AUTONOMOUS SOARING AND SURVEILLANCE IN WIND FIELDS WITH AN UNMANNED AERIAL VEHICLE

by

Chen Gao

A thesis submitted in conformity with the requirements for the degree of Doctor of Philosophy
Graduate Department of Aerospace Science and Engineering
University of Toronto

© Copyright 2015 by Chen Gao
Abstract

Autonomous soaring and surveillance in wind fields with an unmanned aerial vehicle

Chen Gao
Doctor of Philosophy
Graduate Department of Aerospace Science and Engineering
University of Toronto
2015

Small unmanned aerial vehicles (UAVs) play an active role in developing a low-cost, low-altitude autonomous aerial surveillance platform. The success of the applications needs to address the challenge of limited on-board power plant that limits the endurance performance in surveillance mission. This thesis studies the mechanics of soaring flight, observed in nature where birds utilize various wind patterns to stay airborne without flapping their wings, and investigates its application to small UAVs in their surveillance missions. In a proposed integrated framework of soaring and surveillance, a bird-mimicking soaring maneuver extracts energy from surrounding wind environment that improves surveillance performance in terms of flight endurance, while the surveillance task not only covers the target area, but also detects energy sources within the area to allow for potential soaring flight. The interaction of soaring and surveillance further enables novel energy based, coverage optimal path planning. Two soaring and associated surveillance strategies are explored. In a so-called static soaring surveillance, the UAV identifies spatially-distributed thermal updrafts for soaring, while incremental surveillance is achieved through gliding flight to visit concentric expanding regions. A Gaussian-process-regression-based algorithm is developed to achieve computationally-efficient and smooth updraft estimation. In a so-called dynamic soaring surveillance, the UAV performs one cycle of dynamic soaring to harvest energy from the horizontal wind gradient to complete one surveillance task by visiting from one target to the next one. A Dubins-path-based trajectory planning approach is proposed to maximize wind energy extraction and ensure smooth transition between surveillance tasks. Finally, a nonlinear trajectory tracking controller is designed for a full six-degree-of-freedom nonlinear UAV dynamics model and extensive simulations are carried to demonstrate the effectiveness of the proposed soaring and surveillance strategies.
Acknowledgements

The past five years at UTIAS have been a challenging as well as rewarding experience in my life. I would not be able to reach the destination without the tremendous support of many people. My special appreciation goes to my supervisor, Prof. Hugh Liu, who provided me with the opportunity to start this rewarding journey. His guidance, patience, and expertise helped me conquer difficult obstacles throughout every stage of this research. I also would like to thank my research committee members: Prof. Peter Grant and Prof. Craig Steeves, for their insights, guidance, assistance, and warm encouragement throughout the journey. Their valuable feedback and constructive comments during every DEC meeting helped me improve my thesis significantly.

I would like to show my sincere appreciation to all faculty members in the University of Toronto, Institute for Aerospace Studies (UTIAS) for their generous support during the past five years. My special gratitude is extended to all fellow students and colleagues in the flight system and control (FSC) group at UTIAS, especially Keith Leung, Sohrab Haghighat, Jason Zhang, Laurent Heirendt, Connie Phan, Zhongjie Lin, and Wen Fan. Their companionship and encouragement have helped me get to where I am today. Thanks are also extended to all visiting scholars, especially Zhan Li for sharing his photos with me.

The last but not the least, I would like to express my deepest gratitude to my parents for their unconditional love during the past five years. This thesis would not have been possible without their consistent support, encouragement, and care. I also would like to thank my friend Kayla for providing emotional support during those difficult times.
# Contents

1 Introduction .................................. 1  
1.1 Motivation .................................. 1  
1.2 Related work ................................ 2  
1.2.1 Autonomous soaring ....................... 2  
1.2.2 Trajectory planning and control in aerial surveillance .......... 7  
1.3 Thesis contributions .......................... 9  

2 Soaring flight .................................. 11  
2.1 Soaring flight environment ..................... 11  
2.2 Wind models in the planetary boundary layer ................... 12  
2.2.1 The wind gradient model in the surface layer ............... 12  
2.2.2 Updraft models in the mixed layer .................. 13  
2.3 Equations of motion derivation in the presence of winds ........ 16  
2.3.1 Frames of reference .......................... 16  
2.3.2 Equations of motion for a small UAV in the presence of winds ..... 18  
2.4 Static soaring flight ........................... 21  
2.5 Dynamic soaring flight .......................... 22  
2.6 Summary .................................... 25  

3 Soaring Surveillance Problem Formulation ................. 27  
3.1 Aerial surveillance task ....................... 27  
3.2 Autonomous soaring surveillance ............... 27  
3.3 Soaring surveillance problem statements and formulations ....... 29  
3.3.1 Soaring surveillance problem statements .......... 29  
3.3.2 Static soaring surveillance trajectory planning problem formulation ..... 30
3.3.3 Updraft identification problem formulation .......................... 31
3.3.4 Dynamic soaring surveillance trajectory planning problem formulation .......................... 32
3.3.5 Trajectory tracking problem formulation .............................. 33

3.4 Summary ............................................................................. 34

4 Static soaring surveillance in the quasi-static updraft field 35
4.1 Surveillance and exploring points in the surveillance area ................. 35
4.1.1 Surveillance and exploring points in the surveillance area .......... 35
4.1.2 The traveling salesman problem .............................................. 36
4.1.3 The traveling salesman problem for Dubins’ vehicle ................. 38
4.1.4 The exploring point determination ......................................... 40
4.2 Static soaring surveillance approach ...................................... 41
4.2.1 Visit surveillance and exploring points ................................. 43
4.2.2 Updraft identification by Gaussian process regression with boundary
constraints ................................................................. 47
4.2.3 Updraft soaring strategy ...................................................... 53
4.2.4 Updraft soaring results ....................................................... 56
4.2.5 Sensitivity analysis of the static soaring surveillance approach ......... 58

4.3 Static soaring surveillance simulation results ............................. 64
4.3.1 Soaring surveillance demo in a 1 km wind field ....................... 64
4.3.2 Large wind field (10 km) results ......................................... 67
4.4 Summary ............................................................................. 68

5 Dynamic soaring surveillance in a wind gradient field 77
5.1 Dynamic soaring surveillance problem formulation ....................... 77
5.1.1 Surveillance case .............................................................. 77
5.1.2 Problem formulation ......................................................... 78

5.2 Dynamic soaring surveillance trajectory planning ....................... 80
5.2.1 The UAV’s trajectory ........................................................ 80
5.2.2 Dubins’ paths analysis ....................................................... 81
5.2.3 Flight-path-angle-profile determination .............................. 82
5.2.4 Dubins’ path choices: Heading angle rate determination .......... 83
5.2.5 Dynamic soaring trajectory planning approach ....................... 85
6 Trajectory tracking controller design

6.1 Mathematical model ................................................. 99
6.2 Trajectory tracking controller design .......................... 102
   6.2.1 Nonlinear mapping in the outer loop .................... 102
   6.2.2 Inner-loop dynamics ....................................... 105
   6.2.3 Linear Quadratic Regulator V.S. State Dependent Riccati Equation ... 105
   6.2.4 Comparison study .......................................... 107
6.3 Simulation results .................................................. 109
   6.3.1 Static soaring trajectory tracking results .............. 109
   6.3.2 Dynamic soaring trajectory tracking results .......... 110
6.4 Summary .............................................................. 111

7 Conclusions and future work ...................................... 120

7.1 Conclusions ......................................................... 120
7.2 Future work .......................................................... 121

Bibliography ............................................................. 122
# List of Tables

4.1 Surveillance and exploring points ........................................ 36  
4.2 Quantitative analysis of the coverage and traveling distance ........ 41  
4.3 Aerosonde UAV parameters .................................................. 45  
4.4 Updraft parameters ............................................................. 67  
5.1 The relationship between flight path angle and heading angle ....... 82  
5.2 Heading angle change along \{lll\}-type path between each pair of points ... 84  
5.3 Appropriate path look-up table ............................................. 85  
5.4 Energy harvesting results between two points ......................... 90  
5.5 Energy harvesting results ...................................................... 91
List of Figures

1.1 Static soaring: riding updrafts high in the air ........................................ 2
1.2 Dynamic soaring: exploiting wind gradients near sea surface ................. 2

2.1 Troposphere structure ........................................................................ 12
2.2 Planetary boundary layer (PBL) structure ............................................. 12
2.3 The logarithmic-like ($A = 1.2$) wind gradient profile with a gradient slope value $(\beta_{tr} = 0.35 \text{ s}^{-1}, z_{tr} = 30 \text{ m})$ .......................................................... 14
2.4 A plume of rising air and updraft bubbles in the PBL ................................. 15
2.5 The updraft model whose center $(x_1, y_1) = (0, 0) \text{ m}$, radius $R_1 = 50 \text{ m}$ and center strength $W_1 = 10 \text{ m/s}$ ................................................................. 15
2.6 Simulated wind field generated from the weather prediction system .......... 16
2.7 Rotation relationship among the vehicle-carried east-north-up (ENU), flight-path, and wind reference frames ................................................................. 17
2.8 Forces analysis on the UAV in the calm air and vertical winds .................. 22
2.9 Static soaring by loitering around an updraft ........................................ 23
2.10 Forces analysis on the UAV in windward climb, dive, leeward climb, and dive in horizontal wind gradients ................................................................. 24
2.11 Dynamic soaring: windward climbing $(\gamma = \frac{\pi}{4}, \psi = -\frac{\pi}{2}, \dot{\gamma} = 0, \dot{\psi} = 0)$ ................................................................. 25
2.12 Dynamic soaring: leeward diving $(\gamma = -\frac{\pi}{4}, \psi = \frac{\pi}{2}, \dot{\gamma} = 0, \dot{\psi} = 0)$ ................................................................. 25
2.13 Flight states along the windward climbing ........................................ 26
2.14 Flight states along the leeward diving ............................................... 26

3.1 Surveillance area with 50 uniformly-distributed targets .............................. 28
3.2 Surveillance area with 300 uniformly-distributed targets .............................. 28
3.3 Surveillance area with coordinate system (X-Y-Z) .................................... 29
3.4 Soaring surveillance architecture ........................................... 29
3.5 The periodic characteristics of dynamic soaring ...................... 32
3.6 Trajectory planning and control diagram ............................... 34

4.1 Surveillance division and visiting targets .............................. 37
4.2 The sub-optimal visiting sequence (ETSP solution) in Region 1 and 2 39
4.3 Dubins’ surveillance path in Region 1 and 2 .......................... 40
4.4 The original and augmented surveillance paths in Region 1 and 2 .... 42
4.5 The relationship between airspeed and ground speed ............... 44
4.6 Airspeed of steady flight with maximum $L/D$ for various turn rate cases .... 46
4.7 Lift coefficient of steady flight with maximum $L/D$ for various turn rate cases .... 46
4.8 Flight path angle of steady flight with maximum $L/D$ for various turn rate cases ... 47
4.9 Bank angle of steady flight with maximum $L/D$ for various turn rate cases .... 47
4.10 Sink rate of steady flight with maximum $L/D$ for various turn rate cases .... 47
4.11 The flowchart of the Dubins’ surveillance trajectory calculation .......... 48
4.12 Dubins’ surveillance trajectory in Region 1 and 2 in the presence of horizontal winds ......................................................... 49
4.13 Airspeed and lift coefficient along the trajectory in Region 1 .......... 50
4.14 Airspeed and lift coefficient along the trajectory in Region 2 .......... 50
4.15 True and estimated wind fields from Region 1 to Region 7 .......... 53
4.16 The UAV $P_g$ and estimated point $P_c$ on the $X − Y$ plane ........ 54
4.17 Vector field: the desired heading direction ........................... 54
4.18 The flowchart of the proposed updraft soaring strategy ............. 57
4.19 Updraft soaring strategy: Dubins-path-based guidance and estimated point correction ......................................................... 58
4.20 Updraft soaring results (2D soaring path) when height gain is 100 m .... 58
4.21 Airspeed, lift coefficient and height change along soaring flight .......... 59
4.22 Bank angle, sink rate and turning radius change along soaring flight .......... 60
4.23 Dubins’ surveillance path in the $[−100, 100] \times [−100, 100]$ m field .... 61
4.24 Estimation sensitivity with respect to thermal’s position ($x_1(t_0), y_1(t_0)$) m using 19 measurements ................................. 62
4.25 Estimation sensitivity with respect to thermal’s position \((x_1(t_0), y_1(t_0))\) m using 95 measurements .............................................................. 62
4.26 Estimation sensitivity with respect to thermal’s central strength \(W_1\) m/s ........... 63
4.27 Estimation sensitivity with respect to thermal’s radius \(R_1\) m ..................... 64
4.28 Estimation sensitivity with respect to thermal’s oscillation amplitude \(a\) m ........ 65
4.29 Estimation sensitivity with respect to thermal’s oscillation frequency \(b\) rad/s ...... 66
4.30 Vertical and horizontal components of wind speed results on 10 \(\times\) 10 interpolation points ................................................................. 67
4.31 Interpolation results based on 10 \(\times\) 10 wind speed data when \(t = 0\) ............ 68
4.32 Vertical wind estimation results in Region 1 .................................................. 69
4.33 Superposing surveillance trajectory over vertical wind speed estimation error 
\(|f - \bar{f}|\) in Region 1 ................................................................. 69
4.34 2D updraft soaring path ................................................................. 69
4.35 3D updraft soaring path ................................................................. 69
4.36 Dubins’ surveillance trajectory in Region 2 .................................................. 70
4.37 Flight states along the surveillance in Region 2 ........................................... 70
4.38 Dubins’ surveillance trajectory in Region 3 .................................................. 70
4.39 Flight states along the surveillance in Region 3 ........................................... 70
4.40 Dubins’ surveillance trajectory in Region 4 .................................................. 71
4.41 Flight states along the surveillance in Region 4 ........................................... 71
4.42 Dubins’ surveillance trajectory in Region 5 .................................................. 71
4.43 Flight states along the surveillance in Region 5 ........................................... 71
4.44 Dubins’ surveillance trajectory in Region 6 .................................................. 72
4.45 Flight states along the surveillance in Region 6 ........................................... 72
4.46 Dubins’ surveillance trajectory in Region 7 .................................................. 72
4.47 Dubins’ surveillance trajectory in Region 8 .................................................. 73
4.48 Dubins’ surveillance trajectory in Region 9 .................................................. 73
4.49 Flight states along the surveillance in Region 7 ........................................... 74
4.50 Flight states along the surveillance in Region 8 ........................................... 74
4.51 Flight states along the surveillance in Region 9 ........................................... 74
4.52 Average climb rate in each region ......................................................... 74
4.53 Wind speed estimation results in the simulated field ................................... 75
4.54 Wind speed estimation error \(|f - \bar{f}|\) in the simulated field .......................... 75
4.55 10 km \times 10 km wind field ................................................. 75
4.56 4 km \times 4 km estimation ..................................................... 75
4.57 5.5 km \times 5.5 km wind field .................................................. 76
4.58 10 km \times 10 km estimation ..................................................... 76
5.1 Surveillance area ................................................................. 78
5.2 The logarithmic-like \((A = 1.2)\) wind gradient profile with a gradient slope value
\((\beta_{tr} = 0.35 \text{ s}^{-1}, z_{tr} = 30 \text{ m})\) ............................................ 78
5.3 Periodic dynamic soaring ....................................................... 79
5.4 Energy-gain flight in left-turning and right-turning circular flight .............. 82
5.5 Plots of flight-path-angle against heading-angle .................................. 83
5.6 \(\varphi - \gamma\) function: the B-spline curve with arbitrary control points .......... 84
5.7 Location relationship between two specific points ................................ 84
5.8 The ground speed and airspeed ............................................... 86
5.9 Traveling salesman problem solution ......................................... 89
5.10 Energy change along dynamic soaring surveillance trajectory in various wind
gradient cases ................................................................. 90
5.11 Energy change along dynamic soaring surveillance trajectory ............... 92
5.12 Energy change along gliding surveillance trajectory .......................... 93
5.13 3D dynamic soaring surveillance trajectory .................................... 94
5.14 Dynamic soaring surveillance trajectory’s ground path ..................... 95
5.15 Three-dimensional location along the trajectory from point 1 to 2 (green line),
point 2 to 3 (black line), point 3 to 4 (brown line) and point 4 to 1 (red line) .... 96
5.16 Airspeed, flight path angle and heading angle along the trajectory from point 1
to 2 (green line), point 2 to 3 (black line), point 3 to 4 (brown line) and point 4
to 1 (red line) ................................................................. 97
5.17 Angle of attack and bank angle along the trajectory ........................... 98
6.1 Trajectory planning and control ............................................... 100
6.2 Angular rates \((p, q, r)\) tracking performance by LQR and SDRE ................ 108
6.3 Control surface deflection ....................................................... 109
6.4 A part of static soaring trajectory tracking performance ..................... 112
6.5 Airspeed, flight path angle, and heading angle tracking performance static soaring
6.6 Angle of attack, side-slip angle, and bank angle tracking performance static soaring
6.7 Control inputs along the static soaring trajectory
6.8 A part of dynamic soaring trajectory tracking performance
6.9 Airspeed, flight path angle, and heading angle tracking performance in dynamic soaring
6.10 Angle of attack, side-slip angle, and bank angle tracking performance in dynamic soaring
6.11 Control inputs along the dynamic soaring trajectory
Nomenclature

A  The wind gradient profile, which describes the exponential or logarithmic-like profile over altitudes

AR  Aspect ratio

$C_L, C_D$  Lift and drag coefficient

$C_Y$  Side force coefficient

$C_x, C_y, C_z$  Rotation matrices

$C_{D_0}$  Parasitic drag coefficient

$C_l$  Roll moment coefficient

$C_m$  Pitch moment coefficient

$C_n$  Yaw moment coefficient

D  Drag force [N]

E  UAV’s total energy [J]

G  Gravity force [N]

$I_{ij}$  The number of favorable visiting sequences containing the path from $i$ to $j$

L  Lift force [N]

N  The number of concentric expanding regions

$N_t$  The number of uniformly-distributed surveillance targets

$N_{th}$  The number of updrafts that are spatially-distributed in the surveillance area
$O_oX_1Y_1Z_1$ Vehicle-carried East-North-Up (ENU) frame

$O_oX_2Y_3Z_2$ Flight path frame

$O_oX_2Y_3Z_2$ Wind frame

$P$ Transition probability matrix in the cross-entropy method

$P_g, P_c$ UAV’s position and estimated point position

$P_i$ The cross point between the goal circular path and the vector $\vec{P_cP_g}$

$P_{ij}$ The probability that a candidate visiting sequence containing the path from Point $i$ to Point $j$ in the cross-entropy method

$Q_{ij}$ The current update transition probability in the cross-entropy method

$R$ Turning radius [m]

$R_l$ The radius of the small circle in $\{ll\}$-type or $\{lll\}$-type of Dubins path [m]

$R_i$ The radius of the $i$-th mathematical updraft model [m]

$R_{lll}$ The radius of the big circle $\{lll\}$-type of Dubins path [m]

$R_{ll}$ The radius of the big circle $\{ll\}$-type of Dubins path [m]

$S$ The UAV’s wing area [$m^2$]

$S(x)$ The cost of a candidate visiting sequence $x$ [m]

$T_{i,j}$ Lagrange interpolating polynomials

$V$ Airspeed [m/s]

$V_1$ Lyapunov function

$V_g$ Ground speed [m/s]

$W_i$ The central strength of the $i$-th mathematical updraft model [m/s]

$W_x, W_y, W_z$ Wind speed on $x, y, z$ axis in the vehicle-carried ENU frame [m/s]
$X_{i,j}^b$ p interpolation points on the boundary between regions $R_{i,j}$ in Gaussian Progress regression

$X_j$ The locations of observations in Region j

$Y$ Side force [N]

$\bar{M}$ The number of vertical wind speed measurements for Gaussian Process regression

$\bar{f}(x^*)$ The vertical wind speed estimation at location $x^*$

$\bar{l}$ Iteration number in the cross-entropy method

$\frac{dW_x}{dt}, \frac{dW_y}{dt}, \frac{dW_z}{dt}$ Wind gradient on $x, y, z$ axis in the vehicle-carried ENU frame [m/s$^2$]

$\hat{W}_x, \hat{W}_y, \hat{W}_z$ Wind speed measurements [m/s]

$\hat{W}_{zi}$ i-th vertical wind measurements [m/s]

$A_a$ Aerodynamic vector in the wind frame [N]

$A_f$ Aerodynamic vector in the flight path frame [N]

$G_I$ Gravity vector in the vehicle-carried ENU frame [N]

$G_f$ Gravity vector in the flight path frame [N]

$K$ Covariance matrix in Gaussian Process regression

$Q$ State weight matrix

$R$ Control weight matrix

$V_a$ Aircraft’s air-relative velocity [m/s]

$V_g$ Aircraft’s ground-relative velocity [m/s]

$W(X,Y)$ The simulated wind field database

$W$ Wind velocity [m/s]

$X_o$ q uniformly-distributed points on the boundary $R_{i,j}$ in Gaussian Progress regression

$a$ Aircraft’s ground-relative acceleration m/s$^2$
<table>
<thead>
<tr>
<th>Symbol</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>(d)</td>
<td>The coefficient of the linear combination of (\bar{M}) vertical wind speed measurements</td>
</tr>
<tr>
<td>(k)</td>
<td>Covariance vector in Gaussian Process regression</td>
</tr>
<tr>
<td>(x)</td>
<td>A candidate visiting sequence in a traveling salesman problem</td>
</tr>
<tr>
<td>(x^*)</td>
<td>The optimal visiting sequence resulting in the shortest traveling distance in a traveling salesman problem</td>
</tr>
<tr>
<td>(\tilde{n}_1)</td>
<td>The number of favorable visiting sequences</td>
</tr>
<tr>
<td>(b)</td>
<td>The UAV’s wing span [(m)]</td>
</tr>
<tr>
<td>(c_{ij})</td>
<td>The Euclidean distance between Point (i) and Point (j) [(m)]</td>
</tr>
<tr>
<td>(e)</td>
<td>Oswald’s efficiency factor</td>
</tr>
<tr>
<td>(e_s)</td>
<td>UAV’s specific energy (energy height) [(m)]</td>
</tr>
<tr>
<td>(k)</td>
<td>The covariance value between two vertical wind speed measurements</td>
</tr>
<tr>
<td>(m)</td>
<td>UAV’s mass [(kg)]</td>
</tr>
<tr>
<td>(n, \bar{n})</td>
<td>The size of sample set and its subset</td>
</tr>
<tr>
<td>(p)</td>
<td>Roll rate [(\text{radians/s})]</td>
</tr>
<tr>
<td>(q)</td>
<td>Pitch rate [(\text{radians/s})]</td>
</tr>
<tr>
<td>(r)</td>
<td>Yaw rate [(\text{radians/s})]</td>
</tr>
<tr>
<td>(r_c)</td>
<td>The radius of the goal circular path in the vector-based updraft soaring strategy [(m)]</td>
</tr>
<tr>
<td>(x, y, z)</td>
<td>Three dimensional position of the UAV [(m)]</td>
</tr>
<tr>
<td>(z_T)</td>
<td>The altitude at which the wind gradient becomes zero [(m)]</td>
</tr>
<tr>
<td>(z_{tr})</td>
<td>The characteristic altitude in the wind gradient model [(m)]</td>
</tr>
<tr>
<td>(\bar{l})</td>
<td>Roll moment [(kg\cdot m^2)]</td>
</tr>
<tr>
<td>(\bar{m})</td>
<td>Pitch moment [(kg\cdot m^2)]</td>
</tr>
<tr>
<td>(\bar{n})</td>
<td>Yaw moment [(kg\cdot m^2)]</td>
</tr>
</tbody>
</table>
$f(x^*)$ The true value of vertical wind speed at location $x^*$

$(x_1, y_1)(t_0)$ The updraft’s center at initial time $t_0$ [m]

$\alpha$ Angle of attack [radians]

$\alpha_T$ The angle between two circles $O_1$ and $O_2$ in ll-type of Dubins path [radians]

$\alpha_v$ The angle of view of a vision-based on-board sensor [degree]

$\bar{\alpha}$ The weight parameter which provides the weighted sum of the current transition probability $Q_{ij}$ and the previous transition probability $P_{ij}$

$\bar{\chi}$ The set that generated by the cross-entropy method, it is the subset of the sample set $\chi$

$\bar{\gamma}$ The cut-off value which choose favorable visiting sequences in the cross-entropy method

$\bar{\rho}$ The parameter which determines the number of favorable visiting sequences $\bar{\rho}$: $\bar{n}_1 = \bar{\rho} \cdot \bar{n}$

$\beta$ Side-slip attack [radians]

$\beta_{tr}$ The wind gradient slope at the reference altitude [1/s]

$\chi$ The sample set that includes all possible visiting sequences in a TSP

$\delta_a$ Aileron deflection [radians]

$\delta_c$ Elevator deflection [radians]

$\delta_r$ Rudder deflection [radians]

$\epsilon$ Convergence criteria in the cross-entropy method

$\gamma$ Flight path angle: the angle between horizontal the airspeed vector [radians]

$\lambda$ The direction of the vector $\vec{P_cP_g}$[radians]

$\mu$ Bank angle: a rotation of the lift force around the airspeed vector [degree]

$\omega$ Angular rate of the flight path frame rotating from the vehicle-carried East-North-Up (ENU) frame

$\Psi$ Heading angle: the angle between north and the horizontal component of the airspeed vector [radians]
\( \rho \)  Atmospheric density \([kg/m^3]\)

\( \sigma \)  The difference between ground heading angle \( \theta \) and the heading angle \( \varphi \) [radians]

\( \sigma_{f,l} \)  Hyper-parameters in the covariance function of Gaussian Progress regression

\( \sigma_n \)  Measurements noise covariance in Gaussian Process regression

\( \theta \)  Ground speed heading angle: the angle between north and the horizontal component of the ground speed vector [radians]

\( \theta^d \)  The desired heading command [radians]

\( \varphi \)  New-defined heading angle: \( \varphi = -\Psi \) [radians]

\( \{l\} \)  Left turning circle

\( \{r\} \)  Right turning circle

\( \{s\} \)  Straight segment

\( a_o \)  The oscillation amplitude [m]

\( b_o \)  The oscillation frequency [rad/s]

\( t \)  Time [s]
Chapter 1

Introduction

1.1 Motivation

An unmanned aerial vehicle (UAV) is defined as an aircraft which can either be remotely piloted or automatically controlled by an on-board computer. Without a human pilot aboard allows the UAV to perform repetitive and dangerous tasks such as aerial surveillance, reconnaissance and forest fire monitoring. UAVs, particularly small-scale ones whose wingspans range from 1 to 4 meters \[1\], play an active role in the development of a low-cost and low-altitude autonomous aerial surveillance platform that carries sensors, cameras and other equipment that is capable of collecting valuable information for the purpose of military, civil, or scientific research. As such, small UAVs have attracted increasing interests in recent years \[2\].

Aerial surveillance requires a UAV to stay aloft as long as possible or travel long distances to collect up-to-date information from selected targets. The key performance factors of an aerial surveillance operation are endurance and range. They are restricted by the limitation of on-board energy capacity. The advanced design of the aerodynamic shape and engine can improve fuel efficiency and result in an increase in endurance and range. Alternatively, a small UAV may improve performance by exploiting available natural energy sources such as winds.

A popular research topic for UAVs is autonomous soaring, a process of harvesting energy from the wind. The soaring behavior, by which birds can keep airborne without flapping their wings, is mimicked by a UAV with similar weight and size to exploit various wind patterns. Soaring falls into two categories based on how different wind patterns are exploited: static soaring (Fig. 1.1) and dynamic soaring (Fig. 1.2). Static soaring refers to steady flight loitering around updrafts to gain potential energy \[3, 4\], whereas dynamic soaring refers to spatial maneuvers
that utilize wind gradients to harvest additional energy [5, 6, 7, 8, 9].

![Static Soaring](image1.png)  ![Dynamic Soaring](image2.png)

**Figure 1.1: Static soaring: riding updrafts high in the air**  
**Figure 1.2: Dynamic soaring: exploiting wind gradients near sea surface**

In order to improve the performance of aerial surveillance, this thesis proposes the integration of soaring and surveillance, which takes advantage of complementary features between the two. Soaring allows for surveillance performance improvement in terms of extending endurance or range, while surveillance enables searching, identifying, and utilizing soaring sources during flight.

1.2 Related work

1.2.1 Autonomous soaring

**Static soaring**

Autonomous static soaring approaches consider the problem of finding a path that terminates in loitering around updrafts, which are large columns of rising warm air in the mixed layer [10] due to convective circulation. With the aid of the upward wind, which can offset the gliding sink rate, the aircraft may climb (gain potential energy) without consuming on-board energy, so as to extend the flight duration. A heuristic-based strategy [11] has been used by glider pilots to fly around an updraft. In this strategy, the pilots make wide turns when experiencing a fast climb or tight turns when the climb slowing down. The heuristic strategy was extended for UAV’s
static soaring by optimizing the turning radius based on the atmospheric lift measurements via reinforcement learning algorithms \cite{12,13} in order to achieve the maximum climb rate. The optimal turn can also be obtained by iteratively estimating the direction of the updraft center via the simultaneous perturbation stochastic approximation algorithm \cite{14}, the parameter optimization method \cite{15} or the extremum sinking control method \cite{16}. Learning or optimization-based algorithms \cite{12,13,14,15} are usually computationally intensive, particularly for real-time implementation \cite{17}. Allen \cite{17} proposed to use a history of the UAV’s positions and total energy to estimate the location of an updraft based on the statistical updraft model that was developed based on balloon and surface wind speed measurements. A turning radius was then determined to allow for loiter flight around the estimated updraft. The turning radius, calculated by Allen’s loitering algorithm, may not be the optimal one that leads to the maximum climb rate, nevertheless, the simulations \cite{17} and flight tests \cite{18} demonstrated significant energy-harvesting results. Allen \cite{19} also established the quantitative relationships between parameter-selection in the statistical model and the amount of UAV’s endurance extension. The loitering algorithm \cite{17}, which has been demonstrated by flight tests, is modified in this thesis to combine updraft soaring with the aerial surveillance mission to achieve endurance or range improvement.

Another trend in static soaring considers the problem of planning an energy-efficient trajectory from an initial to a goal position. Energy-efficiency implies that the UAV can harvest energy from the vertical component of a wind field so as to conserve on-board energy. Given a wind field, the energy-efficient trajectory can be obtained by a parameter optimization approach \cite{20}. In order to account for wind field changes or disturbances, the receding horizon strategy can be applied to solve this trajectory planning problem. For a time horizon, the optimal strategy was chosen from a predefined search space, which can be a tree-based space \cite{21}, or an optimal energy map \cite{22,23}, or computed by numerical optimization algorithms \cite{24,25}. However, the challenge in using receding horizon approaches lies in finding the correct time horizon. Bower \cite{26} applied the value iteration method to estimate the specific energy rate within the wind field. The method, which included short and long term memory synthesizing of recent and past sensor measurements, can provide the optimal trajectory that could extract the maximum energy from a dense updraft field. Nachmani \cite{27} considered future events by statistical analysis. Based on the statistical knowledge of the wind field, the approximate dynamic programming algorithm was used to design the energy-efficient trajectory, which was then adjusted by in-situ wind measurements. These approaches \cite{20,21,22,23,24,26,27,25} addressed the problem
of two-point trajectory planning in an updraft field. In a surveillance mission, the UAV needs to visit multiple targets to collect information. In order to implement static soaring in the surveillance mission, this thesis proposes the static soaring surveillance approach by which the UAV can visit multiple locations while identifying and harvesting energy from soaring sources.

**Updraft modeling and identification**

Static soaring depends on upward movements of the air (updrafts). There are three types of updrafts in the atmosphere [22]: the uneven ground temperature distribution producing buoyant air masses known as thermals, long period atmospheric oscillations, and orthographic (slope or ridge) lift. This thesis studies the problem of extracting energy from thermally-driven updrafts during surveillance.

In order to investigate the problem of static soaring, an appropriate updraft model is needed. According to atmospheric studies [28, 29, 30], which described the structure and behavior of the thermally-driven updraft in the convective layer of the atmosphere, various generic models were proposed for the study of UAV’s autonomous soaring. In terms of a mathematical model, Wharington [12] utilized the Gaussian function to describe the decreasing wind speed profile away from the updraft center. In terms of an empirical model, Allen [31] proposed the statistical updraft model which was developed based on balloon and surface wind speed measurements. Similarly, Childress [32] built an empirical updraft model by analyzing the flight data log of a manned glider. These models, including the mathematical model and empirical model, cannot describe the downward wind profile in the neighborhood of an updraft. Gedeon [33] extended the Gaussian model by adding the downward wind feature into the Gaussian function. Since Gedeon’s model has been widely utilized in the static soaring study [20, 25, 34], the current work utilizes this model to demonstrate the updraft exploitation strategy.

In order to investigate the problem of energy-efficient surveillance planning, an appropriate wind field model is needed. A wind field can be generated by randomly selecting the parameters of an updraft model to define the updraft’s number, location and strength within a specified area [25, 34]. It can also be generated using the numerical weather prediction system (WRF-ARW) to simulate the ridge lift and mountain wave in the real world [23]. In the current work, the wind field is generated by the large-eddy-simulation module in the numerical weather prediction system (WRF-ARW) to simulate the development of thermally-driven updraft in the convective layer. The large-eddy-simulation-based wind field is utilized as a test case to
demonstrate the effectiveness of the proposed static soaring surveillance approach.

Updrafts are transparent finite-size air masses floating in the atmosphere, whose strength and location are difficult to predict. To fulfill the desire of identifying updrafts in random locations, many studies were performed in recent years. Updraft identification includes the first step of sampling the vertical wind speed and the second step of predicting the wind speed profile via state estimation algorithms.

In terms of wind speed sampling, Langelaan [35] and Pachter [36] utilized an on-board suite (e.g. global positioning system (GPS), inertial system, and Pitot tube) to sample local wind speed and gradients. Rodriguez [37] incorporated data from optical flow with GPS and air data in order to calculate local wind speeds. Myschik [38] and Sachs [39] integrated wind measurements to the navigation system, which can be implemented on-board. Later on, in order to avoid weight and drag penalties associated with the typical air data sensing systems, a small, low-cost air data sensing system [40] utilized pressure sensors to measure airspeed, angle of attack, and angle of sideslip [40]. Current wind speed sampling approaches [35, 36, 37, 38, 39] computed wind speed from UAV’s ground-relative velocity and airspeed. In this thesis, the local wind speed sample is assumed to be available. Instead of studying the wind sampling problem, this thesis focuses on how to generate a local wind field based on available wind speed samples.

In terms of wind speed estimation, both the model-based recursive estimator and the model-free regression technique can be applied to predict the neighboring wind speed distribution. The model-based recursive algorithm includes the Kalman filter [1], the particle filter [41] and the unscented Kalman filter [42]. Later on, a priori information of the region where a thermal is likely to occur was incorporated into the model-based mapping algorithm [16] to improve wind speed estimation in stochastic environments. The inaccuracy of the model in the model-based estimation algorithm may have a negative impact on the estimation. The model-free regression technique relies on statistical approaches to extract the wind-speed patterns from on-board wind measurements. For instance, Lawrance [43] utilized Gaussian process regression to create an estimation of the wind speed map based on wind measurements at multiple locations. The Gaussian process regression method [44] was further extended by introducing the temporal component which accounted for drifting and variation in the wind field. Singh [45] incorporated the temporal component into the Gaussian regression strategy to obtain a temporal wind speed map. Lee [46] utilized a neutral network regression approach to obtain a wind speed map of the field. Model-free regression techniques [43, 45, 46], which avoid incorrect inference from an
imperfect model [1, 41, 42], can be applied to estimate updrafts’ profile by correlating wind measurements in the area. However, the current approaches [43, 45, 46] face computational challenges when they are utilized to identify updrafts in a large-size field. This thesis addresses the computational issue by dividing the field into a number of concentric expanding regions, and investigates the problem of inconsistent estimation results on the boundary between the original and expanded area.

**Dynamic soaring**

Dynamic soaring is a flight strategy that utilizes the wind gradient, a change of horizontal wind speed in a height strip, to harvest additional energy. The horizontal wind speed variation (wind gradient) is usually caused by friction drag along the ground. According to atmospheric science, the wind gradient can be expressed as a logarithmic, exponential, or linear profile [10, 47].

Previous studies focused on the mechanics of energy transmission from the gradient wind into flying birds. Based on observations of soaring birds, Rayleigh [48] first proposed the idea that birds could extract energy from gradient winds by flying in an inclined circle, along which birds climbed up in headwind and dived back in leeward. Later, Sachs [8] analyzed the measured soaring trajectories of birds using a GPS-based tracking method, and demonstrated the energy gain was achieved by a periodic headwind-climb-leeward-dive curve. Furthermore, aerodynamic studies [49, 50, 51] showed that the lift acting on the aircraft was affected by the wind speed change during these climb and dive maneuvers. By adjusting the flight path relative to the wind, the lift can be tilted forward. The forward-tilted lift acts like a thrust force so that the total energy of the aircraft increases. An analytical solution [7] to the aerodynamic equations provided the amount of energy that can be extracted from Rayleigh’s inclined circle flight. In this thesis, the periodic headwind-climb-leeward-dive curve [48, 8] is incorporated into a type of Dubins’ path [52] to determine the dynamic soaring trajectory along which the UAV soars from one target to another to perform surveillance tasks.

The problem of dynamic soaring for a UAV is to find a flight maneuver with the optimal adjustment relative to the wind in order to achieve the goal of the maximum energy-harvesting. The optimum adjustment can be obtained by searching in a given space via a reinforcement learning [53] or genetic algorithm [54]. The flight maneuver can also be developed by a set of Takagi-Sugeno-Kang fuzzy rules [55], in which the parameters are optimized by the evolutionary algorithm to achieve the best wind adjustment. The maneuver optimization
problem can also be solved by Pontryagin’s minimum principle [56, 57] or by the collocation approach [58, 59, 60, 51, 61]. The minimum principle [56, 57] formulated the optimization problem as a boundary value problem, which can be solved by the multiple shooting method numerically. The collocation approaches [58, 59, 60, 51, 61] converted the maneuver optimization problem into a parameter optimization problem, which can be solved by a standard non-linear optimization software (e.g. SNOPT). However, computational challenges arise from these large-scale parameter optimization problems. Furthermore, these studies [56, 57, 58, 59, 60, 51, 61] did not consider the problem of planning an energy-efficient trajectory from an initial position to a goal position in a wind gradient field. The trajectory planning problem is critical in the soaring surveillance study, as the UAV has to fly from one target to another to perform surveillance tasks. In order to address this trajectory planning problem and computational concerns, a type of Dubins’ path [52], a continuously differentiable curve, is utilized in this thesis to connect every two surveillance targets on the reference plane. The energy efficiency is then achieved by optimizing headwind-climb-leeward-dive maneuvers along the Dubins’ path. This approach converts the three-dimensional optimization problem [58, 59] into a height optimization problem, further reducing the computational complexity.

1.2.2 Trajectory planning and control in aerial surveillance

Trajectory planning

The surveillance trajectory planning problem is defined as such: the UAV is required to visit as many points as possible within a certain time frame or on-board energy limit. Therefore, the problem is usually formulated as a traveling distance or time minimization problem.

The shortest path planning problem can be solved by studying the corresponding traveling salesman problem (TSP). The multiple points are treated as cities and the optimal visiting order can be determined by solving the TSP. In order to take account of the bounded curvature constraint of the vehicle, Savla [62] proposed to utilize the shortest Dubins’ path [52], a continuously differentiable curve connecting two points, to replace some edges in the solution to the TSP. McGee [63] extended Savla’s results in the presence of constant horizontal winds. In addition to the TSP, Ceccarelli [64] investigated the problem of visiting multiple ground targets with desired azimuthal viewing angles in the presence of constant winds. Ousingsawat [65] proposed a path planning method for the UAV surveillance problem. The method utilized
Chapter 1. Introduction

the concept of entropy which represents the priority of each part in the area to develop the surveillance path in order to achieve the maximum coverage within the area. Osborne [66] calculated the proximity distance, which ensured smooth coverage of multiple points.

In the previous work [62, 65, 66], the aerial surveillance problem primarily focused on planning a shortest surveillance path to visit predefined locations. There is a lack of research toward aerial surveillance using soaring capable UAVs. Recent studies [67, 68, 16, 69] seek to enable long-durance surveillance via atmospheric energy harvesting using a flock of small UAVs. In the multiple UAVs cooperation soaring and surveillance study, at least one vehicle performs the surveillance task, while the remaining aircrafts explore and exploit the neighboring environment. In this thesis, we investigate the problem of optimal trajectory planning of one small UAV performing surveillance while exploring the opportunity of using soaring maneuvers to improve flight endurance performance. The challenge of soaring surveillance using one UAV lies in how to balance the process of region exploring (surveillance) and energy exploitation (soaring). By studying the mechanics of soaring flight and the requirements for aerial surveillance, the potential synergies between soaring and surveillance can be proposed.

Trajectory tracking control

The ability to track the desired soaring trajectory has an impact on actual energy exploitation [25]. Langelaan [25] addressed the tracking problem via a linear quadratic regulator (LQR) controller based on a linearized longitudinal model of the vehicle. Based on the similar model, Kahveci [70] proposed an adaptive linear controller to improve static soaring trajectory tracking performance. Kyle [71] adopted the nonlinear model predictive method to control the heading and pitch angles of the aircraft to track a static soaring trajectory. Previous studies focused on the static soaring tracking problem. In contrast to static soaring, severe maneuvers, which bring nonlinearity and coupling issues, appear in dynamic soaring flight. Furthermore, the dynamic soaring trajectory has to be followed precisely so as to achieve the desired amount of energy extraction. However, an overemphasis on tracking performance may lead to saturated control inputs. In this thesis, a nonlinear soaring trajectory tracking controller is designed based on the six-degree-of-freedom nonlinear flight dynamics model in order to address the coupling issue associated with severe maneuvers. The soaring trajectory tracking problem is essentially a nonlinear optimal control problem, which can be addressed by solving the Hamilton-Jacobi-Bellman equation that is a first-order partial differential equation. For the nonlinear system, the solution to the partial
Chapter 1. Introduction

differential equation is difficult to obtain [72]. Researchers seek alternative methods which can avoid the HJB solution to yield a stable, optimal, robust, and computationally-feasible control law for the nonlinear system. Among the alternative methods, feedback linearization [73], control Lyapunov function [74], and recursive backstepping [75] are popular trends for solving the nonlinear control problem. However, according to the benchmark test results from a previous workshop on nonlinear control, the performance of these alternative methods ranged from near optimal to very poor for various types of problems [76]. The state-dependent Riccati equation (SDRE) method [72], which is one of the alternatives, is capable of providing promising performance for different types of problems [76]. In this thesis, the state-dependent Riccati equation (SDRE) method, which can address the nonlinearity issue and achieve the best trade-off between tracking performance and achievable control inputs [72, 77], is utilized to design the tracking controller. By this effort, the desired amount of energy extraction from the wind can be fulfilled.

1.3 Thesis contributions

The main contribution of this thesis is the integration of soaring and surveillance to enhance range or endurance performance by extracting extra energy through soaring, and to improve soaring performance by increasing the accuracy of soaring source identification through extensive exploration and wind speed sampling. The integration is further enabled through innovative design of soaring and surveillance algorithms to assure complementary and smooth transition between the two. Specifically, the novel design of these algorithms are highlighted as follows.

1. In static soaring surveillance, the UAV performs surveillance while identifying spatially-distributed updrafts in a region via gliding flight, and switches into soaring mode to collect extra energy from identified updrafts. Surveillance performance is enhanced in the way of concentric expanding regions. A collection of exploration points are strategically selected to increase the range of exploration and the scale of wind speed sampling, and thus enhance the accuracy of soaring source identification. The Gaussian process regression based algorithm is proposed to achieve computationally-efficient and smooth estimation in a series of concentric regions, and to support incremental surveillance task.

2. In dynamic soaring surveillance, the UAV performs one cycle of dynamic soaring to harvest energy from the horizontal wind gradient to complete one surveillance task by
visiting from one target to the next one. In order to achieve a smooth transition between surveillance tasks, each soaring surveillance cycle is designed to start and finish with the same orientations. The Dubins’ path, which utilizes circular arcs to connect two targets with same orientations, determines the longitudinal and lateral motions on the reference plane to accomplish the surveillance task. The vertical motion (height profile) along the Dubins’ path is optimized to achieve the goal of energy-harvesting. The Dubins-path-based trajectory planning approach converts the 3D trajectory planning problem into a 1D (height) optimization problem, further reducing computational complexity.

3. The more accurately the soaring trajectory is followed, the better energy exploitation that can be achieved. The nonlinear trajectory tracking controller is designed for a full six-degree-of-freedom nonlinear UAV dynamics model to address the nonlinearity and coupling issues associated with maneuvers in soaring flight. The state-dependent Riccati equation (SDRE) method is applied to achieve the best compromise between tracking performance and achievable control inputs. Extensive simulations are carried to demonstrate the effectiveness of the proposed soaring and surveillance strategies.
Chapter 2

Soaring flight

Gliding is flight without the use of thrust. In the still air, gliding flight loses energy due to aerodynamic drag. However, energy may be exploited from the wind by soaring. This chapter first introduces the characteristic of the environment for soaring flight. Subsequently, the mechanics of soaring flight is explained physically and mathematically. Finally, static and dynamic soaring are analyzed.

2.1 Soaring flight environment

The small unmanned aerial vehicle utilizes soaring to gain extra energy from various wind patterns. Energy can be harvested from rising air currents (sometimes called updrafts or thermals) via static soaring, or from spatial wind variations (wind gradients) via dynamic soaring. In order to understand where and how to find favorable soaring sources, the study of the soaring flight environment (the atmosphere) is necessary.

The atmosphere can be generally divided into five layers, which are the troposphere, stratosphere, mesosphere, thermosphere and exosphere. For soaring flight, energy mainly comes from various wind patterns which occur in the troposphere. Thus, the troposphere is of most interest to the study of soaring flight.

The troposphere can extend from the surface of the earth, up to an altitude of 9 km at poles or 16 km at the equator \([10]\). The lowest level of the troposphere is called the planetary boundary layer (PBL). The rest of the air in the troposphere is called the free atmosphere region (Fig. 2.1).

The depth of the PBL can be as low as 100 m during the night at poles, while it can rise up
to several kilometers near the equator during the day [10]. In the current work, surveillance takes place during the day in the middle latitude area. As a result, the depth of the PBL over the surveillance area is assumed to be 1 km.

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{figure1.png}
\caption{Troposphere structure}
\end{figure}

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{figure2.png}
\caption{Planetary boundary layer (PBL) structure}
\end{figure}

\section{Wind models in the planetary boundary layer}

The planetary boundary layer can be further categorized into three layers (as shown in Fig. 2.2) by the corresponding dominant wind pattern. The lowest level of the planetary boundary layer (the bottom 5 - 10\% of the PBL [10]) is the surface layer, where the major wind pattern is wind shear (sometimes called wind gradients). The middle layer of the planetary boundary layer is the mixed layer (sometimes called the convective layer). Convective wind patterns such as large diameter updrafts or thermal vortices are dominate in the mixed layer. The top of the planetary boundary layer is the entrainment zone which occupies in the top 10\% of the planetary boundary layer. In the entrainment zone, we usually can find overshoot thermals, turbulence, and clouds [10].

\subsection{The wind gradient model in the surface layer}

In the surface layer of the atmosphere, the dominant wind pattern is the gradient wind which refers to the horizontal wind speed variation along with the height. The wind speed usually slows down close to the ground because of the frictional force of the ground, while increases with height due to the gradient pressure force. The wind gradient profile $W_x(z)$, the horizontal wind speed function of altitude $z$, is described by Eq. 2.1 [58]:
Here $\frac{dW_x}{dz}(z)$ is the wind gradient, $A$ describes the exponential ($A < 1$) or logarithmic-like ($1 < A < 2$) profile over altitudes. $\beta_{tr}$ is the wind gradient slope at the characteristic altitude $z_{tr}$, $z_T = -Az_{tr}^2(1 - A)$ is the altitude where the wind gradient becomes zero.

Figure 2.3 gives a numerical example of the wind gradient with the following specific values ($A = 1.2, \beta_{tr} = 0.35, z_{tr} = 30\ m$.) In this numerical example, $A = 1.2$ defines a logarithmic-like wind speed variation profile, which is consist with the log wind profile statement in the atmospheric study [10]. According to Eq. 2.1, the characteristic altitude $z_{tr} = 30\ m$ determines the zero wind gradient at the altitude $z_T = -A_{z_{tr}}^2 = 90\ m$. $\beta_{tr} = 0.35\ s^{-1}$ defines a 10.5 m/s horizontal wind at the characteristic altitude $z_{tr} = 30\ m$, providing a wind gradient profile within the surface layer (Fig. 2.3). In the thesis, the wind gradient profile (Eq. 2.1) is utilized as the test case to demonstrate the effectiveness of the proposed dynamic soaring surveillance approach.

Remark. $\beta_{tr} = 0.35\ s^{-1}$ indicates a 10.5 m/s wind at the characteristic altitude $z_{tr} = 30\ m$. It is a case of strong wind shear. Later, Chapter 5 will discuss the impact of the wind shear strength on the amount of energy-gain in dynamic soaring flight.

2.2.2 Updraft models in the mixed layer

Updrafts are large columns of rising warm air in the mixed layer [10] due to convective circulation, which is caused by the temperature difference between the ground and the surrounding atmosphere. Hot spots such as towns, parking lots, ploughed fields, or heath fires are typical thermal sources for driving updrafts. Those hot spots on the ground increase the temperature of the surrounding atmosphere. Warm air tends to ascend since it is lighter than its surrounding cool air. As the warm air rises, it mixes with the neighboring air masses and grows gradually [78].

As shown in Fig. 2.4, one type of thermal is continuous updrafts like a plume of air rising up from the hot spot on the ground. The rising air current may tilt from or oscillate around the hot origin depending on the wind currents. Another type is like bubbles floating in the atmosphere. The bubble can rise up to the top of the PBL and drift with prevailing winds. The buoyant
warm core (soarable lift) in thermals is the energy source for static soaring.

The mathematical updraft model

Soarable lift distributes radially around the center of the updraft. The strength of the lift decays from the maximum at the center to minimum at the certain distance away from the center. Gedeon [33] proposed a two-dimensional updraft model (Eq. 2.2) to describe the upward and downward flow of winds in the internal air circulation of a thermal bubble. Since Gedeon’s model (Eq. 2.2) has been widely utilized in the static soaring study [20, 25, 34], the current work utilizes this model to demonstrate the updraft exploitation strategy.

$$W_z(x, y) = W_1 e^{-rac{(x-x_1)^2+(y-y_1)^2}{(R_1)^2}} \left[ 1 - \frac{(x-x_1)^2+(y-y_1)^2}{(R_1)^2} \right]$$

(2.2)

Here $W_z$ represents the vertical wind speed at position $(x, y)$. $W_1$ is the vertical wind speed at the updraft’s center $(x_1, y_1)$. $R_1$ is the radius of the updraft. Figure 2.5 shows a numerical example of updraft whose center $(x_1, y_1) = (0, 0)$ m, radius $R_1 = 50$ m and center strength $W_1 = 10$ m/s.
Figure 2.4: A plume of rising air and updraft bubbles in the PBL

Figure 2.5: The updraft model whose center \((x_1, y_1) = (0, 0)\) m, radius \(R_1 = 50\) m and center strength \(W_1 = 10\) m/s

**Wind field: updraft simulation database**

Updrafts occur as part of thermal convection, which is caused by the temperature difference between the ground and the atmosphere. The numerical weather prediction system (WRF-ARW) provides a module to describe buoyant flow turbulence in the atmosphere using the turbulence-resolving method (Large Eddy Simulation). Further information about the WRF-ARW thermal simulation module can be found in reference [79]. By setting the temperature difference, which is the driving force for thermal turbulence, and defining the size of the area, the numerical weather prediction system can simulate the temperature variation and convective flows associated with turbulence. After the equilibrium is reached, the simulated wind field with
spatially-distributed updrafts can be generated (Fig. 2.6). Since the simulated updraft model mimics the process of thermally-driven updraft development in the convective mixed layer, it is more representative than the mathematical model (Eq. 2.2). In the thesis, the simulated updraft field is utilized as the test case to demonstrate the effectiveness of the proposed soaring surveillance approach.

![Simulated wind field generated from the weather prediction system](image)

Figure 2.6: Simulated wind field generated from the weather prediction system

### 2.3 Equations of motion derivation in the presence of winds

Soaring is a flight strategy by which energy can be extracted from the wind. This section presents an analysis and discussion of the equations of motion for a small UAV in the presence of winds. The objective of this section is to provide the mechanics of energy transmission in soaring flight.

#### 2.3.1 Frames of reference

Three right-handed reference frames (as shown in Fig. 2.7) are provided to define the dynamics of the small UAV.

The first is the vehicle-carried east-north-up (ENU) frame ($O_oX_1Y_1Z_1$). The frame has
origin $O_o$ at the center of the mass of the aircraft and three axes ($X_1, Y_1, Z_1$) are aligned east, north, and upward (perpendicular to the ground, pointing upward) respectively.

The second frame is the flight-path frame ($O_oX_2Y_2Z_2$). The flight path frame has origin $O_o$ fixed to the vehicle at the mass center of the aircraft. The $y$ axis ($Y_3$) points in the direction of the airspeed $V$. The $z$ axis $Z_2$ lies in the vertical plane ($Y_3 - Z_2$), which is perpendicular to the ground. The $x$ axis $X_2$ points to the right of $y$ axis.

The third frame is the wind frame ($O_oX_3Y_3Z_3$). The wind frame has origin fixed to the vehicle at the mass center of the aircraft. The $y$ axis ($Y_3$) is aligned along the direction of the airspeed $V$. The $z$ axis ($Z_3$) lies in the plane of symmetry of the aircraft pointing upwards. The $x$ axis ($X_3$) points to the right of $y$ axis.

![Figure 2.7: Rotation relationship among the vehicle-carried east-north-up (ENU), flight-path, and wind reference frames](image)

From the vehicle-carried east-north-up (ENU) frame $O_oX_1Y_1Z_1$, the wind frame $O_oX_3Y_3Z_3$ can be obtained by a sequence of rotations (Fig. 2.7).

1. A rotation $\Psi$ about $O_oZ_1$, carrying the vehicle-carried $O_oX_1Y_1Z_1$ to $O_oX_2Y_2Z_1$. The rotation matrix is

$$C_z = \begin{bmatrix}
\cos \Psi & \sin \Psi & 0 \\
-\sin \Psi & \cos \Psi & 0 \\
0 & 0 & 1
\end{bmatrix}.$$  

$\Psi$ is the true heading angle.
2. A rotation $\gamma$ about $O_oX_2$, carrying the frame $O_oX_2Y_2Z_1$ to the flight path frame $O_oX_2Y_3Z_2$. The rotation matrix is
\[
C_x = \begin{bmatrix}
1 & 0 & 0 \\
0 & \cos \gamma & \sin \gamma \\
0 & -\sin \gamma & \cos \gamma \\
\end{bmatrix}.
\]
$\gamma$ is the flight path angle.

3. A rotation $\mu$ about $O_oY_3$, carrying the flight path frame $O_oX_2Y_3Z_2$ to the wind frame $O_oX_3Y_3Z_3$. The rotation matrix is
\[
C_y = \begin{bmatrix}
\cos \mu & 0 & -\sin \mu \\
0 & 1 & 0 \\
\sin \mu & 0 & \cos \mu \\
\end{bmatrix}.
\]
$\mu$ is the bank angle.

The sequence of the rotations is presented as:
\[
O_oX_1Y_1Z_1 \xrightarrow{C_z} O_oX_2Y_2Z_1 \xrightarrow{C_x} O_oX_2Y_3Z_2 \xrightarrow{C_y} O_oX_3Y_3Z_3 \quad (2.3)
\]

### 2.3.2 Equations of motion for a small UAV in the presence of winds

The ground-relative velocity of the aircraft $V_g$ can be written as the vector sum of the air-relative velocity $V_a$ and the wind velocity $W$.

\[
V_g = V_a + W \quad (2.4)
\]

By taking the derivatives of the both sides of Eq. 2.4 with respect to time, the ground-relative acceleration $a$ can be obtained as:

\[
a = \dot{V}_g = \dot{V}_a + \dot{W} \quad (2.5)
\]

By applying Newton’s second law in the flight-path frame ($O_oX_2Y_3Z_2$), we can have:

\[
m[a]_f = \sum F_f = G_f + A_f \quad (2.6)
\]

Here subscript $*_f$ means the flight-path frame. $G_f$ and $A_f$ represent the gravitational force ($G = mg$) and aerodynamic forces (Lift $L$, Drag $D$, Side force $Y$) in the flight-path frame. Forces $G_f$ and $A_f$ can be obtained by performing the following rotations:
\[
\mathbf{G}_f + \mathbf{A}_f = C_x C_z \begin{bmatrix}
0 \\
0 \\
-mg
\end{bmatrix} + C_y^T \begin{bmatrix}
Y \\
-D \\
L
\end{bmatrix} = \begin{bmatrix}
Y \cos \mu + L \sin \mu \\
-D - G \sin \gamma \\
L \cos \mu - G \cos \gamma - Y \sin \mu
\end{bmatrix}
\]

Here, \( C_x, C_y, \) and \( C_z \) are rotation matrices.

\( m[a]_f \) in Eq. 2.6 acting on the aircraft in the flight path frame can be presented as:

\[
m[a]_f = m(\dot{\mathbf{V}}_a)_f + \omega \times [\mathbf{V}_a]_f + m \left[ \frac{d\mathbf{W}}{dt} \right]_f \tag{2.7}
\]

Here \( [\dot{\mathbf{V}}_a]_f \) represents air-relative acceleration in the flight-path frame, \( \omega \) represents the rotation rate of the frame, \( [\mathbf{V}_a]_f \) represents air-relative velocity in the flight-path frame, and \( \left[ \frac{d\mathbf{W}}{dt} \right]_f \) can be obtained by rotating from the vehicle-carried east-north-up (ENU) frame:

\[
\left[ \frac{d\mathbf{W}}{dt} \right]_i = C_x C_z \begin{bmatrix}
d\mathbf{W}_x dt \\
d\mathbf{W}_y dt \\
d\mathbf{W}_z dt
\end{bmatrix}
\]

Here \( C_x \) and \( C_z \) are rotation matrices around \( x \)-axis and \( z \)-axis respectively. \( \left[ \frac{d\mathbf{W}}{dt} \right]_i \) represents the wind speed derivative in the vehicle-carried east-north-up frame.

By rearranging Eq. 2.7, we have:

\[
m[a]_f = m([\dot{\mathbf{V}}_a]_f + \omega \times [\mathbf{V}_a]_f) + m \left[ \frac{d\mathbf{W}}{dt} \right]_f \tag{2.8}
\]
Here, $V$ is airspeed, $\Psi$ is the true heading angle, $\gamma$ is the flight path angle, $\mu$ is bank angle. Therefore, Eq. 2.6 can be rewritten as

$$m\begin{pmatrix} -V\dot{\Psi}\cos\gamma \\ \dot{V} \\ V\dot{\gamma} \end{pmatrix} + \begin{pmatrix} \cos\Psi\frac{dW_x}{dt} + \sin\Psi\frac{dW_y}{dt} \\ -\cos\gamma\sin\Psi\frac{dW_x}{dt} + \cos\gamma\cos\Psi\frac{dW_y}{dt} + \frac{dW_z}{dt}\sin\gamma \\ \sin\gamma\sin\Psi\frac{dW_x}{dt} - \sin\gamma\cos\Psi\frac{dW_y}{dt} + \frac{dW_z}{dt}\cos\gamma \end{pmatrix} = \begin{pmatrix} Y\cos\mu + L\sin\mu \\ -D - G\sin\gamma \\ L\cos\mu - G\cos\gamma - Y\sin\mu \end{pmatrix}$$

Equation 2.9

Further, Eq. 2.4: $V_g = V_a + W$ is rewritten as:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} = (C_x C_z)^T \begin{bmatrix} 0 \\ V \\ 0 \end{bmatrix} + \begin{bmatrix} W_x \\ W_y \\ W_z \end{bmatrix}$$

Equation 2.10

Here $x, y, z$ are three-dimensional location of the UAV.

Therefore, the equations of motion for the small UAV in the presence of winds can be presented as Eq. 2.11 and Eq. 2.12.

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} = \begin{bmatrix} -V\cos\gamma\sin\Psi + W_x \\ V\cos\gamma\cos\Psi + W_y \\ V\sin\gamma + W_z \end{bmatrix}$$

Equation 2.11

$$m\begin{pmatrix} \dot{V} \\ -V\dot{\Psi}\cos\gamma \\ V\dot{\gamma} \end{pmatrix} = \begin{pmatrix} -D - G\sin\gamma + m\cos\gamma\sin\Psi\frac{dW_x}{dt} - m\cos\gamma\cos\Psi\frac{dW_y}{dt} - m\frac{dW_z}{dt}\sin\gamma \\ Y\cos\mu + L\sin\mu - m\cos\Psi\frac{dW_x}{dt} - m\sin\Psi\frac{dW_y}{dt} \\ L\cos\mu - G\cos\gamma - Y\sin\mu - m\sin\gamma\sin\Psi\frac{dW_x}{dt} + m\sin\gamma\cos\Psi\frac{dW_y}{dt} - m\frac{dW_z}{dt}\cos\gamma \end{pmatrix}$$

Equation 2.12

Remark. The relationship between $\Psi$ and $\varphi$. According to Fig. 2.7, the counterclockwise (left turn from north direction) rotation indicates a positive true heading angle $\Psi$ (as shown in Fig. 2.7). For the sake of consistency with the previous study [58], we assume $\varphi = -\Psi$. In this situation, the newly-defined heading angle $\varphi$ is negative when rotating counterclockwise. In the following chapters of the thesis, the heading angle refers to $\varphi$. 
Chapter 2. Soaring flight

Substitute \( \Psi = -\varphi \) into Eq. 2.11 and Eq. 2.12, the equations of motion for the small UAV with new-defined heading angle \( \varphi \) can be presented as:

\[
\begin{align*}
\begin{bmatrix} 
\dot{x} \\
\dot{y} \\
\dot{z}
\end{bmatrix} &= 
\begin{bmatrix}
V \cos \gamma \sin \varphi + W_x \\
V \cos \gamma \cos \varphi + W_y \\
V \sin \gamma + W_z
\end{bmatrix} \\
&= 
\begin{bmatrix}
\dot{V} \\
V \dot{\varphi} \cos \gamma \\
V \dot{\gamma}
\end{bmatrix}
\end{align*}
\]

(2.13)

\[
\begin{align*}
\begin{bmatrix}
\dot{V} \\
V \dot{\varphi} \cos \gamma \\
V \dot{\gamma}
\end{bmatrix} &= 
\begin{bmatrix}
-D - G \sin \gamma - m \cos \gamma \sin \varphi \frac{dW_x}{dt} - m \cos \varphi \cos \gamma \frac{dW_y}{dt} - m \cos \gamma \sin \gamma \\
Y \cos \mu + L \sin \mu - m \sin \varphi \cos \gamma \cos \mu \frac{dW_y}{dt} + m \sin \varphi \cos \gamma \cos \mu \frac{dW_z}{dt} - m \cos \gamma \sin \gamma \\
L \cos \mu - G \cos \gamma - Y \sin \mu + m \sin \varphi \cos \gamma \frac{dW_y}{dt} + m \sin \varphi \cos \gamma \frac{dW_z}{dt} - m \cos \gamma \cos \gamma
\end{bmatrix}
\end{align*}
\]

(2.14)

In the following chapters, Eq. 2.13 and Eq. 2.14 are used to analyze static and dynamic soaring flight.

2.4 Static soaring flight

Static soaring is a flight strategy that utilizes the vertical air motion to exploit potential energy for long-endurance flight. By loitering around the vertical air currents (e.g. updrafts), the UAV is capable of gaining height without losing airspeed.

From Eq. 2.13 and Eq. 2.14, with the effect of the wind speed \( W_x, W_y, W_z \), the equations of motion for the small UAV with zero side-slip, assuming \( \frac{dW_x}{dt} = 0, \frac{dW_y}{dt} = 0, \frac{dW_z}{dt} = 0 \), can be derived as:

\[
\begin{align*}
\dot{V} &= -\frac{D}{m} - g \sin \gamma = 0 \\
\dot{x} &= V \cos \gamma \sin \varphi + W_x \\
\dot{y} &= V \cos \gamma \cos \varphi + W_y \\
\dot{z} &= V \sin \gamma + W_z \\
\frac{mV \dot{\gamma}}{\dot{\varphi}} &= L \cos \mu - mg \cos \gamma = 0 \\
\frac{mV \cos \gamma \dot{\varphi}}{\dot{\gamma}} &= L \sin \mu
\end{align*}
\]

(2.15)

In Fig. 2.8, Case (1) illustrates the forces on the steady gliding aircraft in the calm air. When the aircraft flies in the region with constant upward-blowing winds (Fig. 2.8, Case (2)), the strong rising air can offset the sink rate of the aircraft. In this case, the vertical component of the inertial speed of the aircraft is upward (\( \dot{z} > 0 \) in Eq. 2.15). With the aid of the wind (e.g. \( W_x = 0, W_y = 0, W_z = 5 \text{ m/s} \) as shown in Fig. 2.9), the aircraft can gain altitude (potential energy) by loitering around the region. This refers to static soaring flight.
2.5 Dynamic soaring flight

In contrast to static soaring, which depends on vertical winds, dynamic soaring exploits horizontal wind gradients by maneuvering in the wind gradient region.

We consider the following wind conditions (Eq. 2.16), where $W_x$ is the horizontal wind speed (eastbound, along x-axis), and $\frac{dW_x}{dz}$ is the wind gradient.

\[
\begin{align*}
\frac{dW_x}{dt} &= \frac{dW_x}{dz} \dot{z} = \frac{dW_x}{dz} V \sin \gamma \\
\frac{dW_y}{dt} &= 0 \\
\frac{dW_z}{dt} &= 0 \\
W_x &= W_x \\
W_y &= 0 \\
W_z &= 0
\end{align*}
\] (2.16)

Substituting Eq. 2.16 into Eq. 2.13 and Eq. 2.14, the equations of motion for the small UAV with
zero side-slip and horizontal wind gradient $\frac{dW_x}{dz}$ can be derived as:

\[
\begin{align*}
\dot{V} &= -\frac{D}{m} - g \sin \gamma - \frac{dW_x}{dz} V \sin \gamma \cos \gamma \sin \phi \\
\dot{x} &= V \cos \gamma \sin \phi + W_x \\
\dot{y} &= V \cos \gamma \cos \phi \\
\dot{z} &= V \sin \gamma \\
mV \dot{\gamma} &= L \cos \mu - mg \cos \gamma + m \frac{dW_x}{dz} V \sin^2 \gamma \sin \phi \\
mV \cos \gamma \dot{\phi} &= L \sin \mu - m \frac{dW_x}{dz} V \sin \gamma \cos \phi
\end{align*}
\]

(2.17)

where $L$ and $D$ are aerodynamic lift and drag respectively, $m$ is the mass of the UAV,

The total energy of the UAV can be presented as: $E = mgz + \frac{1}{2}mV^2$. The specific energy $e_s$ can be defined as the total energy $E$ per weight:

\[
e_s = \frac{E}{mg} = z + \frac{1}{2g} V^2
\]

(2.18)

Taking the derivative of $e_s$ with respect to time, we can obtain:

\[
\dot{e}_s = -\frac{dW_x}{dz} \frac{V^2}{g} \sin \gamma \cos \gamma \sin \phi - \frac{DV}{mg}
\]

(2.19)

The first term of Eq.2.19 represents the energy from the wind gradient, and the second term of $\dot{e}_s$
represents aerodynamic drag. In order to gain energy, the first term has to be positive to compensate the drag part. Therefore, the flight path angle $\gamma$ and heading angle $\varphi$ have to satisfy the following relationship:

$$
\text{dynamic soaring rule} = \begin{cases} 
\sin \gamma \sin \varphi < 0 & \frac{dW_x}{dz} > 0; \\
\sin \gamma \sin \varphi > 0 & \frac{dW_x}{dz} < 0.
\end{cases}
$$

![Figure 2.10: Forces analysis on the UAV in windward climb, dive, leeward climb, and dive in horizontal wind gradients](image)

Figure 2.10 shows the mechanics of energy transmission in dynamic soaring flight. During the climb and dive, the vehicle obtains an instantaneous change of its airspeed because of the wind gradient. In Fig. 2.10, the speed triangle describes the instantaneous speed change, where $\Delta W_x$, $W_v$, and $W$ represent the wind speed change (relative to the earth), relative wind speed (opposite to the old airspeed), and apparent wind speed (oncoming flow) respectively. Since the aerodynamic lift is orthogonal to the direction of oncoming flow (the direction of the apparent wind speed $W$), the current lift $L'$ is tilted forward as shown Case (1) (3), and backward in Case (2) and (4) in Fig. 2.10. The forward lift can provide the extra “thrust” to the aircraft. However, the backward lift plays the role of aerodynamic drag. Therefore, in order to obtain the “thrust”, the aircraft has to climb windward or dive leeward. In the case of a high $L/D$ ratio, the extra “thrust” from the instantaneous wind speed change can offset
Chapter 2. Soaring Flight

25

aerodynamic drag. This refers to dynamic soaring.

The wind gradient model (Fig. 2.3, \( A = 1.2, \beta_{tr} = 0.35, z_{tr} = 30 \text{ m} \)) is chosen to illustrate dynamic soaring flight. The UAV’s energy height increases during the windward climb and leeward dive as shown in Fig. 2.11 and Fig. 2.12.

Figure 2.13 and Fig. 2.14 show the airspeed and altitude change for the windward climb and leeward dive. Figure 2.13 and Fig. 2.14 also depict the lift \((C_L)\) and drag coefficient \((C_D)\). The drag coefficient is calculated by:

\[
C_D = C_{D0} + \frac{C_L^2}{\pi \epsilon AR}
\]  

(2.20)

Here \(C_L\) is lift coefficient, \(C_{D0}\) is the parasitic drag coefficient. \(AR\) is the aspect ratio, \(\epsilon\) is Oswald’s efficiency factor. In this thesis, we use the Aerosonde UAV model [2]. The parameters are given in Table 4.3 in Chapter 4.

\[\text{Figure 2.11: Dynamic soaring: windward climbing } (\gamma = \frac{\pi}{4}, \psi = -\frac{\pi}{2}, \dot{\gamma} = 0, \dot{\psi} = 0)\]

\[\text{Figure 2.12: Dynamic soaring: leeward diving } (\gamma = -\frac{\pi}{4}, \psi = \frac{\pi}{2}, \dot{\gamma} = 0, \dot{\psi} = 0)\]

2.6 Summary

In this chapter, the environment of the soaring flight was introduced. The mathematical foundation of soaring was provided. Based on the aerodynamic principle of soaring flight, the energy optimal trajectory was investigated. With the aid of soaring, the aircraft can take advantage of the wind instead of consuming its on-board power to fly. Since aerial surveillance faces energy shortage because of the limited energy capacity, soaring is incorporated into aerial surveillance to improve endurance performance.
Figure 2.13: Flight states along the windward climbing

Figure 2.14: Flight states along the leeward diving
Chapter 3

Soaring Surveillance Problem Formulation

This chapter formulates the soaring surveillance problem to achieve energy-efficient aerial surveillance. The aerial surveillance task is defined first, followed by soaring surveillance problem statements and formulations. By solving the proposed problem, soaring can be implemented to obtain an automatic energy refill from various wind patterns to achieve long-endurance, long-range aerial surveillance flight.

3.1 Aerial surveillance task

The surveillance area is assumed to be a flat square area. The small UAV is considered to fly over the square area within the planetary boundary layer (PBL) to harvest energy. The depth of the PBL can be as low as 100 m during night at poles, while it can rise up to several kilometers near the equator during daytime [10]. In the current work, surveillance takes place during daytime in the middle latitude area. As a result, the thickness of the PBL over the surveillance area is assumed to be 1 km.

As a result, the thickness of the PBL over the surveillance area is assumed to be 1 km.

The aerial surveillance task is to observe $N_t$ targets, which are randomly distributed in the square area. Figure 3.1 and Fig 3.2 show two surveillance cases ($N_t = 50$ and $N_t = 300$) respectively. The UAV visits these predefined targets to collect up-to-date information. Inspired by the capability of soaring to harvest extra energy from atmospheric energy sources, soaring surveillance is proposed to achieve energy-efficient, long-endurance flight.

3.2 Autonomous soaring surveillance

Static soaring surveillance can be elaborated as follows. The origin of the coordinate system (X-Y-Z) is at the center point of the surveillance area. The UAV starts to visit the predefined targets from
the central region \([-100, 100] \times [-100, 100] \text{ m}\) of the surveillance area at the top of the planetary boundary layer (PBL) (Fig. 3.3). By strategically selecting a collection of so-called exploration points in the central region, the UAV can visit surveillance targets while maximizing the region coverage to collect vertical wind speed samples. The wind speed map can be generated by processing the wind samples via the approach of Gaussian process regression. According to the map, the location with the strongest upward wind can be identified. If the speed of the upward wind exceeds UAV’s sink rate, the UAV will perform static soaring maneuvers to gain energy. The surveillance region expands concentrically serving as a spiral-type updrafts searching strategy. In the meantime, it allows the UAV to visit each surveillance target inside each concentrically-extending region by flying a Hamiltonian circuit. With the aid of Gaussian process regression with boundary constraints, the wind speed map also expands concentrically. As long as updrafts are identified, the surveillance region can keep extending.

When the UAV reaches the surface layer of the PBL, where the dominant wind pattern is the gradient wind, dynamic soaring surveillance is performed to visit the predefined targets. By designing the energy-harvesting surveillance trajectory, the small UAV is capable of soaring from one target to another. The soaring surveillance architecture is presented in Fig. 3.4.

**Remark.** Dynamic soaring may switch to static soaring when enough energy is gained from wind gradients. With the extra amount of energy, the UAV can climb up to an altitude to find an updraft and resume static soaring surveillance.
Figure 3.3: Surveillance area with coordinate system (X-Y-Z)

Figure 3.4: Soaring surveillance architecture

### 3.3 Soaring surveillance problem statements and formulations

#### 3.3.1 Soaring surveillance problem statements

Soaring is implemented in aerial surveillance to achieve the goal of long endurance therefore surveillance performance is improved.

In static soaring surveillance, we investigate the surveillance trajectory planning problem and the updraft identification problem. In the trajectory planning problem, the optimal surveillance trajectory is designed to visit as many targets as possible with the amount of initial energy. In the updraft identification problem, a wind speed map is generated in each expanding region based on the direct wind speed observation during flight. When an updraft is identified, the UAV performs static soaring to gain energy. The surveillance mission resumes when the UAV reaches the top of the thermal.

In dynamic soaring surveillance, we focus on the trajectory planning problem, by solving which the UAV can travel from one surveillance target to another with the maximum energy extraction from the wind.

After the soaring surveillance trajectory is determined, the trajectory tracking problem will be investigated. By solving the problem, the optimal tracking controller is designed to steer the UAV to follow the static or dynamic soaring surveillance trajectory, so that the desired amount of energy extraction can be achieved.

These problems are formulated in the following sections.
3.3.2 Static soaring surveillance trajectory planning problem formulation

In aerial surveillance with gliding flight, the UAV has to visit as many targets as possible with the amount of initial energy. Therefore, the optimal surveillance trajectory planning can be formulated as the traveling salesman problem, which determines the optimal visiting sequence of predefined targets, and the maximum range problem, by solving which the UAV can fly with the maximum gliding range following the optimal sequence.

**Traveling salesman problem**

The optimal visiting sequence, following which the UAV can visit predefined targets with the minimum traveling distance, can be determined by solving the traveling salesman problem (TSP).

Every pair of targets $i,j$ can be connected by an edge. The length of the edge from $i$ to $j$ represents a cost $c_{ij}$. Given our objective, the cost $c_{ij}$ (Eq. 3.1) is defined as the Euclidean distance between two targets.

$$c_{ij} = \sqrt{(P_{ix} - P_{jx})^2 + (P_{iy} - P_{jy})^2},$$  \hspace{1cm} (3.1)

where $(P_{ix}, P_{iy})$ and $(P_{jx}, P_{jy})$ are the coordinates of targets $i$ and $j$.

Let $x$ represent a candidate sequence (sometimes called a sample sequence), e.g., $x = \{x_1 \rightarrow x_2 \rightarrow \ldots \rightarrow x_n \rightarrow x_1\}$, where each element $x_i$ is the point. Then the overall cost of the tour $S$ is

$$S(x) = \sum_{i=1}^{n-1} c_{x_i, x_{i+1}} + c_{x_n, x_1}.$$  \hspace{1cm} (3.2)

where, $n$ is the number of predefined targets.

Let $\chi$ be the set of all possible visiting sequences, $\chi = \{x_k, k = 1, 2, \ldots, n!\}$, where $n! = 1 \times 2 \times 3 \cdots n$ is the total number of possible visiting sequence.

The optimal visiting sequence of targets $x^*$ minimizes the overall cost (Eq. 3.3). The minimization problem can be solved by the cross-entropy method in Chapter 4.

$$x^* = \arg \min_{x \in \chi} S(x) = \arg \min \left( \sum_{i=1}^{n-1} c_{x_i, x_{i+1}} + c_{x_n, x_1} \right).$$  \hspace{1cm} (3.3)

**Maximum gliding range problem**

The lift-to-drag ratio $\frac{C_L}{C_D}$ determines the gliding range. Following the optimal visiting sequence, the UAV performs steady gliding with the maximum range speed to visit as many points as possible with the certain amount of initial energy.

The equations of motion for the steady gliding turn (constant airspeed $V$, flight path angle $\gamma$, and bank angle $\mu$) in the calm atmosphere (assuming no winds) with zero side-slip can be derived from Eq. 2.13 and Eq. 2.14:
\[\dot{x} = V \cdot \cos \gamma \cdot \sin \varphi \]
\[\dot{y} = V \cdot \cos \gamma \cdot \cos \varphi \]
\[\dot{z} = V \cdot \sin \gamma \]
\[\dot{V} = 0 = \frac{-D}{m} - g \sin \gamma \]
\[\dot{\gamma} = 0 = \frac{L \cos \mu}{mV} - \frac{g \cos \gamma}{V} \]
\[\dot{\varphi} = \frac{L \sin \mu}{mV \cos \gamma} \]

The following optimization problem is formulated to determine the maximum lift-to-drag ratio in different turn rates \(\dot{\varphi}\): 

\[
\begin{align*}
\text{minimize} & \quad V, \gamma, C_{L}^{\min}, \mu \quad [-\tan \gamma = \frac{C_{D}}{C_{L} \cos \mu}] \\
\text{subject to} & \quad mg \cos \gamma = L \cos \mu \\
& \quad -mg \sin \gamma = D \\
& \quad L = 0.5\rho V^{2}SC_{L} \quad (3.5) \\
& \quad D = 0.5\rho V^{2}SC_{D} \\
& \quad C_{D} = C_{D_{0}} + \frac{C_{L}^{2}}{\pi eAR} \\
& \quad \dot{\varphi} = \frac{L \sin \mu}{mV \cos \gamma} \\
& \quad C_{L_{\min}} \leq C_{L} \leq C_{L_{\max}}, \quad \mu_{\min} \leq \mu \leq \mu_{\max}
\end{align*}
\]

The optimization problem is solved by both analytical and numerical approaches in Chapter 4.

### 3.3.3 Updraft identification problem formulation

The wind speed prediction \(\hat{f}(x^*)\) at any location \(x^*\) of the field can be expressed by a linear combination of vertical wind observations \(\hat{W}_{z} = \{\hat{W}_{z_{i}}\}_{i=1}^{M}\) (Eq. 3.6). The updraft is identified by locating the position where predicted wind speed is maximized.

\[
\hat{f}(x^*) = \sum_{i=1}^{M} \bar{d}_{i} \hat{W}_{z_{i}} = \bar{d}^{T} \hat{W}_{z} \quad (3.6)
\]

where \(\bar{d} = \{\bar{d}_{i}\}_{i=1}^{M}\) are the coefficients of the linear combination.

The optimal coefficient \(\bar{d}^{*} = \{\bar{d}_{i}\}_{i=1}^{M}\) can be determined by minimizing the mean squared prediction error between prediction \(\hat{f}(x^*)\) and true value \(f(x^*)\): 

\[E[(\hat{f}(x^*) - f(x^*))^{2}]\] via Gaussian process regression.
in Chapter 4.

3.3.4 Dynamic soaring surveillance trajectory planning problem formulation

In the surface layer, the dominant wind pattern is the wind gradient. Dynamic soaring surveillance is implemented to extract energy from gradient winds to achieve long-endurance flight. According to the dynamic soaring rule, the UAV exploits energy by repeatedly ascending upwind or descending downwind in the wind-gradient region as shown in Fig. 3.5.

Figure 3.5: The periodic characteristics of dynamic soaring

In dynamic soaring, the UAV follows the cyclic trajectory, one cycle of which consists of one windward climb, flight directions changes, and one leeward dive. (Fig. 3.5). Along the cyclic trajectory, the UAV’s heading angle changes periodically. Therefore, one cycle of dynamic soaring is the trajectory along which the UAV starts and finishes with the same heading angle ($\varphi_{\text{initial}} = \varphi_{\text{final}}$). Suppose that the UAV starts the dynamic soaring cycle at the bottom of the wind-gradient region ($z_{\text{initial}} = z_{\text{final}} = 0$), the flight path angles at the initial and final point of the cycle have the following constraint ($\gamma_{\text{initial}} = \gamma_{\text{final}} = 0$).

In dynamic soaring surveillance, the cycle of dynamic soaring, which starts and finishes at the bottom of the wind-gradient region ($z_{\text{initial}} = z_{\text{final}} = 0$), is utilized to connect every two surveillance targets $P(x_p, y_p), Q(x_q, y_q)$. As a result, the energy gain in the dynamic soaring cycle can be represented by the velocity gain. The dynamic soaring surveillance problem can be formulated as follows to achieve the maximum velocity gain (kinetic energy extraction),
maximize \( C_L, \mu, V(0), \phi(0), t_f \) \[ V(t_f) - V(0) \]
subject to

Equations of motion (Eq. 3.8)

\[ V_{\min} \leq V(t), \ x(0) = x_p, \ y(0) = y_p, \ z(0) = 0 \text{ m} \]
\[ x(t_f) = x_q, \ y(t_f) = y_q, \ z(t_f) = 0 \text{ m} \]
\[ \gamma(0) = \gamma(t_f) = 0 \text{ rad}, \ \phi(0) = \phi(t_f) \]
\[ C_{L_{\min}} \leq C_L \leq C_{L_{\max}}, \ \mu_{\min} \leq \mu \leq \mu_{\max} \]

The equations of motion for the small UAV with zero side-slip can be derived by substituting wind gradient conditions (Eq. 2.16 in Chapter 2) into Eq. 2.13 and Eq. 2.14

\[
\begin{align*}
\dot{V} &= -\frac{D}{m} - g \sin \gamma - \frac{dW_x}{dz} V \sin \gamma \cos \gamma \sin \varphi \\
\dot{x} &= V \cos \gamma \sin \varphi + W_x \\
\dot{y} &= V \cos \gamma \cos \varphi \\
\dot{z} &= V \sin \gamma \\
mV \dot{\gamma} &= L \cos \mu - mg \cos \gamma + m \frac{dW_x}{dz} V \sin^2 \gamma \sin \varphi \\
mV \cos \gamma \dot{\phi} &= \dot{\gamma} = L \sin \mu - m \frac{dW_x}{dz} V \sin \gamma \cos \varphi
\end{align*}
\]

The trajectory planning problem is solved by the Dubins-path-based approach.

### 3.3.5 Trajectory tracking problem formulation

Soaring requires spatial or temporal maneuvers to harvest additional kinetic or potential energy from the wind. According to the optimal trajectory planning problem formulation, the soaring trajectory is planned based on the point mass model. In fact, soaring involves maneuvers associated with flight dynamics issues. Neglecting these issues may cause unsatisfactory tracking results. In order to address these issues, the soaring trajectory tracking problem is formulated as a nonlinear optimal control problem.

The optimal control law \( u \) is applied to achieve the best compromise between tracking performance and achievable control inputs. As a result, the desired amount of energy extraction can be achieved.
Figure 3.6 shows the relationship between trajectory planning and tracking controller.

![Figure 3.6: Trajectory planning and control diagram](image)

The trajectory tracking problem is solved by the state-dependent Riccati equation approach. This approach, as a nonlinear feedback controller, has inherent robustness with respect to uncertainty and disturbance [72].

### 3.4 Summary

This chapter defined the aerial surveillance task. In order to achieve long-endurance surveillance, soaring was implemented during surveillance to harvest extra energy from the atmosphere. According to different types of wind energy exploitation, static and dynamic soaring surveillance problems were formulated respectively. In order to track the desired trajectory without violating the limits of control surfaces, the trajectory tracking problem was formulated. The proposed soaring surveillance trajectory planning and control problems in this chapter will be addressed in detail in the following chapters.
Chapter 4

Static soaring surveillance in the quasi-static updraft field

Static soaring surveillance is the flight approach that exploits updrafts to gain potential energy to achieve energy-efficient surveillance. Static soaring surveillance is implemented when the UAV flies in the convective layer of the PBL where abundant updrafts, particularly on warm sunny days, are present because of air convection. In the quasi-static field, the updraft is stationary or oscillating around the hot origin. According to the characteristics of the quasi-static field, the soaring surveillance approach is proposed. The approach is further demonstrated by simulated updraft fields.

4.1 Surveillance and exploring points in the surveillance area

4.1.1 Surveillance and exploring points in the surveillance area

The static soaring surveillance approach allows the on-board energy system to regenerate energy from updrafts. The strength of an updraft decays along the distance from the center. This characteristic requires the UAV to fly through the neighborhood of the updraft to identify and exploit it. Unfortunately, updrafts are transparent air masses floating in the atmosphere. The strengths and locations of these air masses are difficult to make predictions. Therefore, there is always an element of luck to encounter an updraft during surveillance flight [78].

For the purpose of encountering an updraft during surveillance flight, the surveillance area with \( N_t \) randomly distributed targets (e.g. Fig. 3.2 in Chapter 3) is divided equally into grids. The size of the grid is defined to consider the following factors: 1) the grid is small enough to capture an updraft; 2) the UAV can “see” the group of targets inside a grid by visiting the centroid point of group.

In order to capture an updraft, the diameter of which scales from dozens of meters to hundreds of
meters [32, 28], the size of the grid is chosen as: 50 m × 50 m. Furthermore, the field of view (FOV) of the on-board vision-based sensor can be calculated by the following formula:

\[ FOV = 2 \times z \tan \frac{\alpha_v}{2} \]  

(4.1)

where FOV is the diameter of the vision-based sensor, \( z \) is the altitude, and \( \alpha_v \) is the angle of view.

The UAV flies above the surface layer which is the bottom 10% of the PBL to perform static soaring surveillance. According to the assumption of the PBL’s height (1 km) in the current study, the UAV’s altitude changes from 100 m to 1000 m. With a 50 m FOV, the angle of view of the on-board sensor varies from 4° to 39°, which is the typical range of the angle of view in a high-definition on-board telescope system [80]. Therefore, the UAV can “see” the group of targets within the 50 m × 50 m grid by visiting the centroid point of group.

By visiting surveillance points and strategically-selected exploring points (Table 4.1, Fig. 4.1) in grids, the UAV can achieve the surveillance task while maximizing the grid coverage to increase the chance of encountering updrafts.

Table 4.1: Surveillance and exploring points

<table>
<thead>
<tr>
<th>Points</th>
<th>Conditions</th>
<th>Definitions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Surveillance point</td>
<td>A grid includes multiple targets</td>
<td>The centroid of the group of targets</td>
</tr>
<tr>
<td>Surveillance point</td>
<td>A grid includes one target</td>
<td>The location of the target</td>
</tr>
<tr>
<td>Exploring point</td>
<td>A grid includes zero target</td>
<td>The center point of the blank grid</td>
</tr>
</tbody>
</table>

4.1.2 The traveling salesman problem

In order to visit as many surveillance points as possible while using only a certain amount of UAV’s initial energy, the surveillance path planning can be formulated as the traveling salesman problem (TSP), which determines the optimal visiting sequence of the predefined points. In order to alleviate the computational load of solving the TSP, the surveillance area is further partitioned into a series of concentrically expanding regions (Region 2,3,...,N...) as shown in Fig. 4.1. Region 1 is the initial region. The rest of the concentric regions is expanded one grid square wide each time.

The UAV visits surveillance points in each region following an optimal visiting sequence by solving the traveling salesman problem. The optimal visiting sequence \( x^* \) has the minimum overall cost \( S(x) \):

\[ x^* = \arg \min_{x \in X} S(x) = \arg \min \left( \sum_{i=1}^{n-1} \{c_{x_i,x_{i+1}}\} + c_{x_n,x_1} \right), \]  

(4.2)

where the cost \( c_{x_i,x_{i+1}} \) is defined as the Euclidean distance between every two points \( x_i, x_{i+1} \), \( X \) is the
sample set which includes all possible visiting sequences.

The cross-entropy (CE) [81] (pp.51-56) approach provides a feasible and computationally efficient solution to Eq. 4.2 based on the transition probability matrix $P$ (Eq. 4.3) update.

$$P = [P_{ij}] = \begin{bmatrix} 0 & P_{12} & \ldots & P_{1n} \\ P_{21} & 0 & \ldots & P_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ P_{n1} & P_{n2} & \ldots & 0 \end{bmatrix} \quad (4.3)$$

where $P_{ij}$ is the probability of a candidate visiting sequence containing the path from Point $i$ to Point $j$.

At the beginning, the probability is identical: $P_{ij} = \frac{1}{n}$. The transition probability matrix $P$ generates $\tilde{\chi}$: a subset of the sample set $\chi$. The size of the subset is $\bar{n} = 6n^2$ [81] for a TSP with $n$ points. The $\bar{n}$ samples in the subset can be sorted in an increasing order based on each sample’s cost evaluation: $S(x_{j_1}) \leq S(x_{j_2}) \ldots \leq S(x_{j_\bar{n}})$, where $j_1, j_2, \ldots, j_\bar{n}$ is a permutation of $1, 2, \ldots, \bar{n}$.

The first $n_1 = \bar{\rho} * \bar{n}$ ($\bar{\rho} = 0.01$ [81]) samples are defined as favorable visiting sequences. $\bar{\rho}$ is defined as
the cut-off value: \( \bar{\gamma} = S(x_{j\bar{n}}) \).

\( P_{ij} \) can then be updated by \([81]\)

\[
P_{ij} = Q_{ij} = \frac{I_{ij}}{I} \tag{4.4}
\]

where \( I = \bar{n}_1 \) is the total number of favorable visiting sequences and \( I_{ij} \) is the number of favorable visiting sequences containing the path from Point \( i \) to Point \( j \) \([81]\).

In order to prevent \( P \) from going to zero prematurely, the modified version \([81]\) of the update expression is adopted (Eq. 4.5). The modification computes a weighted sum of the current update \( Q_{ij} \) and the previous transition probability matrix \( P_{ij}^{l-1} \). The weight \( \bar{\alpha} \) in Eq. 4.5 set to be 0.7 (given in \([81]\)).

\[
P_i = [P_{ij}]
\]

\[
P_{ij}^{l} = Q_{ij} \ast \bar{\alpha} + (1 - \bar{\alpha}) \ast P_{ij}^{l-1}. \tag{4.5}
\]

The new subset \( \bar{\chi} \) is generated based on the updated \( P \) (Eq. 4.5). Favorable sequences are then selected and \( P \) can be updated again. The CE method involves the iterative update of \( P_{ij} \) until \( (P_{ij}^l - P_{ij}^{l-1}) < \epsilon \) (where \( \epsilon \) is a small number) or the maximum number of iterations is reached. The optimal surveillance path follows the point-visiting sequence which is determined by the elements that are closest to one, e.g., \( P_{12} \approx 1 \) indicating the sequence Point 1 \( \rightarrow \) Point 2.

**Remark.** For the TSP with \( n \) points, there are \( n! = 1 \times 2 \times 3 \cdot \cdots \cdot n \) possible paths. It is obviously intractable if the cost of each path is examined. In the cross-entropy method, we can use a subset \( \tilde{\chi} \) with \( \bar{n} = 6n^2 \) visiting sequences to get an acceptable result. From the computational-complexity’s point of view, the TSP belongs to the class of NP-complete problem whose running time increases exponentially with the number of points. The cross-entropy method reduces the complexity into \( O(n^2) \), where \( n \) is the number of points in the traveling salesman problem.

The sub-optimal visiting order (Fig. 4.2 a) is obtained by the cross-entropy method in the Region 1, which is chosen to be \([-100, 100] \times [-100, 100] \) m, containing 7 surveillance points. From Regions 2 to \( N \), which are in the shape of hollow rings, a Hamiltonian cycle (a circuit that visits each point exactly once) is utilized to generate the visiting sequence (Fig. 4.2 b).

### 4.1.3 The traveling salesman problem for Dubins’ vehicle

The visiting path (Fig. 4.2) is not a practical path for a UAV, because it does not take account of UAV’s turning-rate constraint. The TSP which is formulated in Eq. 4.2 is called the Euclidean traveling salesman solution (ETSP). The Dubins’ traveling salesman problem (DTSP) is formulated to consider the turning-rate constraint of the UAV. Savla’s algorithm \([62]\) provides a provable good approximation to the optimal solution of the DTSP by replacing all even-numbered edges in the ETSP solution (Fig. 4.2) with minimum-length Dubins’ paths, preserving the point visiting-sequence.
Chapter 4. Static soaring surveillance in the quasi-static updraft field

Figure 4.2: The sub-optimal visiting sequence (ETSP solution) in Region 1 and 2

Remark. The Dubins’ path is a continuously differentiable curve to connect two arbitrary points with specified orientations. In the minimum-length Dubins’ path, the magnitude of the curvature is the minimum turn radius of a UAV. The minimum-length Dubins’ paths can be used to construct an approximate optimal solution to the DTSP based on the knowledge of the ETSP’s solution, even though the small turn radius results in large induced drag. The amount of energy loss will be compensated by soaring maneuvers.

The minimum radius is calculated by the following equations. Without severe climbing in static soaring, the small flight path angle approximation \( \cos \gamma \approx 1 \) is applied to obtain \( L \cos \mu = mg \). The load factor \( n_f \) and turning rate \( \dot{\phi} \) can be expressed as:

\[
\begin{align*}
n_f &= \frac{L}{mg} = \frac{1}{\cos \mu} \quad (4.6) \\
mV \dot{\phi} &= mV \frac{V}{R} = L \sin \mu \quad (4.7)
\end{align*}
\]

Equation 4.7 implies

\[
R = \frac{mV^2}{L \sin \mu} = \frac{1}{n_f g \sin \mu} \frac{V^2}{g \sqrt{n_f^2 - 1}}. \quad (4.8)
\]

Substituting \( V = \sqrt{\frac{2n_f mg}{\rho S C_L}} = \sqrt{n_f} \sqrt{\frac{2mg}{\rho S C_L}} \) into Eq. 4.8, the turning radius \( R \) can be expressed as:

\[
R = \frac{n_f \frac{2mg}{\rho S C_L}}{g \sqrt{n_f^2 - 1}}. \quad (4.9)
\]

According to the lift coefficient \( C_L \) and bank angle \( \mu \) limits (\( C_L = 1.5, \mu = 60^\circ \)) in static soaring...
study), the maximum load factor is $n_{f_{\text{max}}} = 2$ in the static soaring study. Applying this to the UAV model ($m = 13.5 \text{ kg}, S = 2.89 \text{ m Table 4.3}$), we can calculate the maximum turning rate and minimum turning radius as follows:

$$R_{\text{min}} = \frac{n_{f_{\text{max}}} \frac{2mg}{\rho SC_{l_{\text{max}}}}}{g\sqrt{n_{f_{\text{max}}}^2 - 1}} = 30.8491 \text{ m}$$

$$\dot{\varphi}_{\text{max}} = \sqrt{\frac{n_{f_{\text{max}}} \frac{2mg}{\rho SC_{l_{\text{max}}}}}{R_{\text{min}}}} = 0.7418 \text{ rad/s}$$

(4.10)

In order to prevent stalling, a turning radius of 35 m is used to generate the curve segment in the Dubins’ path. The resultant Dubins’ paths are used to replace all even-numbered edges in the ETSP solution (Fig. 4.2). Figure 4.3 illustrates surveillance trajectories incorporated with Dubins’ paths in Region 1 and Region 2.

![Figure 4.3: Dubins’ surveillance path in Region 1 and 2](image)

**4.1.4 The exploring point determination**

Figure 4.3 shows that not every grid is covered by the UAV’s surveillance path. Exploring points are then strategically defined in some of the blank grids for better coverage and opportunities of discovering updrafts in surveillance. The number and location of the exploring points are chosen to achieve the trade-off between the augmented traveling distance and the grid coverage.

In Region 1, $n$ surveillance points are connected by Dubins’ surveillance trajectory (Fig. 4.3). Inside each grid which is not covered by the trajectory, the center point is chosen as the exploring point candidate. Selecting $m = 1 \cdots M$ points successively from the exploring-point group with $M$ candidates,
Table 4.2: Quantitative analysis of the coverage and traveling distance

<table>
<thead>
<tr>
<th>Region 1 with 16 grids</th>
<th>Number of covered grid</th>
<th>Traveling distance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Augmented surveillance path with random exploring points</td>
<td>16</td>
<td>2217 m</td>
</tr>
<tr>
<td>Augmented surveillance path with optimal exploring points</td>
<td>16</td>
<td>1494 m</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Region 2 with 20 grids</th>
<th>Number of covered grid</th>
<th>Traveling distance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Augmented surveillance path with random exploring points</td>
<td>20</td>
<td>2001 m</td>
</tr>
<tr>
<td>Augmented surveillance path with optimal exploring points</td>
<td>20</td>
<td>1860 m</td>
</tr>
</tbody>
</table>

the augmented surveillance path connecting \( n + m \) points can be generated by solving the corresponding Dubin’s TSP. By comparing the traveling distance of all combination \( C_M^1 \cdot C_M^2 \cdots C_M^M \), we choose the combination \( C_M^{m^*} \), which leads to the shortest augmented surveillance path and the best grid coverage.

For the concentrically expanding regions (Region 2 to N), we define four exploring point candidates in four corners to force the surveillance path to pass through more grids in each region. Subsequently, we choose \( m^* \) exploring points out of the four, which leads to the shortest augmented surveillance path and the best grid coverage.

Table 4.2 illustrates the coverage and traveling distance in various cases. Figure 4.4 shows the augmented surveillance paths in Region 1 and 2. The results demonstrate that the combination \( C_M^{m^*} \) can provide the trade-off between the augmented surveillance path and grid coverage.

## 4.2 Static soaring surveillance approach

In this thesis, we consider the case of the quasi-static wind field where the updraft is stationary or oscillating around the hot origin. The static soaring surveillance approach includes visiting surveillance and exploring points to achieve the task, observing the vertical wind speed along the way, estimating vertical wind speed map (distribution), and exploiting updrafts.

In Region 1, the UAV visits points and measures the vertical wind speed along the Dubins’ surveillance trajectory. At the end of the surveillance flight, Gaussian process regression (GPR) is applied to estimate the vertical wind speed distribution in Region 1. It is assumed that Region 1 includes updrafts. GPR is utilized to determine the soarable location.

In Region \( i \) (\( i = 2 \cdots N \)) (the concentrically expanding regions), the UAV visits points and measures the vertical wind speed along the Dubins’ surveillance trajectory. At the end of the surveillance flight in Region \( i \), the wind speed distribution is estimated in Region \( (i - 1) \) instead of the current region. The
mismatch between the surveillance and wind field estimation is caused by the boundary constraints which
are incorporated into the Gaussian process regression approach to address the discontinuity estimation
issue between regions. Applying GPR with boundary constraints, the estimation of Region $i$ requires the
measurements of two neighboring regions (e.g. Region 2 estimation requires the measurements in Region
1 and 3 as shown in Fig. 4.1). Therefore, the wind speed distribution in Region $(i - 1)$ is estimated at
the end of the surveillance flight in Region $i$.

Following the soaring surveillance approach, the wind speed distribution in Region 1 is estimated
twice. One (without boundary constraints) is at the end of the surveillance flight in Region 1 and the
other one (with boundary constraints) is at the end of surveillance in Region 2. The results without
boundary constraints help to identify the soarable updraft in the initial region. The results with the
boundary constraints guarantee the consistent estimation results between Region 1 and Region 2. Based
on the wind estimation results, the spot which has the strongest upward wind speed is the estimated
location of the updraft center. The UAV then loiters around the estimated updraft center to gain altitude (potential energy). When the UAV reaches the top of the planetary boundary layer \((z = 1 \text{ km})\), the UAV stops updraft soaring and carries on surveillance flight in the subsequent regions. The static soaring surveillance approach in the quasi-static wind field can be presented in Algorithm 1. In the following sections, each part of Algorithm 1 is presented in detail.

### 4.2.1 Visit surveillance and exploring points

#### Heading angle rate determination

In a real wind field, horizontal winds may occur because of air current circulation. Visiting both surveillance and exploring points, the effect of horizontal winds has to be taken into account.

In the presence of horizontal winds, the analysis of the relationship between the ground speed and airspeed of the UAV is required. The following figure (Fig. 4.5) shows the UAV flies with airspeed \(V\) in the presence of horizontal winds \(W = W_x + W_y\).

Wind speed \(W_x\) and \(W_y\) can be estimated from ground-relative velocity and airspeed measurements via on-board sensor suites (GPS and IMU) \([35]\). Based on the current states (airspeed \(V\), flight path angle \(\gamma\), and heading angle \(\varphi\)), the angle \(\sigma\) can be calculated by Eq. 4.11.

\[
\theta = \arctan \frac{V \cos \gamma \sin \varphi + W_x}{V \cos \gamma \cos \varphi + W_y}
\]

\[
\sigma = \theta - \varphi
\]

\[
\dot{\varphi} = \dot{\theta} - \dot{\sigma}
\]
By comparing the angles $\sigma$ in current and previous steps, $\dot{\sigma}$ can be obtained (at the initial step, $\dot{\sigma} = 0$). $\dot{\theta}$ is the ground speed heading rate which can be determined by the Dubins’ path (the DTSP solution). Therefore, the heading angle rate $\dot{\phi}$ can be solved by Eq. 4.11.

The maximize lift-to-drag speed trajectory: numerical and analytical solutions

From Eq. 2.13 and Eq. 2.14, with the effect of the wind speed $W_x, W_y, W_z$, the equations of motion for the small UAV with zero side-slip, assuming $[\frac{dW_x}{dt} = 0, \frac{dW_y}{dt} = 0, \frac{dW_z}{dt} = 0]$ can be derived as:

$$\dot{V} = -\frac{D}{m} - g \sin \gamma$$
$$\dot{x} = V \cos \gamma \sin \varphi + W_x$$
$$\dot{y} = V \cos \gamma \cos \varphi + W_y$$
$$\dot{z} = V \sin \gamma + W_z$$
$$mV \dot{\gamma} = L \cos \mu - mg \cos \gamma$$
$$mV \cos \gamma \dot{\varphi} = L \sin \mu$$

(4.12)

For each turning rate $\dot{\varphi}$ (Eq. 4.11), the solution to the optimization problem (Eq. 4.13) can provide the steady states (airspeed $V$, flight path angle $\gamma$, lift coefficient $C_L$, and bank angle $\mu$) which lead to a steady turn flight with the maximum gliding range.
minimize \[ V, \gamma, C_L, \mu \]
\[ \gamma = \arctan \left( \frac{C_D}{C_L \cos \mu} \right) \]
subject to
\[ mg \cos \gamma = L \cos \mu \]
\[ -mg \sin \gamma = D \]
\[ L = 0.5 \rho V^2 SC_L \]
\[ D = 0.5 \rho V^2 SC_D \]
\[ C_D = C_{D_0} + \frac{C_L^2}{\pi e AR} \]
\[ \dot{\phi} = \frac{L \sin \mu}{mV \cos \gamma} \]
\[ C_{L_{\min}} \leq C_L \leq C_{L_{\max}}, \quad \mu_{\min} \leq \mu \leq \mu_{\max} \]

where \( \mu \) is bank angle, \( m \) is the mass of the vehicle, \( L \) is lift with coefficient \( C_L \), \( D \) is drag with coefficient \( C_D \), \( C_{D_0} \) is the parasitic drag coefficient, \( AR \) is the aspect ratio, and \( e \) is Oswald’s efficiency factor. In this thesis, we use the Aerosonde UAV model \( [2] \). The parameters are given in Table 4.3.

Table 4.3: Aerosonde UAV parameters

<table>
<thead>
<tr>
<th>Mass (kg)</th>
<th>Wing area (m²)</th>
<th>Wing span (m)</th>
<th>Aspect ratio (AR)</th>
<th>Parasitic drag coefficient ( C_{D_0} )</th>
<th>Oswald’s efficiency factor ( e )</th>
</tr>
</thead>
<tbody>
<tr>
<td>13.5</td>
<td>0.55</td>
<td>2.89</td>
<td>15.24</td>
<td>0.027</td>
<td>0.75</td>
</tr>
</tbody>
</table>

The steady states (airspeed \( V \), flight path angle \( \gamma \), lift coefficient \( C_L \), and bank angle \( \mu \)) can also be obtained by conducting the following analytical analysis.

According to the relationship between \( C_L \) and \( C_D \):

\[ C_D = C_{D_0} + \frac{C_L^2}{\pi e AR} \]  

(4.14)

The lift-to-drag ratio can be presented as:

\[ \frac{L}{D} = \frac{C_L}{C_D} = \frac{C_L}{C_{D_0} + \frac{C_L^2}{\pi e AR}} \]  

(4.15)

We take the derivative with respect to \( C_L \):

\[ \frac{d}{dC_L} \left( \frac{C_{D_0} + \frac{C_L^2}{\pi e AR}}{C_L} \right) = 0 \]  

(4.16)

Therefore, \( C_{L_{\text{opt}}} = \sqrt{\pi e AR C_{D_0}} = 0.9848 \). For various cases of turn rate \( \dot{\phi} \), the steady states (airspeed, bank angle, and lift coefficient) which lead to steady flight with maximum \( L/D \) can be obtained by solving the following three equations:
\[
\gamma = - \arctan \left( \frac{C_D}{C_{L_{\text{opt}}}} \cos \mu \right) \\
V = \sqrt{\frac{mg \cos \gamma}{0.5 \rho S C_{L_{\text{opt}}} \cos \mu}} \\
\tan \mu = \frac{\dot{\varphi} V}{g}
\] (4.17)

Figure 4.6, Fig. 4.7, Fig. 4.8, Fig. 4.9, and Fig. 4.10 show the airspeed, lift coefficient, the flight path angle, bank angle, and the sink rate in steady turning flight with the maximum lift-to-drag ratio.

The problem of the steady turning flight with the maximum lift-to-drag ratio can be solved analytically (Eq. 4.17, red line) and numerically (Eq. 4.13, blue line). According to Eq. 4.16 and Eq. 4.13, the analytical and numerical solutions are obtained by maximizing \( L/D \) and \( L \cos \mu / D \) respectively. In the analytical solution, the centripetal force, a component of lift force that allows steady turning flight, is generated by increasing airspeed, whereas in the numerical solution, the centripetal force is obtained by increasing lift coefficient (i.e., the angle of attack).

According to Fig. 4.10, the numerical solution provides less sink rates for high turning-rate cases. The numerical solution is utilized to generate the static soaring and surveillance trajectory in order to take advantage of mild updrafts (e.g., 2.5 - 4 m/s upward winds).

Dubins' surveillance trajectory

Based on the numerical results in Fig. 4.7 and Fig. 4.9, for each turning rate \( \dot{\varphi} \), the corresponding lift coefficient \( C_L \) and bank angle \( \mu \) can be determined. The steady states (airspeed, flight path angle) which leads to the maximum range can be calculated from the equations of motion (Eq. 4.12). Figure 4.11 provides the flowchart of the Dubins' surveillance trajectory calculation.

Figure 4.12 shows the Dubins' surveillance trajectory in the presence of horizontal winds in Region
Chapter 4. Static soaring surveillance in the quasi-static updraft field

4.2.2 Updraft identification by Gaussian process regression with boundary constraints

Along the Dubins’ surveillance trajectory, the UAV visits predefined points sequentially and observes vertical wind speed $W_z$ via the on-board sensor. According to the soaring surveillance approach, the
surveillance area is decomposed into a number of expanding regions (Region 1, 2, ..., N; see Fig. 4.1), the vertical wind speed map of each region can be obtained by correlating vertical wind speed observations via Gaussian process regression. Since the standard GPR suffers from the problem of discontinuity regression results on the region boundaries, constraints are imposed on the standard GPR to address this problem. Based on wind estimation results, the spot which has the strongest upward wind speed is the estimated location of the updraft center.

**Standard Gaussian process regression (GPR)**

After a set of vertical wind observations \( \hat{W}_z = \{ \hat{W}_{zi} \}_{i=1}^{\hat{M}} \) is made, the prediction \( \hat{f}(x^*) \) at any location \( x^* \) can be expressed by a linear predictor [82]:

\[
\hat{f}(x^*) = \sum_{i=1}^{\hat{M}} d_i \hat{W}_{zi} = d^T \hat{W}_z
\]

(4.18)

where \( d = \{ d_i \}_{i=1}^{\hat{M}} \) are the coefficients of the linear combination of \( \hat{M} \) measurements \( \hat{W}_z \).

The optimal coefficient \( d^* = d_i^{\hat{M}} \) can be determined by minimizing the mean squared prediction error \( E[(\hat{f}(x^*) - f(x^*))^2] \); [82]

\[
\min_d E[(\hat{f}(x^*) - f(x^*))^2] = \min_d (d^T [K(X,X) + \sigma_n^2 I]d - 2d^T k(X,x^*) + k(x^*,x^*))
\]

(4.19)

Here \( d = d_i^{\hat{M}} \) represents the \( \hat{M} \) parameters in the linear predictor (Eq. 4.18). \( f(x^*) \) represents the true value at location \( x^* \). \( X \) represents the set of location at which wind speed is observed by the on-board sensor. \( K(X,X) \) is the covariance matrix, which represents the covariance value between every two observations in the set of measurements. \( k(X,x^*) \) represents the covariance vector which describes
the covariance between the two points. One point belongs to the observation location set \( \mathbf{X} \) and the other one is the estimation location \( x^* \). \( k(x^*, x^*) \) represents the covariance value.

Taking \( \mathbf{K}(\mathbf{X}, \mathbf{X}) \) as an example, it can be expressed as:

\[
\mathbf{K}(\mathbf{X}, \mathbf{X}) = \begin{bmatrix}
  k(p_1, p_1) & k(p_1, p_2) & \cdots & k(p_1, p_n) \\
  k(p_2, p_1) & k(p_2, p_2) & \cdots & k(p_2, p_n) \\
  \vdots & \vdots & \ddots & \vdots \\
  k(p_n, p_1) & k(p_n, p_2) & \cdots & k(p_n, p_n)
\end{bmatrix}
\]

Each element \( k(x_{p_i}, x_{p_j}) \) of the covariance matrix \( \mathbf{K}(\mathbf{X}, \mathbf{X}) \) is defined as the covariance function, which is chosen as the squared exponential function (Eq. (4.20)):

\[
k(p_i, p_j) = \sigma_f^2 \star \exp \left( -\frac{|x_{p_i} - x_{p_j}|^2 + |y_{p_i} - y_{p_j}|^2}{2l^2} \right).
\]

(4.20)

Here \( p_i = [x_{p_i}, y_{p_i}] \) and \( p_j = [x_{p_j}, y_{p_j}] \) are two arbitrary points in the observation location set \( \mathbf{X} \). \( \| \cdot \| \) represents 2-norm distance between Point \( p_i \) and Point \( p_j \) in x and y direction. \( l \) is the characteristic length-scale.

Once wind speed observations (suppose that \( M \) measurements in Region 1, \( \mathbf{\hat{W}}_z = \{\hat{W}_{z_i}\}_{i=1}^M \) are obtained, we determine hyper-parameters \( \sigma_f, l \) in the squared exponential covariance function (Eq. (4.20)) by maximizing the following marginal likelihood [34]:

\[
\max_{\sigma_f^2, l} \{-0.5\mathbf{\hat{W}}_z^T (\mathbf{K} + \sigma_n^2 \mathbf{I})^{-1} \mathbf{\hat{W}}_z - 0.5\log |\mathbf{K} + \sigma_n^2 \mathbf{I}| - 0.5n\log(2\pi)\}.
\]

(4.21)
The optimal hyper-parameters $\sigma_f^*, l^*$ from Eq. 4.21 determine the value of the covariance matrix $K(X, X)$ and the covariance vector $k(X, x^*)$.

By solving Eq. 4.19, the optimal solution is:

$$d^* = [K(X, X) + \sigma_n^2 I]^{-1} k(X, x^*)$$

(4.22)

The standard GPR which is one type of the linear predictor can be expressed as:

$$\tilde{f}(x^*) = d^T \hat{W}_z = k(x^*, X)[K(X, X) + \sigma_n^2 I]^{-1} \hat{W}_z$$

$$\text{cov}(\tilde{f}(x^*)) = k(x^*, x^*) - k(x^*, X)[K(X, X) + \sigma_n^2 I]^{-1} k(X, x^*)$$

(4.23)

Gaussian process regression with boundary constraints

We address the boundary issue by implementing Park’s fast Gaussian process regression approach [82] in the concentrically expanding region. Recall that the solution to the optimization problem (Eq. 4.19) has
the following form [82]:

$$d^* = [K(X, X) + \sigma_1^2 I]^{-1} k(X, x^*) = A \times k(X, x^*)$$  \hspace{1cm} (4.24)$$

The optimization problem (Eq. 4.19) can be redefined by the following equation, whose solution $A$ is independent of the location $(x^*)$ [82].

$$\min_{A \in \mathbb{R}^{N \times N}} \left( k(X, x^*)^T A^T [K(X, X) + \sigma_1^2 I] A k(X, x^*) - 2 k(X, x^*)^T A^T k(X, x^*) \right)$$ \hspace{1cm} (4.25)$$

To guarantee consistent vertical wind speed estimation between regions, boundary constraints are imposed on the minimization problem in Eq. 4.25. The following constrained minimization problem is solved to estimate speed at location $x_j^*$ in Region $j$:

$$\min_{A_j \in \mathbb{R}^{N \times N}} k_j(X_j, x_j^*)^T A_j^T [K_j(X_j, X_j) + \sigma_1^2 I] k_j(X_j, x_j^*) A_j$$ $$- 2 k_j(X_j, x_j^*)^T A_j^T k_j(X_j, x_j^*)$$ \hspace{1cm} (4.26)$$

s.t. K_j(X_j, X_o)^T A_j^T [\hat{W}_j] = R_{j,k}(X_o).$$

where, $\{j\}$ represents parameters in Region $j$. $X_o$ represents $q$ points $X_o = \{[x_o_i]\}_{i=1}^{q}$ uniformly-distributed on the boundary between Region $j$ and $k$. $R_{j,k}(X_o)$ represents the values at locations $X_o$ on the boundary. The solution to the minimization problem in Eq. 4.26 gives the wind prediction at the unobserved location $x_j^*$ in Region $j$ and addresses the discontinuity issue. Park et al. [82] suggested using Lagrange basis polynomials as basis functions to represent the boundary values $R_{j,k}(X_o)$. Here we use the estimation problem with boundary constraints between Region 1 and 2 as an example to illustrate this idea.

Suppose $p$ interpolation points $X_{b,1,2}^b = [x_1^b, x_2^b, \ldots, x_p^b]$ are chosen uniformly on the boundary $R_{1,2}$ between Region 1 and 2. We determine the values ($R_{1,2}(X_{b,1,2}^b)$) of the $p$ interpolation points $X_{b,1,2}^b$ on the boundary $R_{1,2}$ by the following expression [82]:

$$R_{1,2}(X_{b,1,2}^b) = \left( T_{1,2}(X_{b,1,2}^b)^T T_{1,2}(X_{b,1,2}^b) \right)^{-1} T_{1,2}(X_{b,1,2}^b)^T \left( \frac{c_1}{c_1 + c_2} K^T(X_1, X_{b,1,2}^b) h_1 + \frac{c_2}{c_1 + c_2} K^T(X_2, X_{b,1,2}^b) h_2 \right)$$ \hspace{1cm} (4.27)$$

Here,

$$c_1 = \hat{W}_{z_1}^T h_1, \quad h_1 = (K(X_1, X_1) + \sigma_1^2 I)^{-1} \hat{W}_{z_1}.$$
\[ c_2 = \hat{W}_{zz}^T h_2, h_2 = (K(X_2, X_2) + \sigma_2^2 I)^{-1} \hat{W}_{zz} \]

\( T_{1,2}(X^b_{1,2}) \) are Lagrange basis polynomials with \( p \) interpolation points on the boundary between Region 1 and 2 (\( X^b_{1,2} = [x^b_1, x^b_2, \ldots, x^b_p] \)). \( \hat{W}_{z_1} \) and \( \hat{W}_{z_2} \) are \( M_1 \times 1 \) and \( M_2 \times 1 \) vertical wind observation column vectors in Region 1 and 2 respectively. \( X_1 \) and \( X_2 \) represent the locations of the observation in Region 1 and 2 respectively. \( X^b_{1,2} \) is the \( p \) interpolation points on the boundary between Region 1 and 2.

The boundary values \( R_{1,2}(X_o) \) can be further expressed based on the values of the \( p \) interpolation points (\( R_{1,2}(X^b_{1,2}) \)) via Lagrange polynomials [82]:

\[ R_{1,2}(X_o) = T_{1,2}(X_o)R_{1,2}(X^b_{1,2}). \] (4.28)

By substituting the values \( R_{1,2}(X_o) \) into Eq. 4.26, the minimization problem can be solved.

The GPR with boundary constraints on \( q \) points \( X_o = \{ [x_{o,i}] \}_{i=1}^q \) provides a predictor \( \hat{f}_* \) [82]:

\[ \hat{f}_* = \hat{f}_s + \hat{k}(x^*, X_o)G_1 \left( R_{1,2}(X_o) - K^T(X_1, X_o)[K(X_1, X_1) + \sigma_n^2 I]^{-1}\hat{W}_{z_1} \right) \] (4.29)

where \( X_o = \{ [x_{o,i}] \}_{i=1}^q \), \( \hat{f}_s = k(x^*, X_1)[K(X_1, X_1) + \sigma_n^2 I]^{-1}\hat{W}_{z_1} \), and

\[ G_1^{-1} = \left( \text{diag}_{1/2}([K^T(X_o, X_o)K(X_o, X_o)]^{-1}[K(X_o, X_o)]) \right) \circ (K^T(X_1, X_o)K(X_1, X_o)) \text{diag}([K^T(X_1, X_o)K(X_1, X_o)]^{-1}) \],

\[ \hat{k}(x^*, X_o) = \left( (k(x^*, X_o)k(X_o, x^*))^{-1/2}k(X_o, x^*) \right) \circ (K^T(X_1, X_o)K(X_1, X_o)) \text{diag}([K^T(X_1, X_o)K(X_1, X_o)]^{-1}). \]

The operator \( \circ \) represents the Hadamard product. \( \text{diag}_{1/2} \) takes the square of the diagonal elements. The derivation can be found in [82].

Region \( i \) (\( i \geq 2 \)) shares the boundary with Region \( (i + 1) \), as well as Region \( (i - 1) \). Therefore, the wind field in Region \( i - 1 \) is estimated at the end of the aerial surveillance in Region \( i \) (\( i \geq 2 \)). In the current work, each round of wind estimation extends the estimated wind field but does not effect the previously existing results.

**Updraft identification results by GPR with constraints**

The mathematical updraft model using the extended Gaussian function [33] is utilized here to demonstrate the effectiveness of the Gaussian process regression (GPR) with boundary constraints. The wind speed vector \( W(x, y) = [W_x, W_y, W_z] \) at the location \( (x, y) \) is calculated by the mathematical updraft model attached with vertical wind disturbances (Eq. 4.30).
Chapter 4. Static soaring surveillance in the quasi-static updraft field

\[ [W_x(x, y)] = 0 \]
\[ [W_y(x, y)] = 0 \]
\[ [W_z(x, y)] = -\sqrt{2} \sin \left( \frac{2\pi(x)}{200} \right) - \sum_{i=1}^{N_{th}} W_i e^{-\frac{(x-x_i)^2+(y-y_i)^2}{R_i^2}} \left[ 1 - \frac{(x-x_i)^2+(y-y_i)^2}{R_i^2} \right] \]

where \( N_{th} \) is the number of the updrafts that are spatially-distributed in the area.

In the case study, \( N_{th} = 3 \). Three updrafts are assumed to be located at (50, 50) m, (−120, 120) m, and (250, −275) m respectively. The central strength of each updraft \( W_i \) is \( W_1 = 3 \text{ m/s} \), \( W_2 = 5 \text{ m/s} \), and \( W_3 = 7 \text{ m/s} \) respectively. The radius of every updraft is \( R_i = 50 \text{ m} \).

Figure 4.15 b and c show the estimated results with and without continuity constraints for the case study wind field (Eq. 4.30, Fig. 4.15 a). According to the result, Gaussian process regression with boundary constraints can provide consistent estimation between regions.

![Figure 4.15: True and estimated wind fields from Region 1 to Region 7](image)

**4.2.3 Updraft soaring strategy**

The objective of this section is to determine an updraft soaring strategy, by which the UAV can loiter around the thermally-driven updraft to achieve a certain altitude (potential energy).

The updraft soaring strategy is to generate the desired heading command which is determined by the location of an estimated point. The estimated point, which represents the approximate location of the updraft center, is first determined by GPR, then corrected by local wind measurements during updraft
soaring. The Dubins-path-based guidance law allows the UAV to follow the desired heading command. As such, the UAV can loiter around the estimated updraft center along the circular path to gain energy.

**The desired heading command**

Suppose that \( P_g = [x_g, y_g] \), \( P_c = [x_c, y_c] \) are the position of the UAV and the estimated point respectively on the \( X - Y \) plane as shown in Fig. 4.16.

\[
\begin{align*}
    x_g &= x_c + d \sin \lambda \\
    y_g &= y_c + d \cos \lambda
\end{align*}
\]  

where \( d \) is the relative distance between the UAV and the estimated point, and \( \lambda \) the direction of the vector \( \overrightarrow{PC} \).

The equations of motion for the UAV on the \( X - Y \) plane can be presented as:

\[
\begin{align*}
    \dot{x}_g &= V_g \sin \theta \\
    \dot{y}_g &= V_g \cos \theta
\end{align*}
\]  

Figure 4.16: The UAV \( P_g \) and estimated point \( P_c \) on the \( X - Y \) plane

Figure 4.17: Vector field: the desired heading direction
where $V_g$ is the ground speed vector, and $\theta$ is the ground heading angle.

In the current work, we consider the counter-clockwise circle with radius $r_c$ (the dash circle in Fig. 4.16), which is the goal path that the UAV has to follow. According to the following Lemma, the desired heading command $\theta^d$ can be defined as:

\[
\theta^d = \begin{cases} 
\lambda - \pi & \text{if } d > r_c; \\
\lambda - \frac{\pi}{2} & \text{if } d = r_c; \\
\lambda & \text{if } d < r_c.
\end{cases}
\]

**Lemma** Consider the equations of motion of the UAV (Eq. 4.32), and the relative motion (Eq. 4.31), if the UAV follows the desired heading command $\theta^d$, the UAV will loiter around the estimated point along the counter-clockwise circular path with radius $r_c$ (the goal circular path).

**Proof.**

A Lyapunov function can be defined as: $V_1 = (d - r_c)^2$.

Taking the derivatives of the Lyapunov function with respect to time, we can have: $\dot{V}_1 = 2(d - r_c) \dot{d}$. Consider $d$ is the relative distance between the UAV and the estimated point: $d^2 = (x_c - x_g)^2 + (y_c - y_g)^2$. Taking the derivative of $d$, one can obtain:

\[
\dot{d} = \frac{x_c - x_g}{d} (-\dot{x}_g) + \frac{y_c - y_g}{d} (-\dot{y}_g) \quad (4.33)
\]

Substituting Eq. 4.31 and Eq. 4.32 into Eq. 4.33, the relative distance’s derivative becomes:

\[
\dot{d} = V_g \cos(\theta - \lambda) \quad (4.34)
\]

Therefore, the derivative of the Lyapunov function can be presented as: $\dot{V}_1 = 2(d - r_c) V_g \cos(\theta - \lambda)$. When $d > r_c$, the desired heading $\theta^d = \lambda - \pi$, therefore $\dot{V}_1 = -2(d - r_c) V_g < 0$. When $d = r_c$, the desired heading $\theta^d = \lambda - \frac{\pi}{2}$, therefore $\dot{V}_1 = 2(d - r_c) V_g \cos \frac{\pi}{2} = 0$. When $d < r_c$, the desired heading $\theta^d = \lambda$, therefore $\dot{V}_1 = 2(d - r_c) V_g < 0$.

When $d \neq r_c$, $\dot{V}_1 < 0$, therefore, $V_1$ approaches into zero within finite time, which means $d$ approaches into $r_c$. When $d = r_c$, $\dot{V}_1 = 0$. As as result, the UAV will loiter around the estimated point along the circular path with radius $r_c$. Figure 4.17 shows the vector field results based on the desired heading command $\theta^d$. 
Dubins-path-based guidance law

The Dubins’ path connects two arbitrary points on the plane with two arbitrary orientations. One of the points is UAV’s location \( P_g = [x_g, y_g] \), and the other point is the cross point \( P_i \) as shown in Fig. 4.16. The initial orientation is UAV’s heading direction \( \theta \), and the final orientation is the desired heading direction \( \theta^d = (\lambda - \frac{\pi}{2}) \). Here \( \lambda \) describes the location relationship between \( P_g \) and \( P_i \). Following the Dubins’ path, the UAV can loiter around the estimated point to gain energy.

**Remark.** The Dubins-path-based thermal soaring algorithm is generated based on a known thermal. The location knowledge of the updraft is obtained during surveillance via Gaussian process regression. Traditional approaches, which utilize the second derivative of total energy \([83]\) or a predefined energy change queue \([18]\) to generate a turn rate command, are applied to fly a thermal without location knowledge.

Estimated point correction

When the UAV loiters around the approximate updraft center, the upward wind speed \( W_z \) in different position \( X_g, Y_g \) along the circular path is observed. Inspired by the modified center of mass approach \([17]\), the corrected estimated updraft center \( P_{\text{new}}(x_{\text{new}}, y_{\text{new}}) \) which indicates the updraft’s movement can be estimated as:

\[
\bar{x}_{\text{new}} = \frac{\hat{W}_z^T X_g}{\|\hat{W}_z\|_1}
\]

\[
\bar{y}_{\text{new}} = \frac{\hat{W}_z^T Y_g}{\|\hat{W}_z\|_1}
\]

The minimum-length Dubins’ path is applied again to connect the current position \( P_g \) and the new cross point \( P_i \) with respect to the new loitering center \((\bar{x}_{\text{new}}, \bar{y}_{\text{new}})\).

4.2.4 Updraft soaring results

The following diagram (Fig. 4.18) provides the flowchart of the proposed updraft soaring strategy. Based on this approach, the UAV is capable of loitering around an oscillating updraft to gain potential energy.

The mathematical updraft model (Eq. 4.30) \((N_{\text{th}} = 1)\) is utilized here as a case study.

\[
[W_x(x, y, t)] = 0
\]

\[
[W_y(x, y, t)] = 0
\]

\[
[W_z(x, y, t)] = -\sqrt{2} \sin \left( \frac{2\pi x}{200} \right) - W_1 e^{-\frac{(x-x'_1)^2 + (y-y'_1)^2}{R_1^2}} \left[ 1 - \frac{(x-x'_1)^2 + (y-y'_1)^2}{R_1^2} \right]
\]

**Remark.** The Dubins-path-based thermal soaring algorithm is generated based on a known thermal. The location knowledge of the updraft is obtained during surveillance via Gaussian process regression. Traditional approaches, which utilize the second derivative of total energy \([83]\) or a predefined energy change queue \([18]\) to generate a turn rate command, are applied to fly a thermal without location knowledge.
Suppose the updraft’s center \((x_1', y_1')\) oscillates around the original position \((x_1, y_1)\) following the circular wave.

\[
x_1' = x_1 + 5 \times \sin(t); \quad y_1' = y_1 + 5 \times \cos(t);
\]

where \(t\) is the time. The strength of the updraft \(W_1\) in Eq. 4.30 is \(W_1 = 7\) m/s. The updraft radius \(R_1\) in Eq. 4.30 is \(R_1 = 50\) m. The original updraft center \((x_1, y_1)\) is located at \([0,0]\).

**Remark.** The circular wave is used to represent atmospheric disturbance, pushing thermals to move around a hot spot. The more appropriate model will be studied in future work.

Figure 4.19 demonstrates the updraft soaring strategy. At the beginning, the initial estimated point (the blue cross) is determined by GPR. The estimated points (black and red crosses) are then corrected based on the local measurements during soaring flight. Figure 4.20 shows 2D soaring path. Based on the approach, the UAV can loiter around the oscillating updraft center to achieve the desired height. Figure 4.21 shows the airspeed and lift coefficient change of the UAV along soaring flight.

**Remark.** As shown in Fig. 4.21, the speed changes during soaring are caused by constantly switches between straight-line and turning flight along the Dubins’ path. According to the updraft soaring strategy (Fig. 4.18), after the updraft center is corrected by local wind measurements, the Dubins’ path is applied to connect the current position \(P_g\) and the new cross point \(P_i\) with respect to the new loitering center \((\bar{x}_{\text{new}}, \bar{y}_{\text{new}})\) (Eq. 4.35). Lift coefficients, as shown in Fig 4.21 are within the limits \((C_{L_{\text{max}}} = 1.5)\). This avoids stalls during flight.

The bank angle of 50 degree (as shown in Fig. 4.22) is utilized to generate turning portions of the Dubins’ path, providing a radius of 35 m turning circle. This allows the aircraft to soar inside small thermals (e.g. a thermal with radius 50 m). According to the plot of sink rate in Fig. 4.22, the aircraft can climb in a thermal, the speed of which is higher than 2 m/s. In reality, the rate of ascent of the air can be as high as 6 m/s [78].
### 4.2.5 Sensitivity analysis of the static soaring surveillance approach

The performance of the proposed approach depends on the effectiveness of the updraft identification approach (GPR). In this section, we study the sensitivity of GPR results with respect to various cases of updraft fields to analyze the sensitivity of the proposed approach.

For simplicity, the field is chosen as a $[-100, 100] \times [-100, 100]$ m square. The field is further divided into many $5 \times 5$ m small cells. The ground truth (the vertical wind speed) at each cell $(x, y)$ is calculated by the mathematical updraft model (Eq. 4.38 and Eq. 4.39).

$$W_z(x, y) = W_1 e^{-\frac{(x-x_1)^2 + (y-y_1)^2}{(R_1)^2}} \left[ 1 - \left( \frac{(x-x_1)^2 + (y-y_1)^2}{(R_1)^2} \right) \right]$$  \hspace{1cm} (4.38)

Here $W_z$ represents the vertical wind speed at position $(x, y)$. $W_1$ is the vertical wind speed at the updraft’s center $(x_1, y_1)$. $R_1$ is the radius of the updraft. Eq. 4.39 defines the movement of the updraft’s center $(x_1, y_1)$.

$$x_1 = x_1(t_0) + a_o \sin(b_o \cdot t)$$  \hspace{1cm} \hspace{1cm} (4.39)

$$y_1 = y_1(t_0) + a_o \cos(b_o \cdot t)$$

where $t$ is the time. $x_1(t_0)$ and $y_1(t_0)$ define the updraft’s center at initial time $t_0$. Updraft’s center oscillates with the amplitude $a_o$ m and frequency $b_o$ rad/s. Various cases of updrafts are generated by choosing random parameters in Eq. 4.38 and Eq. 4.39.

The UAV measures the vertical wind speed along the surveillance path (Fig. 4.23) in each updraft field case. Gaussian process regression (GPR) is utilized to estimate the vertical wind speed at each cell.
Figure 4.21: Airspeed, lift coefficient and height change along soaring flight

\[(x, y)\) in the \([-100, 100] \text{ m} \times [-100, 100] \text{ m}\] square based on 19 wind measurements (sampling time: 1 s) and 95 wind measurements (sampling time: 5 s). The mean squared error (MSE) (Eq. 4.40) is utilized to evaluate GPR’s performance.

\[
MSE = \frac{1}{n} \sum_{i=1}^{n} (\hat{W}_{zi} - W_{zi})^2
\]  \hspace{1cm} (4.40)

Here, \(\hat{W}_{zi}\) is the estimation result from GPR at \(i\) – th cell. \(W_{zi}\) is the ground truth at \(i\) – th cell. \(n\) is the number of cells.

Random position of the updraft’s center

The position of the updraft’s center \((x_1(t_0), y_1(t_0))\) is chosen from a uniform distribution \(U\):
Figure 4.22: Bank angle, sink rate and turning radius change along soaring flight

\[ x_1(t_0) = U(-100, 100) \text{ m} \]
\[ y_1(t_0) = U(-100, 100) \text{ m} \]  

(4.41)

Other parameters in Eq. 4.39 are defined as: \( W_1 = 5 \text{ m/s} \), \( R_1 = 50 \text{ m} \), \( a_0 = 0 \text{ m} \), \( b_0 = 0 \text{ rad/s} \).

Figure 4.24 demonstrates GPR’s performance using 19 measurements with respect to different locations of the updraft. In comparison with Fig. 4.24, Fig. 4.25, which utilizes 95 measurements, depicts a lower level of the mean squared error (MSE) and sensitivity with respect to position changes.

**Random updraft’s central strength**

The strength \( W_1 \) is chosen randomly from the following uniform distribution \( U \) (Eq. 4.42). Other parameters in Eq. 4.38 and Eq. 4.39 are defined as: \( x_1(t_0) = 0 \text{ m} \), \( y_1(t_0) = 0 \text{ m} \), \( R_1 = 50 \text{ m} \), \( a_0 = 0 \text{ m} \), \( b_0 = 0 \text{ m} \).
Figure 4.23: Dubins’ surveillance path in the $[-100, 100] \times [-100, 100]$ m field

\[ W_1 = U(0, 10) \text{ m/s} \] (4.42)

Figure 4.26 shows the sensitivity of GPR performance with respect to a change of updraft’s central strength. It depicts high sensitivity with respect to different updraft’s strength in the case of GPR with $\bar{M}_1 = 19$ measurements. For the $\bar{M}_2 = 95$ measurements case, GPR provides a lower level of the mean squared error (MSE) and sensitivity.

**Remark.** Stronger thermals reveals more variation within the field. As a result, more measurements are required to capture wind speed variations. The value of mean squared error $MSE = \frac{1}{n} \sum_{i=1}^{n} (\hat{W}_{z_i} - W_{z_i})^2$ is determined by the estimation error $\hat{W}_{z_i} - W_{z_i}$ at $n$ cells as shown in Eq. 4.40.

**Random Updraft’s radius**

The radius $R_1$ is chosen randomly from the following uniform distribution $U$ (Eq. 4.43). Other parameters are defined as: $x_1(t_0) = 0 \text{ m}, y_1(t_0) = 0 \text{ m}, W_1 = 5 \text{ m/s}, a_o = 0 \text{ m}, b_o = 0 \text{ m}.$

\[ R_1 = U(30, 300) \text{ m/s} \] (4.43)

Figure 4.27 shows the sensitivity with respect to the change of updraft’s radius. Since the large updraft, whose radius is beyond 100 m, occupies the entire field (the $[-100, 100] \times [-100, 100]$ m square), a few wind speed measurements can represent the wind pattern in this field. In this situation, GPR easily identifies the updraft even using 19 measurements. Therefore, the results (Fig. 4.27) show that a large updraft radius leads to a better estimation result. GPR’s performance is less sensitive to the
Figure 4.24: Estimation sensitivity with respect to thermal’s position \((x_1(t_0), y_1(t_0))\) m using 19 measurements

Figure 4.25: Estimation sensitivity with respect to thermal’s position \((x_1(t_0), y_1(t_0))\) m using 95 measurements

change of large updraft’s radius.

Random Oscillation amplitude

The sensitivity analysis is performed using a 100 random cases of updraft’s oscillation amplitude as shown in Fig. 4.28. The amplitude \(a_o\) is choose from the uniform distribution \(U\) (Eq. 4.44). Other parameters are chosen as: \(x_1(t_0) = 0\) m, \(y_1(t_0) = 0\) m, \(W_1 = 5\) m/s, \(R_1 = 50\) m, \(b_o = 1\) rad/s.

\[
a_o = U(-50, 50)\ m
\]  

(4.44)

The results (Fig. 4.28) depict a worse GPR estimation result using 19 measurements in the case of small amplitude of oscillation. Because of a small amplitude of oscillation, the updraft is located in the neighborhood of the initial position \(x_1(t_0) = 0\) m, \(y_1(t_0) = 0\) m. Similarly, in position sensitivity analysis (Fig. 4.24), GPR also provides a worse estimation result when the updraft locates around the central point \((0, 0)\). As shown in Fig. 4.23, the surveillance path has less coverage around the central point, and the set with 19 wind measurements includes limited information around the central point, resulting in a worse estimation result. In the case of 95 wind measurements, more wind information can be processed via GPR, resulting in a lower level of the mean squared error (MSE) and performance sensitivity with respect to the amplitude changes.
Random updraft’s oscillation frequency

The frequency $b_o$ is choose from the uniform distribution $U$ (Eq. 4.45). Other parameters are chosen as: $x_1(t_0) = 0 \text{ m}, y_1(t_0) = 0 \text{ m}, W_1 = 5 \text{ m/s}, R_1 = 50 \text{ m}, a_o = 5 \text{ m}$.

$$b_o = U(0, 50) \text{ rad/s} \quad (4.45)$$

Figure 4.29 provides GPR’s performance changes with updraft’s oscillation frequency (Eq. 4.45). The results demonstrate that more measurements help to enhance GPR’s performance.

Discussion

The study describes how sensitive the GPR results are to various cases of wind fields, and also provides a guideline for determining the number of measurements for GPR. Since GPR’s computational complexity grows as $o^4(M)$, where $M$ is the number of measurements, it is necessary to choose the appropriate number of measurements for GPR in order to achieve the trade-off between the acceptable computational load and reasonable estimation accuracy. In this study, the number is decided based on trail-and-error. For instance, wind speed measurements at grid points (e.g. 17 measurements in Region 1) are used for GPR to obtain the acceptable estimation results. Since the current GPR does not include the temporal component, it cannot provide the accurate estimation of a drifting updraft.
4.3 Static soaring surveillance simulation results

4.3.1 Soaring surveillance demo in a 1 km wind field

The 1 km × 1 km simulated wind field, that is a part of the surveillance area, is utilized to demonstrate the use of the proposed soaring and surveillance approach (Algorithm 1).

The 1 km × 1 km wind field is generated by the weather research and forecasting (WRF) model, which is a popular numerical weather prediction system. The WRF model determines the strength, size, number, and spatial distribution of updrafts in the predefined field (1 km × 1 km) via large eddy simulation. Updrafts are buoyant plumes of air that are caused by the temperature difference between the ground and the surrounding atmosphere. The temperature difference thermal turbulence in the (WRF) model is set to be 5 degree. The WRF model generates the 10 × 10 wind speed results $W(X, Y) = [U(X, Y), V(X, Y), W(X, Y)]$ for the 1 km × 1 km field with 100 m × 100 m cells (Fig. 4.30).

In order to represent atmospheric disturbance, the 10 × 10 interpolation points move with a circular wave. After time $t$, each interpolation point $X, Y$ becomes $X', Y'$ by Eq. 4.46:

$$
X' = X + 5 \times \sin(t) \\
Y' = Y + 5 \times \cos(t)
$$

(4.46)

where $t$ is the time. The amplitude $a_0 = 5$ m and frequency $b_o = 1$ rad/s of oscillation is chosen arbitrarily in this case study.

The wind speed vector $W(x, y) = [W_x, W_y, W_z]$ at a location $(x, y)$ is obtained by interpolating the
Chapter 4. Static soaring surveillance in the quasi-static updraft field

Figure 4.28: Estimation sensitivity with respect to thermal’s oscillation amplitude $a$ m

$10 \times 10$ wind speed results via MATLAB 2D interpolation function interp2:

$$
[W_x(x, y, t)] = \text{interpolation}(X', Y', U, x, y) \\
[W_y(x, y, t)] = \text{interpolation}(X', Y', V, x, y) \tag{4.47} \\
[W_z(x, y, t)] = \text{interpolation}(X', Y', W, x, y)
$$

where the matrices $X'$, $Y'$ specify $10 \times 10$ interpolation point after time $t$.

Figure 4.31 shows the interpolation results based on $10 \times 10$ wind speed data when $t = 0$.

The static soaring surveillance approach (Algorithm 1) is applied here to perform soaring surveillance in the simulated wind field. The UAV visits the predefined points in Region 1, $(a [-100, 100] m \times [-100, 100] m$ square), along the Dubins’ surveillance trajectory (Fig. 4.12). At the end of surveillance flight, vertical wind speed measurements are the training data for Gaussian process regression. Region 1 is further divided into many $5 m \times 5 m$ small cells. The vertical wind speed at each small cell is estimated by Gaussian process regression, so that the vertical wind speed map in Region 1 can be generated.

Figure 4.32 shows wind map results in Region 1 obtained by Gaussian process regression. Figure 4.33 provides the wind speed estimation error $|f - \bar{f}|$ superposed by the Dubins’ surveillance trajectory. Figure 4.33 depicts good estimation results in the neighborhood of the surveillance trajectory.

The cell which has the strongest upward wind speed $(-100, -40) m$ is the estimated location of the updraft center. The UAV loiters around the estimated center to gain altitude (potential energy). When the UAV reaches the top of the planetary boundary layer ($z = 1 km$), the UAV stops soaring and resumes surveillance flight in Region 2. Figure 4.34 demonstrates the effectiveness of the updraft soaring strategy.
Figure 4.29: Estimation sensitivity with respect to thermal’s oscillation frequency $b$ rad/s

Figure 4.35 shows the 3D updraft soaring path.

Figure 4.36, 4.38, 4.40, 4.42, 4.44, 4.46, 4.47, 4.48 show the Dubins' surveillance trajectory in each concentrically expanding region. According to the characteristics of the expanding region, the traveling distance of surveillance path increases along with the region expansion. If the UAV cannot visit every point with current potential energy, it will fly back to the updraft which was identified before to gain energy to carry on surveillance.

Figure 4.53 shows wind estimation results in the surveillance area (Region 1 to Region 8) obtained by Gaussian process regression with boundary constraints. Figure 4.54 provides the wind speed estimation error $|f - \bar{f}|$ in the surveillance area (Region 1 to Region 8).

In Fig. 4.37, 4.39, 4.41, 4.43, 4.45, 4.49, 4.50, 4.51, the airspeed which represents the kinetic energy of the UAV, and the altitude which shows the potential energy of the UAV. These flight states are calculated based on the maximum lift-to-drag ratio numerical analysis.

Remark. The speed changes during soaring and surveillance are caused by constantly switches between level and turning flight during Dubins' path. According to the results, lift coefficients along the trajectory are within the limits ($C_{L_{\text{max}}}$ = 1.5). Along with the expansion of region, the location with stronger upward winds keep identifying. Therefore, as illustrated in Fig. 4.52, the average soaring climb rate is steadily increasing from Region 2 to Region 4. From Region 5 to 9, the UAV performs static soaring around the updraft that was identified in Region 4. The different average soaring climb rate is caused by the different staring point in each region for soaring flight.
The vertical component of wind speed (m/s)

The horizontal component of wind speed (m/s)

Figure 4.30: Vertical and horizontal components of wind speed results on $10 \times 10$ interpolation points

Table 4.4: Updraft parameters

<table>
<thead>
<tr>
<th></th>
<th>$W_i$ (m/s)</th>
<th>$R_i$ (m)</th>
<th>$x_i$ (m)</th>
<th>$y_i$ (m)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>4.4</td>
<td>255.3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>6.1</td>
<td>211.1</td>
<td>-4846.0</td>
<td>2446.9</td>
</tr>
<tr>
<td>3</td>
<td>4.9</td>
<td>144.6</td>
<td>-4569.8</td>
<td>-3110.4</td>
</tr>
<tr>
<td>4</td>
<td>4.7</td>
<td>252.9</td>
<td>-3310.1</td>
<td>1867.8</td>
</tr>
<tr>
<td>5</td>
<td>4.8</td>
<td>183.2</td>
<td>1491.2</td>
<td>-3164.9</td>
</tr>
<tr>
<td>6</td>
<td>4.2</td>
<td>137.7</td>
<td>2317.2</td>
<td>-1315.2</td>
</tr>
<tr>
<td>7</td>
<td>5.0</td>
<td>284.7</td>
<td>1477.5</td>
<td>1256.2</td>
</tr>
<tr>
<td>8</td>
<td>5.0</td>
<td>269.0</td>
<td>-490.8</td>
<td>2802.3</td>
</tr>
<tr>
<td>9</td>
<td>6.3</td>
<td>187.5</td>
<td>470.1</td>
<td>-4188.7</td>
</tr>
<tr>
<td>10</td>
<td>6.2</td>
<td>205.6</td>
<td>2036.8</td>
<td>4293.9</td>
</tr>
</tbody>
</table>

4.3.2 Large wind field (10 km) results

The performance of the proposed approach depends on the effectiveness of the updraft identification approach (GPR). In order to demonstrate that thermals can be identified during surveillance, wind estimation results are presented in a $10 \text{ km} \times 10 \text{ km}$ wind field (Fig. 4.55), in which 10 thermals are distributed uniformly. The wind speed at location $(x, y)$ is calculated by the following mathematical updraft model.

$$W_z(x, y) = \sum_{i=1}^{10} W_i e^{-\frac{(x-x_i)^2+(y-y_i)^2}{(R_i)^2}} \left[1 - \frac{(x-x_i)^2+(y-y_i)^2}{(R_i)^2}\right]$$ (4.48)

The central strength $W_i$, size $R_i$, and location $x_i, y_i$ of each thermal in this field are defined in Table 4.4.

Figure 4.56, 4.57, and 4.58 illustrate wind field estimation results for the area of $4 \text{ km} \times 4 \text{ km}$, ...
Chapter 4. Static soaring surveillance in the quasi-static updraft field

Figure 4.31: Interpolation results based on $10 \times 10$ wind speed data when $t = 0$

$5.5 \text{ km} \times 5.5 \text{ km}$, and $10 \text{ km} \times 10 \text{ km}$ respectively. As long as thermals are identified, the UAV will perform static soaring to gain energy, and the surveillance region can keep extending concentrically.

4.4 Summary

In this chapter, a novel static soaring surveillance approach was proposed for a small UAV to perform the surveillance task while exploring opportunities to soar to extend flight endurance. In this approach, the surveillance problem was described as visiting prescribed locations (surveillance points). An optimal path was obtained by solving a traveling salesman problem, incorporating Dubins’ paths to take account of the UAV’s maneuvering dynamics. Further, a collection of so-called exploration points were strategically selected to achieve the surveillance task while maximizing the UAV’s flight coverage to discover the updraft. Once an updraft was identified, the UAV performed static soaring to gain energy, and to expand the surveillance region concentrically, allowing for extended and improved surveillance performance through a Hamiltonian circuit. A sensitivity study was then performed. This study described how sensitive the GPR result was to various cases of wind fields, and also provided a guideline for determining the number of measurements for GPR. The large eddy simulation (LES) model of the weather research and forecasting system (WRF-ARW) was utilized to generate the high-fidelity updraft field. Compared with the mathematical updraft models with an arbitrary strength, size, number and spatial distribution, the large eddy simulation (LES) model describes the physical laws governing the behavior and characteristics of updrafts. In this sense, the simulated wind field is a high-fidelity result. The simulated wind field was used as a test case to demonstrate the effectiveness of the proposed soaring surveillance approach. The field can be extended into large-sized field (e.g. a $10 \text{ km} \times 10 \text{ km}$ field).
Chapter 4. Static soaring surveillance in the quasi-static updraft field

Figure 4.32: Vertical wind estimation results in Region 1

Figure 4.33: Superposing surveillance trajectory over vertical wind speed estimation error $|f - \bar{f}|$ in Region 1

Figure 4.34: 2D updraft soaring path

Figure 4.35: 3D updraft soaring path
Figure 4.36: Dubins’ surveillance trajectory in Region 2

Figure 4.37: Flight states along the surveillance in Region 2

Figure 4.38: Dubins’ surveillance trajectory in Region 3

Figure 4.39: Flight states along the surveillance in Region 3
Chapter 4. Static soaring surveillance in the quasi-static updraft field

Figure 4.40: Dubins’ surveillance trajectory in Region 4

Figure 4.41: Flight states along the surveillance in Region 4

Figure 4.42: Dubins’ surveillance trajectory in Region 5

Figure 4.43: Flight states along the surveillance in Region 5
Chapter 4. Static soaring surveillance in the quasi-static updraft field

Figure 4.44: Dubins’ surveillance trajectory in Region 6

Figure 4.45: Flight states along the surveillance in Region 6

Figure 4.46: Dubins’ surveillance trajectory in Region 7
Figure 4.47: Dubins’ surveillance trajectory in Region 8

Figure 4.48: Dubins’ surveillance trajectory in Region 9
Figure 4.49: Flight states along the surveillance in Region 7

Figure 4.50: Flight states along the surveillance in Region 8

Figure 4.51: Flight states along the surveillance in Region 9

Figure 4.52: Average climb rate in each region
Figure 4.53: Wind speed estimation results in the simulated field

Figure 4.54: Wind speed estimation error $|f - \bar{f}|$ in the simulated field

Figure 4.55: 10 km $\times$ 10 km wind field

Figure 4.56: 4 km $\times$ 4 km estimation
Chapter 4. Static soaring surveillance in the quasi-static updraft field

Figure 4.57: 5.5 km × 5.5 km wind field

Figure 4.58: 10 km × 10 km estimation
Chapter 5

Dynamic soaring surveillance in a wind gradient field

Dynamic soaring surveillance utilizes the wind gradient which stretches out in the surface layer of the atmosphere to harvest extra energy to achieve long-endurance surveillance flight. In this chapter, first, the wind gradient model in the surface layer is presented. Secondly, the dynamic soaring surveillance problem is formulated. A Dubins-path-based method is then proposed to solve the energy-harvesting trajectory planning problem. Finally, results are presented to demonstrate the performance of the proposed approach.

5.1 Dynamic soaring surveillance problem formulation

In the surface layer of the atmosphere, the dominant wind pattern is the gradient wind which refers to the horizontal wind speed variation along the height. The wind speed usually slows down close to the ground because of the frictional force of the ground, and increases with height due to the pressure gradient force. In the current study, it is assumed that the wind gradient profile is known, and the effect of vertical winds is not considered.

5.1.1 Surveillance case

The origin of the coordinate system (X-Y-Z) is at the central point of the surveillance area (Fig. 5.1). Four points: Point 1: [0 m, 0 m, 0 m], 2: [0 m, -30 m, 0 m], 3: [20 m, 5 m, 0 m] and 4: [-8 m, 26 m, 0 m] are defined in the surveillance area. In the current case study, the area contains horizontal winds blowing from the negative to the positive of the x axis. As shown in Fig. 5.2, horizontal wind speed \( W_x \) is presented as [58]:

\[
\]
\[ W_x = \beta_{tr}(Az + \frac{1-A}{z_{tr}} z^2), \quad \frac{dW_x}{dz} = \beta_{tr} A + 2\beta_{tr} \frac{1-A}{z_{tr}} z \quad (0 \leq z \leq z_T) \]

\[ W_x = 0, \quad \frac{dW_x}{dz} = 0 \quad (z < 0) \]

\[ W_x = \beta_{tr}(Az_T + \frac{1-A}{z_{tr}} z_T^2), \quad \frac{dW_x}{dz} = 0 \quad (z > z_T) \]

Here \( A \) describes the exponential (\( A < 1 \)) or logarithmic-like (\( 1 < A < 2 \)) profile over altitudes. \( \beta_{tr} \) is the wind gradient slope at the characteristic altitude \( z_{tr} \), \( z_T = -\frac{Az_{tr}}{1-A} \) is the altitude where the wind gradient becomes zero (Fig. 5.2).

As mentioned in Chapter 2, a logarithmic (\( A = 1.2 \)) wind gradient profile with an arbitrary gradient slope (\( \beta_{tr} = 0.35 \, \text{s}^{-1}, \, z_{tr} = 30 \, \text{m} \)) is utilized as a numerical example to demonstrate the proposed approach. \( \beta_{tr} = 0.35 \, \text{s}^{-1} \) indicates a 10.5 m/s wind at the characteristic altitude \( z_{tr} = 30 \, \text{m} \). It is a case of strong wind shear. Later, Chapter 5 will discuss the impact of the wind shear strength on the amount of energy-gain in dynamic soaring flight.

**Remark.** As shown in Fig. 5.1, the altitude is defined relative to a reference plane. In the current study, the height of the reference plane is selected by taking account of the safe factor. The appropriate choice of height has been studied in boundary layer meteorology [10].

### 5.1.2 Problem formulation

According to the dynamic soaring rule in Chapter 2, the UAV exploits energy by repeatedly ascending upwind or descending downwind in the wind-gradient region as shown in Fig. 5.3.

In dynamic soaring surveillance, one cycle of dynamic soaring, which starts and finishes at the bottom of the wind-gradient region (\( z_{initial} = z_{final} = 0 \)), is utilized to connect every two surveillance targets.
Chapter 5. Dynamic soaring surveillance in a wind gradient field

Figure 5.3: Periodic dynamic soaring

\[ P(x_p, y_p), Q(x_q, y_q). \] In Chapter 3, the dynamic soaring surveillance problem was formulated as follows to achieve the maximum energy extraction:

\[
\begin{align*}
\text{maximize} & \quad [V(t_f) - V(0)] \\
\text{subject to} & \quad \text{Equations of motion (Eq. 5.3)} \\
& \quad 15 \text{ m/s} \leq V, \quad x(0) = x_p, \quad y(0) = y_p, \quad z(0) = 0 \text{ m} \\
& \quad x(t_f) = x_q, \quad y(t_f) = y_q, \quad z(t_f) = 0 \text{ m} \\
& \quad \gamma(0) = \gamma(t_f) = 0 \text{ rad, } \phi(0) = \phi(t_f) \\
& \quad C_{L_{\min}} \leq C_L \leq C_{L_{\max}}, \mu_{\min} \leq \mu \leq \mu_{\max}
\end{align*}
\]

where, \( P(x_p, y_p), Q(x_q, y_q) \) two surveillance points.

**Remark.** One dynamic soaring cycle starts with head-wind and ends with lee-ward heading. The Dubins’ path provides a feasible path connecting every two points with specified heading angles. Therefore, one cycle of dynamic soaring can connect every two points with the aid of Dubins’ path. However, the amount of energy-gain along the cycle depends on the distance between two points.
5.2 Dynamic soaring surveillance trajectory planning

5.2.1 The UAV’s trajectory

The UAV’s trajectory is generally determined by three dimensional positions \((x, y, z)\), airspeed \(V\), flight path angle \(\gamma\) and heading angle \(\varphi\). Flight path angle \(\gamma\) is the angle between the UAV’s velocity vector and the local horizontal (positive above the horizon). Heading angle \(\varphi\) is the angle between the UAV’s velocity vector projected onto the ground and the \(Y\) axis (clockwise rotation means positive change). In the current work, the UAV is assumed to stay aligned with its airspeed. With the effect of wind gradients (Eq. 5.1), the equations of motion for the small UAV can be derived as:

\[
\begin{align*}
\dot{V} &= -\frac{D}{m} - g \sin \gamma - \frac{dW_x}{dz} V \sin \gamma \cos \gamma \sin \varphi \\
\dot{x} &= V \cos \gamma \sin \varphi + W_x \\
\dot{y} &= V \cos \gamma \cos \varphi \\
\dot{z} &= V \sin \gamma \\
\end{align*}
\]

(5.3)

where \(L\) and \(D\) are aerodynamic lift and drag respectively, \(m\) is the mass of the UAV, and \(\frac{dW_x}{dz}\) is the wind gradient. From the equations of the derivatives of \(\gamma\) and \(\varphi\) in Eq.5.3, we obtain the expression of lift as:

\[
L^2 = (mV \dot{\gamma} + mg \cos \gamma - m \frac{dW_x}{dz} V \sin^2 \gamma \sin \varphi)^2 + (mV \cos \gamma \dot{\varphi})^2
\]

(5.4)

\(L\) and \(D\) are expressed as

\[
L = \frac{1}{2} \rho S V^2 C_L, \quad D = \frac{1}{2} \rho S V^2 C_D
\]

(5.5)

where drag coefficient \(C_D\) is related to lift coefficient \(C_L\) by:

\[
C_D = C_{D_0} + \frac{C_L^2}{\pi e AR}
\]

(5.6)

Here \(C_{D_0}\) is the parasite drag coefficient. \(AR\) is the aspect ratio, \(e\) is Oswald’s efficiency factor. In this thesis, we use the Aerosonde UAV model [2]. The parameters are given in Table 4.3 in Chapter 4. The UAV’s trajectory can be obtained by solving Eqs.5.3,5.4,5.5, and 5.6 based on specified values of \(\dot{\gamma}\), \(\dot{\varphi}\), and initial conditions \(V(0), \gamma(0), \varphi(0), x(0), y(0), z(0)\). In this chapter, \(\dot{\varphi}\) is calculated by Dubins’ paths analysis, and \(\dot{\gamma}\) is determined by an optimization process.
5.2.2 Dubins’ paths analysis

The Dubins’ path is a continuously differentiable curve to connect two arbitrary points with specified orientations. The principle result by Dubins [52] shows that, the shortest path can be selected from a Dubins-path candidate set [84], which is a combination of curved and straight-line segments.

\[ \text{Dubins-path set} = \{ lll, lrl, rrr, rlr, rll, lrr, rrl, rsl, lsr, lsl, rsr \} \]  \hspace{1cm} (5.7)

Here, \{l\} or \{r\} represents the curved segment that is part of a left or right turning circle with a certain radius, and \{s\} is the straight segment. Because the path that starts or ends with a straight line (e.g. \{sls\} or \{ssl\}) cannot be the shortest one (Lemma 1 [52]), the path candidate such as \{sls\} or \{ssl\} is not included in the Dubins-path set (Eq. 5.7).

**Remark.** In this chapter, a type of Dubins’ path (not necessarily to be the shortest path), along which the UAV can harvest energy from the wind gradient, is selected from the same Dubins-path set (Eq. 5.7) to determine the 2D surveillance trajectory on the reference plane. Using the same candidate set (Eq. 5.7) may exclude the choices such as \{sls\} or \{ssl\}. However, the flight-path-angle analysis in Section 5.2.3 shows that the Dubins’ path with straight lines is not an appropriate choice for the proposed approach (Fig. 5.5).

**Lemma.** Suppose that the orientations at initial and final points of the Dubins’ paths are defined as: \( \gamma_{\text{initial}} = \gamma_{\text{final}} = 0 \) and \( \phi_{\text{initial}} = 0, \phi_{\text{final}} = 0 \). The UAV can harvest energy from the wind gradient \( \frac{dW_x}{dz} > 0 \) when it files along the \{lll, lrl, lsl\}-type of Dubins’ path.

**Proof.** A flying bird or small UAV has both potential and kinetic energy:

\[ E = mgz + \frac{1}{2} m V^2 \]  \hspace{1cm} (5.8)

The specific energy can be defined as the total energy \( E \) per weight:

\[ e_s = \frac{E}{mg} = z + \frac{1}{2g} V^2 \]  \hspace{1cm} (5.9)

Taking the derivative of \( e_s \) with respect to time, we obtain:

\[ \dot{e}_s = - \frac{dW_x}{dz} \frac{V^2}{g} \sin \gamma \cos \gamma \sin \phi - \frac{DV}{mg} \]  \hspace{1cm} (5.10)

According to Eq.5.10, the term \( \frac{dW_x}{dz} \frac{V^2}{g} \sin \gamma \cos \gamma \sin \phi \) determines the energy gain from the wind. Under the condition of the wind gradient: \( \frac{dW_x}{dz} > 0 \), a relationship: \( \sin \gamma \sin \phi < 0 \) ensures a positive gain from the gradient wind.

Furthermore, according to the heading-angle definition, the right turning circle (i.e. clockwise rotation)
indicates a positive change of heading-angle rate and the left turning circle has a negative change of heading-angle rate. For the sake of positive energy gain, flight path angle $\gamma$ and heading angle $\varphi$ have to satisfy conditions in Fig. 5.4, and further describes in Table 5.1.

![Figure 5.4: Energy-gain flight in left-turning and right-turning circular flight](image)

**Table 5.1: The relationship between flight path angle and heading angle**

<table>
<thead>
<tr>
<th>Condition</th>
<th>Equation</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>if climb: $\gamma: 0 \leq \gamma &lt; \pi/2$ then: $\varphi: 0 \rightarrow -\pi$ (Left-turning)</td>
</tr>
<tr>
<td>2</td>
<td>if dive: $\gamma: -\pi/2 &lt; \gamma \leq 0$ then: $\varphi: -\pi \rightarrow -2\pi$ (Left-turning)</td>
</tr>
<tr>
<td>3</td>
<td>if dive: $\gamma: -\pi/2 &lt; \gamma \leq 0$ then: $\varphi: 0 \rightarrow \pi$ (Right-turning)</td>
</tr>
<tr>
<td>4</td>
<td>if climb: $\gamma: 0 \leq \gamma &lt; \pi/2$ then: $\varphi: \pi \rightarrow 2\pi$ (Right-turning)</td>
</tr>
</tbody>
</table>

With the orientation constraints: $\gamma_{\text{initial}} = \gamma_{\text{final}} = 0$ and $\varphi_{\text{initial}} = 0, \varphi_{\text{final}} = 0$, the UAV has to climb up from the initial point and dive back to the final point. As a result, flight path angles on the first and last segment of the path are $0 \leq \gamma < \pi/2$ and $-\pi/2 < \gamma \leq 0$ respectively. Because of the heading angle constraints: $\varphi_{\text{initial}} = 0, \varphi_{\text{final}} = 0$, we can conclude that the first and last segment of the path have to be left turning curves according to Table 5.1. Based on Table 5.1, heading angle constraints can be rewritten as: $\varphi_{\text{initial}} = 0, \varphi_{\text{final}} = 0 = -2\pi$ and the UAV can harvest energy from the wind gradient $dW/dz > 0$ when it flies along the $\{lll, lrl, lsl\}$-type of path.

### 5.2.3 Flight-path-angle-profile determination

Figure 5.5 shows the sketch of the possible relationship between flight-path-angle and heading-angle when the UAV flies over each of the $\{lll, lrl, lsl\}$-type of path.

For the type of path: $\{lll\}$, flight path angle $\gamma$ can be expressed as a function of heading angle $\varphi$, which can be described by a 4-th order B-spline curve [85] with 2 control points. Figure 5.6 shows a case of B-spline curve with two arbitrarily-chosen control points. The shape of the B-spline curve defines
the flight-path-angle profile, and further determines the height profile along the \{III\}-type of Dubins’ path. By optimizing the the B-spline curve (Eq. 5.11), the goal of energy-harvesting can be achieved. Therefore, the optimization problem (Eq. 5.2) becomes:

\[
\text{maximize } \left[ V(t_f) - V(0) \right] \\
\text{subject to}
\]

Equations of motion (Eq. 5.3)

Heading-angle-profile: \{III\}-type Dubins’ path

Flight-path-angle profile: the B-spline curve with 2 control points

\[15 \text{ m/s} \leq V, \quad x(0) = x_p, \quad y(0) = y_p, \quad z(0) = 0 \text{ m}\]

\[\phi(0) = 0 \text{ rad}, \quad \gamma(0) = 0 \text{ rad}\]

\[C_{L_{\text{min}}} \leq C_L \leq C_{L_{\text{max}}}, \quad \mu_{\text{min}} \leq \mu \leq \mu_{\text{max}}\]  

\[
(5.11)
\]

The \{III\}-type Dubins’ path determines the longitudinal and lateral motions on the reference plane to accomplish the surveillance cycle from Point \((x_p, y_p)\) to Point \((x_q, y_q)\). The vertical motion (height profile) along the Dubins’ path is optimized to achieve the goal of energy-gain. By this effort, the 3D trajectory planning problem is converted into a 1D (height) optimization problem, reducing computational complexity.

### 5.2.4 Dubins’ path choices: Heading angle rate determination

The heading-angle-profile can be determined by the \{III\}-type of Dubins’ path, however, the heading angle goes beyond the range \([0, -2\pi]\) when \(y_{\text{final}} > y_{\text{initial}}\). Case (a) and (b) in Fig. 5.7 show the two location relationships between two points. The \{III\}-type path chooses the internally tangent circle \((O_{\text{III}})\) with tangent points at point \(C_a\) and \(C_b\) to connect the small circles \(O_1, O_2\). Table 5.2 shows the corresponding \{III\}-type path between each pair of points. In Case (b), the heading angle goes beyond
the range $[0, -2\pi]$. Extra control points in B-spline curve are required to describe the flight-path-angle function when the heading angle is at $[-2\pi, -4\pi]$ (see the flight-path-angle function definition Fig. 5.6).

Two specific points \{lll\}-type path between each pair of points

<table>
<thead>
<tr>
<th>Two specific points</th>
<th>{lll}-type path</th>
<th>Heading-angle</th>
</tr>
</thead>
<tbody>
<tr>
<td>Case (a)</td>
<td>$1 \rightarrow C_a \rightarrow C_b \rightarrow 2$</td>
<td>$0 \rightarrow -2\pi$</td>
</tr>
<tr>
<td>Case (b)</td>
<td>$1 \rightarrow C_b \rightarrow C_a \rightarrow 2$</td>
<td>$0 \rightarrow -2\pi \rightarrow -4\pi$</td>
</tr>
</tbody>
</table>

To this end, the \{ll\}-type of Dubins’ path (Case (c) in Fig. 5.7) is proposed to address the issues in the case when $y_{final} > y_{initial}$. According to the third plot in Fig. 5.7, the \{ll\}-type path uses the internally tangent circle $O_{ll}$ with tangent points at point 1 and C to connect Point 1 and the small circle $O_1, O_2$.

We designate the same radii of circles $O_1, O_2$ as $R_{ll}$. Therefore, $R_{ll}$ (Case (c)) and $R_{lll}$ (Case (a)) in
Fig. 5.7 can be expressed as:

\[
R_{lll} = 0.5 \sqrt{((X_1 - X_2)^2 + (Y_1 - Y_2)^2) + R_I}
\]

\[
R_{ll} = \frac{R_{lll} - R_I}{\cos \alpha_T} + R_I
\]  \(5.12\)

where \(\alpha_T\) is the angle between line \(O_1O_2\) and the x axis. \(\alpha_T = \arctan \left| \frac{Y_2 - Y_1}{X_2 - X_1} \right|\). According to Eq. 5.12, \(R_{ll} = R_{lll}\) when \(\alpha_T = 0\) and \(R_{ll} = \infty\) when \(\alpha_T = \frac{\pi}{2}\).

Table 5.3 is proposed to choose \{\(lll\)\}-type or \{\(ll\)\}-type path based on the location relationship between Points 1 \((X_{initial}, Y_{initial})\) and 2 \((X_{final}, Y_{final})\).

<table>
<thead>
<tr>
<th>Location relationship between Point 1 and Point 2</th>
<th>Dubins’ path choice</th>
</tr>
</thead>
<tbody>
<tr>
<td>((Y_{final} \leq Y_{initial}))</td>
<td>{(lll)}-type</td>
</tr>
<tr>
<td>((Y_{final} &gt; Y_{initial}))</td>
<td>{(ll)}-type</td>
</tr>
</tbody>
</table>

Based on the wind measurement \(\hat{\dot{W}_x}\), the appropriate heading-rate \(\dot{\phi}\) can be obtained by solving the following equations (Eq. 5.13):

\[
V_g = \sqrt{V^2 \cos^2 \gamma + \hat{\dot{W}_x}^2 + 2\hat{\dot{W}_x} V \cos \gamma \cos \varphi}
\]

\[
\sigma = \sin^{-1}\left(\frac{\hat{\dot{W}_x}}{V_g} \cos \varphi\right)
\]

\[
\dot{\phi} = \dot{\theta} - \dot{\sigma} = \frac{V_g}{R} - \dot{\sigma}
\]  \(5.13\)

Here, ground speed \(V_g\), airspeed \(V\), flight path angle \(\gamma\), and heading angle \(\varphi\) are shown in Fig. 5.8. \(R\) is the desired turning radius (Eq.5.12) that is either \{\(lll\)\}-type or \{\(ll\)\}-type types of Dubins’ path (Fig. 5.7).

### 5.2.5 Dynamic soaring trajectory planning approach

The UAV visits from one point to another by one cycle of dynamic soaring (headwind climb and leeward dive) to take advantage of complementary features. The soaring surveillance trajectory can be obtained by solving Eqs.5.3,5.4,5.5, and 4.14 based on specified profiles of flight path angle \(\gamma\) and heading angle \(\varphi\), and initial conditions \(V(0), \gamma(0), \varphi(0), x(0), y(0), z(0)\). According to Fig. 5.7 and Eq. 5.12, the heading-angle profile (Eq. 5.13) is a function of \(R_I\), the radius of the small circle in \{\(lll\)\} and \{\(ll\)\}-type of Dubins’ path. The two control points of the B-spline curve determine the flight-path-angle profile.

As a result, the soaring surveillance trajectory problem (Eq. 5.2) can therefore be represented as: regulating the 2 control points, initial speed \(V(0)\), and the Dubins’ circle radius \(R_I\) to maximize the following cost function:
maximize \[ [V(t_f) - V(0)] \]
subject to

Equations of motion: Eq. 5.3

Heading-angle-profile: Eq. 5.13

Flight-path-angle profile: the B-spline curve with 2 control points

\[ 15 \text{ m/s} \leq V(t), \ x(0) = x_p, \ y(0) = y_p, \ z(0) = 0 \text{ m} \]

\[ \varphi(0) = 0 \text{ rad}, \ \gamma(0) = 0 \text{ rad} \]

\[ C_{L_{\min}} \leq C_L \leq C_{L_{\max}}, \ \mu_{\min} \leq \mu \leq \mu_{\max} \]

In Eq. 5.14, the bank angle (\( \mu \)) can be computed by the following equations:

\[ \mu = \arctan \left( \frac{L_2}{L_1} \right) \]

where \( L_1 = mV\dot{\gamma} + mg \cos \gamma - mC_{W_2}V^2 \gamma \sin \varphi \), \( L_2 = mV \cos \gamma \dot{\varphi} + mC_{W_2}V \sin \gamma \cos \varphi \). In the current work, we choose: \( C_{L_{\min}} = -0.3 \), \( C_{L_{\max}} = 1.5 \), \( \mu_{\min} = -65^\circ \), and \( \mu_{\max} = 65^\circ \). The optimization problem is solved by the MATLAB fmincon subroutine.

If the constraint of Dubins' parameterization is removed, the trajectory planning problem can be solved traditionally. First, a series of N nodes are defined uniformly along a candidate trajectory. Secondly, at each node, the values of state variables can be defined as:

\[ [V_i, x_i, y_i, z_i, \gamma_i, \varphi_i, C_{L_i}, \mu_i]_{(i=1:N)} \]
Here \( i = 1 \) and \( i = N \) are the initial and final point of the trajectory. The maximum energy extraction: \((V_N - V_1)\) is achieved by regulating the values of state variables, which in the meanwhile satisfy the equations of motion (Eq. 5.3), and the following path constraints:

\[
C_L_{\text{min}} \leq [C_L]_{i=1}^{N} \leq C_L_{\text{max}} \\
\mu_{\text{min}} \leq [\mu]_{i=1}^{N} \leq \mu_{\text{max}} \\
[z]_{i=1}^{N} \geq 0 \text{ m} \\
[V]_{i=1}^{N} \geq 15 \text{ m/s} \\
z_1 = z_N = 0 \text{ m}
\] (5.16)

**Remark.** The traditional method (without Dubins’ parameterization) searches for the energy-optimal trajectory in the three-dimensional space. According to simulation evaluations, the traditional method can harvest energy from weak wind gradients (e.g. \( \beta_{tr} = 0.08 \text{ s}^{-1} \), implying \( 2.4 \text{ m/s} \) winds at 30 m). Because of the large number of parameters in the optimization process (Eq. 5.16), the traditional approach is computationally expensive. Compared with the traditional approach, which has to optimize a three-dimensional trajectory, the proposed method (Eq. 5.14) simplifies the three-dimensional trajectory planning problem into a one dimension (height profile) optimization problem with the aid of Dubins’ parameterization. The predefined Dubins path may result in a sub-optimal solution. However, the proposed method reduces the complexity of the optimization problem significantly. According to simulation results, the Dubins-path-based approach is applicable for mild or strong wind gradients cases (e.g. \( \beta_{tr} \geq 0.25 \text{ s}^{-1} \), implying \( 7.5 \text{ m/s} \) winds at 30 m).

The dynamic soaring surveillance approach (Algorithm 2) can be presented as follows.

**Algorithm 2: Dynamic soaring surveillance approach**

1. Determine the optimal visiting sequence of the surveillance points
2. Design one dynamic soaring cycle to connect every two points
3. The cycles are connected end-to-end to form a complete trajectory

### 5.3 Simulation results

In the following sections, the case of multiple points surveillance problem (Fig. 5.1) is presented to demonstrate the proposed soaring surveillance approach.

#### 5.3.1 The optimal visiting sequence

The turning radius of the large circle in \( \{lll\} \) or \( \{ll\} \)-type of Dubins’ path is determined by the following equation (Fig. 5.7):
\[ R_{ll} = 0.5 \sqrt{((X_1 - X_2)^2 + (Y_1 - Y_2)^2)} + R_I \]
\[ R_{ll} = \frac{R_{ll} - R_I}{\cos \alpha_T} + R_I \] (5.17)

where \( \alpha_T = \arctan \frac{Y_2 - Y_1}{X_2 - X_1} \), and \( R_I \) is the radius of the small circle in Dubins’ path.

In order to avoid a very large turning radius in {lll} and {ll}-type of Dubins’ path, a traveling salesman problem is formulated as:

Let \( x \) represent a candidate sequence (sometimes called a sample sequence), e.g., \( x = \{x_1 \rightarrow x_2 \rightarrow \ldots \rightarrow x_n \rightarrow x_1\} \). Then the overall cost of the sequence is

\[ S(x) = \sum_{i=1}^{n-1} c_{x_i, x_{i+1}} + c_{x_n, x_1} \] (5.18)

Here, suppose that \( x_i = (X_1, Y_1) \), \( x_{i+1} = (X_2, Y_2) \), and \( \alpha_T = \arctan \frac{Y_2 - Y_1}{X_2 - X_1} \), the cost \( c_{x_i, x_{i+1}} \) is defined as:

\[ c_{x_i, x_{i+1}} = \begin{cases} R_{ll} = 0.5 \sqrt{((X_1 - X_2)^2 + (Y_1 - Y_2)^2)} + R_I & \text{if } Y_2 \leq Y_1, \ X_1 \neq X_2; \\
R_{ll} = \frac{R_{ll} - R_I}{\cos \alpha_T} + R_I & \text{if } Y_2 > Y_1, \ X_1 \neq X_2; \\
\infty & \text{if } Y_2 > Y_1, \ X_1 = X_2. \end{cases} \]

The optimal path (sequence) \( x^* \) minimizes the overall cost:

\[ x^* = \arg \min_{x \in X} S(x) = \arg \min \left( \sum_{i=1}^{n-1} c_{x_i, x_{i+1}} + c_{x_n, x_1} \right). \] (5.19)

Apply the cross-entropy method to solve the proposed traveling salesman problem, the optimal visiting sequence of the four surveillance points (Fig. 5.1) is presented by Fig. 5.9.

The optimal visiting sequence is: point 1: \([0 \ m, 0 \ m, 0 \ m]\) to 2: \([0 \ m, -30 \ m, 0 \ m]\) to 3: \([20 \ m, 5 \ m, 0 \ m]\) to 4: \([-8 \ m, 26 \ m, 0 \ m]\) to point 1: \([0 \ m, 0 \ m, 0 \ m]\).

5.3.2 Initial conditions

According to the optimization problem (Eq. 5.14), six parameters (two control points’ position, initial airspeed \( V(0) \) and \( R_I \)) need to be optimized. The initial condition for the optimization problem (Eq. 5.14) is chosen as:

\[ X_{\text{ini}} = [\frac{\pi}{4}, \frac{\pi}{36}, \frac{3\pi}{2}, -\frac{\pi}{36}, 45 \ m/s, 60 \ m] \] (5.20)

The first four parameters in Eq. 5.20 are the control points of the B-spline curve, which determines the flight path angle profile. The fifth parameter is the initial airspeed. The sixth parameter is the small
turning radius of the Dubins’ path, which determines heading angle profile.

For the segments from Point 2 to Point 3, Point 3 to Point 4, and Point 4 to Point 1, in order to maintain a consistent airspeed profile, the initial airspeed \( V(0) \) of each segment is defined as: \( V(0) \) \text{Point i to Point i+1} = \( V(t_f) \) \text{Point i-1 to Point i} \quad (i \geq 2). \) The other parameters of the initial condition remain the same as the ones in Eq. 5.20. As a result, for these three segments, five parameters (two control points' positions and \( R_I \)) needs to be optimized.

### 5.3.3 Dynamic soaring surveillance simulation results

#### Sensitivity analysis

Figure 5.10 illustrates the change of the energy height along the dynamic soaring surveillance trajectory in three wind gradient cases. Table 5.4 provides the amount of energy that can be achieved at the end of each dynamic soaring cycle. The soaring cycle from Point 1 to 2 can harvest energy in these three cases, however the cycle from Point 2 to 3 cannot gain energy at all. The results demonstrate that the amount of energy-gain depends not only on the strength of the wind gradient, but also on the initial airspeed of the cycle and the location relationship between two surveillance points.

The soaring cycle from Point 1 to 2 can search for the optimal initial airspeed to achieve the goal of energy-gain. However, in order to maintain a consistent airspeed profile, the initial airspeed \( V(0) \) of
Table 5.4: Energy harvesting results between two points

<table>
<thead>
<tr>
<th>Wind gradients (s⁻¹)</th>
<th>Energy-harvesting results (m)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Point 1 to 2</td>
<td>13.67 m</td>
</tr>
<tr>
<td>Point 2 to 3</td>
<td>-8.49 m</td>
</tr>
<tr>
<td>Point 3 to 4</td>
<td>2.65 m</td>
</tr>
<tr>
<td>Point 4 to 1</td>
<td>17.72 m</td>
</tr>
<tr>
<td>0.35</td>
<td>1.3 m</td>
</tr>
<tr>
<td>-1</td>
<td>-0.67 m</td>
</tr>
<tr>
<td>13.67 m</td>
<td>2.65 m</td>
</tr>
<tr>
<td>17.72 m</td>
<td>-0.35 m</td>
</tr>
<tr>
<td>0.25 s⁻¹</td>
<td>1.3 m</td>
</tr>
<tr>
<td>-1</td>
<td>-1.88 m</td>
</tr>
<tr>
<td>13.67 m</td>
<td>-6.5 m</td>
</tr>
<tr>
<td>0.2 s⁻¹</td>
<td>1.3 m</td>
</tr>
<tr>
<td>-1</td>
<td>-5.08 m</td>
</tr>
<tr>
<td>13.67 m</td>
<td>-6.5 m</td>
</tr>
</tbody>
</table>
| cycles Point 2 to 3, 3 to 4, and 4 to 1, are defined as: $V(0)^{\text{Point i to Point i+1}} = V(t_f)^{\text{Point i-1 to Point i}}(i \geq 2)$. The predetermined initial airspeed restricts the optimization process to achieve the promising results. In addition to the initial airspeed, the location relationship also affects the energy-gain result. For instance, the cycle Point 2 to 3 cannot gain energy at all, however, cycles from Point 3 to 4, and 4 to 1 can gain energy in the case of strong winds (e.g. $\beta_{tr} = 0.35 \text{ s}^{-1}$).

Table 5.5 gives the the amount of energy that can be achieved at the end of dynamic soaring surveillance flight ($1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 1$). According to Table 5.5, dynamic soaring surveillance is applicable for mild or strong wind gradient cases (i.e. $\beta_{tr} > 0.25 \text{ s}^{-1}$).

Figure 5.10: Energy change along dynamic soaring surveillance trajectory in various wind gradient cases
Table 5.5: Energy harvesting results

<table>
<thead>
<tr>
<th>Wind gradients</th>
<th>Wind speed at 30 m</th>
<th>Energy</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.35 s⁻¹</td>
<td>10.5 m/s</td>
<td>25.55 m</td>
</tr>
<tr>
<td>0.25 s⁻¹</td>
<td>7.5 m/s</td>
<td>-1.58 m</td>
</tr>
<tr>
<td>0.2 s⁻¹</td>
<td>6 m/s</td>
<td>-15.27 m</td>
</tr>
</tbody>
</table>

Simulation results: the strong wind case (\(\beta_{tr} = 0.35\) s⁻¹)

Figure 5.11 shows the UAV’s total energy (potential and kinetic energy) change along the soaring trajectory. At the end of surveillance, the amount of energy harvested from the wind is 25.55 m. It implies that the UAV can perform the surveillance task by harvesting energy from the wind gradient. For the sake of comparison, Fig. 5.12 shows UAV’s total energy (potential and kinetic energy) change along the gliding trajectory. In this case, the UAV’s total energy drops quickly along the gliding path.

In Fig. 5.13, the green, black, brown, and red line show the energy-efficient surveillance trajectory connecting every two consecutive surveillance points respectively. The designed Dubins’ trajectories are connected end-to-end to form a complete soaring surveillance trajectory. Figure 5.14 shows that the ground path of the trajectory is \(\{lll\}\)-type or \(\{ll\}\)-type of Dubins’ path. The surveillance trajectory passes through the predefined four surveillance points where the UAV is required to visit to collect up-to-date information. Figure 5.15 shows the three dimensional location along the complete surveillance trajectory. Figure 5.16 shows the airspeed, flight path angle and heading angle along the surveillance trajectory. Figure 5.17 shows the angle of attack and bank angle along the surveillance trajectory.

Remark. According to the sensitivity analysis, the minimum wind shear for feasible dynamic soaring in this surveillance case is \(\beta_{tr} = 0.25\) s⁻¹. The instantaneous changes in bank angle and angle of attack are caused by turning radius changes along different turning portions in Dubins’ path. The instantaneous changes exceed rolling rate limit. This may result in unsatisfied tracking performance. The future study of an integrated trajectory and tracking approach can address this issue by taking account of the dynamic characteristics of the aircraft (e.g. rolling rate limit) in soaring trajectory planning.

5.4 Summary

This chapter investigated how to obtain the energy-efficient trajectory for the small UAV to perform the multiple-point surveillance task. During the investigation, the Dubins’ path analysis was used to determine the 2D trajectory along the reference plane. The residual dimension (height) can be further described by the B-spline curve. By choosing the optimal B-spline curve, the energy-efficient trajectory can be obtained. The proposed approach, which defined the 2D trajectory on the reference plane as Dubins’ path, may not provide the optimal solution. However, compared with the traditional trajectory planning approach, which optimized a 3D trajectory, the proposed approach reduced the optimization
Figure 5.11: Energy change along dynamic soaring surveillance trajectory

problem complexity. Simulation results showed that along the soaring surveillance trajectory, the UAV can take advantage of the wind to perform the surveillance task. In this sense, endurance performance can be enhanced with limited on-board energy supply.
Figure 5.12: Energy change along gliding surveillance trajectory
Chapter 5. Dynamic soaring surveillance in a wind gradient field

Figure 5.13: 3D dynamic soaring surveillance trajectory
Figure 5.14: Dynamic soaring surveillance trajectory’s ground path
Figure 5.15: Three-dimensional location along the trajectory from point 1 to 2 (green line), point 2 to 3 (black line), point 3 to 4 (brown line) and point 4 to 1 (red line)
Figure 5.16: Airspeed, flight path angle and heading angle along the trajectory from point 1 to 2 (green line), point 2 to 3 (black line), point 3 to 4 (brown line) and point 4 to 1 (red line)
Figure 5.17: Angle of attack and bank angle along the trajectory
Chapter 6

Trajectory tracking controller design

Soaring requires spatial or temporal maneuvers to harvest additional kinetic or potential energy from the wind. The problem associated with those maneuvers is significant coupling dynamics and nonlinearity. In order to achieve good tracking performance, nonlinear control design techniques are chosen to address the coupling and nonlinearity issues. Figure 6.1 demonstrates the relationship between path planning and tracking controller. The trajectory tracking controller is designed using the state-dependent Riccati equation (SDRE) method based on a full six-degree-of-freedom nonlinear flight dynamics model, which is implemented through the multiple-loop control structure.

This chapter is organized as follows. In the first section, the nonlinear flight dynamics model is presented in detail. The second section presents the multiple-loop structure of the tracking controller design, and the SDRE-based controller which can provide the best trade-off between the tracking performance and achievable control inputs. In the final section, simulation results demonstrate the performance of energy-efficient trajectory tracking.

6.1 Mathematical model

The full six-degree-of-freedom, nonlinear, rigid body flight dynamics model [86] is presented as:
Chapter 6. Trajectory tracking controller design

Figure 6.1: Trajectory planning and control

\[\dot{V} = \left(-\frac{D}{m} + \frac{Y}{m} \sin \beta - gs \sin \gamma - \frac{dW_x}{dz} V \sin \gamma \cos \gamma \sin \varphi \right)\]

\[\dot{\gamma} = \frac{L \cos \mu - mg \cos \gamma - Y \sin \mu \cos \beta + m \frac{dW_x}{dz} \frac{V \sin^2 \gamma \sin \varphi}{mV}}{mV \cos \gamma}\]

\[\dot{\varphi} = \frac{L \sin \mu + Y \cos \mu \cos \beta - m \frac{dW_x}{dz} \frac{V \sin \gamma \cos \varphi}{mV}}{mV \cos \gamma}\]

\[\dot{\alpha} = q - \tan \beta (p \cos \alpha + r \sin \alpha) + \frac{1}{m V \cos \beta} (-L + mg \cos \mu)\]

\[\dot{\beta} = p \sin \alpha - r \cos \alpha + \frac{1}{mV} (mg \cos \gamma \sin \mu + Y \cos \beta)\]

\[\dot{\mu} = \sec \beta (p \cos \alpha + r \sin \alpha) + \frac{L}{mV} (\tan \gamma \sin \mu + \tan \beta) + \frac{Y}{mV} \tan \gamma \cos \mu \cos \beta - \frac{q}{V} \cos \gamma \cos \mu \cotan \beta\]

\[\dot{\phi} = c_1 \dot{\phi} + c_2 \dot{\dot{\phi}} + c_3 p q + c_4 q r\]

\[\dot{\psi} = \frac{1}{I_{yy}} (\ddot{\theta} + (I_{xx} - I_{zz}) \dot{p} + I_{xx} (r^2 - p^2))\]

\[\dot{\rho} = c_5 \dot{\rho} + c_6 \dot{\rho} - c_7 q r + c_8 p q\]  \hspace{1cm} (6.1)

where the angles \(\alpha, \beta, \mu\) are defined as the angle of attack, slide angle, and bank angle, respectively. \(p, q, r\) are roll, pitch, and yaw rates defined in the body axes. \(c_1, c_2, c_3, c_4, c_5, c_6, c_7, c_8\) and aerodynamic
coefficients are defined as:

\[
\begin{align*}
    c_1 &= \frac{I_{zz}}{I_{xx}I_{zz} - I_{xz}^2}, \\
    c_2 &= \frac{I_{xx}I_{yy} + I_{zz}}{I_{xx}I_{zz} - I_{xz}^2}, \\
    c_3 &= \frac{I_{xx}}{I_{xx}I_{zz} - I_{xz}^2}, \\
    c_4 &= \frac{I_{xx}I_{yy} - I_{zz}}{I_{xx}I_{zz} - I_{xz}^2}, \\
    c_5 &= \frac{I_{xx}(I_{xx} - I_{yy}) + I_{zz}^2}{I_{xx}I_{zz} - I_{xz}^2}, \\
    c_6 &= \frac{I_{xx}I_{yy} - I_{zz}}{I_{xx}I_{zz} - I_{xz}^2}, \\
    c_7 &= \frac{I_{xx}I_{yy} + I_{zz}}{I_{xx}I_{zz} - I_{xz}^2}, \\
    c_8 &= \frac{I_{xx}I_{yy} + I_{zz}}{I_{xx}I_{zz} - I_{xz}^2}.
\end{align*}
\] (6.2)

Total aerodynamic forces and moments are given as follows [2]:

\[
\begin{align*}
    L &= \frac{1}{2} \rho V^2 SC_L, \\
    D &= \frac{1}{2} \rho V^2 SC_D, \\
    Y &= \frac{1}{2} \rho V^2 SC_Y, \\
    \bar{l} &= \frac{1}{2} \rho V^2 SbC_l, \\
    \bar{m} &= \frac{1}{2} \rho V^2 S \bar{c} C_m, \\
    \bar{n} &= \frac{1}{2} \rho V^2 SbC_n
\end{align*}
\]

where [2]

\[
\begin{align*}
    C_L &= C_{L_0} + C_{L_\alpha} \alpha + C_{L_\delta_c} \delta_c \\
    C_D &= C_{D_0} + \frac{C_L^2}{\pi e AR} \\
    C_Y &= C_{Y_\beta} \beta + C_{Y_\delta_r} \delta_r \\
    C_l &= C_{l_\alpha} \alpha + C_{l_{1/2}} \frac{pb}{2V} + C_{l_{1/3}} \frac{rb}{2V} + C_{l_{1/4}} \frac{rb}{V} + C_{l_{1/5}} \delta_r + C_{l_{1/6}} \delta_a \\
    C_m &= C_{m_\alpha} + C_{m_\alpha} \alpha + C_{m_\alpha} \frac{qc}{2V} + C_{m_{1/4}} \delta_c \\
    C_n &= C_{n_\alpha} + C_{n_\alpha} \alpha + C_{n_\alpha} \frac{q_{1/2}}{2V} + C_{n_{1/3}} \frac{rb}{2V} + C_{n_{1/4}} \delta_r + C_{n_{1/5}} \delta_a
\end{align*}
\] (6.3)

The control inputs are \(\delta_\alpha, \delta_\beta, \delta_r\) representing the aileron, elevator, and rudder respectively.

The trajectory tracking controller based on the full nonlinear dynamics model (Eq. 6.1) is difficult to design and implement. To this end, the multiple-loop control technique is used here to reduce model complexity. The technique separates the system states in Eq. 6.1 into two groups: command-generator-loop states \((V, \gamma, \varphi, \alpha, \beta, \mu)\) and inner-loop states \((p, q, r)\) based on the fact that actuator inputs \(\delta_\alpha, \delta_\beta, \delta_r\) have direct effects on angular rates dynamics \((p, q, r)\). In the inner loop, the state-dependent Riccati equation (SDRE) approach [72] is utilized to design the optimal nonlinear controller to follow the commands \(p_c, q_c, r_c\). The commands \(p_c, q_c, r_c\) are generated based on the trajectory planning commands: \([\gamma_c, \varphi_c, \dot{\gamma}_c, \dot{\varphi}_c, V_c]\) and their derivatives via a nonlinear mapping in the command generator loop. Figure 6.1 shows the overall architecture of the tracking control design.
6.2 Trajectory tracking controller design

6.2.1 Nonlinear mapping in the outer loop

The nonlinear mapping between \( \alpha_c, \beta_c, \mu_c \) and the derivatives of trajectory planning commands: \([V_c, \gamma_c, \varphi_c, \dot{\gamma}_c, \dot{\varphi}_c] \) in can be presented as:

\[
\alpha_c = \frac{(C_L - C_{L_0})}{C_L} \\
\beta_c = 0 \\
\mu_c = \tan(\frac{L_2}{L_1}) \tag{6.4}
\]

where:

\[
L_1 = mV_c \gamma_c \cos\gamma_c - m \frac{dW_x}{dz} V_c \sin^2\gamma_c \sin\varphi_c \tag{6.5}
\]

\[
L_2 = mV_c \cos\gamma_c \dot{\varphi}_c + m \frac{dW_x}{dz} V_c \sin\gamma_c \cos\varphi_c
\]

The nonlinear mapping between commands \( p_c, q_c, r_c \) and commands: \([\gamma_c, \varphi_c, \alpha_c, \beta_c, \mu_c], [\dot{\alpha}_c, \dot{\beta}_c, \dot{\mu}_c] \) can be presented as:

\[
\begin{bmatrix}
  p_c \\
  q_c \\
  r_c
\end{bmatrix} =
\begin{bmatrix}
  \cos \alpha & 0 & \sin \alpha \\
  -\cos \tan \beta & 1 & \sin \tan \beta \\
  \sin \alpha & 0 & -\cos \alpha
\end{bmatrix}^{-1}
\begin{bmatrix}
  \dot{\alpha}_c \\
  \dot{\beta}_c \\
  \dot{\mu}_c
\end{bmatrix} -
\begin{bmatrix}
  M_1 \\
  M_2 \\
  M_3
\end{bmatrix} \tag{6.6}
\]

where,

\[
M_1 = \frac{\cos \mu}{\cos \beta} \gamma_c - \frac{\cos \gamma \sin \mu}{\cos \beta} \dot{\phi}_c + \frac{dW_x}{dz} \sin \gamma \cos \varphi + \frac{\cos \mu}{\cos \beta}
\]

\[
M_2 = -\sin \mu \gamma_c - \cos \gamma \cos \mu \dot{\phi}_c - \frac{dW_x}{dz} \sin \gamma \cos \varphi \sin \mu + \frac{dW_x}{dz} \sin \gamma \cos \mu \sin \varphi \tag{6.7}
\]

\[
M_3 = \cos \tan \beta \gamma_c + (\sin \gamma + \cos \gamma \sin \tan \beta) \dot{\phi}_c + \frac{dW_x}{dz} \cos \tan \beta \sin \gamma \cos \varphi
\]

\[
+ \frac{dW_x}{dz} \sin \gamma \sin \tan \beta \sin \varphi + \frac{dW_x}{dz} \sin \gamma \cos \gamma \sin \varphi \tag{6.8}
\]

The nonlinear mapping (Eq. 6.6) can be derived based on the following three steps.
Chapter 6. Trajectory tracking controller design

First of all, the derivatives of \((\gamma, \varphi)\) in Eq. 6.1 can be rewritten as:

\[
\dot{\gamma} = \frac{L \cos \mu}{mV} - \frac{g \cos \gamma}{V} - \frac{Y}{mV} \sin \mu \cos \beta + \frac{dW_x}{dz} \sin^2 \gamma \sin \varphi \tag{6.10}
\]

\[
\dot{\varphi} = \frac{L \sin \mu}{mV \cos \gamma} + \frac{Y \cos \mu \cos \beta}{mV \cos \gamma} - \frac{dW_x}{dz} \tan \gamma \cos \varphi \tag{6.11}
\]

Secondly, by manipulating the equations Eq. 6.10 and Eq. 6.11 as follow:

Multiply both sides of Eq. 6.10 by \(-\frac{\cos \mu}{\cos \beta}\):

\[
-\frac{\cos \mu}{\cos \beta} \dot{\gamma} = -\frac{L \cos^2 \mu}{mV \cos \beta} + \frac{Y}{mV} \sin \mu \cos \mu + \frac{g \cos \gamma \cos \mu}{V \cos \beta} - \frac{dW_x}{dz} \sin^2 \gamma \sin \varphi \cos \mu \cos \beta \tag{6.12}
\]

Multiply both sides of Eq. 6.11 by \(-\frac{\cos \gamma \sin \mu}{\cos \beta}\):

\[
-\frac{\cos \gamma \sin \mu}{\cos \beta} \dot{\varphi} = -\frac{L \sin^2 \mu}{mV \cos \beta} - \frac{Y}{mV} \sin \mu \cos \mu + \frac{dW_x}{dz} \sin \gamma \sin \mu \cos \varphi \tag{6.13}
\]

By adding Eq. 6.12 to Eq. 6.13, one can obtain:

\[
-\frac{\cos \mu}{\cos \beta} \dot{\gamma} - \frac{\cos \gamma \sin \mu}{\cos \beta} \dot{\varphi} = \frac{L}{mV \cos \beta} + \frac{g \cos \gamma \sin \mu}{V \cos \beta} - \frac{dW_x}{dz} \sin^2 \gamma \sin \varphi \cos \mu \cos \beta + \frac{dW_x}{dz} \sin \gamma \sin \mu \cos \varphi \tag{6.14}
\]

Multiply both sides of Eq. 6.10 by \(-\sin \mu\):

\[
-\sin \mu \dot{\gamma} = -\frac{L \sin \mu \cos \mu}{mV} + \frac{Y}{mV} \sin \mu \cos^2 \beta + \frac{g \cos \gamma \sin \mu}{V} \tag{6.15}
\]

Multiply both sides of Eq. 6.11 by \(\cos \gamma \cos \mu\):

\[
\cos \gamma \cos \mu \dot{\varphi} = \frac{L \sin \mu \cos \mu}{mV} + \frac{Y}{mV} \cos \mu \cos^2 \beta - \frac{dW_x}{dz} \sin \gamma \cos \mu \cos \varphi \tag{6.16}
\]

By adding Eq. 6.15 to Eq. 6.16, one can obtain

\[
-\sin \mu \dot{\gamma} - \cos \gamma \cos \mu \dot{\varphi} = \frac{Y}{mV} \cos \beta + \frac{g \cos \gamma \sin \mu}{V} - \frac{dW_x}{dz} \sin^2 \gamma \sin \varphi \sin \mu - \frac{dW_x}{dz} \sin \gamma \cos \mu \cos \varphi \tag{6.17}
\]
Multiply both sides of Eq. 6.10 by $\cos \tan \beta$:

$$
\cos \tan \beta \dot{\gamma} = \frac{L}{mV} \cos^2 \tan \beta - \frac{Y}{mV} \sin \mu \cos \mu \sin \beta - \frac{g \cos \gamma}{V} \cos \tan \beta
$$

$$+ \frac{dW_x}{dz} \cos \mu \tan \beta \sin^2 \gamma \sin \varphi \quad (6.18)
$$

Multiply both sides of Eq. 6.11 by $(\sin \gamma + \cos \gamma \sin \mu \tan \beta)$:

$$(\sin \gamma + \cos \gamma \sin \mu \tan \beta) \dot{\phi} = \frac{L}{mV} \tan \gamma \sin \mu + \frac{Y}{mV} \tan \gamma \cos \mu \cos \beta
$$

$$- \frac{dW_x}{dz} \sin^2 \gamma \cos \phi + \frac{L}{mV} \sin \mu \tan \beta
$$

$$+ \frac{Y}{mV} \sin \mu \cos \mu \sin \beta - \frac{dW_x}{dz} \sin \gamma \sin \mu \tan \beta \cos \phi; \quad (6.19)
$$

By adding Eq. 6.18 to Eq. 6.19, one can obtain:

$$
\cos \tan \beta \dot{\gamma} + (\sin \gamma + \cos \gamma \sin \mu \tan \beta) \dot{\phi} = \frac{L}{mV} (\tan \beta + \tan \gamma \sin \mu) + \frac{Y}{mV} \tan \gamma \cos \mu \cos \beta
$$

$$- \frac{g \cos \gamma}{V} \cos \tan \beta + \frac{dW_x}{dz} \cos \mu \tan \beta \sin^2 \gamma \sin \varphi
$$

$$- \frac{dW_x}{dz} \sin \gamma \sin \mu \tan \beta \cos \phi
$$

$$- \frac{Y}{mV} \sin \mu \cos \mu \sin \beta - \frac{dW_x}{dz} \sin \gamma \sin \mu \tan \beta \cos \phi \quad (6.20)
$$

Finally, from Eqs 6.14, 6.17, 6.20, and 6.1, the following equations provide the relationships between $(\dot{\gamma}, \dot{\phi})$ and $(\dot{\alpha}, \dot{\beta}, \dot{\mu})$:

$$\dot{\alpha} = q - \tan \beta (\cos \alpha + \sin \alpha) - \frac{\cos \mu}{\cos \beta} \dot{\gamma} - \frac{\cos \gamma \sin \mu}{\cos \beta} \dot{\phi}
$$

$$+ \frac{dW_x}{dz} \sin^2 \gamma \sin \varphi \frac{\cos \mu}{\cos \beta} - \frac{dW_x}{dz} \sin \gamma \sin \mu \cos \varphi
$$

$$\dot{\beta} = \sin \alpha \beta - \cos \alpha \beta - \sin \mu \dot{\gamma} - \cos \mu \dot{\phi} + \frac{dW_x}{dz} \sin^2 \gamma \sin \varphi \sin \mu
$$

$$+ \frac{dW_x}{dz} \sin \gamma \cos \mu \cos \varphi
$$

$$\dot{\mu} = \sec \beta (\cos \alpha + \sin \alpha) + \cos \mu \tan \beta \dot{\gamma} + (\sin \gamma + \cos \gamma \sin \mu \tan \beta) \dot{\phi}
$$

$$- \frac{dW_x}{dz} \cos \gamma \sin \gamma \sin \varphi + \frac{dW_x}{dz} \sin \gamma \sin \mu \tan \beta \cos \phi + \frac{dW_x}{dz} \sin^2 \gamma \cos \varphi \quad (6.21)
$$

The nonlinear mapping Eq. 6.6 can be obtained by rearranging Eq. 6.21:
Chapter 6. Trajectory tracking controller design

\[
\begin{bmatrix}
   p_c \\
   q_c \\
   r_c
\end{bmatrix} = \begin{bmatrix}
   \frac{\cos \alpha}{\cos \beta} & 0 & \frac{\sin \alpha}{\cos \beta} \\
   -\cos \alpha \tan \beta & 1 & -\sin \alpha \tan \beta \\
   \sin \alpha & 0 & -\cos \alpha
\end{bmatrix}^{-1}
\begin{bmatrix}
   \dot{\alpha}_c \\
   \dot{\beta}_c \\
   \dot{\mu}_c
\end{bmatrix} - \begin{bmatrix}
   M_1 \\
   M_2 \\
   M_3
\end{bmatrix}
\] (6.22)

**Remark.** The bandwidth frequency of the outer loop and inner loop are 50 Hz and 100 Hz respectively. They are determined by the step size of the flight states update in trajectory planning. This approach, as a nonlinear feedback controller, has inherent robustness with respect to uncertainty and disturbance [72].

6.2.2 Inner-loop dynamics

In the inner loop, the state-dependent Riccati equation (SDRE) approach [72] is utilized to design the saturation-respecting controller to follow the commands \([p_c, q_c, r_c]\) (Eq. 6.22).

The inner-loop (sometimes called fast-loop) dynamics are presented as:

\[
\begin{align*}
\dot{p} &= c_1 \bar{l} + c_2 \bar{n} + c_3 p q + c_4 q r \\
\dot{q} &= \frac{1}{I_{yy}} (\bar{m} + (I_{zz} - I_{xx}) pr + I_{xx}(r^2 - p^2)) \\
\dot{r} &= c_5 \bar{l} + c_6 \bar{n} - c_7 q r + c_8 p q
\end{align*}
\] (6.23)

The nonlinear dynamics (Eq. 6.23) can be rewritten as the affine-in-control formulation:

\[
\dot{X} = F(X) + G(X)U,
\] (6.24)

where \(X = [p, q, r]\), \(U = [\delta_e, \delta_a, \delta_r]\),

\[
F(X) = \begin{bmatrix}
   c_1qSb^2C_{l_{\alpha}} + c_2qSb^2C_{n_{\alpha}} & + c_3 p q + c_4 q r + c_1qSb^2C_{l_{\alpha}} + c_2qSb^2C_{n_{\alpha}} & \frac{2V}{2V} \\
   - \frac{I_{xx} p^2}{I_{yy} q^2} + \frac{qSb^2 C_{l_{\alpha}}}{2I_{yy} V} & + \frac{I_{xx} q^2}{I_{yy} p^2} + \frac{qSb^2 C_{n_{\alpha}}}{2I_{yy} V} & 0
\end{bmatrix}
\]

\[
G(X) = \begin{bmatrix}
   0 & c_1qSbC_{l_{\alpha}} & c_2qSbC_{n_{\alpha}} & c_1qSbC_{l_{\alpha}} + c_2qSbC_{n_{\alpha}} \\
   c_5qSbC_{l_{\alpha}} & 0 & 0 & c_5qSbC_{l_{\alpha}} + c_6qSbC_{n_{\alpha}}
\end{bmatrix}, \quad \ddot{q} = \frac{1}{2} \rho V^2.
\]

6.2.3 Linear Quadratic Regulator V.S. State Dependent Riccati Equation

**Linear Quadratic Regulator**

The LQR approach linearizes the system (Eq. 6.24) at a equilibrium point \(X_e\):

\[
\dot{X} = AX + BU,
\] (6.25)
where, $A = \frac{\partial F}{\partial X} |_{X=X_e}$, $B = \frac{\partial G}{\partial X} |_{X=X_e}$. By solving the algebraic Riccati equation:

$$PA + A^T P - PB^T P + Q = 0.$$  \hfill (6.26)

We can obtain the optimal feedback controller: $U = -R^{-1}B^{-1}PX$.

Suppose that the equilibrium point $X_e$ is chosen at the origin: $X_e = [p, q, r] = 0$. According to steady gliding analysis in Chapter 4 (Fig. 4.6, Fig 4.7, Fig. 4.9, and Fig. 4.8), the out-loop equilibrium states (airspeed $V$, angle of attack $\alpha$, and side-slip angle $\beta$) are chosen as:

$$V = 19.95 \text{ m/s}, \alpha = 0.13 \text{ rad}, \beta = 0 \text{ rad}$$  \hfill (6.27)

By calculating $A = \frac{\partial F}{\partial X} |_{X=X_e}$, $B = \frac{\partial G}{\partial X} |_{X=X_e}$ and solving the algebraic Riccati equation (Eq. 6.26), the optimal feedback controller is:

$$U_{\text{lqr}} = \begin{bmatrix} -0.0032 & 0.9534 & 0.0952 \\ -0.6682 & 0.0459 & 0.2760 \\ -0.4997 & 0.0643 & -0.5689 \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix}$$  \hfill (6.28)

The state dependent Riccati equation (SDRE) approach

The SDRE approach utilizes state-dependent coefficients [87] (SDC) parameterization to transform the affine-in-control system:

$$\dot{X} = F(X) + G(X)U$$

into a linear-like formulation:

$$\dot{X} = A(X)X + B(X)U$$

In order to ensure the existence of the linear-like formulation [72], the nonlinearity term $F(X)$ should be equal to zero at the origin, namely, $F(0) = 0$. However, according to Eq. 6.24, the nonlinearity term $F(X)$ includes $C \bar{m}_0$ term, where $C \bar{m}_0(0) \neq 0$.

In order to deal with the $C \bar{m}_0$ term, a stable $\dot{z} = -\lambda z \ (\lambda = 1)$ is augmented to the Eq. 6.24 to handle state-independent terms such that the augmented system $\bar{X} = [p, q, r, z]$ meets the requirement $\bar{F}(0) = 0$. As a result, the linear-like formulation can be expressed as:

$$\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \\ \dot{z} \end{bmatrix} = \bar{A}(p, q, r, z) \begin{bmatrix} p \\ q \\ r \\ z \end{bmatrix} + B(p, q, r, z) \begin{bmatrix} \delta_e \\ \delta_a \\ \delta_r \end{bmatrix}$$  \hfill (6.29)
where, \( \bar{A}(p, q, r, z) = \begin{bmatrix} c_1 \bar{q} S_b^2 C_{l_p} + c_2 \bar{q} S_b^2 C_{n_p} & \frac{c_3 p + c_4 q}{2V} & \frac{c_5 \bar{q} S_b^2 C_{l_r} + c_2 \bar{q} S_b^2 C_{n_r}}{2V} & 0 \\ \frac{-1}{2V} p & \frac{\bar{q} S e^2 C_{m_a}}{2 V} & \frac{I_{x y} - I_{y z} p + I_{y z} r}{2 V} & \frac{\bar{q} S e C_{m_a} + \bar{q} S e C_{m_a} \alpha}{I_{y y}} \\ 0 & c_7 p + c_8 r & \frac{c_5 \bar{q} S_b^2 C_{l_a} + c_2 \bar{q} S_b^2 C_{n_a}}{2V} & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix} \)

\( \bar{B}(p, q, r, z) = \begin{bmatrix} 0 & c_1 \bar{q} S_b C_{l_a} + c_2 \bar{q} S_b C_{n_a} & c_1 \bar{q} S_b C_{l_r} + c_2 \bar{q} S_b C_{n_r} \\ \frac{\bar{q} S e}{I_{y y}} C_{m_a} & 0 & 0 \\ 0 & c_5 \bar{q} S_b C_{l_a} + c_6 \bar{q} S_b C_{n_a} & c_5 \bar{q} S_b C_{l_r} + c_6 \bar{q} S_b C_{n_r} \\ 0 & 0 & 0 \end{bmatrix}, \bar{q} = \frac{1}{2} \rho V^2. \)

By mimicking the Linear Quadratic Regulator (LQR) method to solve the following state dependent Riccati equation [72] at each step \( \bar{X} = [p, q, r, z] \):

\[
\bar{P}(\bar{X}) \bar{A}(\bar{X}) + \bar{A}(\bar{X})^T \bar{P}(\bar{X}) - \bar{P}(\bar{X}) \bar{B}(\bar{X}) R^{-1} \bar{B}^T(\bar{X}) \bar{P}(\bar{X}) + Q = 0. \tag{6.30}
\]

the state feedback controller \( U_{sdre} \) is then obtained as:

\[
U_{sdre} = -R^{-1}(\bar{X}) \bar{B}^{-1}(\bar{X}) \bar{P}(\bar{X}) \bar{X} \tag{6.31}
\]

### 6.2.4 Comparison study

According to the multiple-time-scale technique, the outer-loop states \( (V, \alpha, \beta) \) in the inner-loop dynamics (Eq. 6.23) remains constant when regulating the inner-loop states \( (p, q, r) \). In each out-loop states case, the initial inner-loop states \( p, q, r \) are chosen randomly from the uniform distribution \( U(0, 1.5) \).

\[
p = U(0, 1.5) \\
q = U(0, 1.5) \\
r = U(0, 1.5) \tag{6.32}
\]

For each out-loop states case, the LQR controller (Eq. 6.28) and SDRE controller (Eq. 6.31) are utilized to regulate 100 random initial inner-loop states \( p, q, r \) to follow the commands \( p_c = 0, q_c = 0, r_c = 0 \). Figure 6.2 and Fig 6.3 provides the tracking performance and control inputs of the LQR and SDRE controller.

According to the results, the performance of SDRE is better than the LQR method. This phenomenon can be explained as follows. First of all, compared to Jacobian linearization in LQR, which approximates the nonlinear system around a specific point, the SDC parameterization in SDRE avoids the model mismatch between the linearized system and the nonlinear system by keeping the nonlinearity in the linear-like formulation. Secondly, the LQR controller (Eq. 6.28) based on Jacobian linearizations is valid when the system is operating in the neighborhood of the specific linearization point. Therefore, the LQR method experiences a decrease in performance when the system operates away from the linearization point.
V = 19.95, α = 0.13 rad, p = 0, q = 0, r = 0). To address this issue, a family of controllers is required to cover the different operating points of the system. In control theory, the method is called gain-scheduling. It is usually tedious to choose the appropriate one from the group of controller corresponding to various system operating points. In contrast, the SDRE provides a more efficient way to provide the optimal controller by solving the state dependent Riccati equation (Eq. 6.30) at each step (“operating point”). Finally, the SDRE method inherits the characteristics of the LQR method which can provides the best trade-off between performance and achievable control inputs. Therefore, the SDRE method is applied to design the inner-loop system of the trajectory tracking controller.
6.3 Simulation results

6.3.1 Static soaring trajectory tracking results

In this section, a part of the static soaring trajectory is used here as an example to demonstrate the effectiveness of the proposed trajectory tracking controller. The state weighing matrix $Q$ and the control weighting matrix $R$ determine how much states and inputs contribute to the performance cost:

$$J(x_0, u(\cdot)) = \frac{1}{2} \int_0^\infty \{x^T(t)Q(x(t))x(t) + u^T(t)R(x(t))u(t)\} dt \quad (6.33)$$
For those states and inputs that should remain small, the corresponding weight value should be large. In current work, $Q$ and $R$ are chosen based on a trial-and-error procedure. In static soaring trajectory tracking problem, $Q$ and $R$ in the SDRE controller are chosen as follows:

$$Q = \begin{bmatrix} 100 & 0 & 0 & 0 \\ 0 & 100 & 0 & 0 \\ 0 & 0 & 100 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \quad R = \begin{bmatrix} 4 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \end{bmatrix}.$$

Figure 6.4 shows the static soaring trajectory tracking results using the SDRE-based and LQR tracking controller. Following the desired static soaring trajectory, the UAV is capable of gaining potential energy by loitering around updrafts. Both SDRE and LQR provide good tracking performance, as static soaring does not involve much flight maneuvers.

Figure 6.5 and 6.6 provides state tracking results using the LQR (Eq. 6.28) and the SDRE controller. In these two figures, the x axes represents simulation steps in seconds, and y axes shows airspeed $V$, flight path angle $\gamma$, heading angle $\varphi$, angle of attack $\alpha$, side-slip angle $\beta$, and bank angle $\mu$ respectively. According to the results, the SDRE performs a bit better than LQR.

Figure 6.7 shows the control inputs along the soaring trajectory. According to the simulation results, the UAV is capable of following the desired soaring surveillance trajectory given the actuator constraints.

### 6.3.2 Dynamic soaring trajectory tracking results

Once the dynamic soaring surveillance trajectory is determined, the proposed tracking controller is applied. The state weighing matrix $Q$ and the control weighting matrix $R$ in the SDRE controller are chosen as follows:

$$Q = \begin{bmatrix} 100 & 0 & 0 & 0 \\ 0 & 100 & 0 & 0 \\ 0 & 0 & 100 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \quad R = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

Figure 6.8 shows the promising trajectory tracking results using the SDRE controller. In Fig. 6.9, and Fig. 6.10, the LQR (Eq. 6.28) and SDRE controller are applied to track the desired trajectory states including airspeed $V$, flight path angle $\gamma$, heading angle $\varphi$, angle of attack $\alpha$, side-slip angle $\beta$ and bank angle $\mu$. Results demonstrate the SDRE method performs better than the LQR controller.

Figure 6.11 shows the control inputs along the soaring trajectory. According to simulation results, the UAV is capable of following the desired dynamic soaring surveillance trajectory given the actuator
6.4 Summary

In this section, a trajectory control approach was presented to address the soaring trajectory tracking problem. The trajectory tracking controller was designed using a full six-degree-of-freedom nonlinear rigid-body dynamics model, which was simplified by the multiple-loop control technique. The state dependent Riccati equation (SDRE) method was applied to design the control law. Promising simulation results were presented to demonstrate the performance of the proposed approach.
Figure 6.4: A part of static soaring trajectory tracking performance
Chapter 6. Trajectory tracking controller design

Figure 6.5: Airspeed, flight path angle, and heading angle tracking performance static soaring
Figure 6.6: Angle of attack, side-slip angle, and bank angle tracking performance static soaring
Figure 6.7: Control inputs along the static soaring trajectory
Figure 6.8: A part of dynamic soaring trajectory tracking performance
Figure 6.9: Airspeed, flight path angle, and heading angle tracking performance in dynamic soaring
Figure 6.10: Angle of attack, side-slip angle, and bank angle tracking performance in dynamic soaring
Figure 6.11: Control inputs along the dynamic soaring trajectory
Chapter 7

Conclusions and future work

7.1 Conclusions

Soaring is one flight strategy of exploiting favorable air currents to gain energy from the atmosphere. This thesis presented autonomous soaring surveillance by incorporating the soaring strategy into the problem of aerial surveillance using an unmanned aerial vehicle (UAV).

Gliding and soaring flight were studied respectively in Chapter 2. By investigating the UAV’s kinematics in the presence of winds, we have explained where and how to harvest extra energy from winds. With respect to different wind patterns, soaring can be divided into two types: static and dynamic soaring. In static soaring, the UAV loiters around the spot of the vertical air motion to gain potential energy. In dynamic soaring, the UAV exploits wind speed variations (wind gradients) to harvest extra kinetic energy. Because of different wind energy sources, the soaring surveillance strategy includes static soaring and dynamic soaring surveillance accordingly.

The soaring surveillance problem was formulated in Chapter 3 by defining the surveillance area and task. The small UAV was assumed to have insufficient initial energy to visit all surveillance targets. Therefore, the small UAV should discover available energy sources to refill itself to carry on the surveillance mission. Soaring surveillance can be further elaborated as follows. When the UAV flies beyond the surface layer of the PBL, static soaring surveillance was performed by searching and exploiting updrafts to harvest energy. When the UAV reached the surface layer where the dominant wind is the wind gradient, dynamic soaring surveillance was employed to take the advantage of spatial wind gradients. With aid of soaring, every predefined surveillance point was expected to be visited by the small UAV. In this situation, long-endurance surveillance flight can be achieved.

The static soaring surveillance approach was proposed in Chapter 4 to achieve long-endurance surveillance flight in the wind field with spatially-distributed updrafts. The current work focused on the case of the quasi-static wind field where the updraft was stationary or oscillating around the hot origin.
The surveillance area was partitioned into a series of concentric expanding regions (Regions 2, 3,...,N...) to alleviate the computational load of the surveillance path planning and updrafts identification. The UAV followed the optimal path to visit each surveillance point and measure the vertical wind speed along the way. The wind field (updraft’s distribution) in each expanding region was estimated by Gaussian Process Regression with boundary constraints. The UAV loitered around the strongest identified updraft to gain energy in order to achieve long-endurance surveillance. The large eddy simulation (LES) model of the weather research and forecasting system (WRF-ARW) was utilized to generate the high-fidelity simulated wind field, which was used as a test case to demonstrate the effectiveness of the proposed strategy.

In Chapter 5, the dynamic soaring surveillance approach utilized the wind gradient which stretched out in the surface layer of the atmosphere to harvest extra energy to achieve long endurance surveillance. According to the dynamic soaring rule, the UAV exploited energy by repeatedly ascending upwind or descending downwind in the wind-gradient region. One cycle of dynamic soaring, which started and ended with the same heading angle, can connect two arbitrary surveillance points with the aid of the Dubins’ path. To be more specific, the Dubins path determined the 2D trajectory on the reference plane. The height profile of the trajectory can be described by the B-spline curve, by strategically choosing which the energy-efficient trajectory can be obtained. Along the trajectory, the UAV was capable of soaring from one point to another to perform the surveillance task.

In order to ensure good tracking performance, the trajectory tracking control approach was presented in Chapter 6. The trajectory tracking controller was designed using a full six-degree-of-freedom nonlinear rigid-body dynamics model, which was simplified by the multiple-time-scale technique. The state dependent Riccati equation (SDRE) method was elaborated and applied to design the inner-loop control law. Promising simulation results were presented to demonstrate the performance of the proposed approach.

### 7.2 Future work

The thesis proposes the autonomous soaring surveillance framework using a small unmanned aerial vehicle. With the aid of autonomous soaring, long endurance surveillance flight can be achieved with the limited on-board power capacity. In future work, the following steps need to be studied.

First, it is worthwhile to seek to determine whether a flock of UAVs can improve soaring surveillance flight performance. In this thesis, one UAV performed the surveillance task and detected the soaring location to compensate for on-board energy loss. A single UAV can only provide wind speed information along its trajectory, in contrast with multiple UAVs, which can provide wind speed data for multiple locations at the same time. Wind speed information for multiple locations can be used to estimate the updraft’s location spatially as well as temporally. Furthermore, multiple UAVs increase the chance of
encountering the updraft floating in the atmosphere. Consequently, in the presence of multiple UAVs, the soaring surveillance strategy can be applied in the wind field, which includes stochastic wind motions.

Secondly, in this thesis, the trajectory and the tracking controller were designed separately. The trajectory planning neglected to take into account the dynamic characteristics of the aircraft. Designing the trajectory and the tracking controller in a single loop would be a superior approach. An integrated trajectory planning and tracking control approach may improve soaring flight performance. In addition, this approach minimizes the tracking controller’s instability, a factor that may arise when employing the traditional separate design approach.

The final goal of the future research is to perform flight tests. The proposed method has been evaluated using a numerical simulated wind field. Nonetheless, it is important to validate the effectiveness and capability of the proposed method in a real wind field as well.
Bibliography


[77] C. Gao and H. H. T. Liu. Nonlinear multi-objective receding horizon design utilizing the state-


http://dx.doi.org/10.1175/MWR3406.1, 2007.


[83] K. Andersson, I. Kaminer, and V. Dobrokhodov. Thermal centering control for autonomous soaring:
975, May - June 2012.


University of Minnesota, 1991.

10.1109/ACC.2002.1024785.