FLIGHT VEHICLE CONTROL AND AEROBIOLOGICAL SAMPLING APPLICATIONS

by

Laszlo Techy

Dissertation submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy
in
Aerospace Engineering

Craig A. Woolsey
David G. Schmale III
Christopher D. Hall
Daniel J. Stilwell
Cornel Sultan

November 5, 2009

Keywords: Unmanned Aerial Vehicle, Path Planning, Coordinated Control, Aerobiological Sampling

FLIGHT VEHICLE CONTROL AND AEROBIOLOGICAL SAMPLING APPLICATIONS

Laszlo Techy

ABSTRACT

Aerobiological sampling using unmanned aerial vehicles (UAVs) is an exciting research field blending various scientific and engineering disciplines. The biological data collected using UAVs helps to better understand the atmospheric transport of microorganisms. Autopilot-equipped UAVs can accurately sample along pre-defined flight plans and precisely regulated altitudes. They can provide even greater utility when they are networked together in coordinated sampling missions: such measurements can yield further information about the aerial transport process.

In this work flight vehicle path planning, control and coordination strategies are considered for unmanned autonomous aerial vehicles. A time-optimal path planning algorithm, that is simple enough to be solved in real time, is derived based on geometric concepts. The method yields closed-form solution for an important subset of candidate extremal paths; the rest of the paths are found using a simple numerical root-finding algorithm. A multi-UAV coordination framework is applied to a specific control-volume sampling problem that supports aerobiological data-collection efforts conducted in the lower atmosphere.

The work is part of a larger effort that focuses on the validation of atmospheric dispersion models developed to predict the spread of plant diseases in the lower atmosphere. The developed concepts and methods are demonstrated by field experiments focusing on the spread of the plant pathogen *Phytophthora infestans*.

ACKNOWLEDGMENTS

First of all I would like to express my gratitude to Dr. Craig Woolsey, my advisor, for his guidance and mentorship during my graduate program at Virginia Tech. I took his Nonlinear Control course when I first visited Virginia Tech as an exchange student, and found him one of the most enthusiastic professors I ever met. His advising helped to make the most of my graduate student career; he is a person I could always turn to when I needed advice. I was indeed very fortunate to have him, not only as my professor and advisor, but also as a friend.

I would like to thank Dr. David Schmale, my co-advisor, who introduced me to the field of aerobiology and aerobiological sampling. His enthusiasm and passion for science always gave me motivation. I also thank Dr. Chris Hall, Dr. Dan Stilwell, Dr. Cornel Sultan, Dr. Eugene Cliff and Dr. Naira Hovakimyan for giving me valuable feedback on my work.

I am grateful to Dr. Derek Paley for his generous help he provided implementing the motion coordination algorithm. The collaboration with him was a wonderful experience.

I thank my fellow graduate students and coworkers at the Nonlinear Systems Laboratory: Mark Monda, Jesse Whitfield, Nina Mahmoudian, Amanda Young, Chris Cotting, Bob Kraus, Konda Reddy, Enric Xargay and Lili Ma. I am especially grateful to Justin Murtha, whose help was indispensable during the flight experiments. I also thank Brian McCarter for his help with the aircraft instrumentation.

I am indebted to John Cianchetti for always being ready to help. He managed the field experiments conducted at Virginia Tech's Kentland Farm, and was also the pilot for many

of the UAV flights at Kentland Farm.

I thank my parents, Mária and Zsolt, for their love and support over this long journey. They made it possible for me to pursue my doctoral studies by providing me with the comfort of a safe home I could always go back to. My achievements would not have been possible without their caring support. I also thank the rest of my family for their love and support. I am deeply indebted to Elika for her love and encouragement over the past years. I thank her for always standing beside me.

I am grateful to Monika Gibson for carefully reading through my dissertation and providing stylistic and editorial feedback on the manuscript.

Téchy László

Blacksburg, Virginia

Contents

Table of Contents **v**

List of Figures **vii**

List of Tables **xi**

1 Introduction **1**
1.1 Motivation . . . 3
1.2 Contributions of Dissertation . . . 9
1.3 Dissertation Outline . . . 10

2 Mathematical Preliminaries **12**
2.1 Pontryagin's Minimum Principle . . . 12
2.2 Differential Geometry of Curves . . . 15

3 Minimum-Time Path Planning in Steady Uniform Winds **23**
3.1 Problem Formulation . . . 27
3.2 Properties of Extremal Paths . . . 29
3.3 Computing BSB Trajectories . . . 35
3.3.1 Closed-form Solution for LSL and RSR Trajectories . . . 39
3.3.2 Numerical Solution for LSR and RSL Trajectories . . . 43
3.3.3 Existence of "Unconventional" Extremals . . . 45
3.4 Computing BBB Trajectories . . . 47
3.5 Results . . . 48
3.6 Path Planning with Turn Rate and Acceleration Bounds . . . 54
3.6.1 Vehicle Model . . . 55
3.6.2 Path Planning . . . 58
3.6.3 Results . . . 61

4 Control-Volume Sampling **65**
4.1 Coordinated Control in Aerobiological Sampling . . . 70
4.2 Modeling . . . 71
4.2.1 Particle Motion in an External Flow . . . 71
4.2.2 Curve Model . . . 72

4.3 Particle Motion Coordination . . . 73
4.3.1 Decoupled Curve Control . . . 73
4.3.2 Translation Invariant Control . . . 73
4.3.3 Time-splay Coordination . . . 74
4.4 Strictly Convex Approximation of Linear Path Segments . . . 76
4.5 Parametrization of D-Curves . . . 79
4.6 Simulation Results . . . 80
4.7 Hardware-in-the-Loop Simulation . . . 83
4.8 Summary of the Control-Volume Sampling Method . . . 86

5 Aerobiological Sampling Field Experiments 87
5.1 Field Experiments . . . 88
5.2 UAV Platforms . . . 90
5.3 Coordinated Control Experiments Using Two UAVs . . . 93
5.3.1 Coordinated Control via Speed Modulation . . . 94
5.3.2 Processing of Aerobiological Samples from UAVs . . . 97
5.3.3 Experimental Results . . . 98
5.3.4 Collection of Viable Sporangia of *P. infestans* . . . 100
5.4 Long-distance Sampling Using a Buoyancy-Controlled Weather Balloon . . . 101
5.4.1 Buoyancy-Controlled Balloon . . . 103
5.4.2 Hardware-in-the-Loop Simulation . . . 106
5.4.3 Flight Experiments and Results . . . 107

6 Coordinated Perimeter-Patrol with Minimum-Time Alert Response 111
6.1 The Perimeter Patrol Problem . . . 112
6.2 Simulation Results . . . 114
6.2.1 Ellipsoidal Base Perimeter . . . 115
6.2.2 Wright-Patterson Air Force Base . . . 116

7 Cross-track Control of an Autonomous Underwater Vehicle 120
7.1 Underwater Vehicle Equations of Motion . . . 121
7.2 Directional Stabilization . . . 126
7.3 Cross-Track Control . . . 133
7.4 Simulations . . . 140
7.5 Summary of the Cross-Track Control Algorithm . . . 142

Bibliography 145

A Computing BBB Trajectories: Details of Calculation 154

B Spectral Stability Analysis of the Closed-loop AUV Control System 157

List of Figures

1.1 Late blight of potato, caused by the fungus-like organism Phytophthora infestans, in a source field at Virginia Tech's Kentland Farm. Sporangia form in a cloudy halo around necrotic lesions and are released in turbulent winds for transport in the atmosphere. Photo credit: David G. Schmale III. . 5
1.2 Illustration showing the spread of microscopical fungal sporangia from an infected crop field. The sporangia may be picked up by turbulent airflow over the plant canopy, and may be carried long distances away from the source by ambient winds. Unmanned aerial vehicles using spore sampling devices can assess spore concentrations at various altitudes above ground level. 7

2.1 A helix, parameterized by a single parameter. 17
2.2 Hypocycloid and epicycloid curves. 18
2.3 Trochoidal path is generated by a point P a distance R from the center of a circle of radius r, as the circle is rolling along a straight line. 19
2.4 A clothoid curve, also known as Euler spiral. 20
2.5 Convex region D in the plane. 21

3.1 Venn diagram showing the types of extremals considered in this work. The abbreviations should be interpreted as follows: As in [67], "B" stands for "bang," or maximum rate turn, "S" stands for "straight" (also stands for "singular"), "L" stands for "left," and "R" stands for "right." For example, LSL is an initial left turn followed by a straight path, then followed by a second left turn. 26
3.2 Trochoidal path and trochoidal frame. 28
3.3 Relationship between air speed, wind speed and groundspeed. 33
3.4 The time the UAV spends on an extremal trochoidal segment has to be less than $2t_{2\pi}$. 34
3.5 The dashed lines show candidate extremal paths from initial state to final state. The solid lines are the trochoidal segments at the initial and final states plotted for $t \in [0, t_{2\pi}]$. 37

3.6 Candidate extremals for $V_w = 0$. The figure illustrates how t_A is obtained by scaling $t_{2\pi}$ with $\frac{\alpha}{2\pi} \in [0, 1]$. In the above case $\alpha = \frac{\pi}{2}$. When turning "left," $\delta_1 = -1$, so $t_A = -\frac{1}{4}t_{2\pi} \equiv \frac{3}{4}t_{2\pi}$. 40
3.7 The effect of $m \in \mathbb{Z}$ is a parallel shift with respect to the feasible solution. The effect of $k \in \mathbb{Z}$ is a variety of different tangent lines connecting the initial and final trochoids. 41
3.8 The function $\frac{f(t_A)}{V_a V_w}$ for $t_A \in [-2t_{2\pi}, 2t_{2\pi}]$. There are several roots for $t_A \in [0, t_{2\pi}]$. 44
3.9 Illustration showing that, for some initial conditions, some of the extremals fail to satisfy $t_B \in [0, t_{2\pi}]$. In this example $t_B < 0$ for the LSL trajectory. The figure shows both the inertial path and the air relative path. The circles indicate the position of a "virtual target" [41], traveling upwind with the same speed as the wind speed, that the aircraft would intercept by flying a minimum-time path in the air relative frame. 47
3.10 Candidate BSB extremal paths for randomly selected initial and final states. 51
3.11 Candidate time-optimal paths found by the path planning algorithm for initial conditions $x_0 = 0$ m, $y_0 = -200$ m, $\psi_0 = 0$, $x_f = 0$ m, $y_f = -180$ m $\psi_f = \pi$, $\omega = 0.2832$ rad/s, $V_w = 5$ m/s, $V_a = 20$ m/s. The time-optimal path is LRL trajectory with $t_A = 0.6869$ s, $t_B = 17.4451$ s and $T = 22.4228$ s. 52
3.12 Candidate time-optimal paths found by the path planning algorithm for initial conditions $x_0 = 0$ m, $y_0 = -200$ m, $\psi_0 = 0$, $x_f = 0$ m, $y_f = -180$ m $\psi_f = \pi$, $\omega = 0.2832$ rad/s, $V_w = 5$ m/s, $V_a = 20$ m/s. The time-optimal path is LRL trajectory with $t_A = 0.6869$ s, $t_B = 17.4451$ s and $T = 22.4228$ s. 53
3.13 Bank angle and heading angle for two different cases of turns. In *Case 1* the maximum bank angle is reached, while in *Case 2* the maximum bank angle $\bar{\phi}$ is not reached. 57
3.14 A Type 1 maneuver. In this case both the initial and final turns are saturated, i.e. the maximum bank angle is reached. 63
3.15 A Type 2 maneuver. Both the initial and final turns are completed without reaching the maximum bank angle. 63
3.16 Figure showing all four extremal paths. The candidate minimum-time trajectory is a Type 1 right turn followed by a straight line followed by a Type 1 left turn. 64

4.1 In the control-volume sampling problem one is interested in assessing the net rate of outflow across the boundaries of the volume. By continuity, this equivalently yields the rate of release inside the volume. 66
4.2 Two UAVs flying in the control-volume sampling mission. UAV1 (UAV2) samples only during the upwind (downwind) path segment. The endpoints of the sampling arc are connected with a time-optimal path discussed in Chapter 3. 68
4.3 Approximating the zero curvature segment with a circular arc. 78

4.4 Straight portions of closed convex curves can be approximated by circular arcs to arbitrary precision. The figure on the left shows a curve that was generated by the path planning algorithm of Chapter 3. The straight segment is then approximated by circular arcs of radii $R_0 = 700\text{m}$, $R_0 = 1000\text{m}$ and $R_0 = 2000\text{m}$, as shown in the figure on the right. The approximation method is discussed in Section 4.4. 79
4.5 Simulation results of the time-splay coordination algorithm for control-volume sampling. Wind speed is $V_w = 0$. 81
4.6 Simulation results of the time-splay coordination algorithm for control-volume sampling. Wind speed is $V_w = 5\ \text{m/s}$ from the South. 82
4.7 Simulation results of the time-splay coordination algorithm for control-volume sampling. Wind speed is $V_w = 15\ \text{m/s}$ from the South. 83
4.8 Coordinated flight of two UAVs in a 3-dimensional control-volume sampling mission. The UAVs perform a vertical scan of the sporangia plume to estimate the rate of sporangia generation inside the volume. 84
4.9 Hardware-in-the-loop simulation results for the control-volume sampling problem. The figure shows simulation results over the course of a 45 minute sampling. The bold line shows the last orbit. 85

5.1 Spore sampling devices used onboard UAVs in aerobiological sampling. Figure 5.1(c) shows two UAVs in a synchronized aerobiological sampling experiment on August 14, 2008. 91
5.2 A SPAARO (Small Platform for Autonomous Aerial Research Operations) UAV developed by Murtha et. al. was used in the experiments [49]. 93
5.3 UAVs traveling along a circular flight pattern. The desired phase separation θ^* is achieved by controlling the speed of the vehicles. 94
5.4 Viability test (able to cause disease on greenhouse-grown potatoes) of *Phytophthora infestans* sporangia. The filter paper is immediately removed from the samplers after the sampling flight and transported to the laboratory. . . 98
5.5 3D position data and phase error for two UAVs in a phase synchronized coordinated control experiment on August 14, 2008, 9:30 a.m. EST. UAV1 (bottom) flying at $25\ \text{m}$ AGL; UAV2 (top) flying at $45\ \text{m}$. 100
5.6 3D position data and phase error for two UAVs in a phase synchronized coordinated control experiment on August 15, 2008, 9:30 a.m. EST. UAV1 (bottom) flying at $25\ \text{m}$ AGL; UAV2 (top) flying at $45\ \text{m}$. 101
5.7 Position data and phase error for two UAVs in a phase-synchronized coordinated control experiment on August 15, 2008, 11:30 a.m. EST. Both UAVs fly at the same altitude with radii $130\ \text{m}$ and $160\ \text{m}$, respectively. 102
5.8 Photograph of the buoyancy-controlled weather balloon used in aerobiological sampling experiments. 103
5.9 SPAARO UAV and the buoyancy-controlled weather balloon. 104
5.10 Schematic drawing of the balloon payload. The payload mainly consists of the data logger unit, communication devices and buoyancy-control devices. 105

5.11 Hardware-in-the-loop simulation of the flight of two UAVs tracking a weather balloon. The UAVs' circular flight path is centered on the current position of the balloon; the corresponding UAV trajectories are *trochoid* curves. 2-D plot of the experiments. 107
5.12 Hardware-in-the-loop simulation of the flight of two UAVs tracking a weather balloon. The UAVs' circular flight path is centered on the current position of the balloon; the corresponding UAV trajectories are *trochoid* curves. 3-D plot of the experiments. 108
5.13 Flight test results from August 6, 2009. The UAV tracked the weather balloon after the balloon was released from the potato field. 2-D plot of the experiments. 109
5.14 Flight test results from August 6, 2009. The UAV tracked the weather balloon after the balloon was released from the potato field. 3-D plot of the experiments. 109
5.15 Flight test results from October 8, 2009. 2-D plot of the experiments. 110
5.16 Flight test results from October 8, 2009. 3-D plot of the experiments. 110

6.1 Conceptual sketch of the time-optimal alert response scenario. 113
6.2 Flow diagram of the time-optimal alert response scenario. 114
6.3 Space-time analysis of coordinated perimeter coverage with time-optimal alert response. 115
6.4 Simulation of coordinated perimeter coverage with time-optimal alert response around a general ellipsoidal curve. 117
6.5 Seven points were selected around WPAFB to define a closed convex curve. The curve then serves as the perimeter to be patrolled by a team of UAVs. The straight segments are approximated by arcs of circles of radius $R_0 = 50\ \mathrm{km}$. 118
6.6 Simulation of coordinated perimeter coverage with time-optimal alert response around WPAFB. 119

7.1 Hydrodynamic angles for an axisymmetric body. 124
7.2 Line of sight guidance. 134
7.3 Simulation of the linear controller with $u_d = 0.8$ (solid) and $u_d = 1.3$ (dotted). 143
7.4 Simulation of the nonlinear controller with $u_d = 0.8$ (solid) and $u_d = 1.3$ (dotted). 144

List of Tables

3.1 Path-planning problem types for planar kinematic vehicle models. 24
3.2 The initial conditions were selected randomly in the ranges shown in the table. The final point was selected R distance away from the initial point in the direction defined by θ. 49
3.3 Distribution of minimum-time path types based on Monte Carlo simulations. 49
3.4 Parameters used in simulations. 62

5.1 Summary of the flight experiments conducted during August 14-15, 2008. Viable spores of *P. infestans* were collected during flights LT5 and LT6. . . . 99

7.1 Values of ζ and λ which satisfy equation (7.20). 132
7.2 Vehicle and control parameter values. 141
7.3 State and parameter ranges for simulations. 141

Chapter 1

Introduction

Unmanned vehicles have enjoyed increasing attention from researchers in the past decades. The second half of the 20th century witnessed marvellous technological and scientific achievements, culminating in the development of space technologies. This new era, often referred to as the Space Age, induced the need to develop robotic vehicles that are able to execute missions essential to the advancement of science. Sending spacecraft to unexplored distances required vast commitment and resources, not to mention the unparalleled infrastructure and logistics necessary for the success of such missions. Due to our physiological needs and the additional safety measures required to send humans to uninhabited and hostile environments, it was desired to develop vehicles that are able to explore on their own, and serve as proxies from remote places, such as the far distances of our Solar System or the depths of the oceans [9]. Although unmanned vehicles — vehicles that have no humans onboard — have been frequently employed over the past decades for missions where human involvement would be too dangerous or too expensive, the challenge to furnish them with onboard autonomy — the ability to sense, decide and react on their own — proved to be a challenging problem, inspiring much of exciting, state of the art research.

The advancements in computer systems, especially miniaturization and the increase in computational power, and the emergence of new technologies, such as novel sensors, wireless communication networks, accurate navigation systems and health-monitoring systems to name a few, enabled autonomous vehicles with increased mission capabilities. In recent years, unmanned vehicles have played an increasing role in scientific research. Highly efficient underwater gliders — winged underwater robots that use buoyancy as a passive propulsion source — have collected physical oceanographic measurements over unprecedented spatial and temporal scales [52, 82]. Other autonomous underwater vehicles (AUVs) have been used to measure the heat exchange between ice and seawater [16], and to monitor the distribution of Antarctic krill under the sea ice [1]. Improvements in automatic control systems allowed truly autonomous missions on spacecraft like NASA DS1, and increased the quantity of scientific data returned from the Mars Exploration Rovers Opportunity and Spirit [9]. Unmanned aerial vehicles (UAVs) have made successful flights in Antarctica, carrying miniaturized turbulence probes that measure the detailed structure of wind and temperature along the flight path [75]. The measurements can be used to estimate the heat exchange between the lower atmosphere and sea ice. UAVs have demonstrated their effectiveness in aerobiological research to monitor the movement of plant pathogens in the lower atmosphere [17, 62, 6]. Using autonomous vehicles as individual agents offers some advantages in addressing tasks that can be categorized by the "three D's": dirty, dull, or dangerous. Applications include, but are not limited to, mining [8], search and rescue [48], surveillance and reconnaissance [3] and forest fire monitoring [66]. Unmanned vehicles have the potential to provide even greater utility when networked to accomplish tasks more quickly and efficiently. Accordingly, the problem of coordinating multiple autonomous vehicles to address scientific and other missions has enjoyed increased attention from researchers [52, 60, 69].

This work focuses on path planning and control strategies for autonomous unmanned

flight vehicles. The methods and concepts primarily consider environmental data collection applications, with special emphasis on field experiments aimed to monitor the spread of plant pathogens in the lower atmosphere.

1.1 Motivation

In this work path planning and control strategies are discussed that are to be used in aerobiological sampling field experiments. Aerobiology is a branch of science that studies the processes and factors that influence the motion of aerobiota in the atmosphere. Aerobiota may include plant and animal pathogens, insects, seeds and pollen, or other living organisms that use the air to change habitat [29]. The atmosphere of the Earth is primarily composed of the mixture of two gases, nitrogen ($\approx 78\%$) and oxygen ($\approx 21\%$), commonly known as air. Other than gases, air also contains small particles, like dust, and it may also host microscopical living organisms. Although many of these aerial living organisms have a significant impact on humans, little is known about the processes and mechanisms that affect their motion. One may face numerous fascinating questions when dealing with aerial life forms: In addition to questions regarding their role, and interaction with other living organisms, for example, one is concerned with the aerial transport mechanisms that transfer them from one location to another.

Aerobiological research aims to improve our understanding of these aerial organisms to ensure human health and to help sustain and safely manage terrestrial ecosystems, including important food resources. Airborne bacteria, fungi, viruses and pollen are responsible for many human respiratory diseases and allergies. These diseases include *Pulmonary tuberculosis*, *Diphtheria*, or *Influenza*, for example.

An indirect impact on humans of these living aerial organisms is the devastating effect of plant diseases.

Globally, wheat accounts for an average of between one-third and one-half of each person's daily calories, making wheat a primary food source for most people — on a per capita basis, wheat consumption averaged around $68\ \mathrm{kg}$ [$150\ \mathrm{lbs}$] worldwide in 2008 [74]. *Puccinia graminis*, a fungus that causes wheat stem rust, one of the most devastating diseases of wheat, poses a major threat to the world's primary food supply [65]. This fungus has co-evolved with wheat; a small change in either organism's genetic information can make the wheat resistant or susceptible to the disease. In this genetic race it is only a matter of time when the currently resistant wheat varieties become susceptible to a new, evolved strain of the fungus. In agriculture, plant breeders constantly work on developing new, and more resistant plant varieties, and until recently, wheat stem rust was thought to be defeated. However, in 1999, a new strain of the fungus was discovered in Uganda that can defeat the resistance of most varieties of wheat. The new strain, referred to as Ug99, is feared to reach other parts of the world by aerial transport or deliberate introduction [25].

Another example of a deadly plant disease is potato blight. *Phytophthora infestans* is a "fungus-like" organism (*oomycete*) that is a pathogen of potato and tomato, and responsible for a disease known as late-blight: this organism is the specific focus of the field experiments discussed in Chapter 5. *P. infestans* causes lesions on the plant leaves (Figure 1.1), around which new spore-bearing structures called sporangia are formed. The sporangia are released into the air usually during the early morning hours, where they get picked up by turbulent airflow over the plant canopy. Some of these sporangia may be transported to higher altitudes into the planetary boundary layer of the atmosphere to begin their journey to a new location, where they might infect healthy plants, creating a risk for wide-spread epidemics. Potato late blight disease was introduced to Europe in the middle of the 19th century and had grave impact on potato production, ravaging crops throughout the continent. Although many countries were affected, the consequences were espe-

Figure 1.1 Late blight of potato, caused by the fungus-like organism Phytophthora infestans, in a source field at Virginia Tech's Kentland Farm. Sporangia form in a cloudy halo around necrotic lesions and are released in turbulent winds for transport in the atmosphere. Photo credit: David G. Schmale III.

cially severe in Ireland, where about one-third of the population was entirely dependent on potato. The resulting wide-spread starvation entered history as the infamous "Irish Potato Famine," a period between 1845 and 1852, during which an estimated one million people died and many more were forced to emigrate from the country [79]. *P. infestans* is still a major concern today, as many of the common potato varieties are susceptible to the disease. A risk management system is needed that would help growers make informed decisions about fungicide use, or sanitation.

In order to accurately assess the risk of an infection at some location spreading to other areas, mathematical models of varying complexity may be employed to predict the trajectories of the sporangia, given ambient wind and other environmental conditions [6,7]. It is understood that long-distance aerial transport plays an important role in the evolution of plant disease epidemics [4]. The transport process can be important when studying how the disease may be introduced in a new region on a continental, inter-continental scale, or when studying periodic re-introduction of a fungus in a region where it normally cannot over-season. Long-distance aerial transport models are required to predict the motion of fungal pathogens in the air, in order to support a prediction system that would help make plant-health-management decisions, such as the application of appropriate fungicides.

The transport process can be grouped into five major components [4]: (1) production of sporangia, often denoted as Q (number of sporangia/m^2/s); (2) escape efficiency (the percentage of sporangia that are able to leave the plant canopy); (3) turbulent transport and diffusion in the atmosphere; (4) sporangia survival; (5) and deposition onto susceptible host tissues. Once escaped into the air above the canopy, the initial movement of sporangia downwind from the source may be described as a Gaussian plume; such models are popular because of their analytical elegance and their ability to capture the basic convection and diffusion principles [64, 22]. However, these models fail to incorporate other complex physical and biological features affecting the dispersal process, such as

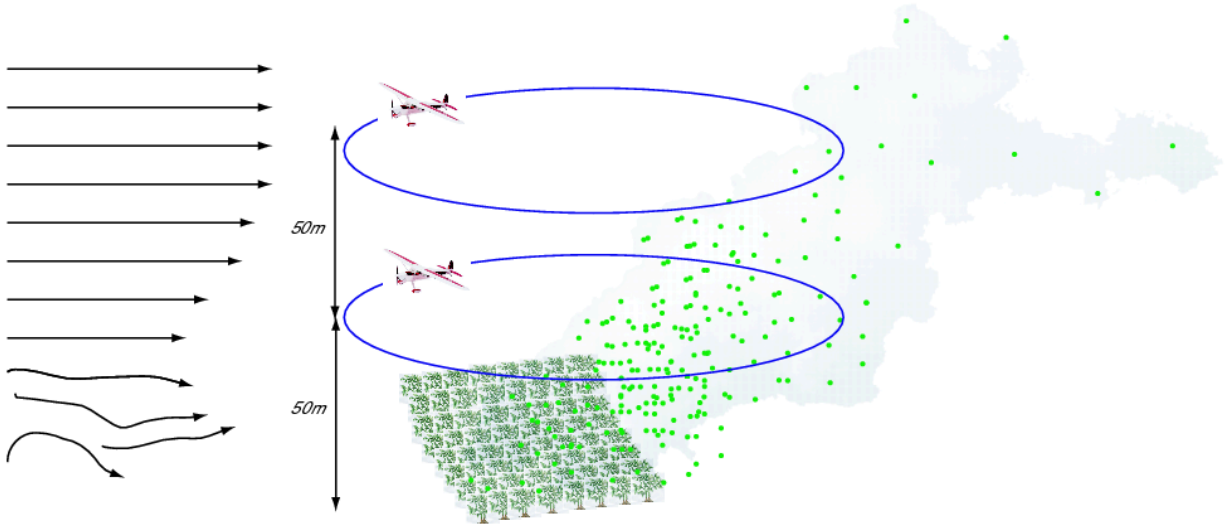


Figure 1.2 Illustration showing the spread of microscopical fungal sporangia from an infected crop field. The sporangia may be picked up by turbulent airflow over the plant canopy, and may be carried long distances away from the source by ambient winds. Unmanned aerial vehicles using spore sampling devices can assess spore concentrations at various altitudes above ground level.

modeling turbulent airflow and deposition of small particles on the potato canopy [35]. Lagrangian stochastic (LS) simulation models may be used, since they are capable of producing quantitatively accurate estimates of Q for sporangia [7].

To assess the accuracy of atmospheric dispersion models, the theoretical findings need to be validated by experiments. Field experiments focusing on the spread of the plant pathogen *P. infestans* are conducted at Virginia Tech's Kentland Farm to support the validation effort. The project's immediate goals are to: (1) collect aerobiological samples of fungal sporangia concentrations at various altitudes and geographical locations above infected potato fields; (2) assess the validity of atmospheric dispersion models used to predict and forecast the movement and dispersal of *P. infestans*. UAVs equipped with microbe-sampling devices are used to measure average concentrations at tens to hundreds of feet above ground level. Remote controlled (RC) UAVs have been used in the past to track the movement of pollen [6], insects [63], seeds [62], and pathogens [7] above

crop fields. Recent work highlighted the additional advantages that autopilot-equipped UAVs can provide in aerobiological sampling field experiments [17,18] in terms of the accuracy in the sample volume and sampling pattern (altitude and geographical location).

The concepts and methods developed in this work were primarily motivated by the need to improve the precision and accuracy of aerobiological sampling missions conducted with UAVs in the lower atmosphere. During this endeavor, an immediate goal was to apply existing technologies and platforms for aerobiological sampling field experiments to support ongoing validation efforts focusing on the spread of *Phytophthora infestans*. At the same time it was desired to develop new concepts and methodologies that would: (1) improve the information content from the data that could be gathered during a sampling interval; (2) introduce new ways of sampling that would yield further information about the aerial transport process than was previously possible. In (1), one may want to gather data with two (or more) autonomous UAVs *simultaneously* at different altitudes in such a way that, by exchanging position information, the vehicles follow the same ground path at the same time. Such measurements give additional information about the vertical spore concentration gradient, and allow the comparison of concentrations between flights, as the UAVs enter, sample, and exit the spore plume at the same time. The *control volume sampling* problem, where one may want to assess the rate of outflow through the boundaries of a control volume is an example of (2). Emerging from the desire to improve the information gain from the sampling missions, several interesting fundamental research problems were formulated, such as time-optimal path planning in the presence of steady uniform winds, and multi-UAV coordination along the resulting time-optimal curves in an external, uniform flow-field.

1.2 Contributions of Dissertation

The contributions of the dissertation — in order of appearance — are as follows

- Development of a time-optimal path planning algorithm for flight vehicles in the presence of a steady, uniform flow-field. Previous research demonstrated that the time-optimal paths have to be composed of straight lines and maximum rate turns. The contribution to the time-optimal path planning field presented in this dissertation is a simple analytical solution for a subset of the candidate extremal paths: those for which an initial maximum rate turn is followed by a straight path followed, by a second maximum rate turn in the same direction as the first one. The rest of the candidate extremals are found using an efficient numerical root-finding routine. Additionally, further light is shed on the properties of extremals by showing that, for some candidate extremal trajectories, no corresponding Dubins path exists in the air-relative frame. The path planning algorithm is extended to the case where the turn acceleration is also bounded in addition to the turn rate.

- The concept of control volume sampling using two autonomous UAVs is applied to a specific aerobiological sampling problem, where the UAVs modulate their sampling activity such that one UAV samples only upwind from the source, while the other UAV samples only downwind from the source. Previous results on particle coordination on convex curves are applied to this problem, with the modification that the UAVs do not share the *same* flight path. Synchronization is made possible by the fact that the flight paths used in the control volume sampling problem — referred to as "D-curves" — are symmetric.

- The time-optimal path planning procedure and particle synchronization algorithm have been implemented on the on-board computer of UAVs for real-time execution. The methods have been tested in high fidelity hardware-in-the-loop simulations.

- In addition to the theoretical work, significant emphasis was placed on the data collection task to support ongoing field experiments during July and August, 2008 and August, 2009. UAV flights were conducted in July and August, 2008, where the UAVs collected *Phytophthora infestans* sporangia at $25\ \mathrm{m}$ to $45\ \mathrm{m}$ above ground level. Some of the plates in the experimental flights contained viable *Phytophthora infestans* sporangia: sporangia able to cause disease on greenhouse-grown potato leaves.

- The time-optimal path planning algorithm is applied to an integrated perimeter patrol problem, where the UAVs synchronize their motion around a convex curve, and take time-optimal shortcut paths to prosecute intrusion alerts along the patrol perimeter.

- Stability analysis for an underactuated autonomous underwater vehicle (AUV) that is controlled using a nonlinear cross-track control algorithm. Previous results on directional stabilization of underactuated AUVs are extended to allow convergence to a straight line in inertial space. The stability of the closed-loop system is assessed using Lyapunov's Indirect Method, and the robustness of the controller is demonstrated by exhaustive Monte Carlo simulations. The performance of the nonlinear controller is compared to a simple line-of-sight guidance controller, and its superiority in terms of a broader performance envelope is shown.

1.3 Dissertation Outline

This dissertation is divided into the following chapters: In Chapter 2 some fundamental concepts of optimal control and differential geometry are reviewed. In Chapter 3 a time-optimal path planning algorithm is described that yields minimum-time trajectories between initial and final points with prescribed headings in the presence of steady,

uniform winds. The methods may be applied for any vehicle path planning problem where the kinematic vehicle model and the assumption of a constant, uniform, external flow-field is reasonable. The applications include but are not limited to flight vehicles in a steady wind, surface vessels in riverine environments and underwater vehicles in ocean currents. The multi-vehicle coordination problem for UAVs is addressed in Chapter 4. Recently developed theory for particle coordination, based on a series of papers including [60] [51] is extended to allow for coordination on time-optimal curves in wind. In this work, the particle coordination method is applied to the *control-volume sampling* application (Chapter 4), where two UAVs modulate their sampling activity to ensure consistent and efficient assessment of outflow of particles across the boundaries of the control-volume. In Chapter 5, coordinated aerobiological sampling field experiments are discussed. The UAV coordination method is directly applicable to other practical problems as well. Chapter 6 presents the coordinated perimeter patrol problem, where a group of UAVs are tasked with the problem of patrolling a certain convex perimeter by minimizing coverage gaps, and at the same time prosecuting intrusion alerts. Chapter 7 is focused on directional stabilization and cross-track control of an underactuated autonomous underwater vehicle (AUV). Although the dissertation is primarily focused on aerobiological sampling using UAVs, the developed methods may be easily applied for oceanographic sampling applications using unmanned underwater vehicles. Although the properties and conditions of the medium that underwater and aerial vehicles operate in are very different, the physical laws that govern their motion are the same. The path-planning and particle synchronization results discussed in later chapters may be immediately applied for AUVs performing data collection missions in ocean currents.

Chapter 2

Mathematical Preliminaries

2.1 Pontryagin's Minimum Principle

The purpose of this section is to introduce Pontryagin's Minimum Principle: a powerful tool that will be used in Chapter 3 to derive necessary conditions for time-optimal paths. The ideas presented here are well established within the classical optimal control literature [33, 14, 55, 36].

The origins of calculus of variations go back to the 17th century. The *brachistochrone* problem was first studied by Galileo Galilei who was interested in finding the shape of the curve, along which a bead would slide down in minimum time. The name *brachistochrone* itself comes from the Greek *brachistos* for "shortest" and *chronos* for "time." The problem was formulated mathematically by Johann Bernoulli in 1696, and it was solved in 1697 by Johann Bernoulli, Jakob Bernoulli, Sir Isaac Newton and others. The *brachistochrone* problem was the first optimal control problem that was formulated with mathematical rigor and eventually solved using the calculus of variations.

Classical calculus of variations gives necessary conditions for a solution to be optimal with respect to some measure — the performance index. A chief drawback of the method

is that in most cases it only provides necessary conditions and gives no insight into how the optimal solutions may be found.

In optimal control one is interested in obtaining the optimal control signal that shapes the evolution of the states of a dynamical system, such that some performance index is minimized. For linear systems, and considering quadratic cost functionals, a closed-form solution can be obtained. For nonlinear systems, however, no general solution exists. The methods of the calculus of variations can be employed with the hope to obtain necessary conditions for optimality, however, the variational approach to the optimal control problem only provides necessary conditions inside an open set. In the classical treatment — presented, for example, in [33] — one has to make rather restrictive smoothness assumptions on the functions that are considered. Moreover, the control $\mathbf{u}^*$ is assumed to be unbounded, even though for minimum time control problems the control is often "bang-bang," i.e. it takes on values on the control boundary. The theory developed by Pontryagin and his students [55] addresses the problem where the control is only assumed to be piecewise continuous, and it can take on values on a compact set, i.e. including the boundaries. An alternative geometric approach to obtain the minimum principle is presented in [36].

Consider the system

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

where $\mathbf{x}(t) \in \mathbb{R}^n$ is the state vector, $\mathbf{u}(t) \in \mathbb{R}^m$ is the vector of control signals, and $\mathbf{f}(\cdot) : \mathbb{R}^n \times \mathbb{R}^m \mapsto \mathbb{R}^n$ is smooth. We call the control *admissible* if

$$\mathbf{u}(\cdot) : [t_0, t_f] \mapsto \Omega \subset \mathbb{R}^m$$

takes values in a closed and compact set. We consider the admissible controls to be arbitrary piecewise continuous functions, i.e. $\mathbf{u} = \mathbf{u}(t)$ which are continuous for all t, with the exception of only a finite number of isolated times, t, at which $\mathbf{u}(t)$ may have discon-

tinuities of the first kind.[1] Assume that the initial condition $\mathbf{x}(t_0) \in \mathbb{R}^n$ is given and the final condition $\mathbf{x}(t_f) \in \mathbb{R}^n$ is specified. Consider the performance index in the form

$$J(\mathbf{u}) = \int_{t_0}^{t_f} f_0(\mathbf{x}, \mathbf{u}) dt.$$

Define the augmented adjoint vector as

$$\tilde{\boldsymbol{\lambda}}(t) = [\lambda_0, \lambda_1(t), ..., \lambda_n(t)]^T$$

and the augmented right-hand side function

$$\tilde{\mathbf{f}}(\mathbf{x}, \mathbf{u}) = [f_0(\mathbf{x}, \mathbf{u}), f_1(\mathbf{x}, \mathbf{u}), ..., f_n(\mathbf{x}, \mathbf{u})]^T,$$

where $f_i(\mathbf{x}, \mathbf{u})$, $i = 1, \ldots, n$, is the ith component of $\mathbf{f}(\mathbf{x}, \mathbf{u})$. Define the *variational Hamiltonian* as

$$\mathcal{H}(\tilde{\boldsymbol{\lambda}}(t), \mathbf{x}(t), \mathbf{u}(t)) \equiv < \tilde{\boldsymbol{\lambda}}, \tilde{\mathbf{f}}(\mathbf{x}, \mathbf{u}) > = \sum_{i=0}^{n} \lambda_i f_i(\mathbf{x}, \mathbf{u}).$$

Theorem 2.1.1. ***Pontryagin's Minimum Principle [55].*** *If $\mathbf{x}^*(t)$, $\mathbf{u}^*(t)$ defined on $[t_0, t_f]$ is an optimal state-control pair, then there exists a real number λ_0 and an absolutely continuous vector valued function $\boldsymbol{\lambda}(t) : [t_0, t_f] \mapsto \mathbb{R}^n$, such that*

- $$\lambda_0 \geq 0, \quad \tilde{\boldsymbol{\lambda}}(t) \neq 0.$$
- $$\dot{\boldsymbol{\lambda}}(t) = -\frac{\partial \mathcal{H}}{\partial \mathbf{x}} \mid_{\mathbf{x}^*, \mathbf{u}^*}$$
- $$\mathcal{H}(\tilde{\boldsymbol{\lambda}}(t), \mathbf{x}^*(t), \mathbf{u}^*(t)) = \inf_{\mathbf{u} \in \Omega} \{\mathcal{H}(\tilde{\boldsymbol{\lambda}}(t), \mathbf{x}(t), \mathbf{u})\}$$

[1] A function $f(x)$ has a discontinuity of the first kind at the isolated point x_0, if $f(x)$ is discontinuous at x_0, but both $\lim_{x \to x_0^+} f(x)$ and $\lim_{x \to x_0^-} f(x)$ exist, and take finite values. Also called a *jump*.

- *At the terminal time t_f the relation*

$$\inf_{\mathbf{u}\in\Omega}\{\mathcal{H}(\tilde{\boldsymbol{\lambda}}(t_f),\mathbf{x}(t_f),\mathbf{u})\} \triangleq \mathcal{M}(\tilde{\boldsymbol{\lambda}}(t_f),\mathbf{x}(t_f)) = 0 \tag{2.1}$$

  is satisfied.

Furthermore, it turns out that the value of the Hamiltonian along the optimal trajectory is constant, so condition (2.1) may be verified at any time $t \in [t_0, t_f]$, and not just at t_f. □

2.2 Differential Geometry of Curves

In this section an overview of elementary differential geometry is presented that will be used in later chapters. The development primarily follows [73]. The basic concepts are discussed for the general three-dimensional Euclidean space, and then more emphasis is placed on the discussion of planar curves. Curves in the plane correspond to trajectories that an autopilot-controlled UAV would follow at a fixed altitude. We will see in Chapter 3 that time-optimal paths in the presence of winds may be composed of *trochoidal* segments and straight lines.

In $\mathbb{R}^3$, the three-dimensional Euclidean space, a vector is a set of three numbers $\mathbf{x} = (x_1, x_2, x_3)^T$. The vectors $\mathbf{i} = (1,0,0)^T$, $\mathbf{j} = (0,1,0)^T$ and $\mathbf{k} = (0,0,1)^T$ form an orthonormal basis of $\mathbb{R}^3$.

Definition 2.2.1. *If $\mathbf{a} = (a_1, a_2, a_3)^T \in \mathbb{R}^3$ and $\mathbf{b} = (b_1, b_2, b_3)^T \in \mathbb{R}^3$, then their scalar product $\mathbf{a}\cdot\mathbf{b}$ and vector product $\mathbf{a}\times\mathbf{b}$ are*

$$\mathbf{a}\cdot\mathbf{b} = a_1b_1 + a_2b_2 + a_3b_3, \quad \mathbf{a}\times\mathbf{b} = \det\begin{pmatrix} \mathbf{i} & \mathbf{j} & \mathbf{k} \\ a_1 & a_2 & a_3 \\ b_1 & b_2 & b_3 \end{pmatrix}.$$

Definition 2.2.2. *If a map $\phi : M \to N$ is continuous and bijective, and if its inverse map $\psi = \phi^{-1} : N \to M$ is also continuous, then ϕ is a* homeomorphism *and M and N are said to be homeomorphic. A differentiable map $\phi : M \to N$ is a diffeomorphism if there is a differentiable map $\psi : N \to M$ such that $\phi \circ \psi = \mathbb{I}$ (where $\mathbb{I}$ is the identity map) and $\psi \circ \phi = \mathbb{I}$.*

A connected set γ in $\mathbb{R}^3$ is a *regular k-fold continuously differentiable curve* if there is a homeomorphism $\phi : G \to \gamma$, where G is either a line segment $[a, b] \in \mathbb{R}$, or a circle of radius 1, where $\phi \in C^k$, and C^k is the space of k times differentiable functions of a real variable. If $k = 1$, the curve is called smooth. Essentially, a curve is a set of points in $\mathbb{R}^3$ that can be smoothly mapped onto a straight line segment (open curve), or onto a circle (closed curve). Fixing a Cartesian coordinate system (O, x, y, z), a curve is completely determined by the functions $x(t)$, $y(t)$, $z(t)$, where $t \in [a, b] \subset \mathbb{R}$. For a curve to be regular, it is required that the derivatives $x'(t)$, $y'(t)$, $z'(t)$ never vanish simultaneously at any t. In other words, the curve never slows to a stop or backtracks itself. The expressions $x(t)$, $y(t)$, $z(t)$ are called a parametrization of the curve, and t is called the path parameter, or just *parameter*.

Definition 2.2.3. *A continuous curve γ is called* piecewise smooth *if it is smooth, except for perhaps a finite number of isolated points P_i, $i = 1, .., k$.*

Example 2.2.4. *Helix*

A helix in $\mathbb{R}^3$ can be parameterized by a single parameter t as follows

$$\gamma(t) : \mathbf{I} \to \begin{pmatrix} r\cos(\omega t) & r\sin(\omega t) & t \end{pmatrix}^T$$

Example 2.2.5. *Cycloid Curve*

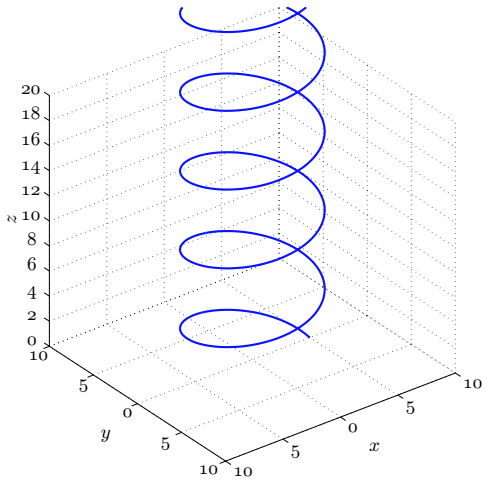


Figure 2.1 A helix, parameterized by a single parameter.

A *cycloid* is the locus of a point on the rim of a circle of radius r rolling along a straight line. The parametric equation for the cycloid is

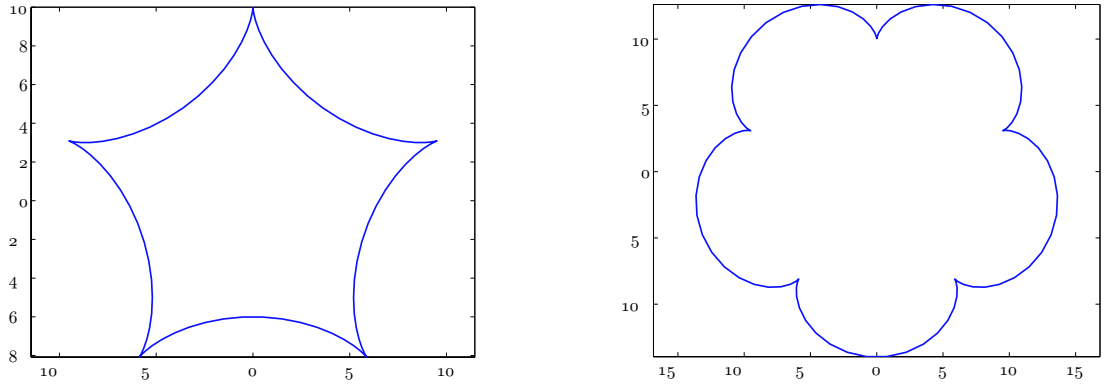
$$\begin{aligned} x &= r(t - \sin t) \\ y &= r(1 - \cos t). \end{aligned}$$

Example 2.2.6. *Hypocycloid*

The trajectory of a point P on a circle of radius r rolling without sliding in the plane around the inside of a larger radius R is called a hypocycloid. The parametrization of such a curve is

$$\begin{aligned} x(t) &= r(m+1)\cos(mt) - rm\cos(mt+t) \\ y(t) &= r(m+1)\sin(mt) - rm\sin(mt+t), \end{aligned}$$

where $m = r/R$ is the *modulus*. By convention $m < 0$ for a hypocycloid. In case of an epicycloid, where the circle of radius r is rolling on the outside of the circle of radius R, $m > 0$ (see Figure 2.2).



(a) Hypocycloid ($m = -1/5$). (b) Epicycloid ($m = 1/5$).

Figure 2.2 Hypocycloid and epicycloid curves.

Example 2.2.7. *Trochoid*

A trochoid is the locus of a point P at a distance R from the center of a circle of radius r rolling on a fixed line (see Figure 2.3). A trochoid has parametric equations

$$\begin{aligned} x(t) &= rt - R\sin(t) \\ y(t) &= r - R\cos(t). \end{aligned}$$

An airplane flying at a constant altitude with a constant turning rate in the presence of constant, uniform wind describes a trajectory that corresponds to a trochoid curve in the plane. In the special case when $r = R$, the trochoid curve is the same as a cycloid curve, corresponding to a situation when the airplane airspeed is equal to the wind speed.

Example 2.2.8. *Clothoid*

A *clothoid* is a curve that has a linearly changing curvature profile. The formula for the Cartesian coordinates along a clothoid is given by the Fresnel integrals [2]

$$\begin{aligned} x(t) &= \int_0^t \sin(\tau^2)d\tau \\ y(t) &= \int_0^t \cos(\tau^2)d\tau. \end{aligned}$$

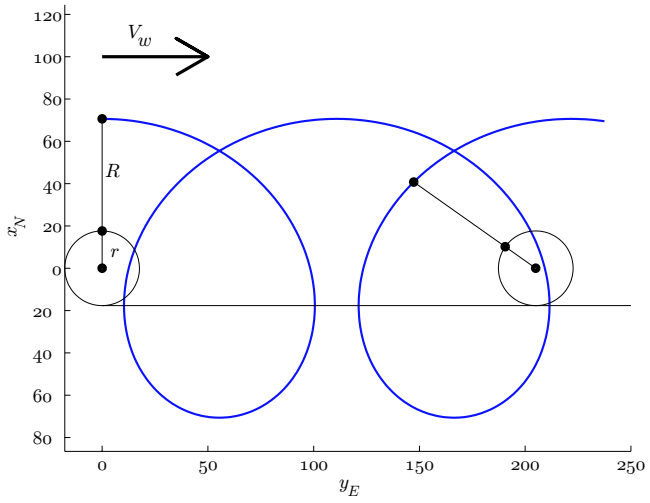


Figure 2.3 Trochoidal path is generated by a point P a distance R from the center of a circle of radius r, as the circle is rolling along a straight line.

Clothoid curves are also known as *Euler spirals* or *Cornu spirals* (see Figure 2.4). A path corresponding to a clothoid curve is generated when a vehicle changes its course of travel such that its angular acceleration is a constant non-zero value. Clothoid curves are important because they form smooth transition segments between straight segments and circular arcs, for example. A vehicle that cannot change its turn rate instantaneously may transition from a straight line to a steady turn by following a clothoid arc until the desired curvature is reached.

Let a smooth curve γ be given by its parametric equations $\mathbf{r}(t) = x(t)\mathbf{i} + y(t)\mathbf{j} + z(t)\mathbf{k}$. The velocity vector of the curve is then $\mathbf{r}'(t) = x'(t)\mathbf{i} + y'(t)\mathbf{j} + z'(t)\mathbf{k}$. The speed of the curve at a point t_0 is defined as $|\mathbf{r}'(t_0)|$.

Definition 2.2.9. *The tangent line of a curve γ at a point $P = \mathbf{r}(t_0)$ is the line drawn through point P in the direction of the velocity vector $\mathbf{r}'(t_0)$.*

The tangent line to a point $P = \mathbf{r}(t_0)$ is the limit of secants to the curve γ that pass through P and an arbitrary point $P_1 = \mathbf{r}(t_1)$, as $t_1 \to t_0$. Denote by $\boldsymbol{\tau}(t_0)$ the unit vector

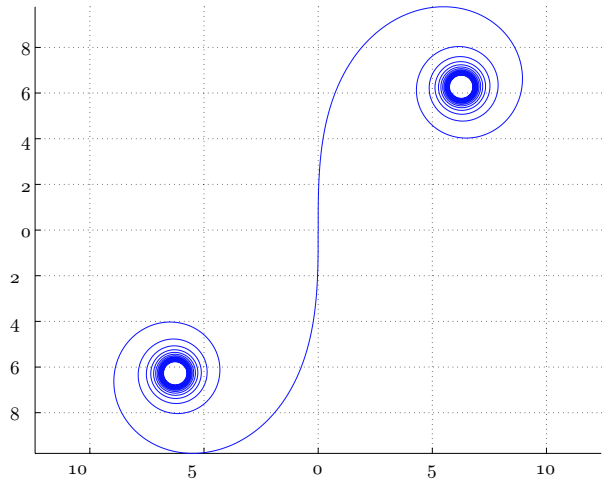


Figure 2.4 A clothoid curve, also known as Euler spiral.

that is parallel to $\mathbf{r}'(t_0)$, that is $\boldsymbol{\tau}(t_0) = \frac{\mathbf{r}'(t_0)}{|\mathbf{r}'(t_0)|}$. A line through P that is orthogonal to the tangent line is called a normal line. At a point $P = \mathbf{r}(t_0)$, where an osculating plane is unique, one may select among all normal directions a unique *principal normal vector*, $\mathbf{n}(t_0)$, which is: (1)orthogonal to the tangent vector $\mathbf{r}'(t_0)$; (2) parallel to the osculating plane; (3) $\mathbf{n}(t_0)$ forms an acute angle with $\mathbf{r}''(t_0)$; (4) $\mathbf{n}(t_0)$ has unit length. The normal vector can be expressed by the formula

$$\mathbf{n} = -\frac{\mathbf{r}' \cdot \mathbf{r}''}{|\mathbf{r}'||\mathbf{r}' \times \mathbf{r}''|}\mathbf{r}' + \frac{|\mathbf{r}'|}{|\mathbf{r}' \times \mathbf{r}''|}\mathbf{r}''.$$

The vector defined by $\boldsymbol{\beta} = \boldsymbol{\tau} \times \mathbf{n}$ is called the *binormal* vector, and the vectors $\boldsymbol{\tau}$, $\mathbf{n}$ and $\boldsymbol{\beta}$ form an orthonormal basis for $\mathbb{R}^3$.

The length of a smooth curve can be written as

$$l(\gamma) = \int_{t_0}^{t_f} |\mathbf{r}'(t)|dt.$$

The arc length of a smooth, regular curve between points $P_0 = \mathbf{r}(t_0)$ and $P = \mathbf{r}(t)$ is

$$s(t) = \int_{t_0}^{t} |\mathbf{r}'(t)|dt,$$

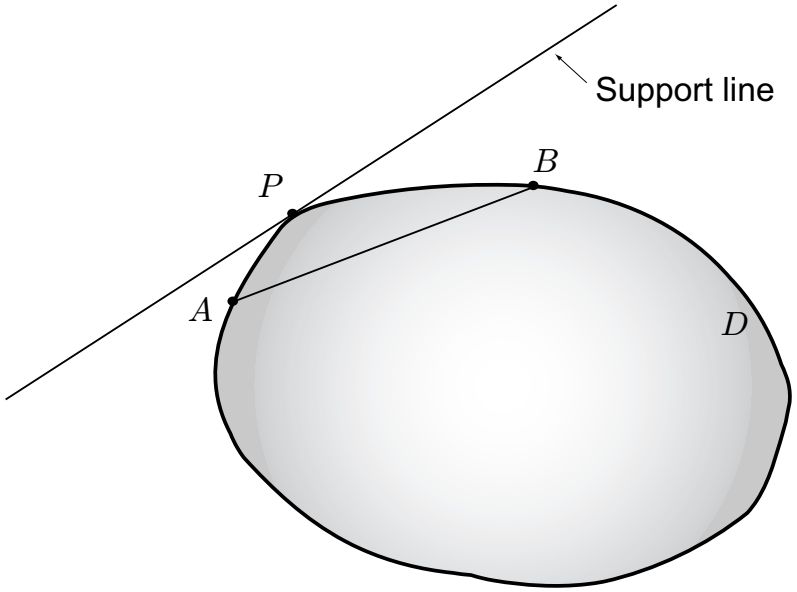


Figure 2.5 Convex region D in the plane.

thus the function $s(t)$ is differentiable, and by the assumption of regularity $s(t) = |\mathbf{r}'(t)| > 0$. By the inverse function theorem there exists an inverse, $t(s)$, such that

$$\frac{dt}{ds} = \frac{1}{|\mathbf{r}'(t(s))|}.$$

Using the inverse function $t(s)$ one may obtain an alternative *arc length parametrization* of the curve γ:

$$\gamma : \mathbf{r}(s) = \mathbf{r}(t(s)).$$

The arc length parametrization of curves has the property that the speed along the curve is unity $|\mathbf{r}'(s)| = 1$, and the formulas for the tangent vector, the principal normal vector and the binormal vector are particularly simple:

$$\boldsymbol{\tau}(s) = \mathbf{r}'(s), \quad \mathbf{n}(s) = \frac{\mathbf{r}''(s)}{|\mathbf{r}''(s)|}, \quad \boldsymbol{\beta}(s) = \frac{\mathbf{r}'(s) \times \mathbf{r}''(s)}{|\mathbf{r}''(s)|}.$$

A closed connected region $D \subset \mathbb{R}^2$ in the plane is called *convex* if for any two points $A, B \in D$, the connecting line segment lies also in D: $AB \subset D$. The boundary of a convex region is called a *convex curve*. An alternative definition of a convex curve is: a curve γ is

convex if each of its points has a support line. A line through a point $P \in \gamma$ is a support line, if γ lies entirely in one of the half planes defined by the line. If a tangent line exists, it is also a support line for a convex curve.

Given two points on a smooth curve $P_0, P_1 \in \gamma$, denote by Δs the arc length P_0P_1 and by $\Delta\theta$ the angle between the tangent vectors $\boldsymbol{\tau}(t_0)$ and $\boldsymbol{\tau}(t_1)$. Then the *curvature* of the curve at point P_0 is defined as

$$\kappa(t_0) = \lim_{P_1 \to P_0} \frac{\Delta\theta}{\Delta s} = \lim_{\Delta s \to 0} \frac{\Delta\theta}{\Delta s}.$$

Let $\gamma \in C^2$ be a regular curve. Then at each of its points the curvature exists. If $\mathbf{r}(t)$ is a regular parametrization of γ, then the curvature is

$$\kappa(t) = \frac{\mathbf{r}' \times \mathbf{r}''}{|\mathbf{r}'|^3}.$$

The curvature of a straight line is $\kappa = 0$, and the curvature of a circle is $\kappa = \frac{1}{R}$, where R is the radius of the circle. There is no plane curve of constant curvature other than the circle and the straight line. Intuitively, the curvature of a curve is the measure of its deviation from a straight line.

Theorem 2.2.10. *[73, Theorem 1.6.3] Let $\kappa(s)$ be an arbitrary continuous function on a line segment $[a, b]$. Then there is a unique (up to a rigid motion) curve γ, for which $\kappa(s)$ is the curvature function and s is the arc length parameter.* □

Plane curves are uniquely determined by their curvature, and the equation $\kappa = \kappa(s)$ is called the natural equation of the curve. The curvature of a plane curve $\gamma : (x, y)^T = (x(t), y(t))^T$ can be written as

$$\kappa = \frac{x'y'' - y'x''}{(x'^2 + y'^2)^{3/2}}.$$

Chapter 3

Minimum-Time Path Planning in Steady Uniform Winds

In this chapter a framework is described for minimum-time path planning in the horizontal plane. We consider a kinematic model for an unmanned aerial vehicle (UAV) flying at constant altitude and constant air-relative speed in a steady, uniform flow-field. In this setting we seek the feasible path that brings the UAV from a given initial point and heading to a given final point and heading in the least amount of time. We assume that the control signal, the turn rate of the vehicle, can only take values in a bounded set to ensure that the resulting path is feasible for real vehicle guidance applications.

Table 3.1 summarizes several minimum-time path-planning problem types in the absence or presence of ambient winds, and with different control bounds. The complexity of the problem increases from the top left to the bottom right. The simplest situation is when there are no bounds on the turn rate, that is, the desired heading/course can be immediately achieved. In the absence of flow this corresponds to a straight line between initial and final points. If there is an external flow-field, the problem is referred to as Zermelo's problem. (One can relax the uniform flow-field assumption in this case.)

Turn Constraint	No Wind	Uniform Wind
None	Straight line	Zermelo's problem [a]
Only Turn Rate	Dubins path	"Convected" Dubins path
Rate and Acceleration	Continuous-Curvature Dubins	Continuous-Curvature, "Convected" Dubins

[a] Zermelo's problem also addresses the case where the external flow is not uniform.

Table 3.1 Path-planning problem types for planar kinematic vehicle models.

If there are bounds on the maximum achievable turn rate, the problem is equivalent to finding the minimum arc-length path of bounded curvature connecting two points in the plane with prescribed initial and final slope. The problem was formulated and studied by Dubins who showed, using geometrical considerations, that a minimum length path contains only maximum-curvature circular arcs and straight segments, and moreover that it contains at most three such segments [20]. The problem has been re-formulated and solved using optimal control theory and Pontryagin's minimum principle in [11]; additional necessary conditions for optimality were provided in [15].

More recently the above methods have been adopted for time-optimal path planning for UAVs traveling in steady uniform winds [42, 41]. In these papers minimum-time trajectories are designed in the air relative frame $\mathcal{F}_A$, an inertial frame that moves in the direction of the ambient wind with the same speed. The desired final point in inertial space corresponds to a point in $\mathcal{F}_A$, a "virtual target," that moves with the same speed as the wind and in the opposite direction. The challenge is to find the point along the virtual target's path at which a Dubins path intercepts the target. The algorithm iteratively solves the Dubins problem in $\mathcal{F}_A$ until the interception error converges to zero. It was

later established that there exists a unique solution for almost every collection of initial and final states (see Theorem 3 in [43]).

An alternative approach is to exploit the geometry of the candidate extremal paths and obtain closed-form solutions. The discussion of the algorithm is the main contribution in this chapter. The key observation is that a UAV flying in a constant ambient wind with a constant maximum turn rate generates a trochoidal path [57], for which closed-form expressions exist. Since extremal trajectories may only contain straight paths and trochoidal segments, one may seek the solution in terms of switching points for which a concatenation of such segments yields a feasible path [71]. Independently of this research, the importance of trochoidal trajectories for minimum-time path planning was recognized in [10]; however, only numerical solutions were presented there.

We provide a detailed description of the minimum-time path planning problem in the plane using trochoidal paths and straight segments. In Section 3.1 we set up the problem and introduce the "trochoidal" frame in which the x-axis is aligned with the direction of the ambient air's motion. The trochoidal coordinates expressed in this frame are an essential part of the development. In Section 3.2 the general character of extremal paths is discussed. We summarize previous results on minimum-time path planning using Pontryagin's minimum principle and provide an additional necessary condition for optimality. Generalizing Dubins' results, we consider only three-segment extremals, which can be grouped into two major categories as shown in Figure 3.1. Borrowing terminology from [67] the "bang-singular-bang" (or "BSB") extremals are those candidate time-optimal paths for which an initial turn is followed by a straight segment followed by a second turn. The solutions for these paths are presented in Section 3.3. When the initial and final turns have the same sense, these paths can be computed on closed form. The results shed light on the character of the BSB paths, in general, and suggest an efficient numerical root-finding routine to obtain the remaining candidate BSB trajectories.

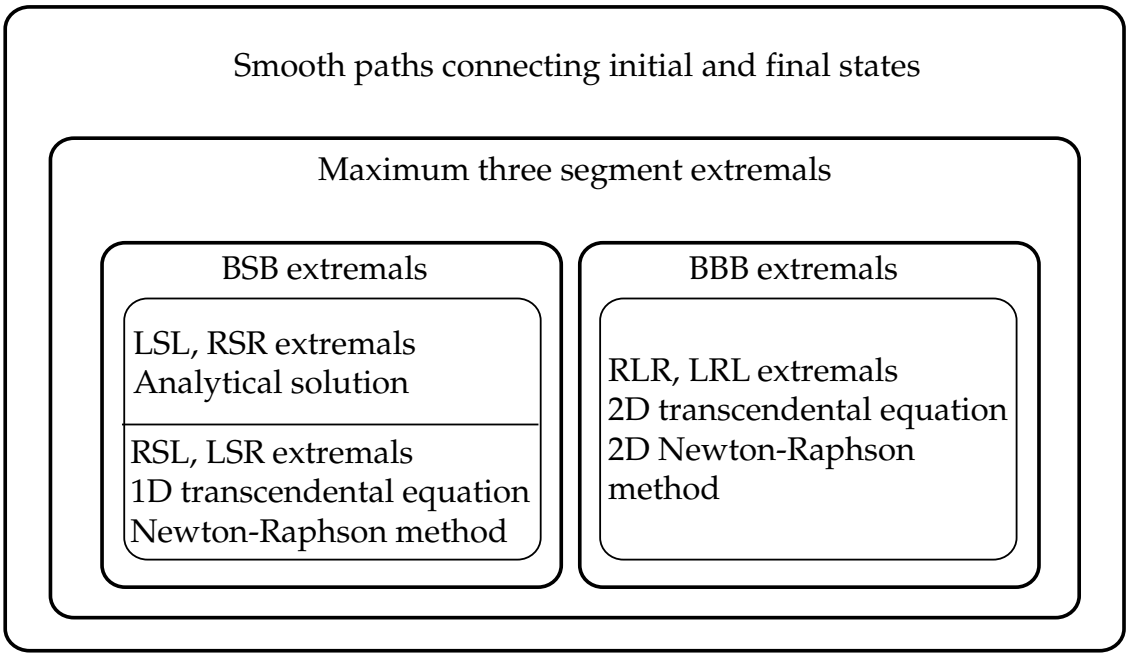


Figure 3.1 Venn diagram showing the types of extremals considered in this work. The abbreviations should be interpreted as follows: As in [67], "B" stands for "bang," or maximum rate turn, "S" stands for "straight" (also stands for "singular"), "L" stands for "left," and "R" stands for "right." For example, LSL is an initial left turn followed by a straight path, then followed by a second left turn.

The "bang-bang-bang" (or "BBB") extremals comprise a turn followed by a second turn in the opposite sense, which is followed by a third turn in the same direction as the first. The solution for these extremals is briefly summarized in Section 3.4 and is described in more detail in Appendix A. The path planning algorithm is demonstrated in Section 3.5, where results are presented for Monte Carlo simulations over a range of randomly selected parameters.

The methods can be generalized to the case, where, in addition to the turn rate limits, the dynamics is also constrained by bounds on the maximum turn acceleration. The approach is described in Section 3.6.

3.1 Problem Formulation

Consider the dynamic system described by the equations

$$\begin{aligned} \dot{x}_N(t) &= V_a \cos\psi(t) + V_x \\ \dot{y}_E(t) &= V_a \sin\psi(t) + V_y \\ \dot{\psi}(t) &= u(t), \end{aligned} \tag{3.1}$$

where the components V_x and V_y of the ambient wind are assumed to be constant. The coordinates $x_N(t)$ and $y_E(t)$ describe the vehicle's position in an inertial frame, while $\psi(t)$ describes its heading measured clockwise from the x_N-axis. V_a is the air speed and $u(t)$ is the turn rate, which we take as a control input.

Suppose the initial and desired terminal conditions are

$$x_N(0) = x_{N_0}, \quad y_E(0) = y_{E_0}, \quad \psi(0) = \psi_0, \tag{3.2}$$

$$x_N(T) = x_{N_f}, \quad y_E(T) = y_{E_f}, \quad \psi(T) = \psi_f. \tag{3.3}$$

The objective is to find an extremal control $u^*(t)$ such that the UAV, starting from its initial state, arrives at the desired final state in minimum time. That is, the objective is to find $u^*(t)$ such that the cost function

$$J = \int_0^T dt = T$$

is minimized subject to the kinematic equations (3.1) and the symmetric control limits

$$-\bar{u} \leq u(t) \leq \bar{u}.$$

Finding minimum-length paths of bounded curvature was studied by Dubins [20] and was more recently adopted as a means of generating minimum-time paths for constant-speed mobile robots with bounded turn rates, leading to the phrase "Dubins' car." (See [11], for example.) One attempt to extend Dubins' results to UAVs in winds is described

in [42, 41], where the problem is transformed to a moving reference frame in which the final position becomes a "virtual moving target" whose speed is equal and opposite to the wind. Path planning is then reduced to a numerical root-finding problem involving iterative solution of the Dubins problem.

This work presents an alternative approach that uses a simple geometric argument to characterize extremal paths. The result relies on the observation that circular (constant turn rate) UAV paths in the air-relative frame correspond to trochoidal paths in the inertial frame [57]. Following [57], we define a trochoidal frame determined by the wind direction, as shown in Figure 3.2.

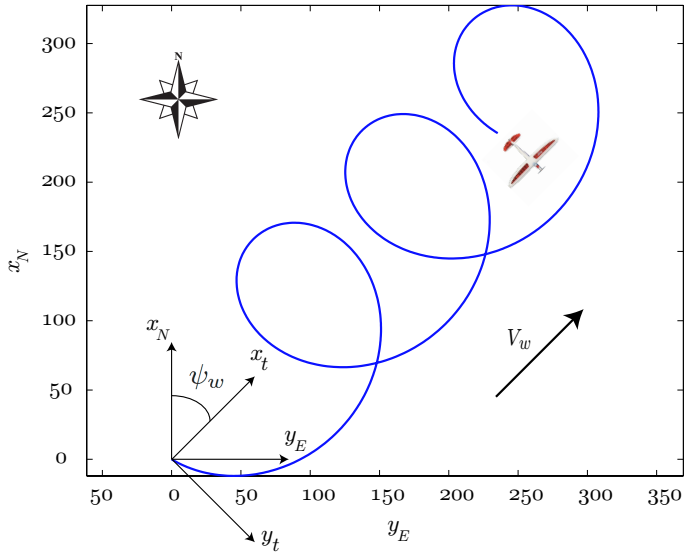


Figure 3.2 Trochoidal path and trochoidal frame.

If χ_w denotes the wind direction (by convention, the direction from which the wind approaches), then $\psi_w = \chi_w \pm \pi$ is the direction of the ambient air's motion. The trochoidal frame, $\mathcal{F}_T$, is then defined such that its x-axis is oriented downwind, the z-axis is into the image, and the y-axis completes the right-handed reference frame. Notice that the trochoidal frame is fixed relative to the ground, while the air relative frame $\mathcal{F}_A$ is convected

downwind. Then the trochoidal coordinates can be found from the inertial coordinates by

$$\begin{pmatrix} x_t(t) \\ y_t(t) \end{pmatrix} = \begin{pmatrix} \cos\psi_w & \sin\psi_w \\ -\sin\psi_w & \cos\psi_w \end{pmatrix} \begin{pmatrix} x_N(t) \\ y_E(t) \end{pmatrix}.$$

The kinematic equations expressed in the trochoidal frame are

$$\dot{x}_t(t) = V_a \cos(\psi(t) - \psi_w) + V_w \tag{3.4}$$

$$\dot{y}_t(t) = V_a \sin(\psi(t) - \psi_w) \tag{3.5}$$

$$\dot{\psi}(t) = u(t) \tag{3.6}$$

where $V_w = \sqrt{V_x^2 + V_y^2}$. We assume that $V_w < V_a$, to ensure that feasible solutions exist. In the case of a turn at constant maximum rate $\omega = |\dot{\psi}_{\max}| = \bar{u}$, the equations can be re-written as

$$\dot{x}_t(t) = V_a \cos(\delta\omega t + \phi_t) + V_w \tag{3.7}$$

$$\dot{y}_t(t) = V_a \sin(\delta\omega t + \phi_t), \tag{3.8}$$

where $\phi_t = \psi(0) - \psi_w$ and $\delta \in \{-1, 1\}$ describes the direction of the turn. The position of a point on the trochoidal path can then be written as

$$x_t(t) = \frac{V_a}{\delta\omega} \sin(\delta\omega t + \phi_t) + V_w t + x_{t_0} \tag{3.9}$$

$$y_t(t) = -\frac{V_a}{\delta\omega} \cos(\delta\omega t + \phi_t) + y_{t_0}. \tag{3.10}$$

The trochoidal path defined here is essential in developing the path planning algorithm described in later sections.

3.2 Properties of Extremal Paths

In this section necessary conditions are derived for time-optimality. Following [11], we begin with equations (3.1) and initial conditions (3.2). Assume that the coordinates are

already expressed in the trochoidal frame such that $V_y = 0$ and $V_x = V_w$.

Let $\boldsymbol{\lambda} = [\lambda_1, \lambda_2, \lambda_3]^T$ denote the vector of adjoint variables (the co-states). The Hamiltonian for the time-optimal control problem is

$$\mathcal{H} = \lambda_0 + \lambda_1(V_a \cos\psi(t) + V_w) + \lambda_2 V_a \sin\psi(t) + \lambda_3 u,$$

where $u \in \mathcal{U}$ and $\mathcal{U} = [-\bar{u}, \bar{u}]$ is the set of admissible controls. The co-state equations are

$$\begin{aligned} \dot{\lambda}_1 &= 0 \\ \dot{\lambda}_2 &= 0 \\ \dot{\lambda}_3 &= \lambda_1 V_a \sin\psi(t) - \lambda_2 V_a \cos\psi(t) \end{aligned}$$

which implies that λ_1 and λ_2 are constant. The minimum principle — Theorem 2.1.1 — states that along an extremal trajectory the following conditions need to be satisfied

- $$\mathcal{H}(\boldsymbol{x}^*(t), \boldsymbol{\lambda}^*(t), u^*(t)) = \inf_{u(t)\in\mathcal{U}} \{\mathcal{H}(\boldsymbol{x}^*(t), \boldsymbol{\lambda}^*(t), u(t))\}$$
- $$\mathcal{H}(\boldsymbol{x}^*, \boldsymbol{\lambda}^*, u^*(t)) \equiv 0 \quad \text{and} \quad \lambda_0 \geq 0.$$

Introducing the change of variables

$$\begin{aligned} \lambda_1 &= \bar{\lambda}\cos\psi_c \\ \lambda_2 &= \bar{\lambda}\cos\psi_c, \end{aligned}$$

where $\bar{\lambda} = \sqrt{\lambda_1^2 + \lambda_2^2}$, we can write the Hamiltonian in the form

$$\mathcal{H} = \lambda_0 + V_a\bar{\lambda}\cos(\psi - \psi_c) + \lambda_1 V_w + \lambda_3 u. \tag{3.11}$$

The third co-state equation takes the form

$$\dot{\lambda}_3 = V_a\bar{\lambda}\sin(\psi - \psi_c). \tag{3.12}$$

Following [11], the following two cases are possible:

- $\frac{\partial \mathcal{H}}{\partial u} = \lambda_3 = 0$. Then $\dot{\lambda}_3 = 0$. In this case first consider when $\bar{\lambda} \neq 0$. Then by equation (3.12) we have that $\psi(t) = \psi_c = \text{const}$ or $\psi(t) = \psi_c + \pi = \text{const}$, i.e. the path is a line segment in the direction defined by the characteristic direction ψ_c. Now consider the case when $\bar{\lambda} \equiv 0$. Then $\lambda_1 = \lambda_2 = \lambda_3 = 0$, hence the Hamiltonian in equation (3.11) is $\mathcal{H} = \lambda_0 \geq 0$. However, by the minimum principle we also have that $\mathcal{H} = 0$, which implies that $\lambda_0 = 0$. Theorem 2.1.1 states that λ_0 and $\boldsymbol{\lambda}(t)$ cannot be all zero at the same time, so this situation is not possible by the minimum principle.

- $\frac{\partial \mathcal{H}}{\partial u} = \lambda_3 \neq 0$. In this case the control signal has to be $u = -\text{sign}(\lambda_3)\bar{u}$ to minimize the Hamiltonian. This corresponds to a maximum rate turn to the left or right, a "maximum effort" or "bang-bang" control.

Thus, the time-optimal path consists of turns at maximum rate and straight segments. Extremal trajectories (or extremals) therefore comprise trochoidal and straight segments. In the absence of winds ($V_w = 0$), these trajectories simplify to circular segments and straight lines, and they are often referred to as Dubins paths. As proved in [20], in that case there are six possibilities to connect initial and final oriented points with minimum arc-length paths. (For constant speed motion "minimum-time" is equivalent to "minimum arc-length.") Let L denote a segment corresponding to a maximum rate turn to the *left* and let R denote a segment corresponding to a maximum rate turn to the *right*. Finally, let S denote a *straight* segment (which is a *singular* arc of an extremal path). Generally, the six possible extremal paths are LSL, RSR, LSR, RSL, RLR and LRL. Finding these candidate paths is a simple geometric exercise: one only needs to find tangents to oriented circles of minimum turn radius. The LRL and RLR paths only exist if the distance between the initial and final points, d, satisfies $d \leq 4R_0$, where R_0 is the maximum rate turning radius. The paths LSR and RSL fail to exist when $d < 2R_0$. In case of equality,

$d = 2R_0$, the intermediate straight segment vanishes [42].

Generalizing Dubins' results to the case where winds are present, we seek trajectories that can be constructed by joining three segments according to BSB or BBB (where B can be either L or R). In Section 3.3 a method is presented to compute BSB trajectories. The LSL and RSR trajectories can be found in closed form; the LSR and RSL trajectories can be found using a simple root-finding method. In Section 3.4 a method to find the BBB trajectories is discussed. The true minimum-time solution is found by comparing the transit times for all candidates. The numerical root-finding algorithm described in this chapter differs from earlier numerical methods: the framework presented in this work yields closed-form expressions in the form of one or two transcendental equations that have to be solved for one or two unknown parameters, respectively. The known form of the transcendental equations and the insight gained through the development of the closed-form expressions allow one to formulate a numerical analysis problem that can be solved efficiently for these unknown parameters.

The number of possible solutions can be confined to a small finite set by introducing a new necessary condition for optimality. Towards that end we express equations (3.4)-(3.6) using the inertial "ground speed" and "course angle" instead of the air relative speed and heading angle:

$$\dot{x}_N(t) = V_g(\psi)\cos\left(\chi(\psi)\right) \tag{3.13}$$

$$\dot{y}_E(t) = V_g(\psi)\sin\left(\chi(\psi)\right) \tag{3.14}$$

$$\dot{\psi}(t) = u(t), \tag{3.15}$$

where the ground speed $V_g(\psi)$ and the course angle $\chi(\psi)$ depend on the heading angle ψ as follows

$$V_g(\psi) = \sqrt{V_a^2 + V_w^2 + 2V_aV_w\cos(\psi)} \tag{3.16}$$

$$\chi(\psi) = \tan^{-1}\left(\frac{V_a\sin\psi}{V_a\cos\psi + V_w}\right). \tag{3.17}$$

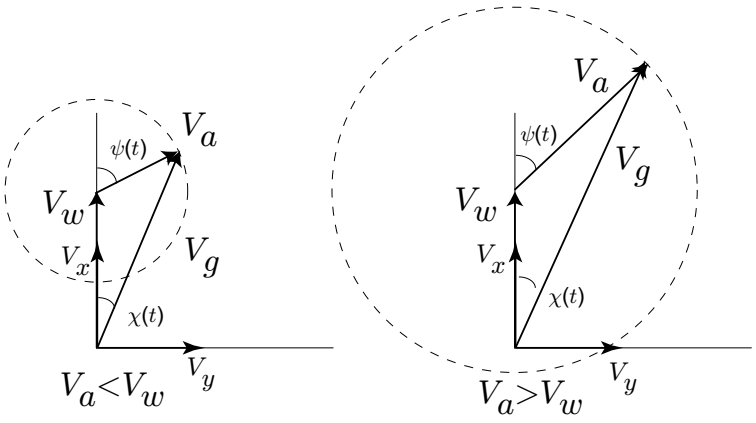


Figure 3.3 Relationship between air speed, wind speed and groundspeed.

Here the 4-quadrant inverse tangent function is used. The relationship between course angle and heading angle is illustrated in Figure 3.3.

Lemma 3.2.1. *Consider the kinematic equations (3.13)-(3.15). The heading angle* $\psi(t) \in [0, 2\pi)$ *and course angle* $\chi(\psi(t)) \in [0, 2\pi)$ *are in one-to-one correspondence if and only if* $V_a > V_w$.

Proof: If $V_a \leq V_w$, then the mapping from $\psi(t) \in [0, 2\pi)$ to $\chi(\psi(t))$ is not one-to-one. Referring to the left side of Figure 3.3, the course angle $\chi(\psi(t))$ will only take values for which the ground relative velocity vector $V_g(\psi(t))$ has a non-negative 'x' component. It follows that $\chi(\psi(t)) \in [-\frac{\pi}{2}, \frac{\pi}{2}]$ for $\psi(t) \in [0, 2\pi)$.

If $V_a > V_w$, then from equation (3.17) we can conclude that the denominator of the argument of the arctan function takes values in $[-V_a + V_w, V_a + V_w]$. The numerator takes values in $[-V_a, V_a]$. Moreover, $\chi(\psi(t))$ takes a unique value in $[0, 2\pi)$ for each $\psi(t) \in [0, 2\pi)$, as can be seen in the right side of Figure 3.3. □

Lemma 3.2.2. *Consider an extremal that contains a trochoidal segment, described by equations (3.9)-(3.10). Assume that the trochoidal segment begins at point* P_0 *at path parameter value* t_0 *and ends at point* P_3 *corresponding to path parameter value* t_3. *A necessary condition for optimality is that* $\Delta t = (t_3 - t_0) < 2t_{2\pi}$, *where* $t_{2\pi} = \frac{2\pi}{\omega} = \frac{2\pi}{|\dot{\psi}_{\max}|}$ *is the time required for the air relative velocity*

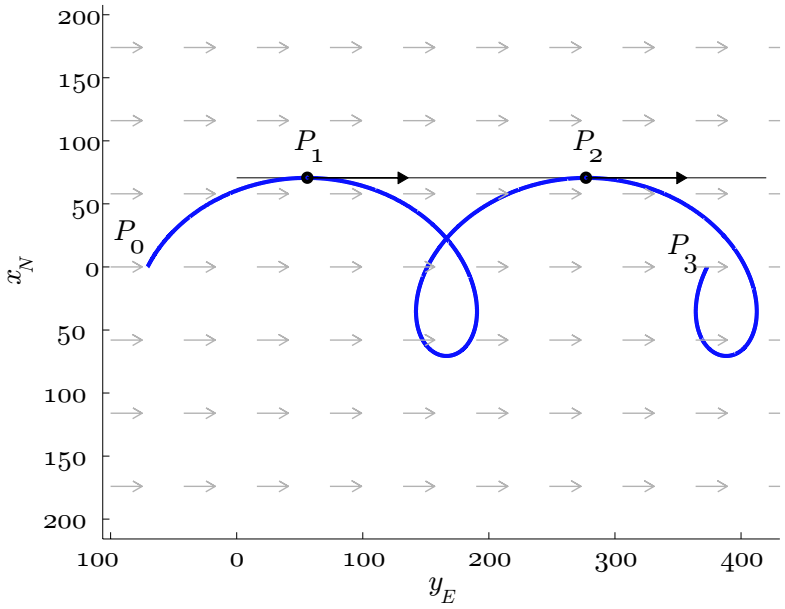


Figure 3.4 The time the UAV spends on an extremal trochoidal segment has to be less than $2t_{2\pi}$.

vector to describe a full revolution.

Proof: To prove by contradiction, assume that there exists a solution for which $\Delta t \geq 2t_{2\pi}$, corresponding to an optimal trajectory. In that case the trochoidal path contains at least two "full loops." By Lemma 3.2.1, there exists a point P_1 and corresponding path parameter value t_1, $t_0 \leq t_1 \leq t_0 + t_{2\pi}$, such that the course angle along the path is aligned with the ambient wind's motion at point P_1. Define the point P_2 to correspond to path parameter value t_2, where $t_2 = t_1 + t_{2\pi}$. The inertial velocities at points P_1 and P_2 align with the wind vector and with the line segment connecting the two points by equations (3.7)-(3.10), therefore the trochoidal segment between points P_1 and P_2 can be replaced by a straight path in the direction of the ambient wind's motion (see Figure 3.4). The distance from point P_1 to P_2 along the trochoidal segment is $D = V_w t_{2\pi} = \frac{V_w}{V_a} 2\pi R_0$ where $R_0 = \frac{V_a}{\omega}$ is the (constant) minimum turn radius in the air relative frame. The time it takes to travel

on the straight line instead of the trochoid is $t_s = \frac{D}{V_w+V_a}$, since the tangent line connecting points P_1 and P_2 has to be aligned with the wind vector. Thus $t_s = \frac{V_w}{V_a+V_w} t_{2\pi} < t_{2\pi}$, consequently the trochoidal segment can be replaced by the shorter straight segment. Since a trajectory can only be optimal, if all sub-trajectories are optimal as well, it follows that the original trochoidal path cannot be optimal, thus $\Delta t < 2t_{2\pi}$ is a necessary condition for optimality. □

3.3 Computing BSB Trajectories

Given the initial and final conditions (3.2) and (3.3), we define two trochoids $(x_{t_1}(t), \quad y_{t_1}(t))^T$ and $(x_{t_2}(t), \quad y_{t_2}(t))^T$ as in (3.9) and (3.10), such that the first trochoid satisfies the initial conditions (3.2) at $t = 0$, and the second trochoid satisfies the final conditions (3.3) at $t = t_{2\pi} = \frac{2\pi}{\omega}$, the time required for the air-relative velocity vector to describe a full circle at the maximum turn rate.

Remark 3.3.1. *There is a slight abuse of notation here, in using the same path parameter t for both curves.*

Let δ_i denote the sense of the turn for $i \in \{1, 2\}$. We seek the four feasible paths of shortest length corresponding to each of the four possible turn sequences: $\delta_1 \in \{-1, 1\}$ and $\delta_2 \in \{-1, 1\}$.

$$x_{t_1}(t) = \frac{V_a}{\delta_1 \omega} \sin(\delta_1 \omega t + \phi_{t_1}) + V_w t + x_{t_{10}} \tag{3.18}$$

$$y_{t_1}(t) = \frac{-V_a}{\delta_1 \omega} \cos(\delta_1 \omega t + \phi_{t_1}) + y_{t_{10}} \tag{3.19}$$

$$x_{t_2}(t) = \frac{V_a}{\delta_2 \omega} \sin(\delta_2 \omega t + \phi_{t_2}) + V_w t + x_{t_{20}} \tag{3.20}$$

$$y_{t_2}(t) = \frac{-V_a}{\delta_2 \omega} \cos(\delta_2 \omega t + \phi_{t_2}) + y_{t_{20}} \tag{3.21}$$

Let us denote the point at which the extremal path leaves the first trochoidal segment as point P_A, and the point where it reaches the second trochoidal segment as point P_B; see Figure 3.5. Define t_A and t_B, relative to the two trochoidal segments, such that

$$\begin{pmatrix} x_A \\ y_A \end{pmatrix} = \begin{pmatrix} x_{t_1}(t_A) \\ y_{t_1}(t_A) \end{pmatrix} \quad \text{and} \quad \begin{pmatrix} x_B \\ y_B \end{pmatrix} = \begin{pmatrix} x_{t_2}(t_B) \\ y_{t_2}(t_B) \end{pmatrix}.$$

We pick the phase angles ϕ_{t_1} and ϕ_{t_2} such that the first trochoid has the desired initial heading ψ_0 at $t = 0$ and the second trochoid has the desired final heading ψ_f at $t = t_{2\pi}$:

$$\phi_{t_1} = \psi_0 - \psi_w, \qquad \phi_{t_2} = \psi_f - \psi_w - \delta_2 \omega t_{2\pi}.$$

Similarly, we can pick the constants

$$x_{t_{10}} = x_0 - V_a/(\delta_1 \omega)\sin(\phi_{t_1}) \tag{3.22}$$

$$y_{t_{10}} = y_0 + V_a/(\delta_1 \omega)\cos(\phi_{t_1}) \tag{3.23}$$

$$x_{t_{20}} = x_f - V_a/(\delta_2 \omega)\sin(\delta_2 \omega t_{2\pi} + \phi_{t_2}) - V_w t_{2\pi} \tag{3.24}$$

$$y_{t_{20}} = y_f + V_a/(\delta_2 \omega)\cos(\delta_2 \omega t_{2\pi} + \phi_{t_2}), \tag{3.25}$$

such that the first trochoid satisfies the initial conditions $[x_{t_1}, y_{t_1}]^T\big|_{t=0} = [x_0, y_0]^T$, and the second trochoid satisfies the final condition $[x_{t_2}, y_{t_2}]^T\big|_{t=t_{2\pi}} = [x_f, y_f]^T$.

Remark 3.3.2. *Here, we assume that the initial and final conditions given in equations (3.2)-(3.3) have been re-expressed in the trochoidal frame and denoted as* $[x_0, y_0]^T$ *and* $[x_f, y_f]^T$.

With the parameter definitions above and referring to Lemma 3.2.2, we seek the path parameters $t_A, \in [0, 2t_{2\pi})$ and $t_B \in (-t_{2\pi}, t_{2\pi}]$. The necessary conditions can be summarized as follows:

- The velocities at point A and point B must be equal:

$$(\dot{x}_{t_1}(t_A), \quad \dot{y}_{t_1}(t_A))^T = (\dot{x}_{t_2}(t_B), \quad \dot{y}_{t_2}(t_B))^T. \tag{3.26}$$

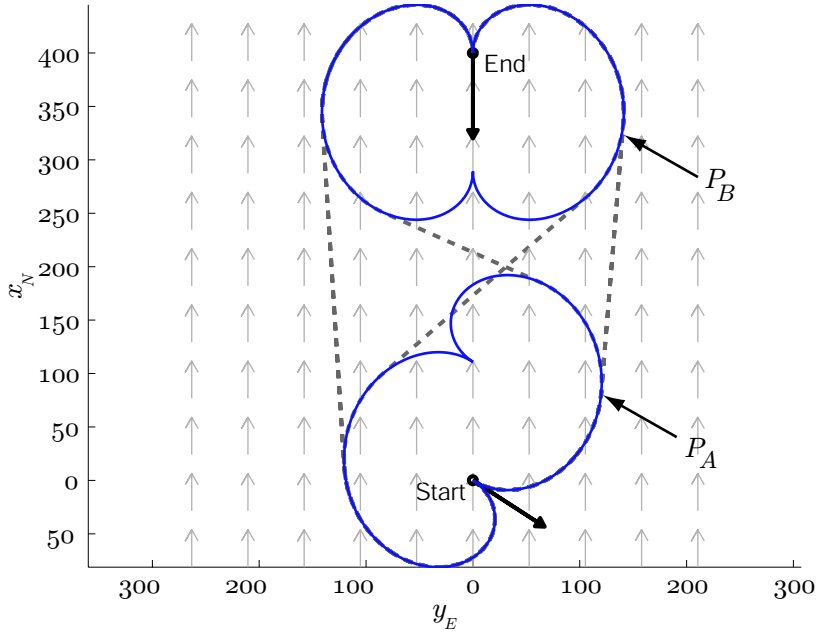


Figure 3.5 The dashed lines show candidate extremal paths from initial state to final state. The solid lines are the trochoidal segments at the initial and final states plotted for $t \in [0, t_{2\pi}]$.

- The line segment joining the points P_A and P_B must be tangent with the velocity vectors at both points:

$$\tan(\alpha) = \frac{y_{t_2}(t_B) - y_{t_1}(t_A)}{x_{t_2}(t_B) - x_{t_1}(t_A)} \tag{3.27}$$

$$= \frac{\dot{y}_{t_2}(t_B)}{\dot{x}_{t_2}(t_B)} = \frac{\dot{y}_{t_1}(t_A)}{\dot{x}_{t_1}(t_A)}. \tag{3.28}$$

- The path parameters must satisfy

$$t_A \in [0, 2t_{2\pi}), \quad t_B \in (-t_{2\pi}, t_{2\pi}]. \tag{3.29}$$

Remark 3.3.3. *For the case in which there is no wind ($V_w = 0$), one only needs to consider $t_A, t_B \in [0, t_{2\pi}]$, since $t_{2\pi}$ is the time it takes for the UAV to arrive back to its starting point. As shown in 3.3.3, this is not the case if winds are present.*

Condition (3.26) is equivalent to the condition

$$\delta_1 \omega t_A + \phi_{t_1} = \delta_2 \omega t_B + \phi_{t_2} + 2k\pi, \quad k \in \mathbb{Z},$$

where $\mathbb{Z}$ is the set of real integers, and we can express t_B as a function of t_A:

$$t_B = \frac{\delta_1}{\delta_2} t_A + \frac{\phi_{t_1} - \phi_{t_2} + 2k\pi}{\delta_2 \omega}, \quad k \in \mathbb{Z}. \tag{3.30}$$

Substituting equation (3.65) into equation (3.27) we get

$$\tan(\alpha) = \frac{(y_{t_{20}} - y_{t_{10}}) + \frac{\delta_2 - \delta_1}{\delta_1 \delta_2 \omega} V_a \cos\left(\delta_1 \omega t_A + \phi_{t_1}\right)}{\frac{\delta_1 - \delta_2}{\delta_1 \delta_2 \omega} V_a \sin\left(\delta_1 \omega t_A + \phi_{t_1}\right) + (x_{t_{20}} - x_{t_{10}}) + V_w \left(\left(\frac{\delta_1}{\delta_2} - 1\right) t_A + \frac{\phi_{t_1} - \phi_{t_2} + 2k\pi}{\delta_2 \omega}\right)}. \tag{3.31}$$

Because of condition (3.28), the tangent slope can also be expressed as

$$\tan(\alpha) = \frac{V_a \sin(\delta_1 \omega t_A + \phi_{t_1})}{V_a \cos(\delta_1 \omega t_A + \phi_{t_1}) + V_w}.$$

Rearranging and using the identity $\sin^2 \alpha + \cos^2 \alpha = 1$, one gets the following implicit equation for t_A:

$$E \cos(\delta_1 \omega t_A + \phi_{t_1}) + F \sin(\delta_1 \omega t_A + \phi_{t_1}) = G, \tag{3.32}$$

where

$$E = V_a \left(V_w \frac{\delta_1 - \delta_2}{\delta_1 \delta_2 \omega} - (y_{t_{20}} - y_{t_{10}}) \right) \tag{3.33}$$

$$F = V_a \left((x_{t_{20}} - x_{t_{10}}) + V_w \left(t_A \left(\frac{\delta_1}{\delta_2} - 1 \right) + \frac{\phi_{t_1} - \phi_{t_2} + 2k\pi}{\delta_2 \omega} \right) \right) \tag{3.34}$$

$$G = V_w (y_{t_{20}} - y_{t_{10}}) + \frac{V_a^2 (\delta_2 - \delta_1)}{\delta_1 \delta_2 \omega}. \tag{3.35}$$

Equation (3.32) has one unknown: t_A. Having found t_A, one may calculate t_B using (3.65). Since (3.32) is transcendental, t_A must be found numerically, in general. However, the problem is significantly simplified if one assumes that

$$\text{sign}(\delta_1) = \text{sign}(\delta_2),$$

that is, that the two trochoids have the same sense.

3.3.1 Closed-form Solution for LSL and RSR Trajectories

In the case that $\text{sign}(\delta_1) = \text{sign}(\delta_2)$, the expression for t_B in (3.65) becomes

$$t_B = t_A + \frac{\phi_{t_1} - \phi_{t_2} + 2k\pi}{\delta_2 \omega}, \quad k \in \mathbb{Z}. \tag{3.36}$$

Then equation (3.31) can be written as

$$\begin{aligned}
\tan(\alpha) &= \frac{-\frac{V_a}{\delta_2\omega}\sin(\delta_2\omega t_B + \phi_{t_2}) + y_{t_{20}} - \left(-\frac{V_a}{\delta_1\omega}\sin(\delta_1\omega t_A + \phi_{t_1}) + y_{t_{10}}\right)}{\frac{V_a}{\delta_2\omega}\sin(\delta_2\omega t_B + \phi_{t_2}) + y_{t_{20}} + V_w t_B - \left(\frac{V_a}{\delta_1\omega}\sin(\delta_1\omega t_A + \phi_{t_1}) + y_{t_{10}} + V_w t_A\right)} \\
&= \frac{y_{t_{20}} - y_{t_{10}}}{x_{t_{20}} - x_{t_{10}} + V_w \frac{\phi_{t_1} - \phi_{t_2} + 2k\pi}{\delta_2\omega}},
\end{aligned}$$

where $k \in \mathbb{Z}$. Using equation (3.26), one obtains the following simplified form of equation (3.32):

$$V_a \tan(\alpha)\cos(\delta_1\omega t_A + \phi_{t_1}) - V_a \sin(\delta_1\omega t_A + \phi_{t_1}) = -V_w \tan(\alpha). \tag{3.37}$$

To solve equation (3.37) for t_A, note that it can also be written in the form

$$-\frac{V_a}{\cos\alpha}\sin(\beta - \alpha) = -V_w \tan(\alpha)$$

where $\beta = \delta_1\omega t_A + \phi_{t_1}$, and where we used the identity $\sin(\beta - \alpha) = \sin\beta\cos\alpha - \cos\beta\sin\alpha$. Solving the equation above for β gives

$$\beta = \sin^{-1}\left(\frac{V_w}{V_a}\sin(\alpha)\right) + \alpha.$$

Substituting back into the previous definition for β, with $\omega = \frac{2\pi}{t_{2\pi}}$, gives

$$t_A = \frac{t_{2\pi}}{\delta_1 2\pi}\left[\sin^{-1}\left(\frac{V_w}{V_a}\sin(\alpha)\right) + \alpha - \phi_{t_1} + 2m\pi\right], \quad m \in \mathbb{Z}. \tag{3.38}$$

Using equation (3.36) one may also find t_B:

$$t_B = t_A + \frac{\phi_{t_1} - \phi_{t_2} + 2k\pi}{\delta_2 \omega}, \quad k \in \mathbb{Z}. \tag{3.39}$$

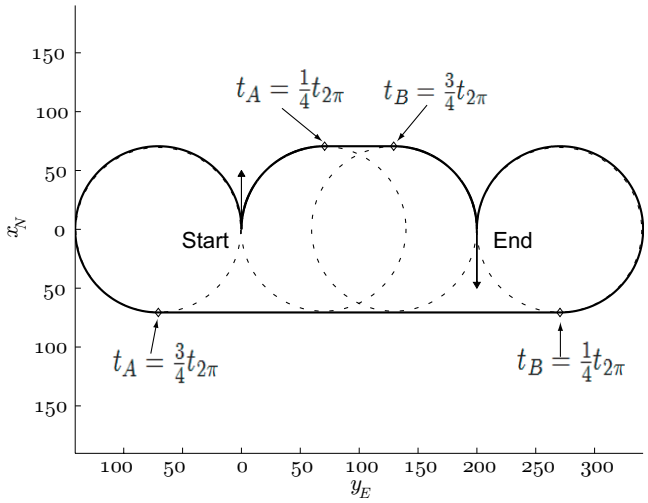
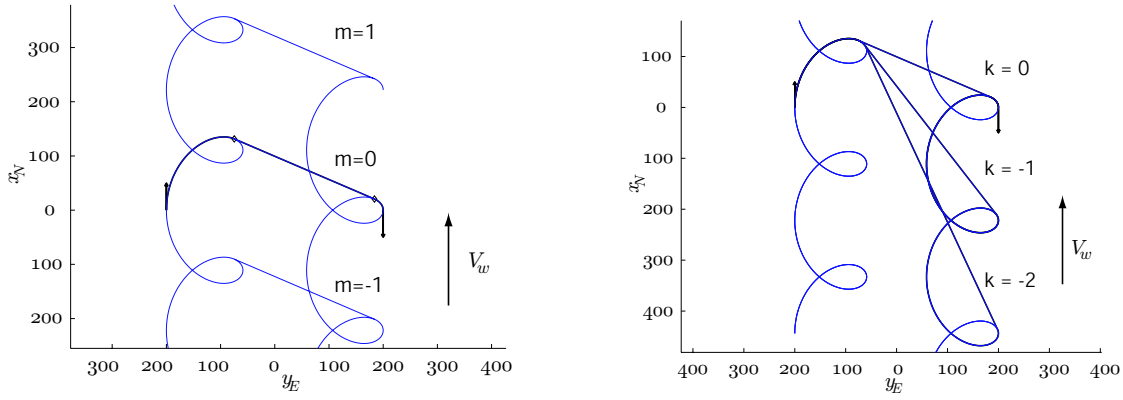


Figure 3.6 Candidate extremals for $V_w = 0$. The figure illustrates how t_A is obtained by scaling $t_{2\pi}$ with $\frac{\alpha}{2\pi} \in [0, 1]$. In the above case $\alpha = \frac{\pi}{2}$. When turning "left," $\delta_1 = -1$, so $t_A = -\frac{1}{4}t_{2\pi} \equiv \frac{3}{4}t_{2\pi}$.

Example 3.3.4. *To better understand the result above, consider a simple case in which there is no wind ($V_w = 0$). Consider the following initial and final conditions*

$$x_{N_0} = 0, \quad y_{E_0} = 0, \quad \psi_0 = 0,$$
$$x_{N_f} = 0, \quad y_{E_f} = 200, \quad \psi_f = \pi.$$

Take $k = m = 0$ so that $t_A = t_{2\pi}\frac{[0+\alpha+0]}{2\pi}$. (The meaning of the integers k and m will be discussed presently.) The parameter α is the slope of the tangent line connecting the two trochoids, and is measured from the same axis as the heading angle. In this example the value of α is $\pi/2$ for both the LSL and RSR extremals, as shown in Figure 3.6. One interpretation of equation (3.38) is that t_A is obtained by scaling $t_{2\pi}$ with $\frac{\alpha}{2\pi} \in [0, 1]$. In Figure 3.6, the RSR solution is then a "quarter turn" to the right, $t_A = \frac{1}{4}t_{2\pi}$, followed by a straight segment, followed by another quarter turn to the right. (Note that $t_B = \frac{3}{4}t_{2\pi}$; recall that we measure the path parameter for the second trochoidal segment backward *from this segment's end time at $t_{2\pi}$.)*



(a) The effect of $m \in \mathbb{Z}$.

(b) The effect of $k \in \mathbb{Z}$.

Figure 3.7 The effect of $m \in \mathbb{Z}$ is a parallel shift with respect to the feasible solution. The effect of $k \in \mathbb{Z}$ is a variety of different tangent lines connecting the initial and final trochoids.

Selecting "m" and "k".

Solutions for different values $m \in \mathbb{Z}$ correspond to parallel shifts of the path corresponding to $m = 0$; see Figure 3.7(a). By equation (3.38), however, it is easy to see that there are only two values of m, for which $t_A \in [0, 2t_{2\pi})$. It is convenient to take t_A from equation (3.38) modulo $t_{2\pi}$ to obtain one of the solutions, call it t_{A_1}; the other solution is then $t_{A_2} = t_{A_1} + t_{2\pi}$. Consequently, we only consider $m = 0$.

The value of k affects the tangent angle α:

$$\alpha = \tan^{-1}\left(\frac{y_{t_{20}} - y_{t_{10}}}{x_{t_{20}} - x_{t_{10}} + V_w \frac{\phi_{t_1} - \phi_{t_2} + 2k\pi}{\delta_2 \omega}}\right), \quad k \in \mathbb{Z}.$$

This results in different solutions that connect the initial trochoid with a "different loop" of the final trochoid; see Figure 3.7(b). Assuming that $\phi_{t_1} - \phi_{t_2}$ is taken modulo 2π, it follows immediately from equation (3.36) that k can only take values in a finite set to ensure that $t_B \in (-t_{2\pi}, t_{2\pi}]$. For example, if $t_{A_1} \in [0, t_{2\pi})$, and $\delta_2 = 1$, this set is $k \in \{-2, -1, 0, 1\}$.

To obtain the most conservative bound on k, first consider $\delta_2 = 1$. In an imaginary worst case, when $t_A = 2t_{2\pi}$ and $\phi_{t_1} - \phi_{t_2} = 2\pi$, any $k < -3$ results in $t_B \leq -t_{2\pi}$, so one only needs to consider $k \geq -3$. On the other hand, when $t_A = 0$ and $\phi_{t_1} - \phi_{t_2} = 0$, any $k > 1$ yields $t_B > t_{2\pi}$, hence k has to take values in the set $k \in \{-3, -2, -1, 0, 1\}$. Similarly, when $\delta_2 = -1$, k has to take values in $k \in \{-2, -1, 0, 1, 2\}$. The number of solutions that one needs to consider depends on t_A, δ_2 and $\phi_{t_1} - \phi_{t_2}$. In practice, it would be unnecessary to check the solutions for every value $k \in \{-3, \ldots, 2\}$. For the sake of generality, however, and because the solutions are trivial to compute, we consider all six possibilities in the following proposition.

Proposition 3.3.5. *Define the path $\gamma(t)$, $t \in [0, T]$, such that*

$$\gamma(t) = \begin{pmatrix} x_{t_1}(t) \\ y_{t_1}(t) \end{pmatrix} \qquad t \in [0, t_A]$$

$$\gamma(t) = \begin{pmatrix} x_{t_1}(t_A) + \dot{x}_{t_1}(t_A)(t - t_A) \\ y_{t_1}(t_A) + \dot{y}_{t_1}(t_A)(t - t_A) \end{pmatrix} \qquad t \in [t_A, t_\beta]$$

$$\gamma(t) = \begin{pmatrix} x_{t_2}(t - t_\beta + t_B) \\ y_{t_2}(t - t_\beta + t_B) \end{pmatrix} \qquad t \in [t_\beta, T],$$

where

$$\begin{aligned} t_\beta &= t_A + \frac{\sqrt{(x_{t_2}(t_B) - x_{t_1}(t_A))^2 + (y_{t_2}(t_B) - y_{t_1}(t_A))^2}}{\sqrt{\dot{x}_{t_2}(t_B)^2 + \dot{y}_{t_2}(t_B)^2}} \\ T &= t_\beta + (t_{2\pi} - t_B), \end{aligned}$$

and

$$\alpha = \tan^{-1}\left(\frac{y_{t_{20}} - y_{t_{10}}}{x_{t_{20}} - x_{t_{10}} + V_w \frac{\phi_{t_1} - \phi_{t_2} + 2k\pi}{\delta_2 \omega}}\right),$$

$$t_A = \frac{t_{2\pi}}{\delta_1 2\pi}\left[\sin^{-1}\left(\frac{V_w}{V_a}\sin(\alpha)\right) + \alpha - \phi_{t_1}\right],$$

$$t_B = t_A + \frac{\phi_{t_1} - \phi_{t_2} + 2k\pi}{\delta_2 \omega}, \quad k \in \{-3, -2, -1, 0, 1, 2\}.$$

The path $\gamma(t)$ satisfies the necessary conditions for time-optimality. □

Remark 3.3.6. *The final path that is selected by the algorithm satisfies the necessary conditions for optimality; however, one may not conclude optimality, in general, as there are additional extremal paths to consider – those for which the two trochoidal segments have the opposite sense (LSR and RSL) and, possibly, the BBB solutions. In the following, numerical root-finding methods are presented to find these other extremals.*

3.3.2 Numerical Solution for LSR and RSL Trajectories

In the previous section a method was presented to find a subset of all possible candidate extremals (two out of six) in closed form. Returning to equation (3.32), define the function

$$f(t_A) = E\cos(\delta_1\omega t_A + \phi_{t_1}) + F\sin(\delta_1\omega t_A + \phi_{t_1}) - G. \tag{3.40}$$

The objective is to find the values of t_A for which $f(t_A) = 0$. The function $\frac{f(t_A)}{V_a V_w}$ is plotted in Figure 3.8 for the following boundary conditions and parameters:

$$\begin{aligned} x_{N_0} &= 0\text{m}, & y_{E_0} &= -200\text{m}, & \psi_0 &= 0, \\ x_{N_f} &= 0\text{m}, & y_{E_f} &= 200\text{m}, & \psi_f &= \pi \\ V_a &= 20\text{m/s}, & V_w &= 5\text{m/s}, & \psi_w &= 0. \end{aligned}$$

To find the roots of the equation $f(t_A) = 0$, one may use any root-finding technique, such as the bisection algorithm, or the Newton-Raphson method; here we consider the latter. Define the mapping

$$g(t_A) = t_A - \frac{f(t_A)}{f'(t_A)},$$

where

$$f'(t_A) = \frac{df(t_A)}{dt_A} = -E\delta_1\omega\sin(\delta_1\omega t_A+\phi_{t_1})+F\delta_1\omega\cos(\delta_1\omega t_A+\phi_{t_1})+V_aV_w\left(\frac{\delta_1}{\delta_2}-1\right)\sin(\delta_1\omega t_A+\phi_{t_1}). \tag{3.41}$$

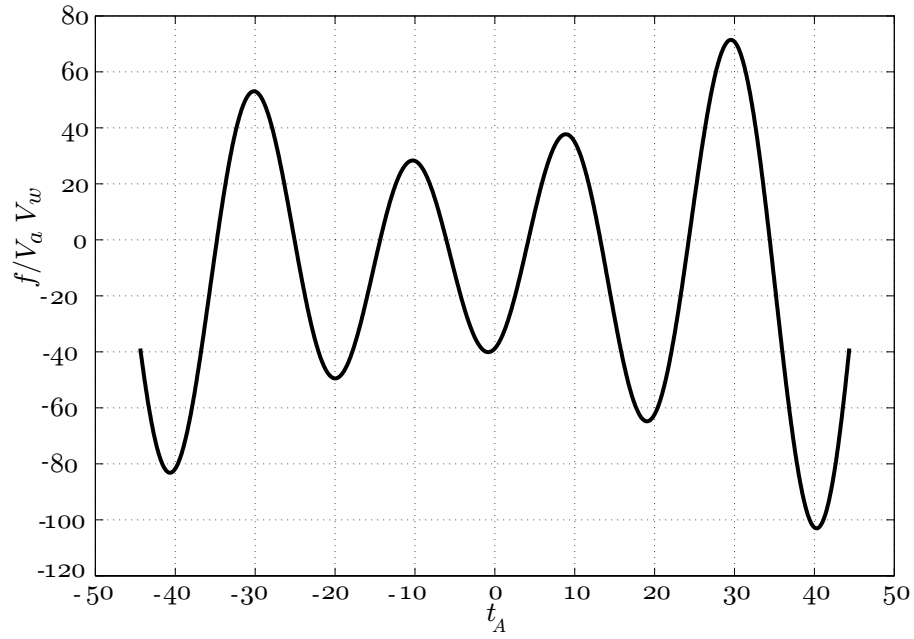


Figure 3.8 The function $\frac{f(t_A)}{V_a V_w}$ for $t_A \in [-2t_{2\pi}, 2t_{2\pi}]$. There are several roots for $t_A \in [0, t_{2\pi}]$.

If the initial guess $\hat{t}_{A_1}$ is close enough to the true solution t_A, then the map defined by

$$\hat{t}_{A_{i+1}} = g(\hat{t}_{A_i}), \qquad i = 1, 2, ...$$

is a contraction map. Then, by the Banach fixed-point theorem, the series converges to t_A [47]. As illustrated in Figure 3.8 as an example, there are several possible roots that one needs to consider. Some of these solutions are infeasible: those for which the straight segment would join and leave the trochoidal paths in the opposite sense. Although the Newton-Raphson method converges to the root only if the initial condition is close enough to the true solution, simulations suggest that the algorithm converges within three to six iterations, on average. Both functions in equations (3.40) and (3.41) are simple, smooth functions and are easy to compute; it is a simple matter to define a sufficiently dense grid of initial conditions on the interval $t_A \in [0, 2t_{2\pi}]$. Once a value for t_A is obtained, the corresponding value of t_B may be found using equation (3.65). If the

obtained solution satisfies all the conditions (3.26)-(3.29), then the corresponding path is a candidate minimum-time trajectory. Results of Monte Carlo simulations are presented in Section 3.5.

Similarly as in the previous section, the number of possible solutions can be limited to a finite set to ensure that $t_B \in (-t_{2\pi}, t_{2\pi}]$. Since $\delta_1/\delta_2 = -1$, we have that $(\delta_1/\delta_2)t_A \in (-2t_{2\pi}, 0]$. Assuming that $\phi_{t_1} - \phi_{t_2}$ is taken modulo 2π, it follows from equation (3.65) that $k \in \{-1, 0, 1, 2, 3\}$. The distinct solutions may be entered into a table, along with the LSL and RSR paths obtained using the closed-form expression discussed in 3.3.1. In cases where BBB solutions don't exist, the true minimum-time solution is selected from these candidates. Although the approach described above employs a numerical root-finding algorithm, the simple closed-form expressions and the small number of required iterations make the run-time predictable and quick; see Section 3.5. The algorithm is quite suitable for implementation as a real-time UAV path planning method.

This simple numerical routine yields those candidate extremals for which the trochoids have opposite sense, i.e. $\text{sign}(\delta_1) \neq \text{sign}(\delta_2)$. In Figure 3.10 candidate minimum-time BSB solutions are shown for randomly selected initial and final conditions.

3.3.3 Existence of "Unconventional" Extremals

Consider the case in which there is no wind or, equivalently, the problem of finding shortest paths of bounded curvature in the plane [20, 11, 15]. In the absence of wind, the trochoidal segments become circular. In this case, a UAV turning at maximum rate would arrive back to its original position in exactly time $t_{2\pi}$, so it is unnecessary to consider solutions for which $t_A, t_B \notin [0, t_{2\pi}]$. Moreover, the following lemma is proved in [15].

Lemma 3.3.7. *[15] Denote the angle of intermediate arcs on the circular segments by u and v. A path R_uSR_v, or L_uSL_v, can not be optimal if $u + v > 2\pi$.*

The equivalent of this lemma in the case where winds are present would state that *for an optimal path* $t_A + (t_{2\pi} - t_B) \leq t_{2\pi}$. In other words, the total time spent on both trochoidal segments must be no greater than $t_{2\pi}$ for optimality.

Every extremal in the inertial frame has a corresponding air frame equivalent, which is the UAV path an observer moving with the wind would see. In the air relative frame, the extremals are composed of circular segments and straight lines. For the purpose of this section, we call extremals "conventional" if there exists a corresponding minimum-time "Dubins path" in the air relative frame $\mathcal{F}_A$. By "Dubins path" we mean any minimum arc-length path as described in [20]. It turns out, however, that if $V_w \neq 0$, then extremals in the inertial frame may not be Dubins paths in the air relative frame. This is illustrated by the following example.

Consider the initial and final conditions and parameters

$$\begin{aligned} x_{N_0} &= 101.3\text{m}, & y_{E_0} &= 297.1\text{m}, & \psi_0 &= 2.48, \\ x_{N_f} &= -581.47\text{m}, & y_{E_f} &= 411.29\text{m}, & \psi_f &= 6.12 \\ V_a &= 20\text{m/s}, & V_w &= 10\text{m/s}, & \psi_w &= 4.04 \qquad \omega = 0.2832. \end{aligned}$$

In Figure 3.9 we can see the resulting LSL and RSR trajectories. For the LSL path (solid line), there is no solution that would satisfy both $t_A \in [0, t_{2\pi}]$ and $t_B \in [0, t_{2\pi}]$. This is due to the fact that the instantaneous center of the maximum turn rate circle in the air relative frame moves distance $d = V_w t_{2\pi}$ in the direction of the wind in time $t_{2\pi}$. Thus we have established the following:

Proposition 3.3.8. *There exist initial and final conditions* (x_0, y_0, ψ_0), (x_f, y_f, ψ_f), *and parameter values* V_a, V_w, ψ_w, ω, *for which some of the candidate extremals fail to satisfy that both* $t_A \in [0, t_{2\pi}]$ *and* $t_B \in [0, t_{2\pi}]$.□

An immediate consequence of Proposition 3.3.8 is that if one were to look for minimum-time paths in the inertial frame by looking for their corresponding minimum-time equiv-

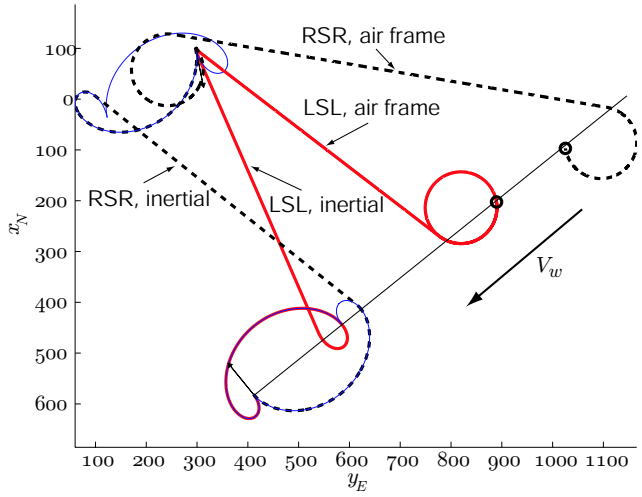


Figure 3.9 Illustration showing that, for some initial conditions, some of the extremals fail to satisfy $t_B \in [0, t_{2\pi}]$. In this example $t_B < 0$ for the LSL trajectory. The figure shows both the inertial path and the air relative path. The circles indicate the position of a "virtual target" [41], traveling upwind with the same speed as the wind speed, that the aircraft would intercept by flying a minimum-time path in the air relative frame.

alent in the air relative frame, some of the possible extremals would not be found. Referring to Figure 3.9, the RSR path would be easily found, but the LSL path would be missed, as its air frame equivalent is not a Dubins trajectory.

3.4 Computing BBB Trajectories

Similarly to the previous section, we define three trochoids as

$$x_{t_1}(t) = \frac{V_a}{\delta_1 \omega} \sin(\delta_1 \omega t + \phi_{t_1}) + V_w t + x_{t_{10}}, \quad t \in [0, t_A] \tag{3.42}$$

$$y_{t_1}(t) = \frac{-V_a}{\delta_1 \omega} \cos(\delta_1 \omega t + \phi_{t_1}) + y_{t_{10}} \tag{3.43}$$

$$x_{t_2}(t) = \frac{V_a}{\delta_2 \omega} \sin(\delta_2 \omega t + \phi_{t_2}) + V_w t + x_{t_{20}}, \quad t \in [t_A, t_B] \tag{3.44}$$

$$y_{t_2}(t) = \frac{-V_a}{\delta_2 \omega} \cos(\delta_2 \omega t + \phi_{t_2}) + y_{t_{20}} \tag{3.45}$$

$$x_{t_3}(t) = \frac{V_a}{\delta_3\omega}\sin(\delta_3\omega t + \phi_{t_3}) + V_w t + x_{t_{30}}, \quad t \in [t_B, T] \tag{3.46}$$

$$y_{t_3}(t) = \frac{-V_a}{\delta_3\omega}\cos(\delta_3\omega t + \phi_{t_3}) + y_{t_{30}}, \tag{3.47}$$

where $\delta_i \in \{-1, 1\}$, with $\delta_1 = \delta_3 = -\delta_2$.

Remark 3.4.1. *Notice that here the path parameter value is common for all three segments (c.f. Remark 3.3.1).*

The objective is to find the path parameter values t_A, t_B, T, and the integration constants $x_{t_{i0}}$, $y_{t_{i0}}$, ϕ_{t_i}, such that $[x_{t_1}, y_{t_1}, \psi_{t_1}]^T\big|_{t=0} = [x_0, y_0, \psi_0]^T$, and $[x_{t_3}, y_{t_3}, \psi_{t_3}]^T\big|_{t=T} = [x_f, y_f, \psi_f]^T$, where $\psi_{t_i} = \delta_i\omega t + \phi_{t_i}$.

The problem can be simplified to obtain two transcendental equations for two unknowns t_A and T. The system of equations can be solved using a numerical root-finding technique to obtain t_A and T. Figure 3.11 shows an example where the minimum-time solution is a LRL trajectory. The details of the calculation are presented in Appendix A.

3.5 Results

The methods described in the preceding sections were tested in Monte Carlo simulations. The aircraft air speed was chosen $V_a = 20$ m/s, the wind speed was $V_w = 5$ m/s, and the maximum turn rate was $\omega = 0.2832$ rad/s, which corresponds to a maximum bank angle $\phi_{\text{MAX}} = 30^\circ$ for an aircraft in a coordinated turn. The boundary conditions were randomly selected from uniform distributions over the ranges shown in Table 3.2.

The path planning algorithm was executed for $N = 10,000$ randomly selected initial and final points. The average run-time on a 2.66 GHz CPU was $T_{\text{AVG}_{\text{RUN}}} = 308$ ms with standard deviation $\sigma_{\text{RUN}} = 18$ ms. The distribution of path types is shown in Table 3.3. In the case when only the BSB trajectories are calculated, the run-time is reduced to $T_{\text{AVG}_{\text{RUN}}} = 3.2$ ms. The grid size for solving (3.40) was $N_i = 10$ for these simulations

x_{N_0}	y_{E_0}	ψ_0	ψ_f	ψ_w	R	θ
$[-500, 500]$ m	$[-500, 500]$ m	$[0, 2\pi]$	$[0, 2\pi]$	$[0, 2\pi]$	$[-500, 500]$ m	$[0, 2\pi]$

Table 3.2 The initial conditions were selected randomly in the ranges shown in the table. The final point was selected R distance away from the initial point in the direction defined by θ.

LSL	RSR	LSR	RSL	RLR	LRL
26%	25%	19.6%	19.9%	4.9%	4.6%

Table 3.3 Distribution of minimum-time path types based on Monte Carlo simulations.

(3.3.2). For the 2D Newton-Raphson method this results to $N_{i_{2D}} = 10 \times 10 = 100$ initial points for the root-finding algorithm, explaining the increased run-time. It is well known that the BBB Dubins paths in the absence of winds only exist if the distance between initial and final points satisfies $d \leq 4R_0$, where R_0 is the maximum rate turning radius. Thus, if it is known that the initial and final points are sufficiently far from each other, one only needs to consider the BSB paths.

As referenced earlier, Figure 3.10 shows the results for the BSB solutions only, for randomly selected initial and final conditions. In the figures the initial conditions and parameters were selected the same way as in the Monte Carlo simulations. The dashed lines show the candidate extremals for the LSL, RSL, LSR and RSR trajectories, and the solid line shows the minimum-time path selected among these candidate extremals.

Figure 3.11 shows the results for all possible three-segment extremals. In the example the initial conditions were selected as $x_0 = 0$ m, $y_0 = -200$ m, $\psi_0 = 0$, $x_f = 0$ m, $y_f = -180$ m $\psi_f = \pi$, and the parameters used for the path planning algorithm were $\omega = 0.2832$ rad/s, $V_w = 5$ m/s, $V_a = 20$ m/s. Each figure 3.11(a)- 3.11(f) shows all candidate

extremals in dashed line. The difference between the figures is that the each has only a set of extremals highlighted in solid line, corresponding to the RSR, RSL, LSR, LSL, LRL and RLR trajectories, respectively. In this case the minimum-time trajectory is the one highlighted with solid line in Figure 3.12, which is a slight initial turn to the left, followed by a turn to the right, then followed by another turn to the left.

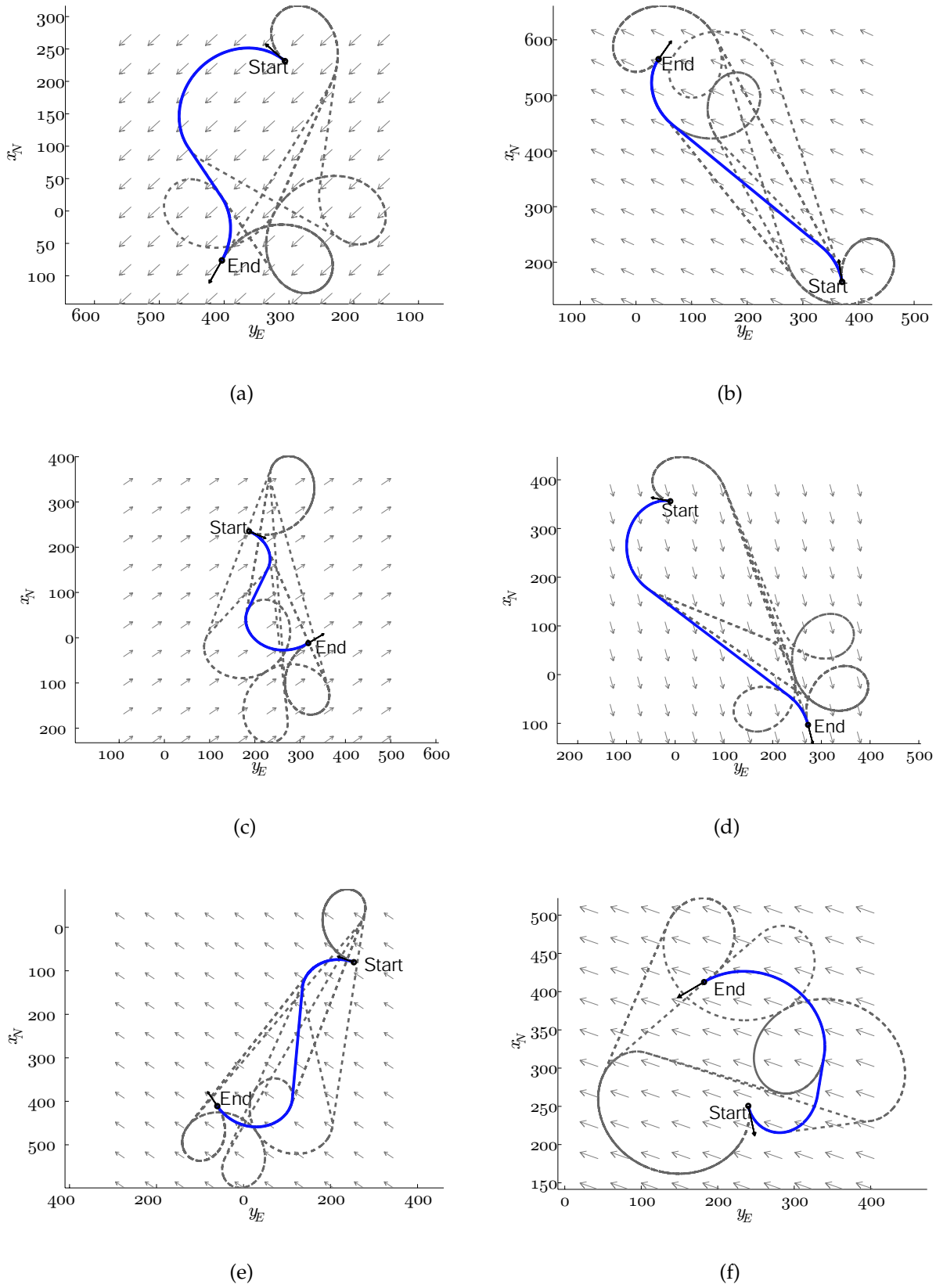
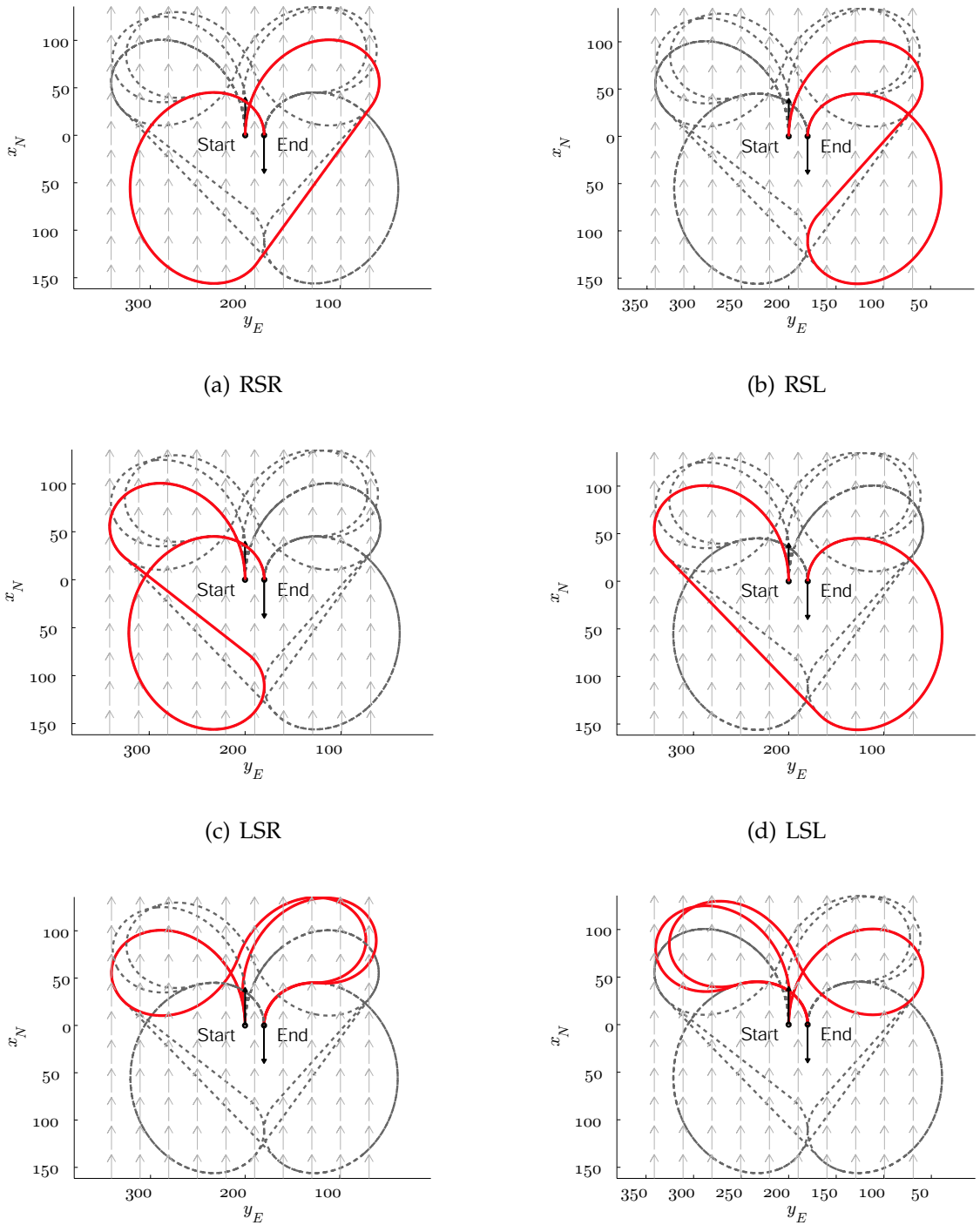


Figure 3.10 Candidate BSB extremal paths for randomly selected initial and final states.



(a) RSR

(b) RSL

(c) LSR

(d) LSL

(e) LRL

(f) RLR

Figure 3.11 Candidate time-optimal paths found by the path planning algorithm for initial conditions $x_0 = 0$ m, $y_0 = -200$ m, $\psi_0 = 0$, $x_f = 0$ m, $y_f = -180$ m $\psi_f = \pi$, $\omega = 0.2832$ rad/s, $V_w = 5$ m/s, $V_a = 20$ m/s. The time-optimal path is LRL trajectory with $t_A = 0.6869$ s, $t_B = 17.4451$ s and $T = 22.4228$ s.

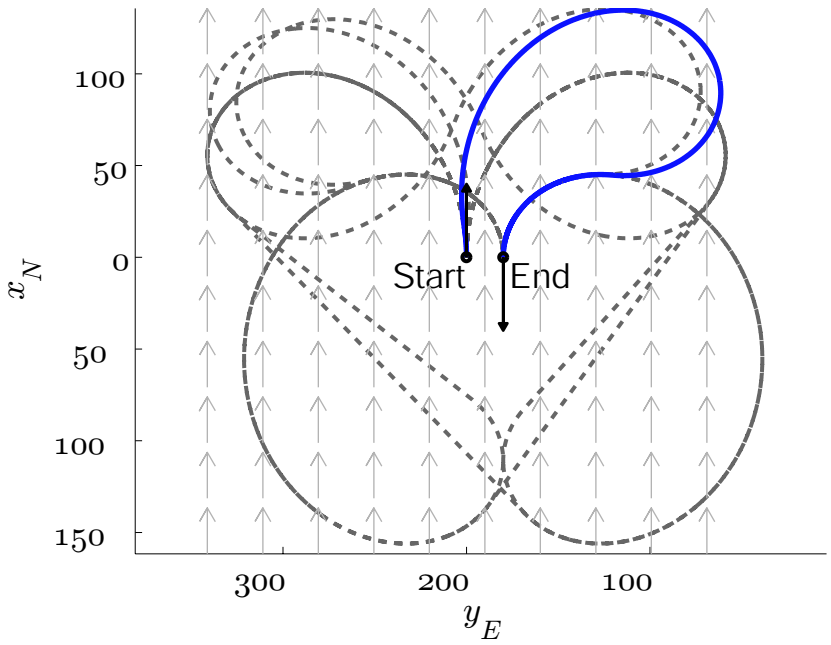


Figure 3.12 Candidate time-optimal paths found by the path planning algorithm for initial conditions $x_0 = 0$ m, $y_0 = -200$ m, $\psi_0 = 0$, $x_f = 0$ m, $y_f = -180$ m $\psi_f = \pi$, $\omega = 0.2832$ rad/s, $V_w = 5$ m/s, $V_a = 20$ m/s. The time-optimal path is LRL trajectory with $t_A = 0.6869$ s, $t_B = 17.4451$ s and $T = 22.4228$ s.

3.6 Path Planning with Turn Rate and Acceleration Bounds

In the previous sections a method was described to find minimum-time paths in the plane with bounds on the maximum turn rate and in the presence of uniform winds. The path-planning method yields trajectories that are composed of either straight lines or trochoidal segments. The transition points where these trajectories are smoothly joined together, however, correspond to points where the path curvature is discontinuous. At these points the vehicles need to change the rate of turn instantaneously, which is only possible for wheeled mobile robots that can stop, re-orient their wheels, and continue the turn in the new orientation. If the transition from zero turn rate to maximum turn rate takes place quickly compared to the duration of the complete turn, then the transition can be neglected. In that case the assumption that the desired turn rate can be instantly achieved is reasonable. However, if that is not the case, one needs to take into account the additional turn rate dynamics explicitly from the modeling phase. We are looking for feasible trajectories that have continuous curvature profile and bring the vehicle from its initial state to its final state in minimum time.

To allow ground robots to continue turning without stopping, continuous-curvature path planning was studied in [12]. It was shown that the extremal paths contain *clothoid* curves, i.e. curves that have linearly changing curvature profile, and straight segments. Later it was shown that these extremal paths may be composed, in general, of an infinite number of such pieces [34]. In [59], a simple path planning method is presented that exploits the geometrical symmetry of the optimal paths. The solutions can be found analytically, and they may be composed of clothoids, circular arcs, and straight segments, generalizing Dubins trajectories to the case where turn acceleration bounds are present. Because of the anticipated complexity of finding all extremal solutions, the search for optimal trajectories is confined to a finite set. The proposed paths are referred to as *Simple*

Continuous-Curvature (SCC) paths. The algorithm essentially finds the points where the clothoids, circular arcs, and straight paths may be smoothly joined together, forming a feasible, continuous-curvature path. The difficulty of working with clothoid curves, for which no explicit analytical expression exists, is relaxed by the observation that the resulting paths are symmetric with respect to a characteristic line. Finding the points where clothoid curves, circles, and straight lines may be smoothly stitched together has been extensively studied in the past in the context of highway engineering [44, 45] and computer graphics [46]. Clothoid curves have the property that the curvature changes linearly along the path, and so does the lateral acceleration experienced by a car or train traveling along such a curve. In highway engineering, such curves are often referred to as *transition spirals* [26].

3.6.1 Vehicle Model

Similarly as in (3.1), we use a simple particle model for the UAV, described by the equations

$$\dot{x}(t) = V_a \cos\psi(t) + V_w \tag{3.48}$$

$$\dot{y}(t) = V_a \sin\psi(t) \tag{3.49}$$

$$\dot{\psi}(t) = \alpha\phi(t) \tag{3.50}$$

$$\dot{\phi}(t) = u(t) \tag{3.51}$$

Here $[x(t), y(t)]^T$ denote the inertial coordinates of the aircraft, $\psi(t)$ is the heading angle measured from the x-axis, V_a is the constant airspeed, and V_w is the constant wind speed, which can be assumed to be aligned with the inertial x-axis without loss of generality.

The model described by equations (3.48)-(3.51) was selected to capture the fact that the turn rate of the vehicle cannot change instantaneously. In steady turning flight, the

heading rate of change can be expressed as a function of bank angle $\phi(t)$, as

$$\dot{\psi}(t) = \frac{g}{V_a} \tan \phi(t)$$

In typical aircraft operations, where the bank angle is small ($\phi(t) < 30°$), we can approximate the above relationship with the simple expression $\dot{\psi}(t) \approx (g/V_a)\,\phi(t) = \alpha\phi(t)$. Hence in equations (3.48)-(3.51) one can identify $\phi(t)$ with the aircraft bank angle. The rate of the bank angle is the control input. The aircraft lateral dynamics are primarily affected by the ailerons, which are differentially operated control surfaces on the wings. A small deflection of the ailerons results in a torque about the aircraft longitudinal axis, and ultimately in an angular acceleration about the same axis. In addition to the limitations imposed by the aircraft dynamics and actuator limits, the rate of the bank angle is also constrained by structural limits on the maximum allowable loads on the wings. The control signal is thus constrained to a compact set $u \in [-\bar{u}, \bar{u}]$.

Considering bang-bang and singular solutions only, we constrain the control signal to the set $u \in \{-\bar{u}, 0, \bar{u}\}$. There are two cases to consider. The total time it takes to complete a turn will be denoted as $\bar{t}$ in both cases.

Case 1: The maximum bank angle is reached. In this case the turn rate is saturated, and the aircraft will continue its maximum constant rate turn for a certain time. After that, the bank angle is decreased until the aircraft reaches straight and level flight again. The turn is initiated by a maximum bank angle rate turn

$$\phi(t) = \bar{u}t$$

The maximum bank angle is reached at time $t_1 = \bar{\phi}/\bar{u} > 0$. The bank angle for the entire turn can be written as

$$\phi(t) = \begin{cases} \bar{u}t, & t \in [0, t_1] \\ \bar{u}t_1, & t \in [t_1, \bar{t} - t_1] \\ -\bar{u}(t - \bar{t}), & t \in [\bar{t} - t_1, \bar{t}] \end{cases} \tag{3.52}$$

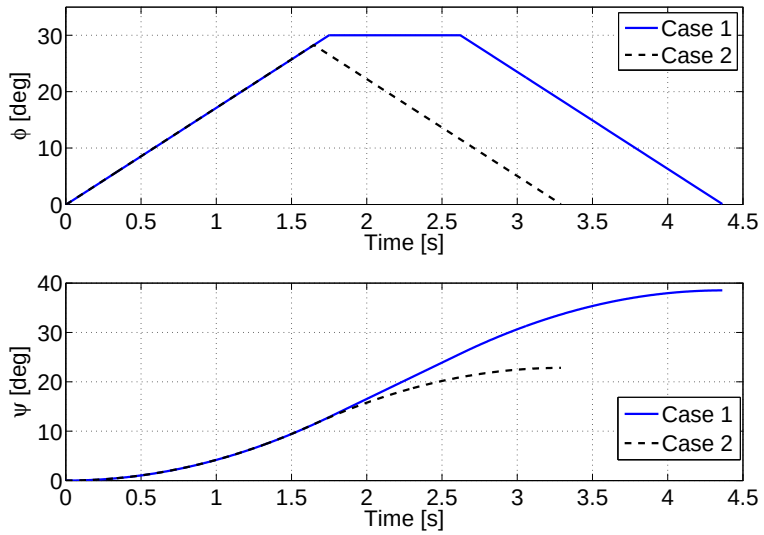


Figure 3.13 Bank angle and heading angle for two different cases of turns. In *Case 1* the maximum bank angle is reached, while in *Case 2* the maximum bank angle $\bar{\phi}$ is not reached.

The heading angle is then

$$\psi(t) = \begin{cases} \alpha\bar{u}\frac{t^2}{2} + \psi_0, & t \in [0, t_1] \\ \alpha\bar{u}t_1 t + \psi_0 + C_1, & t \in [t_1, \bar{t} - t_1] \\ \alpha\bar{u}(t\bar{t} - \frac{t^2}{2}) + \psi_0 + C_2, & t \in [\bar{t} - t_1, \bar{t}] \end{cases} \tag{3.53}$$

The constants can be easily found from continuity conditions at $t = t_1$ and $t = \bar{t} - t_1$:

$$\begin{aligned} C_1 &= -\alpha\bar{u}\frac{t_1^2}{2} \\ C_2 &= -\alpha\frac{\bar{u}}{2}\left(2t_1^2 + \bar{t}^2 - 2\bar{t}t_1\right). \end{aligned}$$

At the end of the turn, $t = \bar{t}$, the heading angle is given by the following expression

$$\psi(\bar{t}) = -\alpha\bar{u}t_1^2 + \psi_0 + \alpha\bar{u}t_1\bar{t}. \tag{3.54}$$

Case 2: The maximum bank angle is not reached. That is, the aircraft will initiate a maximum bank rate turn, and before the maximum bank angle is reached, it will start to come

back to straight and level flight again. The bank angle for the entire turn can be written as

$$\phi(t) = \begin{cases} \bar{u}t, & t \in [0, \bar{t}/2] \\ -\bar{u}(t - \bar{t}), & t \in [\bar{t}/2, \bar{t}] \end{cases} \tag{3.55}$$

The heading angle is then

$$\psi(t) = \begin{cases} \alpha\bar{u}\frac{t^2}{2} + \psi_0, & t \in [0, \bar{t}/2] \\ \alpha\bar{u}(t\bar{t} - \frac{t^2}{2}) + \psi_0 + C, & t \in [\bar{t}/2, \bar{t}] \end{cases} \tag{3.56}$$

The constant C can be easily found from the continuity condition at $t = \bar{t}/2$:

$$C = -\alpha\bar{u}\frac{\bar{t}^2}{4}$$

The different cases are illustrated in Figure 3.13, where the time histories for bank angle and heading angle are shown for a *Case 1* and a *Case 2* turn.

3.6.2 Path Planning

Similarly as before, we seek the solutions in terms of switching points, where the initial and final turns may be smoothly connected with a straight path. The point where the vehicle leaves the initial turn on the straight segment will be denoted as P_A, and the point where it starts the final turn will be denoted P_B. The path parameters at these points will be denoted as t_A and t_B. Consider two paths in the plane, corresponding to the initial and final turns, and defined by the equations

$$x_1(t) = \int_0^t V_a \cos\psi_1(\tau) + V_w d\tau \tag{3.57}$$

$$y_1(t) = \int_0^t V_a \sin\psi_1(\tau) d\tau \tag{3.58}$$

$$x_2(t) = \int_0^t V_a \cos\psi_2(\tau) + V_w d\tau \tag{3.59}$$

$$y_2(t) = \int_0^t V_a \sin\psi_2(\tau) d\tau \tag{3.60}$$

Remark 3.6.1. *As before, we use the same symbol, t, as the path parameter for both curves (c.f. Remark 3.3.1).*

The paths have to satisfy the following conditions:

$$[x_1(0), y_1(0)]^T = [x_0, y_0]^T$$

$$[x_2(0), y_2(0)]^T = [x_f, y_f]^T$$

Note that in this setting the second turn starts at path parameter $t = 0$, and the parameter value $t_B < 0$ corresponds to a configuration reached by integration in reverse time. With no bounds on the turn-acceleration, equations (3.57)-(3.60) could be integrated to obtain analytical expressions for the paths corresponding to trochoid curves [70] [71]. The chief complication in the path planning problem presented in this work is that there are no closed-form solutions to the above integrals with the turn rate signals given in Section 3.6.1. In the air-relative frame the transition curves between straight segments and maximum turn rate segments are clothoid curves (also known as *Cornu spirals* or *Euler spirals*). When $V_w = 0$, equations (3.57)-(3.60) are special forms of the Fresnal-integrals [2], for which no analytical expressions exist. The points $[x_1(t_A), y_1(t_A)]^T$ and $[x_2(t_B), y_2(t_B)]^T$ have to be found by numerical integration.

Since the path connecting points P_A and P_B is a straight line, the heading angles have to satisfy the continuity equation:

$$\psi_1(t_A) = \psi_2(t_B) + 2k\pi, \quad k \in \mathbb{Z}, \tag{3.61}$$

where $\mathbb{Z}$ is the set of real integers.

In what follows, we consider four different types of trajectories. *Type 1* trajectories are those for which both the initial and final turns are *Case 1* turns, i.e. the maximum bank angle is reached. *Type 2* trajectories are those for which both the initial and final turns are *Case 2* turns, i.e. the maximum bank angle is *not* reached. *Type 3* (*Type 4*) trajectories are

those for which the first (second) turn is *Case 1*, and the second (first) turn is *Case 2*. We will present the analysis results only for *Type 1* and *Type 2* paths; the other two types can be computed in a similar fashion.

In both cases the objective is to find the straight path that connects points P_A and P_B. At these two points the following additional continuity equations must be satisfied:

- The velocities at point P_A and point P_B must be equal:

$$(\dot{x}_1(t_A), \quad \dot{y}_1(t_A))^T = (\dot{x}_2(t_B), \quad \dot{y}_2(t_B))^T. \tag{3.62}$$

- The line segment joining the points P_A and P_B must be tangent with the velocity vectors at both points:

$$\tan(\alpha) = \frac{y_2(t_B) - y_1(t_A)}{x_2(t_B) - x_1(t_A)} \tag{3.63}$$

$$\tan(\alpha) = \frac{\dot{y}_2(t_B)}{\dot{x}_2(t_B)} = \frac{\dot{y}_1(t_A)}{\dot{x}_1(t_A)}. \tag{3.64}$$

Type 1 Extremals

In this case both the initial and final turns are *Case 1* turns. Substituting $\bar{t} = t_A$ into equation (3.54), we get the expression

$$\psi_1(t_A) = -\alpha\delta_1\bar{u}t_1^2 + \psi_{1_0} + \alpha\delta_1\bar{u}t_1t_A,$$

where we have introduced the variable $\delta_1 \in \{-1, 1\}$ to denote the direction of the turn (left or right, respectively). Similarly,

$$\psi_2(t_B) = \alpha\delta_2\bar{u}t_1^2 + \psi_{2_0} + \alpha\delta_2\bar{u}t_1t_B.$$

Notice that the sign of the terms has changed because of integration in reverse time, and because of the assumption $t_1 > 0$ and $t_B < 0$. From the continuity equation (3.61) we get the expression

$$t_B = t_A\frac{\delta_1}{\delta_2} + \frac{\psi_{1_0} - \psi_{2_0}}{\alpha\delta_2\bar{u}t_1} - t_1\frac{\delta_1 + \delta_2}{\delta_2} + \frac{2k\pi}{\alpha\delta_2\bar{u}t_1}. \tag{3.65}$$

Using equations (3.62)-(3.64) one may obtain the following equation

$$\dot{y}_1(t_A)(x_2(t_B) - x_1(t_A)) = \dot{x}_1(t_A)(y_2(t_B) - y_1(t_A)).$$

Substituting equation (3.65) into the above expression, one obtains a single equation for t_A. This equation has to be solved numerically. In this work we used the bisection algorithm to obtain the root t_A. Once t_A is known, t_B may be found using equation (3.65).

Type 2 Extremals

Substituting $t = t_A$ into equation (3.56), we get the expression

$$\psi_1(t_A) = \alpha\delta_1\bar{u}\frac{t_A^2}{4} + \psi_{1_0}.$$

Similarly,

$$\psi_2(t_B) = \alpha\delta_2\bar{u}\frac{t_B^2}{4} + \psi_{2_0}.$$

From the continuity equation (3.61) we get the expression

$$t_B = -\sqrt{\left|\frac{4(\psi_{1_0} - \psi_{2_0} + 2k\pi)}{\alpha\delta_2\bar{u}} + \frac{\delta_1}{\delta_2}t_A^2\right|}. \tag{3.66}$$

3.6.3 Results

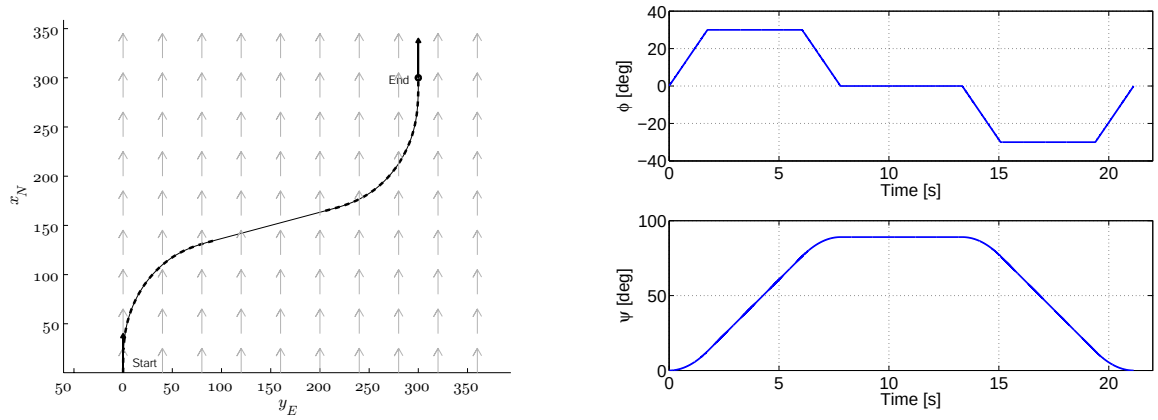
The path planning method has been implemented for Type 1 and Type 2 trajectories. Table 3.4 summarizes the parameter values used in the simulations.

Figure 3.14(a) shows the results of the path planning algorithm for a Type 1 trajectory. The candidate time-optimal path is composed of a right turn followed by a straight path, then followed by a left turn. Both the initial and final turns are composed of three distinct segments: (1) a maximum acceleration turn until the maximum bank angle is reached; (2) followed by a maximum bank angle segment; (3) followed by a maximum acceleration turn bringing the aircraft back to straight and level flight. Figure 3.14(b) shows the time histories of the bank angle and the heading angle during the complete maneuver.

Parameter	Value
V_a	$20\ m/s$
V_w	$5\ m/s$
$\bar{u}$	$0.3\ \mathrm{rad}/s$
α	$0.4905\ 1/s$
$\bar{\phi}$	$30°$

Table 3.4 Parameters used in simulations.

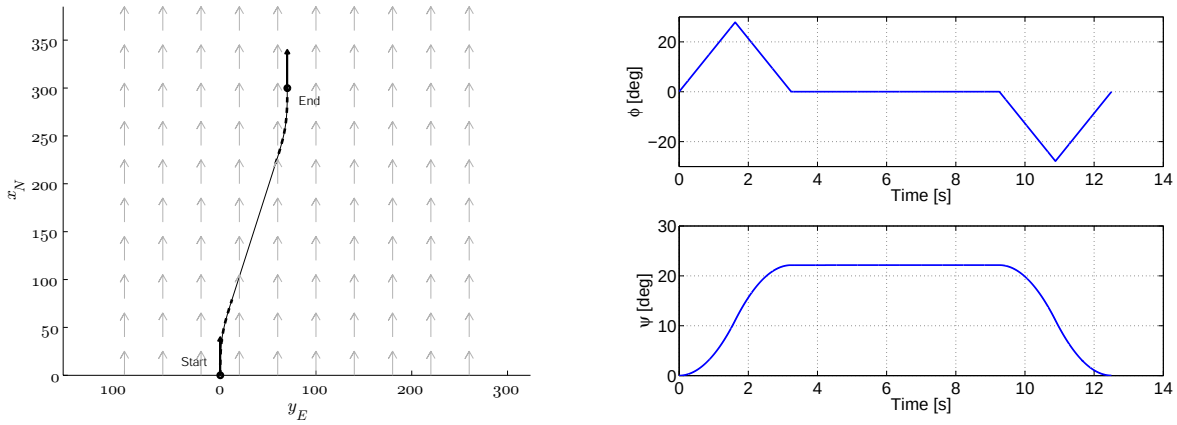
Similarly, Figure 3.15(a) shows an example where the candidate time-optimal path is a Type 2 trajectory. The turns are composed of segments where the maximum bank angle is not reached (see Figure 3.15(b)). Figure 3.16 shows all four candidate extremals. In this particular example all four extremals are Type 1 trajectories.



(a) Type 1 path.

(b) Time histories of $\phi(t)$ and $\psi(t)$.

Figure 3.14 A Type 1 maneuver. In this case both the initial and final turns are saturated, i.e. the maximum bank angle is reached.



(a) Type 2 path.

(b) Time histories of $\phi(t)$ and $\psi(t)$.

Figure 3.15 A Type 2 maneuver. Both the initial and final turns are completed without reaching the maximum bank angle.

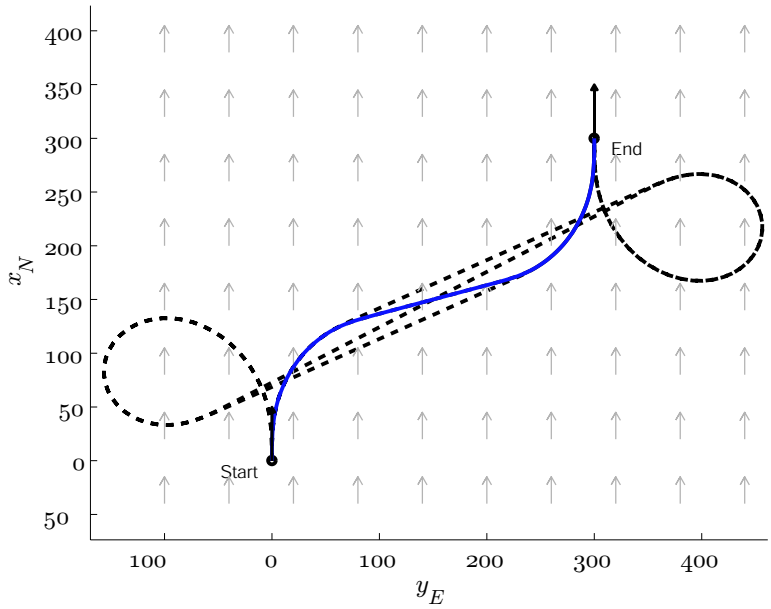


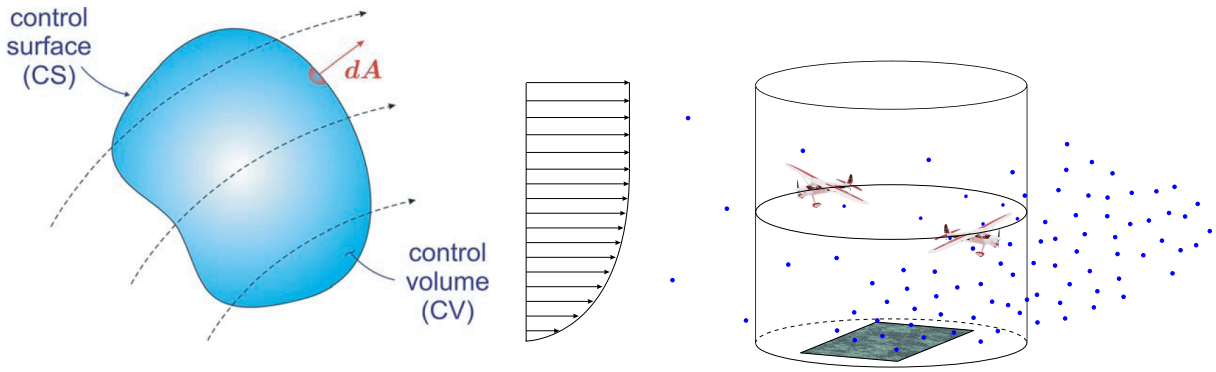
Figure 3.16 Figure showing all four extremal paths. The candidate minimum-time trajectory is a Type 1 right turn followed by a straight line followed by a Type 1 left turn.

Chapter 4

Control-Volume Sampling

In this section a specific environmental sampling framework is presented that can be used to estimate the rate of release of some atmospheric constituent (particles, sporangia, volatile organic compounds, etc.) inside a certain region. In aerobiological sampling experiments, for example, it is very important to know the source strength, Q (number of sporangia/m^2/s), of the pathogen. (Also see Section 1.1.) The measurements collected using sporangia sampler devices may be biased by the presence of the "background disease:" sporangia of the same pathogen that were generated at remote locations independently from the field experiments. Such sporangia can be transferred to the location of the experimental fields by atmospheric transport. To obtain an accurate estimate of the source strength, Q, a *control-volume sampling* technique is presented that can be performed by UAVs equipped with onboard sporangia sampling devices. The discussion focuses on aerobiological sampling, but the methods may also be used to assess the strength of other aerial or underwater constituent sources.

Consider the problem of quantifying the rate of release, Q, within a closed region; that is the control-volume. The control-volume equation for the rate of change of an extensive



(a) Illustration of a control-volume in a flow. (b) Particle flow in an aerobiological sampling problem.

Figure 4.1 In the control-volume sampling problem one is interested in assessing the net rate of outflow across the boundaries of the volume. By continuity, this equivalently yields the rate of release inside the volume.

(scalar or vector) quantity $\boldsymbol{Q}$ in a *fixed* volume of interest is [78]

$$\frac{d\boldsymbol{Q}}{dt} = \iint_{\text{CS}} \frac{d\boldsymbol{Q}}{dm}\, \rho\, \boldsymbol{V} \cdot \boldsymbol{dA} + \frac{\partial}{\partial t} \iiint_{\text{CV}} \frac{d\boldsymbol{Q}}{dm}\, \rho\, dV$$

where $\frac{d\boldsymbol{Q}}{dm}$ represents the intensive value of the quantity (i.e., the quantity per unit mass) at a point in the control-volume and V represents volume. On the left, the total time-rate-of-change is determined by physical principles. The first term on the right accounts for the flow of the quantity across the control surface (the control-volume boundary) and the second term accounts for the rate of change of the property within the control-volume (see Figure 4.1(a)). If $\boldsymbol{Q}$ represents mass, for example, the left-hand side is zero and the equation implies that the rate of increase of mass within the control-volume equals the net rate of *inflow*.

Suppose $\boldsymbol{Q}$ represents the number of particles (e.g., plant pathogen sporangia) in the control-volume. Suppose also that, on the spatial scale of interest, the fluid density remains constant so that $dm = \rho dV$. We may then write a particle continuity equation

$$\iint_{\text{CS}} \frac{d\boldsymbol{Q}}{dV}\, \boldsymbol{V} \cdot \boldsymbol{dA} + \frac{\partial}{\partial t} \iiint_{\text{CV}} \frac{d\boldsymbol{Q}}{dV}\, dV = 0.$$

The first term on the left represents the net flux of particles out of the control-volume and the second represents the rate at which particles are released within the volume (see Figure 4.1(b)).

If one could measure the net outflow across the boundaries of the control-volume, that would, in turn, yield the net rate of release within the control-volume. In the framework described in this section, the outflow across the boundaries of the control-volume is measured using two UAVs equipped with sporangia-collection devices. The samplers may be opened in flight so that they are exposed to the incoming airflow to allow the pathogen sporangia to be caught. Based on the sample time and the measured airspeed, one may assess the average sporangial concentration. Imagine the closed flight path to be separated into two equal length portions (e.g. two halves of a circle) with an imaginary dividing line: one that lies downwind from the source and one that lies upwind from the source. (See Figure 4.2.) The difference in the average concentration measurements along these two arcs characterizes the net outflow across the boundaries, or equivalently the net rate of sporangia release. If the UAVs can be equipped with multiple sampling devices that can be opened and closed in flight, then the control-volume sampling may be performed by one vehicle that opens one set of samplers on the downwind sampling arc and another set of samplers on the upwind arc. However, if only one sampling device can be fitted onboard the UAVs, then two vehicles are necessary. The vehicles in that case modulate their sampling activity such that one of the vehicles samples upwind of the source and the other vehicle samples downwind of the source.

When the vehicles are not sampling, they re-initialize to begin the next sampling leg. Since the vehicles are not sampling during this period, they should re-initialize as quickly as possible to save time and fuel, increasing the total volume of air that is sampled. The idea is illustrated in Figure 4.2, where the semi-circular sampling arcs and connecting re-initialization paths are shown for both vehicles. The path planning algorithm of Chapter 3

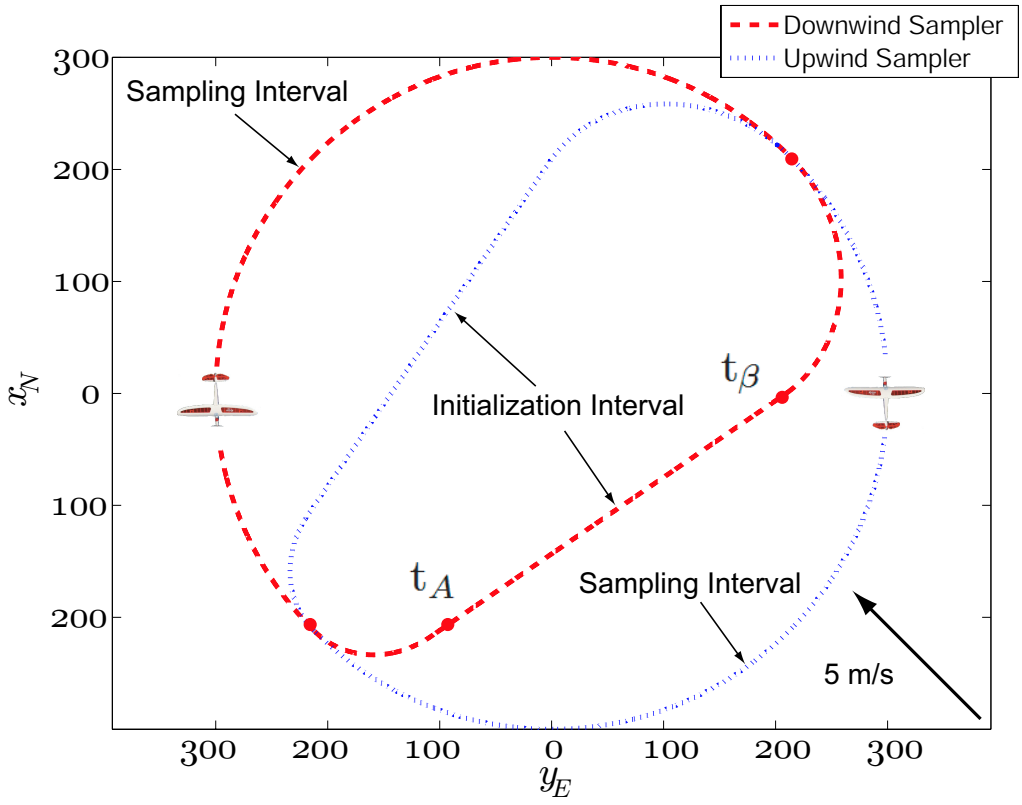


Figure 4.2 Two UAVs flying in the control-volume sampling mission. UAV1 (UAV2) samples only during the upwind (downwind) path segment. The end-points of the sampling arc are connected with a time-optimal path discussed in Chapter 3.

yields the minimum-time trajectories for optimal re-initialization (see also [71, 69]). We refer to the closed, convex curve that results as a D-curve due to its shape.

We note that the re-initialization interval assumes maximum control effort (maximum turn-rate), which constrains one's ability to enforce convergence to the path. In practice, the re-initialization path is generated with an artificial turn-rate limit that is strictly less than the true maximum turn-rate. This ensures that the UAV is able to track the desired path even in the presence of disturbances.

In order to ensure consistent sampling, the vehicles need to coordinate their motion such that they are properly phased along the sampling orbits. In Section 4.1 we formulate

the coordinated control problem and present a literature review on vehicle coordination. In Section 7.1, the particle motion model and convex curves are described. In Section 4.3, we consider coordination of particle formations along closed, convex curves in the presence of a steady, uniform wind. In the motion-coordination framework, the closed curves must be strictly convex. Since time-optimal paths may contain straight segments, an algorithm is devised in Section 4.4 that approximates a linear segment of an otherwise strictly convex path with a circular arc to arbitrary precision. The parametrization of the D-curves is presented in Section 4.5. In Section 4.6, we describe the solution and simulation results of the control-volume sampling problem. Although the methods were developed for UAVs that are to be used for aerobiological sampling applications, the framework is applicable to other vehicle-coordination tasks where the constant-speed particle model is applicable. Such an application example is the coordinated perimeter patrol problem, where a group of UAVs is tasked with patrolling a base perimeter: a topic discussed in Chapter 6.

The coordination method employed for this problem was based on previous research results on particle synchronization [60]. More recently, the particle synchronization method was extended to allow synchronization on general convex curves [51] and synchronization in winds [53]. The methods presented in this work are outcomes of a collaborative research and have been published in [69], [54] and [68]. The contribution to the field was the application of particle coordination theory to specific problems, such as UAV coordination on D-Curves, and the development of the approximation method discussed in Section 4.4.

4.1 Coordinated Control in Aerobiological Sampling

We consider the problem of motion coordination in the presence of steady, uniform winds. Planar control laws are presented that drive the vehicles to orbits around strictly convex curves such that the vehicles are equally spaced in time. The curves contain near time-optimal segments, corresponding to the minimum-time re-initialization paths. The methods directly extend from previous research results on particle coordination. The papers most relevant to this work include [60], in which Lyapunov-based control laws are provided to drive a collective of vehicles to a symmetric distribution along a circular orbit. The results are extended to allow coordination on convex curves [51], and coordination in the presence of winds [53] [50]. Although we focus on the case of all-to-all communication here, the entire framework extends directly to UAV networks in which communication is time-varying and/or directed [61].

We use a simple particle model to describe the motion of UAVs as unit speed (relative to the air) vehicles in the plane. The framework serves as an intermediate level between the lower-level vehicle control and the higher-level mission planning layer. We assume that the control signal is the turn rate-of-change, which, in case of a fixed-wing aircraft, can be controlled by regulating the vehicle bank angle. We assume that the lower level controller — the autopilot — can execute the desired turn-rate commands. Similar vehicle models have been frequently used to design kinematic control laws to track targets with aerial vehicles; see [56] [58] [24] for example.

Here we consider the temporal coordination of multiple identical flight vehicles (e.g., UAVs) along convex curves in a steady, uniform current or wind. We focus on coordination along D-curves, but the methods can be used for arbitrary convex curves in the plane.

4.2 Modeling

4.2.1 Particle Motion in an External Flow

We use a particle model to describe the motion of UAVs in ambient winds, similarly as in [60] [30]. To ensure forward progress of each particle in inertial space, we assume that the ambient flow speed is strictly less than the air speed of the vehicles, i.e., $V_w < V_a$, where V_w is the external flow speed and V_a is the air speed. Identifying the complex plane with the plane of motion, $\mathbb{C} \sim \mathbb{R}^2$, we express the position of each particle by the vector $r_k = x_k + iy_k$, and the flow-relative velocity of each particle by $e^{i\psi_k}$, where $\psi_k \in S^1$ is the orientation of the flow-relative velocity. Without loss of generality, we assume that the ambient flow is aligned with the real axis so that the equations of motion are

$$\begin{aligned} \dot{r}_k &= V_a e^{i\psi_k} + V_w \\ \dot{\psi}_k &= u_k. \end{aligned} \tag{4.1}$$

Here u_k is the turn-rate control signal for the k^{th} particle and $V_w \in \mathbb{R}$. It is convenient to express the equations in terms of the inertial speed and course angle, as opposed to the air-relative speed and heading angle. Following [53] we define the course angle as

$$\chi_k = \arctan\left(\frac{V_a \sin\psi_k}{V_a \cos\psi_k + V_w}\right).$$

Using this definition one can obtain expressions for the inertial speed and course rate of change as follows [53]

$$\begin{aligned} V_{g_k} &= V_w \cos\chi_k + \sqrt{V_a^2 - V_w^2 \sin^2\chi_k} \\ \dot{\chi}_k &= \frac{V_a^2 + V_w(V_{g_k}\cos\chi_k - V_w)}{V_{g_k}^2}\dot{\psi}_k \triangleq \nu_k. \end{aligned}$$

Then equation (4.1) can be written as

$$\begin{aligned} \dot{r}_k &= V_{g_k} e^{i\chi_k} \\ \dot{\chi}_k &= \nu_k. \end{aligned} \tag{4.2}$$

Equation (4.2) will be the model used in the following sections. Notice that, after the change of coordinates, the inertial speed of the particles is heading-dependent, and the new control signal is the course rate-of-change as opposed to the heading rate-of-change. In practice, the control u_k is calculated from ν_k. Note the mapping $\nu_k \mapsto u_k$ is invertible under the assumption $V_w < V_a$, which implies $V_{g_k} > 0$ [53].

4.2.2 Curve Model

Circle If $\nu_k = \omega_0 V_{g_k}$, $\omega_0 \neq 0$, particle k orbits a circle with radius $|\omega_0|^{-1}$ and fixed center

$$c_k = r_k + \omega_0^{-1} i e^{i\chi_k}, \tag{4.3}$$

since, along solutions of (4.2),

$$\dot{c}_k = (V_{g_k} - \omega_0^{-1}\nu_k)e^{i\chi_k} \equiv 0. \tag{4.4}$$

Convex loop If $\nu_k = \kappa_k V_{g_k}$, where $\kappa_k = \kappa(\chi_k) \neq 0$ is the curvature of a strictly convex loop C, then particle k orbits C and the center c_k of C is fixed.[1] Let $\rho_k = \rho(\phi(\chi_k)) = r_k - c_k$, where $\rho : \phi \mapsto \rho(\phi)$ and $\phi : \chi_k \mapsto \phi(\chi_k)$ are smooth maps. If the inertial velocity of particle k is tangent to C, then [51]

$$e^{i\chi_k} = \left|\frac{d\rho}{d\phi}\right|^{-1} \frac{d\rho}{d\phi}, \tag{4.5}$$

and

$$\kappa(\chi_k) = \frac{d\chi_k}{d\sigma}, \tag{4.6}$$

where

$$\sigma(\phi) = \int_0^{\phi} \left|\frac{d\rho}{d\bar{\phi}}(\bar{\phi})\right| d\bar{\phi} \tag{4.7}$$

[1] The center of the convex curve can be, in general, any fixed point inside the region enclosed by the curve that stays fixed with respect to the curve. For common geometrical shapes, such as circles and ellipses, analytical expression exists between the curve center and a point on the curve. (Also see [51].)

is arc length. Using (4.5)–(4.7), we find

$$\kappa_k^{-1} = \frac{d\sigma}{d\chi_k} = \frac{d\sigma}{d\phi}\frac{d\phi}{d\chi_k} = \left|\frac{d\rho}{d\phi}\right| \frac{d\phi}{d\chi_k} \tag{4.8}$$

and

$$\frac{d\rho}{d\chi_k} = \frac{d\rho}{d\phi}\frac{d\phi}{d\chi_k} = e^{i\chi_k}\kappa_k^{-1}. \tag{4.9}$$

Therefore, along solutions of (4.2) with $\nu_k = \kappa_k V_{g_k}$, we have

$$\dot{c}_k = \dot{r}_k - \frac{d\rho}{d\chi_k}\dot{\chi}_k = (V_{g_k} - \kappa_k^{-1}\nu_k)e^{i\chi_k} \equiv 0, \tag{4.10}$$

that is, the center of the convex loop is fixed in the plane.

4.3 Particle Motion Coordination

4.3.1 Decoupled Curve Control

Lemma 4.3.1. *[51] [69] For the model (4.2) with control input*

$$\nu_k = \kappa_k V_{g_k},$$

particle k travels along the curve C with the curve center fixed in inertial space.

The proof follows immediately from (4.10). (Also see [51].)

4.3.2 Translation Invariant Control

We derive decentralized control laws that drive particle k around a strictly convex loop C with an arbitrary center c_0 fixed in inertial space. Define the potential function

$$S(\boldsymbol{r}, \boldsymbol{\chi}) = \frac{1}{2}\langle \boldsymbol{c}, P\boldsymbol{c}\rangle,$$

where P is [60]

$$P = \operatorname{diag}\{\mathbf{1}\} - \frac{1}{N}\mathbf{1}\mathbf{1}^T, \qquad \mathbf{1} = [1, \ldots, 1]^T \in \mathbb{R}^N$$

and $\boldsymbol{c}$ is the vector of curve centers c_k. Note that $S(\boldsymbol{r}, \boldsymbol{\chi}) \geq 0$ and $S(\boldsymbol{r}, \boldsymbol{\chi}) = 0$ if and only if $\boldsymbol{c} = c_0\mathbf{1}$, which implies that all the curve centers coincide. The time derivative of $S(\boldsymbol{r}, \boldsymbol{\chi})$ along the solutions of (4.2) is

$$\dot{S}(\boldsymbol{r}, \boldsymbol{\chi}) = \sum_{j=1}^{N} \langle \dot{c}_j, P_j \boldsymbol{c} \rangle = \sum_{j=1}^{N} \left\langle e^{i\chi_j}, P_j \boldsymbol{c} \right\rangle (V_{g_j} - \kappa_j^{-1} \nu_j), \tag{4.11}$$

where P_j denotes the j^{th} row of P.

Lemma 4.3.2. *[51] [69] The trajectories of the system (4.2) with the control*

$$\nu_k = \kappa_k (V_{g_k} + K_0 \left\langle e^{i\chi_k}, P_k \boldsymbol{c} \right\rangle) \tag{4.12}$$

converge to a state where all particles orbit around the strictly convex curve C with a common center.

Proof: The function $S(\boldsymbol{r}, \boldsymbol{\chi})$ is positive definite in the reduced space of relative curve centers, and its rate can be computed using (4.11) and control (4.12)

$$\dot{S}(\boldsymbol{r}, \boldsymbol{\chi}) = -K_0 \sum_{j=1}^{N} \left\langle e^{i\chi_j}, P_j \boldsymbol{c} \right\rangle^2 .$$

$\dot{S}(\boldsymbol{r}, \boldsymbol{\chi}) = 0$ if and only if

$$\left\langle e^{i\chi_k}, P_k \boldsymbol{c} \right\rangle \equiv 0 \quad k = 1, \ldots, N$$

By LaSalle's invariance principle, all trajectories converge to a set Λ where $\langle e^{i\chi_k}, P_k \boldsymbol{c} \rangle = 0$. In this set $\dot{\chi}_k = \kappa_k^{-1} V_{g_k}$ and $\dot{c}_k = 0$, therefore all solutions in Λ must satisfy $P\boldsymbol{c} = 0$. Since the nullspace of P is the space spanned by $\mathbf{1}$, $\boldsymbol{c} = c_0\mathbf{1}$, in order to satisfy the invariance principle, which is equivalent to the condition that the centers coincide. Application of Lemma 4.3.1 completes the proof. □

4.3.3 Time-splay Coordination

In this section we extend the control law to enforce convergence to the critical set of a phase potential, such that the particles are equally separated in time. It is infeasible to

maintain a constant spatial separation between constant-speed particles moving in an external flow. The temporal separation is a more appropriate means for spatiotemporal regulation [53]. Integrating the closed-loop phase dynamics

$$\dot{\chi}_k = \kappa_k V_{g_k} \tag{4.13}$$

yields

$$t = \int_0^{\chi_k} \frac{d\chi}{\kappa(\chi)s(\chi)}. \tag{4.14}$$

The *time-phase* is [53]

$$\Psi_k = \Psi(\chi_k) = \frac{2\pi}{T}\int_0^{\chi_k} \frac{d\chi}{\kappa(\chi)s(\chi)}, \tag{4.15}$$

where $T > 0$ is the period of a single orbit,

$$T = \int_0^{2\pi} \frac{d\chi}{\kappa(\chi)s(\chi)}. \tag{4.16}$$

Along solutions of (4.2) we have

$$\dot{\Psi}_k = \frac{2\pi}{T}(\kappa_k V_{g_k})^{-1}\nu_k. \tag{4.17}$$

Consider the composite potential

$$V(\boldsymbol{r}, \boldsymbol{\chi}) = S(\boldsymbol{r}, \boldsymbol{\chi}) + \frac{T}{2\pi}U(\boldsymbol{\Psi}), \tag{4.18}$$

where $S(\boldsymbol{r}, \boldsymbol{\chi}) = (1/2)\langle \boldsymbol{c}, P\boldsymbol{c}\rangle$ and $U(\boldsymbol{\Psi})$ is a rotationally symmetric phase potential. Rotational symmetry of U implies $\sum_{j=1}^{N} \frac{\partial U}{\partial \Psi_j} = 0$. Along solutions of (4.2) we have

$$\begin{aligned} \dot{V} &= \sum_{j=1}^{N} \langle e^{i\chi_j}, P_j\boldsymbol{c}\rangle (V_{g_j} - \kappa_j^{-1}\nu_j) + \frac{T}{2\pi}\frac{\partial U}{\partial \Psi_j}\dot{\Psi}_j \\ &= \sum_{j=1}^{N} \left(V_{g_j}\langle e^{i\chi_j}, P_j\boldsymbol{c}\rangle - \frac{\partial U}{\partial \Psi_j}\right)(1 - (\kappa_j V_{g_j})^{-1}\nu_j). \end{aligned} \tag{4.19}$$

Choosing the control law

$$\nu_k = \kappa_k V_{g_k}\left(1 + K\left(V_{g_k}\langle e^{i\chi_k}, P_k\boldsymbol{c}\rangle - \frac{\partial U}{\partial \Psi_k}\right)\right), \quad K > 0, \tag{4.20}$$

enforces convergence of all particles to C with a phase arrangement in the critical set of U.

Theorem 4.3.3. *[60] [53] Consider the particle dynamics (4.2) with a smooth rotationally symmetric phase potential $U(\mathbf{\Psi})$. The control law (4.20) enforces convergence of solutions to the set where all particles travel around the same convex curve, and the curve centers stay fixed in inertial space. The phase arrangement is in the critical set of $U(\mathbf{\Psi})$.*

Proof: See [60, Theorem 3].

We coordinate the time-phase on C by choosing $U(\mathbf{\Psi})$ to be an (M, N)-pattern potential [60]. An (M, N)-pattern is a symmetric arrangement of phases consisting of M clusters uniformly spaced around the curve. In each cluster there are N/M particles. As an example $(M, N) = (2, 4)$ drives four particles into two clusters of two that are uniformly separated along the curve. The (N, N)-pattern is the so-called splay pattern, in which the time-phases are uniformly separated; we call this the time-splay formation.

4.4 Strictly Convex Approximation of Linear Path Segments

In Chapter 3, a method was presented to find minimum time planar trajectories in a steady, uniform wind between initial and final states. The minimum-time trajectories can be used in a variety of important applications, such as finding the optimal re-initialization paths for UAVs in environmental sampling missions, as discussed presently. In these applications the minimum-time paths constitute a segment of the closed curve that is to be tracked. Since the UAV coordination algorithm described in Section 4.3 only considers strictly convex curves, i.e., curves with nonzero curvature, it is desired to approximate the straight (zero curvature) segments with curved segments. Here we establish that for closed convex curves such an approximation is always possible.

Consider a smooth convex curve parameterized by the parameter t that contains a segment where the curvature is zero. Consistently with notation in Chapter 3, let us denote the path parameter where the straight segment begins by t_A and the path parameter value where it ends by t_β. Let us denote the spatial points corresponding to the these values by $\eta(t_A)$ and $\eta(t_\beta)$. The corresponding normal vectors are $i\dot{\eta}(t_A)$ and $i\dot{\eta}(t_\beta)$ (see Figure 4.3).

Remark 4.4.1. *The normals here are defined to point towards the inside of the convex curve. In case of clockwise orbits, this corresponds to a positive rotation about the z-axis. If the orbits are counter-clockwise, the corresponding normals are $-i\dot{\eta}(t_A)$ and $-i\dot{\eta}(t_\beta)$.*

Then the lines defined by the normal vectors at these points are parallel:

$$\eta(t_A) + R_A i\dot{\eta}(t_A) \parallel \eta(t_\beta) + R_B i\dot{\eta}(t_\beta), \tag{4.21}$$

for $R_A, R_B \in \mathbb{R}$. Moreover, as shown in the following lemma, for any specified $R_0 \in \mathbb{R}$ large enough, there exist small parameters t_A^ϵ and t_β^ϵ such that the lines normal to $\eta(t_A - t_A^\epsilon)$ and $\eta(t_\beta + t_\beta^\epsilon)$ intersect at a single point

$$c = \eta(t_A - t_A^\epsilon) + R_0 i\dot{\eta}(t_A - t_A^\epsilon) = \eta(t_\beta + t_\beta^\epsilon) + R_0 i\dot{\eta}(t_B + t_B^\epsilon). \tag{4.22}$$

Lemma 4.4.2. *[69] Consider a smooth, closed, convex curve $\eta(t) \in C^2$, $t \in [t_0, t_f]$, $\eta(t_0) = \eta(t_f)$, $t_f = t_0 + T$ that has a zero curvature segment, where*

$$\kappa(t) = 0, \quad t \in (t_A, t_\beta),$$

for $t_0 < t_A < t_\beta < t_f$, and nonzero curvature segments, where

$$\kappa(t) \neq 0, \quad t \in \{[t_0, t_A], [t_\beta, t_f]\},$$

and the curvature along the curved segments has the same sign. There exists a circle of radius R_0 that is tangent to curve $\eta(t)$ at points $\eta(t_A - t_A^\epsilon)$ and at $\eta(t_\beta + t_\beta^\epsilon)$, for some values of $t_A^\epsilon \in (0, t_A - t_0]$, $t_\beta^\epsilon \in (0, t_f - t_B]$. The radius R_0 can be chosen arbitrarily large.

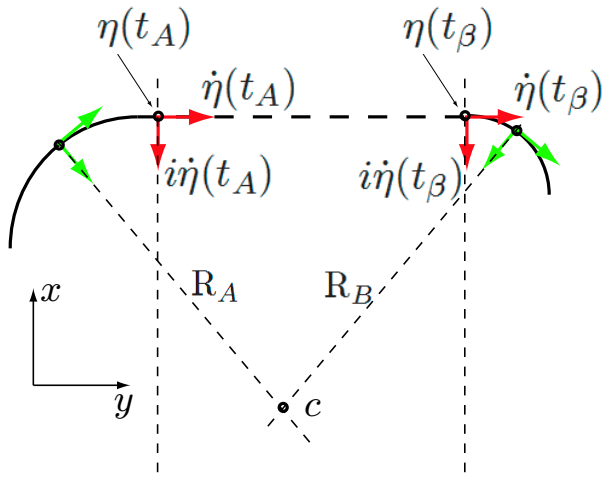


Figure 4.3 Approximating the zero curvature segment with a circular arc.

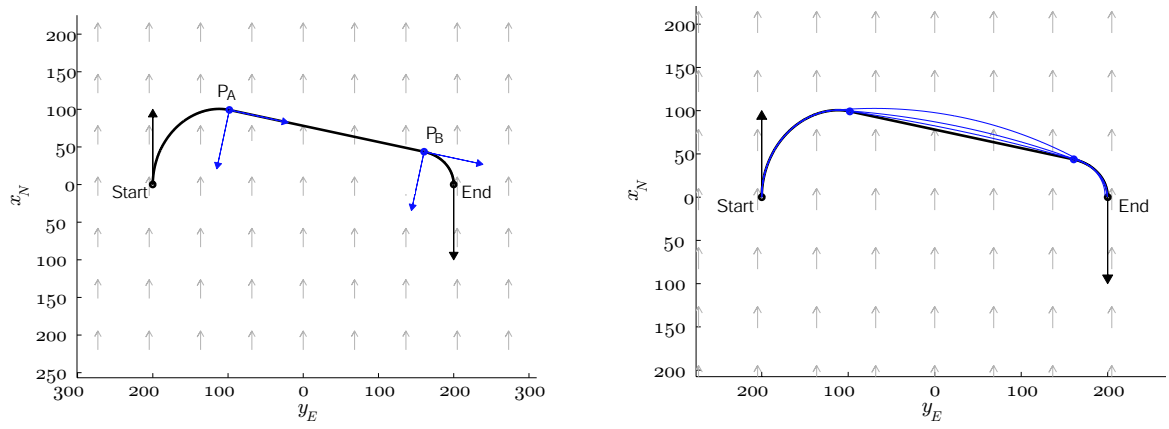
Proof: At the points $\eta(t_A)$ and $\eta(t_\beta)$ the normals are parallel by (4.21). Pick small $t_A^\epsilon > 0$ (leave $t_\beta^\epsilon = 0$) such that the normals intersect at point c, and $||c-\eta(t_\beta)|| < ||c-\eta(t_A-t_A^\epsilon)||$. Such a choice is always possible by continuity of $\eta(t)$ and $i\dot{\eta}(t)$. Now pick $t_\beta^\epsilon(t_A^\epsilon)$ such that

$$R_B(t_\beta^\epsilon) = ||c - \eta(t_\beta + t_\beta^\epsilon)|| \equiv ||c - \eta(t_A - t_A^\epsilon)|| = R_A(t_\beta^\epsilon) = R_0.$$

Then the points $\eta(t_A - t_A^\epsilon)$, $\eta(t_\beta + t_\beta^\epsilon)$ and c form an isosceles triangle with c at the corner of equilateral edges.

To show that such a $t_\beta^\epsilon = t_\beta^\epsilon(t_A^\epsilon)$ exists, assume that it doesn't. Define the continuous function $f(t_\beta^\epsilon) = R_A(t_\beta^\epsilon) - R_B(t_\beta^\epsilon)$. Since $||c - \eta(t_\beta)|| < ||c - \eta(t_A - t_A^\epsilon)||$, it follows that $f(0) > 0$. By convexity of $\eta(t)$ there exists a value $t_{\beta\max}^\epsilon$ where points $c(t_{B\max}^\epsilon)$ and $\eta(t_A - t_A^\epsilon)$ coincide, thus $f(t_{\beta\max}^\epsilon) = -R_B(t_{\beta\max}^\epsilon) < 0$. By continuity of $f(\cdot)$ there has to be a $t_\beta^\epsilon \in [0, t_{\beta\max}^\epsilon]$ value where $f(t_\beta^\epsilon) = 0$. □

In practice, one specifies a large value of the radius R_0 and executes a numerical root-finding algorithm to determine t_A^ϵ and the corresponding value t_β^ϵ (roots of equation (4.22)).



(a) Minimum-time path.

(b) Approximating circular arcs.

Figure 4.4 Straight portions of closed convex curves can be approximated by circular arcs to arbitrary precision. The figure on the left shows a curve that was generated by the path planning algorithm of Chapter 3. The straight segment is then approximated by circular arcs of radii $R_0 = 700\text{m}$, $R_0 = 1000\text{m}$ and $R_0 = 2000\text{m}$, as shown in the figure on the right. The approximation method is discussed in Section 4.4.

4.5 Parametrization of D-Curves

Let $\eta(t) \in \mathbb{C} \sim \mathbb{R}^2$ represent a C^2 curve parameterized by the path parameter t. A C^2 curve in the plane can be uniquely described by specifying its curvature at any instant along the curve. (Also see Theorem 2.2.10.) The parametrization of the D-curves can be written as

$$\kappa(t) = \begin{cases} \frac{V_a^2 + V_w(V_g(t)\cos\chi(t) - V_w)}{V_g(t)^3}\bar{u} & t \in [0, t_A] \\ 0 & t \in [t_A, t_\beta] \\ \frac{V_a^2 + V_w(V_g(t)\cos\chi(t) - V_w)}{V_g(t)^3}\bar{u} & t \in [t_\beta, t_C] \\ \frac{1}{R} & t \in [t_C, T], \end{cases} \tag{4.23}$$

where $\bar{u}$ is the turn-rate limit used by the path generation algorithm to determine the time-optimal trajectory, and the inertial speed $s(t)$ and course angle $\chi(t)$ implicitly depend on the path parameter (time, in this case). In the above definition, the first and third

entries correspond to the maximum turn-rate segments, the second entry corresponds to the straight segment, and the last entry corresponds to a circular segment with radius R. (Note that turn-rate here refers to the vehicle heading rate, which is constrained by structural limits, etc., as opposed to course rate, which depends on the ambient flow.) The first three entries correspond to the time-optimal reset path, and the last entry corresponds to the sampling interval, where the radius of curvature is held constant for the duration of a semi-circle. Notice that the time-optimal reset path has a zero curvature segment. To make the setting amenable to the coordination algorithm described in Section 4.3, we will approximate the zero curvature segment with a circular arc of radius $R_0 \gg R$. In the limit as $R_0 \to \infty$ we obtain the straight segment. Having done so, the definition of the D-curve becomes

$$\kappa(t) = \begin{cases} \frac{V_a^2 + V_w(V_g(t)\cos\chi(t) - V_w)}{V_g(t)^3}\bar{u} & t \in [0, t_A - t_A^\epsilon] \\ \frac{1}{R_0} & t \in [t_A - t_A^\epsilon, t_\beta + t_\beta^\epsilon] \\ \frac{V_a^2 + V_w(V_g(t)\cos\chi(t) - V_w)}{V_g(t)^3}\bar{u} & t \in [t_\beta + t_\beta^\epsilon, t_C] \\ \frac{1}{R} & t \in [t_C, T]. \end{cases} \tag{4.24}$$

4.6 Simulation Results

In the previous section we described the control-volume sampling problem performed by two autonomous vehicles. In the discussed setting the vehicles are sampling at the same altitude and ideally begin their sampling turn simultaneously. To ensure this we employ the coordination technique described in Section 4.3 to properly phase the vehicles. Notice that in the setting of Section 4.3, the vehicles share the same closed curve, and in the control-volume sampling problem the curves are different. However, due to the symmetry in the problem, the time it takes for the vehicles to complete a full period is the same for the upwind sampler as for the downwind sampler; i.e., T in equation (4.16) is the same for both vehicles. This will ensure rotational symmetry in the time-phase (4.14).

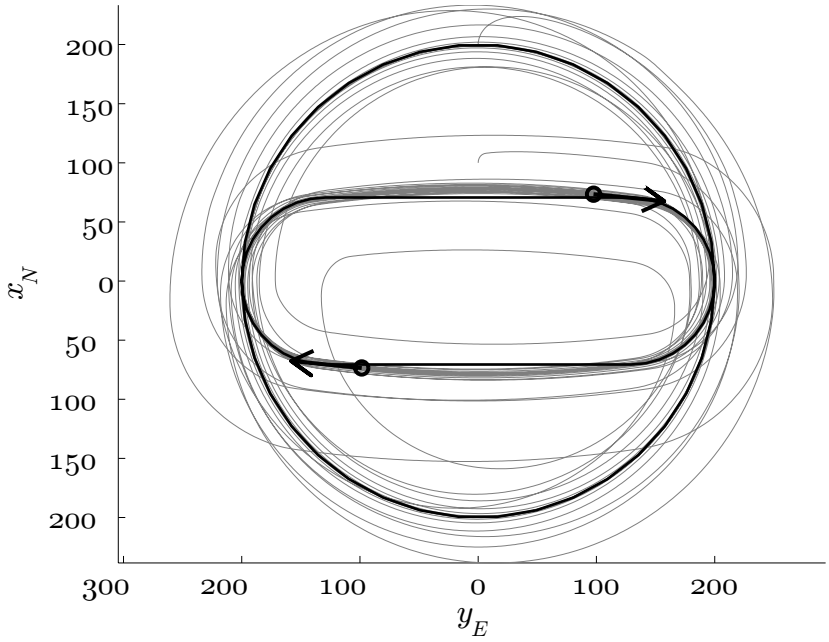


Figure 4.5 Simulation results of the time-splay coordination algorithm for control-volume sampling. Wind speed is $V_w = 0$.

Initializing $t = 0$ in equation (4.14) for both vehicles at the instant they start their sampling interval ensures that the synchronized $(M, N) = (1, 2)$ pattern corresponds to a setting where both vehicles start and finish their sampling turn at the same time.

Remark 4.6.1. *Notice that in this case the $(M, N) = (1, 2)$ pattern corresponds to the splay state, because the UAVs do not share the same path. Coordination is possible because the time it takes to complete the orbit is the same for both curves.*

The coordination algorithm has been simulated and the results can be seen in Figures 4.5–4.7. The figures show three different wind speed cases: $V_w = 0$ m/s, $V_w = 5$ m/s and $V_w = 15$ m/s. In all three cases the UAV air speed was chosen as $V_a = 20$ m/s. After convergence the two UAVs are equally separated temporally, and this separation is conserved during the entire loop.

The results demonstrate that the algorithm is immediately applicable for implementa-

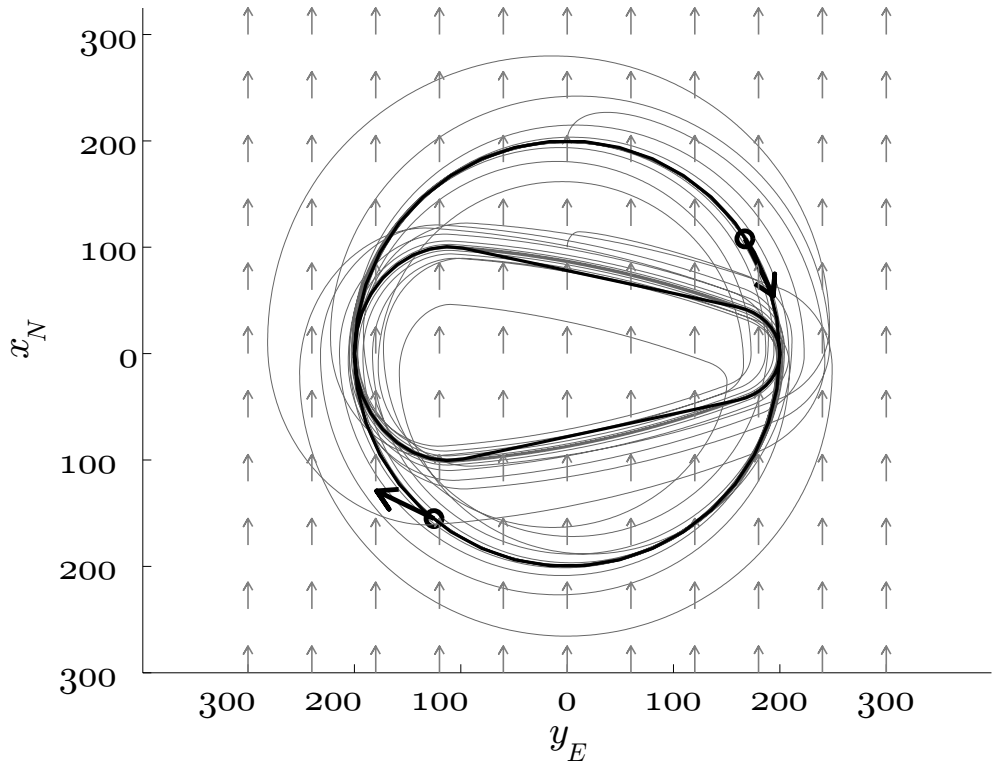


Figure 4.6 Simulation results of the time-splay coordination algorithm for control-volume sampling. Wind speed is $V_w = 5\ \mathrm{m/s}$ from the South.

tion. The desired radius for the sampling mission has been chosen as $R = 200\ \mathrm{m}$. The path has been designed with a maximum steady turn rate $\bar{u} = 0.2832\ \mathrm{rad/s}$ that corresponds to a bank angle limit of $\phi_{\max} = 30^\circ$ on the aircraft bank angle. Note that this restriction does not seem to impact the coordinated control results, which do not presently address turn rate limits.

The above simulations demonstrate the effectiveness of the planar coordination algorithm; however, the control-volume sampling problem is inherently 3-dimensional. If one were to execute the control-volume sampling in purely planar configuration, one might miss the sporangia plume even on a downwind sampling arc, depending on the shape of the plume. In an actual sampling mission it may be beneficial to perform the control-volume sampling repetitively at multiple altitudes: this way one can perform a "vertical

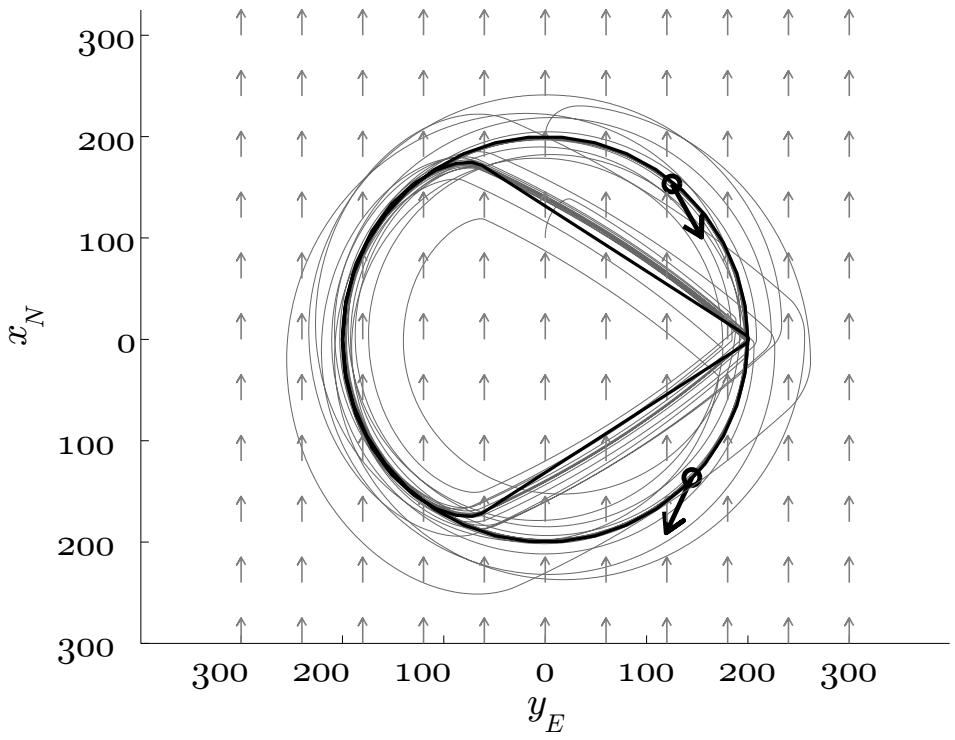


Figure 4.7 Simulation results of the time-splay coordination algorithm for control-volume sampling. Wind speed is $V_w = 15$ m/s from the South.

scan" of the plume. During the time-optimal re-initialization intervals the UAVs don't sample the air, hence this interval can also be used to change the altitude of the vehicles. Since the time it takes to complete the re-initialization path is the same for both vehicles, the same vertical rate command nominally takes the vehicles to the same desired altitude. Figure 4.8 shows simulation results for the 3-dimensional control-volume sampling.

4.7 Hardware-in-the-Loop Simulation

The proposed method has been implemented for real-time execution and tested in hardware-in-the-loop (HIL) simulations. The motion of the two UAVs was simulated on two separate PCs that provided simulated sensor and telemetry information to the autopilots.

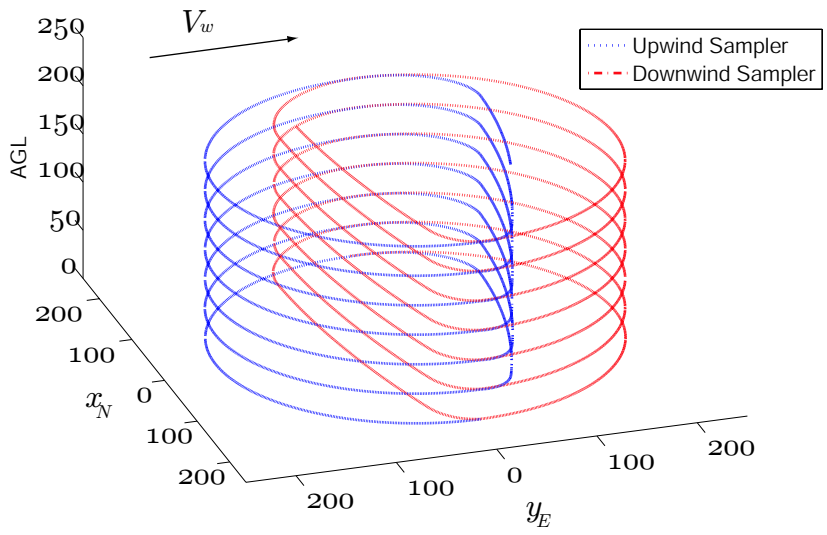


Figure 4.8 Coordinated flight of two UAVs in a 3-dimensional control-volume sampling mission. The UAVs perform a vertical scan of the sporangia plume to estimate the rate of sporangia generation inside the volume.

The autopilots received the simulated telemetry data and ensured stable wings-level or equilibrium turning flight. The telemetry information from the autopilot was also shared with two onboard UAV PC-104 computers over serial RS-232 link. The PC-104 computers were equipped with wireless mesh networking cards to share relative position and phase information that was necessary for the coordination algorithm. Based on these data the PC-104 computers calculated the desired turn-rate commands according to the coordinated control law described by equation (4.20). The turn rate commands were then sent to the autopilots for execution.

The air speed used in HIL simulations was $V_a = 20$ m/s. The wind speed was $V_w = 5$ m/s. The minimum-time paths were designed with the assumption that the maximum bank angle is $\phi_{\max} = 25°$. The radius of the circular sampling path was chosen as $R = 500$ m. Figure 4.9 shows the simulation results of the control-volume sampling over the course of a 45-minute sampling mission. Although the UAVs track the desired path and

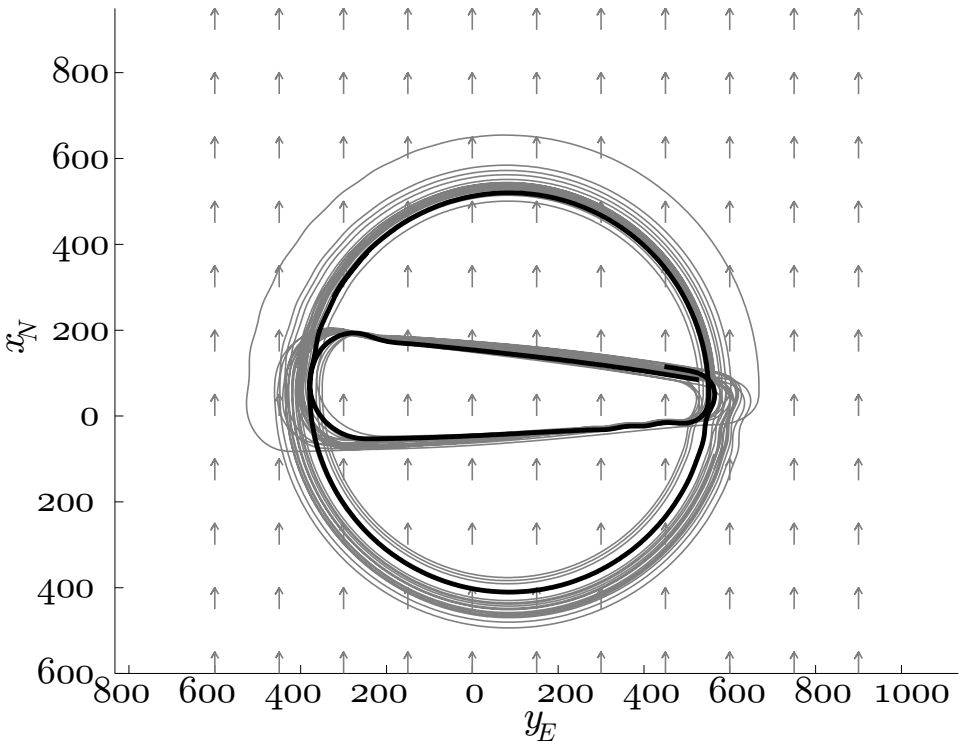


Figure 4.9 Hardware-in-the-loop simulation results for the control-volume sampling problem. The figure shows simulation results over the course of a 45 minute sampling. The bold line shows the last orbit.

coordinate their motion, we can see that the tracking of the desired path is not perfect. The points connecting the different segments around the path correspond to discontinuities in the path curvature. At such points the vehicles would need to change their turn rate instantaneously, which is not possible for fixed-wing aircraft. At these points a slight overshoot can be observed, which is then compensated by feedback. One way to relax the problem is to devise the time-optimal paths of desired to track, assuming a slightly smaller maximum turn-rate bound than the actual true maximum. This leaves room for the controller to compensate by feedback; however, the resulting minimum-time paths become sub-optimal compared to the true solution.

4.8 Summary of the Control-Volume Sampling Method

This chapter described a method to coordinate particle motion on general convex curves in the presence of constant ambient winds. The coordination algorithm controls an arbitrary number of particles to a synchronized state where all particles orbit the same convex loop such that the temporal separation between them is equal and constant. The closed convex curves can be composed of any number of arbitrary segments, and, in particular, they may contain near time-optimal paths where the straight portion is approximated with a circular arc to arbitrary precision.

In the control-volume sampling problem, two UAVs sample the air around the boundaries of a fixed volume to estimate the net rate of outflow of particles across the boundaries. The measurements can be used to estimate the rate of release of particles inside the control-volume. Such measurements are important in aerobiological research to distinguish the generation rate of pathogenic sporangia within a certain field from the background presence of the disease. In such a scenario the UAVs can synchronize their motion to perform consistent sampling during the entire mission and hence ensure that the samples are not independent.

Although the main motivation of the present work was to enable UAVs with advanced sensing capability in aerobiological research, the methods are more general and may be used for other applications as well. The methods are applicable, for example, to a team of UAVs tasked with coordinated perimeter surveillance. That topic is discussed in Chapter 6.

Chapter 5

Aerobiological Sampling Field Experiments

The concepts and methods described in this work were greatly motivated by the need to improve the precision and accuracy of aerobiological data-collection experiments conducted in the lower atmosphere, involving both theoretical and practical challenges. Small unmanned aerial vehicles have been successfully used in the past to collect aerobiota at several tens to hundreds of feet above ground level (see for example [63, 62, 6, 5] and the references therein). Traditionally, these aerial data- collection experiments were performed by remotely piloted vehicles, for which the precision of the sampling flight in terms of the sampling altitude, airspeed, and flight pattern was entirely dependent on the skill and experience of the pilot. More recently, the advancements in unmanned aerial system technologies made it possible to equip small UAVs with autopilots, onboard computers and communication devices, enabling fully autonomous data collection capability [17].

The sampling experiments described in this chapter were part of a larger multi-investigator project that focused on the validation of atmospheric dispersion models used to predict

the motion of *P. infestans* sporangia in the lower atmosphere. The field experiments were performed in two consecutive years, 2008 and 2009, and similar experiments are scheduled for 2010. During the experiments, a large amount of biological and meteorological data was collected and entered into a database for later analysis. The experimental methods described in this chapter were part of this large effort. The main motivation was to introduce novel methods that could provide additional information content gained from a sampling flight.

In Section 5.3 we present the results of coordinated control experiments that were performed in 2008. Coordinating the flight of the UAVs ensures that the samples are consistent between the flights, so that measurements can be compared with each other. In Section 5.4 we introduce a method for long-distance aerobiological sampling that employs a buoyancy-controlled weather balloon to simulate the motion of the sporangia plume as it is convected with ambient winds. Experimental data that was collected in 2009 is presented.

5.1 Field Experiments

A potato field covering approximately 1.5 acres of NY118 potatoes was established to provide a sufficiently large, continuous plant-canopy to serve as the source of inoculum for the experiments. The field was inoculated with a domestic strain of *P. infestans* two weeks prior to the experiments to allow the disease to spread across the potato field. One way the disease may spread from infected plants to healthy plants is by aerial transport. The disease (potato late blight) results in the formation of necrotic lesions on potato leaves, surrounded by sporangia, as shown in Figure 1.1. The sporangia are released into the atmosphere in the early morning hours and may be picked up by turbulent airflow over the plant canopy and reach higher altitudes within the planetary boundary layer.

One goal of the project is to predict the long-distance transport of *P. infestans* in the atmosphere. Lagrangian-stochastic (LS) simulation models have been successfully applied to model the movement of sporangia within meters of infected potato fields [7]. The accuracy of the collected data is crucial in these validation efforts. The sporangia-concentration estimate, C (number of sporangia/m^3), at different altitudes above the plant canopy is assessed in a two-step process. Since the area source strength, Q (number of sporangia/m^2/s), is crucial in any plume modeling effort, this has to be estimated first. This step is done by estimating the total number of sporangia in the field at the beginning of the day (standing crop of sporangia) and distributing it according to the diurnal sporangial release pattern. Sporangia collecting "rotorod-towers" are employed to collect concentration data at heights 0.5 m, 1 m, and 3 m above the ground. These concentration measurements can be used to reconcile with the estimated data to obtain a more accurate estimate of Q. The estimate of Q is then used in the simulation models to find the calculated concentration estimates at different altitudes.

To measure the concentration and viability of sporangia at several tens to hundreds of meters above ground, UAVs were used that were fitted with sporangia-sampling devices mounted under the wings. UAVs are able to sample large volumes of air in a relatively short amount of time [39]. Although millions of sporangia may be released into the air across a large infected potato field, they may be sparsely distributed at increased distances and altitudes from the source. Hence, sampling a large volume of air during a sampling experiment is necessary. The sampling devices onboard the UAVs collect a cumulative sample across an entire flight that can be later analyzed in the laboratory. As the sample is essentially the accumulation of sporangia that is collected during a certain time-interval, it is crucial to have consistent samples during the experiment.

As part of the field trials, three coordinated control experiments were performed at Virginia Tech's Kentland Farm in Blacksburg, Virginia August 14-15, 2008. UAV flights

were coordinated during peak sporangia release (approximately 8 am to 1 pm daily) from the inoculated potato field. The coordinated sampling was performed by two modified Sig Rascal 110 model airplanes as described in Section 5.2. The UAVs coordinated their flight activity to simultaneously collect sporangia of *P. infestans* in the lower atmosphere using the orbit controller discussed in Section 5.3.1.

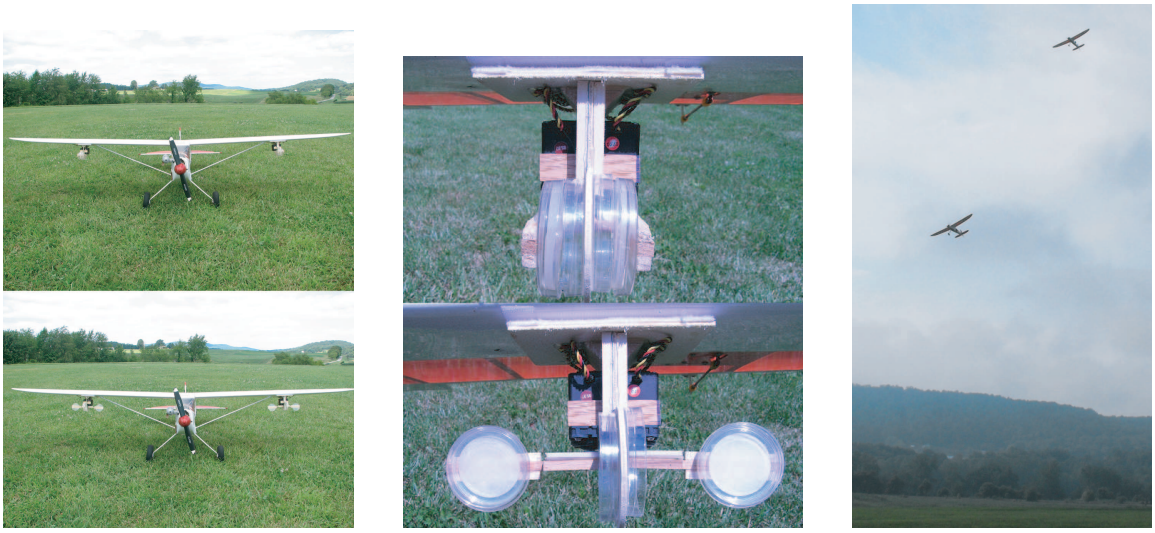
Coordinating the flight of the UAVs ensures that the samples are consistent between the two flights, so that measurements can be compared with each other. If the flights are not coordinated, then the measurements are independent (e.g., the sampling conditions of the UAVs may change from orbit to orbit).

As a result of a transition to new airframes, the 2009 experiment focusing on long-distance sampling was performed with a new UAV platform, as discussed in 5.2.

5.2 UAV Platforms

The UAVs used in the 2008 coordinated sampling experiments were two modified Sig Rascal 110 model airplanes. The Sig Rascal is a high-wing, box-fuselage model aircraft with a wing span of 110 inches. The airframe weighs approximately 14 lbs empty and can carry an additional 10 lbs of payload. The payload bay of the airplanes was modified to host an onboard computer and the flight-critical electronics including the autopilot. These airframes were originally used as the autonomous aerial vehicle platforms in the Nonlinear Systems Lab (NSL) because they were easily accessible (COTS: commercial-off-the-shelf), easy to assemble, and the largest almost-ready-to-fly airplanes.

During the coordinated sampling flight experiments, it was found that the Sig Rascal airframes became dangerously overweight and obsolete. A replacement for the Sig Rascal UAVs became necessary to serve as the workhorse of the NSL's autonomous aerial research operations, including aerobiological data collection. The Small Platform for Au-



(a) (b) (c)

Figure 5.1 Spore sampling devices used onboard UAVs in aerobiological sampling. Figure 5.1(c) shows two UAVs in a synchronized aerobiological sampling experiment on August 14, 2008.

tonomous Aerial Research Operations (SPAARO) UAV was designed and built in the NSL by Murtha et. al. [49] with this goal.

The SPAARO UAV's maximum gross take-off weight is 55 lbs that includes the payload and all the flight-critical electronics. The wingspan of the UAV is $b = 12$ ft and it has an aspect ratio of $\mathrm{AR} = 9$. The propulsion source is a 5.7 hp two-stroke gasoline engine that spins a Zinger 22×10 propeller in a pusher configuration.

Payload The payload bay of the airplanes can host a variety of sensors and electronic components, including the autopilot. The task of the autopilot is to manage single vehicle control by sending control signals to the control surface servos. Communication with the ground station can be established through a 900 MHz radio link. The addressing of each autopilot in the network is implemented using flow control of data streams in which the ground station polls each avionics unit in a round-robin fashion. The multi-homed

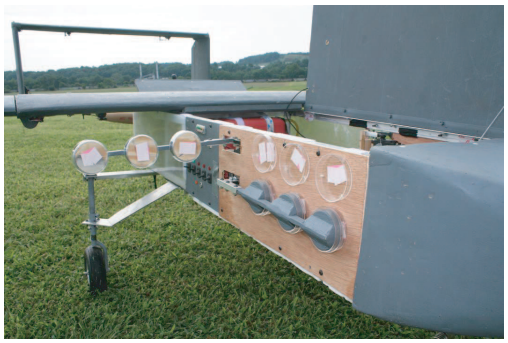
communication implemented in the avionics allows the autopilot to receive packets from two sources: the 900 MHz radio link and an RS-232 serial connection. This allows the autopilot to accept commands for each of its control loops from an onboard computer that manages higher-level coordination tasks and multi-vehicle communication.

The UAV payload further consists of an onboard PC-104+ computer equipped with a 2.4 GHz wireless mesh network card for communication between the UAVs and the ground unit. The computer was a PC-104+ computer EmETX-i701 with an Intel Pentium 4, 1.4 GHz processor with 2 MB L2 cache, and 1 GB DDR SD-RAM. The computer had a PCMCIA extender board, which was used to host the wireless mesh network card. The computer serves as a communication gateway between the autopilot and higher-level mission planning algorithms, such as an algorithm commanding simple GPS waypoints for example. The wireless mesh network enables quick reconfiguration of the network if a communication link is temporarily lost. It also features the possibility of communication between any two network nodes, even if direct link can not be established. Each UAV has the ability to broadcast its position information on the network using Joint Architecture for Unmanned Systems (JAUS) interoperable packets through User Datagram Protocol (UDP). Based on the position information of other UAVs, each UAV can calculate the necessary commands for coordinating its motion in a decentralized way. The commands are then sent to the autopilot for execution.

The UAVs carry special spore sampling devices that can be opened and closed while in flight, as shown in Figure 5.2. When the samplers are opened, the surfaces of the sample devices are exposed to the ambient airflow allowing spores to be deposited on them. In the experiments described in this paper, the surface of the samplers was covered with polycarbonate filter paper coated with 50% glycerol and placed in the center of 1.5% water agar plates. Figure 5.2(b) shows the samplers in closed and open position.

In the coordinated control experiments the speed commands were sent to the autopi-

(a) SPAARO UAV.



(b) Spore samplers mounted of the side of the fuselage of a SPAARO UAV.

Figure 5.2 A SPAARO (Small Platform for Autonomous Aerial Research Operations) UAV developed by Murtha et. al. was used in the experiments [49].

lot through a serial RS-232 link to achieve tracking and coordination, and the altitude loop command was constant. The UAVs' flight pattern was circular and centered on the experimental fields.

In the long-distance sampling experiments the speed command was constant and did not change during the course of the entire mission. The center of the circular flight path was always the current position of the weather balloon to ensure that the UAVs stay inside the spore plume. The GPS coordinates of the balloon were obtained from the onboard data logger of the weather balloon in real time; the coordinates were then reported to the autopilots as circular waypoint commands. The altitude command of the UAV was always the current altitude of the balloon, unless the balloon left a certain domain of pre-defined safety bounds.

5.3 Coordinated Control Experiments Using Two UAVs

This section describes the coordinated control experiments using two autonomous Sig Rascal UAVs. For our experiments, coordinated flight patterns were selected to assess the

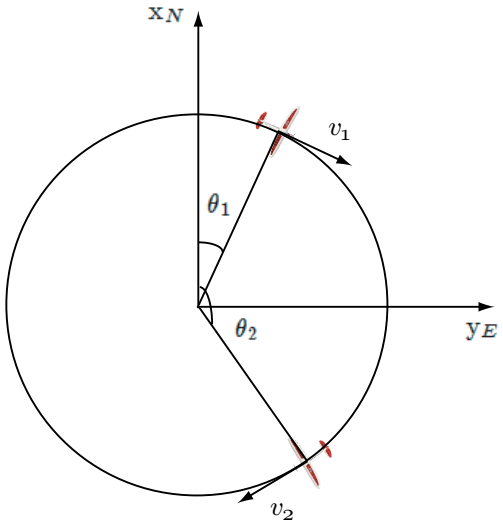


Figure 5.3 UAVs traveling along a circular flight pattern. The desired phase separation θ^* is achieved by controlling the speed of the vehicles.

vertical (two UAVs operating at different altitudes) and horizontal (two UAVs operating at the same altitude but with different orbits) distribution of *P. infestans* sporangia near the infected potato field. The UAVs shared relative position information through a wireless mesh network and coordinated their motion using an orbit controller described in Section 5.3.1.

5.3.1 Coordinated Control via Speed Modulation

In this section we describe a simple coordination algorithm that uses speed commands to control the angular separation between two identical UAVs. The algorithm is referred to as coordination via speed modulation because the synchronized state is achieved by applying small differential changes in the speed commands of each UAV to allow for phase synchronization. In this approach we assume that the tracking problem is handled by the autopilot and both UAVs are on the desired orbit.

Consider the kinematic model of two UAVs

$$\begin{aligned} \dot{x}_{N_k}(t) &= v_k(t)\cos\psi_k(t) \\ \dot{y}_{E_k}(t) &= v_k(t)\sin\psi_k(t), \quad k=1,2 \\ \dot{\psi}_k(t) &= \nu(t) \end{aligned} \tag{5.1}$$

where $x_N(t)$ and $y_E(t)$ are the inertial North and East coordinates with respect to some fixed reference frame, $\psi(t)$ is the heading of the vehicles, and $v_k(t)$ is the airspeed. We assume that the autopilot manages the tracking of a circle of radius R and ensures that the vehicle stays on the path even in the presence of external disturbances (such as wind gusts, etc.) Assume that the inertial origin is located in the center of the circle of interest to track. Let us denote the "phase angle" of the vehicles by

$$\theta_k(t) = \arctan\left(\frac{y_{E_k}(t)}{x_{N_k}(t)}\right), \quad k=1,2, \tag{5.2}$$

and use a simple particle kinematic model for motion around a circle of radius R (as tracked by the autopilot):

$$\dot{\theta}_k(t) = \omega_k(t) = \frac{v_k(t)}{R}.$$

Define

$$\delta\theta(t) = \theta_2(t) - \theta_1(t) - \theta^*, \quad \delta\theta(t) \in [-\pi, \pi] \tag{5.3}$$

as the phase error, where $\theta^* \in [0, \pi]$ is the desired phase advantage of UAV_2 to UAV_1. We assume that

$$v_k(t) = V_a + u_k(t), \quad k=1,2, \tag{5.4}$$

where V_a is the desired average airspeed and u_k is a control signal. Select the control signal as

$$u_k(t) = K(-1)^k \sin\left(\delta\theta(t)\right) \quad k=1,2, \tag{5.5}$$

where $K > 0$. Then the phase error dynamics takes the form

$$\dot{\delta\theta}(t) = -\frac{2K}{R}\sin(\delta\theta(t)). \tag{5.6}$$

Proposition 5.3.1. *The origin of the system (5.6), corresponding to the desired phase arrangement, is almost globally asymptotically stable.*

Proof: Choose the Lyapunov function candidate

$$V(t) = \frac{1}{2}\delta\theta(t)^2 > 0, \quad \forall \delta\theta(t) \neq 0.$$

The rate of V

$$\dot{V}(t) = -\frac{2K}{R}\delta\theta(t)\sin\left(\delta\theta(t)\right) \leq 0, \quad \forall \delta\theta(t) \in [-\pi, \pi]$$

is negative semi-definite and it is equal to zero if and only if $\delta\theta \in \{0, -\pi, \pi\}$. Using Lyapunov's second method we can conclude that the equilibrium point at the origin, which corresponds to the desired phase separation, is stable [31, Theorem 4.1]. To conclude asymptotic stability we resort to LaSalle's invariance principle. Since $V(t) > 0$ and $\dot{V}(t) < 0$, level sets of $V(t)$ define compact, positively invariant sets. Denote such a set Ω. Then trajectories that start inside Ω will converge to the largest invariant set M contained inside the set $E = \left\{\delta\theta(t) : \dot{V}(t) = 0\right\}$ [31]. As we have shown above, this set is $M = \{0, -\pi, \pi\}$, thus trajectories converge to one of the two equilibrium points (identifying $-\pi$ with π, we only have two equilibrium points) for the system. To show that the equilibrium point $\delta\theta(t) = \pi$ is unstable, we resort to local stability analysis. The systems corresponding to the two equilibrium points are

$$\begin{aligned} \delta\theta(t) &= -\frac{2K}{R}\delta\theta(t), & \delta\theta|_{\text{equ}} &= 0, \\ \delta\theta(t) &= \frac{2K}{R}\delta\theta(t), & \delta\theta|_{\text{equ}} &= \pi. \end{aligned}$$

Clearly, the system corresponding to the "out of phase" equilibrium point is unstable. Since the analysis holds in the entire configuration space, excluding the isolated unstable equilibrium point, convergence to the equilibrium point at the origin is almost globally asymptotically stable. □

Independently of this research, a similar orbit controller is presented in [24].

In order to eliminate errors due to slight differences in the calibration of the Pitot tube readings, an integral channel can be added as follows: Define

$$e_k(t) = (-1)^k \sin(\delta\theta(t)) \qquad k = 1, 2.$$

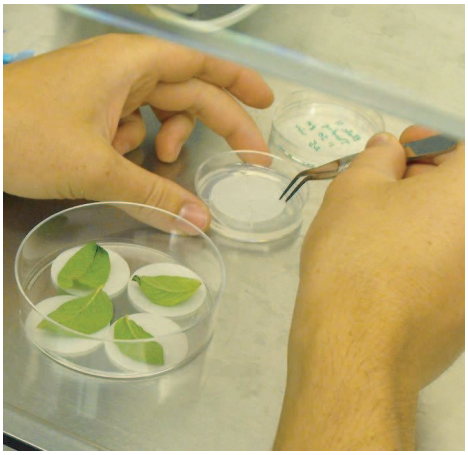
Define a "proportional-integral (PI)" control signal for the k^{th} vehicle:

$$u_k(t) = K_p e_k(t) + K_i \int_0^t e_k(t)dt, \quad k = 1, 2. \tag{5.7}$$

The time-scale for achieving the desired speed is governed by the autopilot and the system parameters and is on the order of seconds. The time-scale for synchronization is orders of magnitude slower. This time-scale separation allows one to treat the UAV dynamics and the synchronization dynamics independently, and assume that the desired velocity can be instantaneously achieved. Proper choice of gains (see also Section 5.3.3) ensures that the synchronization algorithm does not degrade the autopilot's inherent stability.

5.3.2 Processing of Aerobiological Samples from UAVs

Sampling plates were covered and placed in a cooler for transport to the laboratory immediately following the coordinated flights. The filter paper was cut into four pieces in a bio-safety cabinet, and the exposed surface was placed in direct contact with disinfested potato leaflets collected from the greenhouse. Two hundred microliters of sterile deionized water was added to the filter paper to encourage zoospore formation and resulting infection. Cultures were incubated for 3-5 days in the laboratory at ambient room temperature and examined with a microscope to observe symptoms of late blight and signs of *P. infestans*.



(a)

(b)

Figure 5.4 Viability test (able to cause disease on greenhouse-grown potatoes) of *Phytophthora infestans* sporangia. The filter paper is immediately removed from the samplers after the sampling flight and transported to the laboratory.

5.3.3 Experimental Results

Three coordinated control experiments were performed on August 14-15, 2008. The first experiment was conducted on August 14, 2008, 9:30 am EST. The second and third experiments were conducted at 9:30 a.m. and 11:30 a.m. EST on August 15, 2008. Each of the experiments involved two autonomous UAVs that coordinated their flight by sharing their position information through the wireless mesh network. In the first two experiments, two UAVs were vertically stacked at two different altitudes (25 m and 45 m AGL) with identical sampling orbits (radii of 150 m). In the third experiment, two UAVs shared the same altitude (35 m AGL) with different sampling orbits (radii of 130 m and 160 m).

In all experiments the UAVs used the speed control algorithm described by equations (5.4) and (5.7) to coordinate their motion along the flight path. The gains for the speed control algorithm (5.7) were chosen as $K_p = 4$ and $K_i = \frac{K_p}{80}$. The proportional gain was chosen to increase the velocity of the vehicle by 2 m/s if the vehicle's phase lag is $30°$. The integrator was implemented with anti-windup that limits the minimum airspeed at

Flight	Date and Time	AGL Alt.	Radius	Avg. airspeed	Sample time
LT5	08/14/08, 9:30 am	43 m	150 m	22.17 m/s	17 min
LT6	08/14/08, 9:30 am	24.7 m	150 m	20.9 m/s	17 min
LT8	08/15/08, 9:30 am	24.9 m	150 m	20.8 m/s	19 min
LT9	08/15/08, 9:30 am	45 m	150 m	22 m/s	19 min
LT10	08/15/08, 11:30 am	40 m	160 m	26 m/s	16 min
LT11	08/15/08, 11:30 am	34.5 m	130 m	19 m/s	16 min

Table 5.1 Summary of the flight experiments conducted during August 14-15, 2008. Viable spores of *P. infestans* were collected during flights LT5 and LT6.

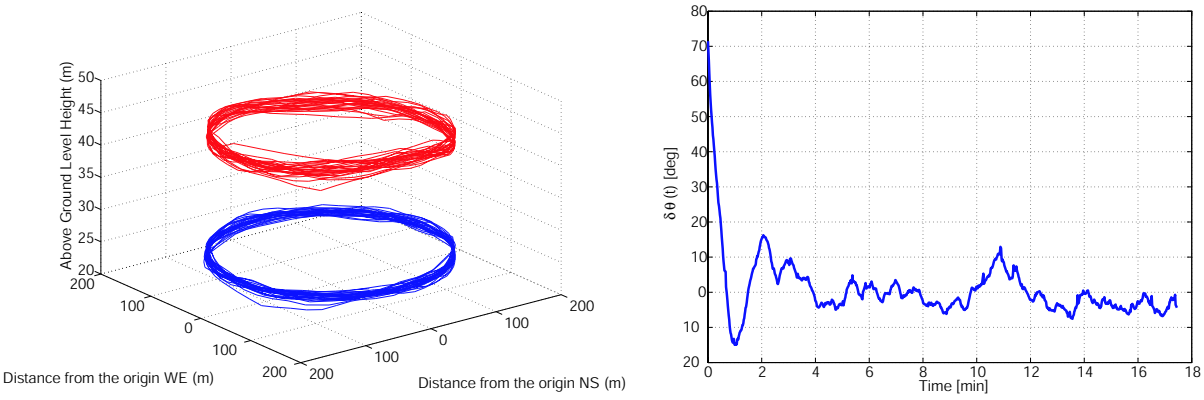
18 m/s and the maximum airspeed at 25 m/s. For the third experiment, where the UAVs were following concentric orbits, a different desired average airspeed, V_a in (5.7), was chosen for each UAV. This guaranteed that in nominal flight the UAVs flew with the same angular rate. The desired angular separation in equation (5.3) was chosen $\theta^* = 0$. This corresponds to the UAVs flying with the same phase $\theta_1(t) = \theta_2(t)$.

Experiment 1: August 14, 2008, 9:30 a.m. EST

A 3D telemetry plot from the first flight can be seen in Figure 5.5(a). The autopilot holds the altitude at the desired value with high precision. For the UAV with DGPS autopilot, the standard deviation for altitude was 0.58 m; for the UAV with regular autopilot, the standard deviation was 1.33 m. In Figure 5.5(b) we can see the time history of the phase error $\delta\theta(t)$. After the algorithm starts, the phase error converges to zero.

Experiment 2: August 15, 2008, 9:30 a.m. EST

A 3D telemetry plot from the second flight can be seen in Figure 5.6(a). In Figure 5.6(b) we can see the time history of the phase error $\delta\theta(t)$. Similarly to the first flight experiment, the



(a) Telemetry plot.

(b) Phase error.

Figure 5.5 3D position data and phase error for two UAVs in a phase synchronized coordinated control experiment on August 14, 2008, 9:30 a.m. EST. UAV1 (bottom) flying at 25 m AGL; UAV2 (top) flying at 45 m.

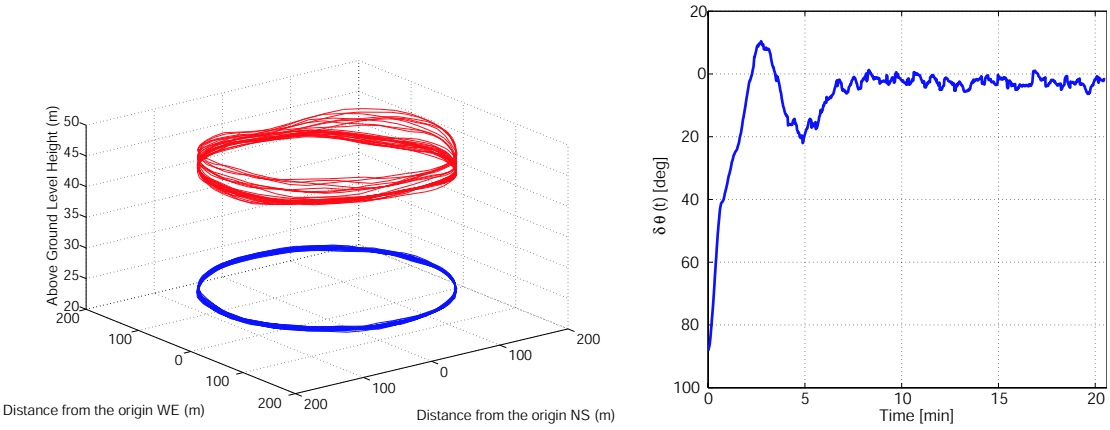
UAVs hold the desired altitude with high precision and converge to a phase synchronized state.

Experiment 3: August 15, 2008, 11:30 a.m. EST

In the third flight the UAVs were flying at the same altitude 35 m AGL. One of them was flying on a $R = 130$ m radius circular flight path, while the other one was flying on a 160 m radius. The telemetry plot and phase error history from the experiment can be seen in Figure 5.7(a) and Figure 5.7(b), respectively.

5.3.4 Collection of Viable Sporangia of *P. infestans*

Viable (able to cause disease on greenhouse-grown potatoes) sporangia of *P. infestans* were recovered from two of the coordinated flights, LT5 and LT6 (Figure 5.4(b)). We did not recover viable sporangia of *P. infestans* from the other sampling missions, but ongoing DNA-based methodologies are currently being optimized in the Schmale laboratory to



(a) Telemetry plot.

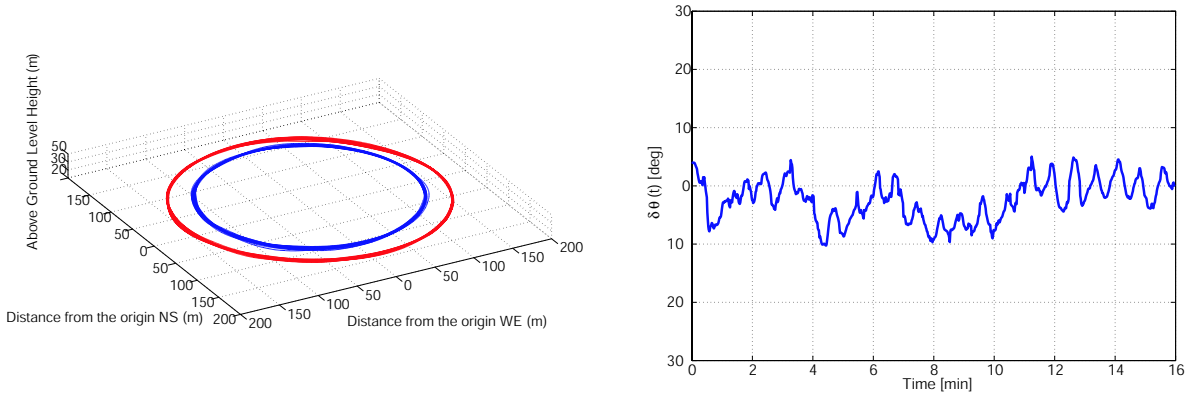
(b) Phase error.

Figure 5.6 3D position data and phase error for two UAVs in a phase synchronized coordinated control experiment on August 15, 2008, 9:30 a.m. EST. UAV1 (bottom) flying at 25 m AGL; UAV2 (top) flying at 45 m.

detect and quantify sporangia of *P. infestans* from these aerobiological samples.

5.4 Long-distance Sampling Using a Buoyancy-Controlled Weather Balloon

In the framework described in this section, the purpose is to assess concentrations of *P. infestans* sporangia at long distances (0.5 to 1 km) from their source (also see [72]). Due to the sparse distribution of sporangia at increased distances from the source, it is important to increase the likelihood of detection by sampling at locations where sporangia are most likely to be found. To monitor the spread of *P. infestans* sporangia, we used a buoyancy-controlled weather balloon to simulate the movement of sporangia in the lower atmosphere (see Figure 5.8). Due to its inertia the balloon is much less prone to capture small turbulent fluctuations in the wind speed and direction that otherwise affect the trace of spores as individual particles, but it is likely to follow the same path that the



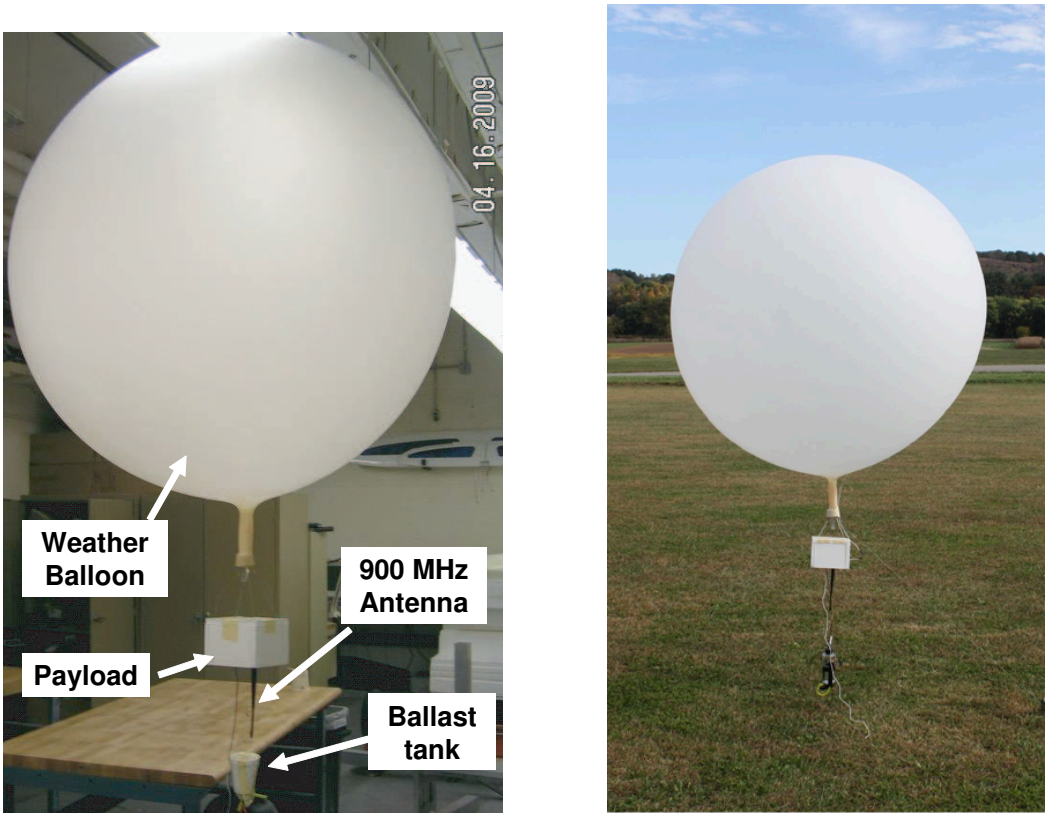
(a) Telemetry plot.

(b) Phase error.

Figure 5.7 Position data and phase error for two UAVs in a phase-synchronized coordinated control experiment on August 15, 2008, 11:30 a.m. EST. Both UAVs fly at the same altitude with radii 130 m and 160 m, respectively.

sporangia plume as a continuous medium follows. Thus, the geographical location of the balloon is expected to indicate a region of the atmosphere containing a high concentration of sporangia where the UAVs are more likely to capture the pathogen with their onboard sampling devices than at other locations.

The tracking of the balloon with UAVs can be implemented by commanding the UAVs to follow a circular flight pattern centered on the current position of the balloon. Assuming that the ambient wind-speed is less than the UAV airspeed ensures that following the balloon in this way is feasible. In case of a steady uniform ambient wind, the corresponding UAV trajectories are *trochoidal* paths [57] (c.f. Figures 2.3 and 3.2). A trochoid is a path that a point P at a distance R from the center of a circle of radius $r < R$ would describe, as the circle is rolling along a straight line. In the setting where the trochoid curves are generated by UAVs flying in winds, the radius r is determined by the wind speed, V_w, and radius R is determined by the air relative speed of the UAVs, V_a.



(a) Balloon in the lab.

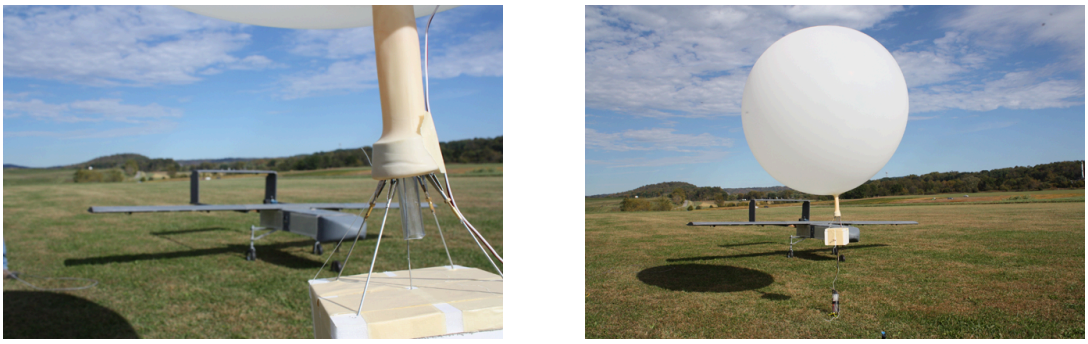
(b) Balloon at Kentland Farm.

Figure 5.8 Photograph of the buoyancy-controlled weather balloon used in aerobiological sampling experiments.

5.4.1 Buoyancy-Controlled Balloon

The purpose of using the weather balloon is to simulate the trace of plant pathogen sporangia in real time as they are being released from the field. The weather balloon is convected in the direction of the wind's motion with the same speed as the ambient airflow.

The geographical location of the balloon is expected to indicate a region of the atmosphere containing sporangia of *P. infestans*, where the UAVs are more likely to collect the pathogen spores with their onboard sampling devices than at other locations. To accurately simulate the sporangia trajectories, it is desirable to release the balloon from ground level at the field; the wind profile changes significantly with height above the ground. It



(a) (b)

Figure 5.9 SPAARO UAV and the buoyancy-controlled weather balloon.

is also desirable that the balloon doesn't ascend too rapidly, since then the balloon would be subject to higher ambient winds much sooner than the majority of spores in the plume would. Ideally, a neutrally buoyant balloon would simulate individual particle trajectories the best. To ensure that the balloon does not run aground, however, a small, steady climb rate is desired.

The weather balloon is a $300\ \mathrm{g}$ sounding balloon that can carry a payload of up to $630\ \mathrm{g}$ and has a nominal inflation diameter of $1.6\ \mathrm{m}$. The balloon can be filled with Helium in order to provide the required lift force that carries the payload into the air (Figure 5.8). The balloon payload consists of a data logger unit, communication devices and a control unit to adjust the buoyancy of the balloon, as shown in Figure 5.10. The telemetry information primarily consists of GPS coordinates, GPS altitude, barometric altitude and temperature. The telemetry is transmitted over a 900 MHz ISM radio band to a ground processing unit, where the telemetry data can be logged and visualized on a PC. It is also possible to relay the position of the balloon to the UAVs. By providing the UAVs with the current GPS coordinate of the balloon, steady orbit around the balloon can be achieved.

The payload of the balloon further consists of a $72\ \mathrm{MHz}$ radio receiver through which command signals can be received from a ground transmitter. The primary purpose of

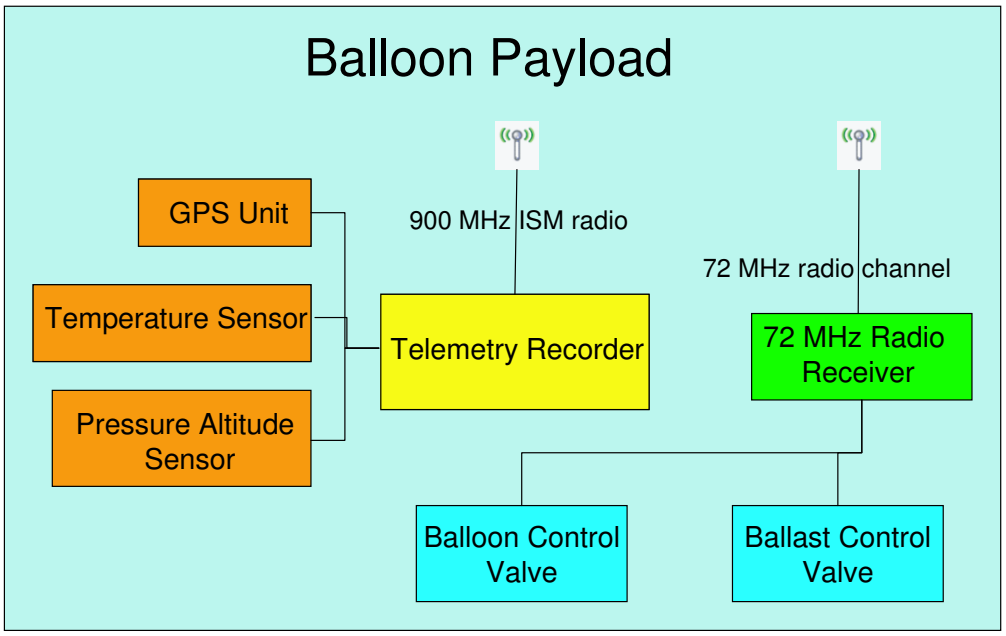


Figure 5.10 Schematic drawing of the balloon payload. The payload mainly consists of the data logger unit, communication devices and buoyancy-control devices.

this channel is to allow remote control of the buoyancy of the balloon. At the bottom of the payload package, a small water container can be filled with a maximum of $200\ \mathrm{ml}$ of liquid to provide ballast. Opening the valve allows water to pass through at a rate of $1\ \mathrm{ml}$ per second. If the valve is opened, the balloon payload loses weight, and consequently the balloon will ascend. The neck of the weather balloon is closed with a cylindrical rubber plug that has a $1/2$ inch diameter hole drilled through the middle. In nominal operation the hole is sealed with a rubber sheet treated with silicon grease to ensure the Helium does not escape through tiny cuts and scrapes on the rubber surfaces. The rubber sheet can be lifted by a push-rod that is attached to a servo motor. The larger distance the pushrod travels, the more Helium can escape. By operating the ballast container valve and the Helium pressure valve, the balloon net buoyancy can be controlled. For neutral buoyancy it is required that the buoyant force equals the total weight of the balloon, that is

$$0 = F_b - W_e - W_{He},$$

where F_b is the weight of the displaced air, W_{He} is the weight of the Helium gas, and W_e is the weight of the equipment that includes the weight of the balloon itself. Ignoring secondary effects like temperature difference and vertical wind components, the volume of the balloon can be calculated as

$$V = \frac{W_e}{g(\rho_{air} - \rho_{He})}.$$

Since the balloon will be operated close to the ground, the air density can be taken as $\rho_{air} = 1.28\ \mathrm{kg/m^3}$. Assuming that the density of the Helium gas inside the balloon is close to the density at atmospheric pressure, $\rho_{He} \approx 0.1786\ \mathrm{kg/m^3}$. Taking the mass of the equipment as $m_e = 0.8\ \mathrm{kg}$ and assuming spherical balloon shape, the balloon radius is $r \approx 0.56\ \mathrm{m}$. Although the above calculation is only approximate, it allows us to estimate the feasibility of the concept and also the expected balloon size.

5.4.2 Hardware-in-the-Loop Simulation

The concept has been simulated using hardware-in-the-loop (HIL) simulations. The motion of two UAVs, including full 6-DOF dynamics, was simulated on two separate PCs. The balloon trajectory was simulated on a third computer and the telemetry information from the balloon was shared with the UAVs over a wireless mesh network. Based on the current position of the balloon, the UAVs updated their orbit center to ensure that they remain in the simulated "plume." The results of the simulation can be seen in Figures 5.11–5.12. In the simulation the airspeed of the UAVs was $20\mathrm{m/s}$ and the wind speed was $2\mathrm{m/s}$ from the West. In that case the direction of the ambient air's motion is $\psi_w = \pi/2$ measured from the North. To simulate variations in the ambient wind conditions, the

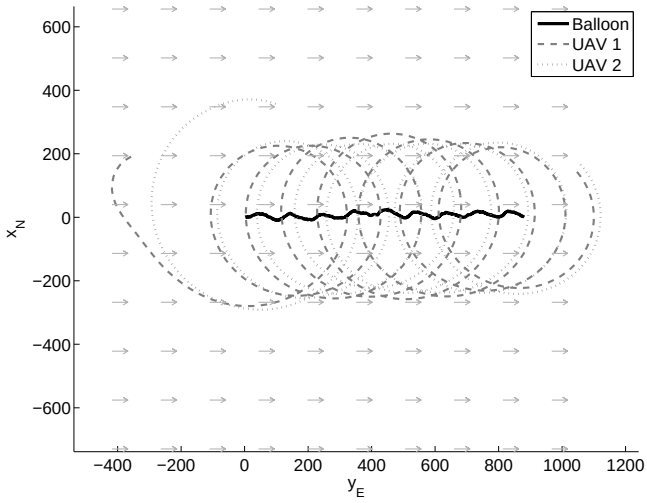


Figure 5.11 Hardware-in-the-loop simulation of the flight of two UAVs tracking a weather balloon. The UAVs' circular flight path is centered on the current position of the balloon; the corresponding UAV trajectories are *trochoid* curves. 2-D plot of the experiments.

wind direction was modified to $\psi_w = \pi/2 + 0.5\sin(\omega_0 t) + \epsilon$, where $\omega_0 = 0.1\mathrm{rad/s}$ and ϵ is a zero mean uniform random variable $\epsilon \in [-0.5, 0.5]$.

5.4.3 Flight Experiments and Results

The flight test was performed on August 6, 2009 at Virginia Tech's Kentland Farm. In the morning hours when the flight test took place, there was a continuous cloud cover, with a temperature around 75 F. The SPAARO UAV take-off time was at 9:58 a.m., EST, by which time the weather balloon was already situated at the potato field, ready to be released. The UAV was immediately sent into orbit around the balloon, while the balloon's GPS coordinates were shared over the wireless network. The balloon was released at 10:02 a.m. EST, at which time the bottom set of samplers of the SPAARO were opened, and the sampling experiment began. The winds were fairly high $V_w \approx 7\ \mathrm{m/s}$, so the flight was very short: approximately two and a half minutes. At that point the mission

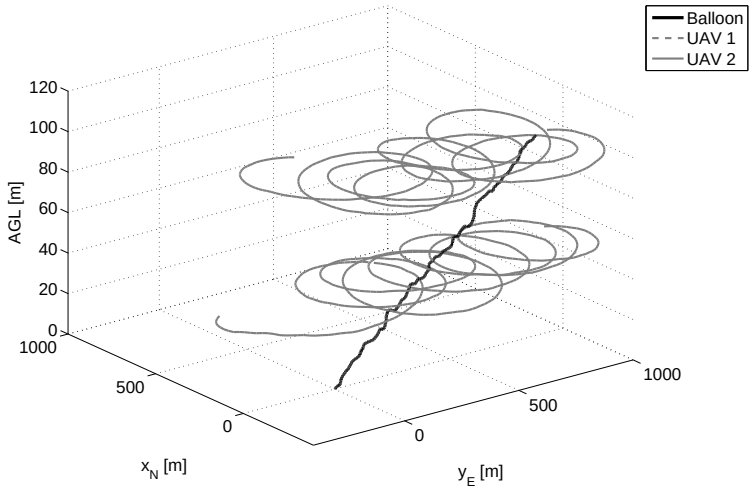


Figure 5.12 Hardware-in-the-loop simulation of the flight of two UAVs tracking a weather balloon. The UAVs' circular flight path is centered on the current position of the balloon; the corresponding UAV trajectories are *trochoid* curves. 3-D plot of the experiments.

was aborted. The planar and 3D telemetry plots of this short sampling flight can be seen in Figures 5.13–5.14. The quantification of *P. infestans* sporangia from the sample plates found that no sporangia were collected during the flight. This can be explained by the relatively short duration of the sampling flight.

The experiments have been repeated on October 8, 2009 to collect additional telemetry data. The telemetry plots from the experiments can be seen in Figures 5.15–5.16.

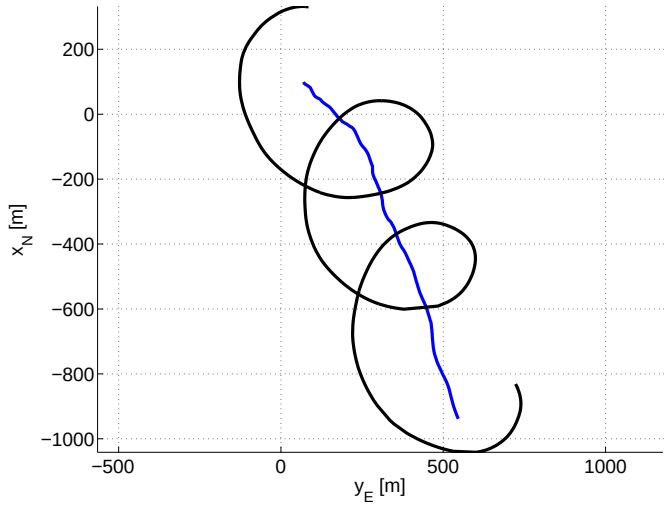


Figure 5.13 Flight test results from August 6, 2009. The UAV tracked the weather balloon after the balloon was released from the potato field. 2-D plot of the experiments.

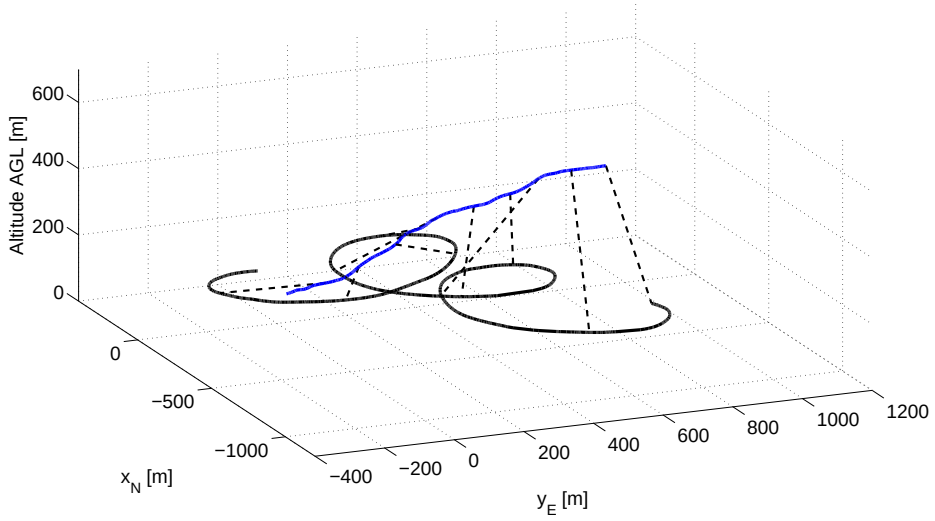


Figure 5.14 Flight test results from August 6, 2009. The UAV tracked the weather balloon after the balloon was released from the potato field. 3-D plot of the experiments.

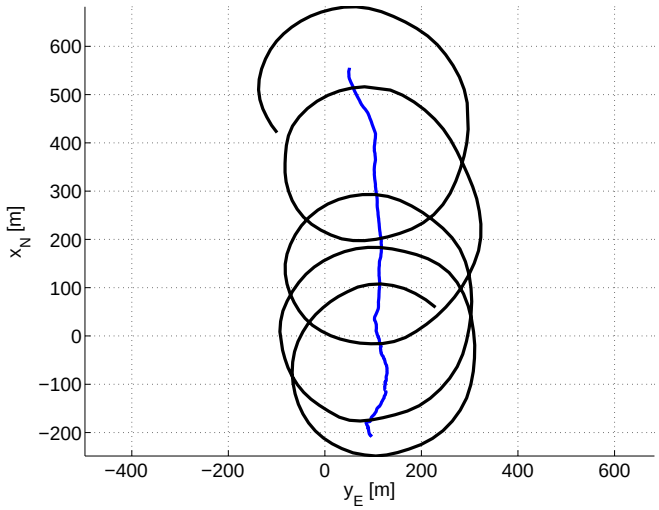


Figure 5.15 Flight test results from October 8, 2009. 2-D plot of the experiments.

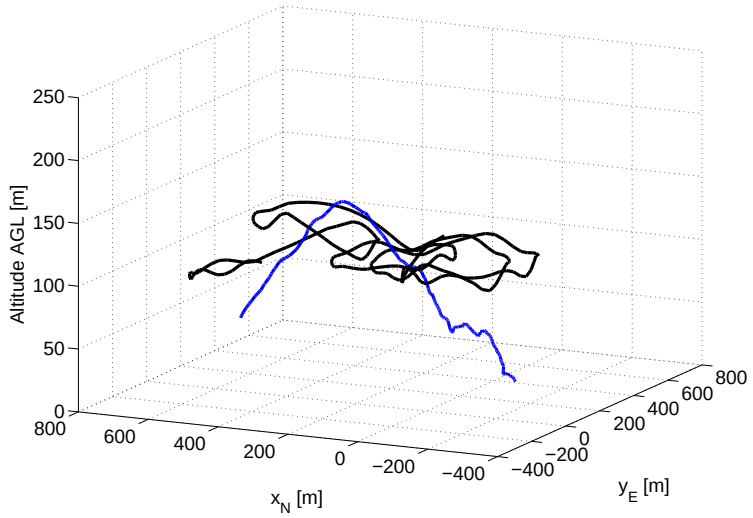


Figure 5.16 Flight test results from October 8, 2009. 3-D plot of the experiments.

Chapter 6

Coordinated Perimeter-Patrol with Minimum-Time Alert Response

This chapter describes a perimeter surveillance algorithm that was developed for UAVs that are tasked with the defense of a military establishment. In previous chapters we introduced a minimum-time path planning procedure (Chapter 3) and a particle coordination algorithm (Chapter 4) that can be used to control vehicle formations to perform aerobiological sampling experiments. The generality of the proposed methods allows them to be used for other applications where the kinematic vehicle model and the assumption of a constant flow-field are reasonable. Here we consider a team of UAVs that can carry downward-looking sensors, cameras for example, to monitor the perimeter of a certain military establishment or base. The role of the UAVs is to detect intrusions along a certain perimeter that bounds the base and to report these intrusions to the appropriate authorities. Nominally the UAVs synchronize their motion along a convex path, the base perimeter, to minimize coverage gaps in space and time. If an intrusion is detected along the patrol perimeter, the next UAV in line prosecutes the alert by following a minimum-time path to the intrusion point. The work described in this chapter has been published

in [54] [68].

6.1 The Perimeter Patrol Problem

Consider a convex perimeter circumscribing a base that must be defended by a collection of N UAVs [54]. We seek an optimal method for patrolling the perimeter *and* responding to intruder alerts along the perimeter, as summarized by the following two objectives:

- In nominal conditions, the UAVs coordinate their motion along the perimeter such that the visitation rate at any given point along the curve is constant.
- If an intruder is detected, one UAV responds in minimum time, while the remaining $N-1$ UAVs continue to patrol the original perimeter.

When a threat is detected, there is a trade-off between these two objectives. In one limiting case, the UAVs don't take any action other than reporting the threat and continuing their original, coordinated flight plan. In this case, only the perimeter-patrol objective is achieved. Alternatively, one of the UAVs (e.g., the one that detected the threat) might divert from the perimeter and loiter above the threat. In this case, the number of vehicles covering the perimeter decreases to $N-1$ and gaps in the perimeter coverage increase.

We propose an approach to perimeter surveillance that simultaneously achieves the two goals. Consider the event in which a threat is detected by one of the UAVs (UAV_1) at point r_f; see Figure 6.1. The alert is reported to the rest of the group, and UAV_1 continues its flight path without diverting to investigate further. The next UAV in sequence (UAV_2) designs a minimum-time flight path from its current location to r_f, and diverts from its original path to reach the threat as quickly as possible.[1]

[1]Here we assume that the threat is static, or slowly moving, relative to the intercept time for UAV_2. In an interesting variation, one might incorporate an estimate by UAV_1 of the threat's trajectory and modify the interception path accordingly.

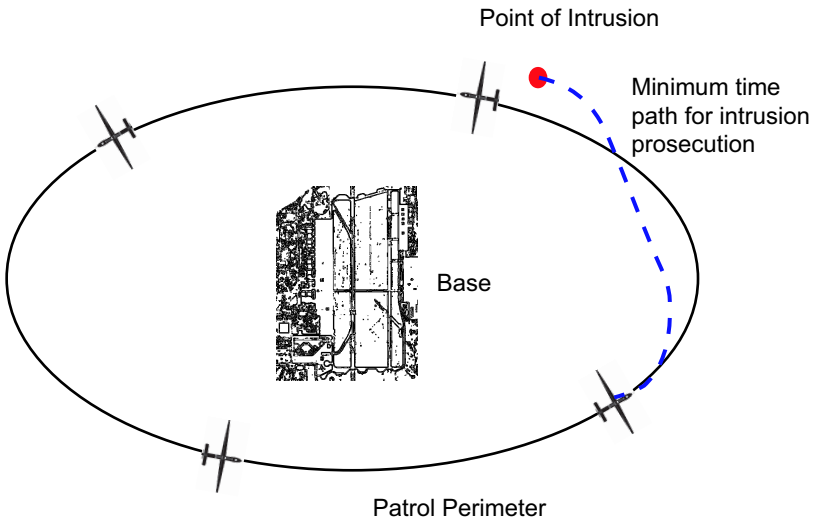


Figure 6.1 Conceptual sketch of the time-optimal alert response scenario.

To ensure that the diversion of UAV_2 does not disrupt the remaining formation (and its perimeter surveillance task), the remaining $N-1$ UAVs assume that UAV_2 is continuing to maintain synchrony, using a virtual-particle representation in the coordination algorithm. Once the threat is detected again (or the endpoint of the time-optimal trajectory is reached, whichever occurs first), UAV_2 returns to the original flight plan. If UAV_2 confirms the threat, the next UAV in sequence (UAV_3) is tasked to arrive at the intrusion point in minimum time, and this cycle repeats. A flow-diagram of this can be seen in Figure 6.2.

Space-time analysis of the base-defense algorithm illustrates how the gaps in perimeter coverage are minimized. Figure 6.3(a) illustrates the spatiotemporal coverage of a convex perimeter achieved by a single UAV in wind. The UAV trajectory is blue; its sensor swath is gray. Since the perimeter is a closed loop, we identify the left and right edges of the space-time plot. Figure 6.3(b) illustrates the spatiotemporal coverage of a convex perimeter achieved by three coordinated UAVs in wind. The coverage gaps are minimized by a time-splay formation. Figure 6.3(c) illustrates the minimum-time response of

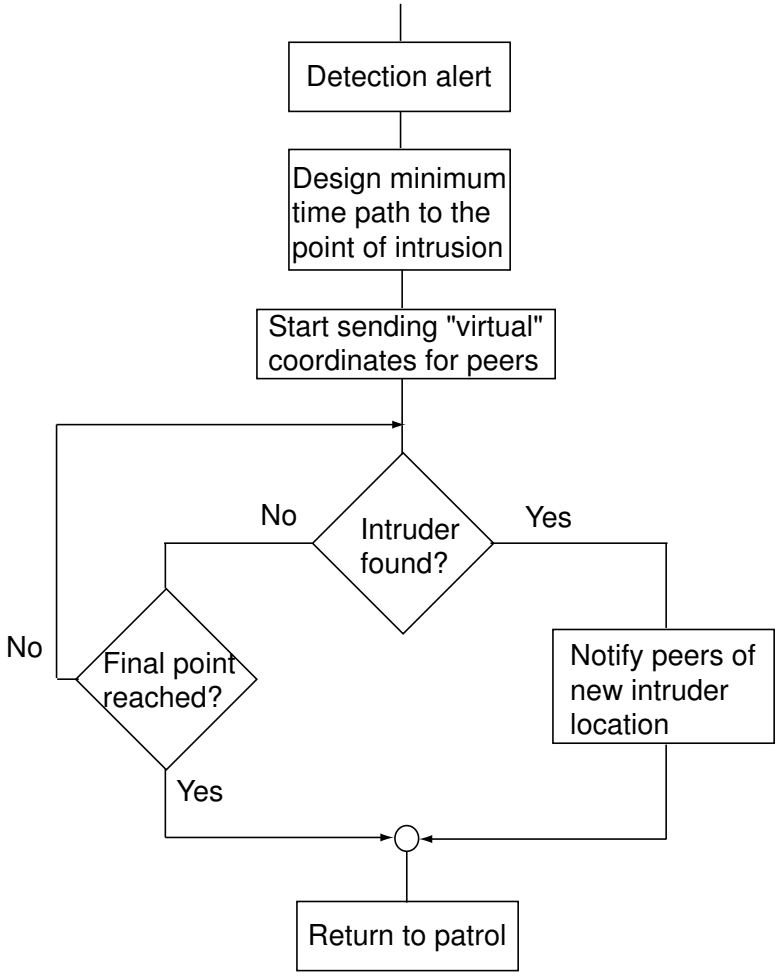
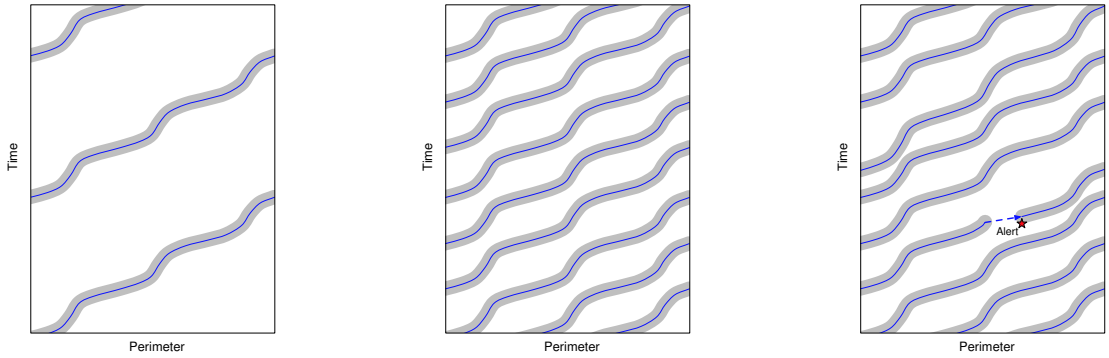


Figure 6.2 Flow diagram of the time-optimal alert response scenario.

a three-UAV patrol to a single intruder alert. The minimum-time shortcut appears as a hop rightward and upward on the space-time projection. Subsequent to prosecuting the alert, the UAVs return to a time-splay formation.

6.2 Simulation Results

The proposed framework has been tested in simulations. We selected the patrol perimeter to bound an existing establishment, Wright-Patterson Air Force Base (WPAFB). The base perimeter was defined by the convex hull of a finite set of waypoints that were selected



(a) Single UAV Patrol. (b) Time-Splay Patrol (3 UAVs). (c) Alert Response.

Figure 6.3 Space-time analysis of coordinated perimeter coverage with time-optimal alert response.

at landmarks around WPAFB. The base perimeter was then patrolled by $N = 4$ UAVs.

6.2.1 Ellipsoidal Base Perimeter

For this case the wind speed was chosen to be $V_w = 5$ m/s from the South; the airspeed of the UAVs was $V_a = 20$ m/s. The initial condition for the UAVs was chosen such that all UAVs were heading North along a straight line ($y_{E_k} = 0, \quad k = 1, ..., N$) when the simulation began. After all UAVs converged to the $(M, N) = (6, 6)$ time-splay formation, an intrusion alert was simulated outside of the perimeter (at location $[x_N, y_E]^T = [230 \text{ m}, -200 \text{ m}]^T$), see Figure 6.4(a). After the intrusion was detected, the next UAV in line left the original patrol perimeter to fly to the intrusion point in minimum time (which for the present simulation was $T = 16.9$ s), as illustrated in Figure 6.4(b). While the UAV was following the minimum-time path to the intrusion location, the virtual co-

ordinates were shared with the rest of the group to make sure the investigation does not brake the formation. After the prosecutor UAV reached the intruder location, it started the synchronization algorithm again (equation (4.20)) to converge back to the formation (Figure 6.4(c)).

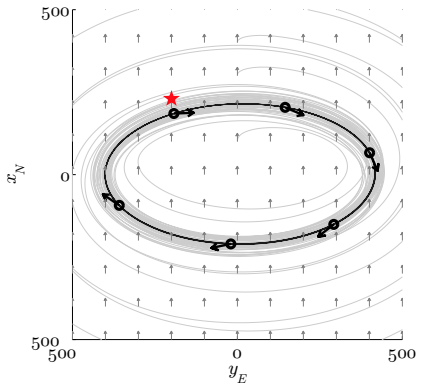
If the prosecutor UAV detected the intrusion again, it would report it to the rest of the group, and the next UAV in line could then begin to follow the minimum time path to the new intrusion location. If the intruder is not detected again (which was the case in the present simulation), the UAVs follow the synchronized perimeter patrol as they did before the detection event.

6.2.2 Wright-Patterson Air Force Base

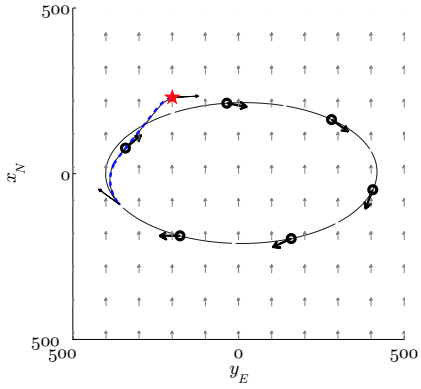
The wind speed was chosen to be $V_w = 10 \text{ m/s}$ from the South; the airspeed of the UAVs was $V_a = 20 \text{ m/s}$. Seven points were selected arbitrarily around the "perimeter" of WPAFB. The points were selected at main road intersections, so that the nominal UAV paths follow main roads around the base. Each point can be associated with a corresponding heading angle: the direction of a straight line leading to the next point. Time-optimal trajectories between the corner-points with the defined initial and final headings were designed using the algorithm described in Chapter 3. The resulting closed path is a convex curve in the plane with maximum rate turns at the curve corners.

The resulting straight segments were then approximated with a circle of radius $R_0 = 50 \text{ km}$ to ensure that the resulting closed curve is strictly convex. Figure 6.5(a) shows a satellite map of WPAFB with the seven selected GPS coordinates around the base, and Figure 6.5(b) shows the resulting closed, strictly convex curve in the plane.

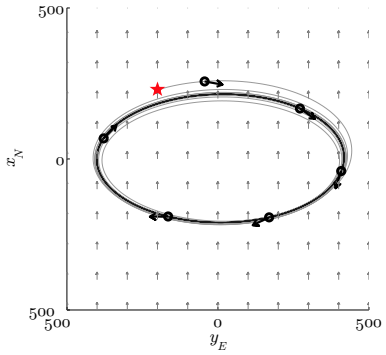
In this case the base perimeter was patrolled by $N = 4$ vehicles. Similarly to the previous case, an intrusion alert was simulated after all UAVs had converged to the desired time-phase arrangement, $(M, N) = (4, 4)$. The location of the intrusion point was



(a) Time-splay synchronization.



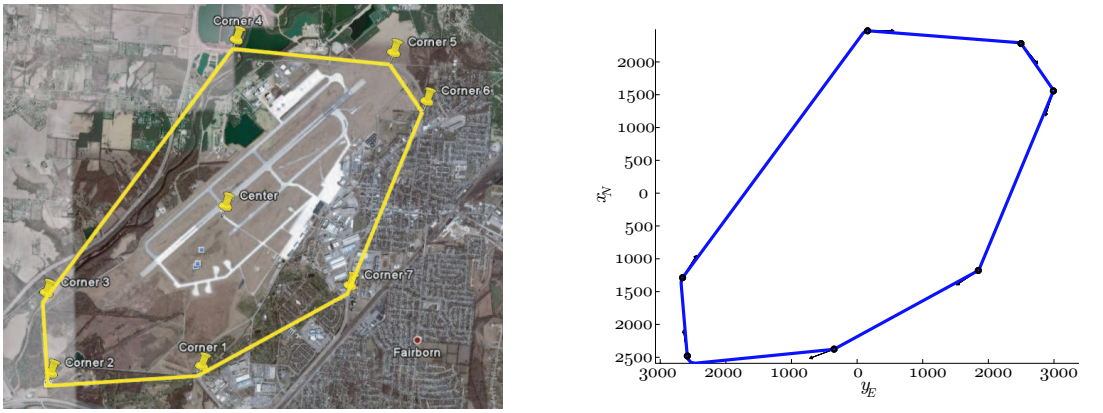
(b) Minimum time alert response.



(c) Returning to patrol.

Figure 6.4 Simulation of coordinated perimeter coverage with time-optimal alert response around a general ellipsoidal curve.

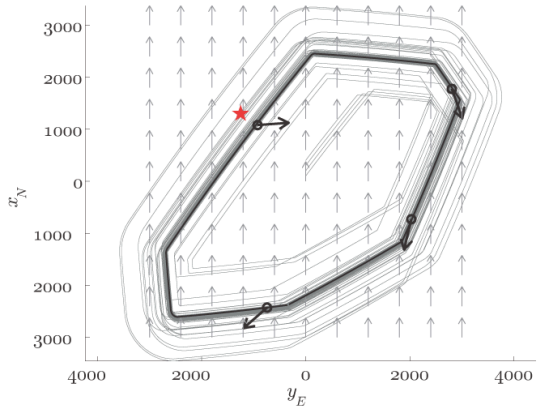
at $[x_N, y_E]^T = [1300\ \text{m}, -1250\ \text{m}]^T$), see Figure 6.6(a). After the intrusion was detected, the next UAV in line left the original patrol perimeter to fly to the intrusion point in minimum time (which for the present simulation was $T \approx 129\,\text{s}$), as illustrated in Figure 6.6(b). While the UAV was following the minimum-time path to the intrusion location, the virtual coordinates were shared with the rest of the group to make sure the investigation does not disrupt the formation. After the prosecutor UAV reached the intruder location, it started the coordination algorithm again (equation (4.20)) to converge back to the formation (Figure 6.6(c)).



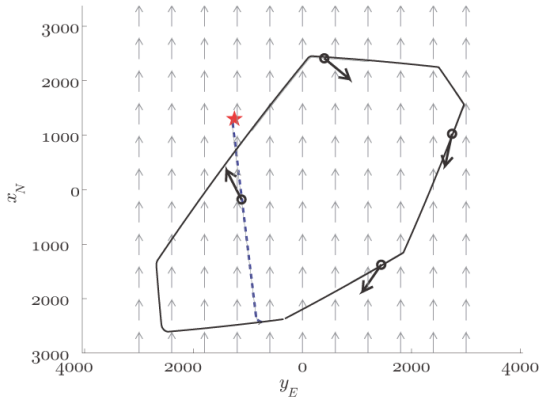
(a) Satellite map of WPAFB.

(b) Strictly convex curve around WPAFB.

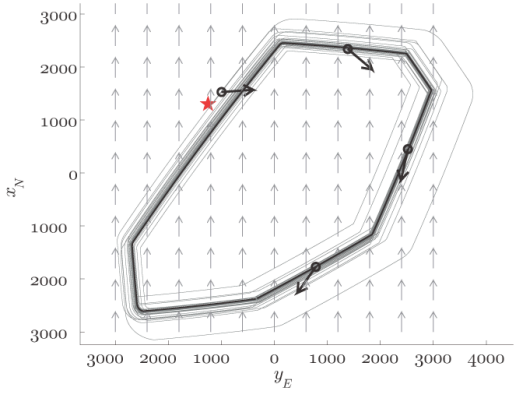
Figure 6.5 Seven points were selected around WPAFB to define a closed convex curve. The curve then serves as the perimeter to be patrolled by a team of UAVs. The straight segments are approximated by arcs of circles of radius $R_0 = 50$ km.



(a) Time-splay coordination.



(b) Minimum-time alert response.



(c) Returning to patrol.

Figure 6.6 Simulation of coordinated perimeter coverage with time-optimal alert response around WPAFB.

Chapter 7

Cross-track Control of an Autonomous Underwater Vehicle

Autonomous underwater vehicles (AUVs) are unmanned robots that are used for a variety of applications, including oceanographic measurements, underwater pipeline monitoring, surveillance and reconnaissance, and underwater mine detection to name a few. Temperature and salinity measurements collected using underwater gliders help better understand large scale motions that take place in the ocean [52]. AUVs can also be employed to detect and localize pollutant sources [22]. Although the properties and conditions of the medium in which underwater and aerial vehicles operate are very different, the physical laws that govern their motion are the same. Both can be categorized as flight vehicles, as the forces and moments acting on them stem primarily from — other than gravity — the fluid that surrounds them. For atmospheric flight vehicles, the aircraft has to generate enough lift using its wings to counter the effect of gravity; buoyant forces don't play a role. AUVs, on the other hand, due to the much larger density of the surrounding medium, experience significant buoyant force. In some applications it is desired to keep the buoyant force as close to the weight as possible to achieve neutral

buoyancy. [77].

In this chapter a neutrally buoyant, underactuated AUV is considered that is equipped with a propeller that provides thrust, and fins that control pitch and yaw moment. By assumption the vehicle is not actuated in the roll direction. The system is underactuated, since there are only three control channels for the six degrees of freedom. Although there is no control channel in the roll direction, we assume that the vehicle is designed such that the center of gravity is below the center of buoyancy. Such a design provides inherent system stability in the roll direction. Previous results on directional stabilization [80] are extended to ensure convergence to a straight line in inertial space [81]: the results of this chapter have been published in [81]. The stability of the closed-loop system is proved using Lyapunov's indirect method. The main contribution to the work described in [81] — and also presented here — was the stability analysis presented in Proposition 7.3.3, and the numerical analysis presented in Section 7.4.

7.1 Underwater Vehicle Equations of Motion

The AUV is modeled as a neutrally buoyant, rigid spheroid of mass m. The vehicle is equipped with a single propulsor, aligned with the axis of symmetry, and with moment actuators that provide independent control in pitch and yaw. (In Section 7.3, we will assume that the vehicle can also produce a roll control moment.)

Let the principal axes of the spheroid define a body-fixed reference frame represented by unit vectors $\boldsymbol{b}_1$, $\boldsymbol{b}_2$, and $\boldsymbol{b}_3$ where $\boldsymbol{b}_1$ is aligned with the axial symmetry axis. Assuming uniform fluid density, the origin of the body frame is the vehicle's center of buoyancy (CB), i.e., the center of mass of the displaced fluid. Let the unit vectors $\boldsymbol{i}_1$, $\boldsymbol{i}_2$, and $\boldsymbol{i}_3$ denote another reference frame, which is fixed in inertial space such that $\boldsymbol{i}_3$ is aligned with the force due to gravity. The location of the body frame with respect to the inertial

frame is given by the inertial vector $\boldsymbol{x}$. The orientation is given by the 3×3 proper rotation matrix $\boldsymbol{R}$, which transforms free vectors from the body frame to the inertial frame.

Let $\boldsymbol{v} = [u,\ v,\ w]^T$ represent the translational velocity and let $\boldsymbol{\omega} = [p,\ q,\ r]^T$ represent the rotational velocity of the AUV with respect to inertial space, where $\boldsymbol{v}$ and $\boldsymbol{\omega}$ are expressed in the body frame. The kinematic equations are

$$\dot{\boldsymbol{R}} = \boldsymbol{R}\hat{\boldsymbol{\omega}} \tag{7.1}$$

$$\dot{\boldsymbol{x}} = \boldsymbol{R}\boldsymbol{v} \tag{7.2}$$

where the character $\hat{\cdot}$ denotes the 3×3 skew-symmetric matrix satisfying $\hat{\boldsymbol{a}}\boldsymbol{b} = \boldsymbol{a} \times \boldsymbol{b}$ for vectors $\boldsymbol{a}$ and $\boldsymbol{b}$.

Let $\boldsymbol{h}$ represent the angular momentum of the body/fluid system about the CB and let $\boldsymbol{p}$ represent the linear momentum. Let $\boldsymbol{M}_{11}$ represent the sum of the rigid body inertia and added inertia matrices. Let the diagonal matrix $\boldsymbol{M}_{22} = \mathrm{diag}(m_1, m_2, m_3)$ represent the sum of the added mass matrix for the spheroid and $m\boldsymbol{I}$, where $\boldsymbol{I}$ is the 3×3 identity matrix. Assuming that the spheroid is prolate, $m_1 < m_2 = m_3$. Finally, let the center of mass (CM) be given by the body vector $\boldsymbol{r}_{\mathrm{cm}}$ and define $\boldsymbol{M}_{12} = m\hat{\boldsymbol{r}}_{\mathrm{cm}}$. Then

$$\begin{pmatrix} \boldsymbol{h} \\ \boldsymbol{p} \end{pmatrix} = \begin{pmatrix} \boldsymbol{M}_{11} & \boldsymbol{M}_{12} \\ \boldsymbol{M}_{12}^T & \boldsymbol{M}_{22} \end{pmatrix} \begin{pmatrix} \boldsymbol{\omega} \\ \boldsymbol{v} \end{pmatrix}. \tag{7.3}$$

The dynamic equations are

$$\dot{\boldsymbol{h}} = \boldsymbol{h} \times \boldsymbol{\omega} + \boldsymbol{p} \times \boldsymbol{v} + \boldsymbol{r}_{\mathrm{cm}} \times mg\left(\boldsymbol{R}^T\boldsymbol{i}_3\right) + \boldsymbol{\tau}_{\mathrm{v}} + \boldsymbol{\tau}_{\mathrm{c}} \tag{7.4}$$

$$\dot{\boldsymbol{p}} = \boldsymbol{p} \times \boldsymbol{\omega} + \boldsymbol{f}_{\mathrm{v}} + \boldsymbol{f}_{\mathrm{c}} \tag{7.5}$$

Note that, while there is no net gravitational force because the vehicle is neutrally buoyant, a gravitational moment appears on the right side of equation (7.4). This "restoring moment" tends to keep the body's CM below the CB. The terms $\boldsymbol{\tau}_{\mathrm{v}}$ and $\boldsymbol{f}_{\mathrm{v}}$ in equations (7.4) and (7.5) represent the viscous moment and force, respectively, while $\boldsymbol{\tau}_{\mathrm{c}}$ and $\boldsymbol{f}_{\mathrm{c}}$

represent the control moment and force. Because the vehicle's single thruster is aligned with the axis of symmetry, one may write $\boldsymbol{f}_{\mathrm{c}} = F_{\mathrm{c}}\ \boldsymbol{b}_1$, where F_{c} represents the (scalar) thrust.

The viscous moment and force are assumed to depend only on translational and angular velocity: $\boldsymbol{\tau}_{\mathrm{v}}(\boldsymbol{v}, \boldsymbol{\omega})$ and $\boldsymbol{f}_{\mathrm{v}}(\boldsymbol{v}, \boldsymbol{\omega})$. Thus, the viscous model assumes quasi-steady flow, as is standard for air and marine vehicle dynamic models. Even under the quasi-steady flow assumption, these terms are quite difficult to express analytically unless there are additional simplifying assumptions (such as small angles of attack and sideslip). Rather than restrict the model's validity by assuming explicit functional forms for $\boldsymbol{\tau}_{\mathrm{v}}$ and $\boldsymbol{f}_{\mathrm{v}}$, one may consider some general modeling assumptions.

Assumption 7.1.1.

$$\begin{aligned} \boldsymbol{\tau}_{\mathrm{v}}(\boldsymbol{v}, \boldsymbol{\omega}) \cdot \boldsymbol{\omega} < 0 \quad &\text{when} \qquad \boldsymbol{\omega} \neq \boldsymbol{0} \\ \boldsymbol{\tau}_{\mathrm{v}}(\boldsymbol{v}, \boldsymbol{\omega}) = \boldsymbol{0} \quad &\text{when} \quad (\boldsymbol{\omega}, \boldsymbol{v}) = (\boldsymbol{0}, u\boldsymbol{b}_1) \quad \forall \ u \in \mathbb{R} \end{aligned}$$

Assumption 7.1.1 states that the viscous moment opposes angular rate, in general, and that it vanishes for pure translation along the longitudinal axis. Assumption 7.1.1 defines a very general class of viscous moments, which includes, as a special case, standard models for angular rate damping.

Somewhat more structure is required for the viscous force model. By definition, drag opposes vehicle velocity, and lift acts orthogonally to the velocity vector. Considering axial symmetry, one may assume that the lift force acts in the plane that is determined by the velocity vector and the vehicle's longitudinal axis. This assumption fails to capture out-of-plane forces due to asymmetric fluid flow; however, it is consistent with standard modeling assumptions. The dependence of $\boldsymbol{f}_{\mathrm{v}}$ on $\boldsymbol{\omega}$ is also neglected, although this assumption can be relaxed to allow affine dependence on $\boldsymbol{\omega}$.

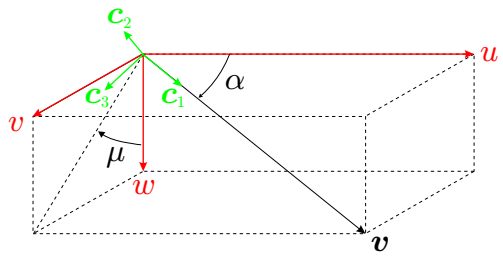


Figure 7.1 Hydrodynamic angles for an axisymmetric body.

To express $\boldsymbol{f}_{\mathrm{v}}(\boldsymbol{v})$, define two "hydrodynamic angles." First, let

$$\mu = \begin{cases} \arctan\left(\frac{v}{w}\right) & v \neq 0 \text{ and/or } w \neq 0 \\ 0 & v = w = 0 \end{cases} \tag{7.6}$$

where the 4-quadrant arctangent is used and $-\pi$ is identified with π. Rotating the body frame through the angle μ about the $\boldsymbol{b}_1$ axis defines an intermediate reference frame in which the velocity vector has components only in the intermediate frame's "1-3" plane. The 3-axis component of velocity in this intermediate frame is non-negative. Let

$$\alpha = \arctan\left(\frac{\sqrt{v^2 + w^2}}{u}\right) \tag{7.7}$$

where, once again, the 4-quadrant arctangent is used. Note that $\alpha \in [0, \pi]$. Rotating through the angle α about the intermediate 2-axis defines a new reference frame in which the 1-axis is aligned with the velocity vector, as shown in Figure 7.1. In the terminology of [23], we call this the "current" frame. Let $\boldsymbol{e}_i$ represent the i^{th} basis vector for $\mathbb{R}^3$; for example, $\boldsymbol{e}_1 = [1,\ 0,\ 0]^T$. The proper rotation matrix

$$\begin{aligned} \boldsymbol{R}_{\mathrm{BC}}(\mu, \alpha) &= e^{-\mu \widehat{\boldsymbol{e}_1}} e^{-\alpha \widehat{\boldsymbol{e}_2}} \\ &= \begin{pmatrix} \cos\alpha & 0 & -\sin\alpha \\ \sin\mu\sin\alpha & \cos\mu & \sin\mu\cos\alpha \\ \cos\mu\sin\alpha & -\sin\mu & \cos\mu\cos\alpha \end{pmatrix}, \end{aligned}$$

transforms free vectors from the current frame to the body frame. The hydrodynamic

angles α and μ are not the standard angle of attack and sideslip angle; however, the given definitions are more convenient for this work.

It is common practice to express lift and drag in terms of non-dimensional force coefficients $C_L(\alpha)$ and $C_D(\alpha)$. (The parametric dependence of C_L and C_D on Reynolds number is neglected here.) To non-dimensionalize the viscous force, define $F_0(\boldsymbol{v})$ as the product of dynamic pressure and a reference area S:

$$F_0(\boldsymbol{v}) = \frac{1}{2}\rho\|\boldsymbol{v}\|^2 S,$$

where ρ is the fluid density, which is assumed to be constant.

Assumption 7.1.2.

$$\boldsymbol{f}_{\mathrm{v}}(\boldsymbol{v}) = -F_0(\boldsymbol{v})\boldsymbol{R}_{\mathrm{BC}}(\mu,\alpha)\begin{pmatrix} C_D(\alpha) \\ 0 \\ C_L(\alpha) \end{pmatrix}.$$

where

- $C_D(\alpha)$ *is a smooth, even, positive function, and*
- $C_L(\alpha)$ *is a smooth, odd function which is positive (negative) when* $e^{i\alpha}$ *lies in the first (second) quadrant of the complex plane.*

For a prolate spheroid, the assumption that $C_D(\alpha)$ is even and positive is an empirical fact, and the properties assumed for $C_L(\alpha)$ are consistent with intuition.

Remark 7.1.3. *Assumption 7.1.2 is satisfied, for example, by*

$$\begin{aligned} C_D(\alpha) &= C_{D_0} + \frac{1}{2}C_{D_1}\left(1-\cos(2\alpha)\right) \\ C_L(\alpha) &= \frac{1}{2}C_{L_\alpha}\sin(2\alpha). \end{aligned}$$

The positive constants C_{L_α}, C_{D_0}, and C_{D_1} can be approximated using semi-empirical methods, as in [27] and [28]. Truncating cubic and higher order terms in the Taylor series expansions of these two expressions gives the standard "small angle" model for the drag and lift coefficient.

The viscous force model we use is quasi-steady, i.e., the force coefficients do not depend on the flow history. In reality, a spheroid moving at a large angle of attack is subject to complicated, unsteady forces that are not captured by simple models; see [76], for example. Such effects are typically ignored in control design with the expectation that well-designed model-based feedback will provide suitable system performance even when the model is imperfect.

Remark 7.1.4. *Note that α is discontinuous when $\boldsymbol{v} = \mathbf{0}$ and μ is discontinuous when $\alpha = 0$. These discontinuities are an artifact of the parametrization; the viscous force $\boldsymbol{f}_{\mathrm{v}}$ is smooth in the velocity $\boldsymbol{v}$.*

7.2 Directional Stabilization

In [38], Leonard introduced a potential energy shaping method for reduced dimensional (noncanonical) Hamiltonian control systems and applied the approach to the problem of stabilizing a fully actuated, bottom-heavy underwater vehicle. Although viscous effects were neglected in that analysis, a similar approach may be applied to an underactuated vehicle that is subject to viscous forces and moments. One first recognizes that the dynamics are invariant under translations of the inertial reference frame and rotations about the direction of gravity. One may then reduce the kinematic and dynamic equations to the

following form:

$$\dot{\boldsymbol{h}} = \boldsymbol{h} \times \boldsymbol{\omega} + \boldsymbol{p} \times \boldsymbol{v} + \boldsymbol{r}_{\mathrm{cm}} \times mg\boldsymbol{\zeta} + \boldsymbol{\tau}_{\mathrm{v}} + \boldsymbol{\tau}_{\mathrm{c}} \tag{7.8}$$

$$\dot{\boldsymbol{p}} = \boldsymbol{p} \times \boldsymbol{\omega} + \boldsymbol{f}_{\mathrm{v}} + \boldsymbol{f}_{\mathrm{c}} \tag{7.9}$$

$$\dot{\boldsymbol{\zeta}} = \boldsymbol{\zeta} \times \boldsymbol{\omega}. \tag{7.10}$$

In these equations,

$$\boldsymbol{\zeta} = \boldsymbol{R}^T \boldsymbol{i}_3$$

is the "tilt vector," that is, the unit vector in the direction of gravity, expressed in the body-fixed reference frame. The formal procedure by which the 12-dimensional dynamics are reduced to the system given above is discussed in [37] and [38]. In those papers, it is shown that, neglecting the control and viscous effects, the dynamic equations are Hamiltonian:

$$\frac{d}{dt}\begin{pmatrix} \boldsymbol{h} \\ \boldsymbol{p} \\ \boldsymbol{\zeta} \end{pmatrix} = \begin{pmatrix} \hat{\boldsymbol{h}} & \hat{\boldsymbol{p}} & \hat{\boldsymbol{\zeta}} \\ \hat{\boldsymbol{p}} & \boldsymbol{0} & \boldsymbol{0} \\ \hat{\boldsymbol{\zeta}} & \boldsymbol{0} & \boldsymbol{0} \end{pmatrix} \nabla H$$

where

$$H(\boldsymbol{h}, \boldsymbol{p}, \boldsymbol{\zeta}) = \frac{1}{2}\begin{pmatrix} \boldsymbol{h} \\ \boldsymbol{p} \end{pmatrix}^T \begin{pmatrix} \boldsymbol{M}_{11} & \boldsymbol{M}_{12} \\ \boldsymbol{M}_{12}^T & \boldsymbol{M}_{22} \end{pmatrix}^{-1} \begin{pmatrix} \boldsymbol{h} \\ \boldsymbol{p} \end{pmatrix} - mg\boldsymbol{r}_{\mathrm{cm}} \cdot \boldsymbol{\zeta}.$$

The gravitational torque $\boldsymbol{r}_{\mathrm{cm}} \times mg\boldsymbol{\zeta}$ in equation (7.8) derives, in this reduced Hamiltonian setting, from the scalar potential function $-mg\boldsymbol{r}_{\mathrm{cm}} \cdot \boldsymbol{\zeta}$ appearing in $H(\boldsymbol{h}, \boldsymbol{p}, \boldsymbol{\zeta})$. This potential function "breaks rotational symmetry" in the pitch and roll directions. The essential idea presented in [38] is to construct an artificial potential energy function in order to obtain a control moment $\boldsymbol{\tau}_{\mathrm{c}}$, which intentionally breaks the remaining rotational symmetry in a way that conserves a new, control-modified energy function. Using the modified energy and other conserved quantities, one then constructs a control Lyapunov function, which provides conditions for closed-loop stability of steady translation in a desired inertial direction.

The objective, in this section, is to stabilize the steady motion

$$\boldsymbol{R}_{\mathrm{e}} = \boldsymbol{R}_{\mathrm{d}}, \quad \boldsymbol{\omega}_{\mathrm{e}} = \boldsymbol{0}, \quad \boldsymbol{v}_{\mathrm{e}} = \boldsymbol{v}_{\mathrm{d}} = u_{\mathrm{d}}\, \boldsymbol{b}_1 \tag{7.11}$$

where $\boldsymbol{R}_{\mathrm{d}}$ is the rotation matrix representing a desired, constant attitude and $u_{\mathrm{d}} > 0$ is a desired, constant speed. Because of axial symmetry, one may assume without loss of generality that the CM rests in the body's longitudinal plane; that is, the body-fixed reference frame is oriented such that $\boldsymbol{r}_{\mathrm{cm}} \cdot \boldsymbol{b}_2 = 0$. To ensure that the body's weight does not induce a roll moment in equilibrium flight, one must also choose $\boldsymbol{R}_{\mathrm{d}}$ such that $\boldsymbol{\zeta}_{\mathrm{d}} \cdot \boldsymbol{b}_2 = 0$. Finally, one requires that the sign of $r_3 = \boldsymbol{r}_{\mathrm{cm}} \cdot \boldsymbol{b}_3$ match the sign of $\zeta_{\mathrm{d}_3} = \boldsymbol{\zeta}_{\mathrm{d}} \cdot \boldsymbol{b}_3$; for convenience, it is assumed that both values are positive. This last requirement ensures that the CM (or rather its projection onto an inertially vertical line through the CB) is below the CB in nominal flight.

Choosing a desired value for $\boldsymbol{\zeta}$ is equivalent to choosing the desired tilt (pitch and roll) angle for the vehicle. The vector $\boldsymbol{\zeta}$ is invariant under changes in heading. To stabilize motion along a particular heading, it is necessary to break the remaining rotational symmetry by introducing an additional direction vector. Following [38], let $\boldsymbol{\xi} = \boldsymbol{R}_{\mathrm{d}} \boldsymbol{b}_1$ represent the desired inertial direction of travel. (Assume that $\boldsymbol{\xi} \nparallel \boldsymbol{i}_3$, so that the nominal motion includes some horizontal component of translation.) Next, define

$$\boldsymbol{\lambda} = \boldsymbol{R}^T \boldsymbol{\xi}.$$

Since $\boldsymbol{\xi} \nparallel \boldsymbol{i}_3$, it follows that $\boldsymbol{\lambda} \nparallel \boldsymbol{\zeta}$ and the proper rotation matrix $\boldsymbol{R}$ can be unambiguously reconstructed using $\boldsymbol{\lambda}$ and $\boldsymbol{\zeta}$. Accordingly, one may replace the matrix differential equation (7.1) with the following rotational kinematic equations

$$\dot{\boldsymbol{\zeta}} = \boldsymbol{\zeta} \times \boldsymbol{\omega} \tag{7.12}$$

$$\dot{\boldsymbol{\lambda}} = \boldsymbol{\lambda} \times \boldsymbol{\omega}. \tag{7.13}$$

Note that equations (7.12) and (7.13) imply that the following quantities are conserved:

$$C_1 = \frac{1}{2}\boldsymbol{\zeta}\cdot\boldsymbol{\zeta}, \quad C_2 = \boldsymbol{\zeta}\cdot\boldsymbol{\lambda}, \quad \text{and} \quad C_3 = \frac{1}{2}\boldsymbol{\lambda}\cdot\boldsymbol{\lambda}.$$

The fact that C_1 and C_3 are constant reflects the fact that $\boldsymbol{\zeta}$ and $\boldsymbol{\lambda}$ are unit vectors by definition. The constant C_2 measures the degree to which the desired direction of travel corresponds to motion in the vertical direction. If $C_2 = 0$, for example, the desired motion is purely horizontal. Since we have assumed that $\boldsymbol{\lambda} \not\parallel \boldsymbol{\zeta}$, it follows that $|C_2| < 1$.

In terms of these variables, the steady motion (7.11) is

$$\begin{aligned} \boldsymbol{\zeta}_{\mathrm{e}} &= \boldsymbol{\zeta}_{\mathrm{d}}, & \boldsymbol{\lambda}_{\mathrm{e}} &= \boldsymbol{\lambda}_{\mathrm{d}} = \boldsymbol{b}_1, \\ \boldsymbol{h}_{\mathrm{e}} &= \boldsymbol{M}_{12}\boldsymbol{v}_{\mathrm{d}}, & \boldsymbol{p}_{\mathrm{e}} &= \boldsymbol{M}_{22}\boldsymbol{v}_{\mathrm{d}}. \end{aligned} \tag{7.14}$$

The feedback control law given in the following proposition is based on results presented in [32] and [38]. Also see [80].

Proposition 7.2.1. *[80] Consider the feedback control law*

$$F_{\mathrm{c}} = F_0(\boldsymbol{v})\left(\cos\alpha C_D(\alpha) - \sin\alpha C_L(\alpha)\right) - k_u(u - u_{\mathrm{d}}) \tag{7.15}$$

and

$$\boldsymbol{\tau}_{\mathrm{c}} = mg\boldsymbol{\zeta}\times\boldsymbol{r}_{\mathrm{cm}} - \frac{mgr_3}{\zeta_{\mathrm{d}_3}}\boldsymbol{\zeta}\times\boldsymbol{\zeta}_{\mathrm{d}} + \boldsymbol{\lambda}\times\left(\boldsymbol{M}_{22} - \tilde{m}\boldsymbol{I}\right)\boldsymbol{v}_{\mathrm{d}}\|\boldsymbol{v}_{\mathrm{d}}\| - \boldsymbol{p}\times\boldsymbol{v}_{\mathrm{d}} - k_\omega\boldsymbol{I}_{23}\boldsymbol{\omega} \tag{7.16}$$

where $\boldsymbol{I}_{23} = \boldsymbol{I} - \boldsymbol{e}_1\boldsymbol{e}_1^T$. The control law asymptotically stabilizes the equilibrium (7.14) if $k_u > 0$, $\tilde{m} > m_1$, and $k_\omega \geq 0$ and provided

$$\frac{mgr_3}{\zeta_{\mathrm{d}_3}} \neq (m_1 - \tilde{m})u_{\mathrm{d}}^2. \tag{7.17}$$

Remark 7.2.2. *The terms in (7.16) involving $\boldsymbol{\zeta}$ and $\boldsymbol{\lambda}$ derive from the scalar potential energy function*

$$mg\boldsymbol{r}_{\mathrm{cm}}\cdot\boldsymbol{\zeta} - \frac{mgr_3}{\zeta_{\mathrm{d}_3}}\boldsymbol{\zeta}_{\mathrm{d}}\cdot\boldsymbol{\zeta} + \|\boldsymbol{v}_{\mathrm{d}}\|\left[\left(\boldsymbol{M}_{22} - \tilde{m}\boldsymbol{I}\right)\boldsymbol{v}_{\mathrm{d}}\right]\cdot\boldsymbol{\lambda}.$$

The first term above cancels the true gravitational potential in the total system energy, and the remaining terms define a new, artificial potential energy that helps to stabilize the desired steady motion. The coefficient of $\boldsymbol{\zeta}_{\mathrm{d}} \cdot \boldsymbol{\zeta}$ in this artificial potential function has been chosen to eliminate the roll component from the control moment. Roll control is unnecessary here, since the roll-stabilizing effect of gravity is preserved in the closed-loop system. Note that the last three terms in (7.16) are orthogonal to $\boldsymbol{b}_1$. Considering the first pair of terms, recall that $\boldsymbol{r}_{\mathrm{cm}} \cdot \boldsymbol{b}_2 = 0$ and $\boldsymbol{\zeta}_{\mathrm{d}} \cdot \boldsymbol{b}_2 = 0$. The observation that $\boldsymbol{\tau}_{\mathrm{c}} \cdot \boldsymbol{b}_1 = 0$ then follows from direct computation.

Proof of Proposition 7.2.1: The proof makes use of the energy-Casimir stability analysis method for reduced Hamiltonian systems; see [40] for a description of the method and [37] for an application to underwater vehicles. Although the system considered here is not Hamiltonian, due to the viscous terms, the energy-Casimir technique can still be used to construct a candidate Lyapunov function. To begin, recognize that any smooth function $\Phi(C_1, C_2, C_3)$ is conserved because the arguments are constant. One may define a candidate Lyapunov function H_Φ by summing a control-modified system energy with the smooth function $\Phi(C_1, C_2, C_3)$:

$$\begin{aligned} H_\Phi = & \frac{1}{2} \begin{pmatrix} \boldsymbol{h} - \boldsymbol{h}_{\mathrm{e}} \\ \boldsymbol{p} - \boldsymbol{p}_{\mathrm{e}} \end{pmatrix}^T \begin{pmatrix} \boldsymbol{M}_{11} & \boldsymbol{M}_{12} \\ \boldsymbol{M}_{12}^T & \boldsymbol{M}_{22} \end{pmatrix}^{-1} \begin{pmatrix} \boldsymbol{h} - \boldsymbol{h}_{\mathrm{e}} \\ \boldsymbol{p} - \boldsymbol{p}_{\mathrm{e}} \end{pmatrix} \\ & - \frac{mgr_3}{\zeta_{\mathrm{d}_3}} \boldsymbol{\zeta}_{\mathrm{d}} \cdot \boldsymbol{\zeta} + \|\boldsymbol{v}_{\mathrm{d}}\| \left[\left(\boldsymbol{M}_{22} - \tilde{m} \boldsymbol{I} \right) \boldsymbol{v}_{\mathrm{d}} \right] \cdot \boldsymbol{\lambda} + \Phi(C_1, C_2, C_3). \end{aligned} \tag{7.18}$$

Showing stability requires showing that the desired equilibrium (7.14) is a strict minimum of H_Φ and that $\dot{H}_\Phi \leq 0$. The former requirement imposes conditions on the equilibrium value of the first and second variations of H_Φ. These conditions are satisfied by choosing $\tilde{m} > m_1$ and by choosing

$$\Phi(C_1, C_2, C_3) = \left(\frac{mgr_3}{\zeta_{\mathrm{d}_3}} \right) C_1 + (\tilde{m} - m_1) \, u_{\mathrm{d}}^2 C_3.$$

Under the conditions given in the proposition, the equilibrium (7.14) is a strict mini-

mum of H_Φ. Next, one computes

$$\dot{H}_\Phi = -k_\omega \boldsymbol{\omega}^T \boldsymbol{I}_{23} \boldsymbol{\omega} + \boldsymbol{\omega} \cdot \boldsymbol{\tau}_\mathrm{v} + (\boldsymbol{v} - \boldsymbol{v}_\mathrm{d}) \cdot (\boldsymbol{f}_\mathrm{c} + \boldsymbol{f}_\mathrm{v}) .$$

The first two terms are non-positive. The third term is

$$\begin{aligned}(\boldsymbol{v} - \boldsymbol{v}_\mathrm{d}) \cdot (\boldsymbol{f}_\mathrm{c} + \boldsymbol{f}_\mathrm{v}) = &- (v \sin\mu + w \cos\mu) (\sin\alpha C_D(\alpha) + \cos\alpha C_L(\alpha)) F_0(\boldsymbol{v}) \\ &+ (u - u_\mathrm{d}) (F_\mathrm{c} - F_0(\boldsymbol{v}) (\cos\alpha C_D(\alpha) - \sin\alpha C_L(\alpha))) . \end{aligned} \tag{7.19}$$

Consider the first term in (7.19). By the definition (7.6) of μ,

$$v \sin\mu + w \cos\mu = \sqrt{v^2 + w^2} \geq 0.$$

Also, given the assumptions on the form of C_D and C_L and the fact that $\alpha \in [0, \pi]$,

$$\sin\alpha C_D(\alpha) + \cos\alpha C_L(\alpha) \geq 0$$

and strictly positive for $\alpha \in (0, \pi)$. Therefore, the first term in (7.19) is non-positive. Defining F_c according to (7.15), one finds that

$$\begin{aligned}\dot{H}_\Phi = &-k_\omega \boldsymbol{\omega}^T \boldsymbol{I}_{23} \boldsymbol{\omega} + \boldsymbol{\omega} \cdot \boldsymbol{\tau}_\mathrm{v} - k_u (u - u_\mathrm{d})^2 \\ &- \sqrt{v^2 + w^2} (\sin\alpha C_D(\alpha) + \cos\alpha C_L(\alpha)) F_0(\boldsymbol{v}) \\ \leq\ & 0. \end{aligned}$$

Having shown that the equilibrium (7.14) is stable, using H_Φ as a Lyapunov function, it remains to show that the equilibrium is asymptotically stable. Because the equilibrium is a strict minimum of H_Φ, level sets of H_Φ in a neighborhood of the equilibrium define compact, positively invariant sets. Define such a set Ω. According to Lasalle's invariance principle [31], trajectories which begin in Ω converge to the largest invariant set M contained in the set $E = \left\{ (\boldsymbol{h}, \boldsymbol{p}, \boldsymbol{\zeta}, \boldsymbol{\lambda}) \subset \Omega \mid \dot{H}_\Phi = 0 \right\}$. It is easy to see that $\dot{H}_\Phi = 0$ if and only if

$$\boldsymbol{v} = \boldsymbol{v}_\mathrm{d} \quad \text{and} \quad \boldsymbol{\omega} = \boldsymbol{0}.$$

To determine the possible values of $\boldsymbol{\zeta}$ and $\boldsymbol{\lambda}$ within M, one must examine the dynamic equations within the set E. From equation (7.4), one finds that

$$\mathbf{0} = -\frac{mgr_3}{\zeta_{\mathrm{d}_3}}\boldsymbol{\zeta} \times \boldsymbol{\zeta}_{\mathrm{d}} + \|\boldsymbol{v}_{\mathrm{d}}\|(m_1 - \tilde{m})\boldsymbol{\lambda} \times \boldsymbol{v}_{\mathrm{d}} \tag{7.20}$$

Further analysis shows that, provided (7.17) holds, the only values of $\boldsymbol{\zeta}$ and $\boldsymbol{\lambda}$ which satisfy (7.20) are those shown in Table 7.1, where

$$\sigma = \frac{mgr_3 - (m_1 - \tilde{m})u_{\mathrm{d}}^2\zeta_{\mathrm{d}_3}}{mgr_3 + (m_1 - \tilde{m})u_{\mathrm{d}}^2\zeta_{\mathrm{d}_3}}.$$

Table 7.1 Values of $\boldsymbol{\zeta}$ and $\boldsymbol{\lambda}$ which satisfy equation (7.20).

$\boldsymbol{\zeta}$	$\boldsymbol{\lambda}$
$\boldsymbol{\zeta}_{\mathrm{d}}$	$\boldsymbol{\lambda}_{\mathrm{d}}$
$-\boldsymbol{\zeta}_{\mathrm{d}}$	$-\boldsymbol{\lambda}_{\mathrm{d}}$
$\pm\frac{1}{\sqrt{1-(1-\sigma^2)C_2^2}}\begin{pmatrix}\sigma C_2\\ 0\\ \sqrt{1-C_2^2}\end{pmatrix}$	$C_2\boldsymbol{\zeta} + \sqrt{1-C_2^2}\,(\boldsymbol{\zeta} \times \boldsymbol{e}_2)$

Each pair $(\boldsymbol{\zeta}, \boldsymbol{\lambda})$ in Table 7.1 corresponds to an isolated equilibrium. Since the desired equilibrium is a strict minimum of H_Φ, one may choose Ω small enough to exclude these other equilibria. It follows from LaSalle's invariance principle that the desired equilibrium is asymptotically stable. □

Remark 7.2.3. *To better understand the other isolated equilibria, consider the special case where the desired equilibrium is steady translation in the horizontal plane ($C_2 = 0$). In this case, the*

four isolated equilibria represent (i) steady motion in the desired direction with the CM below the CB, (ii) steady motion in the desired direction with the CM above *the CB, (iii) steady motion* opposite *the desired direction with the CM below the CB, and (iv) steady motion opposite the desired direction with the CM above the CB. Spectral analysis shows that all save the first of these are unstable equilibria.*

7.3 Cross-Track Control

Having obtained a feedback control law that asymptotically stabilizes motion in a given (non-vertical) direction, it is natural to seek an extension that will asymptotically stabilize motion along a desired path. To do so, one must vary the desired orientation $\boldsymbol{R}_{\mathrm{d}}$ in a way that yields convergence to path. We consider only paths which are straight lines, and we employ a standard guidance scheme: line-of-sight feedback. Applications involving line-of-sight guidance for six-degree-of-freedom underactuated underwater vehicles are discussed in [21], [19], and [13], for example.

Since we are considering only linear paths, we assume without further loss of generality that the desired path coincides with the $\boldsymbol{i}_1$ axis. Let $\boldsymbol{\mu}_1$ be the vector from the body frame origin to a point which is some "look-ahead distance" $k_x L$ further along the desired path:

$$\boldsymbol{\mu}_1 = (k_x L)\boldsymbol{i}_1 - \boldsymbol{I}_{23}\boldsymbol{x}.$$

The length L is a characteristic length, such as the vehicle length, and k_x is a dimensionless control parameter. The vector $\boldsymbol{\mu}_1$ provides coordinates for the lateral translational kinematics:

$$\dot{\boldsymbol{\mu}}_1 = -\boldsymbol{I}_{23}\boldsymbol{R}\boldsymbol{v}. \tag{7.21}$$

(Note that the along-track position is ignored.)

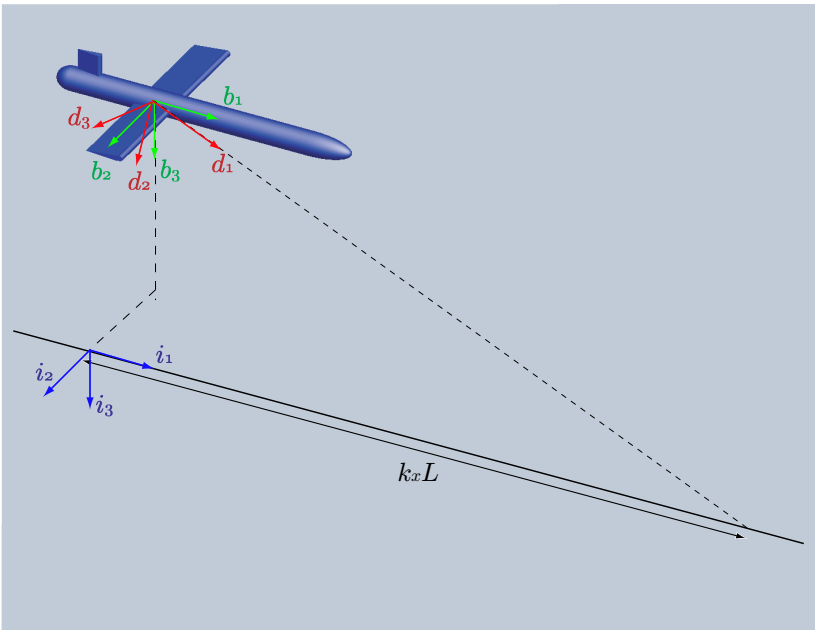


Figure 7.2 Line of sight guidance.

The desired orientation $\boldsymbol{R}_{\mathrm{d}}$ is defined by an orthonormal triad $\boldsymbol{d}_1$, $\boldsymbol{d}_2$, and $\boldsymbol{d}_3$, which can be constructed from $\boldsymbol{\mu}_1$. To this end, define the orthogonal triad $\boldsymbol{\mu}_1$, $\boldsymbol{\mu}_2$, and $\boldsymbol{\mu}_3$, where

$$\boldsymbol{\mu}_2 = \boldsymbol{i}_3 \times \boldsymbol{\mu}_1 \qquad \text{and} \qquad \boldsymbol{\mu}_3 = \boldsymbol{\mu}_1 \times \boldsymbol{\mu}_2.$$

Now let $\boldsymbol{d}_i = \boldsymbol{\mu}_i / \|\boldsymbol{\mu}_i\|$ for $i \in \{1, 2, 3\}$. The desired attitude is

$$\boldsymbol{R}_{\mathrm{d}}(\boldsymbol{x}) = [\boldsymbol{d}_1, \ \boldsymbol{d}_2, \ \boldsymbol{d}_3]\,. \tag{7.22}$$

The desired attitude is constructed such that the desired roll angle is zero to take advantage of the natural "metacentric" roll stability that arises from having the CM below the CB.

It is convenient to re-express the cross-track kinematics (7.21) in the body reference frame, so we define

$$\boldsymbol{m}_1 = \boldsymbol{R}^T \boldsymbol{\mu}_1$$

and compute

$$\begin{aligned}\dot{\boldsymbol{m}}_1 &= \boldsymbol{R}^T\dot{\boldsymbol{\mu}}_1 + \dot{\boldsymbol{R}}^T\boldsymbol{\mu}_1 \\ &= -\boldsymbol{R}^T\boldsymbol{I}_{23}\boldsymbol{R}\boldsymbol{v} + (\boldsymbol{R}\boldsymbol{\omega})^T\boldsymbol{\mu}_1 \\ &= \boldsymbol{m}_1 \times \boldsymbol{\omega} - \boldsymbol{B}\boldsymbol{v} \end{aligned} \tag{7.23}$$

where

$$\boldsymbol{B} = \boldsymbol{R}^T\boldsymbol{I}_{23}\boldsymbol{R}.$$

Note that $\boldsymbol{\lambda} = \boldsymbol{m}_1/\|\boldsymbol{m}_1\|$, so $\boldsymbol{m}_1$ encodes information about the attitude error, as determined by the cross-track error.

Now recall from Section 7.2 that, as long as $\boldsymbol{\lambda} \nparallel \boldsymbol{\zeta}$, one may unambiguously reconstruct $\boldsymbol{R}$ from $\boldsymbol{\lambda}$ and $\boldsymbol{\zeta}$. Note in the definition of $\boldsymbol{\mu}_1$, that $\boldsymbol{m}_1 \nparallel \boldsymbol{\zeta}$ provided $k_x \neq 0$. The information contained in $\boldsymbol{\zeta}$ and $\boldsymbol{m}_1$, however, is not enough to reconstruct $\boldsymbol{R}$, which is necessary to compute $\boldsymbol{B}$ in equation (7.23). To have a closed set of dynamic equations, we incorporate an additional unit vector: $\boldsymbol{\Upsilon} = \boldsymbol{R}^T\boldsymbol{i}_2$. With this definition,

$$\boldsymbol{R} = [\boldsymbol{\Upsilon} \times \boldsymbol{\zeta},\ \boldsymbol{\Upsilon},\ \boldsymbol{\zeta}]^T.$$

The complete cross-track control system dynamics are

$$\dot{\boldsymbol{h}} = \boldsymbol{h} \times \boldsymbol{\omega} + \boldsymbol{p} \times \boldsymbol{v} + \boldsymbol{r}_{\mathrm{cm}} \times mg\boldsymbol{\zeta} + \boldsymbol{\tau}_{\mathrm{v}} + \boldsymbol{\tau}_{\mathrm{c}} \tag{7.24}$$

$$\dot{\boldsymbol{p}} = \boldsymbol{p} \times \boldsymbol{\omega} + \boldsymbol{f}_{\mathrm{v}} + \boldsymbol{f}_{\mathrm{c}} \tag{7.25}$$

$$\dot{\boldsymbol{m}}_1 = \boldsymbol{m}_1 \times \boldsymbol{\omega} - \boldsymbol{B}\boldsymbol{v} \tag{7.26}$$

$$\dot{\boldsymbol{\Upsilon}} = \boldsymbol{\Upsilon} \times \boldsymbol{\omega} \tag{7.27}$$

$$\dot{\boldsymbol{\zeta}} = \boldsymbol{\zeta} \times \boldsymbol{\omega} \tag{7.28}$$

The cross-track error may be obtained from the system state as follows:

$$\boldsymbol{m}_1 \cdot \boldsymbol{\Upsilon} = -y, \qquad \boldsymbol{m}_1 \cdot \boldsymbol{\zeta} = -z.$$

Intuitively, cross-track control may be accomplished by using the directional stabilization algorithm of Section 7.2, but with the desired attitude updated continuously according to the cross-track error. We therefore define the moment control law

$$\boldsymbol{\tau}_{\mathrm{c}} = mg\boldsymbol{\zeta} \times \boldsymbol{r}_{\mathrm{cm}} - mg\beta\boldsymbol{\zeta} \times \boldsymbol{\zeta}_{\mathrm{d}} + \boldsymbol{m}_1 \times (\boldsymbol{M}_{22} - \tilde{m}\boldsymbol{I})\, \boldsymbol{v}_{\mathrm{d}} \left(\frac{\|\boldsymbol{v}_{\mathrm{d}}\|}{L} \right) - \boldsymbol{p} \times \boldsymbol{v}_{\mathrm{d}} \tag{7.29}$$

and define F_c as in (7.15). The moment control law (7.29) compares closely with (7.16); however, it is no longer true that $\boldsymbol{\tau}_{\mathrm{c}} \cdot \boldsymbol{b}_1 = 0$ in general. We must therefore assume that the vehicle has roll control authority.

Proposition 7.3.1. *[81] The dynamic equations (7.24) through (7.28), with the control moment (7.29), take the form*

$$\frac{d}{dt}\begin{pmatrix} \boldsymbol{h} \\ \boldsymbol{p} \\ \boldsymbol{m}_1 \\ \boldsymbol{\Upsilon} \\ \boldsymbol{\zeta} \end{pmatrix} = \begin{pmatrix} \hat{\boldsymbol{h}} & \hat{\boldsymbol{p}} & \hat{\boldsymbol{m}}_1 & \hat{\boldsymbol{\Upsilon}} & \hat{\boldsymbol{\zeta}} \\ \hat{\boldsymbol{p}} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \hat{\boldsymbol{m}}_1 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \hat{\boldsymbol{\Upsilon}} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \hat{\boldsymbol{\zeta}} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{pmatrix} \nabla H_{\mathrm{c}} + \begin{pmatrix} \boldsymbol{\tau}_{\mathrm{v}} \\ \boldsymbol{f}_{\mathrm{v}} + \boldsymbol{f}_{\mathrm{c}} \\ -\boldsymbol{B}\boldsymbol{v} \\ \mathbf{0} \\ \mathbf{0} \end{pmatrix} \tag{7.30}$$

where

$$H_{\mathrm{c}} = \frac{1}{2} \begin{pmatrix} \boldsymbol{h} - \boldsymbol{h}_{\mathrm{e}} \\ \boldsymbol{p} - \boldsymbol{p}_{\mathrm{e}} \end{pmatrix}^T \begin{pmatrix} \boldsymbol{M}_{11} & \boldsymbol{M}_{12} \\ \boldsymbol{M}_{12}^T & \boldsymbol{M}_{22} \end{pmatrix}^{-1} \begin{pmatrix} \boldsymbol{h} - \boldsymbol{h}_{\mathrm{e}} \\ \boldsymbol{p} - \boldsymbol{p}_{\mathrm{e}} \end{pmatrix} - mg\beta\boldsymbol{\zeta}_{\mathrm{d}} \cdot \boldsymbol{\zeta} + \frac{\|\boldsymbol{v}_{\mathrm{d}}\|}{L} \left[(\boldsymbol{M}_{22} - \tilde{m}\boldsymbol{I})\, \boldsymbol{v}_{\mathrm{d}} \right] \cdot \boldsymbol{m}_1 .$$

Proof: The proof reduces to determining ∇H_{c} and verifying that the equations match. It is obvious that

$$\frac{\partial H_{\mathrm{c}}}{\partial \boldsymbol{h}} = \boldsymbol{\omega} \quad \text{and} \quad \frac{\partial H_{\mathrm{c}}}{\partial \boldsymbol{p}} = \boldsymbol{v} - \boldsymbol{v}_{\mathrm{d}},$$

so it remains to compute

$$\frac{\partial H_{\mathrm{c}}}{\partial \boldsymbol{m}_1}, \quad \frac{\partial H_{\mathrm{c}}}{\partial \boldsymbol{\Upsilon}}, \quad \text{and} \quad \frac{\partial H_{\mathrm{c}}}{\partial \boldsymbol{\zeta}}.$$

The chief complication is that $\boldsymbol{\zeta}_\mathrm{d}$ depends on $\boldsymbol{R}_\mathrm{d}$, which now varies with $\boldsymbol{\zeta}$ and $\boldsymbol{m}_1$:

$$\boldsymbol{\zeta}_\mathrm{d} = \begin{pmatrix} \zeta_{\mathrm{d}_1} \\ 0 \\ \zeta_{\mathrm{d}_3} \end{pmatrix} = \begin{pmatrix} \boldsymbol{\zeta} \cdot \left(\frac{\boldsymbol{m}_1}{\|\boldsymbol{m}_1\|} \right) \\ 0 \\ \sqrt{1 - \left(\boldsymbol{\zeta} \cdot \left(\frac{\boldsymbol{m}_1}{\|\boldsymbol{m}_1\|} \right) \right)^2} \end{pmatrix}. \tag{7.31}$$

In Section 7.2, $C_2 = \boldsymbol{\zeta} \cdot \boldsymbol{\lambda}$ was conserved so $\boldsymbol{\zeta}_\mathrm{d}$ remained constant. Here, however, $\boldsymbol{\zeta}_\mathrm{d}$ necessarily varies with the cross-track error. Referring to equation (7.31), note that

$$\boldsymbol{\zeta}_\mathrm{d} = \begin{pmatrix} \sin\theta_\mathrm{d} \\ 0 \\ \cos\theta_\mathrm{d} \end{pmatrix}$$

where $\theta_\mathrm{d} = \sin^{-1}\left(\boldsymbol{\zeta} \cdot (\boldsymbol{m}_1/\|\boldsymbol{m}_1\|)\right)$. Thus, at every instant, the desired roll angle is zero, and the desired pitch angle θ_d is determined by the tilt attitude $\boldsymbol{\zeta}$ and the line of sight vector $\boldsymbol{m}_1$.

It can easily be checked that

$$\frac{\partial}{\partial \boldsymbol{\zeta}} \left(\boldsymbol{\zeta}_\mathrm{d} \cdot \boldsymbol{\zeta} \right) = \boldsymbol{\zeta}_\mathrm{d} - \underbrace{\frac{1}{\zeta_{\mathrm{d}_3}} \left(\boldsymbol{e}_2 \cdot (\boldsymbol{\zeta} \times \boldsymbol{\zeta}_\mathrm{d}) \right)}_{\kappa} \left(\frac{\boldsymbol{m}_1}{\|\boldsymbol{m}_1\|} \right).$$

Also,

$$\frac{\partial}{\partial \boldsymbol{m}_1} \left(\boldsymbol{\zeta}_\mathrm{d} \cdot \boldsymbol{\zeta} \right) = \frac{1}{\|\boldsymbol{m}_1\|} \left[-\kappa \boldsymbol{\zeta} + \left(\frac{\zeta_3}{\zeta_{\mathrm{d}_3}} - \left(\boldsymbol{\zeta}_\mathrm{d} \cdot \boldsymbol{\zeta} \right) \right) \frac{\boldsymbol{m}_1}{\|\boldsymbol{m}_1\|} \right].$$

Noting that

$$\begin{pmatrix} \hat{\boldsymbol{h}} & \hat{\boldsymbol{p}} & \hat{\boldsymbol{m}}_1 & \hat{\boldsymbol{\Upsilon}} & \hat{\boldsymbol{\zeta}} \\ \hat{\boldsymbol{p}} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \hat{\boldsymbol{m}}_1 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \hat{\boldsymbol{\Upsilon}} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \hat{\boldsymbol{\zeta}} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{pmatrix} \begin{pmatrix} \boldsymbol{\omega} \\ \boldsymbol{v} - \boldsymbol{v}_\mathrm{d} \\ -\frac{mg\beta}{\|\boldsymbol{m}_1\|}\left[-\kappa\boldsymbol{\zeta} + \left(\frac{\zeta_3}{\zeta_{\mathrm{d}_3}} - (\boldsymbol{\zeta}_\mathrm{d}\cdot\boldsymbol{\zeta})\right)\frac{\boldsymbol{m}_1}{\|\boldsymbol{m}_1\|}\right] + \frac{\|\boldsymbol{v}_\mathrm{d}\|}{L}\left(\boldsymbol{M}_{22} - \tilde{m}\boldsymbol{I}\right)\boldsymbol{v}_\mathrm{d} \\ \mathbf{0} \\ -mg\beta\left(\boldsymbol{\zeta}_\mathrm{d} - \kappa\,\frac{\boldsymbol{m}_1}{\|\boldsymbol{m}_1\|}\right) \end{pmatrix}$$

$$= \begin{pmatrix} \hat{\boldsymbol{h}} & \hat{\boldsymbol{p}} & \hat{\boldsymbol{m}}_1 & \hat{\boldsymbol{\Upsilon}} & \hat{\boldsymbol{\zeta}} \\ \hat{\boldsymbol{p}} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \hat{\boldsymbol{m}}_1 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \hat{\boldsymbol{\Upsilon}} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \hat{\boldsymbol{\zeta}} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{pmatrix} \begin{pmatrix} \boldsymbol{\omega} \\ \boldsymbol{v} - \boldsymbol{v}_\mathrm{d} \\ \frac{\|\boldsymbol{v}_\mathrm{d}\|}{L}\left(\boldsymbol{M}_{22} - \tilde{m}\boldsymbol{I}\right)\boldsymbol{v}_\mathrm{d} \\ \mathbf{0} \\ -mg\beta\boldsymbol{\zeta}_\mathrm{d} \end{pmatrix}$$

and comparing equation (7.30) with equations (7.24) through (7.28) completes the proof. □

Remark 7.3.2. *The dynamics (7.30) conserve the following quantities*

$$\tilde{C}_1 = \frac{1}{2}\boldsymbol{\zeta}\cdot\boldsymbol{\zeta}, \qquad \tilde{C}_2 = \boldsymbol{\Upsilon}_1\cdot\boldsymbol{\zeta}, \qquad \tilde{C}_3 = \frac{1}{2}\boldsymbol{\Upsilon}\cdot\boldsymbol{\Upsilon}, \qquad \text{and} \qquad \tilde{C}_4 = \boldsymbol{m}_1\cdot(\boldsymbol{\Upsilon}\times\boldsymbol{\zeta}).$$

A promising approach to assessing stability would be to construct a candidate Lyapunov function $H_\mathrm{c} + \tilde{\Phi}(\tilde{C}_1, \tilde{C}_2, \tilde{C}_3, \tilde{C}_4)$ and to apply the energy-Casimir method, as in Section 7.2. Unfortunately, the term $-\boldsymbol{B}\boldsymbol{v}$ that appears as an exogenous force in equation (7.30) makes the rate of change of this candidate function indefinite. Because the energy-Casimir method fails to prove asymptotic stability, we resort to spectral analysis, which provides conditions for local asymptotic stability.

To linearize the dynamic equations, one must assume a specific model for the viscous forces and moments. We generalize the viscous force model of Assumption 7.1.2 by in-

corporating a linear damping term, as suggested in [23]:

$$\boldsymbol{f}_{\mathrm{v}}(\boldsymbol{v}) = -F_0(\boldsymbol{v})\boldsymbol{R}_{\mathrm{BC}}(\mu,\alpha)\begin{pmatrix} C_D(\alpha) \\ 0 \\ C_L(\alpha) \end{pmatrix} - d_v \boldsymbol{I}_{23}\boldsymbol{v}$$

where $d_v > 0$. Linear damping can easily be incorporated into the analysis of Section 7.2 and it simplifies spectral stability analysis of the cross-track control algorithm by eliminating two characteristic zeros. Because we are only considering a local model, we may choose standard expressions for C_L and C_D:

$$C_L(\alpha) = C_{L_\alpha}\alpha, \quad C_D(\alpha) = C_{D_0} + C_{D_1}\alpha^2,$$

where C_{L_α}, C_{D_0}, and C_{D_1} are positive constants. For simplicity, the viscous moment is modeled as linear damping:

$$\boldsymbol{\tau}_v(\boldsymbol{\omega}) = -d_\omega \boldsymbol{\omega},$$

where $d_\omega > 0$. In general, the angular rate damping moment for an underwater vehicle will scale differently about each axis, will be coupled among the three axes, and will depend on $\boldsymbol{v}$ through the dynamic pressure. While the analysis could accommodate such effects, they would have little qualitative impact on the results.

We assume that the component matrices in the generalized mass matrix (7.3) take the following form:

$$\boldsymbol{M}_{11} = \mathrm{diag}(J_1, J_2, J_2) \quad \text{and} \quad \boldsymbol{M}_{22} = \mathrm{diag}(m_1, m_2, m_2)$$

where $J_2 > J_1 > 0$ and $m_2 > m_1 > 0$. To simplify the analysis, we assume that $\boldsymbol{r}_{\mathrm{cm}} = \boldsymbol{0}$ and therefore that $\boldsymbol{M}_{12} = \boldsymbol{M}_{21}^T = \boldsymbol{0}$. Note that the control moment (7.29) cancels the gravitational moment due to $\boldsymbol{r}_{\mathrm{cm}}$, in any case.

The equilibrium motion for the cross-track control system is

$$\boldsymbol{\omega}_e = \boldsymbol{0}, \quad \boldsymbol{v}_e = u_{\mathrm{d}}\boldsymbol{b}_1, \quad \boldsymbol{m}_{1_e} = k_x L \boldsymbol{b}_1, \quad \boldsymbol{\Upsilon}_e = \boldsymbol{b}_2, \quad \boldsymbol{\zeta}_e = \boldsymbol{b}_3 \tag{7.32}$$

where $u_{\mathrm{d}} > 0$. Linearizing the equations about the equilibrium point, we obtain a 15×15 state matrix whose rank is eleven. The rank deficiency corresponds to the four conservation laws described in Remark 7.3.2. Eliminating the rows and columns that correspond to the four-dimensional null space leaves a full rank 11×11 matrix.

Proposition 7.3.3. *[81] If $\tilde{m}$, k_x, β, and d_ω are sufficiently large positive, then the equilibrium (7.32) is locally asymptotically stable.*

Proof: The proof follows from Lyapunov's indirect method [31]. Details of the spectral analysis, including specific conditions on the control parameters, are given in Appendix B.

Proposition 7.3.3 gives sufficient conditions for *local* asymptotic stability. As discussed in the next section, however, exhaustive simulation suggests that almost every initial state gives rise to a closed-loop trajectory that converges to the desired steady motion.

7.4 Simulations

The closed-loop system, under the modeling assumptions described at the end of the previous section, was exhaustively simulated over a large range of initial states to investigate convergence properties, and a large range of parameter values to investigate robustness to parametric uncertainty. The simulation model assumes a spheroid with a 6:1 fineness ratio and a nominal speed $u_{\mathrm{d}} = 1$ m/s. Other parameters used in the simulations are given in Table 7.2. All values are in SI units.

Initial states and parameter values were randomly selected for $20,000$ numerical simulations of the closed-loop equations. The values were selected from uniform distributions over the ranges shown in Table 7.3. (For unit vectors, the components were selected randomly and the vector was then normalized.) Convergence to the desired path was as-

Length, L	0.6	m	3.14	Lift curve slope, C_{L_α}	0.15
Frontal area, S	0.008	m_1	3.28	Coefficient, C_{D_0}	0.18
Bottom-heaviness, $\boldsymbol{r}_{\mathrm{cm}} \cdot \boldsymbol{b}_3$	0.01	m_2	6.02	Coefficient, C_{D_1}	0.0056
Control parameter, $\tilde{m}$	6.57	J_1	0.0031	Damping coefficient, d_ω	0.1
Speed control parameter, k_u	1	J_2	0.0581	Look-ahead factor, k_x	3

Table 7.2 Vehicle and control parameter values.

sessed using a small error tolerance for each eighty-second simulation. Every simulation resulted in a trajectory that converged to the desired path.

y_0	± 15	$\boldsymbol{\omega}_0 \cdot \boldsymbol{e}_i$	± 0.1	ΔC_{L_α}	25%
z_0	± 15	u_0	$u_d \pm 0.5$	ΔC_{D_0}	25%
$\boldsymbol{\Upsilon}_0 \cdot \boldsymbol{e}_i$	± 1	v_0	± 0.5		
$\boldsymbol{\zeta}_0 \cdot \boldsymbol{e}_i$	± 1	w_0	± 0.5		

Table 7.3 State and parameter ranges for simulations.

To compare the performance of the nonlinear controller with a more conventional method, a linear controller was also developed and implemented. The control law is defined as follows:

$$F_c = -\breve{k}_u(u - u_{\mathrm{d}}) \qquad \text{and} \qquad \boldsymbol{\tau}_c = \begin{pmatrix} \breve{k}_p(\phi_{\mathrm{d}} - \phi) \\ \breve{k}_q(\theta_{\mathrm{d}} - \theta) \\ \breve{k}_r(\psi_{\mathrm{d}} - \psi) \end{pmatrix} - d_\omega \boldsymbol{\omega},$$

where the subscript "d" denotes desired values. The desired roll angle is set to zero, and the desired pitch and yaw angles are selected according to the following line of sight

guidance rule

$$\begin{aligned} \theta_{\mathrm{d}} &= \arctan\left(\frac{-y}{k_x L}\right) \\ \psi_{\mathrm{d}} &= \arctan\left(\frac{z}{k_x L}\right). \end{aligned}$$

The controller gains were tuned, by trial and error, to obtain good performance at a nominal speed $u_{\mathrm{d}} = 0.8$ m/s, resulting in the following values:

$$\breve{k}_u = 1, \quad \breve{k}_p = 0.5, \quad \breve{k}_q = 0.5, \quad \breve{k}_r = 1.$$

For comparison, simulations were performed with the following initial conditions:

$$\boldsymbol{x}_0 = \begin{pmatrix} 0 \\ 15 \\ 0 \end{pmatrix}, \qquad \begin{pmatrix} \phi_0 \\ \theta_0 \\ \psi_0 \end{pmatrix} = \begin{pmatrix} 45^\circ \\ 45^\circ \\ 90^\circ \end{pmatrix}, \qquad \boldsymbol{\omega} = \mathbf{0}, \qquad \boldsymbol{v} = \boldsymbol{v}_{\mathrm{d}}.$$

Figure 7.3 illustrates the results for two different (off-nominal) desired speeds: $u_{\mathrm{d}} = 0.8$ m/s (solid line) and $u_{\mathrm{d}} = 1.3$ m/s (dotted line). The vehicle's orientation is indicated by an orthogonal triad. For the nominal case, the controller is clearly able to stabilize the system from this relatively large initial perturbation. At the higher speed, however, the response is quite oscillatory. The simulation illustrates the importance of careful control-parameter tuning for the linear controller and the sensitivity of controller performance to parameter values (in this case, the nominal speed). Figure 7.4 illustrates the performance of the nonlinear controller in response to the same initial state and parameter values. As one should expect, convergence is much less sensitive to the commanded speed.

7.5 Summary of the Cross-Track Control Algorithm

In this chapter we discussed the method of potential energy shaping used to develop a feedback control law, which asymptotically stabilizes longitudinal-axis translation of a

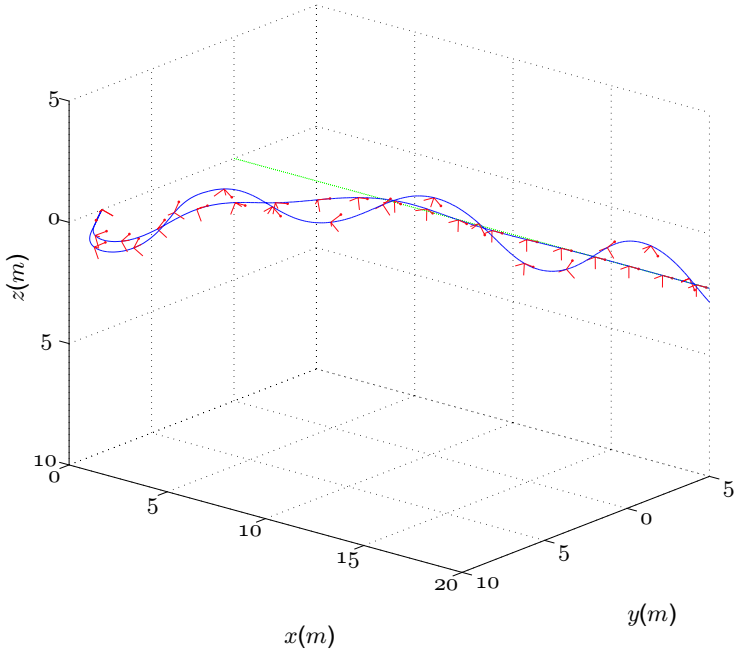


Figure 7.3 Simulation of the linear controller with $u_d = 0.8$ (solid) and $u_d = 1.3$ (dotted).

streamlined underwater vehicle in any desired, non-vertical direction. The result applies to vehicle models that have thrust, pitch, and yaw moment inputs. Because the stability analysis uses very general assumptions about the viscous force and moment, the resulting control law is robust to uncertainty in these terms.

Energy-based control, such as the potential shaping technique employed here, provides a constructive method for proving Lyapunov stability. The analysis provides conditions on the various control parameters for nonlinear stability. Here, the control Lyapunov function was constructed using the energy-Casimir method. The stability analysis proves local asymptotic stability and strongly suggests almost global asymptotic stability of the desired steady motion — a conjecture supported by numerical simulations.

The problem of directional control is a simplification of the more practical problem of path following. Intuitively, having asymptotically stabilized a vehicle's motion in a

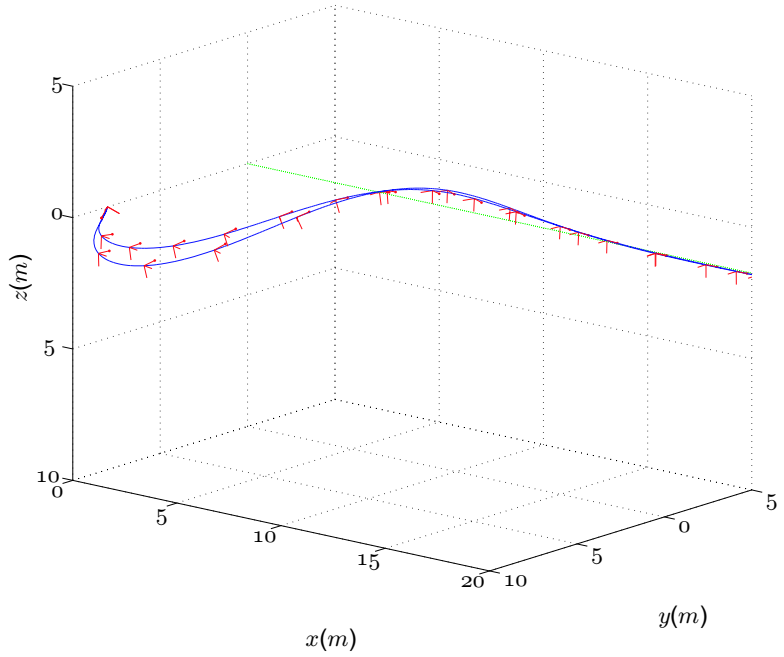


Figure 7.4 Simulation of the nonlinear controller with $u_d = 0.8$ (solid) and $u_d = 1.3$ (dotted).

desired inertial direction, one may choose the desired direction based on the cross-track error from a desired path. Line following is a common guidance problem for AUVs, and one that provides a natural, intermediate step between waypoint navigation and general, curvilinear path-following. Although energy-Casimir analysis failed to prove stability of the proposed cross-track control algorithm, spectral analysis did provide sufficient conditions for local asymptotic stability; simulations suggest that stability is almost globally asymptotic. Moreover, a thorough simulation-based investigation indicates excellent controller performance and robustness to parametric uncertainty.

Bibliography

[1] A S. Brierley et al. Antarctic krill under sea ice: Elevated abundance in a narrow band just south of ice edge. *Science*, 295:1890, 2002.

[2] M. Abramowitz and I. A. Stegun. *Handbook of mathematical functions*. Dover, New York, 1965. page 300.

[3] Anon. Unmanned Aircraft Systems Roadmap 2005-2030. Technical report, Department of Defense, August 2005.

[4] D. E. Aylor. A framework for examining inter-regional aerial transport of fungal spores. *Agricultural and Forest Meteorology*, 38:263 – 288, 1986.

[5] D. E. Aylor. Spread of plant disease on a continental scale: Role of aerial dispersal of pathogens. *Ecology*, 84:1989–1997, 2003.

[6] D. E. Aylor, M. T. Boehm, and E. J. Shields. Quantifying aerial concentrations of maize pollen in the atmospheric surface layer using remote-piloted airplanes and Lagrangian stochastic modeling. *Agricultural and Forest Meteorology*, 45:1003–1015, 2006.

[7] D. E. Aylor, W. E. Fry, H. Mayton, and J. L. Andrade-Piedra. Quantifying the rate of release and escape of Phytophthora infestans sporangia from a potato canopy. *Phytopathology*, 91(12):1189 – 1196, 2001.

[8] J. N. Bakambu and V. Polotski. Autonomous system for navigation and surveying in underground mines: Field reports. *J. Field Robot.*, 24(10):829–847, 2007.

[9] J. G. Bellingham and K. Rajan. Robotics in remote and hostile environments. *Science*, 318(5853):1098 – 1102, 16 November 2007.

[10] Y. Bestaoui, H. Dahmani, and K. Belharet. Geometry of translational trajectories for an autonomous aerospace vehicle with wind effect. In *47th AIAA Aerospace Sciences Meeting*, number AIAA-2009-1352, Orlando, FL, January 2009.

[11] J. D. Boissonat, A. Cerezo, and J. Leblond. Shortest paths of bounded curvature in the plane. In *Proc. of the IEEE International Conference on Robotics and Automation*, pages 2315 – 2320, Nice, France, May 1992.

[12] J. D. Boissonnat, A. Cerezo, and J. Leblond. A note on shortest paths in the plane subject to a constraint on the derivative of the curvature. In *INRIA Technical Report*, number 2160, INRIA, France, 1994.

[13] E. Borhaug and K. Y. Pettersen. Cross-track control for underactuated autonomous vehicles. In *Proc. Conf. on Decision and Control & European Control Conf.*, pages 602–608, 2005.

[14] Arthur E. Bryson and Yu-Chi Ho. *Applied Optimal Control*. Taylor and Francis, USA, 1975.

[15] X. Bui, P. Soukres, J. D. Boissonnat, and J. P. Laumond. Shortest path synthesis for Dubins non-holonomic robot. In *ICRA Technical Report*, INRIA, France, 1994.

[16] E. D. Cokelet, N. Tervalon, and J. G. Bellingham. Hydrography of the west spitsbergen current, svalbard branch: Autumn 2001. *J. Geophys. Res.*, 113:10.1029/2007JC004150, 2008.

[17] D. G. Schmale III, B. R. Dingus, and C. Reinholtz. Development and application of an autonomous unmanned aerial vehicle for precise aerobiological sampling above agricultural fields. *Journal of Field Robotics*, 25(3):133 – 147, 2008.

[18] B. R. Dingus, D. G. Schmale III, and C. F. Reinholtz. Development of an autonomous unmanned aerial vehicle for aerobiological sampling. *Phytopathology*, 97(7), 2007.

[19] K. D. Do and J. Pan. Robust and adaptive path following for underactuated autonomous underwater vehicles. In *Proc. American Control Conf.*, pages 1994–1999, 2003.

[20] L. E. Dubins. On curves of minimal length with a constraint on average curvature, and with prescribed initial and terminal positions and tangents. *American Journal of Mathematics*, 79(3):497 – 516, 1957.

[21] P. Encarnacão and A. Pascoal. 3D path following for autonomous underwater vehicle. In *Proc. IEEE Conf. on Decision and Control*, pages 2977–2982, 2000.

[22] J. A. Farrell, W. Li, S. Pang, and R. Arrieta. Chemical plume tracing experimental results with REMUS AUV. In *Proc. IEEE/MTS OCEANS*, pages 962–968, 2003.

[23] T. I. Fossen. *Marine Control Systems*. Marine Cybernetics, 2002, Trondheim, Norway.

[24] E. W. Frew, D. A. Lawrence, and S. Morris. Coordinated standoff tracking of moving targets using Lyapunov guidance vector fields. *Journal of Guidance, Control, and Dynamics*, 31(2):290–306, 2008.

[25] A. Gamliel, M. L. Gullino, and J. P. Stack. *Crop Biosecurity: Assuring our Global Food Supply*, pages 37–61. Springer, Netherlands, 2008.

[26] T. F. Hickerson. *Route Location and Design*. McGraw-Hill Book Company, New York, 1964. Chapter 6.

[27] S. F. Hoerner. *Fluid Dynamic Drag*. Published by the author, 1965, Midland Park, NJ.

[28] S. F. Hoerner and H. V. Borst. *Fluid Dynamic Lift*. Published by L. A. Hoerner, 1975, Brick Town, NJ.

[29] S. A. Isard and S. H. Gage. *Flow of Life in the Atmosphere*. Michigan State University Press, East Lansing, MI, 2001.

[30] E. W. Justh and P. S. Krishnaprasad. Equilibria and steering laws for planar formations. *Systems and Control Letters*, 52:25 – 38, 2004.

[31] H. K. Khalil. *Nonlinear Systems*. Prentice-Hall, third edition, 2002.

[32] H.-Y. Kim and C. A. Woolsey. Directional control of a streamlined underwater vehicle by feedback passivation. In *Proc. American Control Conf.*, pages 2998–3003, 2004.

[33] Donald E. Kirk. *Optimal Control Theory*. Prentice Hall, Englewood Cliffs, NJ, 1970.

[34] V. Kostov and E. Degtiariova-Kostova. Some properties of clothoids. In *INRIA Technical Report*, number 2752, INRIA, France, 1995.

[35] A. Kuparinen. Mechanistic models for wind dispersal. *Trends in Plant Science*, 11(6):296 – 301, 2006.

[36] George Leitmann. *The Calculus of Variations and Optimal Control*. Plenum Press, New York, NY, 1981.

[37] N. E. Leonard. Stability of a bottom-heavy underwater vehicle. *Automatica*, 33(3):331–346, 1997.

[38] N. E. Leonard. Stabilization of underwater vehicle dynamics with symmetry-breaking potentials. *Systems and Control Letters*, 32:35–42, 1997.

[39] S. L. Maldonado-Ramirez, D. G. Schmale III, E. J. Shields, and G. C. Bergstrom. The relative abundance of viable spores of gibberella zeae in the planetary boundary layer suggests the role of long-distance transport in regional epdemics of fusarium head blight. *Journal of Agricultural and Forest Meteorology*, 132:20 – 27, 2005.

[40] J. E. Marsden and T. S. Ratiu. *Introduction to Mechanics and Symmetry*. Springer-Verlag, 1994.

[41] T. G. McGee and J. K. Hedrick. Optimal path planning with a kinematic airplane model. *Journal of Guidance, Control, and Dynamics*, 30(2):629 – 633, 2007.

[42] T. G. McGee, S. Spry, and J. K. Hedrick. Optimal path planning in a constant wind with a bounded turning rate. In *AIAA Guidance Navigation and Control Conference and Exhibit*, number AIAA-2005-6186, San Francisco, CA, August 2005.

[43] R. McNeeley, R. V. Iyer, and P. Chandler. Tour planning for an unmanned air vehicle under wind conditions. *Journal of Guidance, Control, and Dynamics*, 30(5):1299 – 1306, 2007.

[44] D. S. Meek and D. J. Walton. Clothoid spline transition spirals. *Mathematics of Computation*, 59(199):117–133, 1992.

[45] D. S. Meek and D. J. Walton. A note on finding clothoids. *Journal of Computational and Applied Mathematics*, 170:433–453, 2004.

[46] D. S. Meek and D. J. Walton. A controlled clothoid spline. *Computers and Graphics*, 29:353–363, 2005.

[47] P. D. Miller. *Applied Asymptotic Analysis*. American Mathematical Society, first edition, 2006. pages 232 – 234.

[48] R. R. Murphy and S. Stover. Rescue robots for mudslides: A descriptive study of the 2005 la conchita mudslide response. *J. Field Robot.*, 25(1):3–16, 2008.

[49] J. F. Murtha, C. M. Cotting, A. Wolek, T. Aarons, and C. A. Woolsey. The educational impact of creating a new uav for curriculum enhancement. In *AIAA Atmospheric Flight Mechanics Conference*, pages 1 – 19, Chicago, IL, 2009.

[50] D. A. Paley. Cooperative control of an autonomous sampling network in an external flow field. In *Proceedings of the 47th IEEE Conference on Decision and Control*, Cancun, Mexico, December 2008.

[51] D. A. Paley, N. E. Leonard, and R. Sepulchre. Stabilization of symmetric formations to motion around convex loops. *Systems and Control Letters*, 57(3):209 – 215, March 2008.

[52] D. A. Paley, N. E. Leonard, R. Sepulchre, D. Grunbaum, and J. K. Parrish. Oscillator models and collective motion. *IEEE Control Systems Magazine*, 27(4):89 – 105, August 2007.

[53] D. A. Paley and C. Peterson. Stabilization of collective motion in a time-invariant flowfield. *Journal of Guidance, Control, and Dynamics*, 32(3):771 – 779, 2009.

[54] D. A. Paley, L. Techy, and C. A. Woolsey. Coordinated perimeter patrol with minimum-time alert response. In *AIAA Guidance, Navigation, and Control Conference*, number AIAA 2009-6210, Chicago, IL, 2009.

[55] L.S. Pontryagin, V. G. Boltyanskii, R. V. Gamkrelidze, and E. F. Mishchenko. *The Mathematical Theory of Optimal Processes*. John Wiley and Sons, New York, NY, 1962.

[56] R. T. Rysdyk. Unmanned aerial vehicle path following and target observation in wind. *Journal of Guidance Control and Dynamics*, 29(5):1092 – 1100, 2006.

[57] R. T. Rysdyk. Course and heading changes in significant wind. *Journal of Guidance Control and Dynamics*, 30(4):1168 – 1171, 2007.

[58] R. T. Rysdyk, C. Lum, and J. Vagners. Autonomous orbit coordination for two unmanned aerial vehicles. In *AIAA Guidance Navigation and Control Conference and Exhibit*, number AIAA 2005-6362, San Francisco, CA, August 2005.

[59] A. Scheuer and Th. Fraichard. Continuous-curvature path planning for car-like vehicles. In *Proceedings of the IEEE International Conference on Intelligent Robots and Systems*, pages 997 – 1003, Grenoble, France, 1997.

[60] R. Sepulchre, D. A. Paley, and N. E. Leonard. Stabilization of planar collective motion: All-to-all communication. *IEEE Transactions on Automatic Control*, 52(5):811 – 824, 2007.

[61] R. Sepulchre, D. A. Paley, and N. E. Leonard. Stabilization of planar collective motion with limited communication. *IEEE Trans. Automatic Control*, 53(3):706–719, 2008.

[62] E. J. Shields, J. T. Dauer, M. J. VanGessel, and G. Neumann. Horseweed (*Conyza canadensis*) seed collected in the planetary boundary layer. *Weed Science*, 54:1063–1067, 2006.

[63] E. J. Shields and A. M. Testa. Fall migratory flight initiation of the potato leafhopper, *Empoasca fabae*(*Homoptera: Cicadellidae*): Observations in the lower atmosphere using remote piloted vehicles. *Agricultural and Forest Meteorology*, 97:317–330, 1999.

[64] H.P. Spijkerboer, J.E. Beniers, D. Jaspers, H.J. Schouten, J. Goudriaan, R. Rabbinge, and W. van der Werf. Ability of the Gaussian plume model to predict and describe spore dispersal over a potato crop. *Ecological Modelling*, 155:1 – 18, 2002.

[65] E. Stokstad. Plant pathology: Deadly wheat fungus threatens world's breadbaskets. *Science*, 315(5820):1786 – 1787, 30 March 2007.

[66] P.B. Sujit, D. Kingston, and R. Beard. Cooperative forest fire monitoring using multiple UAVs. In *Proceedings of the 46th IEEE Conference on Decision and Control*, New Orleans, LA, December 2007.

[67] H. J. Sussmann and G. Tang. Shortest paths for the Reeds-Shepp car: A worked out example of the use of geometric techniques in nonlinear optimal control. In *SYCON - Rutgers Center for Systems and Control - Report*, New Brunswick, NJ, 1991.

[68] L. Techy, D. A. Paley, and C. A. Woolsey. UAV coordination on Time-Optimal paths in wind. *Journal of Guidance, Control, and Dynamics*. Submitted, in review.

[69] L. Techy, D. A. Paley, and C. A. Woolsey. UAV coordination on convex curves in wind: An environmental sampling application. In *European Control Conference*, pages 4967 – 4972, Budapest, Hungary, 2009.

[70] L. Techy and C. A. Woolsey. Minimum-time path planning for unmanned aerial vehicles in steady uniform winds. *Journal of Guidance, Control, and Dynamics*, 32(6):1736–1746, 2009.

[71] L. Techy, C. A. Woolsey, and D. G. Schmale III. Path planning for efficient UAV coordination in aerobiological sampling missions. In *Proceedings of the 47th IEEE Conference on Decision and Control*, pages 2814 – 2819, Cancun, Mexico, December 2008.

[72] L. Techy, C. A. Woolsey, and D. G. Schmale III. Monitoring the spread of a plant pathogen in the lower atmosphere using unmanned aerial vehicles and a buoyancy-controlled weather balloon. In *SAE Aerotech Congress*, number 2009-01-3125, Seattle, WA, 2009.

[73] Victor A. Topogonov. *Differential Geometry of Curves and Surfaces*. Birkhäuser, Boston, MA, 2006.

[74] United Nations Food and Agriculture Organization. Food outlook, November 2008. *http://www.fao.org/docrep/011/ai474e/ai474e03.htm*.

[75] A. van den Kroonenberg, T. Spieß, and J. Bange. First wind measurements with the meteorological UAV M2AV CAROLO. *Available online: http://ams.confex.com/ams/pdfpapers/139842.pdf*.

[76] T. G. Wetzel and R. L. Simpson. Unsteady crossflow separation location measurements on a maneuvering 6:1 prolate spheroid. *AIAA Journal*, 38(11):2063–2071, 1998.

[77] W. W. Whitacre. An autonomous underwater vehicle as a spacecraft attitude control simulator. In *AIAA Aerospace Sciences Meeting and Exhibit*, number AIAA 2005-137, 2005.

[78] Frank M. White. *Viscous Fluid Flow*. McGraw-Hill, Inc., New York, NY, second edition, 1991.

[79] C. Woodham and C. Woodham-Smith. *The Great Hunger: Ireland 1845-1849*. Penguin, The Netherlands, 1991.

[80] C. A. Woolsey. Directional control of a slender, underactuated AUV using potential shaping. In *Proc. IEEE Conf. on Decision and Control*, pages 6826–6831, 2006.

[81] C. A. Woolsey and L. Techy. Cross-track control of a slender, underactuated AUV using potential shaping. *Ocean Engineering*, 36(1):82–91, 2009.

[82] F. Zhang, D. M. Fratantoni, D. A. Paley, J. M. Lund, and N. E. Leonard. Control of coordinated patterns for ocean sampling. *International Journal of Control*, 80(7):1186 – 1199, 2007.

Appendix A

Computing BBB Trajectories: Details of Calculation

Given three trochoidal segments

$$x_{t_1}(t) = \frac{V_a}{\delta_1\omega}\sin(\delta_1\omega t + \phi_{t_1}) + V_w t + x_{t_{10}}, \quad t \in [0, t_A] \tag{A.1}$$

$$y_{t_1}(t) = \frac{-V_a}{\delta_1\omega}\cos(\delta_1\omega t + \phi_{t_1}) + y_{t_{10}} \tag{A.2}$$

$$x_{t_2}(t) = \frac{V_a}{\delta_2\omega}\sin(\delta_2\omega t + \phi_{t_2}) + V_w t + x_{t_{20}}, \quad t \in [t_A, t_B] \tag{A.3}$$

$$y_{t_2}(t) = \frac{-V_a}{\delta_2\omega}\cos(\delta_2\omega t + \phi_{t_2}) + y_{t_{20}} \tag{A.4}$$

$$x_{t_3}(t) = \frac{V_a}{\delta_3\omega}\sin(\delta_3\omega t + \phi_{t_3}) + V_w t + x_{t_{30}}, \quad t \in [t_B, T] \tag{A.5}$$

$$y_{t_3}(t) = \frac{-V_a}{\delta_3\omega}\cos(\delta_3\omega t + \phi_{t_3}) + y_{t_{30}}, \tag{A.6}$$

where $\delta_i \in \{-1, 1\}$, with $\delta_1 = \delta_3 = -\delta_2$, we seek to find the path parameter values t_A, t_B, T, and the integration constants $x_{t_{i0}}$, $y_{t_{i0}}$, ϕ_{t_i}, such that $[x_{t_1}, y_{t_1}, \psi_{t_1}]^T\big|_{t=0} = [x_{N_0}, y_{E_0}, \psi_0]^T$, and $[x_{t_3}, y_{t_3}, \psi_{t_3}]^T\big|_{t=T} = [x_{N_f}, y_{E_f}, \psi_f]^T$, where $\psi_{t_i} = \delta_i\omega t + \phi_{t_i}$. From the initial conditions we can determine $x_{t_{10}}$, $y_{t_{10}}$ and ϕ_{t_1} as

$$x_{t_{10}} = x_{N_0} - V_a/(\delta_1\omega)\sin(\phi_{t_1}) \tag{A.7}$$

$$y_{t_{10}} = y_{E_0} + V_a/(\delta_1\omega)\cos(\phi_{t_1}) \tag{A.8}$$

$$\phi_{t_1} = \psi_0 \tag{A.9}$$

From the final condition we have

$$x_{t_{30}} = x_{N_f} - V_a/(\delta_3\omega)\sin(\psi_f) - V_w T \tag{A.10}$$

$$y_{t_{30}} = y_{E_f} + V_a/(\delta_3\omega)\cos(\psi_f) \tag{A.11}$$

$$\phi_{t_3} = \psi_f - \delta_3\omega T \tag{A.12}$$

We are left with six unknowns: $x_{t_{20}}$, $y_{t_{20}}$, ϕ_{t_2}, t_A, t_B and T. One may write a total of six continuity equations at points t_A and t_B (two for position, one for heading at each point) as follows

$$x_{t_1}(t_A) = x_{t_2}(t_A), \qquad x_{t_2}(t_B) = x_{t_3}(t_B) \tag{A.13}$$

$$y_{t_1}(t_A) = y_{t_2}(t_A), \qquad y_{t_2}(t_B) = y_{t_3}(t_B) \tag{A.14}$$

$$\psi_{t_1}(t_A) = \psi_{t_2}(t_A), \qquad \psi_{t_2}(t_B) = \psi_{t_3}(t_B) \tag{A.15}$$

From equation (A.15) and noting that $\delta_1 = -\delta_2$, one may write

$$\phi_{t_2} = 2\delta_1\omega t_A + \phi_{t_1}, \qquad t_B = \frac{\psi_f + \delta_2\omega T - \phi_{t_2}}{2\delta_2\omega} = t_A + \frac{T}{2} + \frac{\psi_f - \psi_0}{2\delta_2\omega}. \tag{A.16}$$

From equations (A.13)-(A.14) we can write four additional continuity equations.

$$\begin{pmatrix} x_{t_{20}} \\ y_{t_{20}} \end{pmatrix} = \begin{pmatrix} x_{t_{30}} - 2\frac{V_a}{\delta_2\omega}\sin(\delta_2\omega t_B + \phi_{t_2}) \\ y_{t_{30}} + 2\frac{V_a}{\delta_2\omega}\cos(\delta_2\omega t_B + \phi_{t_2}) \end{pmatrix} \tag{A.17}$$

$$\begin{pmatrix} x_{t_{10}} \\ y_{t_{10}} \end{pmatrix} = \begin{pmatrix} x_{t_{20}} + 2\frac{V_a}{\delta_2\omega}\sin(\delta_2\omega t_A + \phi_{t_2}) \\ y_{t_{20}} - 2\frac{V_a}{\delta_2\omega}\cos(\delta_2\omega t_A + \phi_{t_2}) \end{pmatrix} \tag{A.18}$$

Substituting $x_{t_{30}}$ and $y_{t_{30}}$ from equations (A.10)-(A.11) into (A.17), and then $x_{t_{20}}$ and $y_{t_{20}}$ from equation (A.17) into (A.18), and using expressions (A.16), we obtain two transcendental equations for t_A and T

$$\mathbf{f}(t_A, T) = \begin{pmatrix} \frac{2V_a}{\delta_1\omega}\sin(\delta_1\omega t_A + \phi_{t_1}) + x_{t_{10}} - x_{t_{30}} + \frac{2V_a}{\delta_2\omega}\sin(\delta_2\omega\frac{T}{2} + \frac{\psi_f}{2} + \delta_1\omega t_A + \frac{\phi_{t_1}}{2}) \\ -\frac{2V_a}{\delta_1\omega}\cos(\delta_1\omega t_A + \phi_{t_1}) + y_{t_{10}} - y_{t_{30}} - \frac{2V_a}{\delta_2\omega}\cos(\delta_2\omega\frac{T}{2} + \frac{\psi_f}{2} + \delta_1\omega t_A + \frac{\phi_{t_1}}{2}) \end{pmatrix} \equiv \mathbf{0},$$

where $x_{t_{30}}(T)$ and $y_{t_{30}}(T)$ depend on T as in equations (A.10)-(A.11). The roots of the equations can be found using a second order Newton-Raphson method. Define the mapping

$$\mathbf{g}(t_A, T) = \begin{pmatrix} t_A \\ T \end{pmatrix} - \mathbf{J}^{-1}(t_A, T)\mathbf{f}(t_A, T),$$

where

$$\mathbf{J}(t_A, T) = \begin{bmatrix} \frac{\partial f_1}{\partial t_A}(t_A, T) & \frac{\partial f_1}{\partial T}(t_A, T) \\ \frac{\partial f_2}{\partial t_A}(t_A, T) & \frac{\partial f_2}{\partial T}(t_A, T) \end{bmatrix}$$

is the Jacobian matrix. If the initial guess $(t_{A_0}, T_0)^T$ is close enough to the true solutions, then the mapping defined by

$$\begin{pmatrix} t_{A_{i+1}} \\ T_{i+1} \end{pmatrix} = \mathbf{g}(t_{A_i}, T_i)$$

converges to the root. In case the Jacobian matrix becomes singular, the root-finding algorithm may be started with a different initial condition.

Appendix B

Spectral Stability Analysis of the Closed-loop AUV Control System

Computing the characteristic polynomial for the 11-dimensional system, we end up with a characteristic polynomial that can be factored into two fourth order, one second order and one first order polynomials. The latter two are

$$s + \frac{k_u}{m_1} = 0$$

and

$$J_1 s^2 + d_\omega s + mg\beta = 0.$$

The former polynomial corresponds to an eigenmode associated with longitudinal speed. The latter corresponds to an eigenmode associated with rolling motion. Note that roll stability requires

$$d_\omega > 0 \quad \text{and} \quad \beta > 0.$$

The remaining two polynomials are denoted $P_1(s)$ and $P_2(s)$.

Routh-Hurwitz Analysis of $P_1(s)$.

Consider the polynomial

$$P_1(s) = a_4 s^4 + a_3 s^3 + a_2 s^2 + a_1 s + a_0$$

where the coefficients a_i are

$$\begin{aligned} a_4 &= J_2 L m_2 \\ a_3 &= L(d_v J_2 + d_\omega m_2) \\ a_2 &= L(d_v d_\omega + u_{\mathrm{d}}^2(m_1^2 + k_x L m_2(\tilde{m} - m_1))) \\ a_1 &= u_{\mathrm{d}}^2(\tilde{m} - m_1)(k_x L d_v + u_{\mathrm{d}}(m_2 - m_1)) \\ a_0 &= u_{\mathrm{d}}^3 d_v(\tilde{m} - m_1). \end{aligned}$$

In order for the polynomial to have roots with negative real parts, each coefficient a_i must be positive, and so must each element in the left-most column of the Routh array:

$$\begin{array}{ccc} a_4 & a_2 & a_0 \\ a_3 & a_1 & \\ b_1 & a_0 & \\ c_1 & & \\ a_0 & & \end{array}$$

where

$$b_1 = \frac{a_3 a_2 - a_4 a_1}{a_3} \qquad \text{and} \qquad c_1 = \frac{b_1 a_1 - b_2 a_3}{b_1}.$$

The coefficients a_i are all positive under existing assumptions and conditions, provided

$$\tilde{m} - m_1 > 0.$$

The term b_1 is

$$b_1 = \frac{C_{b_1}\tilde{m} + C_{b_0}}{d_v J_2 + d_\omega m_2}$$

where

$$\begin{aligned} C_{b_1} &= k_x L(d_{\omega_2} + k_{\omega_2})(m_2 u_\mathrm{d})^2 - J_2 m_2 u_\mathrm{d}^3 (m_2 - m_1) \\ C_{b_0} &= d_v^2 (d_{\omega_2} + k_{\omega_2}) J_2 L + m_1 m_2 u_\mathrm{d}^2 \left(J_2 (m_2 - m_1) u_\mathrm{d} - (d_{\omega_2} + k_{\omega_2}) L (k_x m_2 - m_1) \right). \end{aligned}$$

The term C_{b_1} is positive provided

$$k_{\omega_2} + d_{\omega_2} > \frac{J_2 u_\mathrm{d} (m_2 - m_1)}{k_x L m_2}.$$

The term C_{b_0} is positive provided

$$k_{\omega_2} + d_{\omega_2} > \frac{J_2 u_\mathrm{d} (m_2 - m_1)}{L (k_x m_2 - m_1)}.$$

Note that different role of the look-ahead control parameter k_x in the latter expression. If the characteristic length L is the vehicle length, one would typically pick the look-ahead parameter greater than unity, in practice.

Examining the c_1 coefficient:

$$c_1 = \frac{(\tilde{m} - m_1) u_\mathrm{d}^2 (C_{c_1} \tilde{m} + C_{c_0})}{b_1 (d_v J_2 + d_\omega m_2)}$$

where

$$\begin{aligned} C_{c_1} &= m_2 u_\mathrm{d}^2 \left(d_\omega k_x L m_2 + J_2 (m_1 - m_2) u_\mathrm{d} \right) \left(d_v k_x L + (m_2 - m_1) u_\mathrm{d} \right) \\ C_{c_0} &= d_v^3 J_2 L (d_\omega k_x L - J_2 u_\mathrm{d}) \\ &\quad - m_1 (m_2 - m_1) m_2 u_\mathrm{d}^3 \left(d_\omega L (k_x m_2 - m_1) - J_2 (m_2 - m_1) u_\mathrm{d} \right) \\ &\quad + d_v^2 L \left(d_\omega^2 k_x L m_2 - d_\omega J_2 (m_1 + m_2) u_\mathrm{d} + J_2 k_x L m_1^2 u_\mathrm{d}^2 \right) \\ &\quad - d_v L m_1 u_\mathrm{d} \left(d_\omega^2 m_2 + d_\omega k_x L m_2 (k_x m_2 - m_1) u_\mathrm{d} - J_2 (m_2 - m_1)(m_1 + k_x m_2) u_\mathrm{d}^2 \right) \end{aligned}$$

The term C_{c_1} is positive according to previous conditions, so c_1 is positive provided

$$\tilde{m} > -\frac{C_{c_0}}{C_{c_1}}.$$

If the conditions stated above hold, then each root of has strictly negative real part.

Routh-Hurwitz Analysis of $P_2(s)$. We make a slight abuse of notation by defining

$$P_2(s) = a_4 s^4 + a_3 s^3 + a_2 s^2 + a_1 s + a_0$$

with coefficients

$$\begin{aligned}
a_4 &= J_2 k_x L m_2 \\
a_3 &= k_x L(d_v J_2 + d_\omega m_2) \\
a_2 &= k_x L(d_v(d_{\omega_2} + k_{\omega_2}) + u_{\mathrm{d}}^2(m_1^2 + k_x L m_2(\tilde{m} - m_1)) + m_2 m g \beta) \\
a_1 &= (k_x L u_{\mathrm{d}}^2(\tilde{m} - m_1) + mg\beta)(k_x L d_v + u_{\mathrm{d}}(m_2 - m_1)) \\
a_0 &= d_v u_{\mathrm{d}}(k(\tilde{m} - m_1)u_{\mathrm{d}}^2 + mg\beta).
\end{aligned}$$

Again, in order for the polynomial to have roots with negative real parts, each coefficient a_i must be positive and so must each element in the left-most column of the Routh array. The coefficients a_i are all positive under existing assumptions and conditions.

The term b_1 is

$$b_1 = \frac{C_{b_1}\beta + C_{b_0}}{d_v J_2 + d_\omega m_2}$$

where

$$\begin{aligned}
C_{b_1} &= mgk_x L m_2^2 d_\omega - mgJ_2 u_{\mathrm{d}} m_2(m_2 - m_1) \\
C_{b_0} &= k_x d_v^2 J_2 L d_\omega \\
&\quad + m_2 k_x u_{\mathrm{d}}^2(L d_\omega(m_1^2 - k_x L m_1 m_2 + k_x L m_2 \tilde{m}) - J_2(m_2 - m_1)(\tilde{m} - m_1)u_{\mathrm{d}}) \\
&\quad + k_x d_v L(d_\omega^2 m_2 + J_2 m_1^2 u_{\mathrm{d}}^2)
\end{aligned}$$

The term $C_{b_1} > 0$ under existing assumptions and conditions. If also

$$d_\omega > \frac{J_2 u_{\mathrm{d}}(m_2 - m_1)(\tilde{m} - m_1)}{L(m_1^2 + k_x m_2(\tilde{m} - m_1))},$$

then $C_{b_0} > 0$ so that b_1 will be positive.

The parameter c_1 is

$$c_1 = \frac{(k_x(\tilde{m} - m_1)u_{\mathrm{d}}^2 + mg\beta)(C_{c_1}\beta + C_{c_0})}{b_1(d_v J_2 + d_\omega m_2)}$$

where

$$\begin{aligned} C_{c_1} &= mgm_2(d_\omega k_x L m_2 + J_2(m_1 - m_2)u_{\mathrm{d}})(d_v k_x L + (m_2 - m_1)u_{\mathrm{d}}) \\ C_{c_0} &= -d_v k_x L u_{\mathrm{d}}(d_v J_2 + d_\omega m_2)^2 + (d_v k_x L + (m_2 - m_1)u_{\mathrm{d}}) \\ &\quad \big[J_2 m_2(d_v k_x L + (m_2 - m_1)u_{\mathrm{d}})(k_x(m_1 - \tilde{m})u_{\mathrm{d}}^2) + k_x L(d_v J_2 + d_{\omega_2} m_2) \\ &\quad (d_v d_\omega + (m_1^2 + k_x m_2(\tilde{m} - m_1))u_{\mathrm{d}}^2)\big] \end{aligned}$$

The term C_{c_1} is positive under existing conditions. Choosing

$$\beta > -\frac{C_{c_0}}{C_{c_1}}$$

ensures that c_1 is positive and therefore that every root of $P_2(s)$ has strictly negative real part. The stated conditions are admittedly complicated, but in the physically realistic examples we have considered these conditions are satisfied by the following, simpler conditions:

$$\tilde{m} > m_1, \quad k_x > 0, \quad \beta > 0, \quad \text{and} \quad d_\omega > \frac{J_2 u_{\mathrm{d}}(m_2 - m_1)}{L(k_x m_2 - m_1)}.$$