University of Colorado, Boulder
CU Scholar

Aerospace Engineering Sciences Graduate Theses & Dissertations

Aerospace Engineering Sciences

Spring 1-1-2013

Energy-Aware Path Planning for UAS Persistent Sampling and Surveillance

Wenceslao Eric Shaw-Cortez
University of Colorado Boulder, w.shawcortez@gmail.com

Follow this and additional works at: https://scholar.colorado.edu/asen_gradetds

 Part of the Propulsion and Power Commons

Recommended Citation
Shaw-Cortez, Wenceslao Eric, "Energy-Aware Path Planning for UAS Persistent Sampling and Surveillance" (2013). *Aerospace Engineering Sciences Graduate Theses & Dissertations*. 76.
https://scholar.colorado.edu/asen_gradetds/76

This Thesis is brought to you for free and open access by Aerospace Engineering Sciences at CU Scholar. It has been accepted for inclusion in Aerospace Engineering Sciences Graduate Theses & Dissertations by an authorized administrator of CU Scholar. For more information, please contact cuscholaradmin@colorado.edu.

www.manaraa.com

Energy-Aware Path Planning for UAS Persistent Sampling and Surveillance

by

Wenceslao Shaw-Cortez

B.S., University of Colorado Boulder, 2012

A thesis submitted to the

Faculty of the Graduate School of the

University of Colorado in partial fulfillment

of the requirements for the degree of

Master of Science

Department of Aerospace Engineering Sciences

2013

www.manaraa.com

This thesis entitled:
Energy-Aware Path Planning for UAS Persistent Sampling and Surveillance
written by Wenceslao Shaw-Cortez
has been approved for the Department of Aerospace Engineering Sciences

_____

Prof. Eric Frew

_____

Prof. Brian Argrow

Date _____

The final copy of this thesis has been examined by the signatories, and we find that both the content and the form meet acceptable presentation standards of scholarly work in the above mentioned discipline.

www.manaraa.com

Shaw-Cortez, Wenceslao (M.S., Aerospace Engineering)

Energy-Aware Path Planning for UAS Persistent Sampling and Surveillance

Thesis directed by Prof. Eric Frew

The focus of this work is to develop an energy-aware path planning algorithm that maximizes UAS endurance, while performing sampling and surveillance missions in a known, stationary wind environment. The energy-aware aspect is specifically tailored to extract energy from the wind to reduce thrust use, thereby increasing aircraft endurance. Wind energy extraction is performed by static soaring and dynamic soaring. Static soaring involves using upward wind currents to increase altitude and potential energy. Dynamic soaring involves taking advantage of wind gradients to exchange potential and kinetic energy. The path planning algorithm developed in this work uses optimization to combine these soaring trajectories with the overarching sampling and surveillance mission.

The path planning algorithm uses a simplified aircraft model to tractably optimize soaring trajectories. This aircraft model is presented and along with the derivation of the equations of motion. A nonlinear program is used to create the soaring trajectories based on a given optimization problem. This optimization problem is defined using a heuristic decision tree, which defines appropriate problems given a sampling and surveillance mission and a wind model.

Simulations are performed to assess the path planning algorithm. The results are used to identify properties of soaring trajectories as well as to determine what wind conditions support minimal thrust soaring. Additional results show how the path planning algorithm can be tuned between maximizing aircraft endurance and performing the sampling and surveillance mission. A means of trajectory stitching is demonstrated to show how the periodic soaring segments can be combined together to provide a full solution to an infinite/long horizon problem.

www.manaraa.com

Acknowledgements

I would like to first thank my thesis advisor, Professor Eric Frew, for helping me through my first experience in graduate research. Professor Frew offered ample time and guidance whenever I needed help during my research. I would also like to thank my defense committee and reading committee member Professor Brian Argrow for providing sound advice and a big picture perspective to my research. I'd like to also thank my final committee member Professor Dale Lawrance for asking the tough questions and ensuring that I fully understand my research problem.

www.manaraa.com

Contents

Chapter

www.manaraa.com

www.manaraa.com

www.manaraa.com

Tables

Table

www.manaraa.com

www.manaraa.com

Figures

Figure

www.manaraa.com

www.manaraa.com

www.manaraa.com

Chapter 1

Introduction

1.1 Motivation

The use of Unmanned Aircraft Systems (UAS) has dramatically increased in recent years. UAS are capable of performing missions considered too dangerous for manned flight. Such missions include entering tornadic storm cells for collecting data on tornadogenesis [8] and surveying enemy territory for military operations [7]. Additionally, they can provide communication relays in remote locations [15] and work cooperatively for increased mission performance [12]. However UAS applications are limited to the aircraft's flight time. In order to increase UAS effectiveness, the aircraft must have longer flight endurance.

One way to increase flight endurance is to extract energy from the wind by soaring. Two main types of soaring include static soaring and dynamic soaring. During static soaring the UAS gains potential energy by flying in upward wind currents (Fig. 1.1). These upward wind currents can be a result of uneven ground heating, such as with thermal soaring [3], or from terrestrial formations that direct winds upward, such as ridge soaring. In either case, the vertical wind surpasses the aircraft sink rate and the UAS gains potential energy.

www.manaraa.com

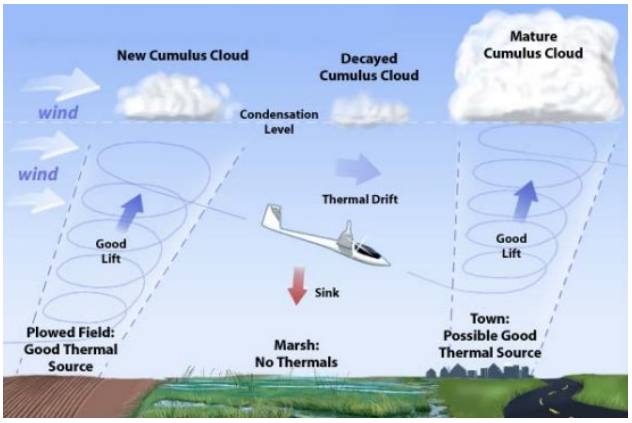


Figure 1.1: Thermal Soaring [2]

Dynamic soaring exploits wind gradients by using flight maneuvers to gain potential and/or kinetic energy. An albatross uses wind gradients above the ocean surface to glide for thousands of miles with little effort [6]. An example of a dynamic soaring trajectory is shown in Fig. 1.2. Upon reaching point 1, the albatross is flying into the wind with a positive pitch angle. It then rolls, exposing the wing surface area to the wind, and turns away from the wind at a negative pitch angle. At point 2, the albatross turns back into the wind to repeat the trajectory.

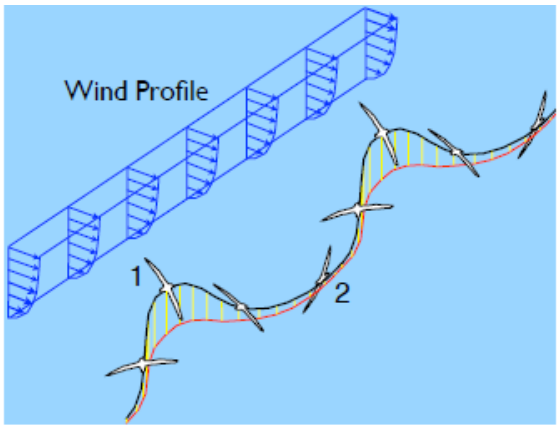


Figure 1.2: Dynamic Soaring Trajectory in boundary layer wind profile [14]

www.manaraa.com

Related works have looked into using optimization to develop soaring trajectories [5] [16]. Those works analyze different problem formulations and cost functions for specifically developing dynamic soaring trajectories. Periodic constraints are used as a means to solve for these cyclic dynamic soaring trajectories. However, the main focus of both works is to increase aircraft endurance. The UAS is not required to perform an additional mission. Also, both works focus specifically on developing soaring trajectories in a boundary layer wind environment.

Other works use reactive controllers for soaring [4] [11]. Those works look into heuristic approaches to develop online soaring trajectories. For static soaring, a reactive controller is used to measure upward wind currents. The aircraft proceeds to orbit these currents in order to gain potential energy [4]. Similar work identifies and exploits characteristics of dynamic soaring when flying in a wind gradient [11]. That work also looks into measuring wind gradients online.

Yet another work combines dynamic soaring trajectories with a surveillance mission [9]. That work uses Dubin's paths to define straight and curved trajectory types that can perform dynamic soaring. The properties of these dynamic soaring Dubin's paths are exploited to perform a surveillance mission while soaring. The Dubin's path segments are shown to splice together to develop a long term trajectory solution.

This work focuses on a broader mix of those related works. Here wind energy extraction applies to all soaring types. The related works are tailored specifically to either dynamic or static soaring, but not soaring in general. Additionally, most of the related works focus only on soaring and not performing an additional mission. This work looks to combine soaring with a sampling and surveillance mission. In order to accomplish this, the optimization approach is used in addition to periodic soaring to develop segments of the sampling and surveillance mission. Similar to the Dubin's path work, these segments can be spliced together for a full trajectory solution.

www.manaraa.com

1.2 Current Work

The goal of this work is to develop a path planning algorithm that creates UAS persistent sampling and surveillance trajectories that maximize aircraft flight endurance by extracting energy from a known, stationary wind field. This path planning algorithm takes into account both static and dynamic soaring as well as powered and gliding UAS.

One of the difficulties in maximizing flight endurance, or equivalently flight time, is that the desired soaring trajectory may allow the UAS to stay aloft for a large amount of time. Solving these potentially long horizon or infinite horizon problems is intractable. The proposed solution divides the long horizon problem into soaring segments confined within a certain time limit. These soaring segments are developed using optimization subject to periodic and sampling/surveillance mission constraints to minimize thrust use. The periodic constraints allow the soaring segments to be stitched together to then provide the solution to the long horizon problem for a known, stationary wind field.

In this work persistent sampling and surveillance is defined as staying inside a desired sampling region for as long as possible. The sampling/surveillance mission can be divided into guidance and loiter missions. The loiter mission is identical to the original sampling and surveillance mission of staying inside the sampling region. The guidance mission aims to reduce the thrust use as the UAS flies to the sampling region to further improve the performance in the loiter mission.

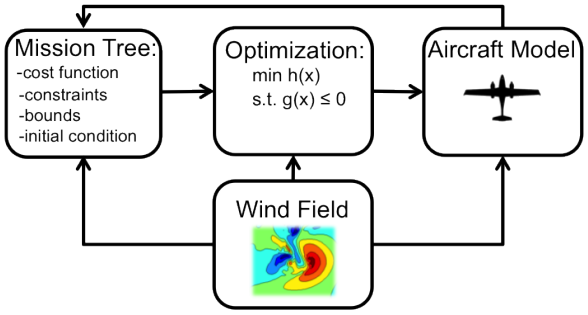


Figure 1.3: Path Planning Algorithm Block Diagram

Figure 1.3 shows the path planning algorithm presented in this work. The trajectory design

www.manaraa.com

portion is represented by the Mission Tree and Optimization blocks. As mentioned previously, the proposed solution splits the long horizon trajectory into soaring segments. Each segment is formulated as an optimization problem in the Mission Tree block. The chosen problem from the mission tree is then sent to the Optimization block. Here an optimizer is used to develop a soaring trajectory subject to the costs and constraints defined from the mission tree.

1.3 Thesis Outline

The system model and equations of motion are derived in Chapter 2. Chapter 3 defines the optimization aspect of the proposed solution for developing soaring trajectories. Chapter 4 defines the decision tree used to formulate the optimization problems. Chapter 5 presents and discusses simulations used to analyze the performance of the path planning algorithm. Finally, Chapter 6 provides concluding remarks and future work.

1.4 Thesis Contributions

The main contributions presented in this work include:

(1) Formulation of optimization problem to develop soaring trajectories

  (a) Formulation of optimization problem as nonlinear program

  (b) Derivation of an initial trajectory development approach using differential flatness

(2) Formulation of the mission tree to appropriately solve various optimization problems

  (a) Formulation of guidance and loiter mission problems subject to soaring constraints

  (b) Design of penalty function to increase versatility in loiter mission

  (c) Design of myopic stitching for creating long term trajectory solutions

(3) Analysis and assessment of path planning algorithm

  (a) Analysis of soaring trajectories in a linear boundary layer wind model

www.manaraa.com

(b) Analysis of wind gradient strength on the ability to soar

(c) Assessment of penalty function to tune between mission objectives

(d) Analysis of myopic stitching for providing long term trajectory solutions

(e) Identification of design applications for the presented path planning algorithm

www.manaraa.com

Chapter 2

System Model

This chapter defines the aircraft model used in the path planning algorithm. The equations of motion are derived for the aircraft model, and the sampling and surveillance mission is outlined.

2.1 Aircraft Model

The UAS is modeled by a three-dimensional point-mass aircraft [5]. The aerodynamic forces acting on the aircraft include the lift, L, drag, D, and thrust, T. The directions of these forces are defined by the orientation angles, which are the air-relative course angle, χ_a, air-relative flight path angle, γ_a, and the roll angle, ϕ. The aircraft model also takes into account the wind velocity vector, $\mathbf{w}$. The air-relative velocity vector, $\dot{\mathbf{p}}_a$, is defined as the velocity of the UAS with respect to this wind vector. Figure 2.1 shows a diagram of the forces, orientations, and velocity vectors.

The inertial coordinate frame for this model is represented by the x, y, z axes. This is a conventional aircraft coordinate frame with the z axis pointing down and the x and y axes abiding by the right hand rule as shown in Fig. 2.1. This coordinate frame is used to define the inertial position, $\mathbf{p} = \begin{pmatrix} x & y & z \end{pmatrix}^{\mathrm{T}}$. The derivative of $\mathbf{p}$ is the inertial velocity vector, $\dot{\mathbf{p}} = \begin{pmatrix} \dot{x} & \dot{y} & \dot{z} \end{pmatrix}^{\mathrm{T}}$. The inertial velocity is the summation of the air-relative velocity and wind velocity, where the wind is assumed known and stationary in time:

$$\mathbf{w} = \begin{pmatrix} w_x & w_y & w_z \end{pmatrix}^{\mathrm{T}} \tag{2.1}$$

where w_x, w_y, and w_z are the wind components in the x, y, and z directions, respectively. The

www.manaraa.com

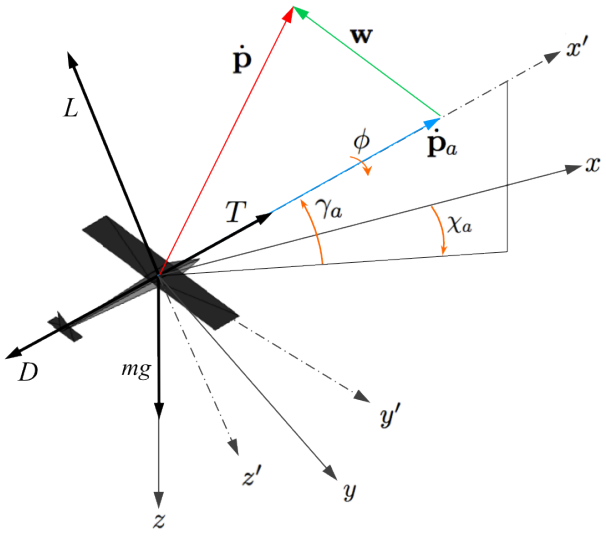


Figure 2.1: UAS Diagram

wind is a function of the aircraft position. The associated Jacobian of the wind, J_w, is defined by:

$$J_w = \begin{bmatrix} \frac{\partial w_x}{\partial x} & \frac{\partial w_x}{\partial y} & \frac{\partial w_x}{\partial z} \\ \frac{\partial w_y}{\partial x} & \frac{\partial w_y}{\partial y} & \frac{\partial w_y}{\partial z} \\ \frac{\partial w_z}{\partial x} & \frac{\partial w_z}{\partial y} & \frac{\partial w_z}{\partial z} \end{bmatrix} \tag{2.2}$$

The aircraft wind frame is fixed to the point mass model of the aircraft and is defined by the x', y', z' axes. The orientation angles define the rotation between the wind frame and inertial coordinate frame. The air-relative course angle, χ_a, rotates from the x-axis to the projection of the wind x'-axis on the $x - y$ plane. The air-relative flight path angle, γ_a, then rotates up to the wind x'-axis. The roll angle, ϕ, rotates about x'. This model assumes the aircraft thrust is always directed into the direction of the air-relative velocity vector.

The rotation matrix, $R_{i/w}$, is used to rotate from the wind frame to the inertial frame. This rotation matrix is composed of rotations about the orientation angles:

$$R_{i/w} = R_z(\chi_a) R_y(\gamma_a) R_x(\phi) \tag{2.3}$$

www.manaraa.com

$$R_z(\chi_a) = \begin{bmatrix} \cos\chi_a & -\sin\chi_a & 0 \\ \sin\chi_a & \cos\chi_a & 0 \\ 0 & 0 & 1 \end{bmatrix}, R_y(\gamma_a) = \begin{bmatrix} \cos\gamma_a & 0 & \sin\gamma_a \\ 0 & 1 & 0 \\ -\sin\gamma_a & 0 & \cos\gamma_a \end{bmatrix}, R_x(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi & \cos\phi \end{bmatrix} \tag{2.4}$$

The lift, drag, and thrust forces are oriented based on the air-relative velocity vector. The lift acts perpendicular to $\dot{\mathbf{p}}_a$ along the negative z'-axis. The lift direction about $\dot{\mathbf{p}}_a$ is dictated by the roll angle. The thrust is assumed to act along the direction of $\dot{\mathbf{p}}_a$, and the drag acts in opposition to $\dot{\mathbf{p}}_a$. The gravity acts on the aircraft mass, m, with acceleration, g, along the z-axis. The lift and drag equations are defined by:

$$L = \frac{1}{2}\rho V_a^2 S C_L \tag{2.5}$$

$$D = \frac{1}{2}\rho V_a^2 S (C_{d_0} + \frac{C_L^2}{\pi AR\mathrm{e}}) \tag{2.6}$$

where ρ is the air density, V_a is the airspeed or the magnitude of the air-relative velocity ($\|\dot{\mathbf{p}}_a\|_2$), S is the wing surface area, AR is the aspect ratio, e is the Oswald efficiency factor, and C_{d_0} is the zero-lift drag coefficient.

For this work, the states, $\mathbf{x}(t)$, and control inputs,$\mathbf{u}(t)$, of the model are defined by:

$$\mathbf{x}(t) = \begin{pmatrix} x & y & z & V_a & \gamma_a & \chi_a & \phi & C_L \end{pmatrix}^{\mathbf{T}}$$

$$\mathbf{u}(t) = \begin{pmatrix} \dot{\phi} & \dot{C}_L & T \end{pmatrix}^{\mathbf{T}}$$

Note that the roll rate, $\dot{\phi}$, and lift coefficient rate, $\dot{C}_L$, are directly controlled.

www.manaraa.com

2.2 Equations of Motion

The equations of motion for the defined states and control inputs define the dynamics of the aircraft model. These dynamics are defined as:

$$\dot{\mathbf{x}}(t) = f(\mathbf{x}(t), \mathbf{u}(t), \mathbf{w}(\mathbf{p}(t))) \tag{2.7}$$

where the state derivatives, $\dot{\mathbf{x}}(t)$, are functions of the states, control inputs, and wind vector.

The function, f, holds the equations of motion, which are derived from aircraft kinematic and dynamic equations. As defined in Fig. 2.1, the inertial aircraft velocity is defined as the summation of the air-relative velocity and the wind velocity vectors with respect to the inertial coordinate frame.

$$\dot{\mathbf{p}} = \dot{\mathbf{p}}_a + \mathbf{w} = R_{i/w} V_a \mathbf{e}_1 + \mathbf{w} \tag{2.8}$$

where $\mathbf{e}_1 = \begin{pmatrix} 1 & 0 & 0 \end{pmatrix}^{\mathrm{T}}$. Similar notation is used throughout this derivation with unit vectors $\mathbf{e}_2 = \begin{pmatrix} 0 & 1 & 0 \end{pmatrix}^{\mathrm{T}}$ and $\mathbf{e}_3 = \begin{pmatrix} 0 & 0 & 1 \end{pmatrix}^{\mathrm{T}}$. In Eq. (2.8), $\dot{\mathbf{p}}_a$ is rewritten as the airspeed along the x' axis rotated into the inertial coordinate frame. The inertial velocity components can now be defined with respect to the orientation angles, airspeed, and wind vector.

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{pmatrix} = V_a \begin{pmatrix} \cos\chi_a \cos\gamma_a \\ \sin\chi_a \cos\gamma_a \\ -\sin\gamma_a \end{pmatrix} + \mathbf{w} \tag{2.9}$$

The derivatives of V_a, γ_a, and χ_a are derived by substituting Eq. (2.3) into Eq. (2.8) and differentiating the resulting equation:

$$\begin{aligned} \ddot{\mathbf{p}} =& V_a \dot{R}_z(\chi_a) R_y(\gamma_a) R_x(\phi) \mathbf{e}_1 + V_a R_z(\chi_a) \dot{R}_y(\gamma_a) R_x(\phi) \mathbf{e}_1 \\ &+ V_a R_z(\chi_a) R_y(\gamma_a) \dot{R}_x(\phi) \mathbf{e}_1 + \dot{V}_a R_z(\chi_a) R_y(\gamma_a) R_x(\phi) \mathbf{e}_1 + \dot{\mathbf{w}} \end{aligned} \tag{2.10}$$

The derivatives of the rotation matrices are defined by:

$$\dot{R}_z(\chi_a) = R_z(\chi_a)(\dot{\chi}_a \hat{\mathbf{e}_3}) = \dot{\chi}_a R_z(\chi_a) \hat{\mathbf{e}_3} \tag{2.11}$$

$$\dot{R}_y(\gamma_a) = R_y(\gamma_a)(\dot{\gamma}_a \hat{\mathbf{e}_2}) = \dot{\gamma}_a R_y(\gamma_a) \hat{\mathbf{e}_2} \tag{2.12}$$

www.manaraa.com

$$\dot{R}_x(\phi) = R_x(\phi)(\dot{\phi}\hat{\mathbf{e}_1}) = \dot{\phi}R_x(\phi)\hat{\mathbf{e}_1} \tag{2.13}$$

where $\hat{}$ represents a skew symmetric matrix of the given vector. These derivatives are substituted back into Eq. (2.10):

$$\begin{aligned}\ddot{\mathbf{p}} =& V_a\dot{\chi}_a R_z(\chi_a)\hat{\mathbf{e}_3}R_y(\gamma_a)R_x(\phi)\mathbf{e}_1 + V_a\dot{\gamma}_a R_z(\chi_a)R_y(\gamma_a)\hat{\mathbf{e}_2}R_x(\phi)\mathbf{e}_1 \\ &+ V_a\dot{\phi}R_z(\chi_a)R_y(\gamma_a)R_x(\phi)\hat{\mathbf{e}_1}\mathbf{e}_1 + \dot{V}_a R_z(\chi_a)R_y(\gamma_a)R_x(\phi)\mathbf{e}_1 + \dot{\mathbf{w}}\end{aligned} \tag{2.14}$$

The $R_x(\phi)$ terms can be eliminated with the multiplication by $\mathbf{e}_1$. Additionally, $\dot{\mathbf{w}}$ can be written in terms of the wind Jacobian and inertial velocity vector:

$$\begin{aligned}\ddot{\mathbf{p}} =& V_a\dot{\chi}_a R_z(\chi_a)\hat{\mathbf{e}_3}R_y(\gamma_a)\mathbf{e}_1 + V_a\dot{\gamma}_a R_z(\chi_a)R_y(\gamma_a)\hat{\mathbf{e}_2}\mathbf{e}_1 \\ &+ \dot{V}_a R_z(\chi_a)R_y(\gamma_a)\mathbf{e}_1 + J_w\dot{\mathbf{p}}\end{aligned} \tag{2.15}$$

The skew symmetric terms are evaluated and the $R_z(\chi_a), R_y(\gamma_a)$ rotation matrices can be pulled out of the rotation terms:

$$\ddot{\mathbf{p}} = R_z(\chi_a)R_y(\gamma_a)\Big(V_a\dot{\chi}_a\cos\gamma_a\mathbf{e}_2 - V_a\dot{\gamma}_a\mathbf{e}_3 + \dot{V}_a\mathbf{e}_1\Big) + J_w\dot{\mathbf{p}} \tag{2.16}$$

$$\ddot{\mathbf{p}} = R_z(\chi_a)R_y(\gamma_a)\begin{pmatrix}\dot{V}_a \\ V_a\dot{\chi}_a\cos\gamma_a \\ -V_a\dot{\gamma}_a\end{pmatrix} + J_w\dot{\mathbf{p}} \tag{2.17}$$

The inertial acceleration, $\ddot{\mathbf{p}}$, can be substituted using Newton's second law.

$$\ddot{\mathbf{p}} = g\mathbf{e}_3 + R_{i/w}(\frac{T-D}{m}\mathbf{e}_1 + \frac{-L}{m}\mathbf{e}_3) \tag{2.18}$$

Substituting Eq. (2.18) into Eq. (2.17) and solving for the vector containing $\dot{V}_a$, $\dot{\chi}_a$, and $\dot{\gamma}_a$ yields:

$$\begin{pmatrix}\dot{V}_a \\ V_a\dot{\chi}_a\cos\gamma_a \\ -V_a\dot{\gamma}_a\end{pmatrix} = R_x(\phi)(\frac{T-D}{m}\mathbf{e}_1 + \frac{-L}{m}\mathbf{e}_3) + R_y(\gamma_a)^{\mathrm{T}}R_z(\chi_a)^{\mathrm{T}}(g\mathbf{e}_3 - J_w\dot{\mathbf{p}}) \tag{2.19}$$

Solving for the individual derivative state components yields the dynamic equations for the

www.manaraa.com

airspeed, air-relative flight path angle, and air-relative course angle.

$$\dot{V}_a = -g\sin\gamma_a + \frac{T-D}{m} - \begin{pmatrix} \cos\gamma_a\cos\chi_a \\ \cos\gamma_a\sin\chi_a \\ -\sin\gamma_a \end{pmatrix}^{\mathrm{T}} J_w\dot{\mathbf{p}} \tag{2.20}$$

$$\dot{\gamma}_a = \frac{1}{V_a}(-g\cos\gamma_a + \frac{L}{m}\cos\phi + \begin{pmatrix} \sin\gamma_a\cos\chi_a \\ \sin\gamma_a\sin\chi_a \\ \cos\gamma_a \end{pmatrix}^{\mathrm{T}} J_w\dot{\mathbf{p}}) \tag{2.21}$$

$$\dot{\chi}_a = \frac{1}{V_a\cos\gamma_a}(\frac{L}{m}\sin\phi + \begin{pmatrix} \sin\chi_a \\ -\cos\chi_a \\ 0 \end{pmatrix}^{\mathrm{T}} J_w\dot{\mathbf{p}}) \tag{2.22}$$

$$\tag{2.23}$$

The equations of motion for the roll angle, ϕ ,and lift coefficient, C_L, are determined directly from the control inputs, $\dot{\phi}$ and $\dot{C_L}$, respectively.

www.manaraa.com

In summary, the full equations of motion are:

$$\dot{\mathbf{x}}(t) = f(\mathbf{x}(t), \mathbf{u}(t), \mathbf{w}(\mathbf{p}(t))) = \begin{pmatrix} V_a \cos\chi_a \cos\gamma_a + w_x \\ V_a \sin\chi_a \cos\gamma_a + w_y \\ -V_a \sin\gamma_a + w_z \\ -g\sin\gamma_a + \frac{T-D}{m} - \begin{pmatrix} \cos\gamma_a \cos\chi_a \\ \cos\gamma_a \sin\chi_a \\ -\sin\gamma_a \end{pmatrix}^{\mathrm{T}} J_w \dot{\mathbf{p}} \\ \frac{1}{V_a}(-g\cos\gamma_a + \frac{L}{m}\cos\phi + \begin{pmatrix} \sin\gamma_a \cos\chi_a \\ \sin\gamma_a \sin\chi_a \\ \cos\gamma_a \end{pmatrix}^{\mathrm{T}} J_w \dot{\mathbf{p}}) \\ \frac{1}{V_a \cos\gamma_a}(\frac{L}{m}\sin\phi + \begin{pmatrix} \sin\chi_a \\ -\cos\chi_a \\ 0 \end{pmatrix}^{\mathrm{T}} J_w \dot{\mathbf{p}}) \\ u_1 \\ u_2 \end{pmatrix} \tag{2.24}$$

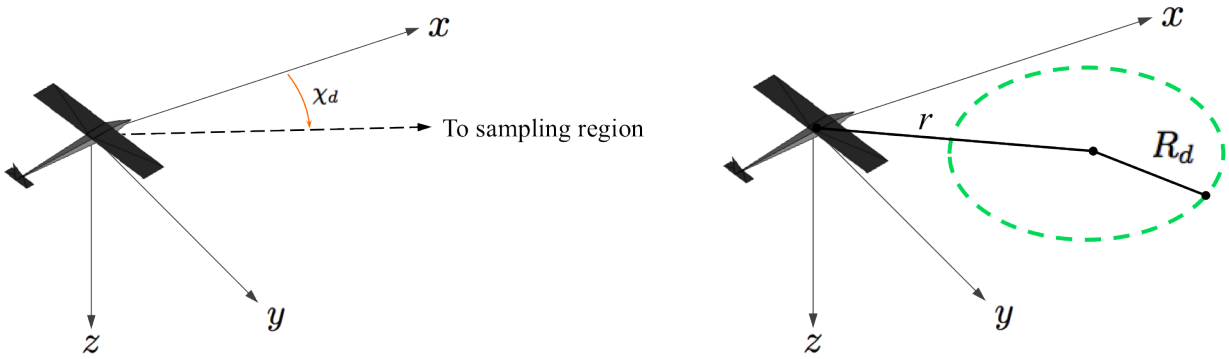
where u_1 and u_2 are the first and second terms of the control input vector, $\mathbf{u}(t)$.

www.manaraa.com

2.3 Sampling and Surveillance Mission

The focus of this work is to extract wind energy in order to increase the amount of time a UAS performs a sampling and surveillance mission. For this work, the sampling and surveillance mission is defined as staying inside of a sampling region. However, in many cases the sampling region is not at the UAS launch site. In order to perform the sampling and surveillance mission, the UAS first needs to traverse to the desired region. The mission can be improved by extracting wind energy as the UAS traverses, in addition to extracting energy upon reaching the desired region. For this reason the full sampling and surveillance mission is divided into the guidance and loiter missions.

The purpose of the guidance mission is to direct the UAS in the direction of the sampling region, while extracting wind energy. The UAS is directed along the heading angle, χ_d. This heading angle lies in the $x-y$ plane between the x axis and the ray pointing from the aircraft's position to the sampling region as depicted by Fig. 2.2 (a).



(a) Guidance Mission (b) Loiter Mission

Figure 2.2: Guidance and Loiter Mission Diagrams

The purpose of the loiter mission is to keep the UAS in the sampling region, while extracting wind energy. In this work, the sampling region is defined as a cylinder of radius R_d. The center of the sampling region is located $\begin{pmatrix} x_g, & y_g, & 0 \end{pmatrix}^{\mathrm{T}}$. The distance from the sampling region center to the UAS is defined as r as shown in Fig.2.2 (b). Note the green, dashed circle is used throughout this paper to represent the cylindrical sampling region.

www.manaraa.com

Chapter 3

Optimization

This chapter formulates the trajectory optimization as a nonlinear program and derives a method for constructing initial trajectories for the optimization.

3.1 Nonlinear Program

The aircraft model discussed in Section 2.1 defines a system with continuous states and control inputs. The first step in formulating the optimization as a nonlinear program is to discretize these continuous states and control inputs into finite vectors as depicted in Fig.3.1.

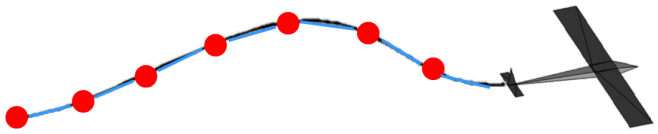


Figure 3.1: Discretization of Continuous UAS Trajectory

The state and control inputs are discretized into N time nodes. The time between these time nodes is fixed by the time step, Δt. A forward Euler approximation is used to enforce the equations of motion on the discretized state vector:

$$\mathbf{x}_{k+1} = \mathbf{x}_k + f(\mathbf{x}_k, \mathbf{u}_k, \mathbf{w}(\mathbf{p}_k))\Delta t, \ \forall k \in [0, N-1] \tag{3.1}$$

The nonlinear program is developed with the decision vector, $\mathbf{z}$, which is the concatenation

www.manaraa.com

of the discretized states and control inputs:

$$\mathbf{z} = \begin{pmatrix} \Delta t & \mathbf{x}_0 & \mathbf{u}_0 & \mathbf{x}_1 & \mathbf{u}_1 & \dots & \mathbf{x}_{N-1} & \mathbf{u}_{N-1} & \mathbf{x}_N \end{pmatrix}^{\mathrm{T}} \tag{3.2}$$

Note the time step, Δt, is included in this decision vector.

The problem formulation for the generic nonlinear program is defined by:

Problem 1

$$\begin{aligned} \textbf{min} \quad & \mathrm{J}(\mathbf{z}) \\ \textbf{s.t.} \quad & \mathcal{F}_f(\mathbf{z}) = 0 \\ & \mathcal{F}_b(\mathbf{z}) \leq 0 \\ & \mathcal{F}_p(\mathbf{z}) = 0 \\ & \mathcal{F}_i(\mathbf{z}) = 0 \\ & \mathcal{F}_o(\mathbf{z}) = 0 \end{aligned}$$

where J is the cost function to minimize subject to the set of constraints, $\mathcal{F}$. The set of constraints include the equation of motion constraints, $\mathcal{F}_f$, system bounds, $\mathcal{F}_b$, periodic constraints, $\mathcal{F}_p$, and initial condition constraints, $\mathcal{F}_i$. Any additional constraints specific to a particular problem are defined in $\mathcal{F}_o$. The equation of motion constraints enforce the forward Euler approximation defined in Eq. (3.1):

$$\mathcal{F}_f(\mathbf{z}) = \mathbf{x}_{k+1} - \mathbf{x}_k - f(\mathbf{x}_k, \mathbf{u}_k, \mathbf{w}(\mathbf{p}_k))\Delta t, \quad \forall k \in [0, N-1] \tag{3.3}$$

www.manaraa.com

The system bounds apply upper and lower boundaries on $\mathbf{z}$.

$$\mathcal{F}_b(\mathbf{z}) = \begin{pmatrix} \Delta t - \Delta t_{max} \\ -\Delta t + \Delta t_{min} \\ x_k - x_{max} \\ -x_k + x_{min} \\ y_k - y_{max} \\ -y_k + y_{min} \\ z_k - z_{max} \\ -z_k + z_{min} \\ V_{a_k} - V_{a_{max}} \\ -V_{a_k} + V_{a_{min}} \\ \gamma_{a_k} - \gamma_{a_{max}} \\ -\gamma_{a_k} + \gamma_{a_{min}} \\ \chi_{a_k} - \chi_{a_{max}} \\ -\chi_{a_k} + \chi_{a_{min}} \\ \phi_k - \phi_{max} \\ -\phi_k + \phi_{min} \\ C_{L_k} - C_{L_{max}} \\ -C_{L_k} + C_{L_{min}} \\ \dot{\phi}_k - \dot{\phi}_{max} \\ -\dot{\phi}_k + \dot{\phi}_{min} \\ \dot{C}_{L_k} - \dot{C}_{L_{max}} \\ -\dot{C}_{L_k} + \dot{C}_{L_{min}} \\ T_k - T_{max} \\ -T_k + T_{min} \end{pmatrix}, \quad \forall k \in [0, N] \tag{3.4}$$

The periodic constraints force the specified states at the final time node, $k = N$, to be equal

www.manaraa.com

to the corresponding states at $k = 0$.

$$\mathcal{F}_p(\mathbf{z}) = \begin{pmatrix} x_N - x_0 \\ y_N - y_0 \\ z_N - z_0 \\ V_{a_N} - V_{a_0} \\ \gamma_{a_N} - \gamma_{a_0} \\ \chi_{a_N} - \chi_{a_0} \\ \phi_N - \phi_0 \\ C_{L_N} - C_{L_0} \end{pmatrix} \tag{3.5}$$

Finally, the initial condition constraints force the specified states and control inputs at $k = 0$ to a specific initial value.

$$\mathcal{F}_i(\mathbf{z}) = \begin{pmatrix} x_0 - x(0) \\ y_0 - y(0) \\ z_0 - z(0) \\ V_{a_0} - V_a(0) \\ \gamma_{a_0} - \gamma_a(0) \\ \chi_{a_0} - \chi_a(0) \\ \phi_0 - \phi(0) \\ C_{L_0} - C_L(0) \\ \dot{\phi}_0 - \dot{\phi}(0) \\ \dot{C}_{L_0} - \dot{C}_L(0) \\ T_0 - T(0) \end{pmatrix} \tag{3.6}$$

The periodic and initial condition constraints may only apply to select states and/or control inputs depending on the problem formulation. In order to address these scenarios, the notation will explicitly define the states and control inputs included or excluded from either the periodic or initial condition constraints. For example, if the periodic constraint excludes the inertial position and airspeed, then the periodic constraint notation is: $\mathcal{F}_p(\mathbf{z} \setminus \{x_k, y_k, z_k, V_{a_k}\}) = 0$. On the other

www.manaraa.com

hand say the initial condition constraint only applies to the inertial position and air-relative flight path angle. The initial condition constraint notation would then be: $\mathcal{F}_i(x_0, y_0, z_0, \gamma_{a_0}) = 0$

The nonlinear program defined in Problem 1 is used to minimize the cost function,J. The focus of this work is to reduce the thrust use in order to reduce the on-board fuel consumption. Therefore the cost function is formulated based on the summation of the thrust. The general form of the cost function is:

$$\mathrm{J}(\mathbf{z}) = \sum_{k}^{N-1} \left(T_k^2 + h_b(\mathbf{z}) \right) + h_t(\mathbf{z}) \tag{3.7}$$

where h_b and h_t are additional cost function terms.

In order to develop a trajectory solution, the nonlinear program is solved using a commercial solver known as the Sparse Nonlinear Optimizer (SNOPT) [1]. SNOPT is a sequential quadratic programming algorithm that is effective for large-scale, sparse optimization problems. SNOPT requires the cost, J, constraints, $\mathcal{F}$, and can exploit expressions for the gradient of the costs and constraints with respect to the input vector $\mathbf{z}$. Recall that many of the constraints only apply to specific states and inputs at a time. For example, the equation of motion constraint only constrains state x_k to the next state x_{k+1}. In the periodicity constraint only the final and initial states of the trajectory are constrained. Therefore the gradient of the constraint becomes a large, sparse gradient matrix, which is ideal for SNOPT. Other optimization parameters associated with SNOPT include the optimality tolerance, tol_{opt}, and feasibility tolerance, tol_{fea}. The optimizer uses the optimality tolerance to determine when a trajectory is deemed optimal. The feasibility tolerance is used by the optimizer to determine when the constraints are satisfied.

www.manaraa.com

3.2 Initial Trajectory Development

Optimizers such as SNOPT require an initial trajectory in order to begin the optimization. To improve the performance of the optimizer, it is beneficial to provide an initial trajectory that satisfies most of the problem constraints. One way to do this is to exploit the differential flatness property of the presented aircraft model [10]. Due to this property, all of the states and control inputs of the system can be determined from a given inertial position, $\mathbf{p}(t)$, inertial velocity, $\dot{\mathbf{p}}(t)$, and inertial acceleration, $\ddot{\mathbf{p}}(t)$. By defining the inertial path of the aircraft, the entire trajectory can be developed to satisfy the equation of motion and initial condition constraints. By choosing an appropriate path, most of the system bounds and periodic constraints can also be satisfied. If not all the constraints can be satisfied, the differential flatness property can still provide SNOPT with an initial trajectory that is close enough to a neighborhood of feasible trajectories.

The states and inputs for this initial trajectory are computed by first defining the components of $R_{i/w}$ in terms of column vectors:

$$R_{i/w} = \begin{bmatrix} \mathbf{r}_1 & \mathbf{r}_2 & \mathbf{r}_3 \end{bmatrix} \tag{3.8}$$

The air-relative velocity can be rewritten using Eq. (3.8).

$$\dot{\mathbf{p}}_a = V_a R_{i/w} \mathbf{e}_1 = V_a \mathbf{r}_1 \tag{3.9}$$

The airspeed is substituted by the magnitude of the air-relative velocity to yield the equation for $\mathbf{r}_1$:

$$\mathbf{r}_1 = \frac{\dot{\mathbf{p}}_a}{||\dot{\mathbf{p}}_a||} \tag{3.10}$$

Equation (2.18) can be rewritten using acceleration terms.

$$\ddot{\mathbf{p}} = g\mathbf{e}_3 + R_{i/w}(a_t \mathbf{e}_1 - a_n \mathbf{e}_3) \tag{3.11}$$

where a_t represents the tangential acceleration in the wind x'-axis due to the thrust and drag, and a_n represents the normal acceleration in the negative wind z'-axis due to the lift.

www.manaraa.com

The tangential acceleration is determined by rearranging Eq. (3.11).

$$\begin{aligned} a_t\mathbf{e}_1 - a_n\mathbf{e}_3 &= R_{i/w}^{\mathrm{T}}(\ddot{\mathbf{p}} - g\mathbf{e}_3) \\ a_t &= \mathbf{e}_1^{\mathrm{T}} R_{i/w}^{\mathrm{T}}(\ddot{\mathbf{p}} - g\mathbf{e}_3) \\ a_t &= \mathbf{r}_1^{\mathrm{T}}(\ddot{\mathbf{p}} - g\mathbf{e}_3) \end{aligned} \tag{3.12}$$

The normal acceleration can be solved for using Eq. (3.11) and Eq. (3.12).

$$\begin{aligned} a_t\mathbf{r}_1 - a_n\mathbf{r}_3 &= \ddot{\mathbf{p}} - g\mathbf{e}_3 \\ -a_n\mathbf{r}_3 &= -a_t\mathbf{r}_1 + \ddot{\mathbf{p}} - g\mathbf{e}_3 \\ -a_n\mathbf{r}_3 &= (-\mathbf{r}_1\mathbf{r}_1^{\mathrm{T}} + I)(\ddot{\mathbf{p}} - g\mathbf{e}_3) \\ a_n &= \|(-\mathbf{r}_1\mathbf{r}_1^{\mathrm{T}} + I)(\ddot{\mathbf{p}} - g\mathbf{e}_3)\| \end{aligned} \tag{3.13}$$

The rotation vector $\mathbf{r}_3$ can also be solved for from the previous derivation.

$$\mathbf{r}_3 = \frac{(-\mathbf{r}_1\mathbf{r}_1^{\mathrm{T}} + I)(\ddot{\mathbf{p}} - g\mathbf{e}_3)}{-a_n} \tag{3.14}$$

Finally the rotation matrix, $R_{i/w}$, can be written in terms of the desired soaring path by computing $\mathbf{r}_2$ from $\mathbf{r}_1$ and $\mathbf{r}_3$.

$$\mathbf{r}_2 = \mathbf{r}_3 \times \mathbf{r}_1 \tag{3.15}$$

The lift coefficient and thrust are computed from the tangential and normal accelerations.

$$C_L = \frac{2ma_n}{\rho V_a^2 S} \tag{3.16}$$

$$T = ma_t + D \tag{3.17}$$

www.manaraa.com

The roll angle is derived from the definition of the rotation matrix, $R_{i/w}$.

$$\begin{aligned} R_{i/w} &= R_z(\chi_a)R_y(\gamma_a)R_z(\phi) \\ \mathbf{e}_3^{\mathrm{T}} R_{i/w} &= \mathbf{e}_3^{\mathrm{T}} R_z(\chi_a)R_y(\gamma_a)R_z(\phi) \\ \mathbf{e}_3^{\mathrm{T}} R_{i/w} &= \mathbf{e}_3^{\mathrm{T}} R_y(\gamma_a)R_z(\phi) \\ \begin{pmatrix} \mathbf{e}_3^{\mathrm{T}}\mathbf{r}_1 \\ \mathbf{e}_3^{\mathrm{T}}\mathbf{r}_2 \\ \mathbf{e}_3^{\mathrm{T}}\mathbf{r}_3 \end{pmatrix}^{\mathrm{T}} &= \begin{pmatrix} -\sin\gamma_a \\ \cos\gamma_a \sin\phi \\ \cos\gamma_a \cos\phi \end{pmatrix}^{\mathrm{T}} \end{aligned}$$

$$\phi = \tan^{-1}\left(\frac{\mathbf{e}_3^{\mathrm{T}}\mathbf{r}_2}{\mathbf{e}_3^{\mathrm{T}}\mathbf{r}_3}\right) \tag{3.18}$$

Finally χ_a and γ_a can be derived directly by re-arranging Eq. (2.9).

$$\chi_a = \tan^{-1}\left(\frac{\dot{y} - w_y}{\dot{x} - w_x}\right) \tag{3.19}$$

$$\gamma_a = \sin^{-1}\left(\frac{-\dot{z} + w_z}{V_a}\right) \tag{3.20}$$

The inputs $\dot{\phi}$ and $\dot{C}_L$ can be approximated using finite differencing over a desired time step, Δt.

$$\dot{\phi} = \frac{\Delta\phi}{\Delta t} \tag{3.21}$$

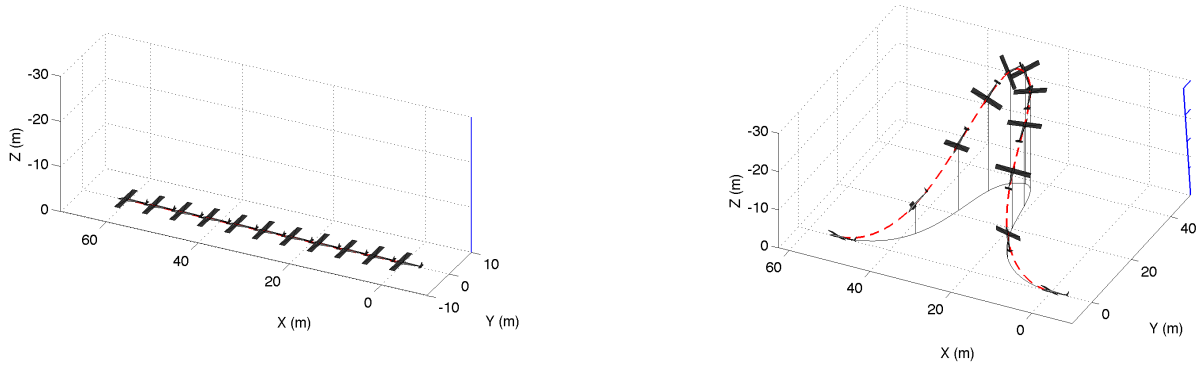
$$\dot{C}_L = \frac{\Delta C_L}{\Delta t} \tag{3.22}$$

With all the states and inputs as functions of inertial position, velocity, and acceleration, initial trajectories can be developed to initialize the optimization.

www.manaraa.com

3.3 Periodicity Validation

With the nonlinear program defined and the optimizer chosen, an example trajectory can be developed to demonstrate the benefit of the periodicity constraint, which allows trajectories to stitch together. Consider solving a trajectory for the guidance mission with an initial steady level flight trajectory.



(a) Initial Steady Level Flight Trajectory

(b) Minimized Thrust Soaring Trajectory

Figure 3.2: Periodic Soaring Example

www.manaraa.com

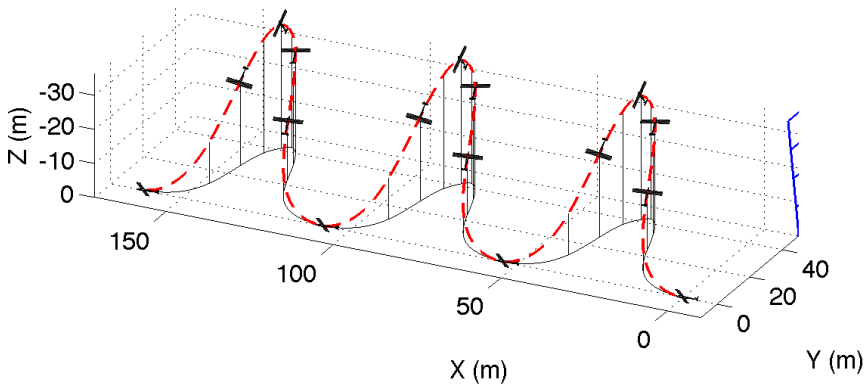


Figure 3.3: Periodic Stitching of Soaring Trajectory

Figure 3.2(b) shows the resulting minimized thrust soaring trajectory that the optimizer develops from the initial trajectory in Fig. 3.2 (a). Figure 3.3 demonstrates how after solving one soaring trajectory, the trajectory can be combined or stitched together to provide a long term trajectory solution.

The full trajectory output by the optimizer for the soaring segment in Fig. 3.2 (b) is shown in the following plots.

www.manaraa.com

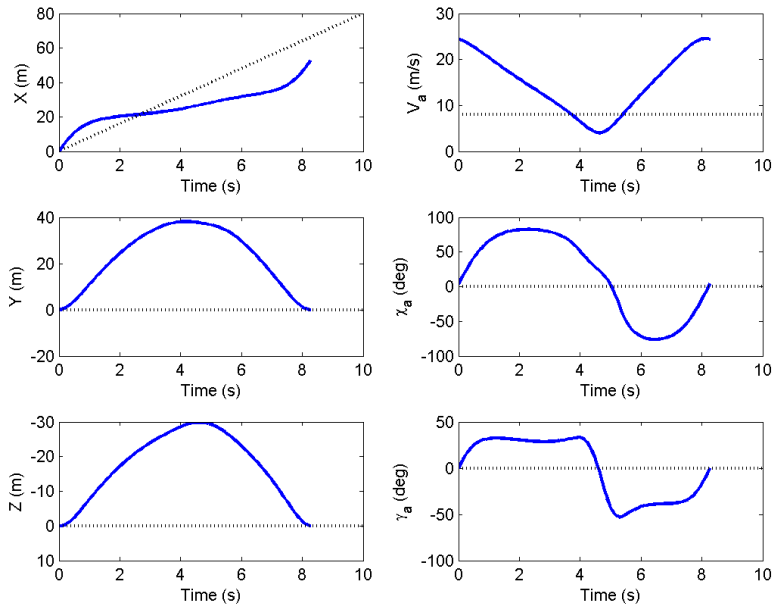


Figure 3.4: Example trajectory output from the optimization (States/Inputs Set 1)

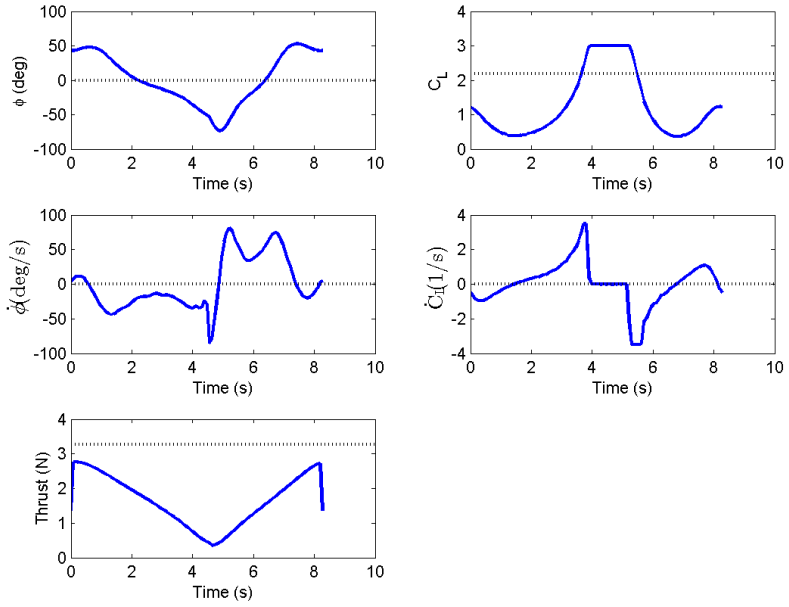


Figure 3.5: Example trajectory output from the optimization (States/Inputs Set 2)

www.manaraa.com

Chapter 4

Mission Tree

The purpose of the mission tree is to provide the optimizer with the desired problem formulation to solve. This problem formulation includes the cost, constraints, initial trajectory, and optimization parameters defined in Chapter 3.

The mission tree is a heuristic algorithm to formulate an appropriate optimization problem. The tree is defined prior to launch by the mission designer so that the UAS can be given a wind model and automatically decide what optimization problems to solve. One difficulty in choosing a problem is that there are many different ways to formulate an optimization problem for a given mission. Some studies analyze only the performance of different cost functions let alone analyzing the variations of cost and constraints [16] [5]. Another difficulty is that a specific problem may be overdetermined with many possible solutions. This would suggest that the degrees of freedom in the problem need to be reduced. Conversely, a specific problem may have no solution. For example, the optimizer can locate a minimum trajectory in the solution space, but cannot reach it due to the constraints of the problem. Here it may be beneficial to relax the constraints to reach a neighborhood of feasible solutions.

Another key component of the mission tree involves setting up the logical structure for choosing a problem. In some cases, iterations may be needed to compare the results between different problems. For example, it may be beneficial to solve a gliding problem for a powered aircraft in case the wind environment provides sufficient energy for zero thrust flight. This would allow the UAS to improve mission performance by focusing on a cost function other than minimizing thrust.

www.manaraa.com

The development of this logic structure is important, but is not the focus of this work. Instead a problem formulation is chosen from the guidance and loiter trees to analyze in the results section.

4.1 Guidance Tree

Recall the purpose of the guidance mission is to direct the UAS towards the sampling region, while extracting energy from the wind.

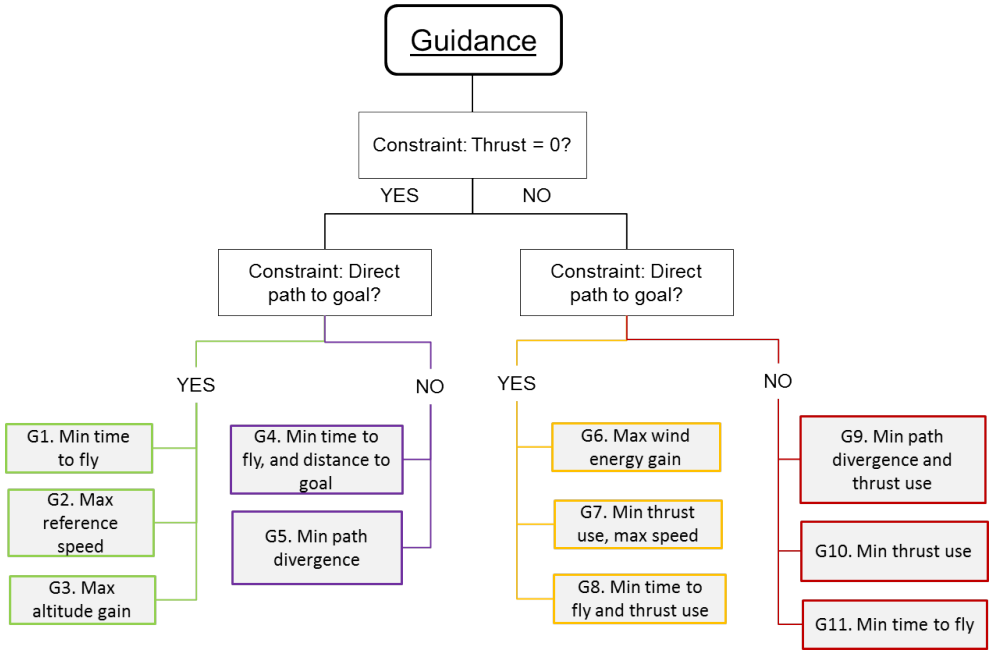


Figure 4.1: Guidance Tree

Figure 4.1 shows the logical flow of how the guidance tree decides which problems to pass to the optimizer. The main decisions in the tree correspond to increasing flight endurance and performing the sampling and surveillance mission. The first decision divides the tree into powered or gliding flight problem formulations. The gliding problems set thrust to zero and focus on minimizing different costfunctions to improve the sampling and surveillance mission. The powered flight problems strive to minimize the thrust and an additional costterm to maximize the aircraft's flight endurance.

www.manaraa.com

The second decision deals with constraining the UAS to fly in the direction of the sampling region. In certain cases, such as when dealing with obstacles, it may be impractical to direct the UAS straight to the sampling region. Instead the problems may include a term in the costfunction to stay close to the direction of the goal.

The multiple problem descriptions seen in the leaves of the guidance tree demonstrate possible problem descriptions that a mission designer could implement. For this work, Problem G.7 is chosen to analyze the performance of the path planning algorithm for developing guidance mission trajectories. Problem G.7 minimizes the thrust use and maximizes the speed for a powered aircraft constrained to flying towards the sampling region. The formal problem definition of Problem G.7 is given by:

Problem 2 (G.7)

$$
\begin{aligned}
\textbf{min} \quad & \mathrm{J}(\mathbf{z}) = \textstyle\sum_{k=0}^{N-1}(k_T \frac{T_k^2}{2}) + k_p(\frac{N\Delta t}{\sqrt{(x_N-x_0)^2+(y_N-y_0)^2}}) \\
\textbf{s.t.} \quad & \mathcal{F}_f(\mathbf{z}) = 0 \\
& \mathcal{F}_b(\mathbf{z}) \leq 0 \\
& \mathcal{F}_p(\mathbf{z} \setminus \{x_k, y_k\}) = 0 \\
& \mathcal{F}_i(x_0, y_0, z_0, \gamma_{a_0}) = 0 \\
& x_N = x_0 + \sqrt{(x_N - x_0)^2 + (y_N - y_0)^2}\cos\chi_d \\
& y_N = y_0 + \sqrt{(x_N - x_0)^2 + (y_N - y_0)^2}\sin\chi_d
\end{aligned}
\tag{4.1}
$$

where $k_T \geq 0$ is the thrust gain and $k_p \geq 0$ is the position gain.

The costin Problem G.7 consists of a summation of the thrust in addition to an average speed term. The thrust component is to develop a minimized thrust trajectory and thereby increase the flight endurance of the UAS. The speed term allows the mission designer to adjust how quickly the UAS flies towards the sampling region. By applying gains to both components, the designer can tune between reducing thrust or increasing the average speed of the trajectory. The speed in this case is the distance between the final and initial inertial positions divided by the total time of the trajectory.

www.manaraa.com

Problem G.7 implements periodic constraints on all states except for the inertial x and y positions. In order to traverse to the sampling region, the final UAS position cannot be constrained to the initial UAS position or else there would be no net movement. Instead directional constraints are applied to x_N and y_N. This constraint ensures that the final x and y positions align with the initial positions such that the soaring trajectory moves toward the sampling region. Recall the heading angle, χ_d, directs the UAS towards the sampling region.

4.2 Loiter Tree

The purpose of the loiter mission is to keep the UAS inside the sampling region, while extracting energy from the wind.

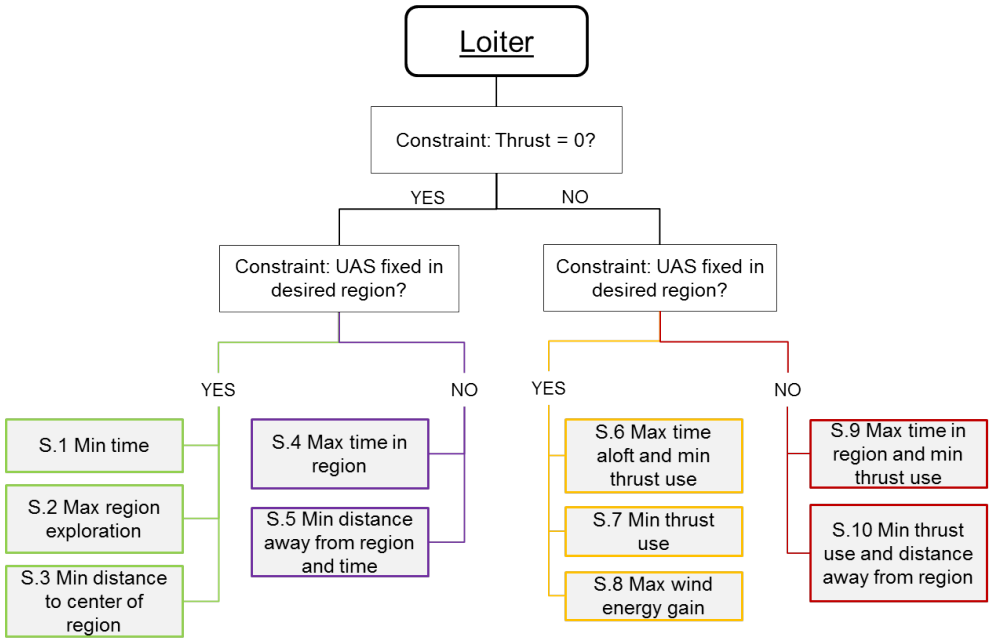


Figure 4.2: Loiter Tree

Figure 4.2 shows the logical flow for choosing a loiter mission problem and has a similar structure to the guidance tree. The first decision once again deals with choosing between a powered or gliding flight problem formulation. The second decision deals with whether the UAS is allowed

www.manaraa.com

to leave the sampling region. At this point the loiter tree must weigh the benefits between leaving the sampling region if for example there is a wind energy source nearby to increase the aircraft's flight endurance.

For this work, Problem S.10 is chosen to analyze the performance of the path planning algorithm for developing loiter mission trajectories. The formal problem definition of Problem S.10 is given by:

Problem 3 (S.10)

$$\begin{aligned} \textbf{min} \quad & \mathrm{J}(\mathbf{z}) = \textstyle\sum_{k=0}^{N-1}(k_T \frac{T_k^2}{2} + k_p h(\mathbf{x})) \\ \textbf{s.t.} \quad & \mathcal{F}_f(\mathbf{z}) = 0 \\ & \mathcal{F}_b(\mathbf{z}) \leq 0 \\ & \mathcal{F}_p(\mathbf{z}) = 0 \\ & \mathcal{F}_i(x_0, y_0, z_0, \gamma_{a_0}) = 0 \end{aligned} \tag{4.2}$$

where $h(\mathbf{x})$ is the penalty function on the UAS position with respect to the sampling region defined by:

$$h(\mathbf{x}) = \begin{cases} \frac{(r_k - R_d)^2}{2}, & r_k \geq R_d, \\ 0, & r_k < R_d \end{cases} \tag{4.3}$$

The costfunction of Problem S.10 consists of a summation of the thrust in addition to the penalty function, $h(x)$. Once again the minimization of thrust is used to increase the flight endurance of the UAS. The penalty function is one way to reduce the amount of time the UAS spends outside of the sampling/surveillance region. The penalty function works by applying a penalty to a trajectory that takes the UAS outside of the sampling region. This penalty is proportional to the square of the difference between the UAS distance from the center of the sampling region and the radius of the sampling/surveillance region. The benefit of including this penalty function in the cost is so the mission designer can tune between reducing the thrust or keeping the UAS inside the sampling region by adjusting the thrust gain, k_T, and the position gain, k_p.

Problem S.10 applies the periodic constraint on all the states and inputs. In the loiter mission, the UAS has already reached the sampling region and is trying to solve the long horizon problem of

www.manaraa.com

maximizing UAS flight endurance. Instead of using a receding horizon approach to solving the long horizon problem, Problem S.10 takes advantage of the periodic trajectory concept as demonstrated with the guidance mission in Chapter 3. The benefit of the periodic constraint is that the optimizer need only solve one soaring segment trajectory for the stationary wind model. This single trajectory can then be stitched together to provide a reduced thrust solution to the long horizon problem.

4.3 Myopic Stitching

One of the key benefits of implementing the periodic soaring constraint presented in Chapter 3.1, is to be able to develop long term trajectory solutions for the guidance and loiter missions. However in order to create the full sampling and surveillance mission trajectory, the guidance and loiter trajectories must be stitched together.

This work uses a myopic stitching approach in which the guidance problem is first solved in order for the UAS to traverse towards the sampling region. Once the UAS approaches the sampling region, a transition guidance problem is needed to actually reach the desired position inside or near the sampling region. Finally, once the UAS reaches the sampling region, the loiter problem is solved to complete the sampling and surveillance mission. This transition problem used to link the guidance and loiter trajectories is defined by:

Problem 4 (Gt.7)

$$\begin{aligned}
\textbf{min} \quad & \mathrm{J}(\mathbf{z}) = \textstyle\sum_{k=0}^{N-1}(k_T \frac{T_k^2}{2}) + k_p(\frac{N\Delta t}{\sqrt{(x_N-x_0)^2+(y_N-y_0)^2}}) \\
\textbf{s.t.} \quad & \mathcal{F}_f(\mathbf{z}) = 0 \\
& \mathcal{F}_b(\mathbf{z}) \leq 0 \\
& \mathcal{F}_p(\mathbf{z} \setminus \{x_k, y_k\}) = 0 \\
& \mathcal{F}_i(\mathbf{z}) = 0 \\
& x_N = x_s \\
& y_N = y_s
\end{aligned} \tag{4.4}$$

where x_s and y_s defines the inertial position in the sampling region from which the loiter problem

www.manaraa.com

will be solved. Problem Gt.7 is equivalent to Problem G.7 presented in Chapter 4.1, except for the initial condition constraints and final constraints on x_N and y_N. The final constraints allow the UAS to reach the specific location (x_s,y_s) in the sampling region. The initial condition constraint in Problem Gt.7 encompasses all of the initial states to be fixed to the final states of the guidance trajectory. This is needed for the stitching between the guidance and transition trajectories to be continuous. The initial condition constraint in Problem Gt.7 must also be applied to Problem S.10 in order for the stitching between the transition and loiter trajectories to be continuous.

www.manaraa.com

Chapter 5

Results

This chapter provides an assessment of the presented path planning algorithm.

5.1 Aircraft Parameters

The aircraft model used in the following simulations are based off the Tempest UAS [13]. The parameters that define the aircraft model are defined in Table 5.1.

Table 5.1: Aircraft Model Parameters

Parameter	Value
m	5.7419 kg
S	0.6316 m^2
e	0.9693
C_{d_0}	0.0310
AR	16.4457
g_a	9.81 m/s^2
ρ	1.2682 kg/m^3

5.2 Wind Models

The two wind models used in the following simulations are the linear boundary layer wind model and thermal wind model. The linear boundary layer wind model is defined by:

$$\mathbf{w} = \begin{pmatrix} 0, & \mu z, & 0 \end{pmatrix}^{\mathrm{T}} \tag{5.1}$$

www.manaraa.com

$$J_w = \begin{bmatrix} 0, & 0, & 0 \\ 0, & 0, & \mu \\ 0, & 0, & 0 \end{bmatrix} \tag{5.2}$$

where μ defines the wind gradient or slope of the linear boundary layer.

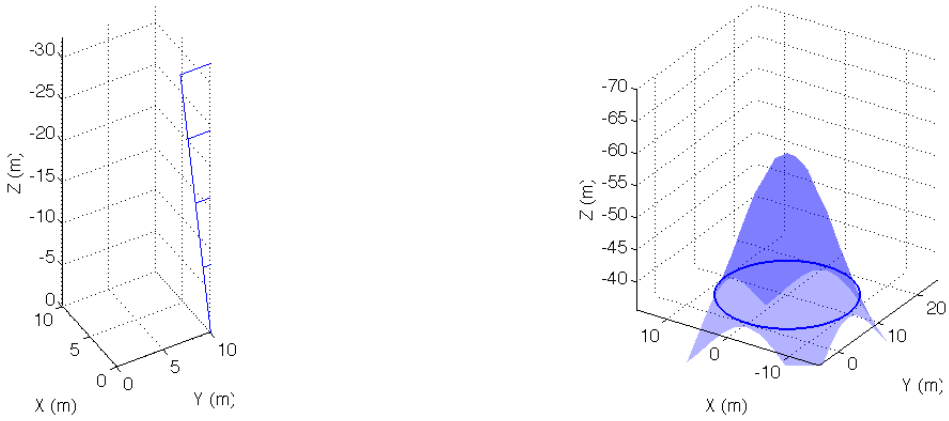
The thermal wind model is defined by:

$$\mathbf{w} = \begin{pmatrix} 0, & 0, & -V_{core} e^{\frac{-r^2}{R_{lift}^2}} \end{pmatrix}^{\mathrm{T}} \tag{5.3}$$

$$J_w = \begin{bmatrix} 0, & 0, & 0 \\ 0, & 0, & 0 \\ 2V_{core} e^{\frac{-r^2}{R_{lift}^2}} \frac{(x-x_{th})}{R_{lift}^2}, & 2V_{core} e^{\frac{-r^2}{R_{lift}^2}} \frac{(y-y_{th})}{R_{lift}^2}, & 0 \end{bmatrix} \tag{5.4}$$

where V_{core} is the core strength of the thermal, (x_{th}, y_{th}) defines the thermal position in the inertial $x-y$ plane, r is the radial distance from the aircraft to the thermal center, and R_{lift} is the radius of the thermal at which the upwind strength is 30% of the core strength.

A picture representation of these two wind models is shown in Fig 5.1. Note that the linear



(a) Linear Boundary Layer

(b) Thermal Wind

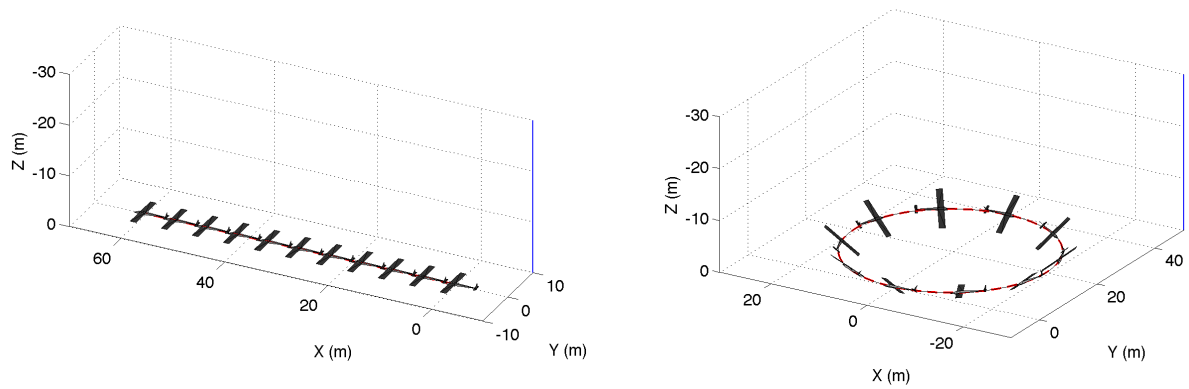
Figure 5.1: Wind Models

boundary layer model is a function of z which is negative for a positive altitude. For this reason, the direction of the wind in the linear boundary layer wind models used in the following simulations is always pointed into the negative y-axis.

www.manaraa.com

5.3 Baseline Trajectories

These baseline trajectories are used to compare the performance of the path planning algorithm, and are used as the initial trajectories given to the optimizer. Figure 5.2 shows the two



(a) Guidance Baseline Trajectory

(b) Loiter Baseline Trajectory

Figure 5.2: Baseline Trajectories

types of baseline trajectories used in this work. Both baseline trajectories are constant altitude flight trajectories to accomplish the guidance and loiter missions, respectively. Both trajectories are developed using the initial trajectory development approach from Chapter 3.2. The path used to develop the guidance baseline trajectory is defined by:

$$\begin{aligned} x(t) &= x_f \frac{t}{tf} + x_0 \\ y(t) &= 0 \\ z(t) &= 0 \end{aligned} \tag{5.5}$$

where x_f defines the distance traveled and $t_f = N\Delta t$ is the total time of the trajectory.

The path used to develop the loiter baseline trajectory is defined by:

$$\begin{aligned} x(t) &= A_x \sin(\frac{2\pi}{t_f}t) + x_0 \\ y(t) &= -A_y \cos(\frac{2\pi}{t_f}t) + A_y + y_0 \\ z(t) &= 0 \end{aligned} \tag{5.6}$$

www.manaraa.com

where A_x and A_y define the sinusoidal amplitude in the x-axis and y-axis, respectively.

Examples of the state history for baseline trajectories in a linear boundary layer wind model are shown in 5.2. The guidance trajectory was developed for a 10 second flight over 60 meters. The loiter trajectory was developed for a 10 second flight with a 20 m radius orbit. The values

Table 5.2: Baseline Trajectory States

	Guidance	Loiter
N	100	100
t_f	10 s	10 s
$V_a(t)$	6.0 m/s	12.6 m/s
$\gamma_a(t)$	0.0°	0.0°
$\chi_a(t)$	0.0°	[0.0° : 360.0°]
$\phi(t)$	0.0°	38.8°
$C_L(t)$	3.9	1.1
$T(t)$	4.8 N	3.6 N

shown in Table 5.2 represent fixed state values over the course of the baseline trajectories. The only exception is the course angle, $\chi_a(t)$, for the loiter case which completes a circular orbit from 0° to 360°.

www.manaraa.com

5.4 Validation Trajectories

This section characterizes the types of trajectories required to reduce thrust in a linear boundary layer wind model. The cost function for both the guidance and loiter missions is: $J(\mathbf{z}) = \sum_{k}^{N-1} \frac{1}{2} k_T T_k^2$. This means that the results will show what type of trajectories are beneficial for only reducing thrust use. Appendix A.1 holds the optimization parameters used to develop the trajectories.

5.4.1 Guidance Example

For the guidance mission, the sampling region is located along the x-axis with a heading angle of $\chi_d = 0°$. The baseline trajectory traverses 80 m along this axis in 10 s. This baseline trajectory requires a steady thrust of 3.3 N, which results in a cost of 5.4x10^4. The wind gradient for the linear boundary wind model is $\mu = 0.12 s^{-1}$.

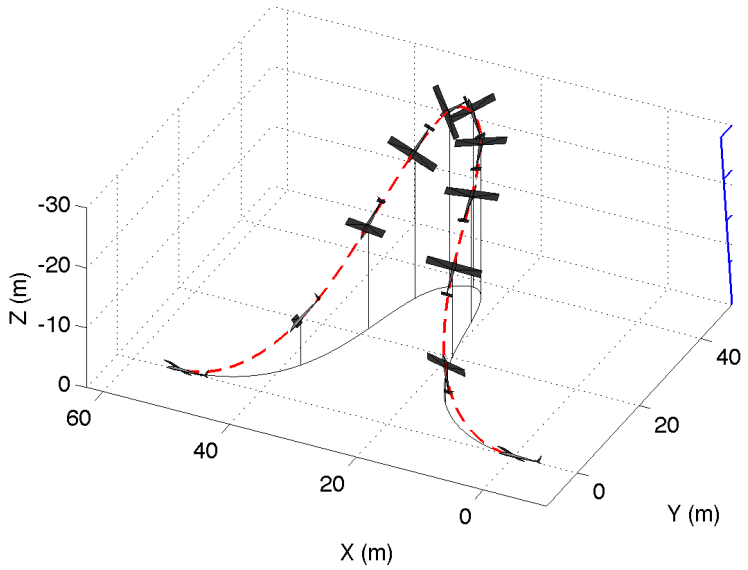


Figure 5.3: Guidance Example 3D Trajectory ($k_T = 100$, $k_p = 0$, $\mu = 0.12\ s^{-1}$)

www.manaraa.com

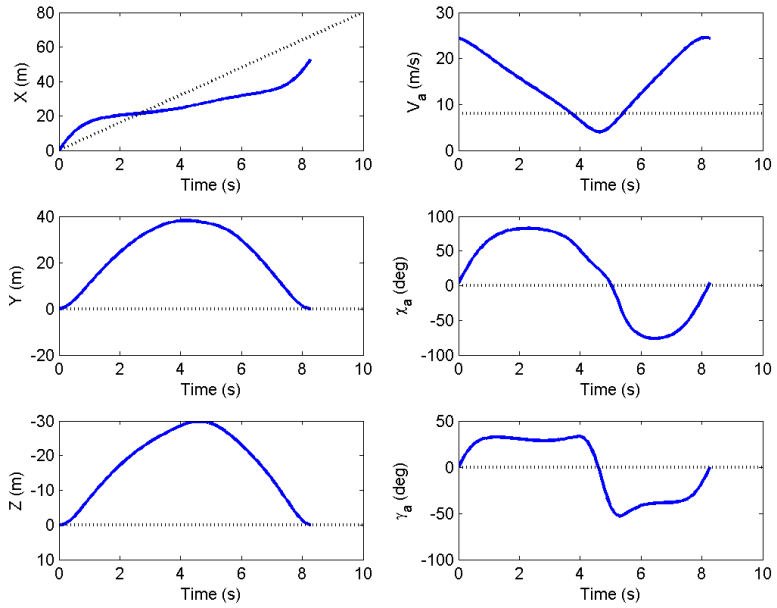


Figure 5.4: Guidance Example States/Inputs Set 1

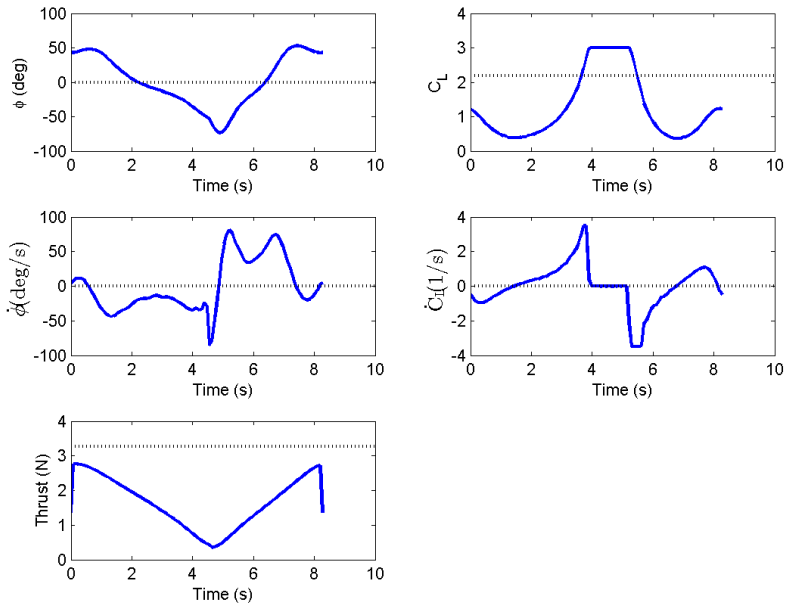


Figure 5.5: Guidance Example States/Inputs Set 2

www.manaraa.com

Figure 5.3 shows the reduced thrust trajectory for this guidance mission. In this trajectory, the UAS banks and climbs into the wind, peaks in altitude, then dives away from the wind to return to the original heading angle. The flight path angle profile in Fig. 5.4 shows the aircraft in a steady climb as it gains altitude between 1 s and 4 s. At around 4.5 s, as the aircraft peaks in altitude the flight path angle becomes negative as the aircraft dives back down.

Figure 5.4 shows the lift coefficient reaching a maximum at the peak of the trajectory and a minimum as the UAS starts to climb into the wind. The maximum lift coefficient plateaus at the upper bound of 3. Conversely, the airspeed reaches a maximum at the lowest altitude and reaches its minimum at the peak altitude of the trajectory. Notice that the airspeed changes linearly during the climb and dive phases of the trajectory.

Figure 5.5 shows the reduced thrust profile with a maximum thrust less than the fixed 3.3 N from the baseline trajectory. The cost of the reduced thrust guidance trajectory is 1.6x10^4. The maximum thrust occurs at the lowest altitude and the minimum thrust occurs at the peak altitude. Note that the thrust and airspeed profiles resemble one another throughout the trajectory.

5.4.2 Loiter Example

For the loiter mission the baseline trajectory banks around a 20 m circular orbit in 10 s, with a steady thrust of 3.6 N. The cost of the baseline trajectory is 6.6x10^4. The wind gradient for the linear boundary wind model is $\mu = 0.1s^{-1}$.

www.manaraa.com

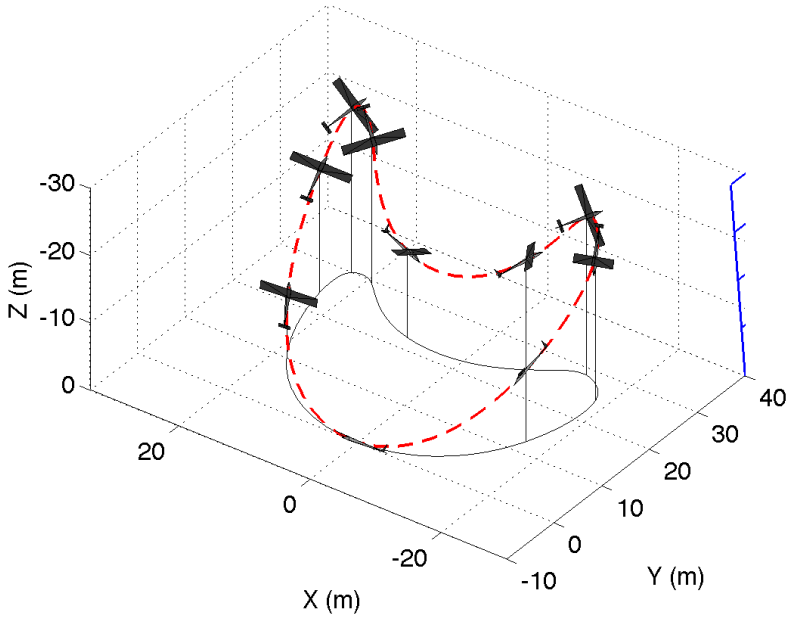


Figure 5.6: Loiter Example 3D Trajectory ($k_T = 100$, $k_p = 0$, $\mu = 0.1\ s^{-1}$)

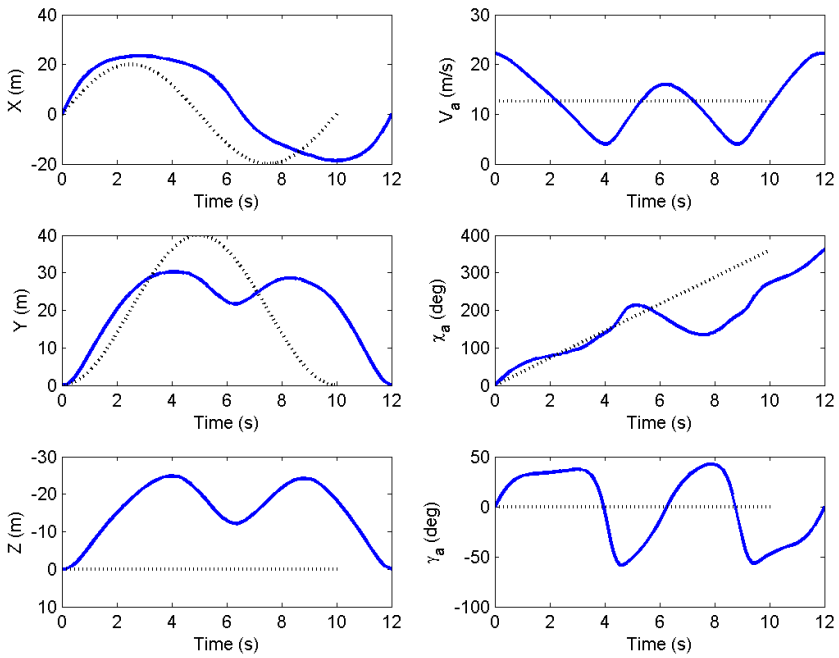


Figure 5.7: Loiter Example States/Inputs Set 1

www.manaraa.com

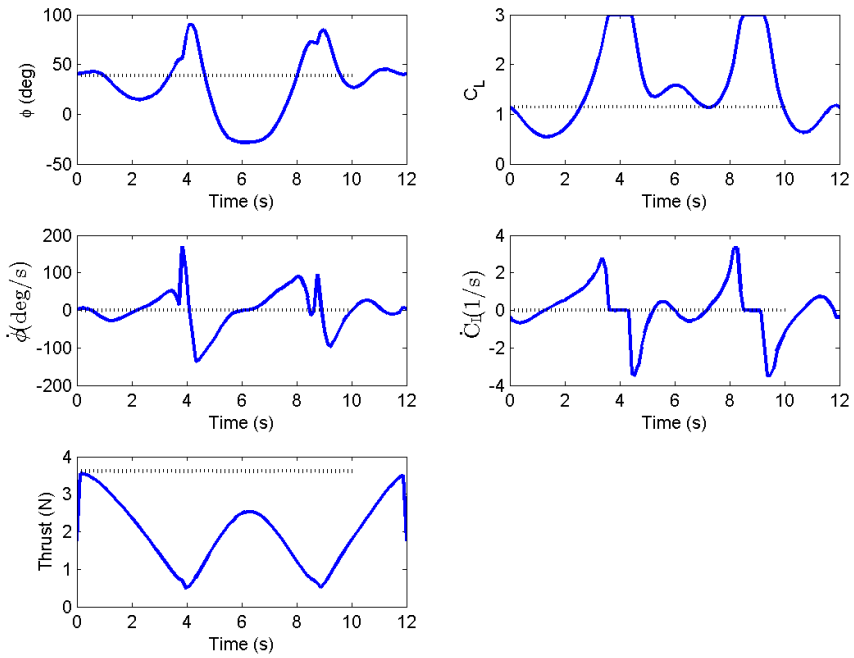


Figure 5.8: Loiter Example States/Inputs Set 2

Figure 5.6 shows the reduced thrust trajectory developed for this loiter mission. The UAS banks and climbs into the wind until it peaks in altitude at 4 s. It performs a shallow dive and then climbs once more facing into the wind. After peaking in altitude for the second time, it banks and dives away from the wind to return to the start of the trajectory.

The flight path angle shows the similar behavior of climbing into the wind and diving away from the wind seen for the guidance case. The thrust profile in Fig. 5.8 shows a thrust reduction from the baseline trajectory's constant thrust of 3.6 N. The cost of the optimized loiter trajectory is $2.3\text{x}10^4$. Once again the thrust reaches maximum values at the low altitudes of the trajectory and minimum values at the trajectory peaks. Similar behavior to the guidance soaring case is seen in the airspeed and lift coefficient. The reduced thrust trajectory requires 12 s to complete as opposed to the original 10 s used by the baseline trajectory.

www.manaraa.com

5.4.3 Discussion of Validation Trajectories

Both the guidance and loiter trajectories resemble the dynamic soaring trajectory from Fig. 1.2. The related state profiles help identify properties of dynamic soaring. One such property includes the behavior of the flight path angle. The UAS climbs when flying into the wind, and dives when flying away from the wind. In addition, there is a trade off between kinetic and potential energy that correlates to the airspeed and altitude of the UAS, respectively.

It is also important to note the behavior of the thrust and lift coefficient. The thrust and airspeed profiles closely resembled one another for both guidance and loiter missions . This suggests that propulsive energy and kinetic energy work together in this reduced thrust, dynamic soaring trajectory. The lift coefficient acts in accordance with the potential energy because it reaches its upper bound as the UAS peaks in altitude. This suggests that improved dynamic soaring performance is capable by using an aircraft with a large maximum lift coefficient in order to gain more potential energy.

5.5 Wind Gradient Strength vs. Thrust Use

This section investigates the impact of the wind gradient strength on the ability to reduce thrust use. The cost function is set to $\mathrm{J}(\mathbf{z}) = \sum_{k}^{N-1} \frac{1}{2} k_T T_k^2$ for both guidance and loiter missions. The optimization parameters used in this study are defined in Appendix A A.2.

5.5.1 Guidance

For the guidance mission the sampling region is located in the direction of the x axis with a heading angle of $\chi_d = 0°$. The initial baseline trajectory traverses 80 m along the x axis over a time span of 10 s. Figure 5.9 shows how the varying wind gradient strength affects the thrust of the resulting trajectories.

Figure 5.9 in general shows decreasing thrust trajectories as the wind gradient strength increases for the guidance mission in a linear boundary layer wind environment. However when μ

www.manaraa.com

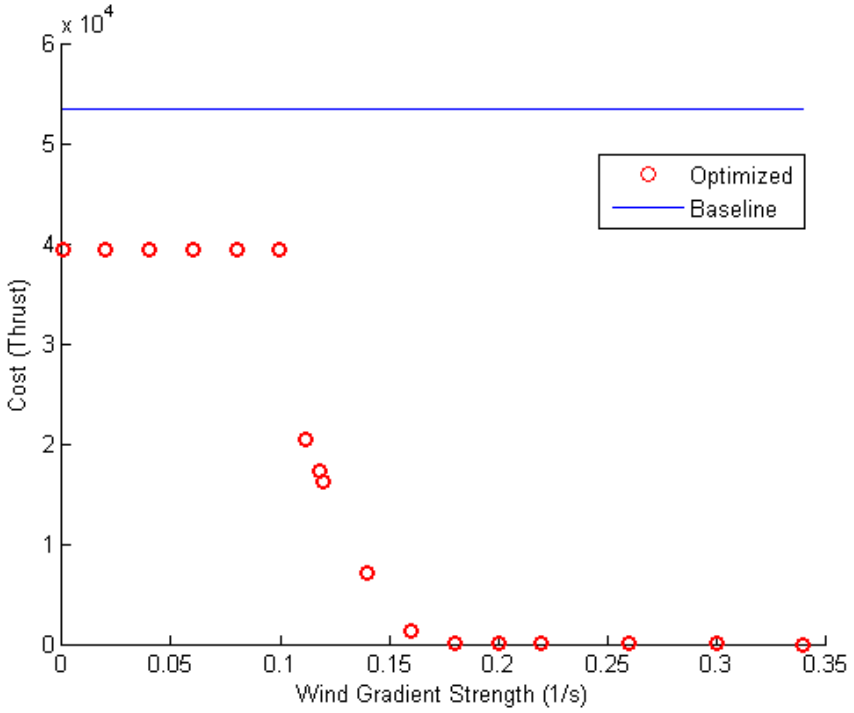


Figure 5.9: Problem G.7 Thrust Component of Cost vs. Wind Gradient Strength

is below a value of 0.1 s^{-1}, the thrust cost plateaus at a fixed value of about 3.9x10^6. Note that even when there is no wind (i.e. $\mu = 0.0\ s^{-1}$), the optimizer outputs a trajectory that reduces the thrust from the initial baseline trajectory. In this plateau region, the trajectories are steady level flight trajectories. When μ is greater than 0.1 s^{-1}, the cost drops as the optimizer outputs soaring trajectories. Once μ surpasses 0.18 s^{-1}, the optimizer outputs zero thrust soaring trajectories .

www.manaraa.com

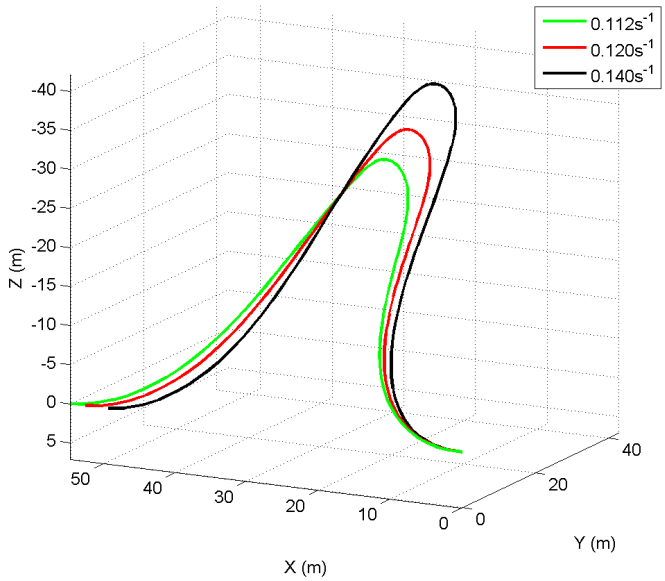


Figure 5.10: Problem G.7 Reduced Thrust Trajectories

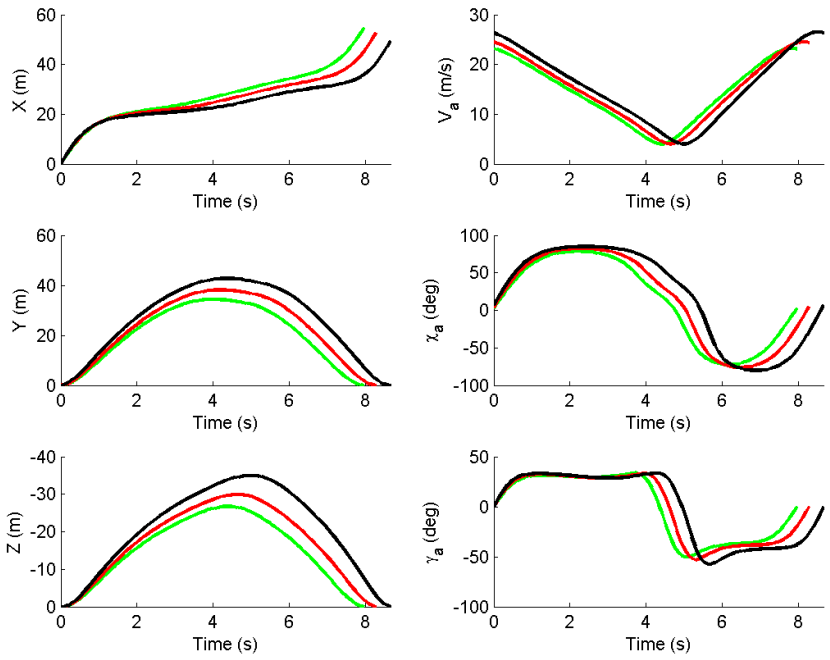


Figure 5.11: Problem G.7 Reduced Thrust States/Inputs Set 1

www.manaraa.com

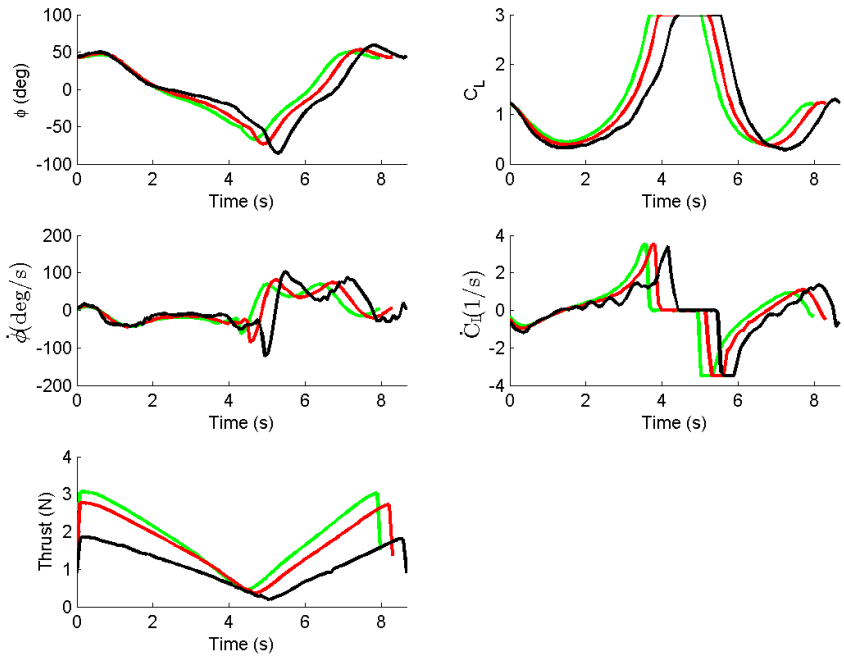


Figure 5.12: Problem G.7 Reduced Thrust States/Inputs Set 2

Figure 5.10 shows some of the trajectories in the reduced thrust region of Fig. 5.9. These trajectories have the same general behavior where the UAS banks and climbs into the wind, peaks in altitude, then turns and dives to the original heading direction. Note that as the wind gradient increases in strength, the trajectories reach higher altitudes.

Figures 5.11 and 5.12 show how the states and control inputs vary with increasing wind gradient strength. Both of these figures show that the profile of each state and control input shifts later in time as the wind gradient strength increases. Note that as the wind gradient increases, the airspeed increases, while the thrust profile decreases. Also, as the wind gradient increases, the UAS spends more time in the trajectory.

www.manaraa.com

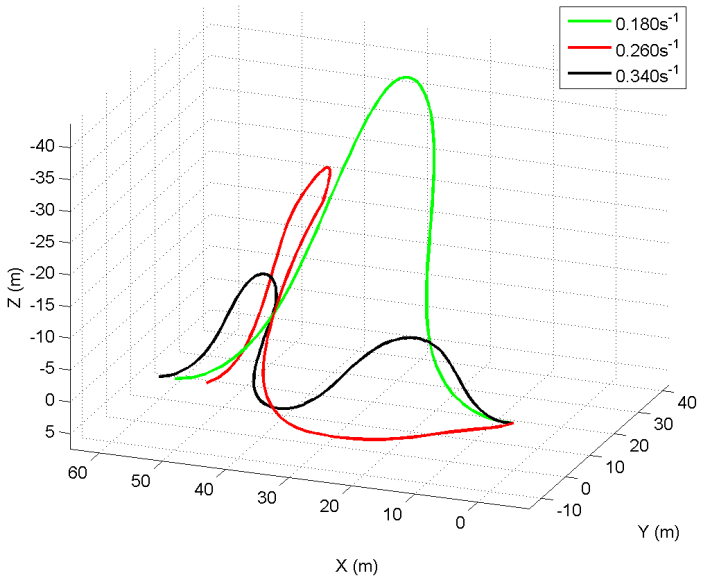


Figure 5.13: Problem G.7 Zero Thrust Trajectories

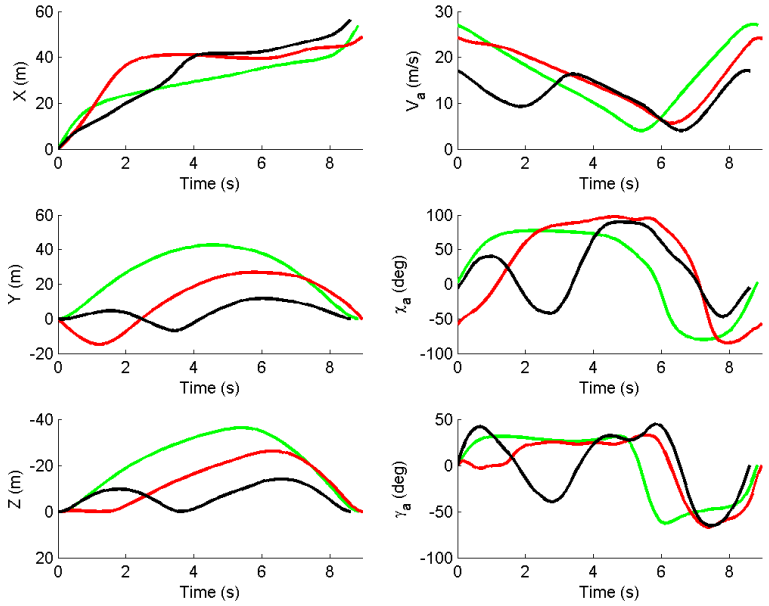


Figure 5.14: Problem G.7 Zero Thrust States/Inputs Set 1

Figure 5.13 shows some zero thrust trajectories from Fig. 5.9. The trajectories shown in Fig. 5.13 have distinct shapes from one another. At $\mu = 0.18s^{-1}$, the trajectory resembles those

www.manaraa.com

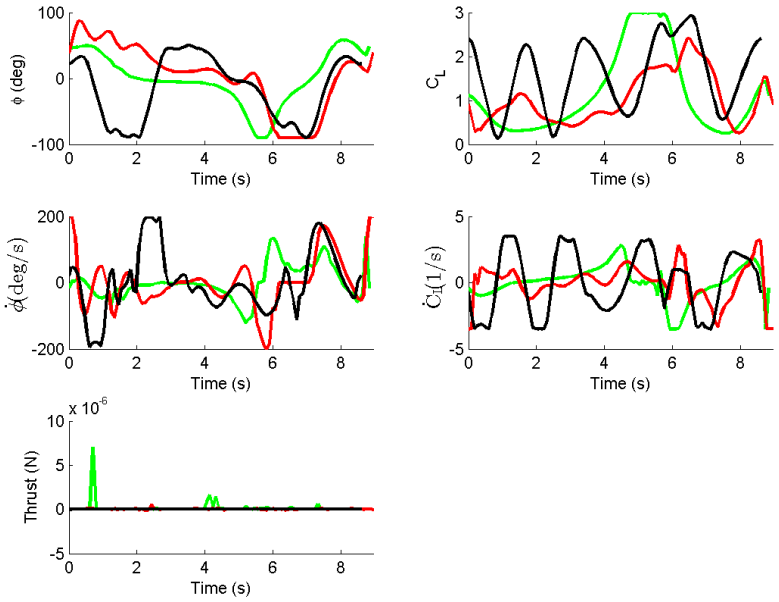


Figure 5.15: Problem G.7 Zero Thrust States/Inputs Set 2

from Fig. 5.10. The trajectory at $\mu = 0.26s^{-1}$ has exaggerated bends and a steep climb and dive phase. The trajectory at $\mu = 0.34s^{-1}$ has two soaring cycles. One general trend between the three trajectories is that as μ increases, the altitude of these zero thrust trajectories decreases.

Figures 5.14 and 5.15 show drastically different states and control inputs for the three zero thrust trajectories. The key point is that the trajectory behavior will be difficult to predict when the wind gradient is strong enough to support zero thrust trajectories.

www.manaraa.com

5.5.2 Loiter

For the loiter mission, the initial baseline trajectory banks in a 20 m radius circular orbit in 10 s. Figure 5.16 shows how the varying wind gradient strength affects the thrust of the resulting trajectories.

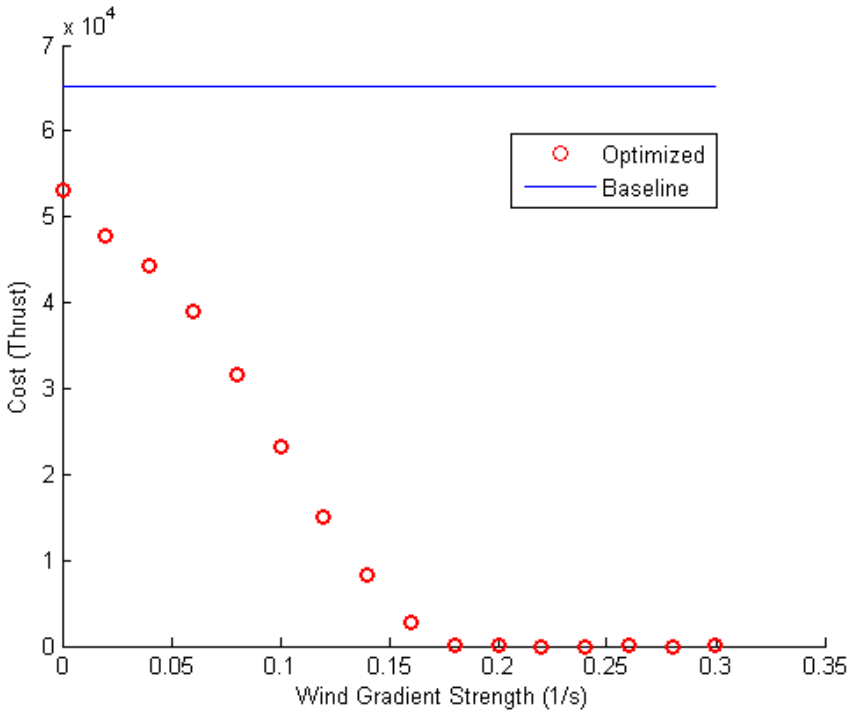


Figure 5.16: Problem S.10 Thrust Component of Cost vs. Wind Gradient Strength

Figure 5.16 shows decreasing thrust trajectories as the wind gradient increases in strength . This trend differs from that of Fig. 5.9 because the thrust does not plateau at a nonzero μ value. Instead, the plot shows that the thrust always decreases as μ increases until the optimized trajectories reach zero thrust. Once again at zero wind ($\mu = 0.0\ s^{-1}$), the optimizer is able to develop a minimized thrust, constant altitude flight trajectory from the baseline trajectory. The optimizer outputs zero thrust trajectories around a wind gradient just above 0.18 s^{-1}, similar to the guidance mission case.

www.manaraa.com

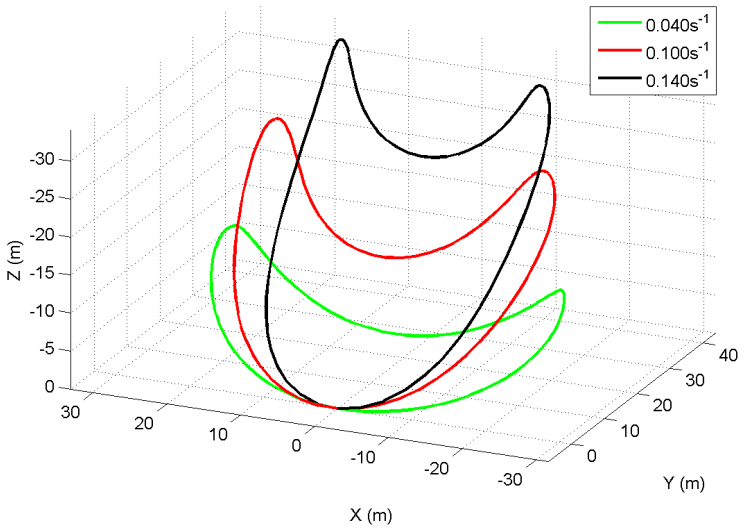


Figure 5.17: Problem S.10 Reduced Thrust Trajectories

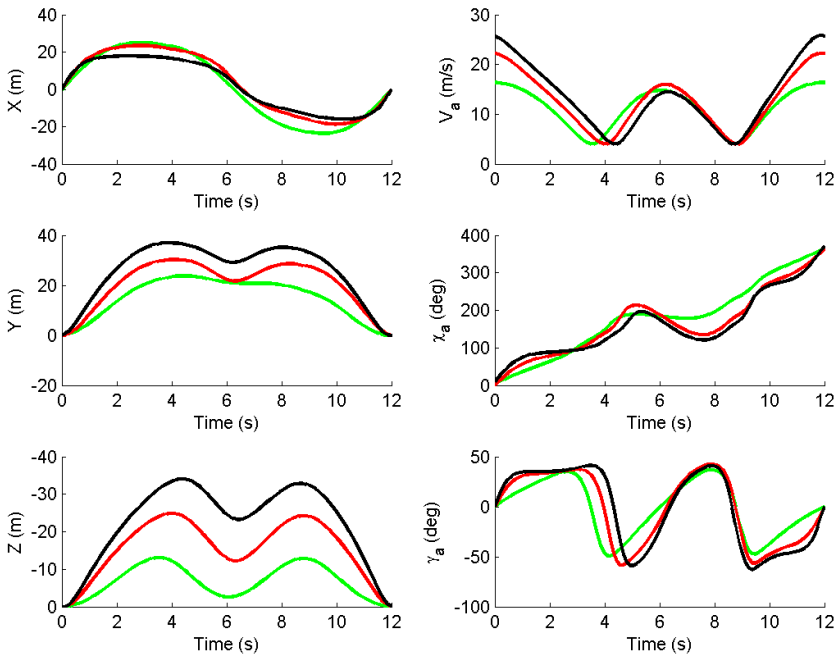


Figure 5.18: Problem S.10 Reduced Thrust States/Inputs Set 1)

www.manaraa.com

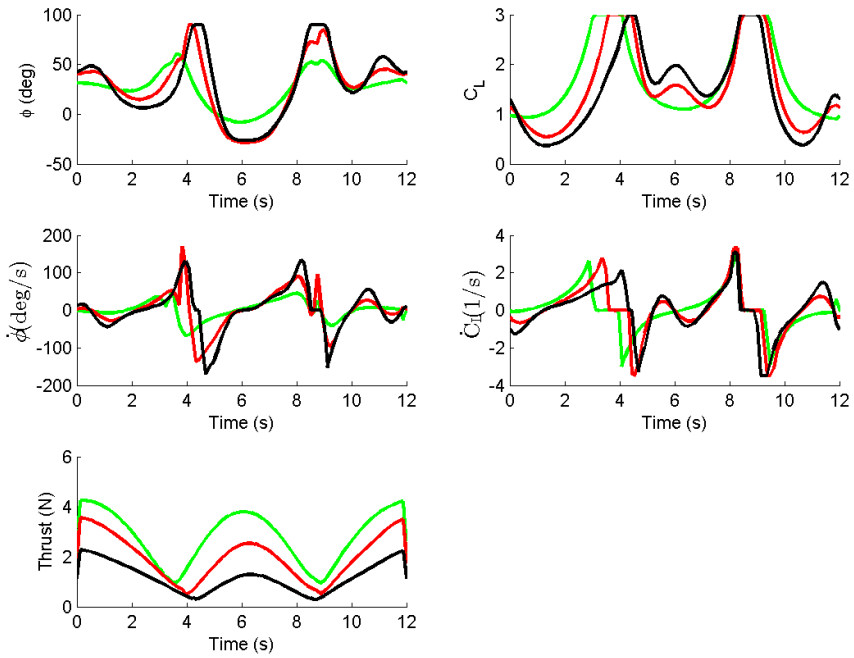


Figure 5.19: Problem S.10 Reduced Thrust States/Inputs Set 2)

Figure 5.17 shows three of the reduced thrust trajectories from Fig. 5.16 . These trajectories have similar shapes to that of Chapter 5.4.2 where the UAS has two climb/dive phases. Similar to the guidance case, the trajectories increase in altitude as the wind gradient strength increases.

Figures 5.18 and 5.19 show similar trends for the three reduced thrust loiter trajectories. As the wind gradient increases in strength, the trajectories not only increase in altitude, but expand in the y direction, while shrink in the x direction. Just as with the reduced thrust trajectories for the guidance mission, there is a direct relation between the airspeed and thrust.

www.manaraa.com

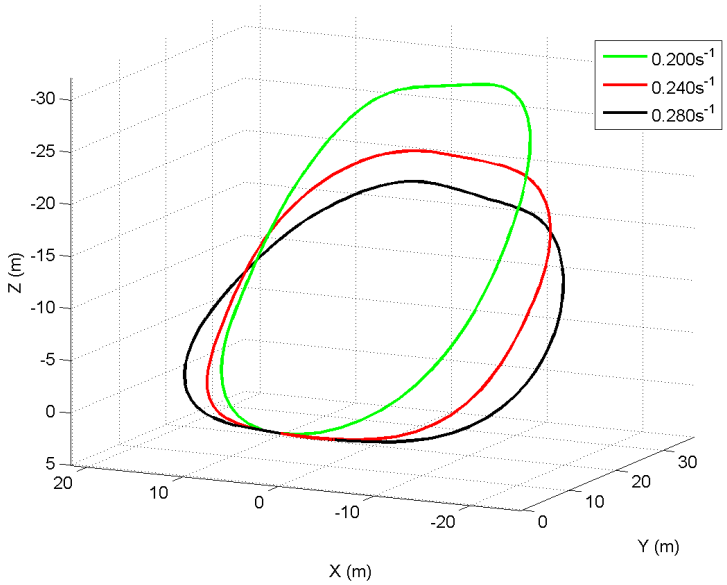


Figure 5.20: Problem S.10 Zero Thrust Trajectories

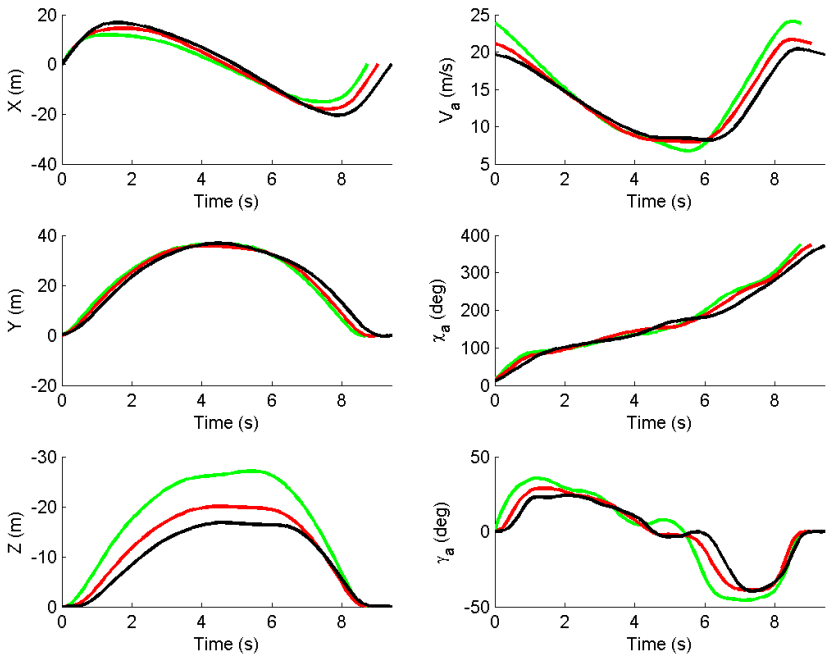


Figure 5.21: Problem S.10 Zero Thrust States/Inputs Set 1

www.manaraa.com

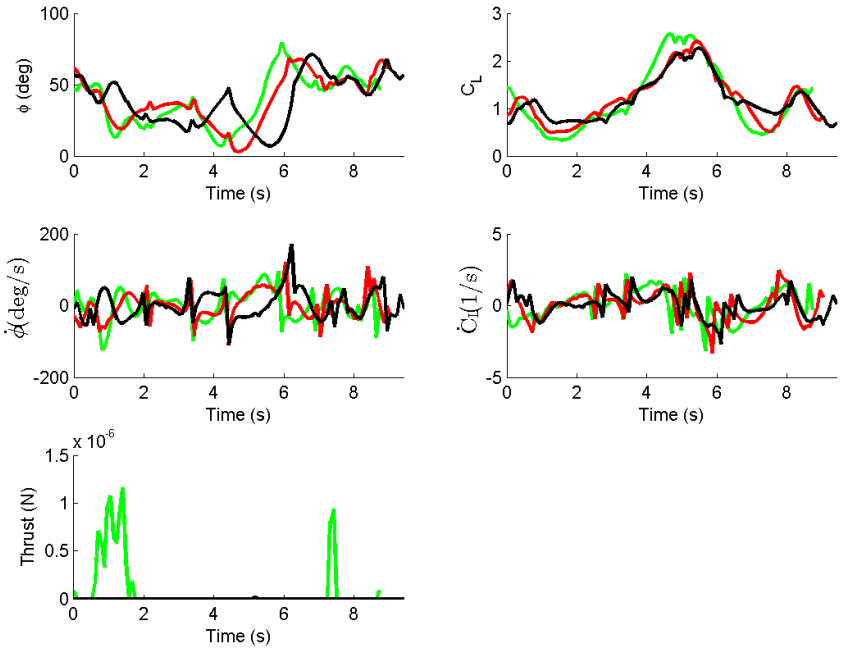


Figure 5.22: Problem S.10 Zero Thrust States/Inputs Set 2

Figure 5.20 shows three zero thrust trajectories from Fig. 5.16. Unlike with the zero thrust guidance trajectories, these zero thrust trajectories share similar behaviors. The UAS starts by banking and climbing into the wind. It then plateaus at a maximum altitude as it turns around,then dives back to the initial position. These zero thrust trajectories do not peak in altitude like the reduced thrust trajectories of Fig. 5.17.

Figures 5.21 and 5.22 show similar trends among the three zero thrust trajectories. As the wind gradient increases in strength, the airspeed actually decreases in airspeed. Note that there is a considerable amount of jitter in the roll rate and lift rate inputs.

5.5.3 Discussion of Wind Gradient Strength vs. Thrust Use

The purpose of this section is to identify how the wind gradient affects the trajectories output from the path planning algorithm. For the guidance mission, it is interesting to note the plateau of thrust below a certain wind gradient strength (Fig. 5.9). This suggests that for these weak wind gradients it is inefficient to deviate from the original heading direction to implement dynamic

www.manaraa.com

soaring. Despite not soaring, the optimizer is still able to find a minimal thrust, constant altitude flight trajectory to maximize the flight endurance. This is also seen for the zero wind case in the loiter mission.

In the loiter mission, the only time the UAS is not soaring is when there is no wind. This suggests that for the loiter case, it is always beneficial to soar so long as there is some wind gradient. This makes sense because due to the circular, periodicity of the loiter trajectory, the UAS will face into and away from the wind at some point in the trajectory. The results show the UAS climb into the wind, and dive away from the wind to perform dynamic soaring. The degree to which the UAS climbs and dives correlates to the strength of the wind gradient. In a reduced thrust trajectory, the UAS performs steeper climbs and dives as the wind gradient increases (Fig. 5.11). Once a zero thrust trajectory is reached, these climbs and dives become shallower.

In both the guidance and loiter missions, the thrust in the trajectories reaches zero once the wind gradient increases past a certain value. These zero thrust trajectories have different shapes than that of the reduced thrust trajectories. However the state profiles resemble the same dynamic soaring properties with the exception of the thrust profile. The different zero thrust soaring trajectories in Fig. 5.13 suggest that there are multiple zero thrust solutions for the same wind gradient. As mentioned the results in this section only focus on reducing the thrust. Therefore the optimizer finds a particular zero thrust trajectory most likely dictated by the initial baseline trajectory, and stops. In such a case it would be beneficial to implement a different problem than Problem G.7 or Problem S.10. By looking at the study of thrust vs. varying wind gradient, the mission tree can determine when the UAS will or will not require thrust inside a certain linear boundary layer wind environment. If zero thrust flight is possible, the first branch in the mission tree can choose a zero thrust problem that minimizes a different cost function. This allows the mission performance to improve by, for example, maximizing the exploration of the UAS to improve data collection inside the sampling region.

In addition to helping design the mission, the study of thrust vs. wind gradient can be used to determine the desired aircraft for flying in a given wind environment. Some of the zero

www.manaraa.com

thrust trajectories contain sharp bends or changes in direction. These sharp bends correspond to large wing loading on the aircraft. Neither Problem G.7 nor Problem S.10 implement a wing loading constraint. By looking at the resulting trajectories output by the path planning algorithm, a mission designer can predict the wing loading a UAS will experience in a given linear boundary layer wind model. As a result, the designer can choose the appropriate aircraft to follow a desired soaring trajectory.

www.manaraa.com

5.6 Effect of Position Gain on Loiter Mission

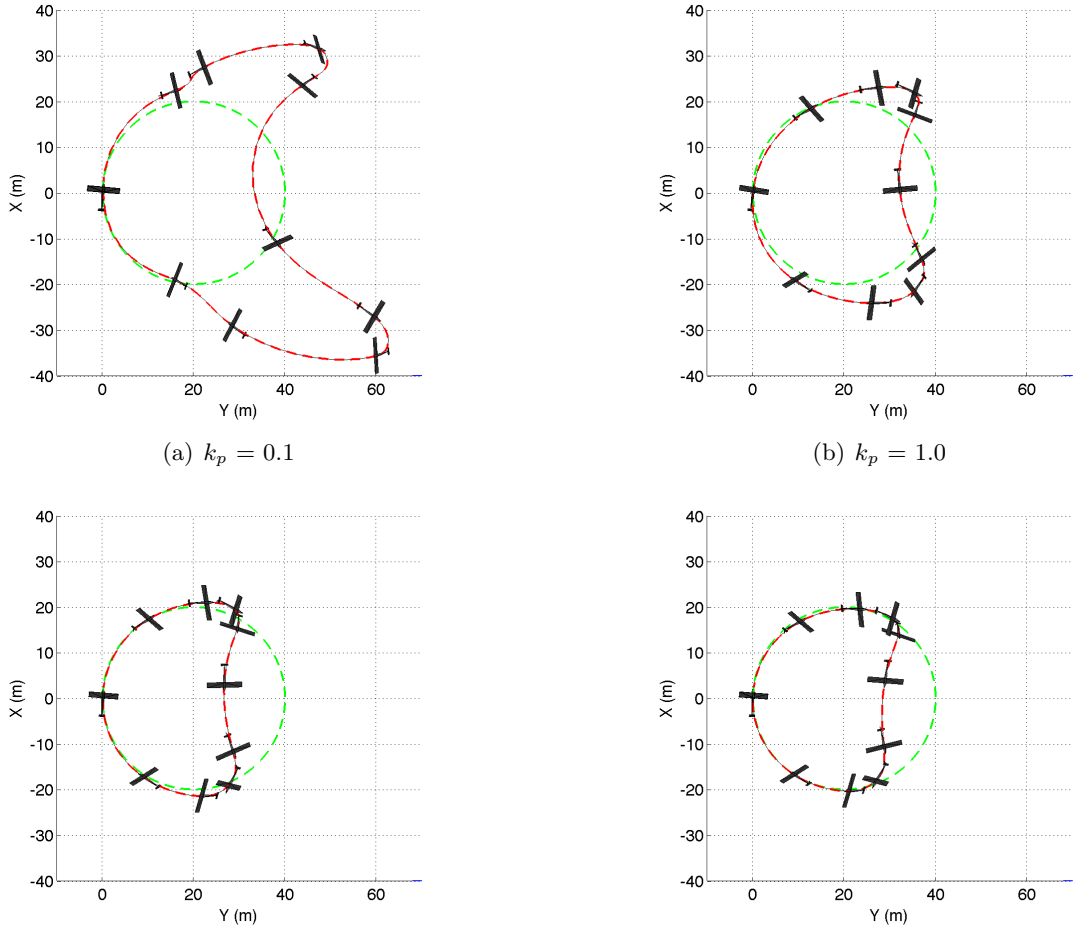
This section investigates the use of the penalty function in tuning between performing the loiter mission and maximizing flight endurance. The penalty function is tested by adjusting the position gain in Problem S.10 given a linear boundary layer wind model. The optimization parameters used to develop the following results are found in Appendix A.3.

5.6.1 Demonstration of Penalty Function

In this example, Problem S.10 is formulated to minimize the thrust and penalize trajectories that leave the 20 m radius sampling region. The given linear boundary layer wind model has a strength of $\mu = 0.07s^{-1}$. Problem S.10 is solved for $k_p = 0.1, k_p = 1, k_p = 10$, and $k_p = 100$. The optimization parameters used in this problem are displayed in Appendix A.3.1.

Figure 5.23 shows the output trajectory shrinking into the sampling region defined by the green dashed circle as k_p increases. Figure 5.23(a) shows the optimized trajectory with a low k_p gain where the optimizer chooses to leave the sampling region to reduce thrust use. As k_p increases, the UAS trajectory spends more of the trajectory inside the sampling region.

www.manaraa.com



(a) $k_p = 0.1$

(b) $k_p = 1.0$

(c) $k_p = 10$

(d) $k_p = 100$

Figure 5.23: Problem S.10: Demonstration of Penalty Function

Figure 5.24 and 5.25 shows how the tuning of the position gain affects the output trajectory. One of the most dramatic effects is in decreasing the trajectory time by half. The thrust profile in Fig. 5.25 for $k_p = 100$ is similar in shape to that of $k_p = 0.1$, but the trajectory at $k_p = 100$ requires more thrust for the same time span. It is interesting to note that many of the states including the altitude, airspeed, and flight path angle share similar profiles despite shrinking the $k_p = 100$ trajectory surface coverage into the sampling region.

www.manaraa.com

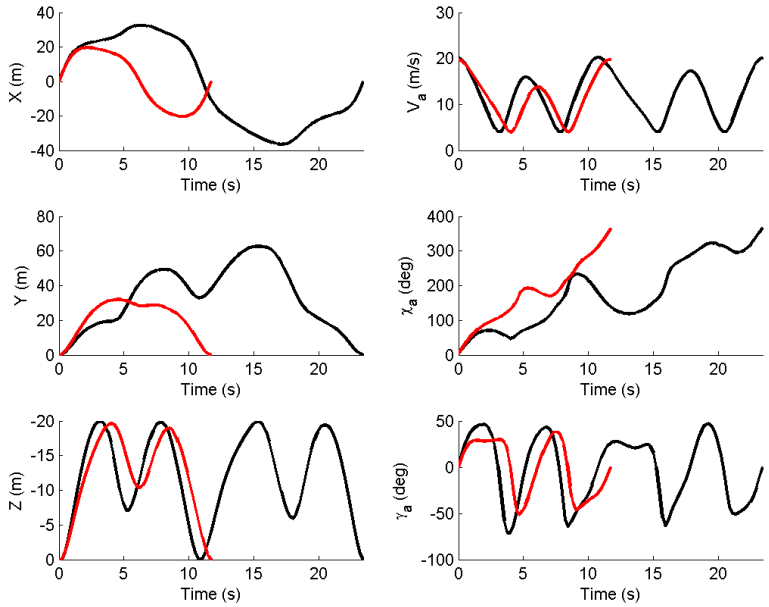


Figure 5.24: Demo of penalty function States/Inputs Set 1, $k_p = 0.1$ (black), k_p= 100 (red)

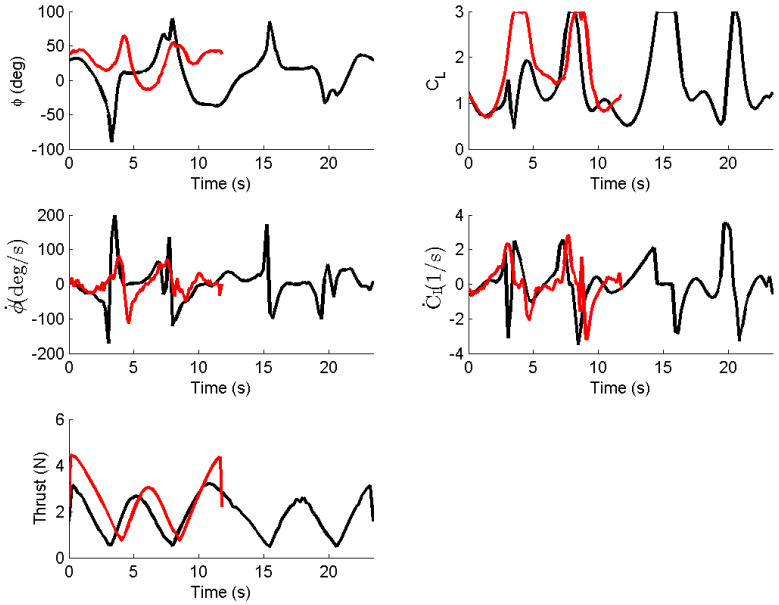


Figure 5.25: Demo of penalty function States/Inputs Set 2, $k_p = 0.1$ (black), k_p= 100 (red)

5.6.2 Thrust Reduction vs. Loiter Mission

This simulation assesses how the penalty function affects the total thrust use and amount of time the UAS spends outside of the 20 m radius sampling region. The optimization parameters

www.manaraa.com

used to develop these results are found in Appendix A.3.2.

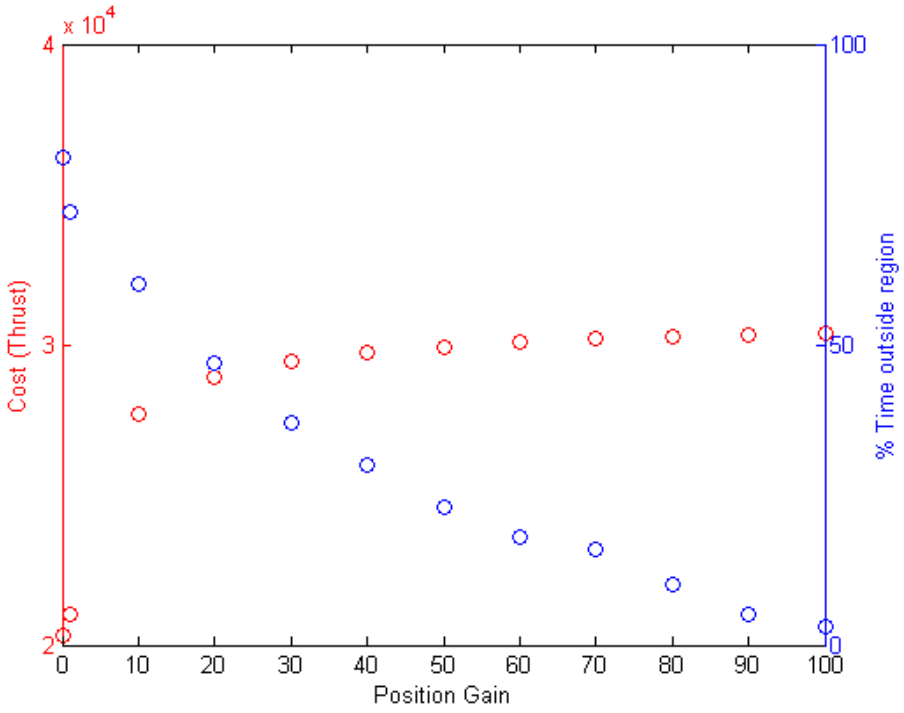


Figure 5.26: Problem S.10 Cost vs. Position Gain ($k_T = 100$, $\mu = 0.1\ s^{-1}$)

Figure 5.26 shows how the position gain affects the thrust and time spent outside of the sampling region. The thrust component of the cost (in red) increases as the position gain increases. The percentage of time outside the sampling region (in blue) decreases as the position gain increases. For the given scenario, the aircraft endurance improves if the UAS is allowed outside of the sampling region. However as the plot shows, there is a trade off between increasing flight endurance and staying inside the sampling region.

www.manaraa.com

5.6.3 Discussion of Position Gain on Loiter Mission

In this section, the position gain is used to balance between performing the sampling mission and increasing the flight endurance of the UAS in the loiter mission. As shown in Fig. 5.26, the larger the position gain, the more time the UAS stays inside the sampling region at the price of increased thrust. This is expected due to the penalty function in Problem S.10 that penalizes trajectories that leave the sampling region. As the position gain increases, more emphasis is placed to minimize this penalty and the resulting trajectories favor performing the sampling mission. As a result, the penalty function implementation in the cost of Problem S.10 allows for mission versatility. The penalty function allows the mission designer to tune between increasing flight endurance and the sampling region instead of having to define a new branch and problem formulation in the mission tree.

The position gain study presented in Fig. 5.26 can be used by mission designers in accordance with the thrust vs. wind gradient studies presented in Chapter 5.5. This study of the position gain provides a trade off between thrust use and amount of time the UAS spends inside the sampling region. If given a thrust-to-fuel model, the designer can determine the amount of time the UAS can stay aloft in the loiter trajectory. One important scenario to consider is when the thrust vs. wind gradient study predicts that a zero thrust trajectory is possible in a certain linear boundary layer wind environment. The UAS will only lose fuel from control surface actuation and non-flight related power consumption. In this scenario, it may be beneficial to set the position gain to zero. This way the amount of time the UAS spends inside the sampling region in one loiter cycle may be reduced, but the total time spent in this region may be maximized due to the total amount of time the UAS stays aloft.

www.manaraa.com

5.7 Trajectory Stitching

This section demonstrates how the guidance and loiter trajectories for a given problem can be stitched together to form a full sampling and surveillance trajectory. In this scenario, Problem G.7 and Problem Gt.7 are used to develop the guidance trajectory to take the UAS from the origin to the sampling region at ($x = 110$, $y = 0$) m. Once this position is reached, Problem S.10 is solved to keep the UAS inside the 20 m sampling region. The parameters used to develop this stitching trajectory are found in Appendix A.4. Figure 5.27 shows the resulting stitched trajectory.

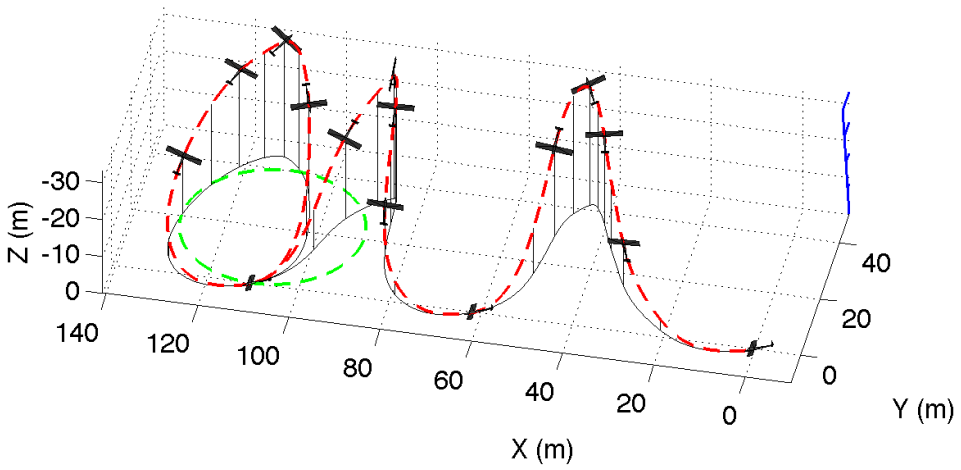


Figure 5.27: Myopic Trajectory Stitching (μ= 0.2 s^{-1})

www.manaraa.com

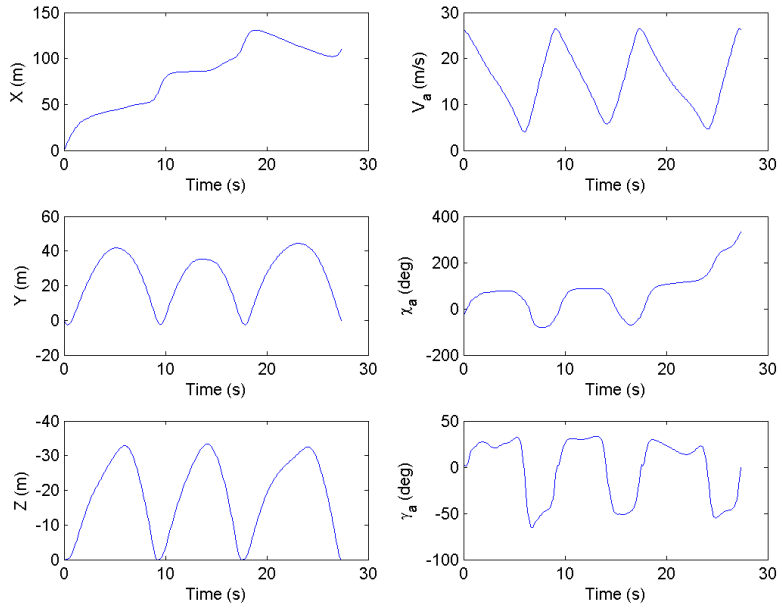


Figure 5.28: Myopic Stitching States/Inputs Set 1 (μ= 0.2 s^{-1})

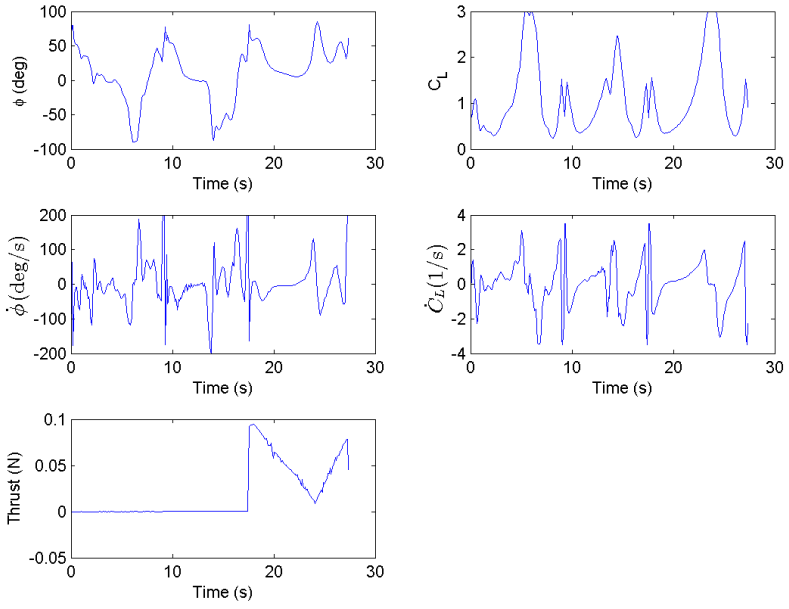


Figure 5.29: Myopic Stitching States/Inputs Set 2 (μ= 0.2 s^{-1})

www.manaraa.com

In Fig. 5.27, the guidance mission, Problem G.7 is first solved and takes the UAS from (0,0,0) m towards the sampling region at (110,0,0) m. The final position of the UAS after this first trajectory is (61.4,0,0) m. The next segment is the transition trajectory that takes the UAS from (61.4,0,0) m to the sampling region at (110,0,0) m. This transition problem is used to transition to the loiter mission. Finally, the loiter mission Problem S.10 is solved to try to keep the UAS inside a 20 meter radius sampling region shown by the green dashed circle. However Problem S.10 includes extra constraints to force the initial and final states/inputs to match the final states/inputs of the previous guidance missions. This allows for the different trajectories to be stitched together into a continuous long term trajectory as shown in Fig. 5.27.

The resulting state profiles of the stitched trajectory are shown in Fig. 5.28. The two guidance trajectories span the time from $t = 0$ s to t= 17.4 s. The time from 17.4 s to 27.3 s is the loiter trajectory. Note that the two guidance trajectories are zero thrust trajectories, but the loiter mission requires thrust.

5.7.1 Discussion of Myopic Stitching

The stitched trajectory presented in Fig. 5.29 shows that the loiter mission requires thrust in the given linear boundary wind model. It is important to note that according to the thrust vs. wind gradient study from Chapter 5.5 there is a possible zero thrust loiter trajectory in the given linear boundary layer environment. However due to forcing the loiter trajectory to match the guidance trajectory states, the resulting loiter mission requires thrust. This suggests the proposed myopic stitching approach is sub optimal. The purpose of the guidance mission is to increase the flight endurance for the loiter mission. However the proposed myopic solution favors the guidance mission instead of the loiter mission. A different stitching approach should be used to not hinder the performance of the loiter mission. Another approach involves developing an unforced loiter and guidance trajectory. This allows the optimizer to find the best soaring trajectories for both the loiter and guidance trajectories. Then the transition trajectory can be developed to transition between the final state of the guidance trajectory to the initial state of the loiter trajectory. This

www.manaraa.com

way the loiter trajectory can be repeated with the minimal thrust needed to soar.

5.8 Thermal Soaring

This section assesses the optimized trajectories developed for a thermal wind model. The optimization parameters used in these simulations are found in Appendix A.5.

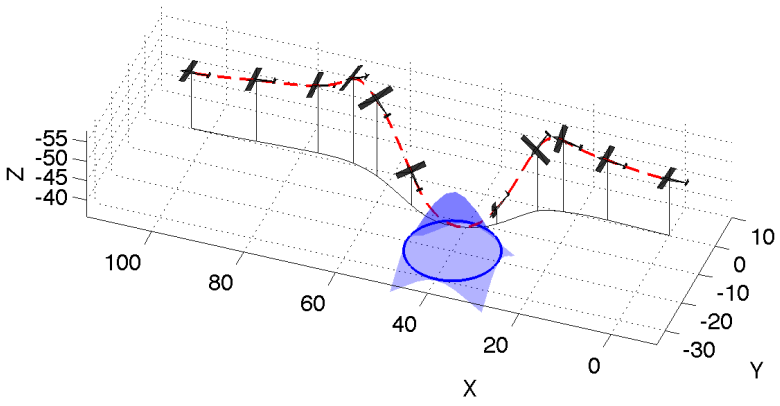


Figure 5.30: Problem G.7 Thermal Soaring Example (V_{core} = 6 m/s)

www.manaraa.com

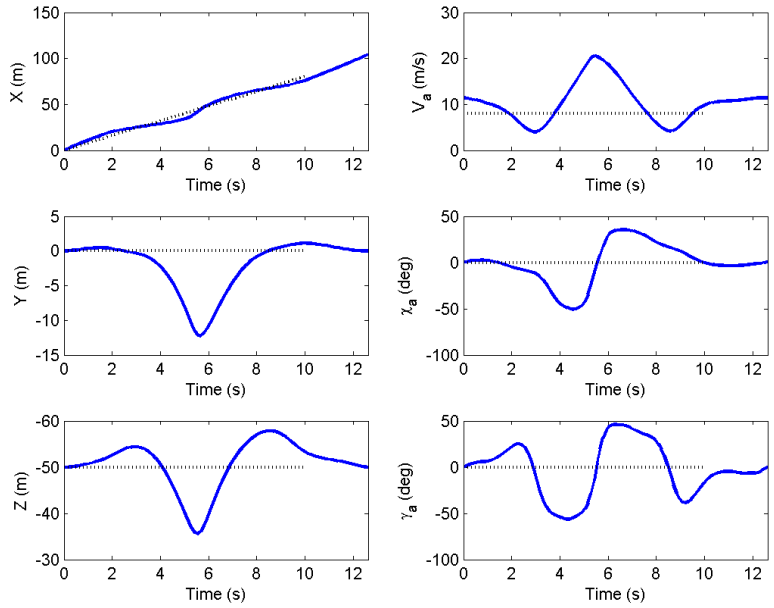


Figure 5.31: Problem G.7 Thermal Soaring States/Inputs Set 1 (V_{core} = 6 m/s)

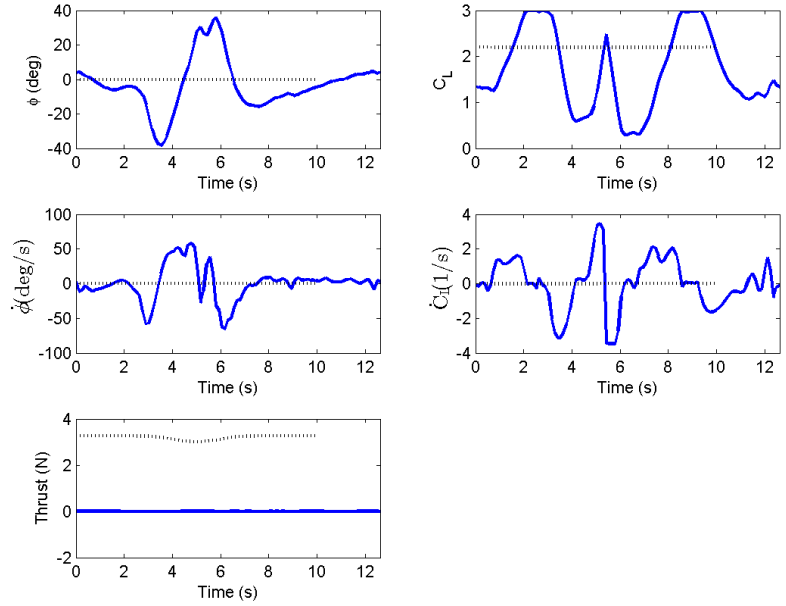


Figure 5.32: Problem G.7 Thermal Soaring States/Inputs Set 2 (V_{core} = 6 m/s)

www.manaraa.com

Figure 5.30 shows the UAS extracting energy from a thermal for a zero thrust guidance trajectory. The trajectory in Fig. 5.30(a) shows the UAS deviate from its heading to dive into and climb out of the thermal. The UAS then regains its original course. As shown in Fig. 5.30(b) the airspeed drops just prior to the UAS diving into the thermal. The airspeed peaks as the UAS finishes its dive, then drops again as the UAS levels out after the climb to its original heading direction. The lift coefficient hits its upper bound just prior to the UAS diving into the thermal and as the UAS climbs out of the thermal. There is an additional peak as the UAS finishes its dive. Note that Problem G.7 still implements periodic soaring constraints that force the optimizer to return the UAS to the original z position.

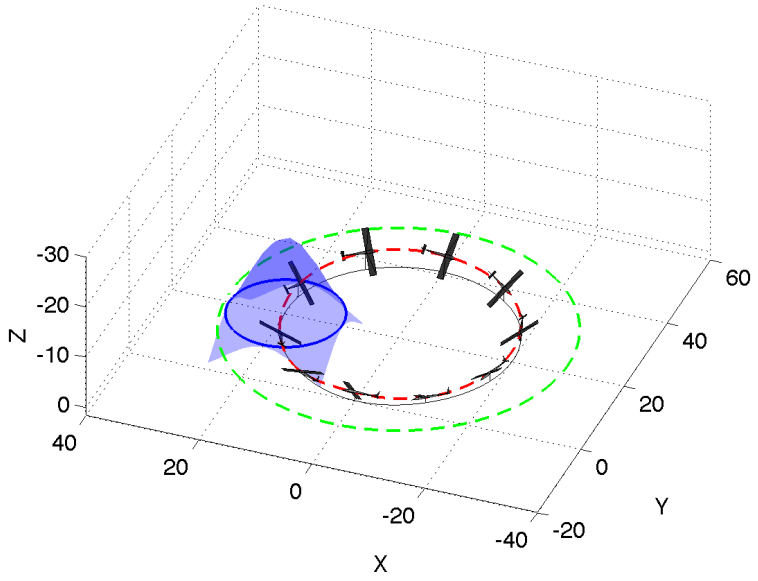


Figure 5.33: Problem S.10 Thermal Soaring Example ($V_{core} = 6$ m/s)

www.manaraa.com

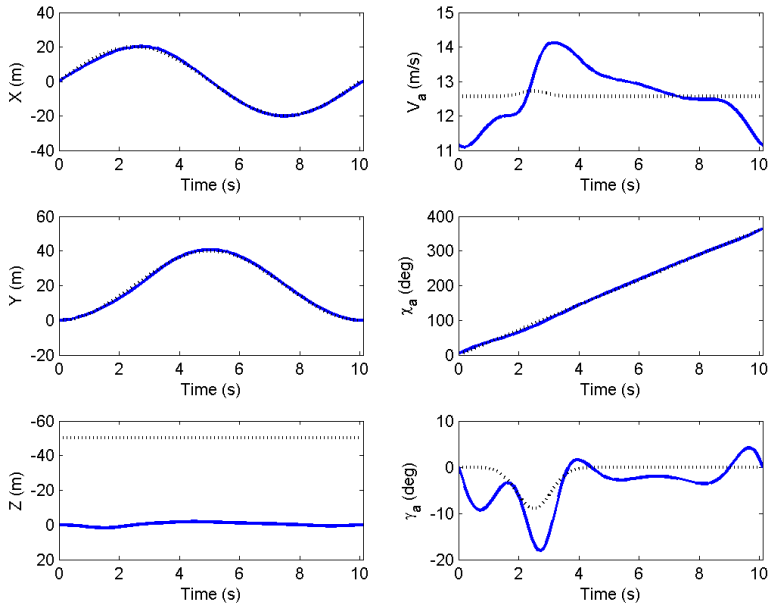


Figure 5.34: Problem S.10 Thermal Soaring States/Inputs Set 1 (V_{core} = 6 m/s)

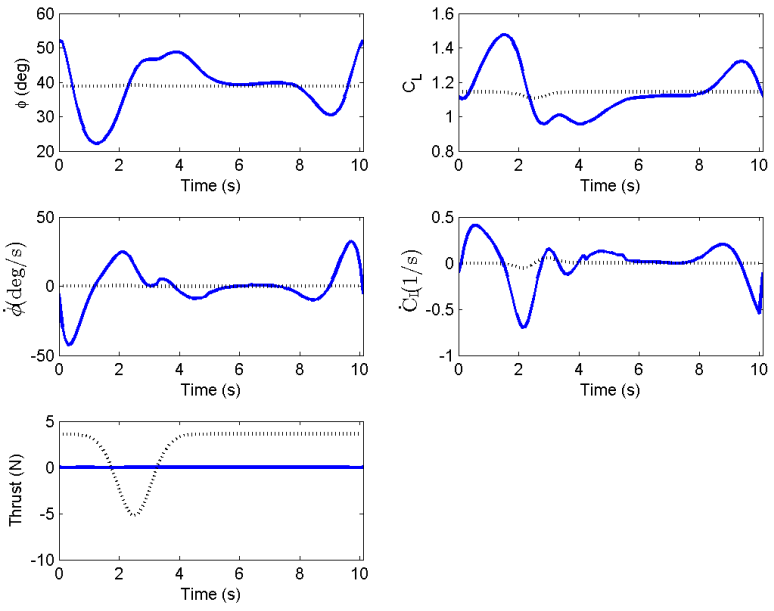


Figure 5.35: Problem S.10 Thermal Soaring States/Inputs Set 2 (V_{core} = 6 m/s)

www.manaraa.com

Figure 5.33 shows the UAS extracting energy from a thermal for a zero thrust loiter trajectory. The trajectory in Fig. 5.33 shows the UAS flying in a circular orbit that passes through a thermal. Figure 5.34 shows the airspeed reaching a maximum as the UAS leaves the thermal. The lift coefficient peaks as the UAS enters the thermal. Also, note the roll angle that drops as the UAS enters the thermal and peaks as the UAS leaves the thermal.

5.8.1 Discussion of Thermal Soaring

So far all the results apply the path planning algorithm to a linear boundary layer wind model. The purpose of this section is to show that the path planning algorithm can develop trajectories in other wind models as well. The results show the algorithm developing zero thrust trajectories for the guidance and loiter missions in a thermal wind environment.

In the guidance trajectory, the results show the UAS deviating from its original heading direction to dive into the thermal. The UAS then climbs out of the thermal to return to the original heading direction towards the sampling region. The UAS however does not gain altitude due to the periodic constraint on z that forces the UAS to return to its original z position. This suggests that the resulting trajectory is not a static soaring trajectory, but a dynamic soaring trajectory. The reason for dynamic soaring is that the thermal is itself composed of a wind gradient that increases the vertical wind speed as the UAS moves closer to the thermal center, and decreases the vertical wind speed as the UAS moves away from the thermal center. The difference with the linear boundary layer case is that instead of a lateral wind field, the wind is vertical. This result of performing dynamic soaring inside a thermal suggests that Problem G.7 prefers dynamic soaring over static soaring. This result may be influenced by the periodic constant on z in addition to the time limit on Δt. The upper bound on the time may prevent longer trajectories where the UAS circles the thermal to gain altitude.

The loiter trajectory result shows that Problem S.10 also prefers dynamic soaring. It is more difficult to see the dynamic soaring directly, but the UAS shows little gain in altitude, which rules out a static soaring trajectory. The behavior of the states in this thermal dynamic soaring

www.manaraa.com

case can be used to demonstrate how dynamic soaring can be implemented in a thermal. In both the guidance and loiter trajectories, the lift coefficient increases prior to entering the thermal, and the airspeed increases as the UAS passes through the thermal. Additionally, the UAS dives when entering the thermal, and climbs when exiting the thermal. Aside from this dynamic soaring behavior, it would be beneficial to determine how to influence static soaring to then compare when dynamic or static soaring is beneficial in a thermal wind environment.

5.9 Limitations

The main limitations with this path planning algorithm include the high degree of freedom in formulating the optimization problem and the computational effort required to develop soaring trajectories. The mission tree component of the path planning algorithm requires a full problem definition including cost, constraints, bounds, initial conditions which involves determining values for a variety of gains, tolerances, and other parameters. This high degree of freedom in the problem formulation is difficult to implement autonomously and still guarantee soaring trajectories similar to those displayed in the results section.

The results were developed using a Macbook Pro with 2.6GHz intel i7 processor with 8GB of RAM. The solutions presented were developed within a time range of 1s to 2 min. In order to implement this algorithm online, the soaring segments need to be developed in less time than it takes to execute a soaring segment. This way the UAS will have a soaring segment to follow as soon as it completes the current soaring segment. The current trajectory development time range does not allow for the system to be implemented online. The path planning algorithm needs to be improved to quickly and robustly develop soaring trajectories.

www.manaraa.com

Chapter 6

Conclusion

6.1 Summary of Findings

In this work, a path planning algorithm was presented to tractably solve the long horizon problem of maximizing the flight endurance of a UAS, while performing a persistent sampling and surveillance mission. In Chapter 1 the motivation for developing persistent sampling and surveillance trajectories was presented. In addition, an overview of the path planning algorithm was provided.

In Chapter 2, the equations of motion for an aircraft in wind were derived. The states and control inputs were defined for a 3D point mass model of an aircraft. The sampling and surveillance mission was defined and broken down into guidance and loiter missions.

In Chapter 3, the nonlinear program was constructed to solve the nonlinear optimal control problem of developing soaring trajectories. The commercial optimizer SNOPT was introduced as the solver used in the presented path planning algorithm. The initial trajectory development approach was presented to show to develop initial baseline trajectories as a function of a desired inertial path and its first and second derivatives. The periodic constraints were presented for developing periodic soaring segments.

In Chapter 4, the guidance and loiter trees were presented as decision trees to choose appropriate optimization problems to solve. The organization of the trees was presented to show the flow of how a particular problem can be chosen depending on the mission objectives and environment. One problem from each tree was chosen to use in the results section. Each problem was defined

www.manaraa.com

with explanations for the costs and constraints.

In Chapter 5, the path planning algorithm was analyzed for a linear boundary layer wind model and thermal wind model. Validation trajectories were shown to demonstrate the output of the path planning algorithm for both guidance and loiter missions in a linear boundary layer wind model. The properties of dynamic soaring trajectories were then identified based on the behavior of the aircraft's state throughout the trajectory.

The effect of the wind gradient on aircraft thrust was discussed. The results show that under certain weak wind gradients, it is more beneficial to follow a steady level flight trajectory for the guidance mission. On the other hand, it is always beneficial to implement dynamic soaring when performing the loiter mission in a linear boundary layer wind field. In both guidance and loiter missions zero thrust trajectories were possible when in sufficiently strong linear boundary layer wind gradients. The study of thrust vs. wind gradient was applied as a sampling and surveillance mission design tool. Examples demonstrated how to use this study to choose an appropriate problem definition in the mission tree, and how to choose an appropriate aircraft for a specific sampling/surveillance mission.

The effect of a position gain on the performance of the sampling/surveillance mission was also discussed. The results show that the use of the penalty function in the defined loiter problem allows for a mission designer to tune between performing the sampling/surveillance mission or maximizing the flight endurance of the UAS. The study of position gain on mission performance offered a trade off between thrust use and time spend outside of the surveillance region for a given linear boundary layer wind model. An example showed how the study of position gain on mission performance coupled with the thrust vs. wind gradient strength study can be used to improve the performance of the overall sampling/surveillance mission for a specific linear boundary layer wind model.

An example of trajectory stitching approach was demonstrated and analyzed. The proposed myopic stitching showed how the guidance and loiter trajectories could be stitched together with an additional transition trajectory to develop a long term sampling/surveillance solution. This stitching approach was shown to be suboptimal and a new approach was suggested to improve

www.manaraa.com

performance in trajectory stitching.

A thermal soaring wind model was used with the path planning algorithm to show that solutions are not specific to a specific wind model. The thermal soaring results show that the path planning algorithm was able to find zero thrust soaring trajectories for both the guidance and loiter missions. However the results showed that both guidance and loiter trajectories were implementing dynamic soaring, and not static soaring, to extract energy from the thermal wind model. An explanation was provided for this phenomena and the properties for dynamic soaring in a thermal wind model were identified.

Finally, the limitations of the presented path planning algorithm were identified. The two main limitations included the high degree of freedom in formulating an optimization problem and the computational effort in developing soaring trajectories. The high degree of freedom limitation deals with the many non-trivial optimization parameters that need to be defined for each soaring segment to be developed. The computational limitation hinders the algorithm from being implemented on an online system.

6.2 Future Work

Future work can be tailored to many aspects of the presented work. The final goal of this work is to use the path planning algorithm online with UAS flight experiments. With this respect, the future work includes exploring efficient trajectory development methods. Some efficient development methods may combine the nonlinear program optimization with a trajectory tracking optimal control algorithm. In this approach, the nonlinear program could be used to quickly develop a coarse soaring trajectory that becomes the initial baseline trajectory for the optimal control algorithm. This optimal control algorithm can be used to quickly develop a refined soaring trajectory.

Other future work can include testing the solutions of the path planning algorithm presented in this thesis with a trajectory tracking controller. The 3D point mass model used in this work uses many assumptions that are invalid for an autopilot system. For example, the aircraft inertias are

www.manaraa.com

ignored, the body frame is assumed to always point into the relative wind, and the lift coefficient rate and roll rate are assumed to be directly controlled. When applied to an actual UAS, the developed trajectory will need to be converted into control surface deflections and throttle actuation. It would be beneficial to then develop a trajectory tracking controller around an autopilot simulation. The trajectory controller will attempt to keep the simulated UAS on the trajectory from the path planning algorithm. The results will then define performance bounds on how well the aircraft model used in this work develops actual UAS soaring trajectories.

www.manaraa.com

Bibliography

[1] SNOPT 7.2. http://www.sbsi-sol-optimize.com/ asp/solproductsnopt.htm, 2011. [Online; accessed 29-Oct-2012].

[2] Effect of types of terrain and clouds on thermal strength. http://www.aerospaceweb.org/question/nature/q0253.shtml, 2012. [Online; accessed 21-Aug-2013].

[3] Zsuzsa Akos, Mate Nagy, Severin Leven, and Tamas Vicsek. Thermal soaring flight of birds and unmanned aerial vehicles. In Bioinspiration and Biomimetics, volume 5, 2010.

[4] Michael J Allen. Guidance and Control of an Autonomous Soaring UAV. In NASA/TM-2007-214611, 2007.

[5] Geoffrey C Bower. Boundary layer dynamic soaring for autonomous aircraft: Design and validation. Department of Aeronautics and Astronautics, Stanford University, 2011.

[6] J.P. Croxall, J.R.D. Silk, R.A. Phillips, V. Afanasyev, and D.R. Briggs. Global circumnavigations: Tracking year-round ranges of nonbreeding albatrosses. 307(5707):249, 2005.

[7] Jason Dangel. Uavs role key ingredient to success in iraq. 2008. [Online; accessed 12-Apr-2013].

[8] Jack Elston, Brian Argrow, Adam Houston, and Eric Frew. Design and validation of a system for targeted observations of tornadic supercells using unmanned aircraft. 2010.

[9] Chen Gao and Hugh H. Liu. Dynamic Soaring Surveillance in a Gradient Wind Field. In AIAA Guidance, Navigation, and Control Conference, 2013.

[10] John Hauser and Rick Hindman. Aggressive flight maneuvers. In Proceedings of the 36th Conference on Decision and Control, San Diego, California, 1997.

[11] Nicholas R J Lawrance and Salah Sukkarieh. A Guidance and Control Strategy for Dynamic Soaring with a Gliding UAV. In IEEE International Conference on Robotics and Automation, volume 34, 2009.

[12] Nikhil Nigam and Ilan Kroo. Persistent surveillance using multiple unmanned air vehicles. 20007.

[13] Jason Roadman, Jack Elston, Brian Argrow, and Eric W. Frew. Mission Performance of the Tempest UAS in Supercell Storms. In AIAA Journal of Aircraft, 2012.

المنارة للاستشارات

www.manaraa.com

[14] G. Sachs. Minimum shear wind strength required for dynamic soaring of albatrosses. In Ibis, volume 147(1):1-10, 2004.

[15] Pengcheng Zhan, Kai Yu, and A. Lee Swindlehurst. Wireless relay communications with unmanned aerial vehicles: Performance and optimization. 2010.

[16] Yiyuan J. Zhao. Optimal patterns of glider dynamic soaring. In Optimal Control Applications and Methods, volume 25:67-89, 2004.

www.manaraa.com

Appendix A

Problem Setup

A.1 Validation Trajectories

Table A.1: Optimization Bounds- Validation Trajectories

Δt_{min}	0.05 s	$V_{a_{min}}$	4.0 m/s	$C_{L_{min}}$	-0.5
Δt_{max}	0.12 s	$V_{a_{max}}$	50.0 m/s	$C_{L_{max}}$	3.0
x_{min}	$-1e20$ m	$\gamma_{a_{min}}$	$-90°$	$\dot{\phi}_{min}$	-200 deg/s
x_{max}	$1e20$ m	$\gamma_{a_{max}}$	$90°$	$\dot{\phi}_{max}$	200 deg/s
y_{min}	$-1e20$ m	$\chi_{a_{min}}$	$-1e20°$	$\dot{C}_{Lmin}$	-200 s^{-1}
y_{max}	$1e20$ m	$\chi_{a_{max}}$	$1e20°$	$\dot{C}_{Lmax}$	200 s^{-1}
z_{min}	$-1e20$ m	ϕ_{min}	$-90°$	T_{min}	0 N
z_{max}	0 m	ϕ_{max}	$90°$	T_{max}	$1e20$ N

www.manaraa.com

A.1.1 Guidance Example Trajectory Data

Table A.2: Optimization Parameters- Guidance Example Trajectory

t_f	10 s	N	100	μ	0.12 s^{-1}
$x(0)$	0.0 m	k_T	100	χ_d	0.0 deg
$y(0)$	0.0 m	k_p	0.0	tol_{opt}	1x10^{-4}
$z(0)$	0.0 m	A_x	80.0 m	tol_{fea}	1x10^{-5}
$\gamma_a(0)$	0.0 deg	A_y	0.0 m	optimal cost	1.6362x10^4
				baseline cost	5.3879x10^4

A.1.2 Loiter Example Trajectory Data

Table A.3: Optimization Parameters- Loiter Example Trajectory

t_f	10 s	N	100	μ	0.10 s^{-1}
$x(0)$	0.0 m	k_T	100	R_d	30.0 m
$y(0)$	0.0 m	k_p	0.0	tol_{opt}	1x10^{-4}
$z(0)$	0.0 m	A_x	20.0 m	tol_{fea}	1x10^{-5}
$\gamma_a(0)$	0.0 deg	A_y	20.0 m	optimal cost	2.3486x10^4
		(x_g, y_g)	(0.0,20.0) m	baseline cost	6.5859x10^4

www.manaraa.com

A.2 Wind Gradient Strength vs. Thrust Use Data

Table A.4: Optimization Bounds- Wind Gradient Strength vs. Thrust Use

Δt_{min}	0.05 s	$V_{a_{min}}$	4.0 m/s	$C_{L_{min}}$	-0.5
Δt_{max}	0.12 s	$V_{a_{max}}$	50.0 m/s	$C_{L_{max}}$	3.0
x_{min}	$-1e20$ m	$\gamma_{a_{min}}$	$-90°$	$\dot{\phi}_{min}$	-200 deg/s
x_{max}	$1e20$ m	$\gamma_{a_{max}}$	$90°$	$\dot{\phi}_{max}$	200 deg/s
y_{min}	$-1e20$ m	$\chi_{a_{min}}$	$-1e20°$	$\dot{C}_{Lmin}$	-200 s^{-1}
y_{max}	$1e20$ m	$\chi_{a_{max}}$	$1e20°$	$\dot{C}_{Lmax}$	200 s^{-1}
z_{min}	$-1e20$ m	ϕ_{min}	$-90°$	T_{min}	0 N
z_{max}	0 m	ϕ_{max}	$90°$	T_{max}	$1e20$ N

www.manaraa.com

Table A.5: Optimization Parameters- Problem G7: Reduced Thrust Trajectories

t_f	10 s	N	100	μ	0.0 s^{-1}
$x(0)$	0.0 m	k_T	100	χ_d	0.0 deg
$y(0)$	0.0 m	k_p	0.0	tol_{opt}	1x10^{-4}
$z(0)$	0.0 m	A_x	80.0 m	tol_{fea}	1x10^{-5}
$\gamma_a(0)$	0.0 deg	A_y	0.0 m	optimal cost	3.9478x10^4
				baseline cost	5.3879x10^4

A.2.1 Problem S10: Zero Thrust Trajectories

Table A.6: Optimization Parameters- Problem G7: Reduced Thrust Trajectories

t_f	10 s	N	100	μ	0.112 s^{-1}
$x(0)$	0.0 m	k_T	100	χ_d	0.0 deg
$y(0)$	0.0 m	k_p	0.0	tol_{opt}	1x10^{-4}
$z(0)$	0.0 m	A_x	80.0 m	tol_{fea}	1x10^{-5}
$\gamma_a(0)$	0.0 deg	A_y	0.0 m	optimal cost	2.0569x10^4
				baseline cost	5.3879x10^4

www.manaraa.com

Table A.7: Optimization Parameters- Problem G7: Reduced Thrust Trajectories

t_f	10 s	N	100	μ	0.12 s^{-1}
$x(0)$	0.0 m	k_T	100	χ_d	0.0 deg
$y(0)$	0.0 m	k_p	0.0	tol_{opt}	1x10^{-4}
$z(0)$	0.0 m	A_x	80.0 m	tol_{fea}	1x10^{-5}
$\gamma_a(0)$	0.0 deg	A_y	0.0 m	optimal cost	1.6362x10^{4}
				baseline cost	5.3879x10^{4}

Table A.8: Optimization Parameters- Problem G7: Reduced Thrust Trajectories

t_f	10 s	N	100	μ	0.14 s^{-1}
$x(0)$	0.0 m	k_T	100	χ_d	0.0 deg
$y(0)$	0.0 m	k_p	0.0	tol_{opt}	1x10^{-3}
$z(0)$	0.0 m	A_x	80.0 m	tol_{fea}	1x10^{-5}
$\gamma_a(0)$	0.0 deg	A_y	0.0 m	optimal cost	7.1137x10^{3}
				baseline cost	5.3879x10^{4}

Table A.9: Optimization Parameters- Problem G7: Zero Thrust Trajectories

t_f	10 s	N	100	μ	0.18 s^{-1}
$x(0)$	0.0 m	k_T	100	χ_d	0.0 deg
$y(0)$	0.0 m	k_p	0.0	tol_{opt}	1x10^{-4}
$z(0)$	0.0 m	A_x	80.0 m	tol_{fea}	1x10^{-5}
$\gamma_a(0)$	0.0 deg	A_y	0.0 m	optimal cost	1.6436x10^{-5}
				baseline cost	5.3879x10^{4}

Table A.10: Optimization Parameters- Problem G7: Zero Thrust Trajectories

t_f	10 s	N	100	μ	0.26 s^{-1}
$x(0)$	0.0 m	k_T	100	χ_d	0.0 deg
$y(0)$	0.0 m	k_p	0.0	tol_{opt}	1x10^{-4}
$z(0)$	0.0 m	A_x	80.0 m	tol_{fea}	1x10^{-5}
$\gamma_a(0)$	0.0 deg	A_y	0.0 m	optimal cost	3.0619x10^{-11}
				baseline cost	5.3879x10^{4}

Table A.11: Optimization Parameters- Problem G7: Zero Thrust Trajectories

t_f	10 s	N	100	μ	0.34 s^{-1}
$x(0)$	0.0 m	k_T	100	χ_d	0.0 deg
$y(0)$	0.0 m	k_p	0.0	tol_{opt}	1x10^{-4}
$z(0)$	0.0 m	A_x	80.0 m	tol_{fea}	1x10^{-5}
$\gamma_a(0)$	0.0 deg	A_y	0.0 m	optimal cost	1.5046x10^{-3}4
				baseline cost	5.3879x10^{4}

www.manaraa.com

Table A.12: Optimization Parameters- Problem S10: Reduced Thrust Trajectories

t_f	10 s	N	100	μ	0.0 s^{-1}
$x(0)$	0.0 m	k_T	100	R_d	30.0 m
$y(0)$	0.0 m	k_p	0.0	tol_{opt}	1x10^{-4}
$z(0)$	0.0 m	A_x	20.0 m	tol_{fea}	1x10^{-5}
$\gamma_a(0)$	0.0 deg	A_y	20.0 m	optimal cost	5.3197x10^4
		(x_g, y_g)	(0.0,20.0) m	baseline cost	6.5859x10^4

Table A.13: Optimization Parameters- Problem S10: Reduced Thrust Trajectories

t_f	10 s	N	100	μ	0.04 s^{-1}
$x(0)$	0.0 m	k_T	100	R_d	30.0 m
$y(0)$	0.0 m	k_p	0.0	tol_{opt}	1x10^{-4}
$z(0)$	0.0 m	A_x	20.0 m	tol_{fea}	1x10^{-5}
$\gamma_a(0)$	0.0 deg	A_y	20.0 m	optimal cost	4.4612x10^4
		(x_g, y_g)	(0.0,20.0) m	baseline cost	6.5859x10^4

Table A.14: Optimization Parameters- Problem S10: Reduced Thrust Trajectories

t_f	10 s	N	100	μ	0.1 s^{-1}
$x(0)$	0.0 m	k_T	100	R_d	30.0 m
$y(0)$	0.0 m	k_p	0.0	tol_{opt}	1x10^{-4}
$z(0)$	0.0 m	A_x	20.0 m	tol_{fea}	1x10^{-5}
$\gamma_a(0)$	0.0 deg	A_y	20.0 m	optimal cost	2.3486x10^4
		(x_g, y_g)	(0.0,20.0) m	baseline cost	6.5859x10^4

Table A.15: Optimization Parameters- Problem S10: Reduced Thrust Trajectories

t_f	10 s	N	100	μ	0.14 s^{-1}
$x(0)$	0.0 m	k_T	100	R_d	30.0 m
$y(0)$	0.0 m	k_p	0.0	tol_{opt}	1x10^{-4}
$z(0)$	0.0 m	A_x	20.0 m	tol_{fea}	1x10^{-5}
$\gamma_a(0)$	0.0 deg	A_y	20.0 m	optimal cost	8.4098x10^3
		(x_g, y_g)	(0.0,20.0) m	baseline cost	6.5859x10^4

Table A.16: Optimization Parameters- Problem S10: Zero Thrust Trajectories

t_f	10 s	N	100	μ	0.2 s^{-1}
$x(0)$	0.0 m	k_T	100	R_d	30.0 m
$y(0)$	0.0 m	k_p	0.0	tol_{opt}	1x10^{-4}
$z(0)$	0.0 m	A_x	20.0 m	tol_{fea}	1x10^{-5}
$\gamma_a(0)$	0.0 deg	A_y	20.0 m	optimal cost	3.8167x10^{-10}
		(x_g, y_g)	(0.0,20.0) m	baseline cost	6.5859x10^4

www.manaraa.com

Table A.17: Optimization Parameters- Problem S10: Zero Thrust Trajectories

t_f	10 s	N	100	μ	0.24 s^{-1}
$x(0)$	0.0 m	k_T	100	R_d	30.0 m
$y(0)$	0.0 m	k_p	0.0	tol_{opt}	1x10^{-4}
$z(0)$	0.0 m	A_x	20.0 m	tol_{fea}	1x10^{-5}
$\gamma_a(0)$	0.0 deg	A_y	20.0 m	optimal cost	0.0
		(x_g, y_g)	(0.0,20.0) m	baseline cost	6.5859x10^4

Table A.18: Optimization Parameters- Problem S10: Zero Thrust Trajectories

t_f	10 s	N	100	μ	0.28 s^{-1}
$x(0)$	0.0 m	k_T	100	R_d	30.0 m
$y(0)$	0.0 m	k_p	0.0	tol_{opt}	1x10^{-4}
$z(0)$	0.0 m	A_x	20.0 m	tol_{fea}	1x10^{-5}
$\gamma_a(0)$	0.0 deg	A_y	20.0 m	optimal cost	1.2894x10^{-14}
		(x_g, y_g)	(0.0,20.0) m	baseline cost	6.5859x10^4

www.manaraa.com

A.3 Effect of Position Gain on Trajectory Development Data

A.3.1 Demonstration of Barrier Function

Table A.19: Optimization Bounds- Demonstration of Barrier Function

Δt_{min}	0.05 s	$V_{a_{min}}$	4.0 m/s	$C_{L_{min}}$	-0.5
Δt_{max}	0.5 s	$V_{a_{max}}$	50.0 m/s	$C_{L_{max}}$	3.0
x_{min}	$-1e20$ m	$\gamma_{a_{min}}$	$-90°$	$\dot{\phi}_{min}$	-200 deg/s
x_{max}	$1e20$ m	$\gamma_{a_{max}}$	$90°$	$\dot{\phi}_{max}$	200 deg/s
y_{min}	$-1e20$ m	$\chi_{a_{min}}$	$-1e20°$	$\dot{C}_{Lmin}$	-200 s^{-1}
y_{max}	$1e20$ m	$\chi_{a_{max}}$	$1e20°$	$\dot{C}_{Lmax}$	200 s^{-1}
z_{min}	$-1e20$ m	ϕ_{min}	$-90°$	T_{min}	0 N
z_{max}	0 m	ϕ_{max}	$90°$	T_{max}	$1e20$ N

A.3.2 Thrust Reduction vs. Loiter Mission

Table A.20: Optimization Parameters- Demonstration of Barrier Function

t_f	10 s	N	100	μ	0.07 s^{-1}
$x(0)$	0.0 m	k_T	100	R_d	20.0 m
$y(0)$	0.0 m	k_p	0.1	tol_{opt}	0.0037
$z(0)$	0.0 m	A_x	20.0 m	tol_{fea}	$1\text{x}10^{-5}$
$\gamma_a(0)$	0.0 deg	A_y	20.0 m	optimal cost	$2.1705\text{x}10^4$
		(x_g, y_g)	(0.0,20.0) m	baseline cost	$6.5859\text{x}10^4$

www.manaraa.com

Table A.21: Optimization Parameters- Demonstration of Barrier Function

t_f	10 s	N	100	μ	0.07 s^{-1}
$x(0)$	0.0 m	k_T	100	R_d	20.0 m
$y(0)$	0.0 m	k_p	1	tol_{opt}	0.0037
$z(0)$	0.0 m	A_x	20.0 m	tol_{fea}	1x10^{-5}
$\gamma_a(0)$	0.0 deg	A_y	20.0 m	optimal cost	3.6837x10^4
		(x_g, y_g)	(0.0,20.0) m	baseline cost	6.5859x10^4

Table A.22: Optimization Parameters- Demonstration of Barrier Function

t_f	10 s	N	100	μ	0.07 s^{-1}
$x(0)$	0.0 m	k_T	100	R_d	20.0 m
$y(0)$	0.0 m	k_p	10	tol_{opt}	0.0037
$z(0)$	0.0 m	A_x	20.0 m	tol_{fea}	1x10^{-5}
$\gamma_a(0)$	0.0 deg	A_y	20.0 m	optimal cost	3.6833x10^4
		(x_g, y_g)	(0.0,20.0) m	baseline cost	6.5859x10^4

Table A.23: Optimization Parameters- Demonstration of Barrier Function

t_f	10 s	N	100	μ	0.07 s^{-1}
$x(0)$	0.0 m	k_T	100	R_d	20.0 m
$y(0)$	0.0 m	k_p	100	tol_{opt}	0.0037
$z(0)$	0.0 m	A_x	20.0 m	tol_{fea}	1x10^{-5}
$\gamma_a(0)$	0.0 deg	A_y	20.0 m	optimal cost	3.7690x10^4
		(x_g, y_g)	(0.0,20.0) m	baseline cost	6.5859x10^4

www.manaraa.com

Table A.24: Optimization Bounds- Thrust Reduction vs. Loiter Mission

Δt_{min}	0.05 s	$V_{a_{min}}$	4.0 m/s	$C_{L_{min}}$	-0.5
Δt_{max}	0.2 s	$V_{a_{max}}$	50.0 m/s	$C_{L_{max}}$	3.0
x_{min}	$-1e20$ m	$\gamma_{a_{min}}$	$-90°$	$\dot{\phi}_{min}$	-200 deg/s
x_{max}	$1e20$ m	$\gamma_{a_{max}}$	$90°$	$\dot{\phi}_{max}$	200 deg/s
y_{min}	$-1e20$ m	$\chi_{a_{min}}$	$-1e20°$	$\dot{C}_{Lmin}$	-200 s^{-1}
y_{max}	$1e20$ m	$\chi_{a_{max}}$	$1e20°$	$\dot{C}_{Lmax}$	200 s^{-1}
z_{min}	$-1e20$ m	ϕ_{min}	$-90°$	T_{min}	0 N
z_{max}	0 m	ϕ_{max}	$90°$	T_{max}	$1e20$ N

www.manaraa.com

A.4 Trajectory Stitching

Table A.25: Optimization Bounds- Trajectory Stitching

Δt_{min}	0.05 s	$V_{a_{min}}$	4.0 m/s	$C_{L_{min}}$	-0.5
Δt_{max}	0.12 s	$V_{a_{max}}$	50.0 m/s	$C_{L_{max}}$	3.0
x_{min}	$-1e20$ m	$\gamma_{a_{min}}$	$-90°$	$\dot{\phi}_{min}$	-200 deg/s
x_{max}	$1e20$ m	$\gamma_{a_{max}}$	$90°$	$\dot{\phi}_{max}$	200 deg/s
y_{min}	$-1e20$ m	$\chi_{a_{min}}$	$-1e20°$	$\dot{C}_{Lmin}$	-200 s^{-1}
y_{max}	$1e20$ m	$\chi_{a_{max}}$	$1e20°$	$\dot{C}_{Lmax}$	200 s^{-1}
z_{min}	$-1e20$ m	ϕ_{min}	$-90°$	T_{min}	0 N
z_{max}	0 m	ϕ_{max}	$90°$	T_{max}	$1e20$ N

Table A.26: Optimization Parameters- Guidance Trajectory

t_f	10 s	N	100	μ	0.2 s^{-1}
$x(0)$	0.0 m	k_T	100	χ_d	0.0 deg
$y(0)$	0.0 m	k_p	0.1	tol_{opt}	$1\text{x}10^{-4}$
$z(0)$	0.0 m	A_x	80.0 m	tol_{fea}	$1\text{x}10^{-5}$
$\gamma_a(0)$	0.0 deg	A_y	0.0 m	optimal cost	$1.4954\text{x}10^{-2}$
				baseline cost	$5.3879\text{x}10^{4}$

www.manaraa.com

Table A.27: Optimization Parameters- Guidance Transition Trajectory

t_f	10 s	N	100	μ	0.2 s^{-1}
$x(0)$	61.4 m	k_T	100	χ_d	0.0 deg
$y(0)$	0.0 m	k_p	0.0	tol_{opt}	1x10^{-3}
$z(0)$	0.0 m	A_x	80.0 m	tol_{fea}	1x10^{-5}
$V_a(0)$	26.4 m/s	A_y	0.0 m	optimal cost	1.0700x10^{-7}
$\gamma_a(0)$	0.0 deg			baseline cost	5.3879x10^{4}
$\chi_a(0)$	-26.7 deg				
$\phi(0)$	61.0 deg				
$C_L(0)$	0.91				

Table A.28: Optimization Parameters- Loiter Trajectory

t_f	10 s	N	100	μ	0.2 s^{-1}
$x(0)$	110.0 m	k_T	100	R_d	20 m
$y(0)$	0.0 m	k_p	0.01	tol_{opt}	1x10^{-3}
$z(0)$	0.0 m	A_x	20.0 m	tol_{fea}	1x10^{-5}
$V_a(0)$	26.4 m/s	A_y	20.0 m	optimal cost	20.2414
$\gamma_a(0)$	0.0 deg	(x_g, y_g)	(110,20) m	baseline cost	6.5859x10^{4}
$\chi_a(0)$	-26.7 deg				
$\phi(0)$	61.0 deg				
$C_L(0)$	0.91				

www.manaraa.com

A.5 Thermal Soaring

Table A.29: Optimization Bounds- Thermal Soaring

Δt_{min}	0.05 s	$V_{a_{min}}$	4.0 m/s	$C_{L_{min}}$	-0.5
Δt_{max}	0.5 s	$V_{a_{max}}$	50.0 m/s	$C_{L_{max}}$	3.0
x_{min}	$-1e20$ m	$\gamma_{a_{min}}$	$-90°$	$\dot{\phi}_{min}$	-200 deg/s
x_{max}	$1e20$ m	$\gamma_{a_{max}}$	$90°$	$\dot{\phi}_{max}$	200 deg/s
y_{min}	$-1e20$ m	$\chi_{a_{min}}$	$-1e20°$	$\dot{C}_{Lmin}$	-200 s^{-1}
y_{max}	$1e20$ m	$\chi_{a_{max}}$	$1e20°$	$\dot{C}_{Lmax}$	200 s^{-1}
z_{min}	$-1e20$ m	ϕ_{min}	$-90°$	T_{min}	0 N
z_{max}	0 m	ϕ_{max}	$90°$	T_{max}	$1e20$ N

www.manaraa.com

Table A.30: Optimization Parameters- Thermal Soaring: Guidance Trajectory

t_f	10 s	N	100	V_{core}	6.0 m/s
$x(0)$	0.0 m	k_T	100	χ_d	0.0 deg
$y(0)$	0.0 m	k_p	0.1	tol_{opt}	1x10^{-4}
$z(0)$	-50.0 m	A_x	80.0 m	tol_{fea}	1x10^{-5}
$\gamma_a(0)$	0.0 deg	A_y	0.0 m	optimal cost	1.2148x10^{-6}
		$(x_{th}, y_{th}$	(40,-20) m	baseline cost	5.3879x10^4
		R_{lift}	10.0 m		

Table A.31: Optimization Parameters- Thermal Soaring: Loiter Trajectory

t_f	10 s	N	100	V_{core}	6.0 m/s
$x(0)$	0.0 m	k_T	100	R_d	30.0 m
$y(0)$	0.0 m	k_p	0.1	tol_{opt}	1x10^{-3}
$z(0)$	-50.0 m	A_x	20.0 m	tol_{fea}	1x10^{-5}
$\gamma_a(0)$	0.0 deg	A_y	20.0 m	optimal cost	1.0731x10^{-6}
		(x_g, y_g)	(0.0,20.0) m	baseline cost	6.6161x10^4
		$(x_{th}, y_{th}$	(20.0,20.0) m		
		R_{lift}	10.0 m		

www.manaraa.com