre-flight program based on the results of the previous research conducted by Dr. Joseph Conner.

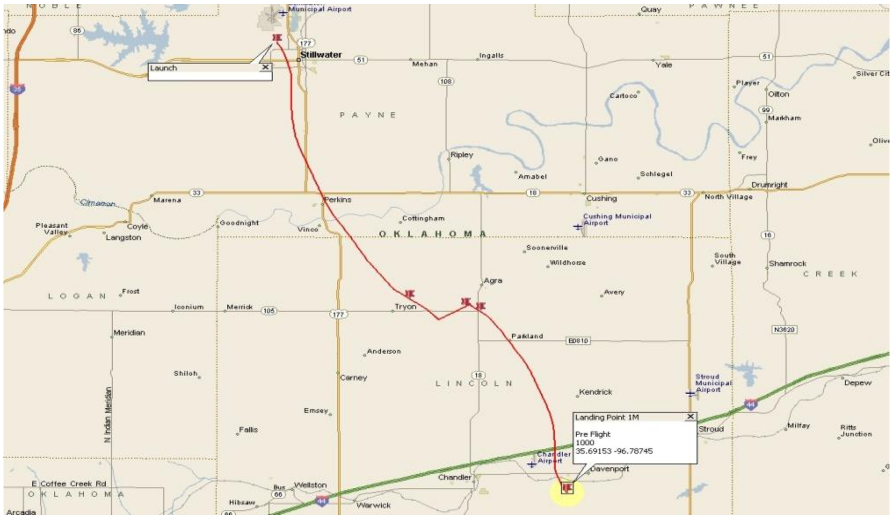


Figure 112Pre-flight

With the burst point from this simulation, the deployment altitude that was a lower speed point than the max parafoil gliding speed was decided according to the wind profile. The parafoil has limitation to glide and control. Therefore, the feasible range to land from the deployment position needs to be calculated.

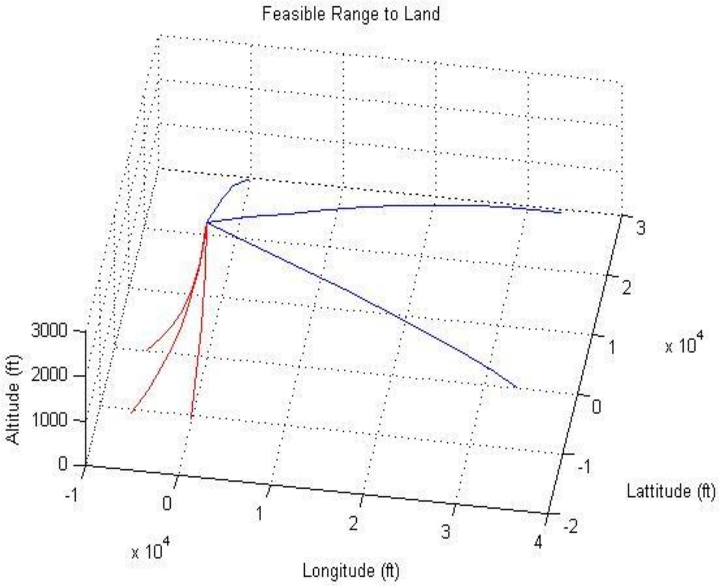


Figure 113Feasible Range to land

From this calculated range, the safe and easily-accessible place can be chosen for the parafoil to land.

6.1 Flight Simulator Verification

For the flight simulation, Matlab/Simulink program was used and applied to the dynamic equations from Chap. 2.

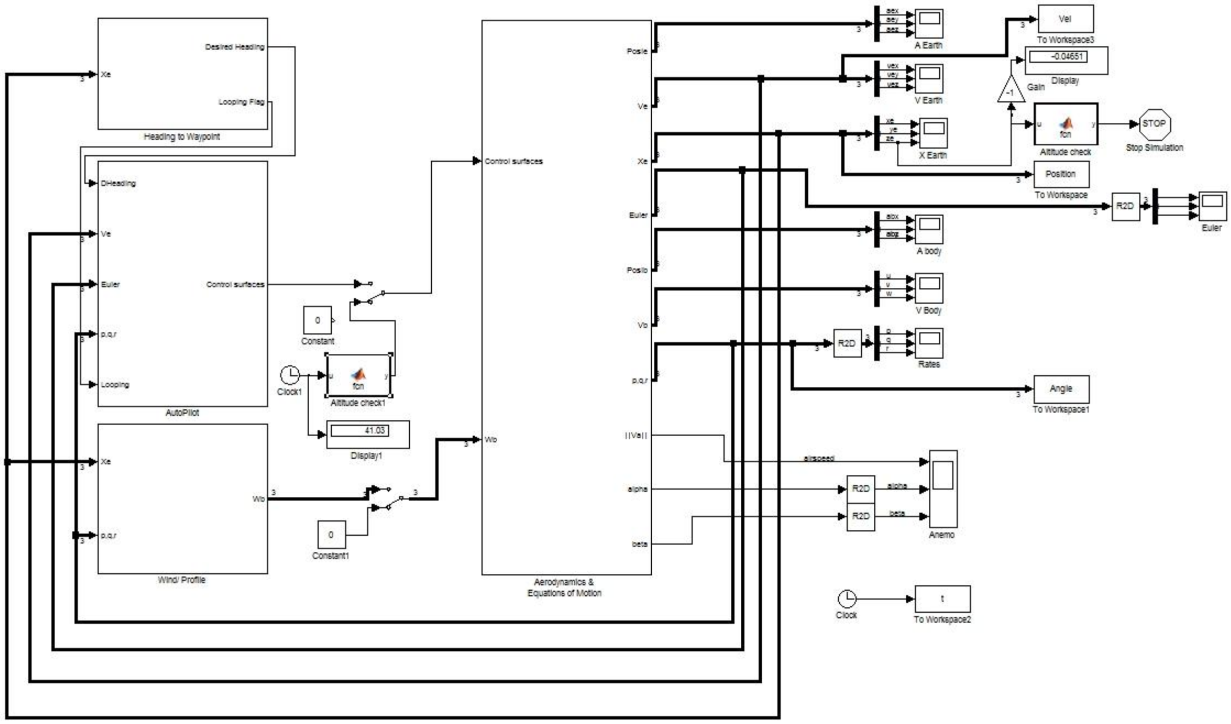
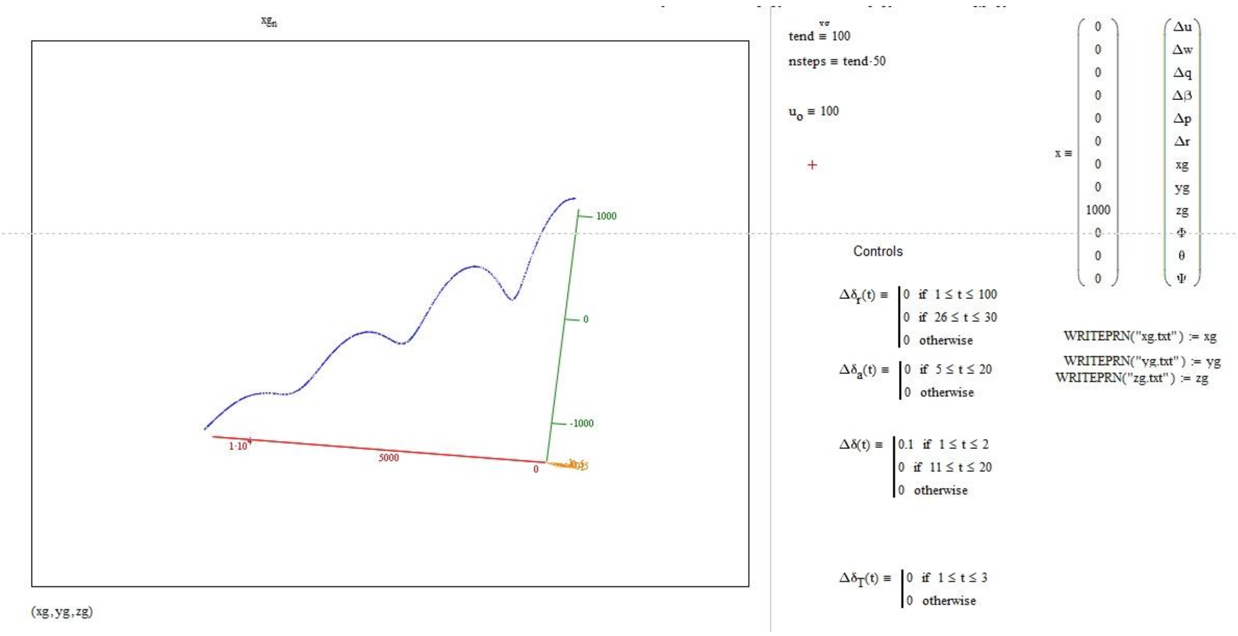


Figure 114Flight Simulation Simulink Module

The program was built but it required verification by the comparison with the other already certified methods. A six DOF flight simulation program from Nelson’s book was chosen. However, both simulation codes were designed to a different scale, in order to use different equations, and also to be designed for different vehicles. Therefore, neither of them had the ability to show exact movements on small input. As a result, Phugoid motion was compared with the fixed wing airplane, NAVION.



$tend \equiv 100$

$nsteps \equiv tend\cdot 50$

$u_o \equiv 100$

$$x \equiv \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1000 \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad \begin{pmatrix} \Delta u \\ \Delta w \\ \Delta q \\ \Delta\beta \\ \Delta p \\ \Delta r \\ xg \\ yg \\ zg \\ \Phi \\ \theta \\ \Psi \end{pmatrix}$$

Controls

$$\Delta\delta_r(t) \equiv \left| \begin{array}{l} 0 \ \text{if} \ 1 \le t \le 100 \\ 0 \ \text{if} \ 26 \le t \le 30 \\ 0 \ \text{otherwise} \end{array} \right.$$

$$\Delta\delta_a(t) \equiv \left| \begin{array}{l} 0 \ \text{if} \ 5 \le t \le 20 \\ 0 \ \text{otherwise} \end{array} \right.$$

$$\Delta\delta(t) \equiv \left| \begin{array}{l} 0.1 \ \text{if} \ 1 \le t \le 2 \\ 0 \ \text{if} \ 11 \le t \le 20 \\ 0 \ \text{otherwise} \end{array} \right.$$

$$\Delta\delta_T(t) \equiv \left| \begin{array}{l} 0 \ \text{if} \ 1 \le t \le 3 \\ 0 \ \text{otherwise} \end{array} \right.$$

WRITEPRN("xg.txt") := xg

WRITEPRN("yg.txt") := yg

WRITEPRN("zg.txt") := zg

Figure 115Phugoid motion from MathCAD

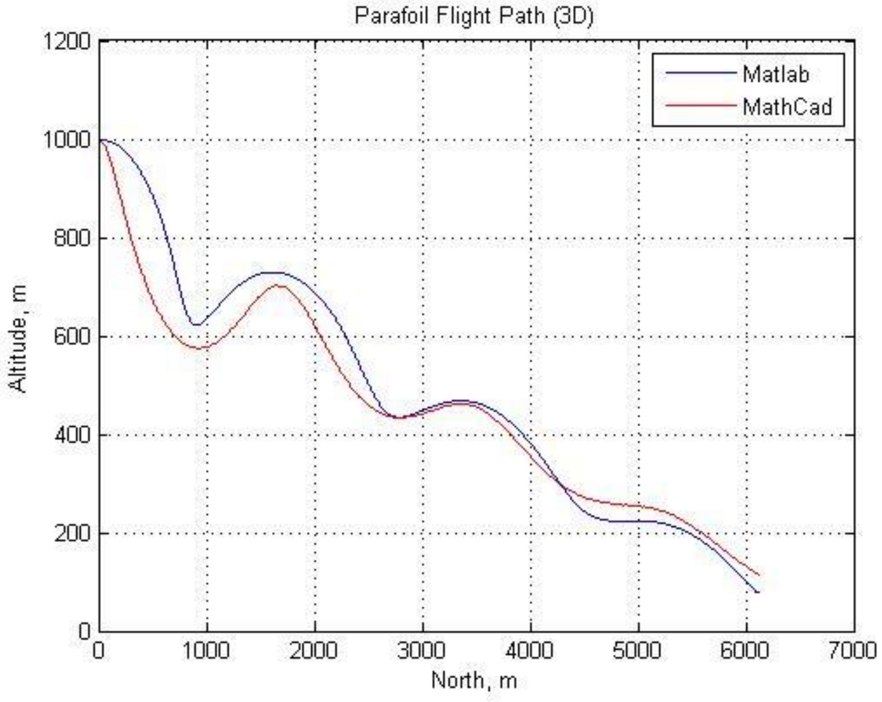


Figure 116Comparison MathCAD and Matlab result

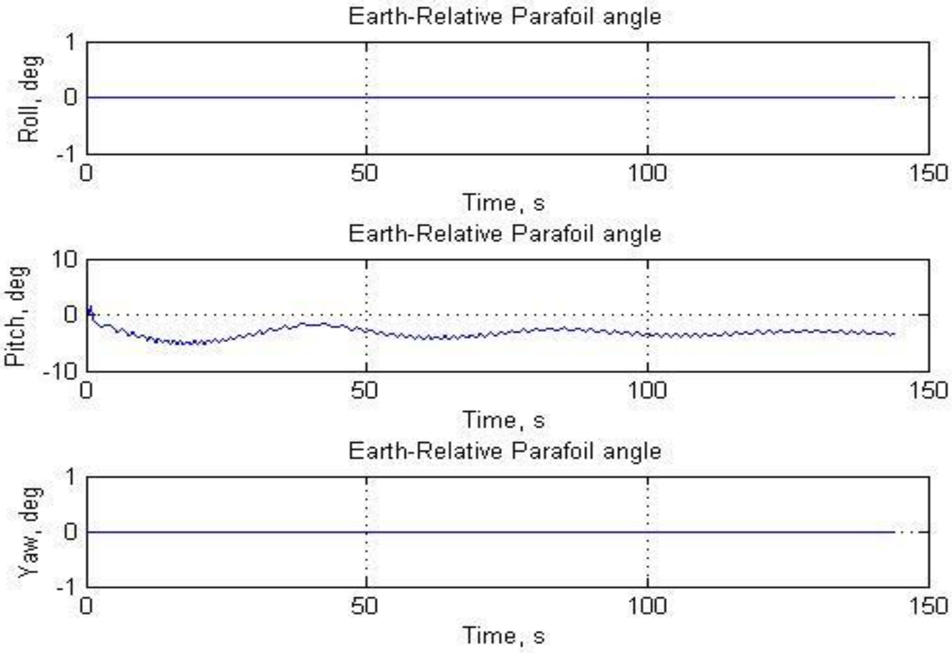


Figure 117Roll, Pitch and Yaw from Matlab

Table 3Phugoid Motion Comparison between Matlab and MathCAD

	MathCAD **(Nelson's)**	**Matlab (Seong-Jin)**	**Difference**
Phugoid	$t_{1/2} = 40.3s$ Period = 29.3s	$t_{1/2} = 37.1s$ Period = 26.8s	5.4% 8.5%

After inputting 0.1 rad. input between the 1st second and the 2nd second on both simulation runs, the flight paths were compared, as shown in Table 3.

As expected, both simulations are not exactly matched, but they show very close enough frequency, in spite of differences.

6.2Parafoil Flight Simulation Result

The simulation was performed mainly on the Simulink module with code generation, definition, and plotting in Matlab.

As mentioned above, there are issues that present a number of challenges to the guidance of our vehicle. The most hindering challenge was the vehicle’s very light weight and slow speed; it was also influenced by the relatively weak winds and disturbances. To examine the performance with wind and the disturbances present, a simple On/Off switch was designed to apply to the wind data and to also take out the wind profile.

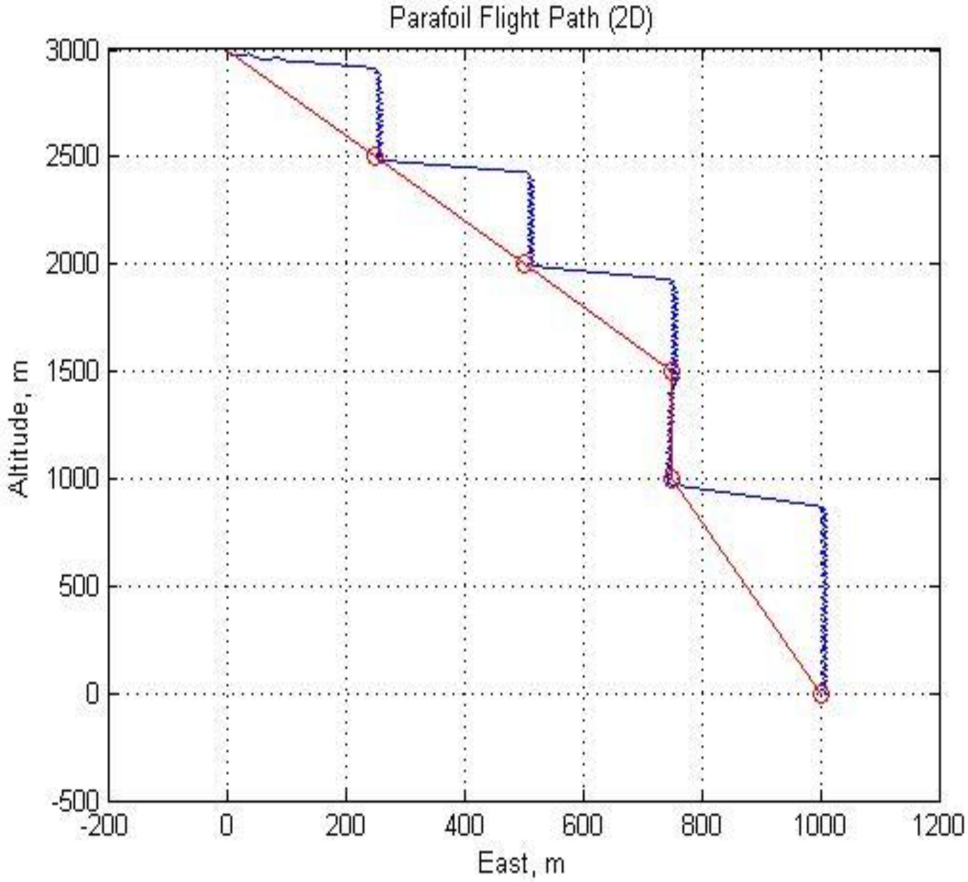


Figure 118Flight path without wind (Side view)

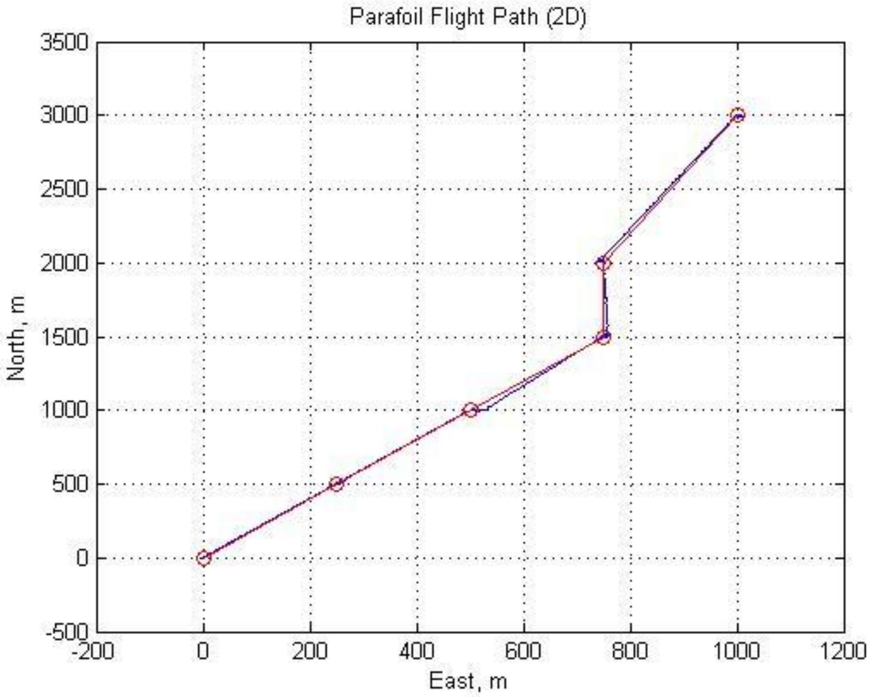


Figure 119Flight path without wind (Top view)

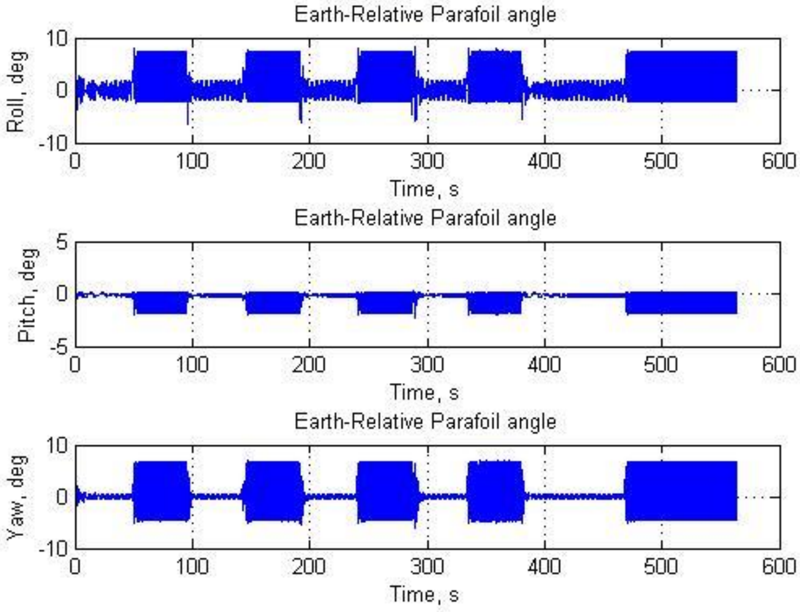


Figure 120Roll, Pitch Yaw response without wind

Without windy conditions present, the vehicle was tracked on the desired path very well. During moving to the next waypoint to reach the designated waypoint, the vehicle took a few seconds to calculate heading to the next point. However, it could navigate the path soon, and accordingly, started to track the path again.

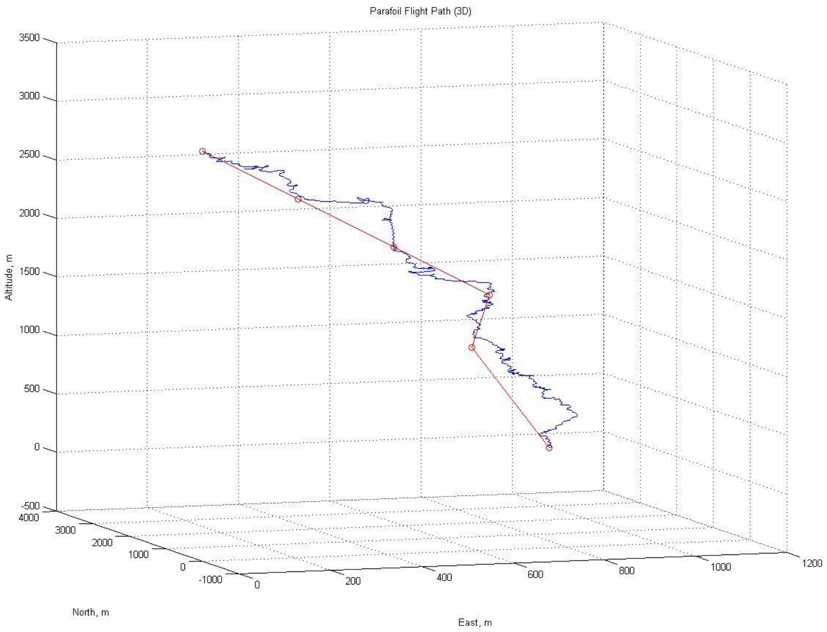


Figure 121Flight path with wind 3D

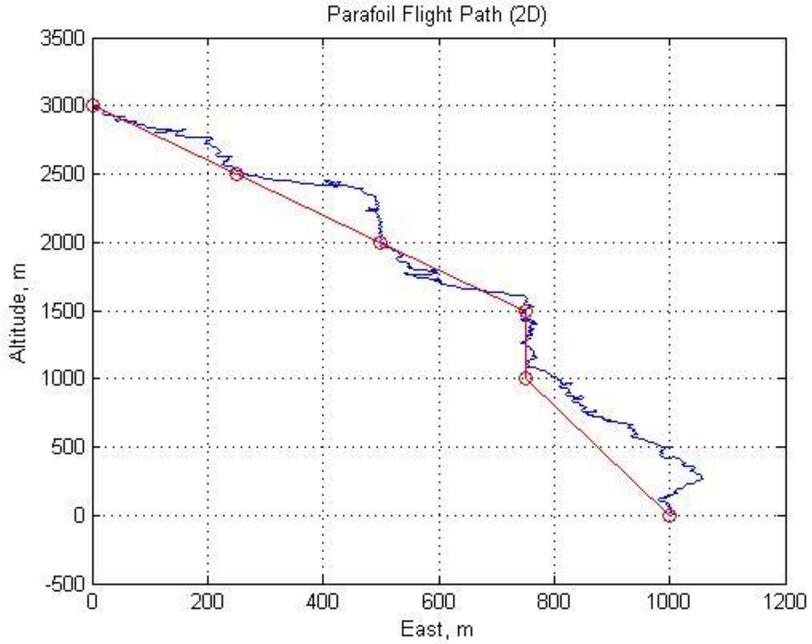


Figure 122Flight path with wind (Side view)

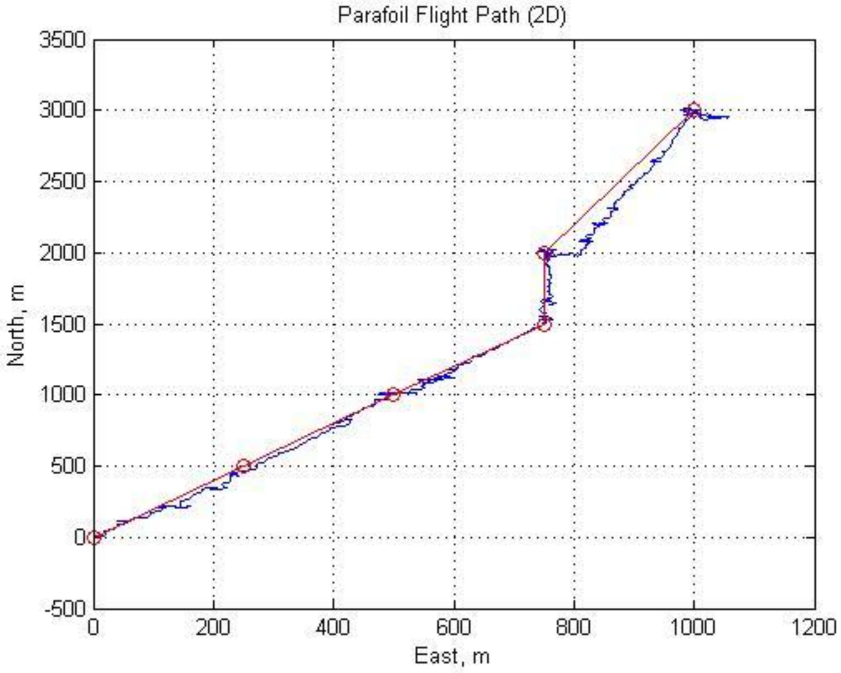


Figure 123Flight path with wind (Top view)

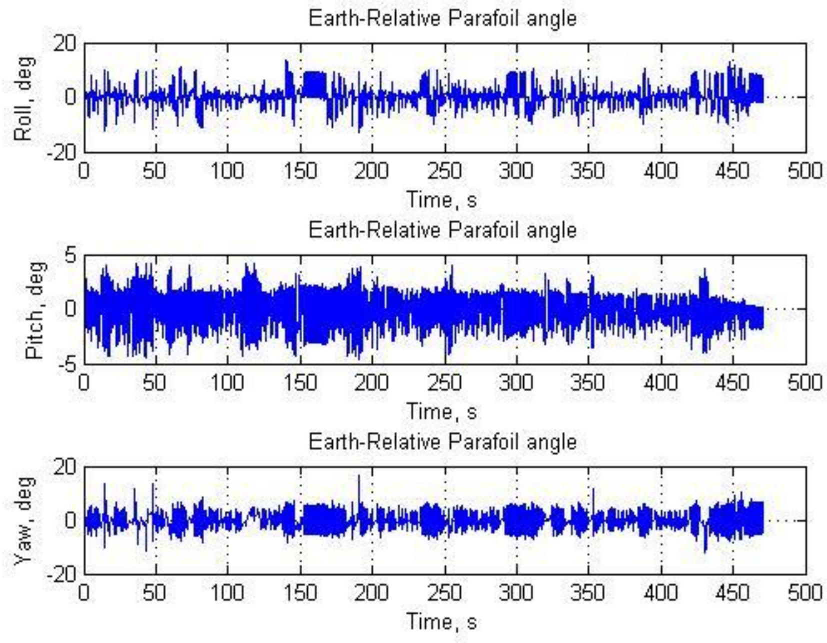


Figure 124Roll, Pitch and Yaw response with wind

On the other hand, in windy conditions, the vehicle showed signs of noisy movement as we had estimated. In slower wind conditions, about half of the max parafoil speed, from 2500m to 2000m, the flight path was similar to the path without the wind. Overall, it can be concluded that the vehicle did not miss the waypoint and tracked the path well during Control stage.

CHAPTER VII

PROPOSED RESEARCH

In recent years, the advent of enabling technology in sensing, computation, and communications has Unmanned Aerial Vehicles (UAVs) to become cost-effective, reliable, and safe alternatives to conventional piloted aircraft in a wide variety of applications. Autonomous vehicles have great potential to perform high precision and repetitive tasks, since a whole new range of sensing devices can provide them with information that is not perceptible by a pilot.

Several different problems with autonomous aircraft can be found in Return to Point Vehicle (RPV) of our project. First of all, the RPV is started from near space, around 100,000ft, and then travel to desired landing point. While the RPV sails to the target several major influences will be encountered. For example, wind, cloud, air density, etc. Among them,the wind as a function of altitude is the most important aspect. In order to gain an understanding of the type of problems that might be encountered several different pathplanning schemes have been looked at.

One of the main advantages of the parafoil method of delivery is the longer flight time, which can increase mission flexibility. With a gliding capability that is unavailable for parachute, a parafoil could potentially be deployed miles away from the predicted point. It could also use that extra time to spiral above the target or the waypoint. But the slow descent complicates other aspects of the mission. Careful attention must be made to the path to target, and a trajectory for the vehicle must be planned in advance. Therefore knowledge of wind speed and direction is required before a successful trajectory can be predicted. The issue becomes a key to successful path planning that will make the vehicle to landon desiredpoint. The biggest issues that will have during the plan and implementation of a path trajectory for the parafoil will be that the winds aloft is change rapidly in magnitude and direction.

After planned the path, there are issues that present a challenge to the guidance of our vehicle. The vehicle is a very light, and its parafoil makes it slow and very susceptible to wind and other disturbances. Keeping the vehicle on a planned trajectory becomes more difficult with the longer flight durations. Accurate guidance requires a decent ability to navigate throughout the descent.

These two issues, path-planning and control to follow the path, are my proposed research topics.

7.1 Proposed Statement of Research

The following is a list of the major tasks that will be carried out in the realization of the proposed research project.

- Testing of a control scheme using a simple 2-D wind model. In this model the winds will remain at a constant velocity but vary in direction with altitude.
- Testing of a control scheme using 3-D wind model. In the 3-D model the winds will vary in both direction and velocity with altitude.
- Analysis of flight control system to determine minimum amount of computational power required to control RPV on return to landing zone. The main objective here is to keep the computational power to a minimum so that it can be implemented using a simple microcontroller such as the BASIC Stamp.
- Testing of flight hardware

- Determine flight performance of RPV
- How does the payload(s) interact with the flight system
- Determine guidance system accuracy
- Signal reacquisition time
- Determine performance of full system in the field
- Testing of system with low altitude drops
- Testing of system with high altitude drops
- Analysis of field test result

REFERENCES

1. *Development of a Return to Point Vehicle for High Altitude Ballooning.* **Joseph P. Conner Jr.** PhD Dissertation, 2009.

2. *Development of a Small parafoil Vehicle for Precision Delivery.* **Damian Toohey** Master's Thesis, 2005.

3. *Dynamic Programming for chemical engineering.* **R. Russell Rhinehart, John D. Beasely**1987. Chemical Engineering.

4. *Process plants, cost estimating to project management, information systems* **R. Russell Rhinehart, John D. Beasely**1993. Encyclopedia of chemical processing and design, Marcel Dekker, Inc.

5. *Autonomous Target Following by Unmanned Aerial Vehicles.* **Rafi, F., Khan, S, Shafiq, K., Shah, M.** 2006, SPIE Defense and Security Symposium.

6. *Optimal Path Planning in a Constant Wind with a Bounded Turning Rate.* **Mcgee, T., Spry, S., and Hedrick, J.** 2005, AIAA-2005-6186.

7. *Conversion of Earth-centered Earth-fixed coordinates to geodetic coordinates.* **Zhu, J.** 1994, IEEE Transactions on Aerospace and Electronic Systems , pp. 957-961.

8. *Optimal Path Planning for Unmanned Air Vehicles with Kinematic and Tactical Constraints.* **Yang, G., Kapila, V.** 2002, IEEE, pp. 1301-1306.

9. *Griffon: A Man-Portable Hybrid UGV/UAV.* **Yamauchi, B., Rudakevych, P.** 2004, Industrial Robot, pp. 443-450.

10. *On the Development of a Scalable 8-DOF Model for a Generic Parafoil-Payload Delivery System.* **Yakimenko, O.** 2005, AIAA, pp. 642-654.

11. *Applications of Scientific Ballooning Technology to High Altitude Airships.* **Smith, M., Rainwater, E.** 2003, AIAA.

12. *The Return of the Balloon as an Aerospace Test Platform.* **Smith, M. and Allision, G.** Arcachone, France : AeroStar, 1999. ESA Reentry Vehicles Symposium.

13. *Steering Algorithms for GPS Guidance of RAM-AIR Parachutes.* **Smith, B.** 1995, Proceedings of the 8th International Technical Meeting of the Institute of Navigation, pp. 12-15.

14. *Aspects of Control for a Parafoil and Payload System.* **Slegers, N., Costello, M.** 2003, Journal of Guidance, Control, and Dynamics, pp. 898-905.

15. *Development of Flight Test of a Deployable Precision Landing System.* **Sim, A., Murray, J., and Neufeld, D.** 1994, AIAA Journal of Aircraft, pp. 1101-1108.

16. *Guidance and Control for Flat-Circular Parachutes.* **Scott, D., Richard, B., and Glen, B.** 2001, Journal of Aircraft, pp. 809-817.

17. *Unmanned Aerial Vehicle Path Following for Target Observation in Wind.* **Rysdyk, R.** 2006, AIAA Journal of Guidance, Control, and Dynamics.

18. *The Trans-Pacific Crossing: Long Range Adaptive Path Planning for UAVs Through Variable Wind Fields.* **Rubio, J. , Kragelund, S.** 2003. AIAA/IEEE Digital Avionics Systems Conference.

19. *Unmanned Aerial Vehicle Path Following for Target Observation in Wind.* **Rolf, R.** 2006, Journal of Guidance, Control, and Dynamics , pp. 1092-1100.

20. *A Flight Simulation Algorithm for a Parafoil Suspending an Air Vehicle.* **Redelinghuys, C.** 2007, Journal of Guidance, Control, and, Dynamincs, pp. 791-803.

21. *Multibody Dynamics of Parachute and Balloon Flight Systems for Planetary Exploration.* **Quadrelli, M., Cameron, J., and Kerzhanovich, V.** 2004, Journal of Guidance, Control, and Dynamics, pp. 564-571.

22. *Modeling and simulation of a ship launched glider cargo delivery system.* **Puranik, A. , Parker, G., Passerelle, C., Bird III, J., Yakimenko, O., Kaminer, I.** 2006, AIAA Guidance, Navigation, and Control Conference, pp. 5332-5341.

23. **Prakash, O.***Aerodynamics, Longitudinal Stability and Glide Performance of Parafoil/Payload System.* Bombay : Department of Aerospace Engineering, Indian Institute of Technology, 2004.

24. *Entry Guidance and Trajectory Control for Reusable Launch Vehicle.* **Ping, L.** 1997, Journal of Guidance, Control, and Dynamics , pp. 143-149.

25. *Vector Field Path Following for Miniature Air Vehicles.* **Nelson, D.R., Barber, D.B., McLain, T.W., and Beard, R.W.** 2007, IEEE Transactions on Robotics, pp. 519 - 529.

26. *Entry Guidance and Trajectory Control for Reusable Launch Vehicle.* **Lu, P.** 1997, Journal of Guidance, Control, and Dynamics, pp. 143-149.

27. *Novel Method for Identification of Aircraft Trajectories in Three-Dimensional Space.* **Korgul, A.** 2000, Journal of Guidance, Control, and Dynamics, pp. 1021-1029.

28. *On the Development of GNC Algorithm for a High-Glide Payload Delivery System.* **Kaminer, I., Yakimenko, O.** 2003, Proceedings of the 41st IEEE Conference on Decision and Control Volume 5, pp. 5438 – 5443.

29. *Development of Control Algorithm for the Autonomous Gliding Delivery System.* **Kaminer, I., Yakimenko, O.** 2003, AIAA.

30. *Testing of Guided Parafoil Cargo Delivery Systems.* **Kaesemyer, S.** 2005, 18th AIAA Aerodynamic Decelerator Systems Technology Conference and Seminar. AIAA 2005-1668.

31. *Trajectory Tracking for Autonomous Vehicles: An Integrated Approach to Guidance and Control.* **Isaac, K., Antonio, P., Eric, H., and Carlos, S.** 1998, Journal of Guidance, Control, and Dynamics, Vol. 21, pp. 29-38.

32. *Lateral-Directional Aerodynamics from a Large Scale Parafoil Test Program.* **Iacomini, C, and Cerimele, C.** 1999, AIAA, pp. 218-228.

33. *Applying Parachute Canopy Control and Guidance Methodology to Advanced Precision Airborne Delivery Systems.* **Hogue, J., and Jex, H.** 1995, AIAA Aerodynamic Decelerator Systems Technology Conference, pp. 49-59.

34. **Hoerner, S. F.***Fluid Dynamic Drag.* Bakersfield : authors family Hoerner Fluid Dynamics, 1965.

35. *Autonomous Large Parafoil Guidance, Navigation and Control System Design Status.* **Hattis, P., Carter, D., George, S., McConley, M., Rasmussen, S., Singh, L., Tavan, S.** 2007, 19th AIAA Aerodynamic Decelerator Systems Technology Conference and Seminar.

36. *Optimal Control of a Gliding Parachute System.* **Gimadieva, T.** 2001, Journal of Mathematical Sciences, pp. 54-60.

37. *Guidance and Control for Flat-Cirular Parachutes.* **Dellicker, S., Benney, R., and Brown, G.** 2001, Journal of Aircraft, pp. 809-817.

38. *An Optimal Guidance Law for Planetary Landing.* **D'Souza, C.** 1997, Proceeding of the AIAA Guidance, Navigation, and Control Conference.

39. *Autonomous Large Parafoil Guidance, Navigation, and Control System Design Status.* **Carter, D., George, S., Hattis P., McConley, M., Rasmussen, S., and Singh, L.** Williamsburg, VA : AIAA, 2007. 19th AIAA Aerodynamic Decelerator Systems Technology Conference and Seminar.

40. *Autonomous Guidance, Navigation, and Control of Large Parafoils.* **Carter, D., George, S., Hattis, P., Singh, L., and Tavan, S.** 2005, AIAA Aerodynamic Decelerator Systems Conference .

41. **Bryson, A.***Dynamic Optimization.* s.l. : Addison Wesley Longman, 1999.

42. *Principles of Guidance-Based Path Following in 2D and 3D.* **Breivik, M. and Fossen, T.** 2005, 44th IEEE Conference on Decision and Control, pp. 627 – 634.

43. *Guidance-Based Path Following for Autonomous Underwater Vehicles.* **Breivik, M. and Fossen, T.** 2005, IEEE.

44. *Aerodynamic Improvements of Paraglider Performance.* **Babinsky, H.** 1999, AIAA, pp. 362-367.

45. *Novel Method for Identification of Aircraft Trajectories in Three-Dimensional Space.* **Artur, K.** 2000, Journal of Guidance, Control, and Dynamics, pp. 1021-1029.

46. *Trajectory-Tracking and Path-Following of Underactuated Autonomous Vehicles With Parametric Modeling Uncertainty.* **Aguiar, A., Hespanha, J.** 2007, IEEE Transactions on Automatic Control, pp. 1362-1379.

47. **NASA Glenn Research Center.** Earth Atmosphere Model.

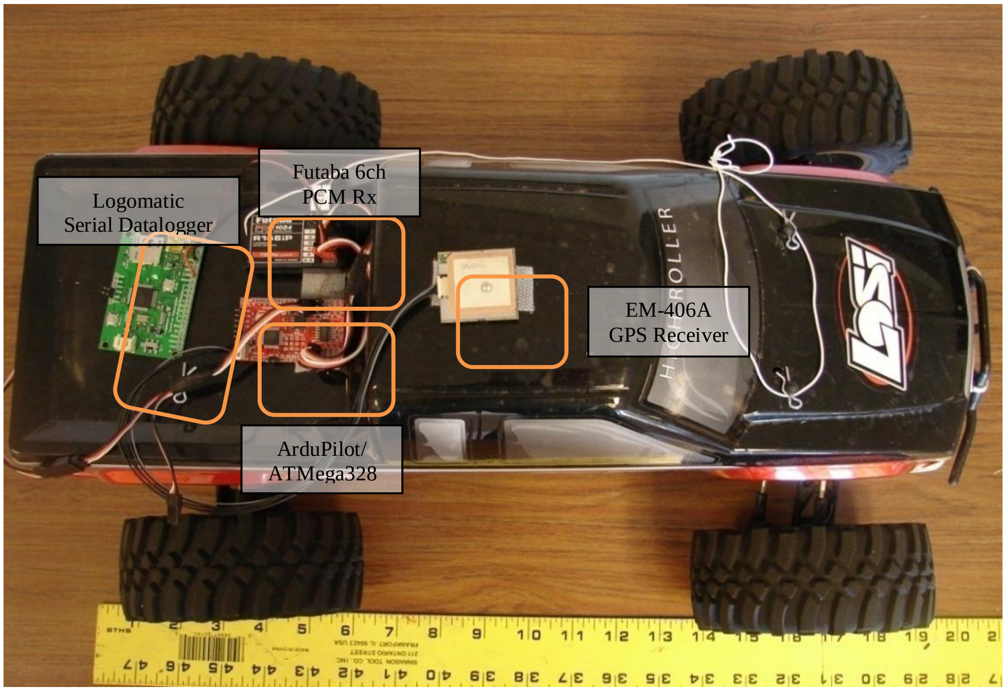
48. MMIST Sherpa - GPS Guided Parafoil. *MMIST*. [Online] Mist Mobility Integrated Systems Technology Inc.

APPENDIX A

Previous Ground & Flight Test

A. Ground Test

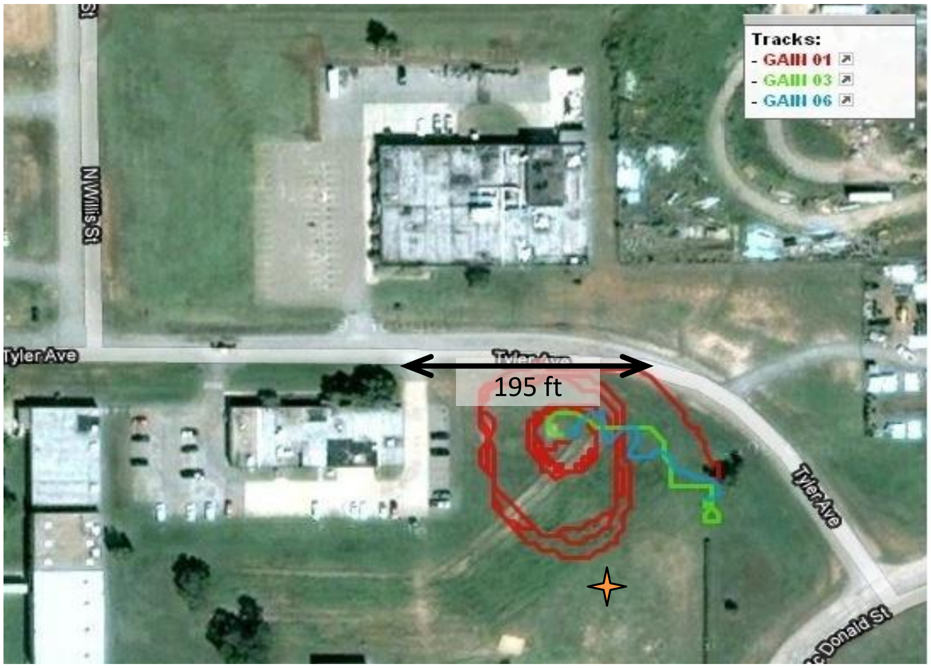
For initial debugging of the system, we decided to implement the autopilot on a ground vehicle. The ground vehicle chosen was Losi's HIGHroller Lifted Truck. This vehicle was selected for several reasons: The ground vehicle allowed safe testing and debugging of the autopilot hardware and software. The ground system could be tested regardless of wind conditions. The vehicle was large enough to support all the gear required for the autopilot, as well as external data logging devices. And the vehicle turned by use of a single servo with 'Turn right' and 'Turn left' commands.



ParaCar Autopilot Testbed

In order to test the autopilot, it was decided to start with a very small gain of 0.1 and then observe the ground track to determine the systems behavior. Before the test was initiated, the vehicle was assigned a home waypoint (36.132332°N, 97.081171°W) and then moved to a start location 150 feet to the southeast, and given a heading of 110°. After placement at this location, the autopilot was started, and the vehicle was given a small forward velocity, approximately 5 mph, and then allowed to autonomous navigate its way back to the home target. Once the vehicle reached to within a 2 meter radius of the target, the logger was stopped and the data from the vehicle was processed. This process was then repeated for a total of six gains ranging from 0.1 to 0.6. We decided to plot the results of three of the gains to show the effects of the gain on the tracking of the system to the assigned home. The results for gains 0.1, 0.3, and 0.6 were selected as they demonstrated the three basic cases of tracking a single point with only a proportional controller. The resulting ground tracks were overlaid on an aerial map of the field using the website GPS Visualizer. The resulting behavior of each of the gains indicates how the vehicles search for the waypoint is directly affected by its gain setting.

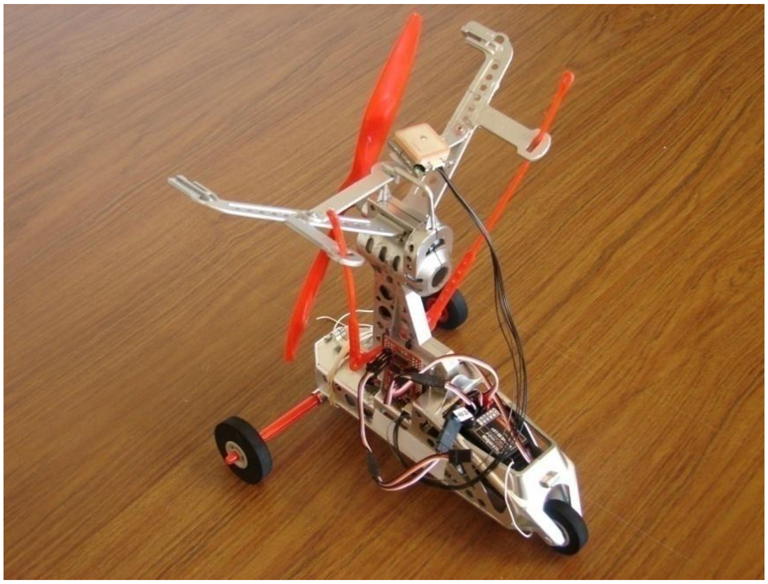
The ground track for the 0.1 gain shows a tendency for the vehicle to 'spiral' in toward the target. If one follows the ground track, they will note that for each pass of the target the distance from the target decreased. The overall time for the 'Para-Car' to reach the target was well over 440 seconds; and on the first circuit around the target, the maximum error was on the order of 150 feet. The gains of 0.3 and 0.6 both dramatically improved the time to target as both were able to reach the target in just over 60 seconds. The major difference between the two gains is that the 0.6 gain resulting in some over correction to the target, which can be seen in the serpentine ground track. Since gains below 0.3 tended to produce a spiral ground track and gains above it serpentine, it was decided to use the gain of 0.3 for the parafoil runs.



ParaCarAutopilot Ground Tests with Varying Gains

B. Flight Test (Low Altitude with Small parafoil)

We decided to move to the next stage of testing the autopilot on board a powered parafoil. The parafoil selected was a COTS powered parafoil produced by Gold Rosita. This parafoil has the following dimensions: a span of 50 inches, a chord of 18 inches, an aspect ratio of 2.78, and a flying weight of 0.68 lbs. The system selected is actually a powered parafoil, but this selection of powered does not affect the behavior of the system in heading. The only advantage the powered system has over a non-powered parafoil is that the system can be hand launched and then use power to gain altitude. In a near-space mission, the balloon would actually provide the system with its initial altitude and thus would necessitate the need for power.



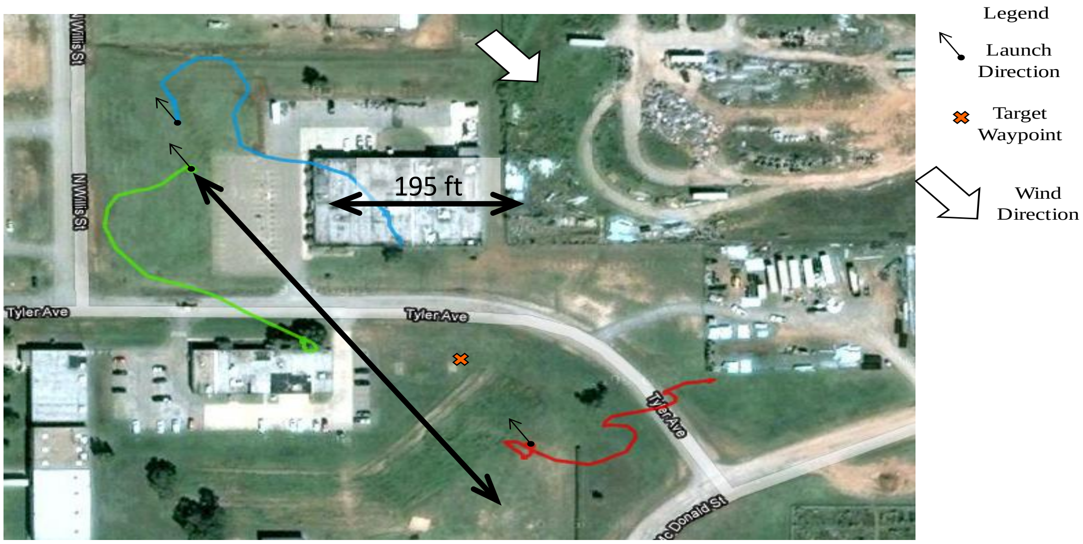
Parafoil Gondola Testbed

Parafoil in Flight

Once the parafoil had been selected several modifications had to be made in order for it to correctly interface with the autopilot. The first major modification made was to remove the controller that came with the system. This controller managed the power settings for the motor as well as translated the pilots turn commands into servo commands. Since it was still desirable to preserver both motor and servo commands, the original controller was replaced by a Futaba six channel PCM receiver and a speed controller. The power for the autopilot and receiver came from the speed controllers built in Battery Elimination Circuit or BEC.

Once the autopilot had been interfaced with the parafoil it was time to test fly the unit in the field. Since it had been found through ground tests that a gain of 0.3 produced desirable results, the gain was left at this setting. Again the system was taken outside and assigned a landing waypoint (36.132332°N, 97.081171°W). To conduct this test, we decided to test how well the parafoil behaved under two different conditions: upwind and downwind. The first test was conducted with the launch location approximately 150 feet downwind of the landing point. The parafoil was launched and then just afterwards the autopilot was turned on. The ground track, shown below in red, indicates that while the parafoil was not able to penetrate the winds, it was able to point in the direction of the home. The reason the parafoil never advanced toward the target was the during the tests the wind speed was noted to be approximately 10 MPH out of NNE and the parafoil's maximum flight speed is only 10MPH, so it was not physically able to penetrate the winds to reach the target.

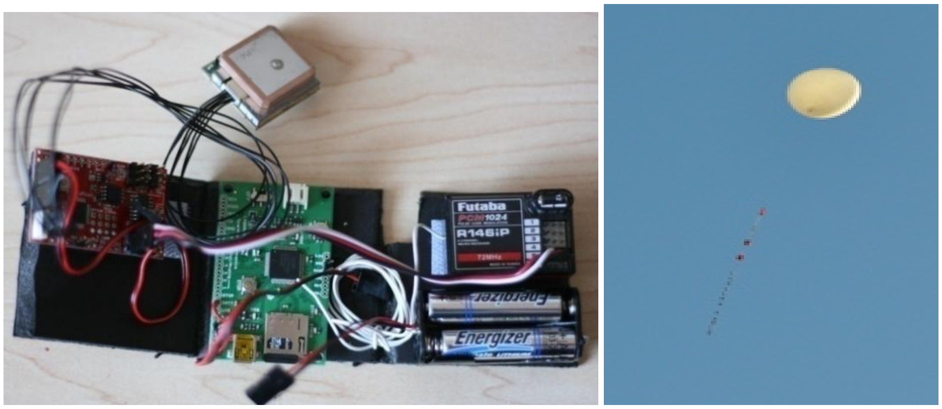
The next two tests were conducted upwind of the landing point at distances of approximately 400 and 460 feet upwind of the landing point. The first upward flight, shown in the green ground track, was launched from an upward distance of 400 feet from the landing zone and again just after launch the autopilot was switched on. During this flight the autopilot initiated the flight with a counterclockwise turn until it had aligned itself with the landing point and then attempted to maintain this heading for the remaining flight. Unfortunately before the parafoil was able to reach the target landing zone, its flight path intercepted a tree which abruptly ended the flight attempt. Shortly after retrieving the parafoil from the tree the second upwind flight was made. This flight was initiated at a distance of 460 feet upwind from the landing point. Again shortly after launch, the autopilot was started and took control of the parafoil. This time the autopilot initiated a clockwise turn toward the landing zone and then continued correct the heading until it had actually overshot the landing zone at which time it commanded a clockwise turn. This correction to the ground track continued until the system had locked until a relative straight line solution to the landing point. This flight was terminated shortly after the parafoil's glide slope intercepted the roof of the DML, which was the last recorded position as seen in the figure below.



Parafoil Autopilot Aerial Ground Tracks

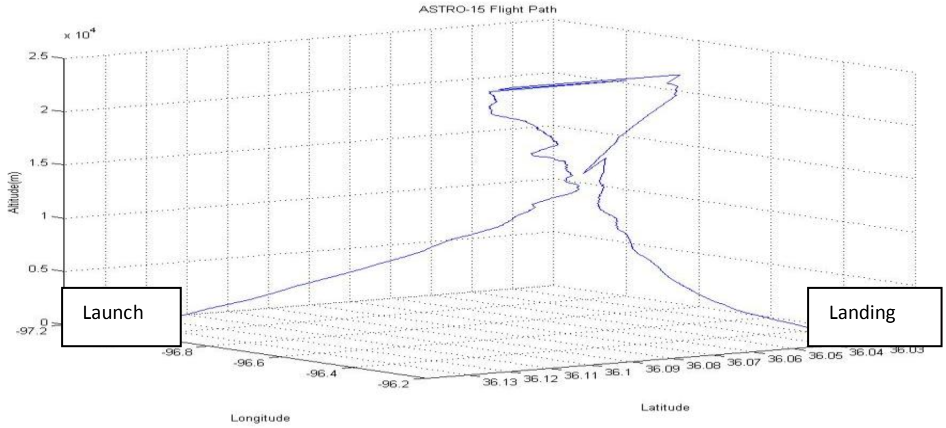
C. Hardware Field Test

Before the flight test, we decided to make sure all component of hardware is appropriate in real flight. For this, we build small payload that includes Autopilot, GPS, RC receiver, data logger and battery shown Figure. Same program with Gold Rosita flight test was loaded on Autopilot.



Autopilot payload &ASTRO-15

The payload was launched on ASTRO-15 that was for student-built satellite test on November 2009.

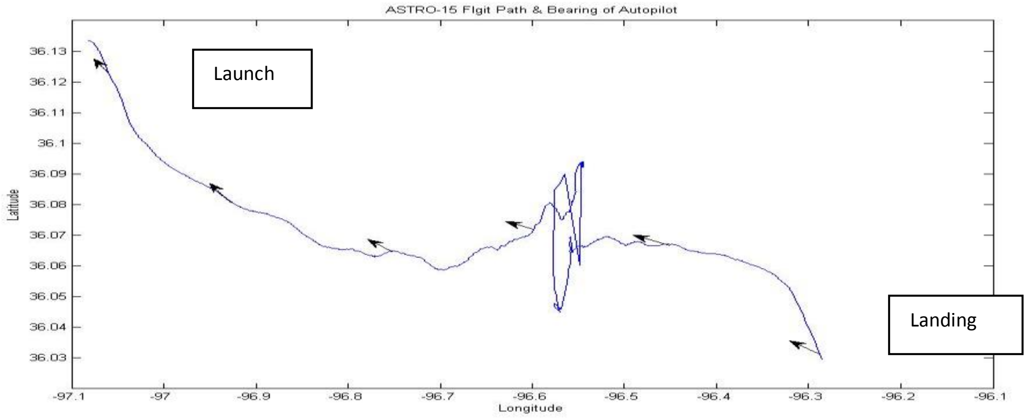


ASTRO-15 Flight Path from Autopilot payload

During this whole flight, all components worked very well in harsh condition of high altitude as shown Figure. As refer, the GPS has altitude limitation up to 18,000 meter (60,000), but it worked until 23,735m (77,850ft), instead of the updating a higher altitude, same altitude (23,735m) was recorded every one second.

Due to this GPS limitation issues, this GPS (EM-406A) will be replaced to Garmin GPS 18 unit that is used in primary tracking module like Figure.

Autopilot Bearing Data



Autopilot Bearing Data

From Figure, the Autopilot's bearing was varying between 270° to 305°. It was reasonable result which is head to DML, Stillwater, which is designated to 'Home'.Our test objects were accomplished from this flight. All of Autopilot part workedvery well in high altitude flight test.

APPENDIX B

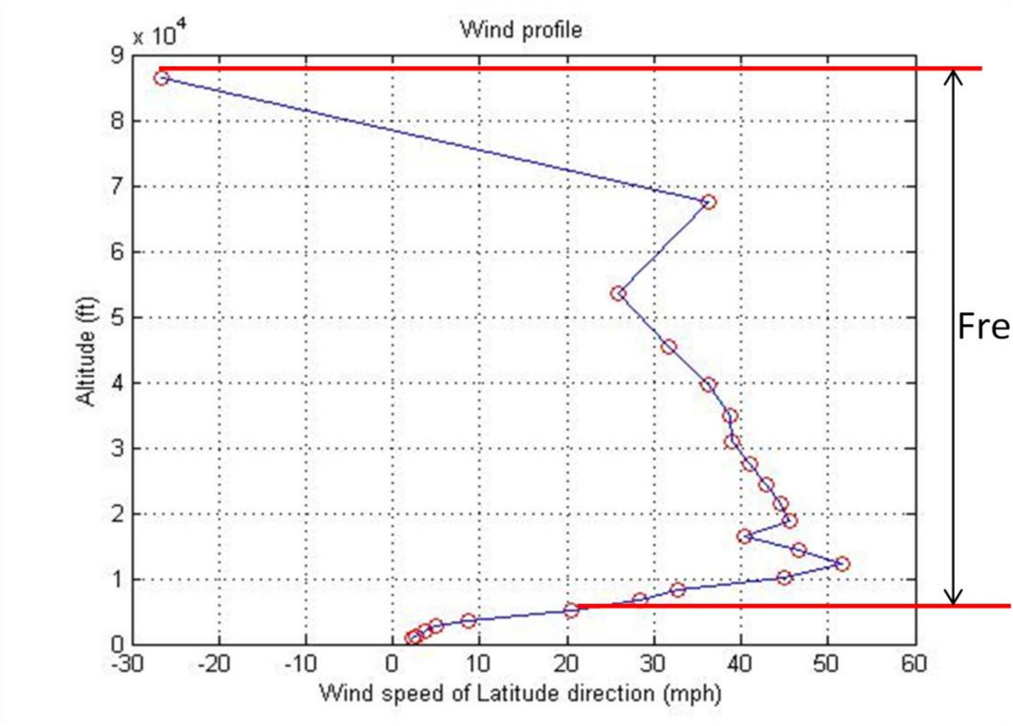
Trajectory Planning

One of the main challenges of this project was the trajectory planning. The payload was dropped from a very high altitude (about 100,000ft), and the wind also varied during long flight distances. Two delivery methods, HAHO (High Altitude – High Opening) and HALO (High Altitude – Low Opening), were considered and compared for the payload as follows:

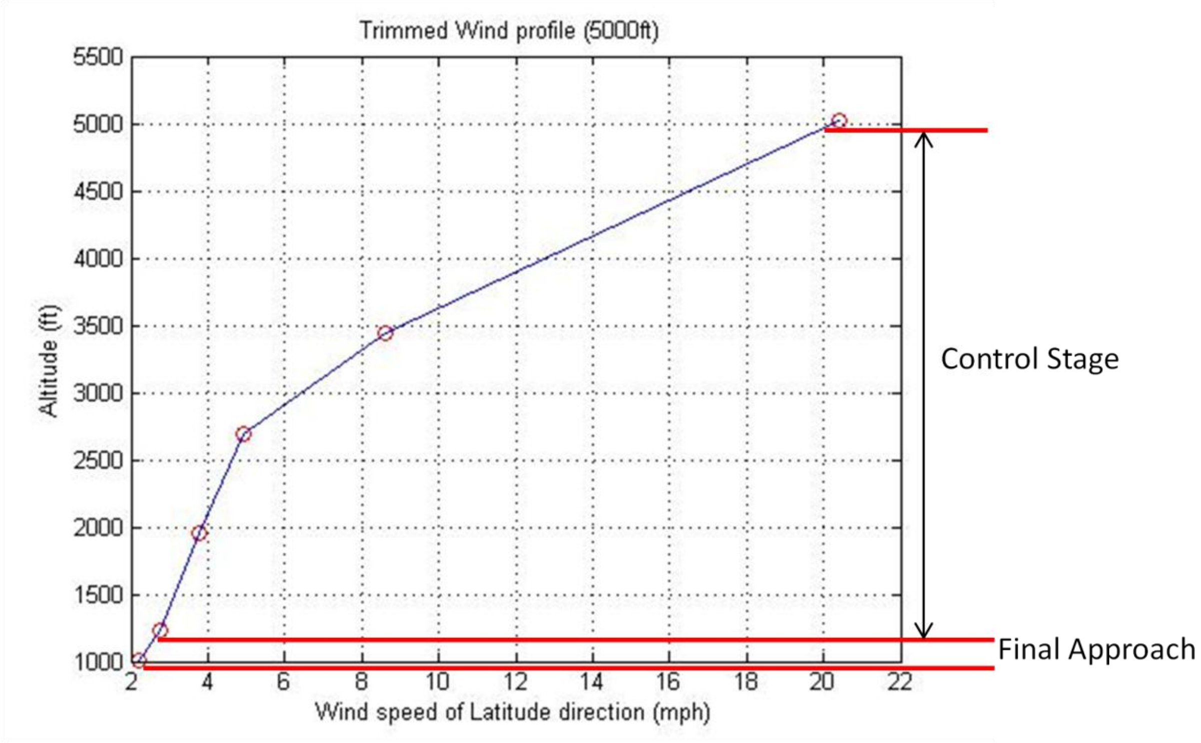
Comparison with HAHO and HALO

		HAHO	HALO
	Weight		
Reliability	5	-1	1
Simplicity	2	1	0
Cost	2	1	0
Safety	3	-1	1
Flight time	1	-1	1
Total		-5	9

After the comparison, HALO was decided to suitable method to use this project. Therefore, we classified the descent phase in three different stages: Free-fall (HALO) stage, Control stage, and Final approach. Figure is the sample wind profile able to launch the balloon with stable atmosphere conditions. Generally, the payload is released around 100,000ft, but the wind profile is provided from about 90,000ft. In this plot, high wind area is from the top altitude to 50,000ft, when the estimated maximum parafoil speed was 20mph. Therefore, this range is defined as Free-fall stage which is from the burst point to the equal point which is both wind speed and maximum parafoil speed. In this stage, parafoil is wrapped in the back, because this vehicle can't make any efficient movement in the high wind area. In figure, Control and Final approach stages are also drawn. Control Stage is from the equal point to 1000ft from the ground. The wind speed of the Control stage is lower than the parafoil's speed, so the parafoil can be controllable. Final approach is the phase approach with heading into the wind for safe landing on the ground.



Wind profile of Free-fall stage



Wind profile of Control stage and Final approach

This is the proposed procedure for the trajectory planning:

Step 1: Simulating and estimating the path for the ascent phase and the burst point from the pre-flight programs.

Step 2: Collecting the wind data from the burst point to the ground.

Step 3: Setting the deployment altitude (D-Alt) with the analyzed wind data.

Step 4: Calculating the possible landing range started from the D-Alt (Matlab).

Step 5: Picking the safe and easily-accessible landing point from the range.

Step 6: Finding the optimized waypoint and path from Matlab/Simulink’s path-finding program.

Step 7: Running the flight simulation program and verifying the path.

For Step 1, the pre-flight program by a previous researcher, Dr. Conner, was used, and for all the other steps, the appropriate Matlab simulation programs were built. The programs will be explained in Control Stage (3.2). Next, we will go over the details of each stage.

A.Free-fall Stage

Free-fall stage is the phase without any dragging component and control system. This idea comes from High Altitude Low Opening (HALO) jump of Special Forces in the Army. It is the method of delivering personnel, equipment, and supplies from a transport aircraft at a high altitude via free-fall parachutes. In this technique, the parachutists open their parachute at a low altitude (between 2,500ft and 3,000ft) after free-falling for a period of time and being dispatched from a high altitude (between 15,000ft and 35,000ft). The biggest advantage of this idea in our project is that there is no distance loss when the vehicle and payloads go through the area of higher wind speed than the maximum parafoil gliding speed. Also, this method will save considerable flight time.

The challenging problems in this stage are the high falling speed and the deployment of parafoil at a desired altitude with high accuracy. To better understand the circumstances during the free-falling stage, the descent speed of the payload is calculated. An object falling towards the surface of the Earth will fall 32.18 ft/s faster every second from basic physics. The reason an object reaches a terminal velocity is that the drag force resisting motion is approximately proportional to the square of its speed. Mathematically, terminal velocity – without considering buoyancy effects – is given by:

$$V_t = \sqrt{\frac{2mg}{\rho A C_d}}$$

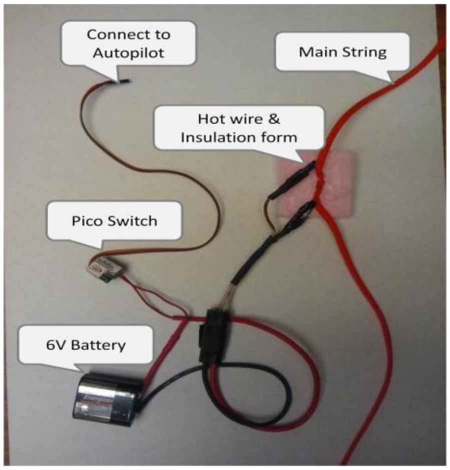
where V_t = terminal velocity, m = mass of the object, g = gravity acceleration, C_d = drag coefficient, ρ = density of the air, and A = projected area.

From equation #, we can calculate the terminal velocity of our payload at any desired altitude like follows, where m = 12lb, g = 32.18 ft/s^2, C_d = 0.8 (Cube's drag coefficient), ρ = 0.074887 lb/ft^3 and A = 0.5 ft^2.

As a result, the estimated terminal velocity of the payloads is 160.57 ft/s.

Due to this high velocity, the deployment of the parafoil produces high impact on canopy and strings between the canopy and payload; therefore, a small parachute which results in the smooth deployment of the main canopy needs to be attached to main parafoil in order to reduce the speed until the parafoil's fully deployment.

Another issue is the deployment of the parafoil at the desired altitude. To resolve this, an auto cut-down system which is operated by Ardupilot and GPS will be utilized. This system is based on the previous cut-down system by HAM radio signal from ground. This is a reliable system, but the operator has to watch the tracking program and send radio signals from the ground when the RPV reaches the desired altitude. The new auto cut-down system works through radio controlled relay (PIcoSwitch), Ardupilot by GPS altitude data, and hot-wire connections instead of the tracking program and radio transmitter.



Auto Cut-down system

BControl Stage

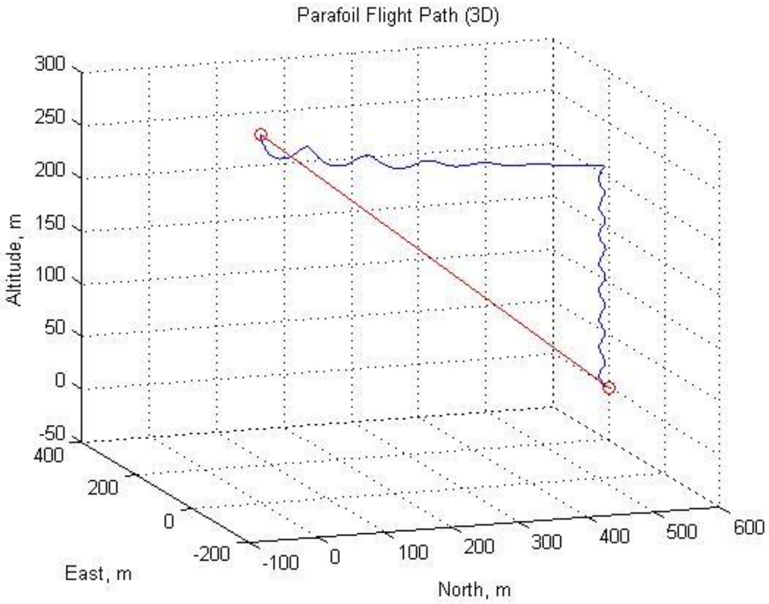
Control stage is the second stage, from the parafoil’s deployment altitude to 1000ft from the ground. As mentioned earlier, the deployment altitude is based on the conditions between wind speed and the parafoil’s maximum speed. In special cases where wind speed is not stable, higher wind speed in the middle of this range is not considered. The RPV cannot launch with a balloon due to highly uncertain flight movements, long flight distance, and the high possibility of losing the tracking signals. Therefore, this stage can be assumed to have lower wind speed than the vehicle. As a result, the wind speed is less than the maximum parafoil speed, in which case the vehicle is able to maneuver where we desired, although it cannot perform like no-wind conditions.

Two important concepts are applied to return the vehicle back to the desired point. They are Dynamic programming to find an optimized path during Control Stage in chapter 4 and Proportional-Integral-Derivative (PID) control system for the guidance and control of the parafoil in chapter 5.

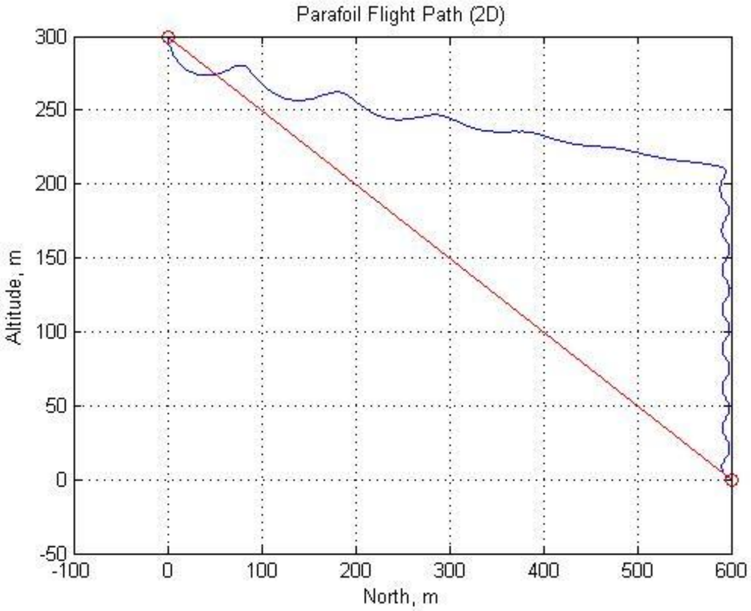
B.1Spiral Mode

In this stage, the most difficult issue is that the pitch of the parafoil cannot be controlled. In other words, the vehicle can move to the desired latitude and longitude position, but it is to be above the desired altitude or below that point.

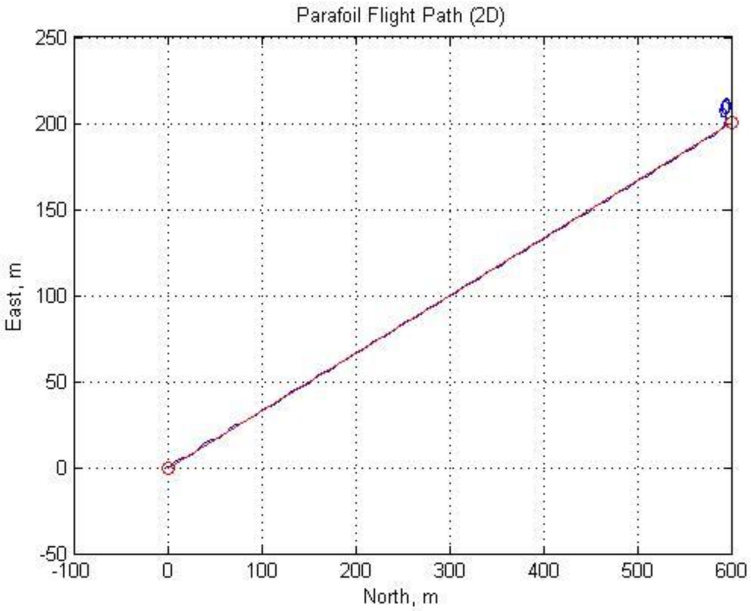
To solve this problem, the spiral turn which performs the turn with max roll control input is applied above the target:



Spiral Mode in 3D



Spiral Mode (Side view)



Spiral Mode (Top view)

When the vehicle is within the 50ft of radius at the target position of Latitude and Longitude, the autopilot sends the max roll input until the vehicle reaches the target altitude. In case the payload glides out of the boundaries, spiral mode is turned off, so that the vehicle can move within the boundaries, if the vehicle does not rise above the desired altitude.

This method can serve the main purposes of the vehicle, and it is also able to survey the landing or to take wind measurements in the future.

C. Final Approach

The purpose of this stage is to guide the parafoil and payload to safely land on the ground. This stage will start from 1000ft above the target destination. Basically, the parafoil will perform spiral turn with maximum rolling input. This stage will reduce the errors which might cause the vehicle to not reach the final way point, and also avoids the obstacles of the ground, such as tall trees and power poles.

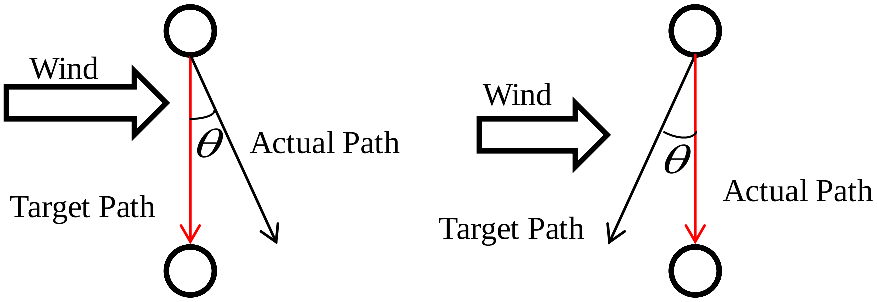
In order to land safely with less impact from the ground, the best way for the vehicle is to fly to head the wind direction, but it requires more complicated control programming. Thus, this will be considered later after the complete spiral turn landing.

APPENDIX C

Optimized Path Finding

A.Equations for Path Calculation for Dynamic Programming

The effect of wind upon the parafoil is similar to the affect of a river current upon a motorboat. If a motorboat were to head straight across the river, it would not reach the shore directly across from its starting point. The river current influences the motion of the boat and carries it downstream like in Figure. The motorboat may be moving with a velocity of 10 mph directly across the river; yet the resultant velocity of the boat will be greater than current velocity and at an angle in the downstream direction.

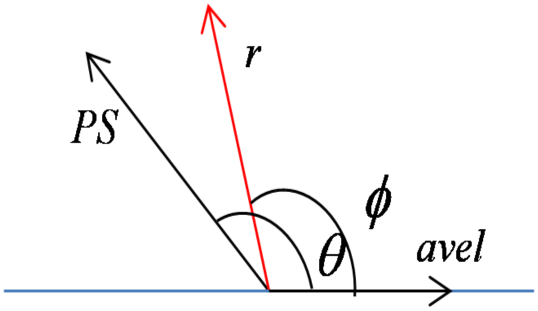


Path of Parafoil with wind a) Directly across the wind, b) Corrected Path

Our goal is to reach the desired point at each stage, so the heading of the boat should be corrected to the angle of θ to get to the point like in Figure. To obtain θ, the equations provided in following equations are under the assumptions of the wind vector being perpendicular to the target path and the parafoil keeping the same speed during the travel:

$$x = avel - PS \times \cos(\theta)$$

$$y = PS \times \sin(\theta)$$



Vector diagram of real velocity and parafoil velocity

From Figure and Equations, x is a vector of the x axis, and y is a vector of the y axis, where $avel$ is the average wind speed between point-to-point and PS is parafoil speed. These equations should be similar to the following equations:

$$x = avel - PS \times \cos(\theta) = r \times \cos(\phi)$$

$$y = PS \times \sin(\theta) = r \times \sin(\phi)$$

where r is the vector of the resultant path, and ϕ is the resultant heading angle.

From the above equations, we can obtain the following:

$$\frac{\sin(\phi)}{\cos(\phi)} = \frac{PS \times \sin(\phi)}{avel - PS \times \cos(\theta)}$$

$$avel = \frac{PS}{\sin(\phi)}(\sin(\theta)\cos(\phi) + \cos(\theta)\sin(\phi))$$

$$\frac{avel}{PS} = \frac{\sin(\theta + \phi)}{\sin(\phi)}$$

$$\theta = \sin^{-1}\left(\frac{avel \times \sin(\phi)}{BS}\right) + \phi$$

Now we find the equation for θ, and ϕ can be calculated from the known stage and stage conditions like the following:

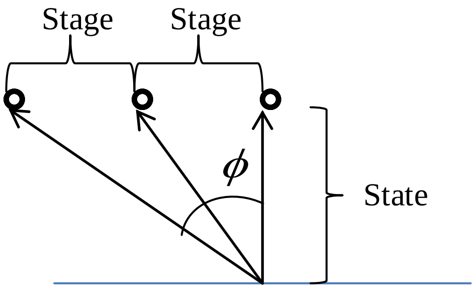


Diagram for the angle of target

$$\phi = \tan^{-1}\left(\frac{stage}{N \times state}\right)$$

$$r = \frac{PS \times \sin(\theta)}{\sin(\phi)}$$

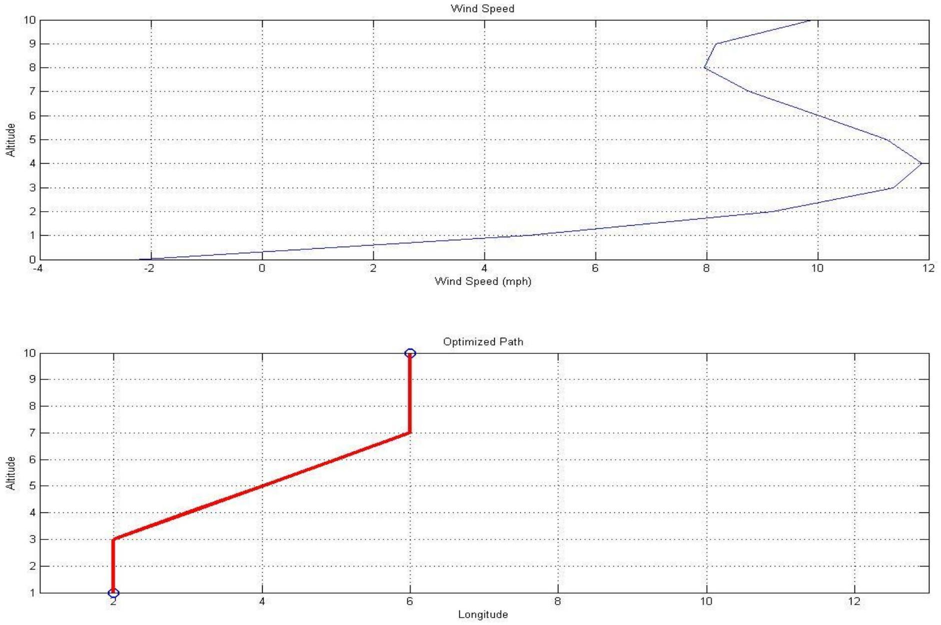
The objective of optimization is to minimize the time from the initial state to the final state; therefore, we are able to calculate the travel time of state-to-state from the given θ, ϕ, BS , r and the distance of the stage .

B.Simulated Optimal Path

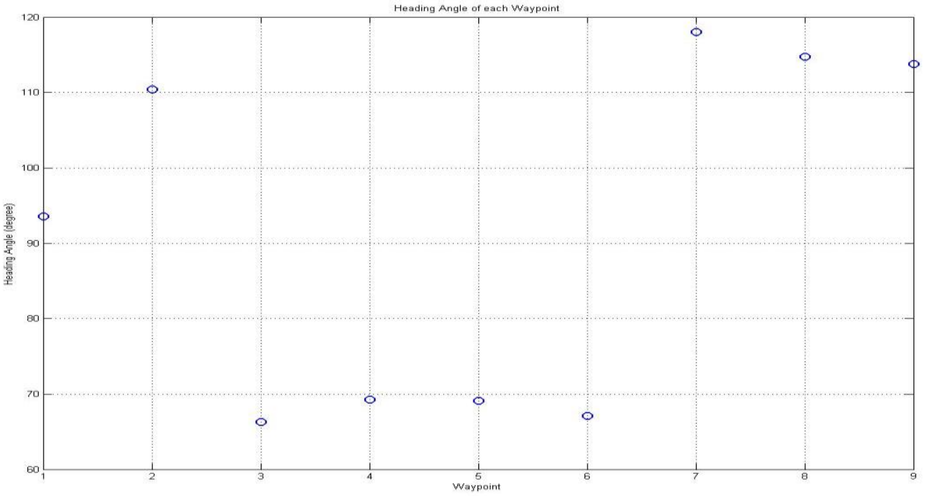
With equations from the previous sections, the simulation was performed in 2D and 3D environments in Matlab program. In this simulation, the second order polynomial function of Curved-fitting is used for the wind data from the previous sections. This, however, does not provide vertical wind data (i.e. wind of Z axis), so the estimated latitude wind data was used in 2D simulation, and the estimated latitude and longitude wind data were used in 3D simulation. Table # shows the other setup parameters for the 2D simulation.

Simulation parameters

Parameter	Value
# of Stage	10
Distance between Stages	100ft
# of State	12
Distance between States	10ft
Parafoil Gliding Speed	15mph
Initial Point	(2,1)
Final Point	(6,10)



Simulated optimized path with wind data in 2D



Heading angle of parafoil to reach the each waypoint

C.Proposed Strategy to Apply to the Autopilot

To apply dynamic programming method to the Autopilot, two ways were considered. They are uploading the waypoints from the ground before the flight and calculating the waypoints from the Autopilot during the flight. The first way is effective in reducing the working load of the Autopilot, and simplifies calculating the optimal path from a computer and building Autopilot program; however, the Autopilot cannot react to inaccurate wind data provided by the Air Resources Laboratory (ARL). The second case is exactly opposite to the first one. It is so complicated to design a program for Autopilot, and there is the possibility of making the Autopilot stop due to heavy working load, but the second is flexible to changing wind situations.

APPENDIX D

Aerodynamic Parameter

A. XFLR5

```
                         XFLR5 v6.06
                    Wing name :        Ozon1m
  Wing polar name :  T1-15.0 mph-VLM1-x0.000in-z-20.000in-Inviscid
                   Freestreamspeed :  15.000 mph


     alpha      CL         ICdPCdTCd        CY        Cm        RmYmIYmQInf        XCP
__________ _________  _________ __________ __________ __________ __________ __________
                        __________ __________ __________ __________
   -20.000  -0.941087   0.108826  0.000000  0.108826  0.000000  0.019237   0.000000
                0.000000  0.000000    6.7056     0.1875
   -19.500  -0.921870   0.103917  0.000000  0.103917  0.000000  0.026012  -0.000000
                0.000000  0.000000    6.7056     0.1867
   -19.000  -0.902322   0.099083  0.000000  0.099083  0.000000  0.032421  -0.000000
                0.000000  0.000000    6.7056     0.1860
   -18.500  -0.882447   0.094326  0.000000  0.094326  0.000000  0.038462  -0.000000
                0.000000  0.000000    6.7056     0.1853
   -18.000  -0.862253   0.089651  0.000000  0.089651  0.000000  0.044133   0.000000
                0.000000  0.000000    6.7056     0.1846
   -17.500  -0.841745   0.085061  0.000000  0.085061  0.000000  0.049434  -0.000000
                0.000000  0.000000    6.7056     0.1839
  -17.000  -0.820929   0.080559  0.000000  0.080559  0.000000  0.054361   0.000000  -
                0.000000  -0.000000    6.7056     0.1833
   -16.500  -0.799813   0.076149  0.000000  0.076149  0.000000  0.058914   0.000000
                0.000000  0.000000    6.7056     0.1827
   -16.000  -0.778403   0.071835  0.000000  0.071835  0.000000  0.063091  -0.000000
                0.000000  0.000000    6.7056     0.1821
   -15.500  -0.756707   0.067620  0.000000  0.067620  0.000000  0.066890   0.000000
                0.000000  0.000000    6.7056     0.1815
  -15.000  -0.734730   0.063507  0.000000  0.063507  0.000000  0.070312   0.000000  -
                0.000000  -0.000000    6.7056     0.1809
   -14.500  -0.712482   0.059500  0.000000  0.059500  0.000000  0.073354  -0.000000
                0.000000  0.000000    6.7056     0.1804
   -14.000  -0.689968   0.055601  0.000000  0.055601  0.000000  0.076016   0.000000
                0.000000  0.000000    6.7056     0.1799
  -13.500  -0.667198   0.051813  0.000000  0.051813  0.000000  0.078297   0.000000  -
                0.000000  -0.000000    6.7056     0.1794
```

```
-13.000  -0.644179   0.048139   0.000000   0.048139   0.000000   0.080196  -0.000000
0.000000   0.000000    6.7056     0.1789
-12.500  -0.620918   0.044583   0.000000   0.044583   0.000000   0.081713  -0.000000
0.000000   0.000000    6.7056     0.1784
-12.000  -0.597425   0.041147   0.000000   0.041147   0.000000   0.082847   0.000000
0.000000   0.000000    6.7056     0.1780
-11.500  -0.573707   0.037833   0.000000   0.037833   0.000000   0.083598  -0.000000  -
0.000000  -0.000000    6.7056     0.1776
-11.000  -0.549772   0.034645   0.000000   0.034645   0.000000   0.083965  -0.000000  -
0.000000  -0.000000    6.7056     0.1772
-10.500  -0.525630   0.031584   0.000000   0.031584   0.000000   0.083949   0.000000
0.000000   0.000000    6.7056     0.1768
-10.000  -0.501289   0.028653   0.000000   0.028653   0.000000   0.083550  -0.000000  -
0.000000  -0.000000    6.7056     0.1764
-9.500  -0.476758   0.025854   0.000000   0.025854   0.000000   0.082767   0.000000   0.000000
0.000000    6.7056     0.1761
-9.000  -0.452046   0.023190   0.000000   0.023190   0.000000   0.081601  -0.000000
0.000000   0.000000    6.7056     0.1757
-8.500  -0.427162   0.020662   0.000000   0.020662   0.000000   0.080053   0.000000   0.000000
0.000000    6.7056     0.1754
-8.000  -0.402115   0.018272   0.000000   0.018272   0.000000   0.078122   0.000000  -
0.000000  -0.000000    6.7056     0.1751
-7.500  -0.376915   0.016023   0.000000   0.016023   0.000000   0.075809   0.000000   0.000000
0.000000    6.7056     0.1749
-7.000  -0.351570   0.013915   0.000000   0.013915   0.000000   0.073115  -0.000000  -
0.000000  -0.000000    6.7056     0.1746
-6.500  -0.326091   0.011951   0.000000   0.011951   0.000000   0.070041  -0.000000
0.000000   0.000000    6.7056     0.1744
-6.000  -0.300486   0.010132   0.000000   0.010132   0.000000   0.066588   0.000000  -
0.000000  -0.000000    6.7056     0.1741
-5.500  -0.274766   0.008460   0.000000   0.008460   0.000000   0.062757   0.000000  -
0.000000  -0.000000    6.7056     0.1739
-5.000  -0.248940   0.006935   0.000000   0.006935   0.000000   0.058549  -0.000000  -
0.000000  -0.000000    6.7056     0.1738
-4.500  -0.223017   0.005560   0.000000   0.005560   0.000000   0.053965   0.000000  -
0.000000  -0.000000    6.7056     0.1736
-4.000  -0.197009   0.004334   0.000000   0.004334   0.000000   0.049006   0.000000   0.000000
0.000000    6.7056     0.1735
-3.500  -0.170924   0.003259   0.000000   0.003259   0.000000   0.043675  -0.000000
0.000000   0.000000    6.7056     0.1733
-3.000  -0.144773   0.002336   0.000000   0.002336   0.000000   0.037973   0.000000   0.000000
0.000000    6.7056     0.1732
-2.500  -0.118565   0.001566   0.000000   0.001566   0.000000   0.031901  -0.000000  -
0.000000  -0.000000    6.7056     0.1731
-2.000  -0.092310   0.000949   0.000000   0.000949   0.000000   0.025461  -0.000000  -
```

```
          0.000000  -0.000000    6.7056     0.1730
 -1.500  -0.066019   0.000485   0.000000   0.000485   0.000000   0.018656  -0.000000
          0.000000   0.000000    6.7056     0.1730
 -1.000  -0.039702   0.000175   0.000000   0.000175   0.000000   0.011488  -0.000000  -
          0.000000  -0.000000    6.7056     0.1729
 -0.500  -0.013369   0.000020   0.000000   0.000020   0.000000   0.003957  -0.000000
          0.000000   0.000000    6.7056     0.1729
0.000   0.012971   0.000019   0.000000   0.000019   0.000000  -0.003932  -0.000000   0.000000
          0.000000    6.7056     0.1730
 0.500   0.039307   0.000172   0.000000   0.000172   0.000000  -0.012178  -0.000000  -
          0.000000  -0.000000    6.7056     0.1730
1.000   0.065629   0.000479   0.000000   0.000479   0.000000  -0.020778  -0.000000   0.000000
          0.000000    6.7056     0.1730
1.500   0.091927   0.000941   0.000000   0.000941   0.000000  -0.029730  -0.000000   0.000000
          0.000000    6.7056     0.1730
2.000   0.118191   0.001556   0.000000   0.001556   0.000000  -0.039030   0.000000   0.000000
          0.000000    6.7056     0.1731
 2.500   0.144411   0.002325   0.000000   0.002325   0.000000  -0.048677  -0.000000  -
          0.000000  -0.000000    6.7056     0.1732
 3.000   0.170577   0.003246   0.000000   0.003246   0.000000  -0.058667  -0.000000  -
          0.000000  -0.000000    6.7056     0.1733
3.500   0.196678   0.004319   0.000000   0.004319   0.000000  -0.068996   0.000000   0.000000
          0.000000    6.7056     0.1734
4.000   0.222706   0.005543   0.000000   0.005543   0.000000  -0.079663   0.000000   0.000000
          0.000000    6.7056     0.1735
4.500   0.248650   0.006918   0.000000   0.006918   0.000000  -0.090663   0.000000  -0.000000
         -0.000000    6.7056     0.1737
5.000   0.274499   0.008442   0.000000   0.008442   0.000000  -0.101994  -0.000000   0.000000
          0.000000    6.7056     0.1739
5.500   0.300246   0.010113   0.000000   0.010113   0.000000  -0.113651   0.000000   0.000000
          0.000000    6.7056     0.1741
6.000   0.325879   0.011932   0.000000   0.011932   0.000000  -0.125632   0.000000   0.000000
          0.000000    6.7056     0.1743
 6.500   0.351389   0.013896   0.000000   0.013896   0.000000  -0.137932  -0.000000  -
          0.000000  -0.000000    6.7056     0.1745
7.000   0.376766   0.016005   0.000000   0.016005   0.000000  -0.150548   0.000000  -0.000000
         -0.000000    6.7056     0.1747
7.500   0.402002   0.018255   0.000000   0.018255   0.000000  -0.163476   0.000000   0.000000
          0.000000    6.7056     0.1750
8.000   0.427086   0.020646   0.000000   0.020646   0.000000  -0.176713   0.000000   0.000000
          0.000000    6.7056     0.1753
8.500   0.452010   0.023176   0.000000   0.023176   0.000000  -0.190253   0.000000  -0.000000
         -0.000000    6.7056     0.1756
9.000   0.476763   0.025843   0.000000   0.025843   0.000000  -0.204093   0.000000  -0.000000
         -0.000000    6.7056     0.1759
```

```
9.500   0.501338   0.028645   0.000000   0.028645   0.000000  -0.218230   0.000000   0.000000
                  0.000000    6.7056     0.1763
   10.000   0.525724   0.031580   0.000000   0.031580   0.000000  -0.232657  -0.000000
                  0.000000  0.000000    6.7056     0.1766
   10.500   0.549914   0.034645   0.000000   0.034645   0.000000  -0.247372  -0.000000
                  0.000000  0.000000    6.7056     0.1770
  11.000   0.573898   0.037839   0.000000   0.037839   0.000000  -0.262369   0.000000  -
                  0.000000  -0.000000    6.7056     0.1774
  11.500   0.597668   0.041158   0.000000   0.041158   0.000000  -0.277644  -0.000000  -
                  0.000000  -0.000000    6.7056     0.1778
  12.000   0.621215   0.044601   0.000000   0.044601   0.000000  -0.293192   0.000000  -
                  0.000000  -0.000000    6.7056     0.1782
   12.500   0.644531   0.048164   0.000000   0.048164   0.000000  -0.309009   0.000000
                  0.000000  0.000000    6.7056     0.1787
   13.000   0.667608   0.051845   0.000000   0.051845   0.000000  -0.325090   0.000000
                  0.000000  0.000000    6.7056     0.1792
  13.500   0.690437   0.055642   0.000000   0.055642   0.000000  -0.341430  -0.000000  -
                  0.000000  -0.000000    6.7056     0.1796
  14.000   0.713011   0.059550   0.000000   0.059550   0.000000  -0.358023  -0.000000  -
                  0.000000  -0.000000    6.7056     0.1802
   14.500   0.735322   0.063568   0.000000   0.063568   0.000000  -0.374866  -0.000000
                  0.000000  0.000000    6.7056     0.1807
   15.000   0.757363   0.067692   0.000000   0.067692   0.000000  -0.391952  -0.000000
                  0.000000  0.000000    6.7056     0.1812
   15.500   0.779125   0.071919   0.000000   0.071919   0.000000  -0.409277  -0.000000
                  0.000000  0.000000    6.7056     0.1818
  16.000   0.800603   0.076245   0.000000   0.076245   0.000000  -0.426834  -0.000000  -
                  0.000000  -0.000000    6.7056     0.1824
   16.500   0.821788   0.080669   0.000000   0.080669   0.000000  -0.444620  -0.000000
                  0.000000  0.000000    6.7056     0.1830
   17.000   0.842673   0.085185   0.000000   0.085185   0.000000  -0.462629   0.000000
                  0.000000  0.000000    6.7056     0.1836
   17.500   0.863253   0.089790   0.000000   0.089790   0.000000  -0.480854  -0.000000
                  0.000000  0.000000    6.7056     0.1843
  18.000   0.883521   0.094482   0.000000   0.094482   0.000000  -0.499290  -0.000000  -
                  0.000000  -0.000000    6.7056     0.1850
  18.500   0.903470   0.099256   0.000000   0.099256   0.000000  -0.517933   0.000000  -
                  0.000000  -0.000000    6.7056     0.1857
  19.000   0.923094   0.104109   0.000000   0.104109   0.000000  -0.536776   0.000000  -
                  0.000000  -0.000000    6.7056     0.1864
   19.500   0.942387   0.109036   0.000000   0.109036   0.000000  -0.555813   0.000000
                  0.000000  0.000000    6.7056     0.1871
   20.000   0.961343   0.114035   0.000000   0.114035   0.000000  -0.575038   0.000000
                  0.000000  0.000000    6.7056     0.1879
```

B. AVL Editor

Alpha	Cxtot	Cmtot	Cztot	CLtot	CDtot	CLa	Cma
-15	0.134810	-0.014989	0.719308	-0.729690	0.055954	3.186775	-0.201306
-13	0.100156	-0.021647	0.610821	-0.617695	0.0039816	3.228004	-0.180021
-11	0.070358	-0.027546	0.500240	-0.504475	0.026385	3.257031	-0.157859
-9	0.045561	-0.032659	0.388106	-0.390456	0.015713	3.273717	-0.134929
-7	0.025886	-0.036959	0.274965	-0.276070	0.007817	3.278029	-0.111340
5	0.011428	-0.040425	0.161367	-0.161749	0.002679	3.270043	-0.087210
-3	0.002258	-0.043042	0.047866	-0.047919	0.000250	3.249940	-0.062654
-1	-0.001578	-0.044796	-0.064984	0.065002	0.000444	3.218004	-0.037794
1	-0.000064	-0.045678	-0.176635	0.176607	0.003147	3.174622	0.012749
3	0.006795	-0.045685	-0.286542	0.286505	0.008211	3.120274	0.012358
5	0.018964	-0.044816	-0.394169	0.394322	0.015463	3.055534	0.037405
7	0.036384	-0.043076	-0.498993	0.499708	0.024699	2.981059	0.062269
9	0.058971	-0.040473	-0.600503	0.602335	0.035694	2.897586	0.086831
11	0.086614	-0.037019	-0.698203	0.701902	0.048201	2.805920	0.110969
13	0.119179	-0.032732	-0.791619	0.798139	0.061951	2.706926	0.134566
15	0.156507	-0.027632	-0.880295	0.890807	0.076662	2.601524	0.157508

APPENDIX E

Matlab/Simulink Code

A. Matlab

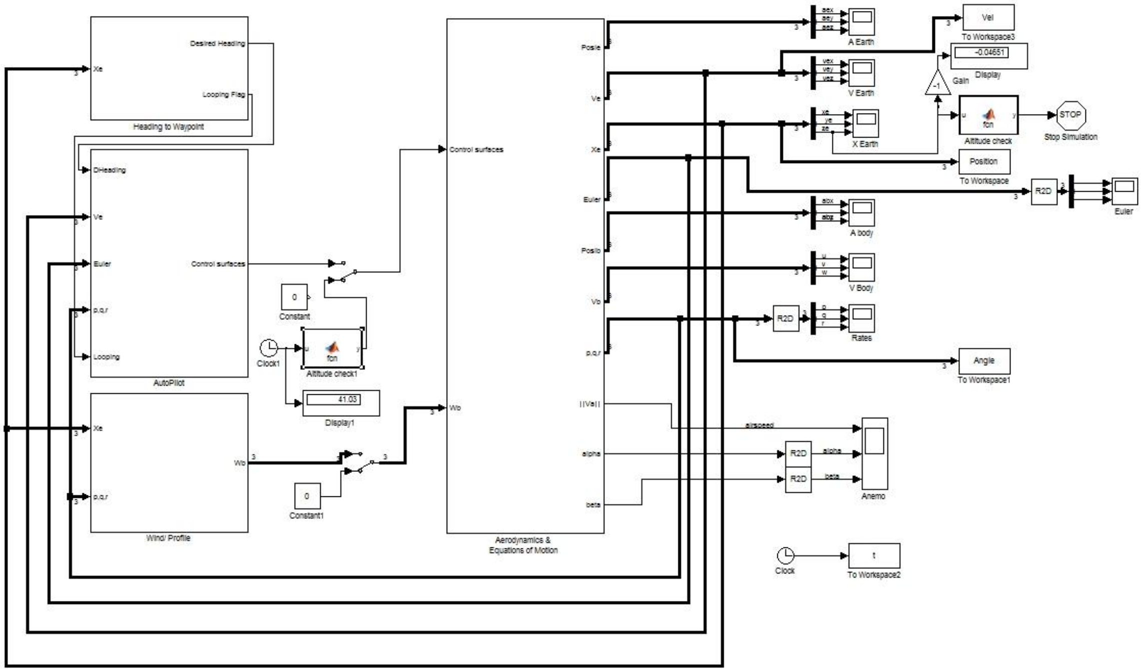
```
clearall;
clc;

% Parafoil Data
% ==================== %

Vmax=  5. ;       % m/s     max speed (11mph)

% Geometric data
Sref=  0.79 ;          % reference area (m^2)
Cref=  0.6350 ;        % reference chord (m)
Bref=  1.5240 ;        % reference span  (m)

% Default location about which moments and rotation rates are defined
Xref=  0 ;     % m
Yref=  0.22 ;  % m
Zref=  -1.18 ; % m

Mass =  5 ;     % kg

Ixx  =  0.423 ;   Ixy  =  0. ;     Ixz  =  0.0298 ; % kg.m2
Iyx  =  Ixy ;     Iyy  =  0.401 ;  Iyz  =  0. ;
Izx  =Ixz ;     Izy  =  Iyz ;    Izz  =  0.0529 ;

alpha = [-180.     -90.        -5.         0.         5.        10.
15.        90.       180.       ] ;

CZtot = [ 0.        10.         0.102580  -0.382560  -0.864950  -
1.339350  -1.876470 -10.         0.        ] ;
CXtot = [ 0.03       0.        -0.060470  -0.054540   0.003010
0.075230  -0.109600   0.         0.03      ] ;
Cmtot = [ 0.         2.         0.097340   0.036740  -0.047230  -
0.137480  -0.110850  -2.         0.        ] ;
CZq   = [ 0.         0.        -9.116812  -8.745563  -8.333030  -
7.894211  -7.541241   0.         0.        ] ;
CXq   = [ 0.         0.        -0.507980  -0.147635   0.201468
0.570160   1.026440   0.         0.        ] ;
Cmq   = [-12.3     -12.3      -12.3      -12.3      -12.3      -12.3
-12.3      -12.3      -12.3      ] ;
```

```
CYv   = [ 0.         0.        -0.204297  -0.201261  -0.208921  -
0.263082  -0.644075   0.         0.        ] ;
CYp   = [ 0.         0.        -0.217328  -0.145848  -0.074906  -
0.009213   0.015563   0.         0.        ] ;
CYr   = [ 0.         0.         0.049136   0.076784   0.103757
0.129586   0.152006   0.         0.        ] ;
CYrud = [-0.0016    -0.0016    -0.0016    -0.0016    -0.0016    -0.0016
-0.0016    -0.0016    -0.0016    ] ;
Clv   = [ 0.         0.        -0.098934  -0.137640  -0.176627  -
0.217863  -0.281340   0.         0.        ] ;
Clp   = [-0.5000    -0.5000    -0.5000    -0.5000    -0.5000    -0.5000
-0.5000    -0.5000    -0.5000    ] ;
Clr   = [ 0.         0.         0.064176   0.123690   0.182594
0.241567   0.304727   0.         0.        ] ;
Clrud = [-0.000045  -0.000045  -0.000045  -0.000045  -0.000045  -
0.000045  -0.000045  -0.000045  -0.000045 ] ;
Cnv   = [ 0.         0.         0.011072   0.000613  -0.009400  -
0.018783  -0.019168   0.         0.        ] ;
Cnp   = [ 0.         0.         0.027829  -0.031472  -0.087978  -
0.137944  -0.158838   0.         0.        ] ;
Cnr   = [ 0.         0.        -0.041685  -0.040010  -0.043374  -
0.056065  -0.120576   0.         0.        ] ;
Cnrud = [ 0.         0.         0.000628   0.000668   0.000698
0.000717   0.000725   0.         0.        ] ;

%Initial position/velocity data
uo = 30.48;       % Initial velocity x (m/s)
vo = 0;       % Initial velocity y (m/s)
wo = 0;       % Initial velocity z (m/s)

xo = 0;       % Initial position x (m)
yo = 0;       % Initial position y (m)
zo = 1000;    % Initial position z (m)

% Waypoint input & Desired point to related on altitude
homex = 2000;
homey = 6000;

    Loop = 15; % Looping radius
H_gain = 0.1; % Heading error gain

%     % Several Waypoints
%     wayx  =   [homex 1500 1500 1000 500 xo];  % Northerly waypoint,
%     wayy  =   [homey 4000 3000 2000 1000 yo]; % Easterly waypoint,
%     wayx    = 0.5*wayx;
%     wayy    = 0.5*wayy;
%     wayz  =   [0 zo/3 zo/2 2*zo/3 5*zo/6 zo]; % waypoint Height,

% One Waypoints
```

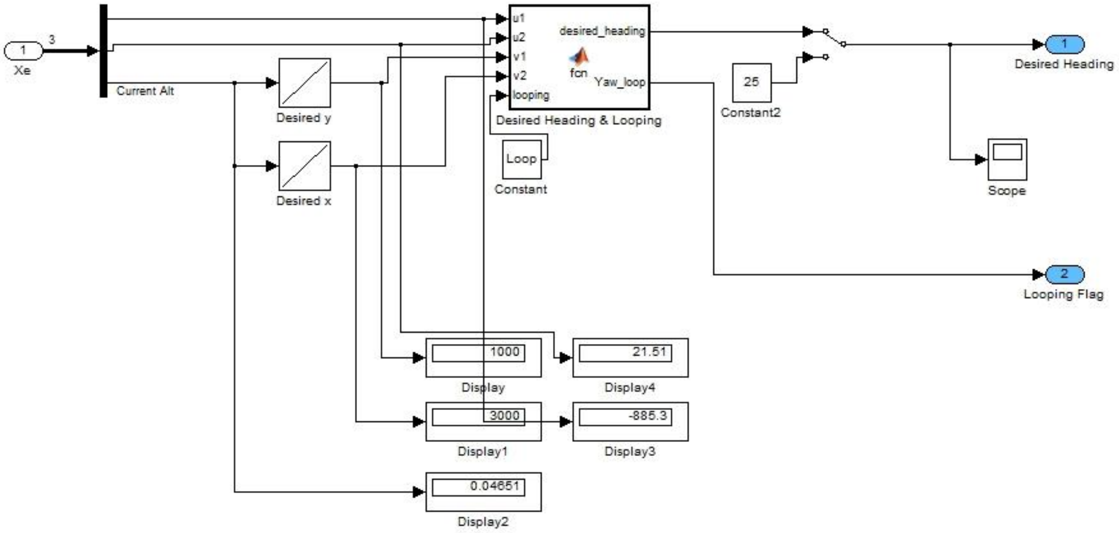
```
wayx    =   [homex xo]; % Northerly waypoint,
wayy    =   [homey yo]; % Easterly waypoint,
wayx    = 0.5*wayx;
wayy    = 0.5*wayy;
wayz    =   [0 zo]; % waypoint Height,

fwayx   =   fliplr(wayx);
fwayy   =   fliplr(wayy);
fwayz   =   -fliplr(wayz);

% Wind profile
windh   =   [0 100 200 500 1000 1500 2000 2500 3000];   % Height, m
windx   =   [0 2 4 4 5 5 6 5 5];    % Northerly wind, m/s
windy   =   [0 1 1 4 4 4 6 5 5];    % Easterly wind, m/s

windx   = 0.9*windx;
windy   = 0.9*windy;
fwindh  =   -fliplr(windh);
fwindx  =fliplr(windx);
fwindy  =fliplr(windy);

% Run the Flight simulation program Para_sim
sim('Para_sim');

% 3D position Plot
figure;
plot3(Position(:,2),Position(:,1),-Position(:,3))
gridon
holdon
% plot3(wayx,wayy,wayz,'ro-')
xlabel('East, m'), ylabel('North, m'), zlabel('Altitude, m')
title('Parafoil Flight Path (3D)')

figure;
plot(Position(:,2),Position(:,1))
gridon
holdon
% plot(wayx,wayy,'ro-')
xlabel('East, m'), ylabel('North, m')
title('Parafoil Flight Path (2D)')

figure;
plot(Position(:,1),-Position(:,3))
gridon
holdon
% plot(wayy,wayz,'ro-')
xlabel('North, m'), ylabel('Altitude, m')
title('Parafoil Flight Path (2D)')
```

```
figure;
plot(Position(:,2),-Position(:,3))
gridon
holdon
% plot(wayx,wayz,'ro-')
xlabel('East, m'), ylabel('Altitude, m')
title('Parafoil Flight Path (2D)')

% 2D position plot
figure
subplot(3,1,1)
plot(t,Position(:,1))
xlabel('Time, s'), ylabel('North, m'), grid
title('Earth-Relative Parafoil Location')

subplot(3,1,2)
plot(t,Position(:,2))
xlabel('Time, s'), ylabel('East, m'), grid
title('Earth-Relative Parafoil Location')

subplot(3,1,3)
plot((sqrt(Position(:,1).*Position(:,1) +
Position(:,2).*Position(:,1))),-Position(:,3))
xlabel('Range, m'), ylabel('Altitude, m'), grid
title('Earth-Relative Parafoil Location')

% Angle Plot (Roll,Pitch, Yaw)
figure
subplot(3,1,1)
plot(t,Angle(:,1))
xlabel('Time, s'), ylabel('Roll, deg'), grid
title('Earth-Relative Parafoil angle')

subplot(3,1,2)
plot(t,Angle(:,2))
xlabel('Time, s'), ylabel('Pitch, deg'), grid
title('Earth-Relative Parafoil angle')

subplot(3,1,3)
plot(t,Angle(:,3))
xlabel('Time, s'), ylabel('Yaw, deg'), grid
title('Earth-Relative Parafoil angle')
```

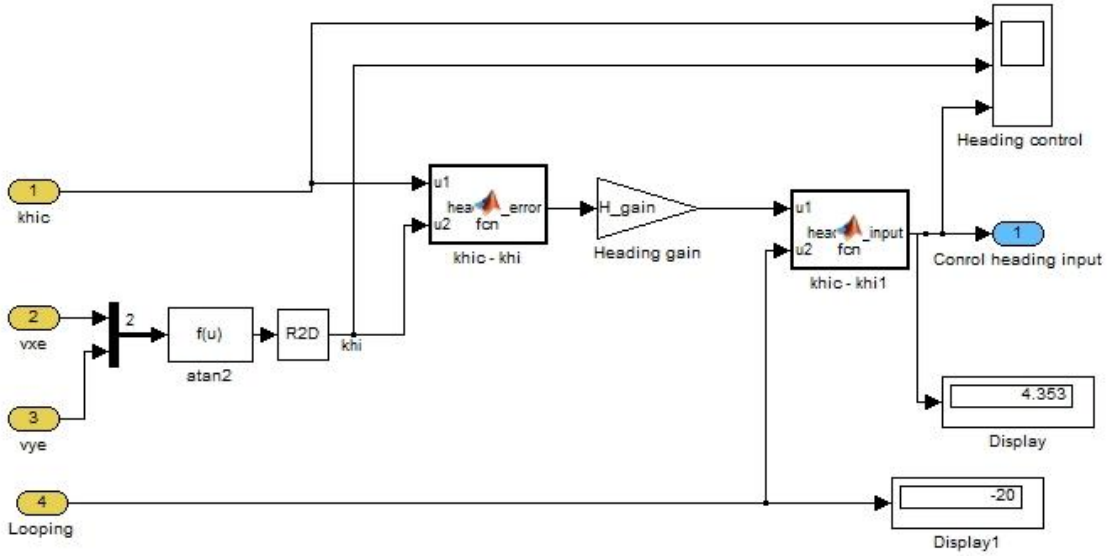
B. Simulink



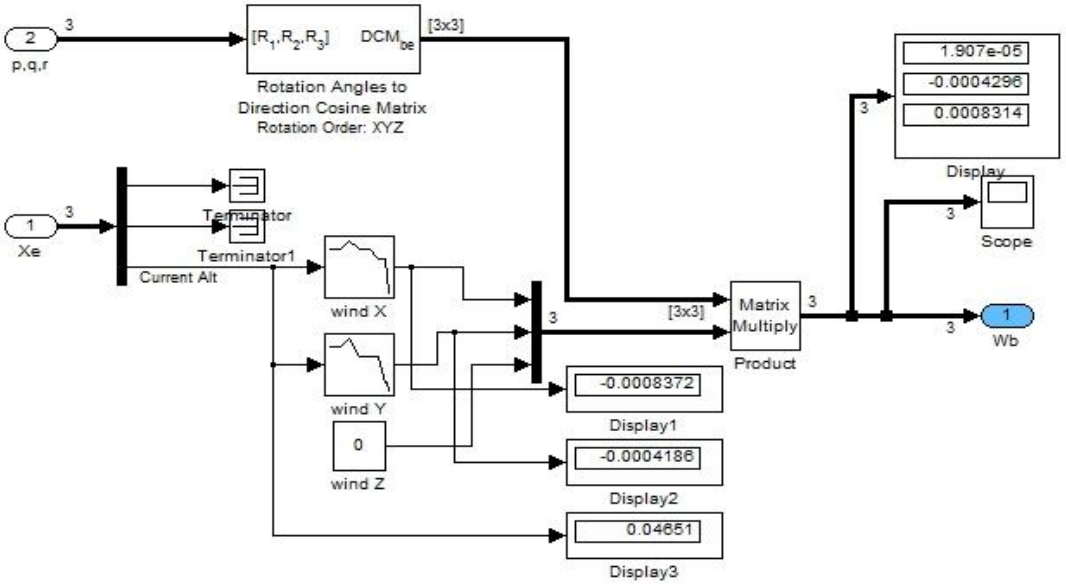
Autonomous Flight Simulation system



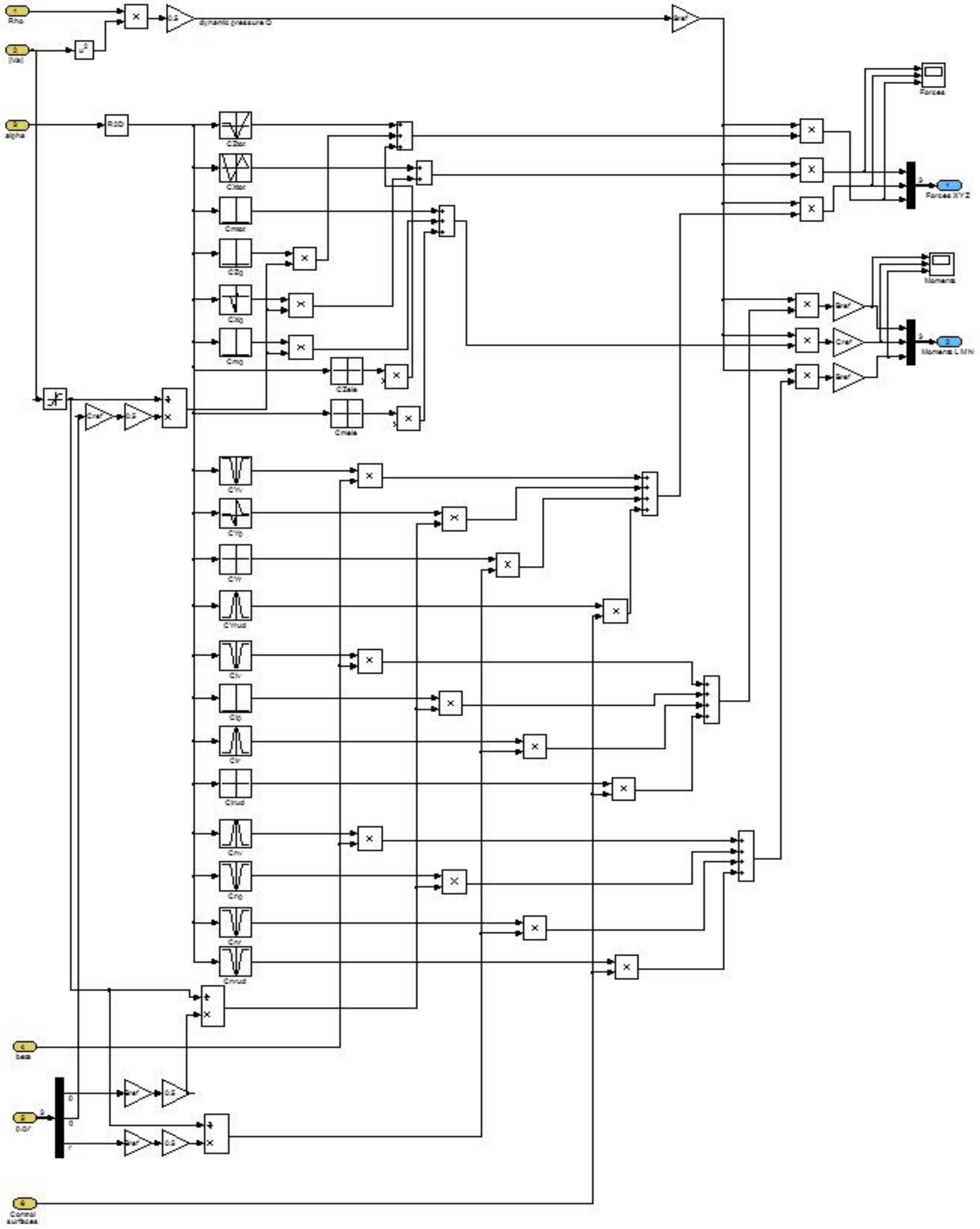
Waypoint calculation and Spiral turn Module



Heading Calculation Module



Wind Calculation module



Aerodynamic parameter module

APPENDIX F

Arduino Code

```
/************************************/
/*ArduPilot Header file*/
/************************************/
#define MIXING_MODE 0 //Servo mixing mode 0 = Normal, 1 = V-tail (v tail not tested yet).
#define REVERSE_ROLL 1 //To reverse servo roll, PUT -1 to invert it!!!
#define REVERSE_PITCH 1 //To reverse servo pitch, PUT -1 to invert it!!!
#define RADIO_SWITCH_ACTION 0 // 0: TX Switch centered = waypoint mode & full = RTL mode. 1:
#define GPS_PROTOCOL 1 // 0 = NMEA, 1=SIRF, 2=uBlox, Choose protocol
#define ATTITUDE_RATE_OUTPUT 250 //the output rate of attitude data in milliseconds. Useful
#define POSITION_RATE_OUTPUT 4  //This number will be multiplied by ATTITUDE_RATE_OUTPUT, the
#define REMEMBER_LAST_WAYPOINT_MODE 1 //If set 1 = will remember the last waypoint even if
#define INTPUT_VOLTAGE 5200.0 //voltage in millis your power regulator is feeding your
#define REVERSE_THROTTLE 0 // 0 = Normal mode. 1 = Reverse mode...
#define FLY_BY_WIRE_GAIN_ROLL .5 //Decrease the value to increase the response of the sticks.
#define FLY_BY_WIRE_GAIN_PITCH .5 //The same as roll.
#define FLY_BY_WIRE_SPEED_SETPOINT 20 //The airspeed you want to hold in fly by wire mode.
#define GPS_ERROR_SPEED_SETPOINT 3 // In -m/s; ,
#define REV_FLY_BY_WIRE_CH1 1 //-1 will invert it
#define REV_FLY_BY_WIRE_CH2 1 //-1 will invert it
#define SERVO_AILE_MAX 2400 //Range of Ailerons
#define SERVO_AILE_MIN 600
#define SERVO_ELEV_MAX 2400 //Range of Elevator
#define SERVO_ELEV_MIN 600
#define head_P 0.35 //Heading error proportional (same used to move the rudder
#define head_I 0 //heading error integrator.
#define head_D 0 //Derivative not used, but someday....
#define head_error_max 20 //35 The maximum output in degrees to control the roll setpoint
#define head_error_min -20 //-35 The min output in degrees to control the roll setpoint
#define roll_abs .2 //Set point absolute...(Not Used)
#define roll_P .35 //roll PID proportional
#define roll_I .35 //roll PID integrator
#define roll_min -25 //PID output limit in servo degrees
#define roll_max  25 //PID output limit in servo degrees
#define roll_Integrator_max 10 //Limit the integrator, to avoid overshoots
#define roll_Integrator_min -10
#define pitch_P .65 //Pitch Proportional
#define pitch_I .35 //Pitch integrator
#define pitch_min -25 //Pitch limits
#define pitch_max  25
#define pitch_Integrator_max 10 //Pitch integrator limits
```

```
#define pitch_Integrator_min -10
#define PITCH_COMP .30
#define throttle_max 1800 //Servo range In milliseconds.
#define throttle_min 1200 //
#define throttle_dead_zone 20 //In percent %
#define throttle_absolute 3 //Absolute
#define throttle_kp 3 //Proportional
#define throttle_ki 1 //Integrator
#define throttle_max 85 //Limits
#define throttle_Integrator_max 70 //Integrator limit.
#define ALTITUDE_ERROR_MAX 0 //
#define ALTITUDE_ERROR_MIN -10 //
#define ALTITUDE_ERROR_PITCH_PROPORTIONAL 1.5 //Altitude error proportional, pitch setpoint
#define ALTITUDE_ERROR_PITCH_MAX 0 //Limits, EasyStar climb by itself,
#define ALTITUDE_ERROR_PITCH_MIN -14
#define AIRSPEED_CENTRAL 22 //Airspeed  central point in m/s, normal flight...
#define ALTITUDE_ERROR_AIRSPEED_PROPORTIONAL 2
#define ALTITUDE_ERROR_AIRSPEED_MAX 10
#define ALTITUDE_ERROR_AIRSPEED_MIN -10
#define FAKE_BEARING 0 //If set to 1, will fake the bearing and will try to always head to
#define DESIRED_FAKE_BEARING 45 //Will try to go NorthEast,
#define FAKE_GPS_LOCK 0 //
#define PRINT_WAYPOINTS 1 //
#define TEST_THROTTLE 0 //
#define WALK_AROUND 0 //
#define CALIBRATE_SERVOS 0
#define TEST_SENSORS 0 //
```

VITA

Seong-Jin Lee

Candidate for the Degree of

Doctor of Philosophy

Thesis: DEVELOPMENT OF OPTIMIZED PATH PLANNING AND AUTONOMOUS CONTROL FOR RETURN-TO-POINT VEHICLE OF HIGH ALTITUDE BALLOONING

Major Field: Mechanical and Aerospace Engineering

Biographical:

Personal Data: Born in Gwangju, South Korea. Married Nada Yoon and reside with our children, Grace Haeun and Kate Sieun, in Stillwater, Oklahoma.

Education:

Completed the requirements for the Doctor of Philosophy in Mechanical & Aerospace Engineering at Oklahoma State University, Stillwater, Oklahoma in August, 2014.

Completed the requirements for the Master of Engineering in Mechanical & Aerospace Engineering at Cornell University, Ithaca, NY, 2007.

Completed the requirements for the Bachelor of Science in Mechanical & Aerospace Engineering at Korea Aerospace University, South Korea, 2002.

Experience:

Visiting Assistant Professor for Mechanical Engineering at Alfred University, NY, Appointed 2014

Assist with Aerospace Engineering Capstone Design Course, Spring 2009– Current

Assist with Introduction to Engineering (Aerospace) Course, Fall 2009 - Current