Uncertainty Management for Aerial Vehicles: Coordination, Deconfliction, and Disturbance Rejection

Prachya Panyakeow

A dissertation

submitted in partial fulfillment of the

requirements for the degree of

Doctor of Philosophy

University of Washington

2015

Reading Committee:

Prof. Mehran Mesbahi, Chair

Prof. Kristi Morgansen

Prof. Anshu Narang-Siddarth

Program Authorized to Offer Degree:

Aeronautics and Astronautics

©Copyright 2015

Prachya Panyakeow

University of Washington

Abstract

Uncertainty Management for Aerial Vehicles:
Coordination, Deconfliction, and Disturbance Rejection

Prachya Panyakeow

Chair of the Supervisory Committee:

Professor Mehran Mesbahi

Aeronautics and Astronautics

The presented dissertation aims to develop control algorithms that deal with three types of uncertainties managements. First, we examine the situation when unmanned aerial vehicles (UAVs) fly through uncertain environments that contain both stationary and moving obstacles. Moreover, a guarantee of collision avoidance is necessary when UAVs operate in close proximity of each other. Second, we look at the communication uncertainty among the network of cooperative UAVs and the efforts to establish and maintain the connectivity throughout their entire missions. Third, we explore the scenario when the aircraft flies through wind gust. The introduction of an appropriate control scheme to actively alleviate the gust loads can result into weight reduction and consequently lower the fuel cost.

In the first part of this dissertation, we develop a deconfliction algorithm that guarantees collision avoidance between a pair of constant speed unicycle-type UAVs as well as convergence to the desired destination for each UAV in presence of static obstacles. We use a combination of navigation and swirling functions to direct the unicycle vehicles along the planned trajectories while avoiding inter-vehicle collisions. The main feature of our contribution is proposing means of designing a deconfliction algorithm for unicycle vehicles that more closely capture the dynamics of constant speed UAVs as opposed to double integrator models. Specifically, we consider the issue of UAV turn-rate constraints and proceed to explore the selection of key algorithmic parameters in order to minimize undesirable trajectories and overshoots induced by the avoidance algorithm. The avoidance and convergence analysis of the proposed algorithm is then performed for two cooperative UAVs and simulation results are provided to support the viability of the proposed framework for more general mission scenarios.

For the uncertainty of the UAV network, we provides two approaches to establish connectivity among a collection of UAVs that are initially scattered in space. The goal is to find shortest trajectories that bring the UAVs to a connected formation where they are in the range of detection of

one another and headed in the same direction to maintain the connectivity. Pontryagin Minimum Principle (PMP) is utilized to determine the control law and path synthesis for the UAVs under the turn-rate constraints. We introduce an algorithm to search for the optimal solution when the final network topology is specified; followed by a nonlinear programming method in which the final configuration is emerged from the optimization routine under the constraints that the final topology is connected. Each method has its own advantages based on the size of corporative networks.

For the uncertainty due to gust turbulence, we choose a model predictive control (MPC) technique to address gust load alleviation (GLA) for a flexible aircraft. MPC is a discrete method based on repeated online optimization that allows direct consideration of control actuator constraints into the feedback computation. Gust alleviation systems are dependent on how the structural flexibility of the aircraft affects its dynamics. Hence, we develop a six-degree-of-freedom flexible aircraft model that can integrate rigid body dynamic with structural deflection. The structural stick-and-beam model is utilized for the calculation of aeroelastic mode shapes and airframe loads. Another important feature of MPC for GLA design is the ability to include the preview of gust information ahead of the aircraft nose into the prediction process. This helps raising the prediction accuracy and consequently improves the load alleviation performance. Finally, the aircraft is modified by the addition of the flap-array, a composition of small trailing edge flaps throughout the entire span of the wings. These flaps are used in conjunction with the distributed spoilers. With the availability of the control surfaces closer to the wing root, the MPC with flap-array can reduce the wing bending moment from different mode shapes and achieve better load alleviation performance than the original aircraft.

Contents

List of Figures

Acknowledgements

The University of Washington has been the biggest part of my life. As I am leaving, I am privileged to acknowledge the personal and academic relationships that have made my PhD degree a reality. These life-long connections came from priviledged opportunities to work with these hard-working, creative, and talented people.

I would like to thank my academic advisor Professor Mehran Mesbahi for his continued support. It is through his enlighten guidance, his willingness to engage in wonderful discussions, his steady encouragement, and most important of all, his believing in my abilities. Professor Mehran Mesbahi, thank you!

The RAIN Lab has given me a supportive working environment for my studies. The diversity of people in the group always made me feel welcomed. My conversations with Dr. Airlie Chapman and Eric Schoof have consistently be fun and helped me understand Australian cultures. Thanks to Dr. Unsik Lee for inspiring me with research ideas and letting me share with him many stories and passion for flying. I always enjoyed our monthly Pho lunches on University avenue, Unsik! I must thank Saghar Hosseini and Haibin Shao for being a great desk neighbors and listening to my complains about life. And of course, thank you to Professor Ran Dai from Iowa State University for great research discussions as well as her honesty and friendship.

In addition to the the RAIN Lab, I would like to thank Jake Quenzer from Nonlinear Dynamics and Control Laboratory for fun discussions on development of GTA model. It's no doubt that due to our collegiate chemistry, and his enthusiasm for this project, I will be collaborating after I leave the university for Boeing.

Lastly, I would like to thank my family for their support both financially and academically. I would not have been able to complete my PhD without the support that I have received from each one of you!

Chapter 1

Introduction

Research in the area of multi-vehicle systems has been a topic of great interest in the recent years due to broad applications in many areas including cooperative control of unmanned aerial vehicles (UAVs), formation control [44], [45], and flocking [52]. The paradigm of having one UAV performing all mission tasks raises a number of issues related to the cost of development, integration, and mission completion. This has motivated the concept of multi-vehicle systems where each vehicle only performs a subset of the entire mission. In such a setup, the communication between the vehicles defined by the topology of the network has a significant effect on the performance of the multi-vehicle system. When the relative distances between vehicles are considered in the construction of a geometry-based network (via wireless network technology), an information-exchange link is assumed to exist when two vehicles are within their maximum communication range. Collision avoidance and deconfliction becomes of paramount importance when such missions require that UAVs operate in close proximity of each other.[1] It has thus become necessary to develop efficient algorithms to ensure that vehicles stay conflict-free under environmental uncertainties such as terrain or presences of other UAVs while completing their intended missions. On the other hand, since UAVs are required to disperse to perform tasks such as monitoring [39], imaging, reconnaissance, data processing, etc., the relative distances between vehicles may exceed their maximum communication range creating uncertain network connectivity maintenance. In order to reestablish network connectivity for data exchange and to resume mission operations as a group, the problem of controlling vehicles that are initially out of range of detection

[1] Deconfliction refers to collision avoidance with the additional requirement that vehicles stay as close as possible to their original intended path or trajectories.

to an area where they can sense each other becomes important. Another form of uncertainty that affects the performance of the aerial vehicle's missions is the weather. Wind gusts cause additional fuel expenditure, metal fatigue, structural deformation, as well as reduction in flight comfort. By using controlled deflections of tail and wing control surfaces, it is possible to minimize the amplitude and the number of transient bending cycles to which the structure may be subjected in flight.

This research topic deals with these three types of flight uncertainty management. First we provide an approach for the deconfliction problem when a UAV flies through uncertain environments that contain both stationary and moving obstacles. The avoidance and convergence analysis is performed under turn-rates constraints and to minimize deviations from desirable trajectories. The second problem involves the communication uncertainty among the proximity network of cooperative UAVs and the efforts to establish and maintain connectivity throughout entire missions. Optimization and nonlinear programming methods are used to determine the shortest trajectories that bring the UAVs to a connected formation where they are in the range of detection of one another and oriented in the same direction to maintain the connectivity. The third area of research involves uncertainty due to wind gusts. Airframe weight reduction is a typical design goal for modern aircraft in order to lower fuel consumption and consequently to increase payload and range and reduce operating cost. In general, aircraft structures are required to withstand critical loading conditions during maneuvers or when flying through atmospheric turbulence. Various loading scenarios such as discrete gust and continuous turbulence are defined according to the airworthiness regulations CS-25 [84]. These design condition often contribute to the load envelop. For instance, the strength of the wing roots structure can thereby translate to the weight of the wing. The introduction of an appropriate control scheme to actively alleviate the gust loads can result into weight saving without compromising safety.

1.1 UAVs Deconfliction

There have been a number of research works on conflict and collision avoidance between autonomous vehicles; see [1] for a survey. In [2–4] the authors propose a prescribed set of maneuvers and protocols for each vehicle when a conflict has been detected. The work of [5] involves a heuristic search for a tree of feasible avoidance maneuver using velocity obstacles. Optimization-based schemes on the other hand, as exemplified in [6, 7], revolve around solving optimization problems, often in centralized way, to resolve conflicts, either deterministically or in a probabilistic setting. In [6], [8–11], the notion

of collision cone has been introduced for each vehicle, parameterizing the zone where the collision will occur if all other vehicles keep their heading unchanged. The geometric optimization approach in [12] also utilizes the collision cone by minimizing the velocity vector changes such that the relative velocity between two conflicting vehicles stay off the cone. The work of [38] develops the reactive collision avoidance algorithm based from Lyapunov stability theory and relative geometry between the aircraft and static obstacles. In [36] and [37], the authors present a vision-based collision avoidance for miniature air vehicles (MAVs) using navigation frame mapping and path planning.

Closer to our work are approaches that are developed with the aid of a potential or a navigation function. A navigation function landscapes the configuration space of the vehicles such that the goal points for the vehicle group can be reached by following the function's descent direction. In [13] and [14], such functions have been adopted for holonomic robots. Rimon and Koditschek [15] on the other hand, introduced a navigation function for robot obstacle avoidance in a generalized "star world" with a static environment. A convergence analysis for the trajectories generated by the navigation function has also been explored in [16].

Recently, the use of *decentralized navigation functions* has gained attention in various research works in the area of cooperative and formation control; see for example, [17,18]. Being "decentralized" implies that each vehicle does not require knowledge of the states and desired destinations of all vehicles and merely relies on limited information, such as relative position and velocity from the "neighboring" vehicles, to adjust its steering strategy. The decentralization is particularly suited for large number of vehicles as it reduces the computational aspects of gradient computation as well as the associated communication and sensing requirements. In this direction, we mention the works of Dimarogonas and Kyriakopoulos [19,20] who have utilized the notion of decentralized navigation functions for addressing collision avoidance between multiple *holonomic agents* in the absence of control constraints. Collision avoidance for nonholonomic vehicles has been considered by Aicardi *et al.* [30,31], utilizing the velocity vector field and tracking of 2D path to avoid obstacles, while in [33], the author adopted a decentralized optimization approach that requires large computational resources. In [29], a kinematic controller and dipolar navigation function has been proposed to avoid collisions and configure nonholonomic agents to the desired orientation at their respective destinations.

Our approach to the unicycle-model UAV deconfliction [26, 27] is presented in chapter 2. The algorithm builds upon the following desirable properties for the UAV trajectories:

1. guaranteed collision-free convergence to the final desired destination for each UAV in the presence of static obstacles,

2. ensured proximity to the nominal path for each UAV when permissible- thus making our approach suited for *deconfliction* rather than pure collision avoidance,

3. guaranteed collision avoidance with other UAVs within the range of detection, and finally,

4. guaranteed operation within the maximum turn-rate for each unicycle UAV during the entire mission.

The first two requirements necessitate the introduction of a navigation function– combining path planning and obstacle avoidance in order to create the deconfliction control laws. However, the fourth requirement forces us to employ *unicycle models* for the UAVs with turn-rate constraints in order to more accurately capture the non-holonomic dynamics of UAVs with constant speed. In the meantime, the third requirement invites us to consider an extension of the work in [21], where a swirling effect is introduced for double integrator type models in order to break the symmetry and avoid saddle points around the moving obstacles. The works of [47] and [48] also provide similar effects on steered particles to converge to a circular motion of all particles around the same circle. The swirling effect has to be done in conjunction with examining the vehicles' turn-rates, trajectory overshoots, and minimum distances between vehicles during the deconfliction maneuver as consistent with other requirements.

The contribution of our work in UAV deconfliction is threefold. First, we show how constraints such as UAV turn rate limits and effective range of detection can be taken into consideration in designing navigation-based deconfliction algorithms. Moreover, we theoretically demonstrate how the tuning parameters for the navigation function can be selected such that vehicles' actuator commands do not exceed prescribed limits without sacrificing deconfliction performance. And lastly, and more importantly, the collision avoidance and the global convergence of the algorithm are analyzed for two conflicting unicycle type UAVs and verified for more general scenario through simulations.

1.2 Optimal Trajectories for Network Establishment for UAVs

Recent works on network connectivity have been focused on maintaining network connectivity during the entire mission [53, 54, 56]. In [42], the author considers the integration of communication and control with respect to the task of coordinated heading control for a group of nonholonomic vehicles.

The heading control uses the Kuramoto model of nonlinearly coupled oscillators. In [55], the authors derived an optimal control scheme to establish the connectivity among a group of spacecraft. For our setup, we consider a group of n nonholonomic vehicles that are initially out of the communication range, and propose methods to establish connectivity with optimal trajectories, subject to a turn-rate constraint. Similar to our deconfliction work, the unicycle model is chosen to capture the dynamics of UAVs that fly at constant cruising speed as opposed to linear models. Since the fuel usage is a prominent concern, in the case of constant speed vehicles, the optimality refers to minimizing maneuver time or path-length. The final configuration is in the form of relative positions and headings between connected vehicles with a graph representation of the connectivity in the form of a tree network. The solution is derived using Pontryagin Minimum Principle (PMP). However, the question of path synthesis encourages us to look into the Dubin's problem [40], [41] that involves finding the shortest smooth path between two given points by a car, for which the starting and ending directions are specified. With the final orientation of the vehicles constrained to head in the same direction, the optimal solution for each vehicle comprises of three segments: two circular arcs of maximum curvature and a straight line tangent to the arcs. The task of determining how to combine these pieces is derived by solving a set of necessary conditions.

In Chapter 3, we develop two approaches to establish the connectivity among UAVs. First we propose the algorithm to search for the global optimal solution when a target "tree" graph connectivity structure has been specified. However, searching for the best solution across all possible trees requires long computational time when applying to more than a few UAVs. The alternative approach is to transform the nonlinear optimal control into a parameter optimization problem without specifying the target graph, relaxing the on/off connectivity by using an exponential penalty function [43] to approximate the communication strength, and imposing the constraint that the final topology is connected. The control scheme remains the same while the time on each path segment is expressed as an unknown parameter. The connectivity constraint is then satisfied using the Cholesky decomposition. The solution can subsequently be solved using nonlinear programming solvers [49].

1.3 Gust Load Alleviation for Flexible Aircraft

Gust load alleviation (GLA) systems use motion sensor feedback to drive an aircraft's control surfaces in order to attenuate aerodynamics loading induced by wind gusts. The use of active control has

(a) (b) (c)

Figure 1.1: (a) An efficient deconfliction algorithm is necessary when UAVs operate in close proximity of each other. (b) UAVs network establishment method is required when their relative distance fall out of the detection range. (c) An appropriate gust load alleviation system can reduce aircraft weight and fuel cost.

been examined extensively, for example, the work in maneuver load alleviation [72], [73], [74] and the suppression of aeroelastic effects [75], [77]. In [75], the author applied a static output feedback controller to a simple wing section to perform gust load alleviation (GLA) and flutter suppression. Some early works on gust load alleviation include [65] and [69] where the authors used LQR and LQG approaches without performance weights for flexible aircraft models. In [70] and [71], optimal and robust control techniques were employed for GLA problems, but the rigid-body dynamics of the aircraft were used. The researches in aeroservoelasticity were based on flexible aircraft models consisting of rigid body dynamics and a series of second-order differential equation representing the structural mode [65], [66], [67]. McLean and Prasad [66] used a linear quadratic optimal controller to reduce the wing bending moment of C-5A Galaxy aircraft. The simulation showed that the controller led to less oscillatory motion in the wing bending moment. In [67] and [68], the authors utilized $\mathcal{H}_2$, $\mathcal{H}_\infty$ and μ-controller on a modified variant of the B-52 aircraft where the horizontal canards were included. The controller can reduce the transient peak of structural load from the stochastic gust excitation. The application of adaptive feed-forward controller for suppression of vibration was also studied in [86] and [87]. In [87], the author applied a steepest descent algorithm in order to increase the robustness of the performance of the feed-forward control system against modeling errors and variations of wing. Moulin and Karpel [78] investigate the effectiveness of different control surfaces of a large transport aircraft for GLA problems. The underwing and wing-tip control surfaces were introduced and shown to be more effective at relaxing the wing structural load than a conventional aileron-spoiler control under a discrete gust.

In Chapter 4, a nonlinear six-degree-of-freedom equation of motion for the generic transport aircraft (GTA) is developed[2]. The aerodynamic model is derived using the method similar to [82] while the structural dynamics utilize a stick-and-beam finite element model provided by the vehicle description in the ZAERO user's manual [80]. The mode shape is extracted using MSC/NASTRAN. The developed equation can effectively integrate large rigid body motion with structural deflections. As a suitable control scheme to address GLA for flexible aircraft, we propose model predictive control (MPC), or receding horizon control. MPC is a discrete online optimization-based control strategy that can optimize the load over a finite prediction horizon. It is a structured approach which allows us to work with either linear or nonlinear dynamics, includes the load directly in the cost function, considers constraints on state, inputs, and output, as well as incorporates the disturbance into the dynamics. The stability of the method can be guaranteed by picking the right terminal weighting function that satisfies the cost monotonicity condition. The GLA works that utilize MPC approach include [92], [93], and [94]. In [94], the authors introduced an additional feedback loop to increase the prediction accuracy.

The contribution of our work in gust load alleviation is twofold. On one hand, the performance of GLA can be improved by sensing capability. Predictive controllers can be adversely affected by inaccurate prediction. Hence we use the MPC that takes measured or estimated gust information into account to improve the prediction accuracy. The incoming gust may be measured via light detection and ranging (LIDAR), a method that was applied in [86] and [92]. On the other hand, the load alleviation can be enhanced by actuation capability. We did this by adding a "*flap-array*", a composition of small trailing edge flaps throughout the entire span of the wing. The idea of active microflaps has also been explored in [76] where the small flaps on wind turbine blades are used for dynamic load control. In our work, these flaps are distributed evenly along the wingspan and may be used in conjunction with the distributed spoilers. This opens the opportunity to alter the wing lift and drag distribution more delicately and thus reduce the contribution of wing bending moment from different mode shapes more efficiently.

1.4 Contribution to Knowledge

This thesis presents the following contributions to knowledge:

[2]The work in collaboration with Jake Quenzer [91].

1. A deconfliction algorithm for unicycle UAVs that can guarantee collision avoidance with both static and moving obstacles, convergence to the desired trajectory and destination, and operation within the maximum turn-rate.

2. A complete analysis for the selection of design parameters such that the vehicles' actuator commands do not exceed prescribed limits without sacrificing deconfliction performance.

3. Two approaches to optimally establish the connectivity of unicycle UAVs networks and analysis on scalability and computational issue for both methods.

4. A complete model for a generic transport aircraft that integrates rigid body motion with structural dynamics and includes load calculation suitable for gust load alleviation control design.

5. A model predictive control framework for gust load alleviation and performance enhancement efforts via improved sensing capabilities and the introduction of a flap-array.

1.5 Summary

In summary, this thesis describes the development of control platforms that deal with three types of environmental flight uncertainties. Chapter 2 presents a deconfliction algorithm for unicycle UAVs. This focuses on the construction of the algorithm and the proof of guaranteed collision avoidance under the turn-rate constraints. Chapter 3 presents the two approaches for optimal path planning of UAVs network establishment. The discussion on the performance of both methods are provided. Chapter 4 derives the mathematical model of a flexible generic transport aircraft and the gust load alleviation design. This includes the introductions of flap-array and improved sensing capability via LIDAR. Chapter 5 contains a summery of findings, concluding remarks, and recommendations for future work.

Chapter 2

Deconfliction Algorithms for UAVs

This chapter covers the first area of researches on collision avoidance between UAVs and both stationary and moving objects. The deconfliction algorithm that guarantee collision avoidance and tracking performance under aircraft turn-rate constraints is developed. The main feature of our contribution is proposing means of designing a algorithm for unicycle vehicles that more closely capture the dynamics of constant speed UAVs as opposed to double integrator models. Specifically, we use the framework of navigation and swirling functions to determine the desired heading and flight path angles for each UAV when a "conflict" is detected. Since the navigation function reflects the desire to follow the nominal trajectory while avoiding constraints, we refer to the resulting algorithm as *deconfliction* as opposed to collision avoidance, which is reserved for avoiding collisions *without* being concerned with the nominal trajectory. We consider the issue of UAV turn-rate constraints and proceed to explore the selection of key algorithmic parameters in order to minimize undesirable trajectories and overshoots induced by the avoidance algorithm. The avoidance and convergence analysis of the proposed algorithm is then performed for two cooperative UAVs and simulation results are provided to support the viability of the proposed framework for more general mission scenarios.

The organization of this chapter is as follow. We begin by providing the model setup for a pair of identical unicycle UAVs and their representative mission scenarios in Section 2.1. In Section 2.2, the decentralized navigation function is presented followed by the corresponding deconfliction control laws, as well as the way-point steering method. In this section, we also examine the convergence of the UAVs to their respective destinations along with the guaranteed collision avoidance in presence

of static obstacles. Section 2.3 introduces the swirling function and presents the proof of guaranteed avoidance for a pair of unicycle UAVs using LaSalle's Invariance Principle. Section 2.4 provides the analysis of the tuning parameters that effect UAV turn-rates during the deconfliction maneuver and proposes a method for improving the algorithm's performance. Section 2.5 illustrates simulations scenarios, with particular attention to UAV turn-rates and trajectory overshoots as guided by the proposed algorithm. Concluding remarks are provided in Section 2.6.

2.1 UAVs Path Planning Model

Consider a group of n autonomous UAVs flying in three-dimensional space $\mathbf{R}^3$ modeled as non-holonomic unicycles

$$\begin{aligned} \dot{x}_i(t) &= U\cos\gamma_i(t)\cos\psi_i(t), \\ \dot{y}_i(t) &= U\cos\gamma_i(t)\sin\psi_i(t), \\ \dot{z}_i(t) &= U\sin\gamma_i(t), \\ \dot{\psi}_i(t) &= u_i(t), \\ \dot{\gamma}_i(t) &= v_i(t), \end{aligned} \tag{2.1}$$

where $i = 1, 2, ..., n$, $x_i(t), y_i(t), z_i(t)$ specify the coordinates of the i-th vehicle measured relative to the earth fixed frame, and $\psi_i(t)$ and $\gamma_i(t)$ specify the heading and the flight path angle of the UAV at time t, respectively. We assume that each UAV flies at a constant speed U. The Euler angles define the orientation for each UAV: $\psi \in [-\pi, \pi]$ for heading and $\gamma \in [-\frac{\pi}{2}, \frac{\pi}{2}]$ for flight path; the corresponding rates are defined via the control inputs u_i and v_i for each i. These inputs are restricted to the vehicle's actuation limitations as

$$-\dot{\psi}_{\max} \le u_i(t) \le \dot{\psi}_{\max} \quad \text{and} \quad -\dot{\gamma}_{\max} \le v_i(t) \le \dot{\gamma}_{\max}. \tag{2.2}$$

We assume that the UAVs are identical and have their own nominal trajectories specified by a sequence of way-points in $\mathbf{R}^3$; we incorporate the minimum turn radius for each UAV to allow for a smooth transition between the way-points. Figure 2.1 depicts the vehicles' paths for the considered scenario. We also incorporate means of avoiding collisions between UAVs in case of malfunctions

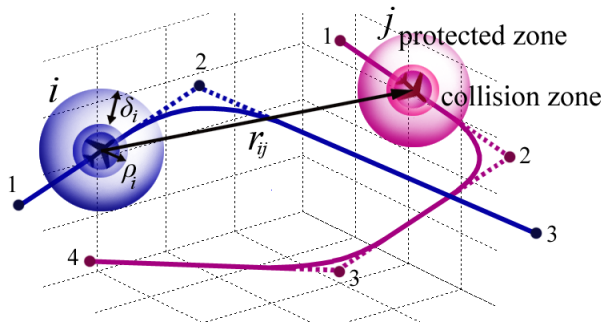


Figure 2.1: Trajectories of UAVs are composed of the sequences of desired way-points.

and disturbances, such as wind gusts. The resolution is developed in a cooperative manner which means that both conflicting UAVs run the same algorithm. The proposed deconfliction algorithm is subsequently integrated in the control law for the vehicles such that each vehicle only requires access to its own position, its planned path, and the location of the vehicle that is within a certain detection range.

A conflict between two vehicles is defined by two virtual spheres shown in Figure 2.1, referred to as the *collision* and the *protected* zones that specify distinct levels of safety constraints. In this figure, the i-th vehicle has a collision zone of radius $\rho_i > 0$ and a larger protected zone with radius $\rho_i + \delta_i$, with $\delta_i > 0$.

Definition 2.1.1 (Collision)*: A collision between vehicle i and j occurs when their collision zones intersect or $\|\mathbf{r_{ij}}\| < \rho_i + \rho_j$, where $\mathbf{r_{ij}}$ is a vector from vehicle i to j as measured from the vehicles' centers of mass.*

Definition 2.1.2 (Conflict)*: Two vehicles i and j are considered to be in a* conflict *or* loss of separation *if their protected zones overlap, i.e., when $\|\mathbf{r_{ij}}\| < \rho_i + \rho_j + \delta_i + \delta_j$.*

In the meantime, a group of vehicles are said to be *safe* or *conflict free* when their respective protected zones do not intersect during a given maneuver.

In order to implement our deconfliction algorithm, we assume that each unicycle has the capability to detect other vehicles within the range r_{ss}. The detection mechanism can be found in [34, 35]. We define the *neighborhood set* of the i-th vehicle as

$$\mathcal{N}_i \triangleq \{j \neq i : \|\mathbf{r_{ij}}\| \leq r_{ss}\}. \tag{2.3}$$

We note that since the set $\mathcal{N}_i$ depends on the inter-vehicle distances, it is essentially a dynamic set.

Our goal is to design the admissible inputs u_i and v_i for each UAV (3.1) in order to

(A) guarantee collision free convergence to the final destination or way-point in presence of static obstacles,

(B) stay in the proximity of the nominal path when permissible,

(C) guarantee deconfliction with the other UAV in the range of detection r_{ss} by not entering its protected zone, and

(D) guarantee turn-rates in accordance with (3.2).

The first two requirements necessitate the introduction of a *navigation function* that combines both goal and path attraction to create the control law (discussed in Section 2.2) while the third and the fourth requirements necessitate the use of a *swirling function* while ensuring acceptable turn-rates, trajectory overshoots, and minimum separation between the UAVs (discussed in Section 2.3).

2.2 Navigation Functions for UAVs in Presence of Static Obstacles

One of the ingredients for the proposed algorithm is a navigation function for each UAV that utilizes the location of the obstacles as well as its nominal trajectory. Navigation functions are developed from the intuition that the trajectory of a dynamical system asymptotically approaches a state corresponding to the minimum potential if it follows the descent direction of the potential. As detailed in the introduction, this approach has a long tradition in robotics and aerospace community. In the context of this work, the navigation function of each UAV is a mapping from the product of the *free configuration space* for each UAV, denoted by $\mathcal{F}_i$, to a real-valued potential, as

$$V_i : \mathcal{F} \to \mathbf{R}, \tag{2.4}$$

where $\mathcal{F} = \mathcal{F}_i \times \mathcal{F}_j$. In our case, the free configuration space for each vehicle will be the original spherical configuration space $\mathcal{C}_i$ with removed regions representing static obstacles. Let $\mathbf{q}_i(t) = [x_i(t), y_i(t), z_i(t)]^T$ represent the position of the i-th vehicle at time t and let $\mathbf{q}(t)$ consist of the

position vectors for the vehicle pair. The navigation function leads to a mechanism for steering the pair of UAVs from the initial state $\mathbf{q}_o$ to the destination $\mathbf{q}_d$ while staying in the product of their respective free configuration spaces. In this direction, each UAV is steered using the negative gradient of the underlying potential function by setting its desired velocity as

$$\mathbf{v}_i^d(t) = -\nabla_{\mathbf{q}_i} V_i(\mathbf{q}). \tag{2.5}$$

Such a velocity guidance, in turn, will lead the UAV to travel in a direction such that the potential is lowered and subsequently reaches its minimum. Therefore a navigation function should have a unique minimum at a destination state $\mathbf{q}_d$ and a maximum at the boundary of $\mathcal{F}$. Other properties are also required to guarantee desirable trajectories.

Definition 2.2.1 *Let $\mathbf{q}_{d_i} \in \mathcal{F}_i$ be the goal point for UAV i. A real-valued map $\mathrm{V}_i : \mathcal{F} \to \mathbf{R}$ is a navigation function for UAV i if for any fixed $\mathbf{q}_j$, as a function of $\mathbf{q}_i$, it is*

(a) a Morse function (its critical points are non-degenerate),

(b) smooth on $\mathcal{F}_i$ (at least a twice differentiable function),

(c) polar at $\mathbf{q}_{d_i}$ (is a unique minimum), and

(d) admissible on $\mathcal{F}_i$, i.e., it is uniformly maximal on the boundary of $\mathcal{F}_i$.

With these properties intact, specific navigation functions can be designed for many robotic path planning applications.

In [15] the navigation function is constructed based on a static environment, where the proof of convergence to $\mathbf{q}_{d_i}$ for each i while avoiding collisions with *static obstacles* is examined. In this work, we adopt the approach in [15] to construct one of the components of the proposed algorithm for UAV trajectory planning. It is important to note that the collision avoidance among the closest pair of moving vehicles is not guaranteed using the machinery developed in [15]; in subsequent sections, we introduce the so-called "swirling function" to address this latter issue. The form of the navigation function employed in our work for the i-th vehicle is parameterized as [15],

$$V_i(\mathbf{q}) = \frac{\Gamma_i(\mathbf{q})}{\sqrt[k]{\Gamma_i^k(\mathbf{q}) + \beta_i(\mathbf{q})}} \tag{2.6}$$

where,

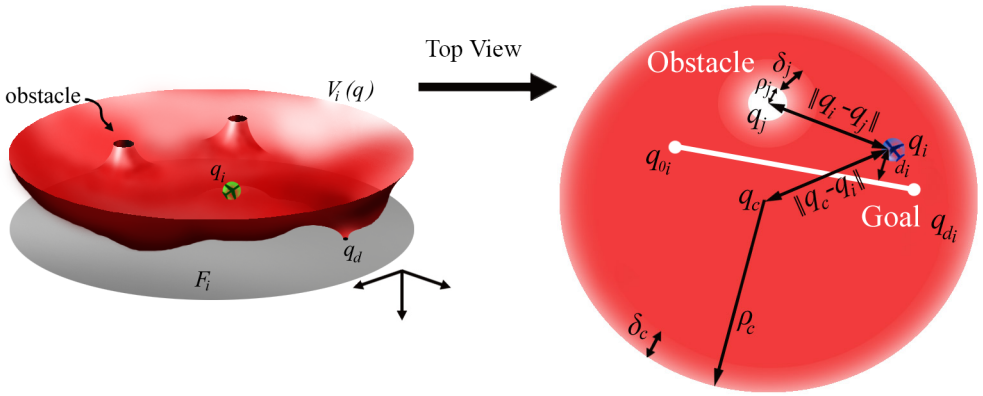


Figure 2.2: Free configuration space and navigation function when $\mathbf{q}_j$ is fixed.

- $\Gamma_i(\mathbf{q})$: is *the path planning function* which includes goal and path attraction terms, and reaches a unique minimum when the i-th vehicle is at its goal,
- $\beta_i(\mathbf{q})$: is the *obstacle function* that is designed to fade out when the i-th vehicle is about to enter the other vehicles protected zone or collide with static obstacles, and
- k is a positive tuning parameter.

In the work of Rahmani *et al.* [21], the function Γ_i has been chosen by adding the distance from the goal to the nominal path d_i weighted by a parameter K_p. This function provides the necessary emphasis between goal and path attraction, i.e.,

$$\Gamma_i(\mathbf{q}) = \|\mathbf{q}_i - \mathbf{q}_{d_i}\|^2 + K_p \beta \left(\frac{\|\mathbf{q}_i - \mathbf{q}_{d_i}\|}{r_p} \right) d_i^2, \tag{2.7}$$

where

$$d_i = \frac{\|(\mathbf{q}_{o_i} - \mathbf{q}_{d_i}) \times (\mathbf{q}_i - \mathbf{q}_{d_i})\|}{\|\mathbf{q}_{o_i} - \mathbf{q}_{d_i}\|}, \tag{2.8}$$

and β is a smooth function that transitions from zero to one as defined subsequently (see (2.12) below); it aims to dim down the path attraction component once the vehicle gets closer than r_p to the goal. Note that $\Gamma_i(\mathbf{q})$ is continuously differentiable on $\mathcal{F}$. In order to prevent two or more vehicles from converging to the same point and at the same time, while also trying to avoid each other, the following assumption for the initial and goal points has been adopted:

Assumption 2.2.1 *The initial and final positions for each vehicle is separated by the range of detection r_{ss}, i.e.,*

$$\|\mathbf{q}_{o_i} - \mathbf{q}_{o_j}\| > r_{ss} \quad \textit{and} \quad \|\mathbf{q}_{d_i} - \mathbf{q}_{d_j}\| > r_{ss}. \tag{2.9}$$

For the case with more than two way-points or when each vehicle has different number of way-points, this assumption should only be applied to all of initial points and the last way-points. This will be discussed further in Section 2.2.2. Moreover, as we would like the vehicles to arrive at their respective destinations around the same time, we assume that the initial and final positions of each vehicle is such that the total desired path lengths traversed by the vehicles are equal, i.e.,

$$\textit{for all } i \neq j, \quad \|\mathbf{q}_{d_i} - \mathbf{q}_{o_i}\| = \|\mathbf{q}_{d_j} - \mathbf{q}_{o_j}\|. \tag{2.10}$$

For missions with more than two way-points, we assume that the overall desired path lengths traversed by the vehicles are the same.

The "obstacle" function $\beta_i(\mathbf{q})$ is constructed from the product of pairwise functions of vehicles in $\mathcal{N}_i$ as well as the static obstacles.

Figure 2.2 depicts the free configuration space which is a sphere of radius ρ_c centered at $\mathbf{q}_c$; in this case,

$$\beta_i(\mathbf{q}) = \beta(r_{ic}). \prod_{j \in \mathcal{N}_i \cup \mathcal{S}} \beta(r_{ij}), \tag{2.11}$$

where $\mathcal{S}$ represents the static obstacles and $r_{ij} = \frac{(\|\mathbf{q}_j - \mathbf{q}_i\| - \rho_j)}{\delta_j}, r_{ic} = \frac{(\rho_c - \|\mathbf{q}_c - \mathbf{q}_i\|)}{\delta_c}$. In the meantime, the bump function has the form

$$\beta(r_{ij}) = \begin{cases} 0 & r_{ij} < 0, \\ f(r_{ij}) & 0 \leq r_{ij} < 1, \\ 1 & r_{ij} \geq 1, \end{cases} \tag{2.12}$$

where $f(r_{ij})$ is a high order polynomial, as shown in Figure 2.3, that smoothly transitions the obstacle function from uniformly zero at the collision zone boundary, to one at the protected zone boundary; the parameter δ_j measures the distance between these boundaries and δ_c measures the protected

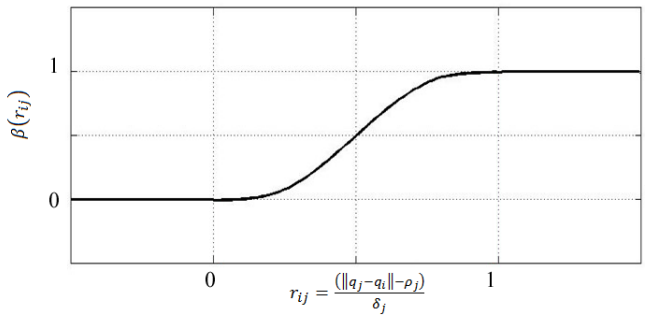


Figure 2.3: Sigmoidal shape of the collision bump function $\beta(r_{ij})$.

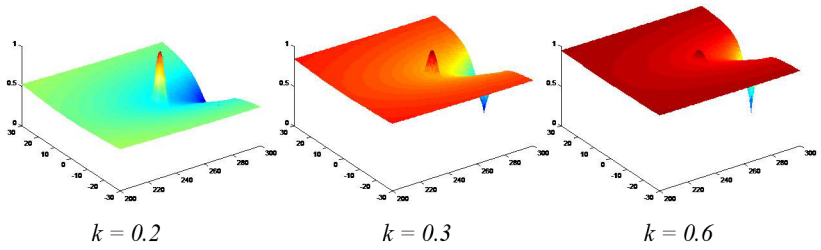


Figure 2.4: The effect of the parameter k on the navigation function.

distance to the boundary of the configuration space. The polynomial is chosen such that $\beta(\mathbf{q})$ is twice differentiable. It is noted that the choice of the bump function effects the turn-rate of the vehicles during collision avoidance; this will be discussed in more details in Section 2.4.3. The configuration radius ρ_c is chosen to be large enough such that $\mathbf{q}_{o_i}$, $\mathbf{q}_{d_i}$, and UAVs' trajectories stay away from the configuration boundary by more than δ_c. Since $\beta(r_{ij})$ is always one when $\mathbf{q}_i$ is outside $\mathbf{q}_j$'s protected zone, we do not need to include vehicles or static obstacles that are further away from vehicle j in order to compute the function β.

For each $\mathbf{q}_j$, at the boundary of $\mathcal{F}_i$, $\beta_i(\mathbf{q}) = 0$ and the function (2.6) uniformly reaches its maximum of one; on the other hand when $\mathbf{q}_i = \mathbf{q}_{d_i}$, $\Gamma_i(\mathbf{q}) = 0$, and the potential reaches its unique minimum of zero. The function is twice differentiable on $\mathcal{F}$ with the range being the unit interval. The parameter k changes the slope of the navigation function in a way that emphasizes obstacle avoidance or attraction toward the goal, as shown in Figure 2.4.

In order to check that the function (2.6) is in fact a navigation function according to Definition 2.2.1,[1] we still have to prove that it is a Morse function and smooth on $\mathcal{F}$. In this direction, first, we observe that the critical points of (2.6) are found by letting

$$\nabla_{\mathbf{q}_i} V_i(\mathbf{q}) = \frac{[k\beta_i(\mathbf{q})\nabla_{\mathbf{q}_i}\Gamma_i(\mathbf{q}) - \Gamma_i(\mathbf{q})\nabla_{\mathbf{q}_i}\beta_i(\mathbf{q})]}{k(\Gamma_i(\mathbf{q})^k + \beta_i(\mathbf{q}))^{1/k+1}} = 0, \tag{2.13}$$

that is

$$k\beta_i(\mathbf{q})\nabla_{\mathbf{q}_i}\Gamma_i(\mathbf{q}) = \Gamma_i(\mathbf{q})\nabla_{\mathbf{q}_i}\beta_i(\mathbf{q}). \tag{2.14}$$

On the other hand, this condition is valid in the following distinct cases:

(a) when $\nabla_{\mathbf{q}_i}\Gamma_i(\mathbf{q}) = 0$ (when $\Gamma_i(\mathbf{q}) = 0$), $\nabla_{\mathbf{q}_i}\beta_i(\mathbf{q}) = 0$, and $\beta_i(\mathbf{q}) = 1$: in this case, the identities $\nabla_{\mathbf{q}_i}\beta_i(\mathbf{q}) = 0$ and $\beta_i(\mathbf{q}) = 1$ imply that the i-th vehicle is far from the boundary of the configuration space and other moving obstacles. In the meantime, the identity $\nabla_{\mathbf{q}_i}\Gamma_i(\mathbf{q}) = 0$ means that the vehicle is at the destination. We can prove that this equilibrium is in fact the minimum of the navigation function by evaluating the Hessian $\nabla^2_{\mathbf{q}_i} V_i(\mathbf{q})$ as shown in the following proposition.

Proposition 2.2.2 *The navigation function (2.6) has a non-degenerate minimum when the i-th vehicle reaches its destination; this minimum is outside of all other vehicles' protected zones.*

Proof. See Appendix A ■

(b) when $\nabla_{\mathbf{q}_i}\Gamma_i(\mathbf{q}) = 0$ (when $\Gamma_i(\mathbf{q}) = 0$), $\nabla_{\mathbf{q}_i}\beta_i(\mathbf{q}) = 0$, and $\beta_i(\mathbf{q}) = 0$: The first condition means the vehicle is at the destination. The identities $\nabla_{\mathbf{q}_i}\beta_i(\mathbf{q}) = 0$ and $\beta_i(\mathbf{q}) = 0$ also imply that the vehicle is at the boundary of an obstacle collision zone. This critical point is located at the boundary of the configuration space. However, Assumption 2.2.1 suggests that the vehicle's destination is away from obstacles. Therefore, the vehicle can never be at this critical point.

(c) when $\nabla_{\mathbf{q}_i}\Gamma_i(\mathbf{q}) = 0$ (when $\Gamma_i(\mathbf{q}) = 0$) and $\nabla_{\mathbf{q}_i}\beta_i(\mathbf{q}) \neq 0$ (when $0 < \beta_i(\mathbf{q}) < 1$): The first condition means that the i-th vehicle is at the destination. The second condition translates to having the vehicle near other moving or static obstacles. Substituting these conditions into

[1] We will subsequently use this fact to infer that each UAV will converge to it goal point while avoiding static obstacles.

(5.1) yields $\nabla^2_{q_i} V_i(\mathbf{q}_i) = \frac{\nabla^2_{q_i}\Gamma_i(\mathbf{q}_i)}{(\beta_i(\mathbf{q}))^{1/k}} = \frac{2I}{(\beta_i(\mathbf{q}))^{1/k}}$ which implies that this point is non-degenerate. However, Assumption 2.2.1 suggests that the vehicle can never be at such a critical point.

(d) when $\nabla_{\mathbf{q}_i}\Gamma_i(\mathbf{q}) \neq 0$ (when $\Gamma_i(\mathbf{q}) \neq 0$), $\nabla_{\mathbf{q}_i}\beta_i(\mathbf{q}) = 0$, and $\beta_i(\mathbf{q}) = 0$: The first condition means that the i-th vehicle is not at the destination. The identities $\nabla_{\mathbf{q}_i}\beta_i(\mathbf{q}) = 0$ and $\beta_i(\mathbf{q}) = 0$ also imply that the vehicle is at the boundary of an obstacle collision zone. Substituting these conditions into (5.1) yields

$$\nabla^2_{q_i} V_i(\mathbf{q}_i) = -\frac{\nabla^2_{q_i}\beta_i(\mathbf{q}_i)}{k\Gamma_i^k(\mathbf{q})}.$$

Since we chose the bump function such that $f''(0) = 0$, it follows that $\nabla^2_{q_i}\beta_i(\mathbf{q}_i) = 0$ at this point. However this type of critical point is not in the interior of the configuration space.

(e) when $\nabla_{\mathbf{q}_i}\Gamma_i(\mathbf{q}) \neq 0$ (when $\Gamma_i(\mathbf{q}) \neq 0$), $\nabla_{\mathbf{q}_i}\beta_i(\mathbf{q}) \neq 0$ (when $0 < \beta_i(\mathbf{q}) < 1$), and $k\beta_i(\mathbf{q})\nabla_{\mathbf{q}_i}\Gamma_i(\mathbf{q}) = \Gamma_i(\mathbf{q})\nabla_{\mathbf{q}_i}\beta_i(\mathbf{q})$: The first condition means that the i-th vehicle is not at the destination; the second condition translates to having the vehicle near other moving obstacles. The equilibrium is not the desired one. In the meantime, the third condition leads to

$$k\frac{\nabla_{\mathbf{q}_i}\Gamma_i(\mathbf{q})}{\Gamma_i(\mathbf{q})} = \frac{\nabla_{\mathbf{q}_i}\beta_i(\mathbf{q})}{\beta_i(\mathbf{q})},$$

which means that the gradients, due to the path planning and obstacle avoidance are in the same direction, but scaled by parameter k, and there exists a big enough k that causes this undesired equilibrium to be pushed further toward the moving obstacle. The following proposition analyzes the stability of this type of critical points.

Proposition 2.2.3 *The critical points analyzed above are in the interior of the free configuration space $\mathcal{F}$. There are at least as many saddle points as the number of obstacles. In fact, it is possible to find a lower bound for the parameter k such that*

1. *each critical point of the navigation function is close to one of the obstacles, and*
2. *the critical points near the obstacles are not a local minimum, but only a saddle point, and not degenerate.*

Proof. See Ref. [51]. ■

From Propositions 2.2.2 and 2.2.3, we can conclude that (2.6) is a Morse function in $\mathcal{F}_i$.

Proposition 2.2.4 *For each $\mathbf{q}_j$, the navigation function (2.6) is twice differentiable on $\mathcal{F}_i$.*

Proof. From the expression for the Hessian of $V_i(\mathbf{q})$ in (5.1), we can see that $\Gamma_i(\mathbf{q})$ and $\beta_i(\mathbf{q})$ can not be zero at the same time in the interior of $\mathcal{F}_i$. Hence, the terms $\Gamma_i(\mathbf{q})^k + \beta_i(\mathbf{q}) > 0$ and $\nabla^2_{\mathbf{q}_i} V_i(\mathbf{q})$ are defined everywhere on $\mathcal{F}_i$. ■

While we show that at any instant of time, the function (2.6) satisfies all of the conditions in Definition 2.2.1, these conditions can only guarantee a goal point convergence, path attraction, and static obstacles avoidance, but not the collision avoidance with another (moving) UAV. This latter property will be guaranteed with the help of a swirling function presented in Section 2.3. However, we first discuss two aspects of the navigation-based control law in the absence of swirling.

2.2.1 Navigation-based UAV Control

The inner-loop controller of the UAVs to follow the desired velocity is of the form

$$\begin{aligned} u_i(t) &= K_{\psi_i}(\psi_i^d(t) - \psi_i(t)) + \dot{\psi}_i^d(t), \\ v_i(t) &= K_{\gamma_i}(\gamma_i^d(t) - \gamma_i(t)) + \dot{\gamma}_i^d(t), \end{aligned} \tag{2.15}$$

where

$$\psi_i^d = \mathbf{atan_2}\left(v_{y_i}^d, v_{x_i}^d\right), \gamma_i^d = \mathbf{atan_2}\left(v_{z_i}^d, \sqrt{(v_{x_i}^d)^2 + (v_{y_i}^d)^2}\right);$$

the superscript "d" indicates the desired value of the corresponding state, and $\mathbf{atan_2}$ denotes the four quadrants arctangent function. We now set $[v_{x_i}^d, v_{y_i}^d, v_{z_i}^d]^T = \mathbf{v}_i^d(\mathbf{q}) = -\nabla_{\mathbf{q}_i} V_i(\mathbf{q})$, and K_{ψ_i} and K_{γ_i} are heading and pitching regulation gains, respectively. The feed-forward terms $\dot{\psi}_i^d$ and $\dot{\gamma}_i^d$ can now be calculated using $[\dot{v}_{x_i}^d, \dot{v}_{y_i}^d, \dot{v}_{z_i}^d]^T = \dot{\mathbf{v}}_i^d(\mathbf{q}) = -\nabla_{\mathbf{q}} \nabla_{\mathbf{q}_i} V_i(\mathbf{q}) \dot{\mathbf{q}}$, which represents the rate of change of the potential's slope as we travel in the $\dot{\mathbf{q}}_i$ direction. These terms, in turn, increase the tracking performance of the UAVs to track the desired velocity $\mathbf{v}_i^d(\mathbf{q})$ once the error has converged to zero. By solving (2.15) we have that $\psi_i(t) = [\psi_i(0) - \psi_i^d(0)]e^{-K_{\psi_i} t} + \psi_i^d(t)$ and that implies that the error dynamics $\psi_i(t) - \psi_i^d(t) \to 0$ as $t \to \infty$.

Figure 2.5 depicts the gradient vector field through the configuration space and the parameters corresponding to UAV deconfliction including the swirling radius r_{ss} that will be discussed in Section 2.3 The gradient is numerically computed locally to save the computational cost. We note that

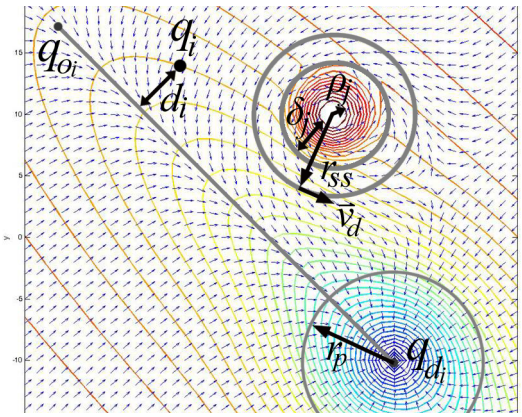


Figure 2.5: The gradient vector field throughout the free configuration space when the swirling effect is included.

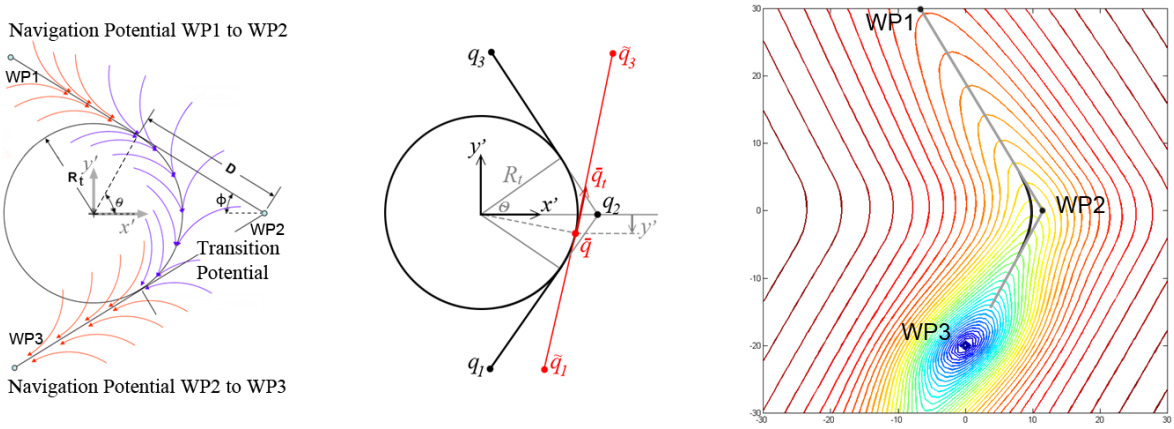


Figure 2.6: Transition potential at a way-point corner.

the vector field varies in accordance with the location of the vehicle and its neighbor, and therefore it is a function of time and has to be recalculated at every time-step.

2.2.2 Way-points Steering

One of the methods to direct a UAV through way-points is to incorporate the aircraft minimum turn radius to form curve paths around the way-points corners. The flight path design involves similar consideration as e.g. Chandler *et al.* [28]. This arc is defined in an auxiliary coordinate frame with the origin at the center of the circle of the arc and the x-axis toward the way-point corner as shown in Figure 2.6. The transition mode begins when the aircraft approaches the primary way point (less

than D apart) and is required to turn toward the next way-point. The parameter D is defined as $D = R_t/\tan\phi$, where R_t is a minimum turn radius and ϕ is half of the angle at the way-point corner. Since the potential is re-calculated at every time step, it is adjusted such that the desired path is,

$$\hat{x} = \begin{cases} x' - R_t\cos\theta + \sqrt{R_t^2 - y'^2}, & |y'| \le R_t\sin\theta, \\ x' - (|y'| - R_t\sin\theta)\tan\theta, & \text{otherwise}, \end{cases} \tag{2.16}$$

where x' and y' are aircraft coordinates, $\hat{x}$ is an adjusted coordinate, and θ is half of the turning angle. Once the vehicle arrives at a distance D from the way-point, the relation (2.16) describes the prescribed trajectory for it to follow. We note that regardless of how the location of the vehicle maps on the $x' - y' - z'$ coordinate, we utilize the corresponding point on this curve and its tangent to define a path for that time instance. This path, along with the location of the moving obstacles in the configuration space, are then used to construct the navigation function (2.6). The desired destination during the transition is subsequently chosen along the tangent of the curve where the potential value matches the boundary condition at the initiation of the transition.

To describe how this navigation function is constructed, we assign a three way-point coordinates expressed in $x' - y' - z'$-frame as $\mathbf{q_1}$, $\mathbf{q_2}$, and $\mathbf{q_3}$ where $\mathbf{q_2}$ represents the middle way-point as shown in Figure 2.6. Since we will only discuss vehicle-i way-points steering algorithm, the subscript i will be omitted in the notation.

We incorporate all three way-points into constructing path-planning function $\Gamma(\mathbf{q})$. The location of the final goal point and the initial point are adjusted base on the the value of y' in an auxiliary coordinate frame. These adjusted values are situated along the tangent of the curve during the transition. The point along the curve can be represented as $\bar{\mathbf{q}} = [\sqrt{R_t^2 - y'^2}; y'; 0]$ whereas the unit vector tangential to it is $\bar{\mathbf{q}}_\mathbf{t} = [-y'/R_t; \sqrt{R_t^2 - y'^2}/R_t; 0]$. The adjusted goal point and the adjusted initial point can then be represented as

$$\tilde{\mathbf{q}}_\mathbf{3} = \begin{cases} \mathbf{q_2} + \frac{\|\mathbf{q_3} - \mathbf{q_2}\|}{\|\mathbf{q_2} - \mathbf{q_1}\|}(\mathbf{q_2} - \mathbf{q_1}), & y' \le -R_t\sin\theta, \\ \bar{\mathbf{q}} + \left[\frac{R_t\sin\theta - y'}{\cos\theta} + \|\mathbf{q_3} - \mathbf{q_2}\| - R_t\tan\theta\right]\bar{\mathbf{q}}_\mathbf{t}, & |y'| < R_t\sin\theta, \\ \mathbf{q_3}, & y' \ge R_t\sin\theta, \end{cases}$$

and

$$\tilde{\mathbf{q}}_1 = \begin{cases} \mathbf{q_1}, & y' \leq -R_t \sin\theta, \\ \bar{\mathbf{q}} - \left[\frac{R_t \sin\theta + y'}{\cos\theta} + \|\mathbf{q_2} - \mathbf{q_1}\| - R_t \tan\theta\right] \bar{\mathbf{q}}_\mathbf{t}, & |y'| < R_t \sin\theta, \\ \mathbf{q_2} - \frac{\|\mathbf{q_2} - \mathbf{q_1}\|}{\|\mathbf{q_3} - \mathbf{q_2}\|}(\mathbf{q_3} - \mathbf{q_2}), & y' \geq R_t \sin\theta, \end{cases}$$

respectively. The adjusted path-planning function then becomes

$$\tilde{\Gamma}(\mathbf{q}) = \|\mathbf{q} - \tilde{\mathbf{q}}_\mathbf{3}\|^2 + K_p \beta \left(\frac{\|\mathbf{q} - \tilde{\mathbf{q}}_\mathbf{3}\|}{r_p}\right) \tilde{d}^2,$$

where

$$\tilde{d} = \frac{\|(\tilde{\mathbf{q}}_\mathbf{1} - \tilde{\mathbf{q}}_\mathbf{3}) \times (\mathbf{q} - \tilde{\mathbf{q}}_\mathbf{3})\|}{\|\tilde{\mathbf{q}}_\mathbf{1} - \tilde{\mathbf{q}}_\mathbf{3}\|}.$$

If desired, the obstacle function can also be included as suggested by (2.6). For the application where the way-point corners are small, this method requires the UAV to turn long before reaching the way-point. This may not be practical for a surveillance or monitoring mission. For a case like this, two way-points should be assigned accordingly to allow the UAV to make a turn surrounding the way-point of interest.

2.3 Swirling Function for UAV Deconfliction

Due to the nature of the unicycle model, necessitating control over heading only, and the fact that other UAVs are not "static obstacles," the collision avoidance of the algorithm can not be guaranteed by merely following the negative gradient of the navigation function. Moreover, the navigation function can lead to an unbounded force or turn-rate commands which are not suitable for a unicycle model with turn-rate constraints. For example, if the protected zone radius is set too small as compared with the vehicle's speed, the UAV needs to turn sharply in order to avoid the moving obstacle. For practical reasons, large turn-rates are neither desirable nor practical. In order to alleviate this situation, swirling effect has been introduced in Ref. [21] in order to improve the deconfliction performance and to help guarantee the collision avoidance under such dynamic limitations. The swirling effect also helps breaking the symmetry in order to avoid being trapped in saddle points of the navigation function

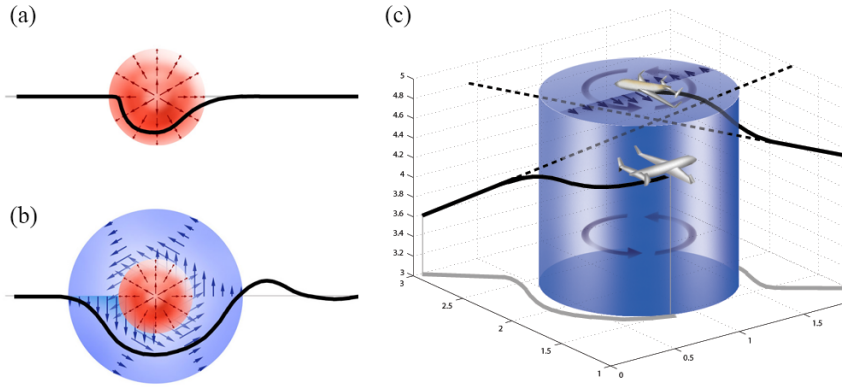


Figure 2.7: Deconfliction trajectories: (a) without the swirling effect, the inner circle notify the moving obstacle protected zone with the gradient vectors pointing outward, (b) when the swirling term has been added along the outer circle, (c) the swirling effect is applied in 3D case when both aircraft is ascending but within the range of detection

around the moving obstacle.

A swirling effect is added to the gradient in order to "rotate" the aircraft counterclockwise around the nearby moving obstacle. This effect is augmented gradually from the distance when the deconfliction sensor detects the moving obstacle at distance r_{ss}; the effect is initialized with zero magnitude and is allowed to increase to the maximum level at the boundary of the collision zone. The parameter K_s defines the maximum relative size of the swirling effect with respect to the original gradient. The spinning gradient vector field around the moving obstacles tends to break the symmetry for the case when two vehicles are exactly heading onto each other.

For the case when the vehicles are ascending or descending along their respective desired paths, adjusting the heading command in the vertical direction can create higher a angle of attack which causes the aircraft to stall; therefore the swirling effect is added only along the horizontal direction while the desired vertical gradient at a corresponding location stays unaffected. The effect can be viewed as a vertical circular cylinder of a rotating vector field added to the original gradient of the navigation function as shown in Figure 2.7(c). The adjusted gradient for vehicle i can be derived as,

$$\begin{aligned} \nabla V_i^{\text{new}} &= \nabla V_i + K_s \beta \left(\frac{r_{ss} - \|\mathbf{q}_i - \mathbf{q}_j\|}{r_{ss}} \right) \|\nabla V_i\| \mathbf{v}_j^{ss} \\ \mathbf{v}_j^{ss} &= \frac{(\mathbf{q}_i - \mathbf{q}_j) \times k_z}{\|\mathbf{q}_i - \mathbf{q}_j\|}, \end{aligned} \tag{2.17}$$

where $\mathbf{q}_i$ and $\mathbf{q}_j$ are the locations of the UAV and the moving obstacle, respectively, and k_z is the unit vector in the z direction of the earth frame. The idea can also be extended for the case when more than one vehicle is within the range of detection, in this case the superposition of the effects from each vehicle may be used with a bump function (2.12) in order to emphasize the effects from the closer vehicles

$$\nabla V_i^{\text{new}} = \nabla V_i + \sum_{j \in \mathcal{N}_i} K_{s_j} \beta \left(\frac{r_{ss_j} - \|\mathbf{q}_i - \mathbf{q}_j\|}{r_{ss_j}} \right) \|\nabla V_i\| \mathbf{v}_j^{ss}. \tag{2.18}$$

Figure 2.5 shows the swirling effect around a moving obstacle with radius r_{ss}. Figures 2.7(a) and (b) compare the trajectories for the scenarios with and without the swirling effect, when two vehicles fly head-on into each other. For a completely symmetric case of this example, the original gradient direction will change to the opposite direction once the UAV enters the other UAV's protected zone and creates a stagnation point, accompanied by a high turn-rate command. The counterclockwise swirling effect helps the UAV to always turn to the "right" when encountering an obstacle. However this effect creates a trajectory overshoot once the conflict is resolved as the swirling component still continues to rotate the aircraft after it passes the moving vehicle; it might also steer the aircraft into a high turn rate zone. We thereby propose an overshoot alleviation mechanism by reducing the swirling effect following the resolution of the conflict.

2.3.1 Trajectories Overshoot and Swirling Effects Reduction

The overshoot caused by the swirling effect reduces the aircraft tracking performance and results in some delay for the aircraft to go back to the nominal path. One of the ways to solve this problem is to reduce the swirling effect once two vehicles are out of the course of collision. For this purpose, we employ the concept of collision cone and relative velocity used in Refs. [6, 9, 12]. A collision cone is defined by the area inside the lines drawn from the vehicle to the obstacle's collision zone. The two vehicles are out of the course of collision if their relative velocity stays outside the collision cone. Next, let us define the *safety angle* to help measure how far the two vehicles are from the collision.

Definition 2.3.1 *A safety angle of a collision for vehicle i from vehicle j, $\theta_{s_{ij}}$, is the angle between the relative velocity $\mathbf{v}_{ij} = \mathbf{v}_i - \mathbf{v}_j$ and the collision cone drawn from vehicle i to vehicle j's collision*

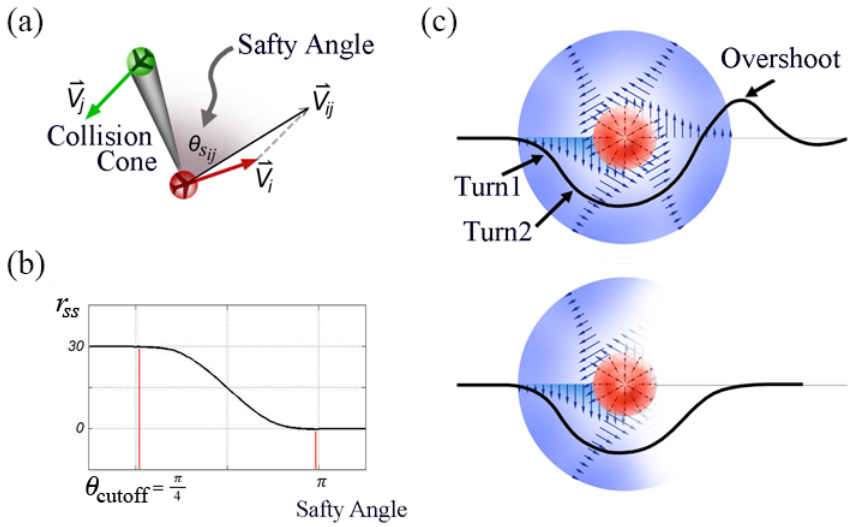


Figure 2.8: (a) The collision cone and the safety angle. (b) The plot of swirling radius adjusted by the safety angle. (c) The overshoot disappears after swirling effect is dimmed down

zone, where $\mathbf{v}_i$ *indicates the velocity vector of vehicle* i *(this is shown in Figure 2.8(a)).*

As this relative velocity stays farther from the collision cone (larger safety angle), the two vehicles are "less in conflict." Our approach is to reduce the swirling radius according to the safety angle from r_{ss} at the cut-off angle θ_{cutoff} to zero when the safety angle equals 180 degree. The swirling reduction is done by the same pattern using the bump function in (2.12). For smooth transition of the aircraft's trajectory, the cutoff angle is chosen to be a value in the interval of $[\pi/4, 3\pi/8)$ as shown in Figure 2.8(b). The swirling radius is then adjusted as

$$r_{ss}^{\text{adjusted}}(\theta_s) = \begin{cases} r_{ss} & \theta_s \leq \theta_{\text{cutoff}}, \\ r_{ss}\beta\left(\frac{\pi-\theta_s}{\pi-\theta_{\text{cutoff}}}\right) & \theta_s > \theta_{\text{cutoff}}. \end{cases} \tag{2.19}$$

For the case when more than 2 vehicles are in the range of detection, the swirling effect from each neighbor are summed according to (2.18) and the radius r_{ss_j}, or the relative magnitude K_{s_j} is adjusted based on the safety angle $\theta_{s_{ij}}$ between each pair of vehicles. Figure 2.8(c) shows that this approach helps prevent the vehicle from having an overshoot into the high turn-rate zone, a topic that will be discussed further in Section 2.4.3.

2.3.2 Swirling Effect and Guaranteed Deconfliction

In this section, we explore the guaranteed convergence of the deconfliction algorithm to collision-free state under the navigation law dictated by the swirling effect. In particular, we prove that when two vehicles are within the radar range of each other (distance r_{ss} apart), their protected zones will not intersect at all times until they exit the proximity detection zone. In order to simplify the proof, we will adopt the following three assumptions, where the second and the third will be subsequently relaxed:

Assumption 2.3.1 *Since the swirling effect is added along the horizontal direction while the vertical desired heading derived from the navigation function remains unchanged, the guaranteed deconfliction can be viewed from the projection of the flight paths onto the horizontal plane. It is thus sufficient to show that the swirling effect keeps the two vehicles apart if it achieves this objective on the projected horizontal plane. Therefore, we can examine the guaranteed deconfliction for the case of two vehicles on the same plane and adopt the model,*

$$\begin{aligned} \dot{x}_i(t) &= U_{h_i} \cos\psi_i(t), \quad i = 1, 2, \\ \dot{y}_i(t) &= U_{h_i} \sin\psi_i(t), \\ \dot{\psi}_i(t) &= K_{\psi_i}(\psi_i^d(t) - \psi_i(t)) + \dot{\psi}_i^d(t), \end{aligned} \tag{2.20}$$

where ψ_i^d $(i = 1, 2)$ is the desired headings for the UAV i as derived from (2.16) and (2.17) and U_{h_i} is the projection of its velocity to the horizontal plane.

Assumption 2.3.2 *For simplification of the proof, we first look at the case where two conflicting UAVs has the same flight path angle or $\cos\gamma_1 = \cos\gamma_2$ and that $U_{h_1} = U_{h_2} = U_h$. Since we only look at projection in 2D, the subscript h is dropped in the notation. The result when $U_{h_1} \neq U_{h_2}$ will be discussed toward the end of the section.*

Assumption 2.3.3 *During the deconflicting maneuver, the vehicles' aim to avoid entering each other protected zones; hence the gradient due to the navigation function comes from the path planning term only. The original gradient (before adding swirling term) is rather uniform around the area that is far from the destination. By neglecting this term with respect to the swirling effect, the behavior of deconflicting maneuver is expected to be minimally effected. Thus, we shall first drop the contribution*

of the navigation function in our collision avoidance proof below; in Section 2.3.3 , we provide a collision avoidance analysis when this term is added back. In this former case (2.17) assumes the form

$$\nabla \mathrm{V}_i \quad = \quad \frac{(\mathbf{q}_i - \mathbf{q}_j) \times \vec{k}_z}{\|\mathbf{q}_i - \mathbf{q}_j\|},$$

and accordingly the desired heading is

$$\psi_i^d(t) \approx \psi_i^{d,ss}(t) = \mathbf{atan}_2(x_i(t) - x_j(t), y_j(t) - y_i(t)), \tag{2.21}$$

where the superscript "ss" denotes the desired heading obtained from the swirling effect. Hence, the feed-forward term can be derived from

$$\begin{aligned}
\dot{\psi}_i^{d,ss}(t) &= \frac{(y_j(t)-y_i(t))(\dot{x}_j(t)-\dot{x}_i(t))-(x_j(t)-x_i(t))(\dot{y}_j(t)-\dot{y}_i(t))}{-(x_j(t)-x_i(t))^2+(y_j(t)-y_i(t))^2} \\
&= U\left[\frac{\cos\psi_i^{d,ss}(t)(\cos\psi_j(t)-\cos\psi_i(t))+\sin\psi_i^{d,ss}(t)(\sin\psi_j(t)-\sin\psi_i(t))}{-\sqrt{(x_j(t)-x_i(t))^2+(y_j(t)-y_i(t))^2}}\right] \\
&= U\left[\frac{\cos(\psi_j^{d,ss}(t)-\psi_j(t))+\cos(\psi_i^{d,ss}(t)-\psi_i(t))}{\sqrt{(x_j(t)-x_i(t))^2+(y_j(t)-y_i(t))^2}}\right].
\end{aligned}$$

Now by letting $\bar{x}(t) = x_1(t) - x_2(t)$, $\bar{y}(t) = y_1(t) - y_2(t)$, the system of two vehicles under the swirling effects can be represented as,

$$\begin{aligned}
\dot{\bar{x}}(t) &= U(\cos\psi_1(t) - \cos\psi_2(t)), && \bar{x}(0) = -r_{ss} \\
\dot{\bar{y}}(t) &= U(\sin\psi_1(t) - \sin\psi_1(t)), && \bar{y}(0) = 0 \\
\dot{\psi}_1(t) &= K(\psi_1^{d,ss}(t) - \psi_1(t)) + U\left(\frac{\cos(\psi_2^{d,ss}(t)-\psi_2(t))+\cos(\psi_1^{d,ss}(t)-\psi_1(t))}{\sqrt{\bar{x}^2(t)+\bar{y}^2(t)}}\right), && \psi_1(0) = \psi_{1_0} \\
\dot{\psi}_2(t) &= K(\psi_2^{d,ss}(t) - \psi_2(t)) + U\left(\frac{\cos(\psi_2^{d,ss}(t)-\psi_2(t))+\cos(\psi_1^{d,ss}(t)-\psi_1(t))}{\sqrt{\bar{x}^2(t)+\bar{y}^2(t)}}\right), && \psi_2(0) = \psi_{2_0}
\end{aligned}$$

with

$$\psi_1^{d,ss}(t) = \mathbf{atan}_2(\bar{x}(t), -\bar{y}(t)), \psi_2^{d,ss}(t) = \mathbf{atan}_2(-\bar{x}(t), \bar{y}(t)), \tag{2.22}$$

where K represents the inner-loop control gain of both vehicles K_{ψ_1} and K_{ψ_2}; additionally, ψ_{1_0} and ψ_{2_0} are the initial headings when the UAVs enter the swirling zone at distance r_{ss} apart. Figure 2.9

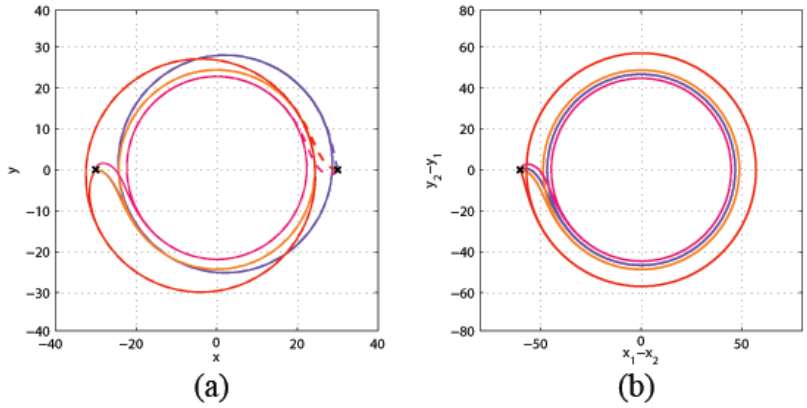


Figure 2.9: (a) Trajectories of 2 UAVs solely under the swirling effects with 4 different initial headings. (b) Trajectories on $(x_1 - x_2), (y_1 - y_2)$-plane.

depicts the trajectories of two vehicles according to (2.22) with 4 different pairs of initial headings. Note that the swirling effect drives the two vehicles to converge into the same circular limit cycle and have them stay on the opposite side of the circle heading in opposite directions. The headings at the equilibrium are equal to the desired headings, i.e., $\psi_i = \psi_i^{d,ss}$. In the meantime, different pairs of initial heading results in convergence to different limit cycles. In order to prove the behavior of the pair of UAVs under the swirling effect, we use LaSalle's Invariance Principle to show that the vehicles in (2.22) converge to a circular limit cycle, whose diameter is the minimum distance between the two UAVs. In fact, this limit cycle can be written as an invariant set

$$\mathcal{M} = \{(\bar{x}, \bar{y}, \psi_1, \psi_2) \in \mathbf{R}^4; \bar{x}^2 + \bar{y}^2 = R^2, \psi_1 = \mathbf{atan_2}(\bar{x}, -\bar{y}), \psi_2 = \mathbf{atan_2}(-\bar{x}, \bar{y})\}, \tag{2.23}$$

where R is a radius of the limit cycle.

Theorem 2.3.2 *The system of two UAVs under the swirling effect in (2.22) will converge to the limit cycle (2.23) and will keep a distance greater than* $2R$ *when initialized from* r_{ss} *apart with* ψ_{1_0} *and* ψ_{2_0} *initial headings.*

Proof. First, we show that $\mathcal{M}$ is positively invariant by

$$\begin{aligned}
\tfrac{d}{dt}(\bar{x}^2+\bar{y}^2-R^2) &= 2(\bar{x}\dot{\bar{x}}+\bar{y}\dot{\bar{y}}) \\
&= 2U(\bar{x}(\cos\psi_1-\cos\psi_2)+\bar{y}(\sin\psi_1-\sin\psi_2)) \\
&= 4UR(\sin(\psi_1^{d,ss}-\psi_1)+\sin(\psi_2^{d,ss}-\psi_2))=0 \\
\tfrac{d}{dt}\left(\mathbf{atan}_2\left(\bar{x},-\bar{y}\right)-\psi_1\right) &= \tfrac{d}{dt}\left(\mathbf{atan}_2\left(\bar{x},-\bar{y}\right)\right)-\dot{\psi}_1 \\
&= \tfrac{d}{dt}(\mathbf{atan}_2(\bar{x},-\bar{y}))-\left[K(\mathbf{atan}_2(\bar{x},-\bar{y})-\psi_1)-\tfrac{d}{dt}(\mathbf{atan}_2(\bar{x},-\bar{y}))\right] \\
&= -\left[K\left(\mathbf{atan}_2\left(\bar{x},-\bar{y}\right)-\psi_1\right)\right] \\
&= 0=\tfrac{d}{dt}\left(\mathbf{atan}_2\left(-\bar{x},\bar{y}\right)-\psi_2\right).
\end{aligned}$$

and hence, if $(\bar{x}(0),\bar{y}(0),\psi_1(0),\psi_2(0))\in\mathcal{M}$, then $(\bar{x}(t),\bar{y}(t),\psi_1(t),\psi_2(t))\in\mathcal{M}$, for $t\geq 0$. The motion in the set $\mathcal{M}$ is characterized by $\dot{\psi}_1(t)=\dot{\psi}_2(t)=U/R$ which shows that $\mathcal{M}$ is a limit cycle for (2.22), where the state vector moves counterclockwise. To examine whether $\mathcal{M}$ is attractive, consider the continuously differentiable function $V:\mathbf{R}^4\to\mathbf{R}$ representing the deviation of the system from being on the limit cycle as

$$V(\bar{x},\bar{y},\psi_1,\psi_2)=(\bar{x}^2+\bar{y}^2-R^2)^2+(\psi_1^{d,ss}-\psi_1)^2+(\psi_2^{d,ss}-\psi_2)^2$$

and consider its time derivative

$$\dot{V}(\bar{x},\bar{y},\psi_1,\psi_2)=4(\bar{x}^2+\bar{y}^2-R^2)(\bar{x}\dot{\bar{x}}+\bar{y}\dot{\bar{y}})+2(\psi_1^{d,ss}-\psi_1)(\dot{\psi}_1^{d,ss}-\dot{\psi}_1)+2(\psi_2^{d,ss}-\psi_2)(\dot{\psi}_2^{d,ss}-\dot{\psi}_2).$$

It is straightforward to see that when the vehicles start on the limit cycle $\mathcal{M}$ or $(\bar{x}(0),\bar{y}(0),\psi_1(0),\psi_2(0))=(r_{ss},0,\ \psi_1^{d,ss}(0),\psi_2^{d,ss}(0))$, they will stay on $\mathcal{M}$ for $t\geq 0$ and $\bar{x}^2+\bar{y}^2=R^2,\psi_1=\psi_1^{d,ss},\psi_2=\psi_2^{d,ss}$. Hence in this case $\dot{V}=0$ We now proceed to examine the case when the two vehicles start outside the limit cycle. In this venue, let

$$\begin{aligned}
\dot{V}_1 &= 4(\bar{x}^2+\bar{y}^2-R^2)(\bar{x}\dot{\bar{x}}+\bar{y}\dot{\bar{y}}) \\
&= 4U(\bar{x}^2+\bar{y}^2-R^2)(\bar{x}(\cos\psi_1-\cos\psi_2)+\bar{y}(\sin\psi_1-\sin\psi_2)) \\
&= 4U(\bar{x}^2+\bar{y}^2-R^2)\sqrt{\bar{x}^2+\bar{y}^2}(\sin(\psi_1^{d,ss}-\psi_1)+\sin(\psi_2^{d,ss}-\psi_2)).
\end{aligned}$$

We now check the inequality $\dot{V}_1<0$ by examining the following distinct cases:

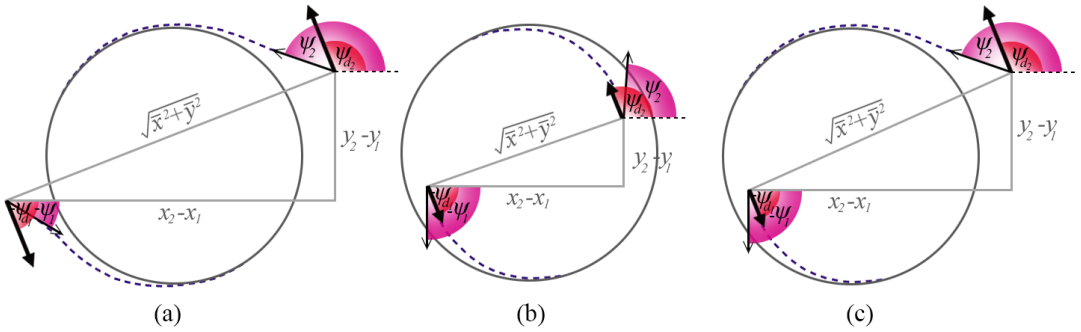


Figure 2.10: The headings and desired headings of 2 vehicles under swirling effects when they are: (a) in the course of collision, (b) facing away from each other, and (c) in similar directions.

(a) Two vehicles start with a course of collision: As ψ_{1_0}, ψ_{2_0} face each other (or $\psi_{1_0} \in (-\frac{\pi}{2}, \frac{\pi}{2}), \psi_{2_0} \in (\frac{\pi}{2}, \frac{3\pi}{2})$), swirling effects will direct the vehicles by $\psi_1^{d,ss}, \psi_2^{d,ss}$ to go around each other counter-clockwise and converge to $\mathcal{M}$. Both vehicles converge to $\mathcal{M}$ from outside and that $\bar{x}^2 + \bar{y}^2 > R^2$. Figure 2.10(a) shows that $-\pi < \psi_1^{d,ss} - \psi_1 < 0$, $-\pi < \psi_2^{d,ss} - \psi_2 < 0$ for all time before the trajectories approach $\mathcal{M}$ and $\sin(\psi_1^{d,ss} - \psi_1) < 0, \sin(\psi_2^{d,ss} - \psi_2) < 0$. Hence, $\dot{V}_1 < 0$.

(b) Two vehicles start facing away from each other: As ψ_{1_0}, ψ_{2_0} face away from each other (or $\psi_{1_0} \in (\frac{\pi}{2}, \frac{3\pi}{2}), \psi_{2_0} \in (-\frac{\pi}{2}, \frac{\pi}{2})$), both vehicles converge to $\mathcal{M}$ from inside and that $\bar{x}^2 + \bar{y}^2 < R^2$. Figure 2.10(b) shows that $0 < \psi_1^{d,ss} - \psi_1 < \pi$, $0 < \psi_2^{d,ss} - \psi_2 < \pi$ for all time before the trajectories approach $\mathcal{M}$ and $\sin(\psi_1^{d,ss} - \psi_1) > 0$, $\sin(\psi_2^{d,ss} - \psi_2) > 0$. Hence, $\dot{V}_1 < 0$.

(c) Two vehicles start with a similar direction: without loss of generality, we consider the case when $\psi_{1_0} \in (-\frac{\pi}{2}, \frac{\pi}{2}), \psi_{2_0} \in (-\frac{\pi}{2}, \frac{\pi}{2})$ (Figure 2.10(c)) and that $\sin(\psi_1^{d,ss} - \psi_1) < 0$ but $\sin(\psi_2^{d,ss} - \psi_2) > 0$. We notice that if $|\sin(\psi_1^{d,ss} - \psi_1)| > |\sin(\psi_2^{d,ss} - \psi_2)|$, it requires more control effort for aircraft one to converge to $\mathcal{M}$ and the trajectory in $\bar{x}\bar{y}$-plane converges to $\mathcal{M}$ from outside and that $\bar{x}^2 + \bar{y}^2 > R^2$. Hence, $\dot{V}_1 < 0$. The case when $|\sin(\psi_1^{d,ss} - \psi_1)| < |\sin(\psi_2^{d,ss} - \psi_2)|$, the trajectory in $\bar{x}\bar{y}$-plane converges to $\mathcal{M}$ from inside. Since $\bar{x}^2 + \bar{y}^2 < R^2$ in this case, the inequality $\dot{V}_1 < 0$ also holds.

It is noted that in practice, swirling effects are not applied for cases (b) and (c), since the vehicles will move farther from r_{ss} from each other and the safety angle will be $\frac{\pi}{4}$ off from the corresponding collision cone.

We now observe that

$$\begin{aligned}\dot{V}_2 &= 2(\psi_1^{d,ss}-\psi_1)(\dot{\psi}_1^{d,ss}-\dot{\psi}_1)+2(\psi_2^{d,ss}-\psi_2)(\dot{\psi}_2^{d,ss}-\dot{\psi}_2)\\ &= 2(\psi_1^{d,ss}-\psi_1)(\dot{\psi}_1^{d,ss}-K(\psi_1^{d,ss}-\psi_1)-\dot{\psi}_1^{d,ss})+2(\psi_2^{d,ss}-\psi_2)(\dot{\psi}_2^{d,ss}-K(\psi_2^{d,ss}-\psi_2)-\dot{\psi}_2^{d,ss})\\ &= -2K[(\psi_1^{d,ss}-\psi_1)^2+(\psi_2^{d,ss}-\psi_2)^2]<0,\end{aligned}$$

which translates to having $\dot{V}<0$ anywhere outside $\mathcal{M}$. Next, let $\beta>0$ and define

$$\mathcal{D}_c=\{(\bar{x},\bar{y},\psi_1,\psi_2)\in\mathbf{R}^4; V(\bar{x},\bar{y},\psi_1,\psi_2)\leq\beta\}.$$

We note that $\dot{V}\leq 0$ everywhere in $\mathcal{D}_c$. Next, defining $\mathcal{R}=\{(\bar{x},\bar{y},\psi_1,\psi_2)\in\mathbf{R}^4;\dot{V}(\bar{x},\bar{y},\psi_1,\psi_2)=0\}$; it thus follows that the largest invariant set in $\mathcal{R}$ is $\mathcal{M}$. Hence, the system of two vehicles, under swirling effect in (2.22), with $(\bar{x}(0),\bar{y}(0),\psi_1(0),\psi_2(0))=(-r_{ss},0,\psi_{1_0},\psi_{2_0})\subseteq\mathcal{D}_c$, one has $(\bar{x}(t),\bar{y}(t),\psi_1(t),\psi_2(t))\to\mathcal{M}$ as $t\to\infty$. ■

Theorem 2.3.2 shows that two vehicles converge to a limit cycle assuming that the gradient from the goal and path attraction is negligible as compared with that of the swirling effect. The effect of the neglected terms are analyzed in Section 2.3.3. Moreover, when the swirling effect reduction method is applied, both UAVs will converge and stay on the limit cycle only until they point away from each other, at which point they will continue on their original paths. Our result formalizes the fact the swirling effect keeps the vehicles apart by the diameter of the limit cycle $\mathcal{M}$.

It is also interesting to see the relationship between designed parameters and the minimum distance between two aircraft. In fact, the distance between the two vehicles in (2.22) is $d(t)=\sqrt{\bar{x}(t)^2+\bar{y}(t)^2}$ and its time derivative can be found as

$$\begin{aligned}\dot{d}(t) &= \frac{1}{d(t)}(\bar{x}(t)\dot{\bar{x}}(t)+\bar{y}(t)\dot{\bar{y}}(t))\\ &= U\left(\frac{\bar{x}(t)(\cos\psi_1(t)-\cos\psi_2(t))+\bar{y}(t)(\sin\psi_1(t)-\sin\psi_2(t))}{\sqrt{\bar{x}(t)^2+\bar{y}(t)^2}}\right)\\ &= U(\sin(\psi_1^{d,ss}(t)-\psi_1(t))+\sin(\psi_2^{d,ss}(t)-\psi_2(t))).\end{aligned}$$

For $i = 1, 2$, let $\alpha_i(t) = \psi_i^{d,ss}(t) - \psi_i(t)$; then

$$\begin{aligned} \dot{\alpha}_i(t) &= \dot{\psi}_i^{d,ss}(t) - [K(\psi_i^{d,ss}(t) - \psi_i(t)) + \dot{\psi}_i^{d,ss}(t)] \\ &= -K\alpha_i(t). \end{aligned} \tag{2.24}$$

The distance between two vehicles under swirling effect in (2.22) is thereby governed by

$$\begin{aligned} \dot{d}(t) &= U[\sin\alpha_1(t) + \sin\alpha_2(t)], \quad && d(0) = r_{ss} \\ \dot{\alpha}_1(t) &= -K\alpha_1(t), && \alpha_1(0) = \alpha_{1_0} \in [-\tfrac{\pi}{2} - \theta_{\text{cutoff}}, 0] \\ \dot{\alpha}_2(t) &= -K\alpha_2(t), && \alpha_2(0) = \alpha_{2_0} \in [-\tfrac{\pi}{2} - \theta_{\text{cutoff}}, 0] \end{aligned} \tag{2.25}$$

where the initial headings $\alpha_{1_0}, \alpha_{2_0} \in [-\frac{\pi}{2} - \theta_{\text{cutoff}}, 0]$ are only for case (a). The minimum distance between two vehicles can then be derived by solving (2.25).

Theorem 2.3.3 *The system of two vehicles under swirling effects (2.22) has the minimum distance between them as*

$$d_{min} = r_{ss} - \frac{U}{K}[\mathbf{Si}(|\alpha_{1_0}|) + \mathbf{Si}(|\alpha_{2_0}|)],$$

where $\mathbf{Si}(x)$ *is a sine integral function defined as [23];*

$$\int_0^x \frac{\sin t}{t}\, dt.$$

Proof. From (2.25), we have $\alpha_1(t) = \alpha_{1_0} e^{-Kt}$ and $\alpha_2(t) = \alpha_{2_0} e^{-Kt}$ and the distance between two vehicles is thus

$$\begin{aligned} d(t) &= d(0) + \int_0^t U[\sin(\alpha_{1_0} e^{-Kt}) + \sin(\alpha_{2_0} e^{-Kt})]\, dt \\ &= r_{ss} - \frac{U}{K}\left[\int_{\alpha_{1_0}}^{\alpha_{1_0} e^{-Kt}} \frac{\sin\alpha_1}{\alpha_1}\, d\alpha_1 + \int_{\alpha_{2_0}}^{\alpha_{2_0} e^{-Kt}} \frac{\sin\alpha_2}{\alpha_2}\, d\alpha_2\right] \\ &= r_{ss} - \frac{U}{K}[\mathbf{Si}(|\alpha_{1_0}|) - \mathbf{Si}(|\alpha_{1_0}| e^{-Kt}) + \mathbf{Si}(|\alpha_{2_0}|) - \mathbf{Si}(|\alpha_{2_0}| e^{-Kt})]. \end{aligned}$$

The minimum distance is hence found from

$$d_{\min} = \lim_{t\to\infty} d(t) = r_{ss} - \frac{U}{K}[\mathbf{Si}(|\alpha_{1_0}|) + \mathbf{Si}(|\alpha_{2_0}|)]. \tag{2.26}$$

The worst initial headings that lead to the shortest minimum distance occur when the **Si**-function reaches its maximums at $\alpha_{1_0} = \alpha_{2_0} = -\pi$ (This is not for the case when a "safety angle" has been used). The minimum distance therefore becomes $d_{\min} = r_{ss} - \frac{2U}{K}[\mathbf{Si}(\pi)] \approx r_{ss} - \frac{3.704U}{K}$. ■

The formula (2.26) shows the relationship between aircraft cruising speed U, control gain K, the effective range of detection r_{ss}, initial headings, as well as the minimum distance between the two vehicles. In tuning of the parameters for the deconfliction algorithm, we thus aim to have the minimum distance be greater than the radius of vehicles' protected zones.

For the case when $U_{h_1} \neq U_{h_2}$, the UAVs converge to two different limit cycles that share the same center but different in radius and stay on the opposite side of the center heading in opposite direction. The limit cycle can be written similar to (2.23) in the space of $(\bar{x}, \bar{y}, \psi_1, \psi_2)$. The minimum distance can also be found from $d_{\min} = r_{ss} - \frac{U_{h_1}}{K}\mathbf{Si}(|\alpha_{1_0}|) - \frac{U_{h_2}}{K}\mathbf{Si}(|\alpha_{2_0}|)$.

2.3.3 Guaranteed Deconfliction for Combined Navigation and Swirling Functions

In this section, we explore the minimum distance between a pair of conflicting UAVs when the navigation-based trajectory generation, swirling effect, and swirling effect reduction, are combined. In this venue, we continue to adopt Assumption 2.3.1 and Assumption 2.3.2 but drop Assumption 2.3.3 to see the effects of the path planning gradient on the minimum distance between the vehicles. First, we restate (2.17) in the polar coordinates as

$$\|\nabla V_i^{\text{new}}\| \angle \psi_i^d = 1 \angle \psi_i^{d,\text{path}} + K_s \beta \left(\frac{r_{ss} - \|\mathbf{q}_i - \mathbf{q}_j\|}{r_{ss}} \right) \angle \psi_i^{d,ss}, \tag{2.27}$$

where $\psi_i^d, \psi_i^{d,\text{path}}, \psi_i^{d,ss}$ are the desired heading, the desired heading from the path planning, and the desired heading from the swirling effect, respectively. The following assumption is introduced in order to streamline the subsequent proof.

Assumption 2.3.4 *At a location far from the destination, $\psi_i^{d,path}$ is uniform and pointed toward the goal. Rather than neglecting this term, we invoke that during the avoidance maneuver, the path*

planning gradient points in the same direction as when the vehicle enters the swirling zone, i.e., that $\psi_i^{d,path} \approx \psi_{i_0}$.

Since $\psi_i^d \neq \psi_i^{d,ss}$, for $i = 1, 2$, we define $\hat{\alpha}_i(t) = \psi_i^d(t) - \psi_i(t)$ and we have that $\dot{\hat{\alpha}}_i(t) = -K\hat{\alpha}_i(t)$. When $t = 0$ at the beginning of the swirling zone where $\|\mathbf{q}_1 - \mathbf{q}_2\| = r_{ss}$, $\beta(0) = 0$, $\hat{\alpha}_1(0) = \hat{\alpha}_2(0) = 0$ which translates to having $\hat{\alpha}_1(t) = \hat{\alpha}_2(t) = 0$ for all $t > 0$. This shows that $\psi_i(t) = \psi_i^d(t)$ as long as $\psi_i^d(t)$ is smooth and the feedback gain K is large enough to track $\psi_i^d(t)$. The minimum distance between two UAVs can now be derived from

$$d_{\min} = d(0) + \int_0^{t_{\min}} U[\sin(\psi_1^{d,ss}(t) - \psi_1^d(t)) + \sin(\psi_2^{d,ss}(t) - \psi_2^d(t))]dt, \tag{2.28}$$

where $t_{\min}$ is the time when the distance between two vehicles reaches its minimum value. The following lemma describes the main feature of the function $\psi_i^{d,ss}(t) - \psi_i^d(t), i = 1, 2$.

Lemma 2.3.4 *For* $i = 1, 2$*, the function* $|\alpha_i(t)| = |\psi_i^{d,ss}(t) - \psi_i^d(t)|$ *is a decreasing function of time on the interval* $(0, t_{min}]$*, where* $\alpha_i(0) = \alpha_{i_0} \in [-\frac{\pi}{2} - \theta_{cutoff}, 0]$*,* $\alpha_i(t_{min}) = 0$ *and* $\frac{d}{dt}|\psi_i^{d,ss}(t) - \psi_i^d(t)|_{t=0} = 0$.

Proof. From (2.27), we have that $\psi_i^{d,\text{path}} = \psi_{i_0}$ and the term $\beta\left(\frac{r_{ss} - \|\mathbf{q}_i - \mathbf{q}_j\|}{r_{ss}}\right)$ grows away zero towards the value of one as the two vehicles come closer to each other, which monotonically decreases $|\alpha_i(t)|$ until the minimum distance between the two UAVs is reached. In addition to this, $\psi_i^{d,ss}$ also heads toward ψ_{i_0} as the swirling effect progresses until $|\alpha_i(t)| = |\psi_i^{d,ss}(t) - \psi_i^d(t)| = 0$ at $t = t_{\min}$. After reaching the minimum distance, the value of $\psi_i^{d,ss}$ grows on the opposite side from ψ_i^d to push the vehicle back to the nominal path heading ψ_{i_0}. The plot in Figure 2.11(a) shows $\psi_i^{d,ss}(t) - \psi_i^d(t), i = 1, 2$ for the head-on case. Furthermore, the desired heading can also be written of the form $\psi_i^d = \psi_{i_0} + \mathbf{atan_2}\left(K_s\beta\sin(\psi_i^{d,ss} - \psi_{i_0}), 1 + K_s\beta\cos(\psi_i^{d,ss} - \psi_{i_0})\right)$ and by letting $t = 0$, $\beta(0) = 0$, and $\beta'(0) = 0$, the time derivative of the desired heading becomes $\frac{d}{dt}\psi_i^d(t)|_{t=0} = \frac{d}{dt}\psi_i^{d,ss}(t)|_{t=0}$ or $\frac{d}{dt}|\psi_i^{d,ss}(t) - \psi_i^d(t)|_{t=0} = 0$. ■

We can now use the result from Lemma 2.3.4 to obtain a bound on the minimum distance in (2.28) when the trajectories of the UAV pair are generated via the combined navigation and swirling functions.

Theorem 2.3.5 *For a pair of UAVs (2.20) for which the desired headings* $\psi_i^d, i = 1, 2$ *are derived from navigation-based path-planning and swirling effect in (2.27), there exists a constant* $\bar{K}$*,* $0 < \bar{K} \leq K$*,*

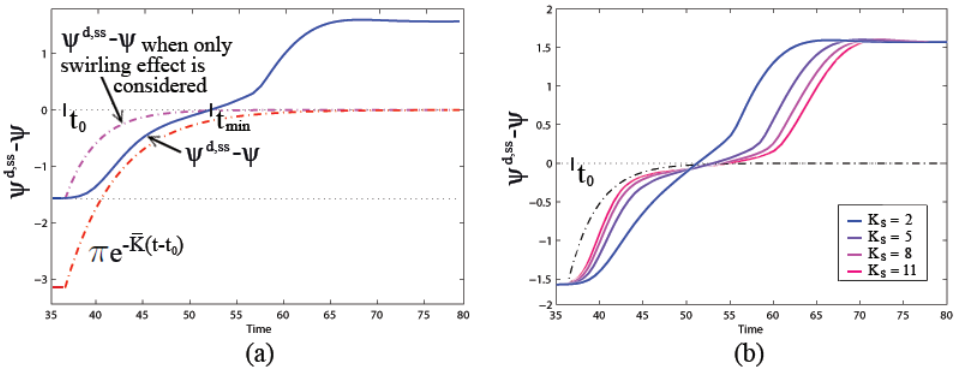


Figure 2.11: Discrepancy between aircraft heading and its desired heading from the swirling effect for the head-on case: (a) compared with the purely swirling case and the bound $-\pi e^{-\bar{K}(t-t_0)}$, (b) compare with different values of K_s.

such that the minimum distance between the UAVs is guaranteed to be bounded by $r_{ss} - \frac{2U}{\bar{K}}[\mathbf{Si}(\pi) - \mathbf{Si}(\pi e^{-\bar{K}t_{min}})]$ *during their entire trajectories.*

Proof. From Section 2.3.1, let us choose θ_{cutoff} to be in $[\frac{\pi}{4}, \frac{3\pi}{8}]$ which implies that $\alpha_i(0) \in [-\frac{7\pi}{8}, 0]$. Lemma 2.3.4 also shows that $|\alpha_i(t)|$ is a decreasing function on the time interval $(0, t_{\min}]$, $\alpha_i(t_{\min}) = 0$ and $\frac{d}{dt}|\alpha_i(t)|_{t=0} = 0$. Therefore, we can always find $\bar{K}$, $0 < \bar{K} \leq K$, such that $-\pi < -\pi e^{-\bar{K}t} < \alpha_i(t) < 0$ for all $t \in [0, t_{\min}]$. Furthermore, $\sin(e^{-\bar{K}t}) < 0$ for all $t \in [0, t_{\min}]$, and thus the minimum distance is

$$\begin{aligned} d_{\min} &= r_{ss} + \int_0^{t_{\min}} U[\sin(\alpha_1(t)) + \sin(\alpha_2(t))]\, dt \\ &> r_{ss} + \int_0^{t_{\min}} 2U[\sin(\pi e^{-\bar{K}t})]\, dt \\ &= r_{ss} - \frac{2U}{\bar{K}}[\mathbf{Si}(\pi) - \mathbf{Si}(\pi e^{-\bar{K}t_{\min}})]. \end{aligned} \tag{2.29}$$

■

For the design purpose, we would like determine how the swirling parameter K_s affects the value of $\bar{K}$ and $t_{\min}$ which would in turn, determine the minimum distance between the two vehicles. The plot in Figure 2.11(b) shows that the larger values for K_s leads the function $\alpha_i(t)$ to behave similar to the case of purely swirling in Theorem 2.3.3, that is $\alpha_i(t)$ behaves similar to $\alpha_{i_0}e^{-Kt}$ as K_s increases. This suggests that the more emphasis of swirling effect results in the bigger value of $\bar{K}$ and $t_{\min}$ and hence a larger bound on the minimum distance. However, the relationship between K_s and $\bar{K}$ is nonlinear and in practice, the value of K_s can be found by numerically integrating (2.28).

The selection of the parameters in the combined navigation and swirling functions-based trajectory generation algorithm can now be presented as follows:

1. Given the required minimum distance $d_{\min}$, the cruising speed U, the control gain K, and the initial heading $\alpha_{1_0}, \alpha_{2_0}$, for the pair of UAVs, one can use (2.26) to approximate r_{ss} or how early the swirling effect should take place. Since (2.26) is made under Assumption 2.3.3 and the actual trajectory "rotates" slower than the one generated by the purely swirling effects base on K_s, one can embed a safety factor into (2.26) by substituting K with $\bar{K}$, where $0 < \bar{K} \leq K$. For smooth trajectories, one can choose $\bar{K} = 0.7K$.

2. Based on the chosen value of r_{ss}, the on-board computer for the UAVs can determine the value of K_s by first setting it comparatively high, (e.g., $K_s = 10$) and integrating (2.28) numerically from $t = 0$ to $t_{\min}$, where $\psi_i^{d,ss}(t) - \psi_i^d(t) = 0$, determining the value $d_{\min}$. This process can then be iterated up with reduced values of K_s until the resulting value for $d_{\min}$ reaches the radius of the protected zones for the UAVs.

When the proposed deconfliction algorithm is applied to n vehicle scenarios with $n > 2$, in most cases, the intersection of regions of detection occurs between a pair of vehicles at a time and the conflict is resolved before detecting another vehicle. Hence the swirling effect is only applied to the closest UAV and (2.29) can guarantee the minimum distance between the two deconflicting vehicles. For the case when other vehicles have overlapping detection zones with a UAV at exactly the same instance, or before the swirling effect between two vehicles vanishes, the superposition of the effects from each vehicles can be used with bump function in order to emphasize the effect from the closer vehicles as shown in (2.18). The construction of Lyapunov-like function to prove the convergence for this more general case turns out to be more involved as the number of variables increases and the corresponding limit cycles also vary with the different conflicting scenarios. For example, when n vehicles symmetrically come into conflict from $\frac{2\pi}{n}$ degrees apart and (2.18) is applied (along with Assumption 2.3.3), the vehicles will converge to a circular limit cycle and stay equidistant around the circle with $\frac{2\pi}{n}$ phase different, and move counterclockwise until each of them reaches its nominal path. For the case when n vehicles come into conflict but not in a symmetrical way, the limit cycle is no longer unique and each vehicle will have its own periodic limit cycle. However, the theoretical guarantees for the general n unicycle-type UAVs with turn-rate constraints is not established in the present work and is the subject of future work.

2.4 UAV Deconfliction under Turn-Rates Constraints

For better tracking performance, the algorithm should generate commands that are within the aircraft acceleration constraints (3.2). In order to limit the turn-rate commands during the deconfliction maneuver, we proceed to examine the spatial derivatives of the gradient vector across the free configuration space. Since the magnitude of the gradient is not used in our approach, we only consider the change of its direction. The goal is to create a navigation function and swirling effect with derivatives that fall within the turn rate limit.

Intuitively the gradient should change with a lower rate when the navigation function gets smoother and less steep around the moving obstacle. One way to achieve this is to increase the protected zone radius as shown in Figure 2.12. However, the swirling radius is modeled based on how early the other vehicle can be detected. Once the vehicle enters other vehicle's protected zone, the turn-rate increases dramatically, as there is a component of the gradient pointed out directly from the moving obstacle. This observation, points out that expanding the protected zone is not the judicious way to reduce the vehicle's turn rate. In turn, the vehicle should avoid entering this zone and we assume that the vehicle always stay within the region $\mathcal{Q} = \{\mathbf{q} \,|\, \delta_j < \|\mathbf{q} - \mathbf{q}_j\| < r_{ss}\}$, when it rotates around the moving obstacle. Subsequently in this section, a mechanism for controlling the vehicle to stay within this region based on speed, control gain, swirling parameters, and conflicting scenarios, is discussed. In the following, we examine the design parameters that influence the UAV turn rates during the deconfliction maneuver.

2.4.1 Tuning Parameter k

As mentioned in the previous section, the parameter k scales the emphasis between obstacle avoidance and goal attraction. Figure 2.4 shows that the potential level can swing up and down based on different values of k. However, as shown below, the value of k does not affect the aircraft turn rate.

Proposition 2.4.1 *The tuning parameter k of the potential function in (2.6) does not change the direction of its gradient if the vehicle stays outside other vehicle's protected zone.*

Proof. From (2.13), the gradient of the navigation function is

$$\nabla_{\mathbf{q}_i} V_i(\mathbf{q}) = \frac{k\beta_i(\mathbf{q})\nabla_{\mathbf{q}_i}\Gamma_i(\mathbf{q}) - \Gamma_i(\mathbf{q})\nabla_{\mathbf{q}_i}\beta_i(\mathbf{q})}{k(\Gamma_i(\mathbf{q})^k + \beta_i(\mathbf{q}))^{1/k+1}}.$$

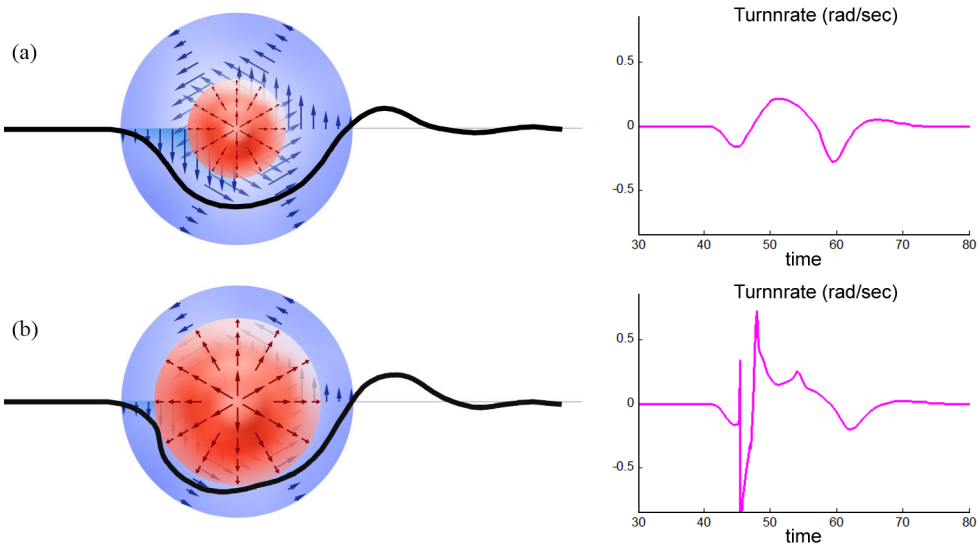


Figure 2.12: Avoidance trajectories: (a) Avoid the moving obstacle without entering protected zone; (b) Turn rate increases dramatically when the UAV enters other vehicle's protected zone.

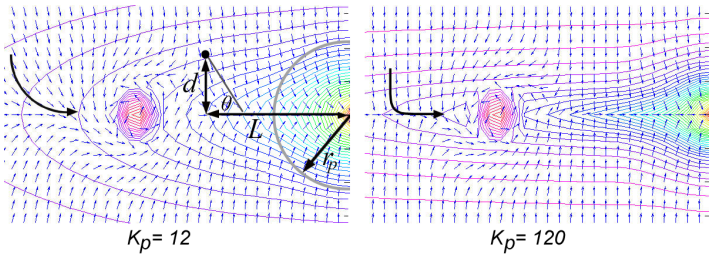


Figure 2.13: The effect on turn rate as the path attraction variable increases.

While the aircraft stays outside other vehicles' protected zones, $\beta(\mathbf{q})$ stays constant at one and $\nabla_{q_i}\beta_i = \mathbf{0}$. Thus the gradient becomes $\nabla_{\mathbf{q}_i} V_i(\mathbf{q}) = (\Gamma_i^k + 1)^{-\frac{k+1}{k}} \nabla_{\mathbf{q}_i}\Gamma(\mathbf{q})$ and (2.7) shows that $\nabla_{\mathbf{q}_i}\Gamma(\mathbf{q})$ is not a function of k. Hence, k can only change the magnitude, but not the direction, of the gradient, and hence, the turn-rate is independent of the value of k. ■

2.4.2 The Path Attraction Parameters K_p and r_p

Increasing the parameter K_p provides a scale for emphasizing the path attraction and higher turn rate when the vehicle swings back to the nominal path to head toward its goal. Figure 2.13 demonstrates how the prescribed trajectory changes with different values of K_p. This effect is dimmed down when the vehicle gets closer to the goal within the distance r_p, thus the turn rate varies conversely with r_p.

These parameters have more pronounced effect on the vehicle trajectory when the aircraft turns into its nominal path and not when they turn to avoid other vehicles. The following proposition provides the analytic formula that assists in choosing K_p and r_p such that the path attraction turn rate does not exceed the UAV dynamic constraints.

Proposition 2.4.2 *When an aircraft (3.1) with a unity constant speed and the deconfliction algorithm (2.5), (2.6), (2.7), and (2.8), turn to the planned-path toward the goal point, the path attraction turn rate is bounded by* $\frac{K_p\sqrt{2}}{3\sqrt{3}r_p}$.

Proof. We first note that when the vehicle stays off its nominal path by d at the distance $L \geq r_p$ from the goal, the term $\beta_i\left(\|\mathbf{q}_i - \mathbf{q}_{d_i}\|/r_p\right) = 1$ and (2.7) becomes

$$\Gamma = (L^2 + d^2) + K_p d^2 = L^2 + (1 + K_p)d^2.$$

Since we consider only the position outside other vehicle's protected zone, the function β in (2.6) remains constant at one. The angle θ between the gradient vector and the path is then

$$\theta = \tan^{-1}\left(\frac{\frac{\partial V}{\partial d}}{\frac{\partial V}{\partial L}}\right) = \tan^{-1}\left(\frac{\frac{\partial V}{\partial \Gamma}\frac{\partial \Gamma}{\partial d}}{\frac{\partial V}{\partial \Gamma}\frac{\partial \Gamma}{\partial L}}\right) = \tan^{-1}\left(\frac{(1+K_p)d}{L}\right);$$

we thus consider the path attraction turn rate along the line perpendicular to the path at distance L from the goal as a function of θ,

$$\begin{aligned}
\omega(\theta) &= \lim_{\Delta t \to 0}\left[\theta - \tan^{-1}\left(\frac{(1+K_p)[d - \Delta t \sin\theta]}{L - \Delta t \cos\theta}\right)\right]/\Delta t \\
&= \lim_{\Delta t \to 0}\left[\theta - \tan^{-1}\left(\frac{L\tan\theta - \Delta t(1+K_p)\sin\theta}{L - \Delta t\cos\theta}\right)\right]/\Delta t \\
&= \frac{K_p}{L}\sin\theta\cos^2\theta.
\end{aligned}$$

We now proceed to find the angle with maximum turn rate along the line by setting $\omega'(\theta) = 0$; hence we have $\theta = \tan^{-1}\frac{1}{\sqrt{2}}$. The maximum turn rate at distance L from the goal is then

$$\omega(\theta = \tan^{-1}(1/\sqrt{2})) = \frac{K_p\sqrt{2}}{3\sqrt{3}L}. \tag{2.30}$$

The closest to the goal that this formula applies is $L = r_p$, and thereby, the maximum path attraction turn rate is $\frac{K_p\sqrt{2}}{3\sqrt{3}r_p}$. We note that for the region within r_p-distance from the goal, the turn-rate is

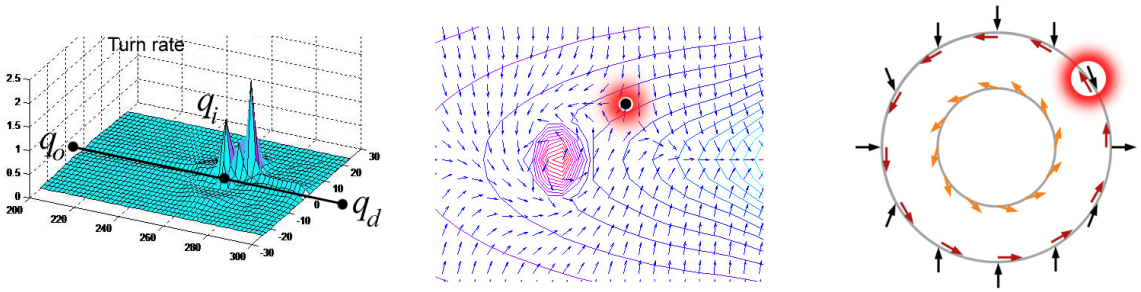


Figure 2.14: Without the swirling effect reduction: (a) The plot of turn-rate for the region around the moving obstacle, (b)-(c) The singular point with high turn-rate at the boundary of swirling radius.

scaled down to $\beta(\frac{\sqrt{(L^2+d^2)}}{r_p})\frac{K_p\sqrt{2}}{3\sqrt{3}L}$. ■

2.4.3 The Swirling Parameters K_s and r_{ss}

By exploring the adjusted gradient (2.17) around the moving obstacle, we find a singular point at the boundary of the swirling radius behind the obstacle where the turn-rate goes unbounded; see Figure 2.14. This means that by adding swirling effect to the navigation function, the property of being a $\mathcal{C}^2$-function has been compromised. This is due to a sudden change of the gradient to the opposite direction caused by the swirling term. However, we can avoid this region by swirling effect reduction described in Section 2.3.1. Meanwhile, the parameters K_s and r_{ss} also effect the turn-rates inside the swirling zone. As we increase the swirling radius r_{ss}, this effect grows at a slower rate, starting from the farther distance to the obstacle, resulting in a lower turn rate for the UAV. In the contrary, if we increase the relative magnitude of the swirling effect K_s, the relation (2.17) shows that the effect is more emphasized and the aircraft will turn at a higher rate.

Proposition 2.4.3 *When a vehicle heads directly toward the moving obstacle, and enters the swirling zone with unit velocity, the turn rate is bounded by* $\frac{mK_s}{r_{ss}}$*, where* m *denotes the maximum slope of the bump function (2.12).*

Proof. If a vehicle heads directly toward the moving obstacle, swirling vectors will be added perpendicular to the original gradient direction as shown in Figure 2.15. This modification has an effective radius of r_{ss} and maximum relative size of K_s with respect to the norm of the original gradient at the obstacle point and dims to zero at the distance r_{ss}, analogous to the bump function (2.12).

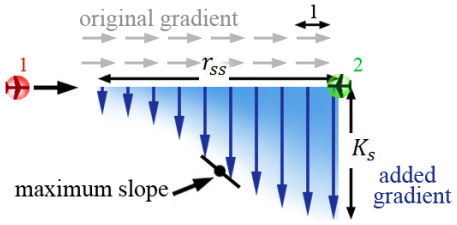


Figure 2.15: Swirling parameters.

Since (2.12) is normalized to unity, the change in this augmented swirling term is scaled by K_s and $1/r_{ss}$. Thus, the adjusted gradient has the rate bounded by mK_s/r_{ss} and the turn rate satisfies $\omega < \lim_{\Delta t \to 0} \left(\frac{\tan^{-1}\left(\frac{mK_s \Delta t}{r_{ss}}\right)}{\Delta t} \right) = \frac{mK_s}{r_{ss}}$. ■

The bound given in Proposition 2.4.3 can not be applied when the vehicle's heading is not directly toward the moving obstacle. However, the plot in Figure 2.14(c) suggests that the angles between the original gradients and the added swirling gradients do not deviate much from 90^o except in the region near the singular point.

2.4.4 Control Gains K_{ψ_i} and K_{γ_i}

As we have shown $\psi_i(t) - \psi_i^d(t) \to 0$ as $t \to \infty$, which translates to having $\psi_i(t)$ track $\psi_i^d(t)$ well as long as K_{ψ_i} and K_{γ_i} are large enough. Moreover $\psi_i^d(t)$ does not undergo a sudden change and

$$\begin{aligned} -\dot{\psi}_{\max} \leq K_{\psi_i}(\psi_i^d(t) - \psi_i(t)) + \dot{\psi}_i^d(t) \leq \dot{\psi}_{\max}, \\ -\dot{\gamma}_{\max} \leq K_{\gamma_i}(\gamma_i^d(t) - \gamma_i(t)) + \dot{\gamma}_i^d(t) \leq \dot{\gamma}_{\max}. \end{aligned} \tag{2.31}$$

The feedback term $K_{\psi_i}(\psi_i^d - \psi_i)$ and $K_{\gamma_i}(\gamma_i^d - \gamma_i)$ are important only at the beginning when the initial states are set different from the paths or when there are sudden changes to the way-points. The gain K_{ψ_i} and K_{γ_i} should be set according to (2.31). However, setting these gains too low may lead to the slower responses to the small tracking error. Therefore, we choose the control gains such that the turn-rate limits are reached when the error is around 15^o and impose the saturation to the control commands when the error is larger in the beginning.

Propositions 2.4.1, 2.4.2, and 2.4.3, along with (2.31) can help landscape the navigation function and adjust swirling effects such that the desired heading command stays within the aircraft turn-rate limits. Proposition 2.2.3 suggests that the parameter k should be chosen large enough such that

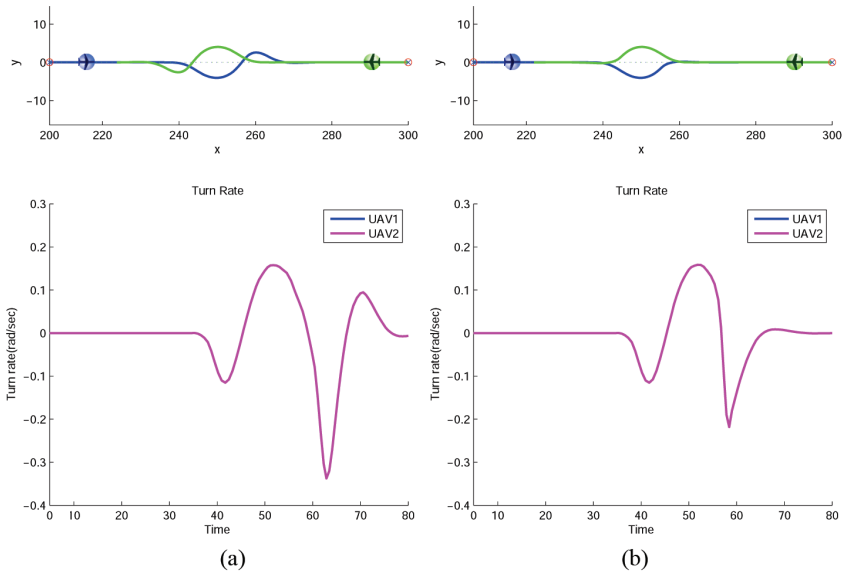
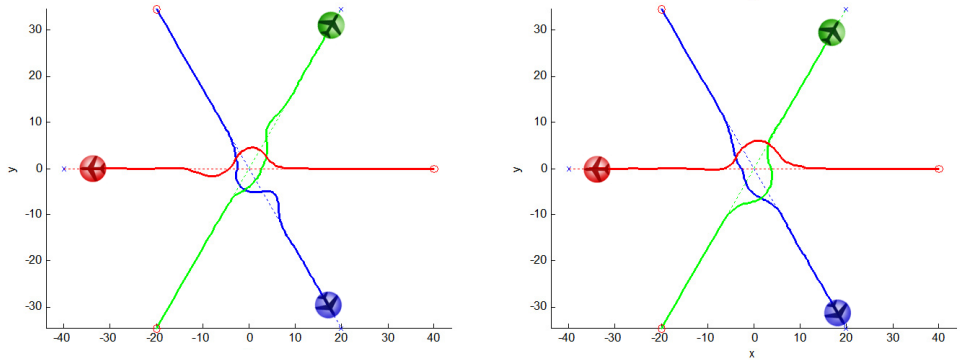


Figure 2.16: The head-on case with $r_{ss} = 30$: (a) Trajectory and turn rate as swirling effect creates overshoots. (b) The overshoot disappears after swirling effect is dimmed down by safety angle

the local minima are pushed toward the obstacle. Since the local minima stay inside other vehicle protected zone while the vehicle maneuvers outside other vehicles' protected zone, we allow the tuning parameter k to be chosen freely since it does not affect the turn rate of the vehicle. For simplification, we choose k such that a the potential value at a state far away from the boundary and the target has a value of 0.5 (according to Figure 2.4, $k = 0.2$). The parameters K_p, r_p, K_s, and r_{ss}, are then chosen based on Propositions 2.4.2 and 2.4.3.

2.5 Simulation Results

Figure 2.16 compares the simulation results for the head-on case. The turn-rate limit is set at 0.5 rad/sec. The parameters are chosen based on formulas presented in Section 2.4 in order to meet the turn-rate constraints and guarantee the minimum distance between vehicles: $k = 0.2, r_p = 20, K_p = 20, K_s = 2$, and $r_{ss} = 30$. Other parameters are set as follow: $\delta_j = \delta_c = 5, \rho_j = 0, \rho_c = 2000$ and the control gain $K_{\psi_i} = K_{\gamma_i} = 0.6$. The bump function (2.12) is chosen to be the tenth-order polynomial $f(r_{ij}) = 94.26r_{ij}^{10} - 679.49r_{ij}^{9} + 1936.96r_{ij}^{8} - 2926.49r_{ij}^{7} + 2594.56r_{ij}^{6} - 1383.19r_{ij}^{5} + 422.61r_{ij}^{4} - 63r_{ij}^{3}$ where $f'(0) = f''(0) = 0$ for the potential to be twice differentiable. After we reduce the swirling effect according to Figure 2.8(b), the trajectories no longer have overshoots, the turn rates are down to



(a) (b)

Figure 2.17: Simulation trajectories for the case where 3 vehicles are 120 degree apart: (a) $r_{ss} = 20$, (b) $r_{ss} = 30$ with swirling reduction concept applied.

around 0.2 rad/sec, and the aircraft return to the nominal paths almost 10 seconds faster. Figure 2.17 depicts the application of this approach on the symmetric case of 3 vehicles from 120 degree apart that fly through the same point. First, the swirling effect is applied constantly at $r_{ss} = 20$ to the closest vehicle detected. Then, the parameter r_{ss} is raised to 30 in order to reduce the turn rate and gradually decrease r_{ss} to zero once the conflict is resolved. Figure 2.18 shows a 3D example with 5 UAVs flying through the same coordinates. Figure 2.19 depicts yet another example of the deconfliction algorithm with two groups of vehicles that have intersecting trajectories.

2.6 Remarks

In this chapter, we have developed a deconfliction algorithm that guarantees collision avoidance between a pair of constant speed unicycle-type UAVs as well as convergence to the desired destination for each UAV in presence of static obstacles. In addition, we were able to ensure that the UAVs stay close to their nominal paths, and satisfy maximum turn-rate limitations for each UAV throughout the entire mission. The performance of the algorithm has then been validated with respect to key algorithmic parameters and the mission specific restrictions such as turn rate limits and constant UAV velocities. In this avenue, the proposed navigation function has been judiciously landscaped by adjusting the swirling effect such that the vehicles operate within their turn rate limitations. Finally, the collision avoidance analysis has been performed both for the scenario where the swirling effect is dominant, as well as for the case when it is combined with the navigation function. The simula-

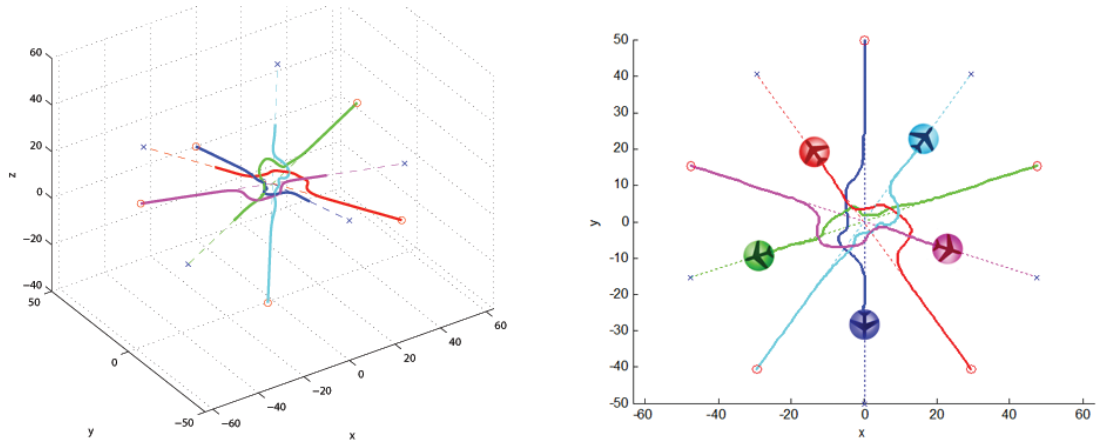


Figure 2.18: Simulation trajectories in 3D for the case where 5 vehicles are 72 degree apart with different altitudes and fly through the same point.

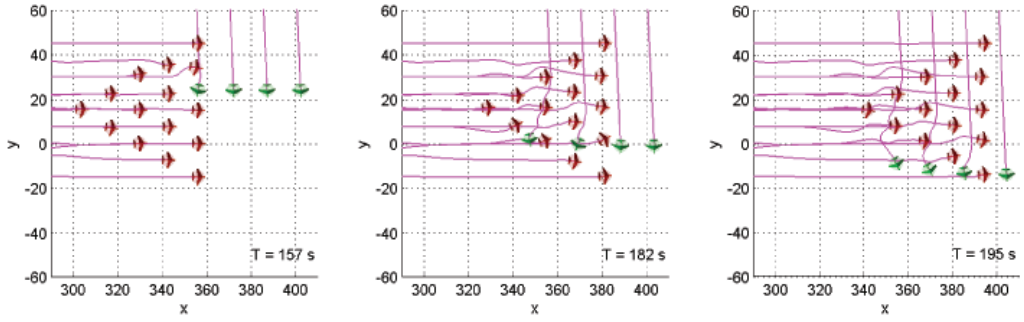


Figure 2.19: Deconfliction trajectories for two group of vehicles that have cross paths.

tion examples demonstrate that the deconfliction algorithm guarantees collision avoidance between unicycle-type UAVs as well as convergence to the desired destination.

There are several possible extensions of this work. First, velocity information of the sensed obstacles can be considered in the potential function construction process in order to improve the collision avoiding performance. The second extension involves allowing UAV speed to vary between its stall and maximum speed, as well as making use of the magnitude of the gradients of the navigation function. Including higher fidelity models for UAVs that include drift or dynamic coupling term provides yet another venue for the extension of this work. Path-length calculation can also be included to adjust the speed of each UAV in order the minimize the effect of the deconfliction maneuver on the time

of arrival at a particular way point. Moreover, exploring the convergence of the algorithm when n vehicles come into the range of detection at the same time should be formalized. Finally, examining the robust performance of the proposed deconfliction algorithm is of prime importance for actual multiple UAV missions.

Chapter 3

Optimal Path Planning for Network Establishment for UAVs

In this chapter, we develop two approaches to establish a proximity network among a collection of unmanned aerial vehicles (UAVs) that are initially scattered in space. The goal is to find the shortest trajectories that bring the UAVs to a connected formation where they are in the range of detection of one another and heading in the same direction to maintain the connectivity. Similar to the work of Deconfliction in Chapter 2, constant-speed unicycles are chosen to represent the UAVs kinematics flying steady around their cruising speed. Pontryagin Minimum Principle (PMP) is utilized to determine the control law and path synthesis for the UAVs under the turn-rate constraints. We introduce an algorithm to search for the optimal solution when the final network topology is specified beforehand; followed by a nonlinear programming method in which the final configuration is emerged from the optimization routine under the constraints that the final topology is connected. We provide the discussion on the performance of both methods toward the end of the chapter.

The organization of this chapter is as follow. We begin by giving the preliminary background on graph theory and the UAVs model in Section3.1. The control law and path synthesis are developed in Section3.2. In this section, we propose the algorithm to search for the global optimal solution when a target "tree" graph connectivity structure has been specified. A method that requires long computational time when applying to more than a few UAVs. The second method that involve transforming the problem to nonlinear optimization is introduced in Section3.3. Simulation results

for both methods are illustrated in Section3.4. Conclusion and remarks are drawn in Section 3.5

3.1 Preliminary and Problem Formulation

The multi-vehicle systems of UAVs usually perform their missions at the same altitude; therefore, we consider a group of n homogeneous vehicles in 2D instead of 3D space represented by non-holonomic unicycle model

$$\begin{aligned} \dot{x}_i(t) &= V_i \cos\theta_i(t),\ i = 1, ..., n \\ \dot{y}_i(t) &= V_i \sin\theta_i(t), \\ \dot{\theta}_i(t) &= u_i(t), \end{aligned} \tag{3.1}$$

where (x_i, y_i) specifies the location of vehicle i measured relative to the earth frame. Every vehicle is assumed to have the same unit constant speed. The heading angle defines the orientation $\theta_i \in [-\pi, \pi]$ and its rate defines the input u_i. The constrainted turn-rates are imposed as

$$-\dot{\theta}_{max} \leq u_i(t) \leq \dot{\theta}_{max}; \tag{3.2}$$

without loss of generality, we assume that $\dot{\theta}_{max} = 1$. We use the graph $\mathcal{G} = (\mathcal{I}, \mathcal{E})$ to represent the connection topology between vehicles. $\mathcal{I} = \{1, ..., N\}$ is the set of vertex indices where each vertex represents a vehicle and $\mathcal{E} \subseteq \mathcal{I} \times \mathcal{I}$ is the set of edges. The connection among the vehicles in the undirected network $\mathcal{G}$ is expressed by the entries of the adjacency matrix $\mathcal{A}(\mathcal{G}) = [a_{ij}]$ with $a_{ij} = 1$ when $(i, j) \in \mathcal{E}$, and $a_{ij} = 0$ otherwise.

An edge between vehicle i and vehicle j indicates that vehicle i is in the sensing range D of vehicle j and vice versa, i.e., vehicle i is adjacent to vehicle j or

$$(i, j) \in \mathcal{E} \Leftrightarrow d_{ij} = \|(x_i, y_i) - (x_j, y_j)\| \leq D.$$

Vehicles i and j are said to be *connected* if $\mathcal{G}$ contains a path from i to j. A *group* of vehicles is *connected* if every pair of i and j in $\mathcal{G}$ are connected.

There are several matrix representations of a graph $\mathcal{G}$. The *degree matrix* $\Delta(\mathcal{G})$ is a diagonal matrix given by $[\Delta(\mathcal{G})]_{ii} = \sum_{j=1}^{n} a_{ij}$ and $[\Delta(\mathcal{G})]_{ij} = 0$ $(i \neq j)$. The *incidence matrix* $B(\mathcal{G}) = [b_{jl}] \in \mathbb{R}^{N \times |\mathcal{E}|}$

is the matrix where each column corresponds to an edge in $\mathcal{E}$. The column $l \in \{1, ..., |\mathcal{E}|\}$ corresponds to the edge $(i, j) \in \mathcal{E}$ and has entries $b_{jl} = 1 = -b_{il}$, where the remaining entries of the column are zero. The *Laplacian matrix* $\mathcal{L}(\mathcal{G})$ is a square matrix defined as $\mathcal{L}(\mathcal{G}) \triangleq \Delta(\mathcal{G}) - \mathcal{A}(\mathcal{G})$. We denote the eigenvalues of $\mathcal{L}(\mathcal{G})$ by $0 = \lambda_1(\mathcal{G}) \leq \lambda_2(\mathcal{G}) \leq ... \leq \lambda_n(\mathcal{G})$, where $\lambda_2(\mathcal{G}) > 0$ is a necessary and sufficient condition for connectivity of $\mathcal{G}$. A tree is a connected graph without cycles and a connected graph with the least number of edges. A tree of n vertices has $n - 1$ edges.

The goal is to control n unicycle vehicles that are initially out of sensing range to follow shortest trajectories under the turn-rate constraints to a final state where all the vehicles are connected and heading in the same direction. We refer to the graph representation at time t_f as $\mathcal{G} = \mathcal{G}(\mathbf{q}_f)$, where $\mathbf{q}_f$ is a short-hand notation for the collection of vehicle states at the final time. Since all UAVs flies at the same speed, every vehicle will travel the same path-length until it reaches the final location.

The problem can be decomposed into two parts: (1) searching for the optimal controls that bring the vehicles to a specified tree-graph connectivity structure, headed in the same direction, and (2) comparing the solutions among all possible target tree-graphs. The method to solve step (1) is presented in the following section. The "tree" structure is chosen as it is a connected graph with the least number of edges, or in our case, the least number of final constraints.

To simplify the problem, we assume that the closed path connecting initial locations of all vehicles forms a convex polygon as shown in Figure 3.1(a). This is to avoid having some vehicles initially lie in the convex hull of other vehicles.

3.2 Optimal Path Planning for Target Tree-Graph Connectivity

The time-optimal control problem that brings n unicycle vehicles (3.1) to a specified tree-graph connectivity structure and heading in the same direction, is formulated as follow:

$$\min_{u_1,...,u_n} \int_0^{t_f} 1 \, dt, \tag{3.3}$$

subject to

$$
\begin{aligned}
&\dot{x}_i(t) = \cos\theta_i(t),\ i = 1, ..., n && \text{system dynamics}\\
&\dot{y}_i(t) = \sin\theta_i(t),\\
&\dot{\theta}_i(t) = u_i(t),\\
&\\
&x_i(0) = x_{i0},\ i = 1, ..., n && \text{initial condition}\\
&y_i(0) = y_{i0},\\
&\theta_i(0) = \theta_{i0}, \forall (i,j), d_{ij}(0) > D\\
&\\
&\|u_i(t)\| \leq 1,\ i = 1, ..., n && \text{control constraint}\\
&\\
&(x_i(t_f) - x_j(t_f))^2 + (y_i(t_f) - y_j(t_f))^2 \leq D^2,\ \forall (i,j) \in \mathcal{T}, && \text{final constraint}\\
&\theta_i(t_f) - \theta_j(t_f) = 0,
\end{aligned}
$$

where $\mathcal{T}$ is an edge set that forms a tree, $\mathbf{q} = [\mathbf{q}_1, ..., \mathbf{q}_n]^T$, $\mathbf{q}_i = [x_i, y_i, \theta_i]^T$, $\mathbf{q}_f = \mathbf{q}(t_f)$, $\mathbf{f} = [\mathbf{f}_1, ..., \mathbf{f}_n]^T$, $\mathbf{f}_i = [\cos\theta_i, \sin\theta_i, u_i]^T$, and $\mathbf{u} = [u_1, ..., u_n]^T$.

The initial condition is chosen such that $\forall (i,j) \in \mathcal{I} \times \mathcal{I},\ \|(x_i(0), y_i(0)) - (x_j(0), y_j(0))\| > d$, where as The final constraints can be written as a smooth end-point manifold

$$M_f = \{\mathbf{q}_f : \Omega(\mathbf{q}_f) = 0\},$$

where $\Omega(\mathbf{q}_f) = [\Omega_1(\mathbf{q}_f), ..., \Omega_{n-1}(\mathbf{q}_f), \tilde{\Omega}_1(\mathbf{q}_f), ..., \tilde{\Omega}_{n-1}(\mathbf{q}_f)]^T$, $\Omega_l(\mathbf{q}_f) = (x_i(t_f) - x_j(t_f))^2 + (y_i(t_f) - y_j(t_f))^2 + s_l^2 - D^2$, s_l is a slack variable, and $\tilde{\Omega}_l(\mathbf{q}_f) = \theta_i(t_f) - \theta_j(t_f), l = 1, ..., n-1$.

The Pontryagin Minimum Principle (PMP) approach involves introducing of an adjoint variable vector $\lambda = [\lambda_1, ..., \lambda_n]^T$ where $\lambda_i = [\lambda_{xi}, \lambda_{yi}, \lambda_{\theta i}]^T$ and construct a Hamiltonian

$$H(\mathbf{q}, \lambda) = \min_{\mathbf{u}} \mathcal{H}(\mathbf{q}, \lambda, \mathbf{u}), \tag{3.4}$$

where $\mathcal{H}(\mathbf{q}, \lambda, \mathbf{u}) = 1 + \lambda^T \mathbf{f} = 1 + \sum_{i=1}^{n} \lambda_{xi} \cos\theta_i + \lambda_{yi} \sin\theta_i + \lambda_{\theta i} u_i$. The following proposition provides necessary conditions for optimality of the control law.

Proposition 3.2.1 *(PMP-Endpoint Manifolds) Suppose u^* transfers the system from an initial state to a state in the final manifold M_f with the minimum cost. Then u^* satisfies the following necessary conditions:*

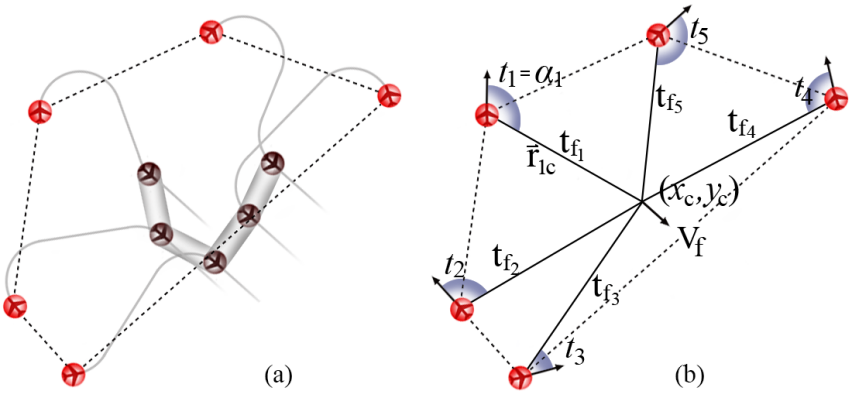


Figure 3.1: (a) A convex polygon (dot) formed by the closed path connecting vehicles initial points, the grey lines shows the optimal path that connect 5 vehicles to a Path-graph. (b) The centroid of the convex polygon is used to pick a control candidate along the with initial guesses for t_i, $\tilde{t}_i$, and t_f.

1. *There exists a nonzero costate* λ, *such that* $\dot{\lambda} = -\frac{\partial H}{\partial \boldsymbol{q}}$,

2. $\boldsymbol{u}^* = \arg\min \mathcal{H}(\boldsymbol{q}, \lambda, \boldsymbol{u})$, $\forall t \in [0, t_f]$,

3. $H(\boldsymbol{q}, \lambda) = 0$, $\forall t \in [0, t_f]$, *and*

4. $\lambda(t_f)$ *is orthogonal to the end-point manifold* M_f.

Proof. See [40]: Proposition 3.10. ■

3.2.1 Characterizing the Optimal Control

The following proposition presents the control law for problem (3.3) which is similar to that of Dubin's car [40].

Proposition 3.2.2 *The control law for (3.3) is of the form:*

$$u_i^* = \begin{cases} -1 & \lambda_{\theta i} < 0 \text{ (turn right maximum)}, \\ 0 & \lambda_{\theta i} = 0 \text{ (go straight)}, \\ 1 & \lambda_{\theta i} > 0 \text{ (turn left maximum)}. \end{cases} \tag{3.5}$$

Proof. From Proposition 3.2.1, the control law is derived from the pointwise minimizing argument of the Hamiltonian

$$\mathbf{u}^* = \arg \min_{\|u_i\| \leq 1} \mathcal{H}(\mathbf{q}, \lambda, \mathbf{u}) = \arg \min_{\|u_i\| \leq 1} \sum_{i=1}^{n} \lambda_{\theta i} u_i,$$

which states that the optimal control only depends on the sign of $\lambda_{\theta i}$, i.e., $u_i^* = \begin{cases} -1 & \lambda_{\theta i} < 0 \\ 1 & \lambda_{\theta i} > 0 \end{cases}$. However, if $\lambda_{\theta i}$ vanishes on a non-zero time-interval $\mathfrak{I}$, PMP can not provide information on the optimal control. For this singular case, we look at the first condition of Proposition 3.2.1 which requires

$$\begin{aligned} \dot{\lambda}_{xi} &= -\tfrac{\partial H}{\partial x_i} = 0, \\ \dot{\lambda}_{yi} &= -\tfrac{\partial H}{\partial y_i} = 0, \\ \dot{\lambda}_{\theta i} &= -\tfrac{\partial H}{\partial \theta_i} = \lambda_{xi} \sin\theta_i - \lambda_{yi} \cos\theta_i. \end{aligned} \tag{3.6}$$

This shows that both λ_{xi} and λ_{yi} are constants. Furthermore, the third condition of Proposition 3.2.1 demands that

$$H|_{\lambda_{\theta i}=0} = 1 + \sum_{i=1}^{n} \lambda_{xi} \cos\theta_i + \lambda_{yi} \sin\theta_i = 0, \ \forall t \in [0, t_f]. \tag{3.7}$$

Since λ is a non-zero vector, $[\lambda_{xi}, \lambda_{yi}]$ can not be zero and (3.7) can only hold if and only if $\theta_i(t)$ is constant or $u_i(t) = 0$ for all $t \in \mathfrak{I}$; this yields the control law (3.5). ■

3.2.2 Optimal Path Synthesis

Since the control of problem (3.3) comprises of having each vehicle turn left or right at the maximum rate or go straight, the problem reduces to finding the sequence of commands and the duration spent on each segment. The following proposition presents the control sequence for each vehicle in (3.3).

Proposition 3.2.3 *The control for each vehicle for the Optimal Target Tree problem (3.3) is of the*

form:

$$u_i^*(t) = \begin{cases} \pm 1 & 0 \le t < t_i, \\ 0 & t_i \le t \le t_f - \tilde{t}_i, \\ \pm 1 & t_f - \tilde{t}_i \le t \le t_f, \end{cases} \tag{3.8}$$

where t_i and $\tilde{t}_i$ are switching times. The trajectory during the intervals $0 \le t < t_i$, $t_i \le t \le t_f - \tilde{t}_i$, and $t_f - \tilde{t}_i \le t \le t_f$ are called first, second, and third segment for vehicle i, respectively.

Proof. The last condition of Proposition 3.2.1 requires $\lambda(t_f)$ to be orthogonal to M_f. In other words, $\lambda(t_f)$ in the space spanned by $\nabla_{\mathbf{q}}\Omega(\mathbf{q}_f)$. Since $\Omega_l(\mathbf{q}_f)$ and $\tilde{\Omega}_l(\mathbf{q}_f)$ are prescribed by the different pair of vehicles that form an edge of a tree-graph, they are linearly independent, and M_f possesses a unique tangent plane at every point $\mathbf{q}_f$ in M_f. The final condition for the adjoint variable becomes

$$\begin{aligned} \lambda(t_f) &= \sum_{l=1}^{n-1} \nu_l \nabla_{\mathbf{q}} \Omega_l(\mathbf{q}_f) + \tilde{\nu}_l \nabla_{\mathbf{q}} \tilde{\Omega}_l(\mathbf{q}_f) \\ &= \sum_{l=1}^{n-1} B(\mathcal{T})_l \otimes \left(\nu_l \begin{bmatrix} x_i(t_f) - x_j(t_f) \\ y_i(t_f) - y_j(t_f) \\ 0 \end{bmatrix} + \tilde{\nu}_l \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \right), \end{aligned} \tag{3.9}$$

where ν_l and $\tilde{\nu}_l$ are multipliers and $B(\mathcal{T})_l$ is the lth-column of the incidence matrix of the tree $\mathcal{T}$ representing the edge l, connecting vehicles i and j.

From (3.9), we have $\lambda_{\theta i}(t_f) = \sum_{l=1}^{n-1} B(\mathcal{T})_{il} \tilde{\nu}_l \neq 0$. This means that every vehicle needs to make the final turn to get to the prescribed relative distance and heading in the same direction, hence $u_i^*(t) = \pm 1$ when $\tilde{t}_i \le t \le t_f$.

Meanwhile, (3.6) indicates that $\lambda_{\theta i}(t)$ evolves sinusoidally when $u_i(t) = \pm 1$ and that $\lambda_{\theta i}(t = t_f - \tilde{t}_i) = 0$. Proposition 3.2.2 suggests that the control vanishes throughout the time interval $\mathfrak{I}$ that $\lambda_{\theta i}(t) = 0$. Therefore, $\exists t_i \le t_f - \tilde{t}_i$ such that $u_i(t) = 0$ when $t_i \le t \le t_f - \tilde{t}_i$ or every vehicle fly on a straight path before reaching the final turn. In order to connect the last two segments of the optimal trajectory with the initial headings and locations, it is transparent that the vehicle has to turn at its maximum rate in the first segment as $\lambda_{\theta i}(t)$ evolves sinusoidally toward zero when $t = t_i$. This thus implies that the control sequence is of the form (3.8). ■

Now, solving for the solution of (3.3) reduces to determining the choices of controls $u_i(t \leq t_i)$ and $u_i(t_f - \tilde{t}_i \leq t \leq t_f)$ (either 1 or -1) and the switching times t_i and $\tilde{t}_i$. For n vehicles, we have 2^{2n} candidates for the optimal solution: turn left or right on the first and third segments and for n vehicles; the method to select this will be presented in the next subsection. Meanwhile, the switching time t_i and $\tilde{t}_i$ are obtained by solving the following necessary conditions.

Proposition 3.2.4 *(Path Synthesis) The switching times $t_i, \tilde{t}_i, i = 1, ..., n$ in (3.8) and the final time t_f of the Optimal Target Tree problem (3.3) are determined by solving the following intermediate/final conditions:*

1. *$(2n-2)$ final constraints of the states: $\forall (i,j) \in \mathcal{T}$, $(x_i(t_f) - x_j(t_f))^2 + (y_i(t_f) - y_j(t_f))^2 + s_{ij}^2 - D^2 = 0$ and $\theta_i(t_f) - \theta_j(t_f) = 0$,*
2. *$(n-1)$ constraints of the slack variable: $\forall (i,j) \in \mathcal{T}$, $\nu_{ij}((x_i(t_f) - x_j(t_f))^2 + (y_i(t_f) - y_j(t_f))^2 - D^2) = 0$*
3. *n intermediate constraints of the adjoint variable: $\dot{\lambda}_{\theta i}(t_i) = \lambda_{xi} \sin\theta_i(t_i) - \lambda_{yi} \cos\theta_i(t_i) = 0$,*
4. *n final constraints of the adjoint variable: $\lambda_{\theta i}(t_f) = \sum_{l=1}^{n-1} B(\mathcal{T})_{il} \tilde{\nu}_l = \int_{t_f - \tilde{t}_i}^{t_f} \lambda_{xi} \sin\theta_i(t) - \lambda_{yi} \cos\theta_i(t)\, dt$,*
5. *a final condition of the Hamiltonian $H(\boldsymbol{q}(t_f), \lambda(t_f)) = 0$,*

where $\nu_{ij} \doteq \nu_l$ and $s_{ij} \doteq s_l$ are multiplier/slack variable for edge l between vehicle i and j, $l = 1, ..., n-1$.

Proof. Since the necessary conditions from Proposition 3.2.1 are used to determine the control law and the candidates of the control sequences in Proposition 3.2.2 and Proposition 3.2.3, respectively, we use the system dynamics along with the intermediate/final conditions of states and the adjoint variable to determine the switching times. Let u_i be a short-hand notation for $u_i(0 \leq t < t_i)$ and $\tilde{u}_i$ for $u_i(t_f - \tilde{t}_i \leq t \leq t_f)$. We now have $5n-2$ unknowns which include $t_i, \tilde{t}_i, (i = 1, ..., n)$, $\nu_l, \tilde{\nu}_l, s_l, (l = 1, ..., n-1)$ and t_f, hence $5n-2$ nonlinear equations are required to solve for the unknowns.

Starting with system dynamics, and integrating the dynamics through the switching times to the final time yields

$$
\begin{aligned}
x_i(t_f) &= x_{i0} + \tfrac{\sin(\theta_{i0} + u_i t_i) - \sin\theta_{i0}}{u_i} + \cos(\theta_{i0} + u_i t_i)(t_f - t_i - \tilde{t}_i) + \tfrac{\sin(\theta_{i0} + u_i t_i + \tilde{u}_i \tilde{t}_i) - \sin(\theta_{i0} + u_i t_i)}{\tilde{u}_i}, \\
y_i(t_f) &= y_{i0} - \tfrac{\cos(\theta_{i0} + u_i t_i) - \cos\theta_{i0}}{u_i} + \sin(\theta_{i0} + u_i t_i)(t_f - t_i - \tilde{t}_i) - \tfrac{\cos(\theta_{i0} + u_i t_i + \tilde{u}_i \tilde{t}_i) - \cos(\theta_{i0} + u_i t_i)}{\tilde{u}_i}, \\
\theta_i(t_f) &= \theta_{i0} + u_i t_i + \tilde{u}_i \tilde{t}_i, \ i = 1, ..., n.
\end{aligned}
$$

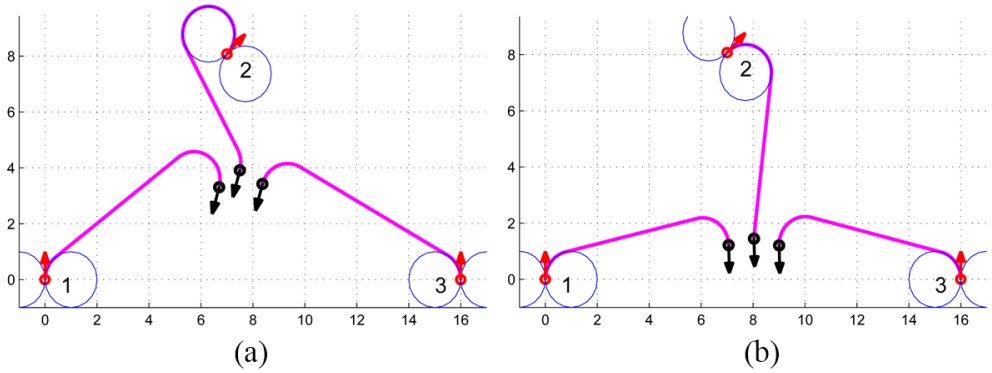


Figure 3.2: Optimal trajectories that brings 3 UAVs to a tree: (a) Candidate RLL-RRL, $t_f = 9.4$. (b) Candidate RRL-RLL (Global solution), $t_f = 8.3$.

(3.10)

Substituting these into the final constraints $(x_i(t_f)-x_j(t_f))^2+(y_i(t_f)-y_j(t_f))^2+s_{ij}^2-D^2 = 0$ and $\theta_i(t_f)-\theta_j(t_f)=0, \forall (i,j)\in\mathcal{T}$, gives us $2n-2$ nonlinear equations as a function of $t_1,...,t_n$, and t_f.

The condition for s_l is derived from $\nabla_{s_l}\Omega_l(\mathbf{q}_f)=\nu_l s_l = 0$. Multiplying the final constraint with ν_{ij} yields $\nu_{ij}((x_i(t_f)-x_j(t_f))^2+(y_i(t_f)-y_j(t_f))^2-D^2)=0$, another $n-1$ equation.

Next, the necessary conditions associated with $\lambda_{\theta i}$ are derived base on the second segment where $\lambda_{\theta i}=0 \rightarrow \dot{\lambda}_{\theta i}=0=\lambda_{xi}\sin\theta_i-\lambda_{yi}\cos\theta_i$. Substituting λ_{xi} and λ_{yi} with the value in (3.9) yields $\left[\sum_{j\in\mathcal{N}_i}\nu_{ij}(x_i(t_f)-x_j(t_f))\right]\sin\theta_i(t_i) = \left[\sum_{j\in\mathcal{N}_i}\nu_{ij}(y_i(t_f)-y_j(t_f))\right]\cos\theta_i(t_i)$, where $\mathcal{N}_i$ is a neighboring index set of vertex i of a tree defined as $\mathcal{N}_i=\{j:(i,j)\in\mathcal{T}\}$; this gives us another n equations.

Subsequently, the final constraints $\lambda_{\theta i}(t_f)$ can be derived by integrating $\dot{\lambda}_{\theta i}$ in (3.6) toward the final time: $\lambda_{\theta i}(t_f)=\sum_{l=1}^{n-1}B(\mathcal{T})_{il}\tilde{\nu}_l=\int_{t_f-\tilde{t}_i}^{t_f}\lambda_{xi}\sin\theta_i(t)-\lambda_{yi}\cos\theta_i(t)\,dt=\lambda_{xi}\left[\frac{\cos(\theta_{i0}+u_it_i)-\cos(\theta_{i0}+u_it_i+\tilde{u}_i\tilde{t}_i)}{\tilde{u}_i}\right]$ $+\lambda_{yi}\left[\frac{\sin(\theta_{i0}+u_it_i)-\sin(\theta_{i0}+u_it_i+\tilde{u}_i\tilde{t}_i)}{\tilde{u}_i}\right]$, $i=1,...,n$,

Finally, the last equation involves the Hamiltonian that vanishes along the optimal trajectory toward the final time. This can be expressed as $H|_{t_f}=0=1+\sum_{i=1}^{n}\lambda_{xi}(t_f)\cos\theta_i(t_f)+\lambda_{yi}(t_f)\sin\theta_i(t_f)$ $=1+\sum_{i=1}^{n}\left[\left(\sum_{j\in\mathcal{N}_i}\nu_{ij}(x_i(t_f)-x_j(t_f))\right)\cos\theta_i(t_f)+\left(\sum_{j\in\mathcal{N}_i}\nu_{ij}(y_i(t_f)-y_j(t_f))\right)\sin\theta_i(t_f)\right]$. These give us $5n-2$ nonlinear equations to solve for $5n-2$ unknowns which can be done using available numerical tools. ■

3.2.3 Proposed Algorithm for Finding the Optimal Path

It is important to choose the control candidate corresponding to the global optimal solution that yields the shortest path. For n vehicles, we have 2^{2n} candidates where each vehicle has a choice to turn right or left in the first and third segments. Taking a wrong turn results in a longer path corresponding to a local optimal solution. Figure 3.2(a) shows the optimal trajectory candidate RLL-RRL (a short hand notation for vehicles 1, 2, and 3 turn right, left, and left during the time intervals $0 \leq t < t_1$, $0 \leq t < t_2$, and $0 \leq t < t_3$, respectively, and also turn right, right, and left during the time intervals $t_f - \tilde{t}_1 \leq t \leq t_f$, $t_f - \tilde{t}_2 \leq t \leq t_f$, and $t_f - \tilde{t}_3 \leq t \leq t_f$, respectively) that connects vehicles 1 to 2 and 2 to 3 in 9.4 seconds, while Figure 3.2(b) shows the global optimal solution RRL-RLL with $t_f = 8.3$ seconds.

The time-optimal path for each vehicle consists of two arcs of maximum curvature in concatenation with a straight line tangent to the arcs. We noticed that if the arc covers more than π, turning the other direction may result in a shorter path. All of the arcs in Figure 3.2(b) are less than π, hence yielding the global optimal solution. We use this fact to determine the control for each vehicle as follow:

Step 1 (Determine the Candidate) Initial data are the given initial conditions $[x_{i0}, y_{i0}, \theta_{i0}]$. We use the centroid of the polygon $[x_c, y_c]^T$ to estimate as the point that every vehicle needs to travel to. The idea is depicted in Figure 3.1(b). We then use the line connecting each initial location to the centroid as an initial guess for the optimal path. Since these lines do not have the same length and the turning portion of the path are not considered, this guess may not result in the right candidate. Let $\mathbf{r}_{ic}$ denote a vector from $[x_{i0}, y_{i0}]^T$ to $[x_c, y_c]^T$ and $\mathbf{v}_{i0} = [\cos\theta_{i0}, \sin\theta_{i0}]^T$ denote vehicle i initial velocity. The angle between $\mathbf{v}_{i0}$ and $\mathbf{r}_{ic}$ is denoted as α_i. Vehicle i needs to turn from its initial heading to coincide with $\mathbf{r}_{ic}$. We determine the choice of control by choosing the direction with a smaller turn (less than π). The control for the first segment is chosen from

$$u_i^*(t < t_i) = \begin{cases} 1 & k_z \cdot (\mathbf{v}_{i0} \times \mathbf{r}_{ic}) > 0 \ , \\ -1 & k_z \cdot (\mathbf{v}_{i0} \times \mathbf{r}_{ic}) \leq 0 \ , \end{cases} \tag{3.11}$$

where k_z is the unit vector perpendicular to the xy-plane.

Furthermore, we use the vector sum of all $\mathbf{r}_{ic}$ to estimate the final heading of the formation $\theta_i(t_f), \forall i$. Let $\mathbf{v}_f = [\cos\theta_i(t_f), \sin\theta_i(t_f)]^T = \sum_{i=1}^{n} \mathbf{r}_{ic}$ denote the final velocity of the formation. The

angle between $\mathbf{v}_f$ and $\mathbf{r}_{ic}$ is denoted as $\tilde{\alpha}_i$. The control for the third segment is then chosen in the direction corresponding to the smaller turn

$$u_i^*(t_f - \tilde{t}_i < t < t_f) = \begin{cases} 1 & k_z \cdot (\mathbf{r}_{ic} \times \mathbf{v}_f) > 0 \ , \\ -1 & k_z \cdot (\mathbf{r}_{ic} \times \mathbf{v}_f) \leq 0 \ , \end{cases} \tag{3.12}$$

It is also noted that if either $k_z \cdot (\mathbf{v}_{i0} \times \mathbf{r}_{ic})$ or $k_z \cdot (\mathbf{r}_{ic} \times \mathbf{v}_f)$ or both have a small value or value close to π, we need to find the solution for both candidates and compare the results. This will be discussed further in Step 3.

Step 2 (Solve for the Switching Times) The switching times is determined by solving nonlinear equations in Proposition 3.2.4. This often involves the use of a numerical solver which requires that we pick the initial guess for the solution. The unknowns are $t_i, \tilde{t}_i, \nu_l, \tilde{\nu}_l, s_l$, and t_f, where $i = 1, ..., n, l = 1, ..., n-1$. According to Figure 3.1(b), the vehicle needs to turn and align with $\mathbf{r}_{ic}$ then proceed straight toward the centroid. This estimation is made as if the vehicle does not move forward while turning.

As the maximum turn-rate is set to 1, the time required for vehicle i to turn toward the centroid equals the angle between $\mathbf{v}_{i0}$ and $\mathbf{r}_{ic}$. Hence, the initial guess for t_i should be α_i. Similarly, the initial guess for $\tilde{t}_i$ is made to be $\tilde{\alpha}_i$. Furthermore, with a unit speed, the time to travel to $[x_c, y_c]^T$ equals $\|\mathbf{r}_{ic}\|$ and the total time for vehicle i can be estimated as $\|\mathbf{r}_{ic}\| + \alpha_i + \tilde{\alpha}_i$. Since every vehicle should have the same t_f, the initial guess is picked as the average of this value $\frac{1}{n}\sum_{i=1}^{n}(\|\mathbf{r}_{ic}\| + \alpha_i + \tilde{\alpha}_i)$.

As for ν_l, Proposition 3.2.4 (3) shows that the velocity vector during the second segment of vehicle i is formed by the linear combination of that from its neighbors with the factor of ν_{ij}, $j \in \mathcal{N}_i$. Since the final location for each vehicle is likely to stay inside the convex hull of the initial locations, ν_l shall have the same sign. Therefore we choose the initial guess as $\forall l, \nu_l = 1$ and similarly $\tilde{\nu}_l = 1$. For s_l, since the final constraint in (3.3) is expected to close to equality, we choose $\forall l, s_l = 0$.

Step 3 (Verify Results) It is important to verify if the solver gives $0 < t_i, \tilde{t}_i < \pi$ when we pick $0 < t_i^{\text{guess}}, \tilde{t}_i^{\text{guess}} < \pi$. If not, it means that our estimation using the centroid of the polygon is not valid. When this happens, we need to compare the result with the candidate where vehicle i turns the other direction. Furthermore, if t_i has the value close to 0 or π, it is not clear whether the control u_i yields the global solution.

Our proposed algorithm can be summarized in Protocol 1. It is noted that solving the Optimal

Protocol 1 Optimal Target Tree-Graph Algorithm

Step 1 **for** $i = 1 \rightarrow n$
  if $k_z \cdot (\mathbf{v}_{i0} \times \mathbf{r}_{ic}) > 0$ **then**
    $u_i^*(0 < t < t_i) = 1$
  else
    $u_i^*(0 < t < t_i) = -1$
  end
end (Repeat step 1 for $u_i^*(t_f - \tilde{t}_i < t < t_f)$)

Step 2 set $t_i^{\text{guess}} = \alpha_i, \tilde{t}_i^{\text{guess}} = \tilde{\alpha}_i, t_f^{\text{guess}} = \frac{1}{n}\sum_{i=1}^{n}(\|\mathbf{r}_{ic}\| + \alpha_i + \tilde{\alpha}_i)$
Solve for $\forall i, t_i, \tilde{t}_i, t_f$ using NL solver

Step 3 **for** $i = 1 \rightarrow n$
  if $t_i > \pi$ **then**
    $u_i^{*\text{new}}(t < t_i) \Leftarrow -u_i^*(t < t_i)$ Repeat step 2 to find $\forall i, t_i^{\text{new}}, t_f^{\text{new}}$
    if $t_f^{\text{new}} < t_f$ **then**
      $t_f \Leftarrow t_f^{\text{new}}, \forall i, t_i \Leftarrow t_i^{\text{new}}$
    end
  end
end (Repeat step 3 for $\tilde{t}_i$ and $u_i^*(t_f - \tilde{t}_i < t < t_f)$)

Target Tree-graph problem across all possible labeled tree provides a method to solve the original problem that requires $\lambda_2(\mathcal{G}) > 0$. In fact, solving for $\lambda_2(\mathcal{G}) > 0$ induces a connected graph $\mathcal{G}$ which necessarily contains at least one tree $\mathcal{T}$. However, the number of labeled trees on n nodes is n^{n-2}, which makes the search for the best solution across all possible trees impractical when applying this algorithm to more than five UAVs due to the long computational time. In the next section, we introduce a relaxation method to solve the original problem for the larger number of UAVs.

3.3 Nonlinear Programming Method

In this section, we present a NLP algorithm to search for the parameterized optimal solutions. We focus on searching for the optimal controls and switching times that satisfy the terminal graph connectivity constraints. The existence of an edge between two vehicles is encoded as an integer variable constrained by Euclidean distance at time t_f. The elements of the Laplacian for the graph at the final time are therefore linear functions of these nonlinear integer variables. The complications of constraining λ_2 for this integer-valued Laplacian matrix suggest finding a relaxation to the discrete form of the connectivity function. The data communication rate using WiFi devices, for example, drops off as the distance between the vehicles grows beyond a given threshold. We use the bump function [43]

$$w(d_{ij}) = 1/(1 + e^{\alpha(d_{ij} - \rho)}), \tag{3.13}$$

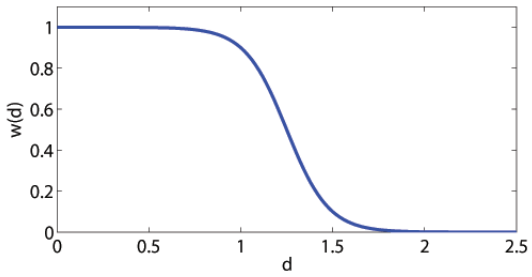


Figure 3.3: The bump function $w(d)$ with $\rho_1 = 1$, $\rho_2 = 1.5$ and $\epsilon = 0.1$.

where $\alpha = \frac{1}{\rho_2 - \rho_1}\log(\frac{1-\epsilon}{\epsilon})$ and $\rho = \frac{\rho_1+\rho_2}{2}$, to represent the communication strength between a pair of UAVs. This leads to a weighted adjacency matrix with entries $a_{ij} = w(d_{ij}(t_f))$ to represent the element a_{ij} for graph $\mathcal{G}$ at time t_f. The plot of this function is shown in Figure 3.3. However, constraining λ_2 for the weighted Laplacian matrix is still non-trivial. We thus use the Cholesky decomposition to convert the eigenvalue constraint to an equivalent semidefinite constraint.

Corollary 3.3.1 *For a graph Laplacian $\mathcal{L}(\mathcal{G})$, there exist μ such that $\mathcal{L}(\mathcal{G}) + \mu \mathbf{1}\mathbf{1}^T/n \succ 0$ is equivalent to $\lambda_2(\mathcal{G}) > 0$.*

Proof. See [55]: Corollary 6.4. ■

Lemma 3.3.2 *[50] Every symmetric positive definite matrix M has a unique factorization of the form $M = LL^T$, where L is a lower triangular matrix with positive diagonal entries.*

From Cholesky decomposition, when the network is connected, we can find a lower triangular matrix L with positive diagonal entries for the positive definite matrix $\mathcal{L}(\mathcal{G}) + \mu \mathbf{1}\mathbf{1}^T/n$. We use this property to transform the problem into the following parameter optimization problem where the matrix L, the switching time $t_i, \tilde{t}_i, i = 1, ..., n$, and t_f are expressed as unknown variables, without specifying the target graph:

$$\min_{t_1,...,t_n,\tilde{t}_1,...,\tilde{t}_n,t_f,l_{ij}} t_f, \tag{3.14}$$

subject to

$$\theta_i(t_f) = \theta_j(t_f),\ 0 \leq t_i, \tilde{t}_i \leq \pi$$

$$d_{ij}(t_f) = \|(x_i(t_f), y_i(t_f)) - (x_j(t_f), y_j(t_f))\|$$

$$a_{ij} = 1/(1 + e^{\alpha(d_{ij}(t_f)-\rho)})\, \forall\, i, j \in 1, \ldots, n$$

$$\mathcal{L}(\mathcal{G}) + \mu \mathbf{1}\mathbf{1}^T/n - \xi I = LL^T$$

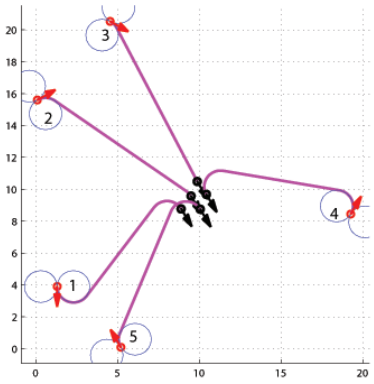
$$L = \begin{bmatrix} l_{11} & & & \\ l_{21} & l_{22} & \mathbf{0} & \\ \vdots & \ddots & \ddots & \\ l_{n1} & \ldots & \ldots & l_{nn} \end{bmatrix}, l_{ii} > 0, \forall i \in 1, \ldots, n,$$

where $x_i(t_f)$ and $y_i(t_f)$ is derived from (3.10), the control u_i is derived from (3.11) and (3.12), and $\xi \in \mathbb{R}$ is a positive small number to guarantee that the weighted graph is connected. In this method, the final connected configuration corresponds with the optimal trajectories is obtained from the optimization routine. However, due to the nonlinearity of the problem, this method may return a local optimal solution depending on the initial picks of the unknown variables.

3.4 Simulation Results

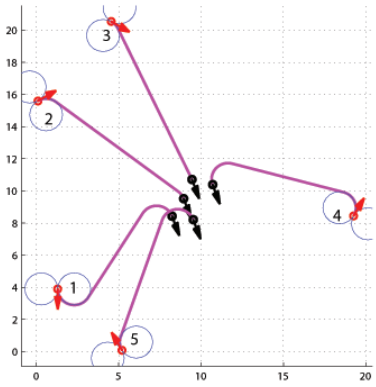
In this section, we compare the time-optimal trajectories using both methods from Section 3.2 and Section 3.3 for the same set of initial conditions. Figure 3.4 depicts the trajectories of five unicycle-UAVs corresponding to the shortest travel time to establish the tree-graph connectivity using the method developed in Section 3.2. The tree is prescribed as a Path-Graph where vehicle 1 is connected to 2, vehicle 2 to 3, vehicle 3 to 4, and vehicle 4 to 5, and $t_f = 11.35$ seconds. On the other hand, Figure 3.5 shows the trajectories from the NLP method with $t_f = 11.00$ seconds as we pick $\alpha = 4.4$, $\rho = 1$, and $\xi = 0.3$.

We notice that the final time from the NLP method is slightly faster than that from the Optimal Target Tree-Graph method. This is because the NLP algorithm produces a different connected configuration, e.g., vehicle 5 is "more" connected to vehicle 1 and 2, but "less" connected to 4.



$$\mathcal{L}(\mathcal{G}(t_f)) = \begin{bmatrix} 1 & -1 & 0 & 0 & 0 \\ -1 & 2 & -1 & 0 & 0 \\ 0 & -1 & 2 & -1 & 0 \\ 0 & 0 & -1 & 2 & -1 \\ 0 & 0 & 0 & -1 & 1 \end{bmatrix}.$$

Figure 3.4: Simulation using the algorithm in Section 3.2 for the Optimal Target Tree-graph Problem where the tree is prescribed as a path-graph: vehicle 1 is connected to 2, vehicle 2 to 3, vehicle 3 to 4, and vehicle 4, to 5.



$$\mathcal{L}(\mathcal{G}(t_f)) = \begin{bmatrix} 0.424 & -0.211 & -0.001 & -0.001 & -0.212 \\ -0.211 & 0.565 & -0.212 & -0.014 & -0.128 \\ -0.001 & -0.212 & 0.424 & -0.210 & -0.001 \\ -0.001 & -0.014 & -0.210 & 0.226 & -0.002 \\ -0.212 & -0.128 & -0.001 & -0.002 & 0.343 \end{bmatrix}.$$

Figure 3.5: Simulation trajectories using NLP method when the initial positions and headings are set the same as Figure 3.4.

3.5 Remarks

In this chapter, two approaches for establishing a proximity network among a collection of scattered unicycle-vehicles are developed. The control law and path synthesis are derived using Pontryagin minimum principle. The algorithm for finding the optimal solution for the fixed final network topology is first introduced, followed by a nonlinear programming method in which the final connected configuration corresponds with the optimal paths emerges from the optimization routine. While the first method gives the exact global optimal solution, the computational time can get very long and is not suitable for applying to more than 5 vehicles. The NLP method, on the other hand, is more

suitable with large UAV networks.

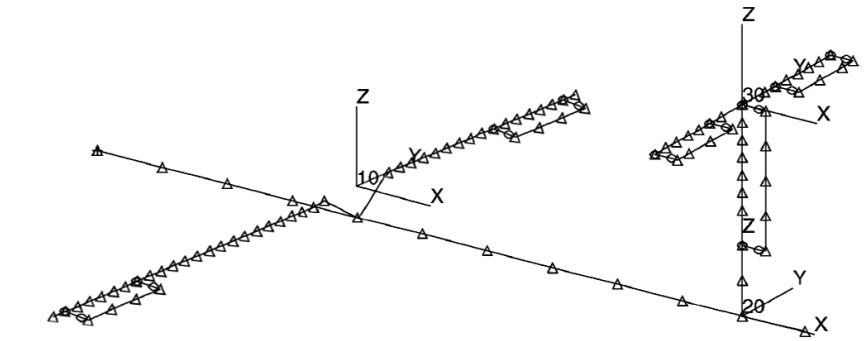
There are several possible extensions of this work. First, the assumption that the initial locations of the UAVs form a convex polygon should be relaxed. This may require a set of commands that allow some vehicles to loiter while other vehicles arrive into the area. One possible extension is to apply the algorithm in the opposite direction, instead of bringing the UAVs into connection, we can disperse the formation to the specified area with the minimum time. The other extension involves allowing the UAV speed to vary between stall and maximum speeds with some vehicle traveling faster than others if necessary. This can open up the possibility of developing an algorithm that not only brings the UAVs to a connected network, but also forms a desired formation shape. Finally, we can consider the scenario that combines the optimal path-planning for network connectivity, maintaining the formation, and optimal network dispersion.

Chapter 4

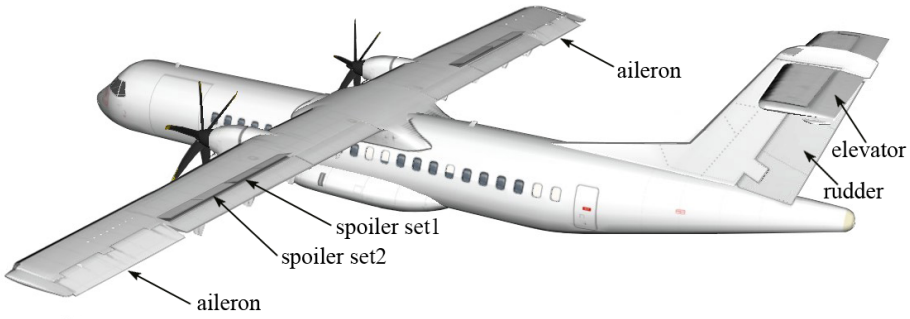
Gust Load Alleviation for Flexible Aircraft

The third area of research explored in this dissertation involves the environmental uncertainty due to wind gusts. Gust load alleviation (GLA) systems can be used to reduce the effects of gust turbulence on vertical acceleration of the aircraft, to decrease the airframe load, and to improve passenger comfort. The guarantee of load reduction around various critical airframe locations can result in weight reduction, and consequently lower the fuel expenditure, making the aircraft more eco-efficient and attractive for the customers.

In this chapter, we discuss the development of a six-degrees-of-freedom model for the generic transport aircraft (GTA) that includes rigid body dynamics, structural characteristics and the coupling between the two. The aerodynamic model is derived based on a method similar to that discussed in [82]. The stick-and-beam finite element model from the vehicle description in the ZAERO user's manual [80] is utilized to create a structural dynamic model. The modifications of mass and mass distribution are made to decrease the structural frequencies and to improve the aircraft static margin. The developed unified dynamics can effectively integrate large rigid body motion with structural deflections. As a suitable control method to address GLA for flexible aircraft, we propose model predictive control (MPC). MPC is a discrete online optimization-based control strategy that can minimize the load over a finite prediction horizon. It allows us to work with either linear or nonlinear dynamics, directly consider constraints on the state, inputs, and outputs, and incorporate the preview



(a)



(b)

Figure 4.1: (a) GTA structural model. (b) GTA aerodynamic model.

of gust information into the dynamics. The prediction accuracy of MPC can be improved by taking estimated gusts ahead of the aircraft into account while performing the optimization routine. Gusts may be measured using light detection and ranging (LIDAR) system.

With the desire to reduce the bending and torsion load on the wing, the idea of a "*flap-array*" is introduced. The flap-array is composed of small trailing edge flaps throughout the entire span of the wing. It may be used in conjunction with the spoilers that are divided evenly along the span. This gives the ability to change the lift profile more delicately and thus efficiently reduce the wing loads.

The organization of this chapter is as follows. Section 4.1 presents the GTA model derivation which includes rigid body and structural dynamics and the coupling between the two. Section 4.2 provides the model setup for the GLA design. The wind model, actuators, and load calculation are discussed. In Section 4.3, the detail formulation for various cases of model predictive controllers are introduced. Section 4.5 illustrates simulation scenarios with discrete and continuous gusts. Concluding remarks are provided in Section 4.6.

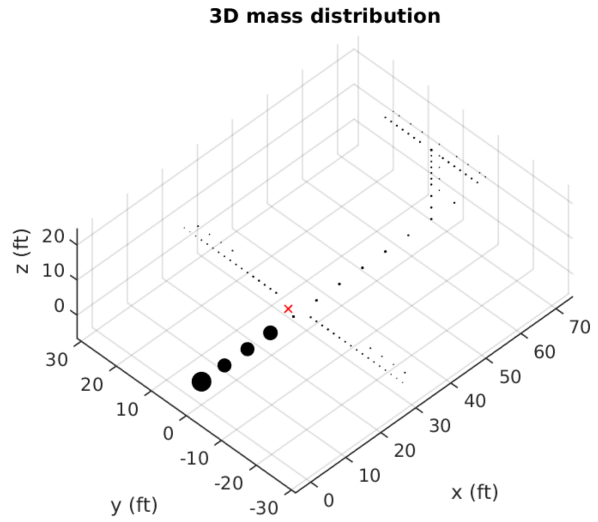


Figure 4.2: GTA Mass distribution (sphere size indicates amount of lumped mass).

4.1 Model for Generic Transport Aircraft

A fully flexible model for the Generic Transport Aircraft (GTA) with a straight wing and T-tail is developed based on the vehicle description in the ZAERO user's manual [80]. That description provides a beam-and-stick finite element model of the GTA for calculating normal mode using MSC/NASTRAN as depicted in Figure 4.1(a). In the ZAERO manual, the GTA is defined for use in demonstrating a gust loading aeroelastic dynamic analysis. The model contains 117 grid points and 137 structural elements. The aerodynamic model is created based on the structural definition as depicted in Figure 4.1(b). The center of gravity of the structural model is so far forward that the static margin of the aircraft was between 70 to 80%. Hence, the modification has been made to the original model by adding weight into a lumped mass toward the tail of the aircraft. The updated mass distribution is shown in Figure 4.2. The modified center of gravity is located 0.1 feet behind the wing aerodynamic center. The mass of all elements are also doubled in order to reduce the first structural frequency to 5.6Hz which is more realistic for the aircraft of its size. The weight of the modified model has become 31696 lbs.

4.1.1 Vehicle Geometry and Aerodynamics properties

The GTA vehicle is similar in size and configuration to the ATR-42 or Bombardier Dash-8-100. It is a twin engine, turbo-prop transport. The vehicle has overall dimensions depicted in Figure 4.3.

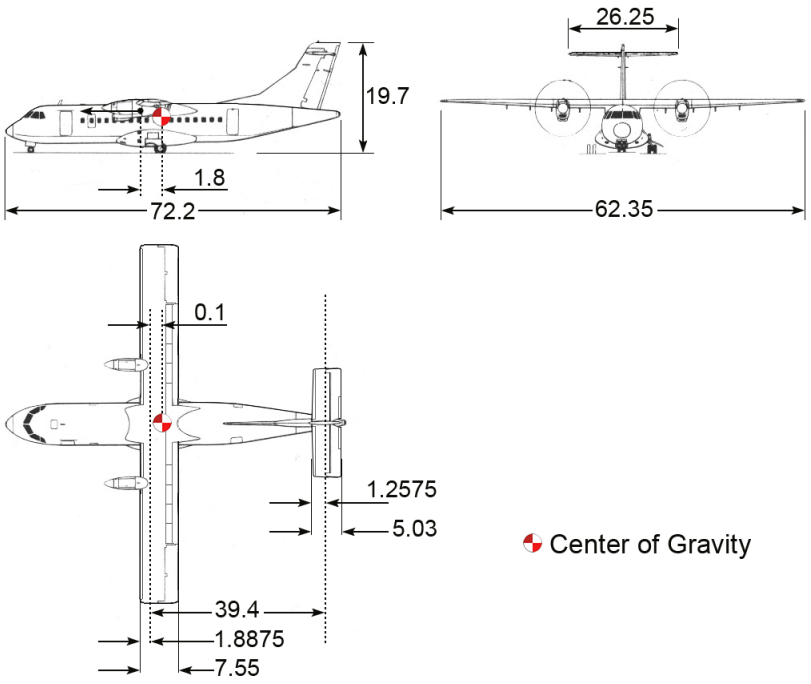


Figure 4.3: Three views of the Generic Transport Aircraft (dimensions in feet), the location of the engine thrust is shown to be 1.8 feet in front of the CG and on the same height of the CG.

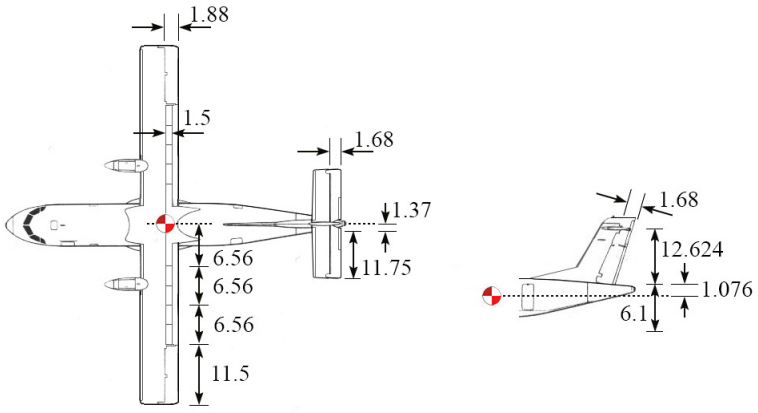


Figure 4.4: Control surfaces of Generic Transport Aircraft (dimensions in feet).

The main wing is rectangular in form and is mounted above the fuselage. It is of constant chord and not swept. The tail is a T-tail configuration. The control surfaces are shown in Figure 4.4. Given the rectangular shape, the aspect ratio of the wing is $A_w = 8.257$ and the Oswald efficiency factor

for the wing e_w, horizontal tail e_H, and vertical tail e_V are taken to be 0.9. The main wing airfoil is a NACA 0009. The flaperons (a combined flap and ailerons) are 25% of the wing chord results in an aileron effectiveness coefficient $\tau_{\delta_a} = \tau_{\delta_f} = 0.3$. The flap and aileron deflection are noted as δ_f and δ_a, respectively. There are 2 sets of spoilers. Each takes 20% of the wing chord. The spoilers effectiveness is $\tau_{\delta_{sp1}} = \tau_{\delta_{sp2}} = -0.14$. The horizontal and vertical stabilizer airfoil are also NACA 0009. Both elevator and rudder take 33% chord yielding the effectiveness coefficient of $\tau_{\delta_e} = \tau_{\delta_r} = 0.75$. The elevator, rudder, and spoiler deflection are noted as δ_e, δ_r, and δ_{sp}, respectively. The overall dimensions of lifting and control surfaces are summarized in Table 4.1.

Table 4.1: Lifting and control surfaces geometry

Property	Value	Units	Property	Value	Units
Wing Area S_w	470.67	ft^2	Horizontal tail Area S_H	132.04	ft^2
Wing MAC $\bar{c}_w$	7.55	ft	Horizontal tail MAC $\bar{c}_H$	5.03	ft
Wing Span b_w	62.34	ft	Horizontal tail Span b_H	26.25	ft
Wing Incidence i_w	1.0	deg	Horizontal tail Incidence i_H	0.0	deg
Flaperon Area (both sides)	43.24	ft^2	Elevator Area (both sides)	36.96	ft^2
Flaperon MAC	1.88	ft	Elevator MAC	1.68	ft
Flaperon Span (both sides)	23	ft	Elevator Span (both sides)	23.50	ft
Flaperon effectiveness coefficient	0.3		Elevator effectiveness coefficient	0.75	
Spoiler1,2 Area (both sides)	19.68	ft^2	Rudder Area	21.21	ft^2
Spoiler1,2 MAC	1.5	ft	Rudder MAC	1.68	ft
Spoiler1,2 Span (both sides)	13.12	ft	Rudder Span	12.624	ft
Spoiler1,2 effectiveness coefficient	-0.14		Rudder effectiveness coefficient	0.75	
Vertical tail Area S_V	71	ft^2	Vertical tail MAC $\bar{c}_V$	5.03	ft
Vertical tail Height b_V	14.115	ft			

4.1.2 Rigid Body Dynamics

GTA six degrees of freedom rigid body dynamics derived from force and moment equations in fuselage-reference axes (see Figure 4.5) has the form

$$
\begin{aligned}
\dot{u} &= -qw + vr - g\sin\theta + F_x/m, \\
\dot{v} &= -ru + pw + g\cos\theta\sin\phi + F_y/m, \\
\dot{w} &= -pv + qu + g\cos\theta\cos\phi + F_z/m, \\
\dot{q} &= \frac{1}{I_{yy}}[(I_{zz} - I_{xx})pr + I_{xz}(r^2 - p^2) + M], \\
\begin{bmatrix} \dot{p} \\ \dot{r} \end{bmatrix} &= \left(\frac{1}{I_{xx}I_{zz} - I_{xz}^2}\right) \begin{bmatrix} I_{zz} & I_{xz} \\ I_{xz} & I_{xx} \end{bmatrix} \begin{bmatrix} I_{xz}pq + (I_{yy} - I_{zz})rq + \hat{L} \\ -I_{xz}qr + (I_{xx} - I_{yy})pq + N \end{bmatrix},
\end{aligned}
\tag{4.1}
$$

where $\mathbf{V} = [u, v, w]^T$ represents vehicle velocity vector expressed in the fuselage reference frame with u, v, and w as the components in x, y, and z axis, respectively, $\omega = [p, q, r]^T$ represents the vehicle angular velocity vector expressed in the fuselage reference frame with p, q, and r as the component in x, y, and z axis, respectively, $I_{xx}, I_{yy}, I_{zz}, I_{xz}$ are the product of inertia around the principle axes, m is the vehicle mass, and g is the gravitational constant at sea level. The vectors $\mathbf{F} = [F_x, F_y, F_z]^T$ and $\mathbf{M} = [\hat{L}, M, N]^T$ represent external forces and moments expressed in the fuselage reference frame. Both forces and moments are composed of aerodynamic terms and propulsive terms, e.g., $F_x = F_{A_x} + F_{P_x}$. In most cases, $u \gg w$ and $u \gg v$ so w and v can be expressed as an angle of attack $\alpha \approx w/\|\mathbf{V}\|$ and a sideslip angle $\beta \approx v/\|\mathbf{V}\|$.

In addition to equation (4.1), the kinematic equation describing the relationship between angular rates and Euler angles ϕ, θ, and ψ is also included. The kinematic equation has the form

$$
\begin{aligned}
\dot{\phi} &= p + q\sin\phi\tan\theta + r\cos\phi\tan\theta, \\
\dot{\theta} &= q\cos\phi - r\sin\phi, \\
\dot{\psi} &= q(\sin\phi + r\cos\phi)\sec\theta.
\end{aligned}
\tag{4.2}
$$

In some cases, the kinematic equation describing the relationship between vehicle altitude h and

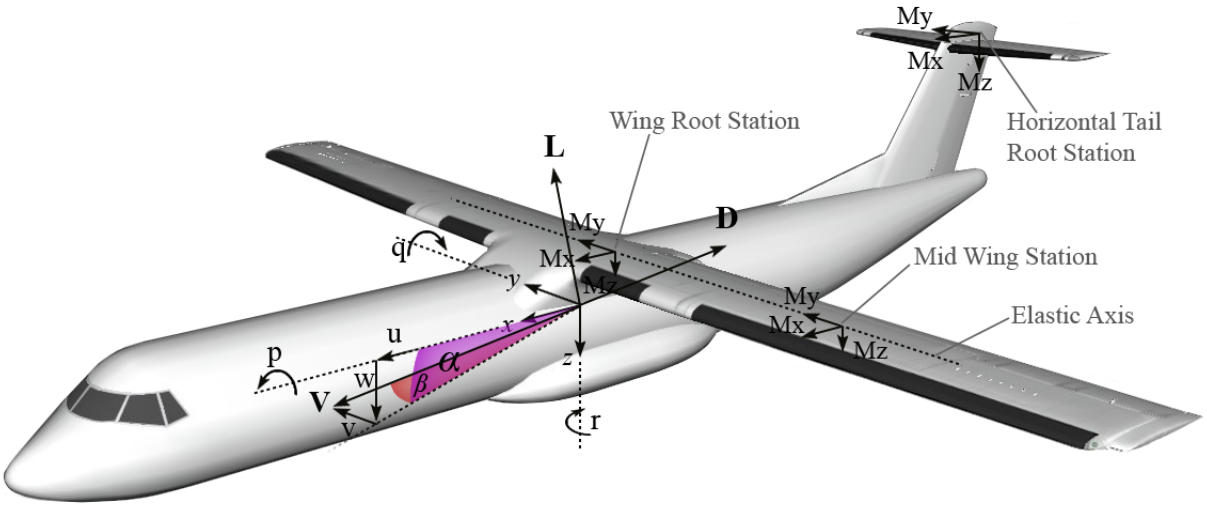


Figure 4.5: Velocity, angular velocity, and aerodynamic forces (side force is neglected for cleanliness of the figure) expressed in fuselage reference axes system; load stations are also shown in gray.

velocity $\mathbf{V}$ maybe included as well. The change in altitude is given by

$$\dot{h} \quad = \quad u \sin\theta - v \sin\phi \cos\theta - w \cos\phi \cos\theta. \tag{4.3}$$

Now, equations (4.1), (4.2), and (4.3) are composed to be a GTA rigid body equation of motion. The contribution of aerodynamic forces F_{A_x}, F_{A_y}, and F_{A_z} are lift L, drag D, and side force S, while the contribution of propulsive force F_{P_x}, F_{P_y}, F_{P_z} are engine thrust T and normal forces F_{N_Z} and F_{N_Y}. The aerodynamic forces L and D along with velocity and angular velocity in fuselage reference frame are depicted in Figure 4.5. The components of aerodynamic forces in the vehicle-fixed axes are given by

$$\begin{aligned} F_{A_x} &= C_x q_\infty S_w = -D\cos\alpha\cos\beta - S\sin\alpha\cos\beta + L\sin\alpha, \\ F_{A_y} &= C_y q_\infty S_w = -D\sin\beta + S\cos\beta, \\ F_{A_z} &= C_z q_\infty S_w = -D\sin\alpha\cos\beta - S\sin\alpha\sin\beta - L\cos\alpha, \end{aligned} \tag{4.4}$$

where C_x, C_y, C_z are non-dimensional coefficients of F_{A_x}, F_{A_y}, F_{A_z} respectively, $q_\infty = \frac{1}{2}\rho\|\mathbf{V}\|^2$ is the dynamic pressure, and S_w is the wing area. Lift L, drag D, and side force S can be expressed in terms

of the corresponding coefficients as $L = C_L q_\infty S_w$, $D = C_D q_\infty S_w$, and $S = C_S q_\infty S_w$ where C_L, C_D, C_S are lift, drag, and side force coefficients, respectively. Furthermore, the aerodynamic moments can be expressed as $\hat{L}_A = C_{\hat{L}} q_\infty S_w b_w$, $M_A = C_M q_\infty S_w \bar{c}_w$ and $N_A = C_N q_\infty S_w b_w$, where $C_{\hat{L}}$, C_M, C_N are coefficients of aerodynamic moments $\hat{L}$, M_A, and N_A, respectively, b_w is the wing span, and $\bar{c}_w$ is the wing chord.

The engines of GTA are mouthed such that the thrust is parallel to body x-axis. The location of thrust is $1.8ft$ forward of the CG ($x_T = 1.8$) and at the same height of that of CG ($z_T = 0$). Both engines are symmetrical about $x - z$ plane. The components of GTA propulsive forces in the vehicle-fixed axes are then given by

$$\begin{aligned} F_{P_x} &= C_{P_x} q_\infty S_w = T(\mathbf{V}, h, \pi), \\ F_{P_y} &= C_{P_y} q_\infty S_w = F_{N_Y}, \\ F_{P_z} &= C_{P_z} q_\infty S_w = F_{N_Z}, \end{aligned} \tag{4.5}$$

while the component of GTA propulsive moments are given by

$$\begin{aligned} \hat{L}_P &= C_{P_{\hat{L}}} q_\infty S_w b_w = 0, \\ M_P &= C_{P_M} q_\infty S_w \bar{c}_w = F_{N_Z} x_T, \\ N_P &= C_{P_N} q_\infty S_w b_w = F_{N_Y} x_T, \end{aligned} \tag{4.6}$$

where C_{P_x}, C_{P_y}, C_{P_z}, $C_{P_{\hat{L}}}$, C_{P_M}, C_{P_N} are non-dimensional coefficients of F_{P_x}, F_{P_y}, F_{P_z}, $\hat{L}_P$, M_P, N_P, respectively, π is the throttle setting, and F_{N_Y} and F_{N_Z} are normal forces on the engine in the body y and z-axis directions.

In order to integrate equation (4.4) into (4.1), we may write the coefficients for aerodynamic lift, side force, and drag, as

$$\begin{aligned} C_L &= C_{L_0} + C_{L_\alpha}\alpha + C_{L_{\dot{\alpha}}}\dot{\alpha} + C_{L_u} u + C_{L_q} q + C_{L_{\delta_e}}\delta_e + C_{L_{\delta_f}}\delta_f + C_{L_{\delta_{sp1}}}\delta_{sp1} + C_{L_{\delta_{sp2}}}\delta_{sp2}, \\ C_S &= C_{S_\beta}\beta + C_{S_p} p + C_{S_r} r + C_{S_{\delta_r}}\delta_r + C_{S_{\delta_a}}\delta_a, \\ C_D &= C_{D_0} + \hat{C}_{D_{\delta_{sp1}}}\delta_{sp1} + \hat{C}_{D_{\delta_{sp2}}}\delta_{sp2} + \left(\frac{C_{L_w}^2}{\pi A_w e_w} + \frac{C_{L_H}^2}{\pi A_H e_H}\frac{q_H S_H}{q_\infty S_W} + \frac{C_{S_V}^2}{\pi A_V e_V}\frac{q_V S_V}{q_\infty S_V} \right), \end{aligned} \tag{4.7}$$

where C_{L_0} and C_{D_0} are lift and drag coefficients when $\alpha = i_H = \delta_e = \delta_f = \delta_{sp1} = \delta_{sp1} = 0$ and i_H

is the tail incidence angle, C_{L_α}, $C_{L_{\dot{\alpha}}}$, C_{L_u}, C_{L_q}, $C_{L_{\delta_e}}$, $C_{L_{\delta_f}}$, $C_{L_{\delta_{sp1}}}$, and $C_{L_{\delta_{sp2}}}$ are lift coefficient derivatives with respect to α, $\dot{\alpha}$, u, q, δ_e, δ_f, δ_{sp1}, and δ_{sp2}, respectively, C_{S_β}, C_{S_p}, C_{S_r}, $C_{S_{\delta_r}}$, and $C_{S_{\delta_a}}$ are side force coefficient derivatives with respect to β, p, r, δ_r, and δ_a, respectively, $\hat{C}_{D_{\delta_{sp1}}}$ and $\hat{C}_{D_{\delta_{sp2}}}$ are the direct effects of δ_{sp1} and δ_{sp2} on the drag coefficient, respectively, q_H is the dynamic pressure at tail, S_H is the horizontal tail area, and S_V is the horizontal tail area. To use the above equation for drag coefficients, we may apply the following model for lift on the wing and tail surfaces,

$$\begin{aligned} C_{L_w} &= C_{L_{\alpha_w}}(\alpha + i_w - \alpha_{0_w} + \tau_{\delta_f}\delta_f + \tau_{\delta_{sp1}}\delta_{sp1} + \tau_{\delta_{sp2}}\delta_{sp2}), \\ C_{L_H} &= C_{L_{\alpha_H}}((1 - \frac{d\epsilon}{d\alpha})(\alpha + i_w) - \frac{d\epsilon}{d\alpha}\alpha_{0_w} + i_H - \alpha_{0_H} + \tau_{\delta_e}\delta_e), \\ C_{S_V} &= C_{S_{\beta_V}}(\beta + \tau_{\delta_r}\delta_r), \end{aligned} \tag{4.8}$$

where i_w and i_H are the wing and horizontal tail incidence angles, respectively, α_{0_w} and α_{0_H} are zero lift angle of attack at the wing and tail respectively, and $\frac{d\epsilon}{d\alpha}$ is the rate of downwash with respect to the change of angle of attack. The drag derivatives can be derived by substituting the terms C_{L_w}, C_{L_H}, and C_{S_V} from (4.8) into (4.7) and take derivative with respect to each parameter. For example, $C_{D_{\delta_{sp1}}} = \hat{C}_{D_{\delta_{sp1}}} + 2C_{L_w}C_{L_{\alpha_w}}\tau_{\delta_{sp1}}/(\pi A_w e_w)$. The second term indicates the induced drag. The terms $\hat{C}_{D_{\delta_{sp1}}}$ and $\hat{C}_{D_{\delta_{sp2}}}$ in our work are calculated based from the C_D characteristic in [83] where the authors show the increased 2D drag coefficient at fixed seven-degrees spoiler deflection for the NACA2415 airfoil[1]. We use this data to approximate the 2D drag coefficients per radian deflection of the spoilers. In [83], the drag coefficient appears to be increased by 0.09 per seven-degrees spoiler deflection uniformly across all operational angle of attack (we use the case where spoiler is located at 0.7 of the chord). Assuming that the drag-spoiler relationship is linear[2], the drag derivatives with respect to spoiler deflection are derived to be $\hat{C}_{D_{\delta_{sp1}}} = \hat{C}_{D_{\delta_{sp2}}} = \frac{0.09}{7}(\frac{180}{\pi})$ per radian. Generally the 3D drag coefficient decreases due to the 3D relieving effect and the fact that the spoiler only span so much. We take the conservative approach by using the 2D data to approximate the 3D drag-spoiler derivatives.

Figure 4.6 shows longitudinal aerodynamic forces and moments for the wing and horizontal tail

[1]The data on drag characteristics reported in [83] is for the NACA2415 airfoil which is different from the NACA0009 airfoil used in the GTA model. However, we believe the approximation of drag-spoiler relationship of NACA2415 is reasonable for the proof of concept for the GLA controller design for GTA as the corresponding data for the NACA0009 airfoil is not available in the open literature.

[2]The wind-tunnel data in NASA technical reports [89], [90] indicate that the relationship between drag coefficient and spoiler tends to be linear for small deflections. The report [89] provides the 3D drag data for a DC-10-30ER, while the reference [90] provides the 3D drag data for a small jet aircraft; neither study specifies the airfoil type.

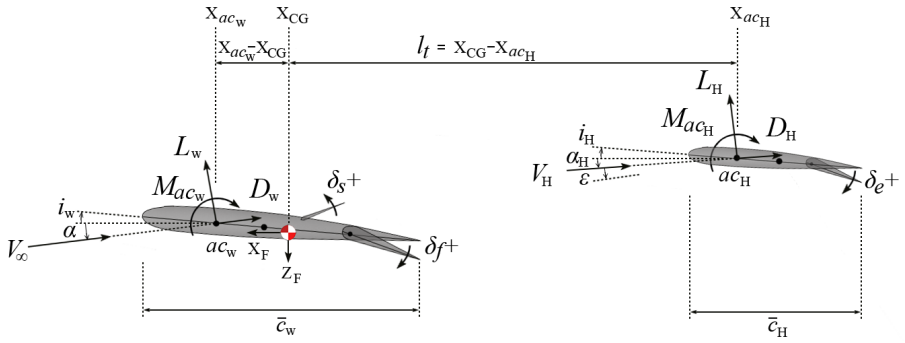


Figure 4.6: A 2D diagram showing longitudinal aerodynamic forces and moments for the wing and horizontal tail, moment arms, and positive deflection of control surfaces.

and related parameters appearing in equation (4.8). Similar to (4.7), we may write the expression for roll, pitch, and yaw moment coefficients as

$$\begin{aligned} C_{\hat{L}} &= C_{\hat{L}_\beta}\beta + C_{\hat{L}_p}p + C_{\hat{L}_r}r + C_{\hat{L}_{\delta_a}}\delta_a + C_{\hat{L}_{\delta_r}}\delta_r, \\ C_M &= C_{M_0} + C_{M_\alpha}\alpha + C_{M_{\dot{\alpha}}}\dot{\alpha} + C_{M_q}q + C_{M_{\delta_e}}\delta_e + C_{M_{\delta_f}}\delta_f, \\ C_N &= C_{N_\beta}\beta + C_{N_p}p + C_{N_r}r + C_{N_{\delta_r}}\delta_r + C_{N_{\delta_a}}\delta_a. \end{aligned} \tag{4.9}$$

Similarly, the propulsive forces and moments can be linearized around trim conditions as

$$\begin{aligned} F_{P_x} &= F_{P_{x_0}} + F_{P_{x_u}}u + F_{P_{x_\alpha}}\alpha + F_{P_{x_{\dot{\alpha}}}}\dot{\alpha} + F_{P_{x_\beta}}\beta + F_{P_{x_q}}q + F_{P_{x_{\delta_T}}}\delta_T, \\ F_{P_y} &= F_{P_{y_0}} + F_{P_{y_\beta}}\beta + F_{P_{y_r}}r + F_{P_{y_{\delta_T}}}\delta_T, \\ F_{P_z} &= F_{P_{z_0}} + F_{P_{z_u}}u + F_{P_{z_\alpha}}\alpha + F_{P_{z_{\dot{\alpha}}}}\dot{\alpha} + F_{P_{z_\beta}}\beta + F_{P_{z_q}}q + F_{P_{z_{\delta_T}}}\delta_T, \\ \hat{L}_P &= \hat{L}_{P_0} + \hat{L}_{P_\beta}\beta + \hat{L}_{P_p}p + \hat{L}_{P_r}r, \\ M_P &= M_{P_0} + M_{P_u}u + M_{P_\alpha}\alpha + M_{P_{\dot{\alpha}}}\dot{\alpha} + M_{P_q}q, \\ N_P &= N_{P_0} + N_{P_\beta}\beta + N_{P_r}r, \end{aligned} \tag{4.10}$$

where $F_{P_{x_0}}$, $F_{P_{y_0}}$, $F_{P_{z_0}}$, $\hat{L}_{P_0}$, M_{P_0}, and N_{P_0} are the propulsive forces and moments at trim conditions, e.g., level flight, and all other propulsive force derivatives are of the form similar to $F_{P_{x_u}}$ that represent the derivative of propulsive forces in body x-axis with respect to the surge velocity u.

Now with the help of (4.4), (4.5), (4.6), (4.7), (4.9), and (4.10), the forces and moments in the rigid body dynamics (4.1) can be expressed as a function of the state variables u, $\alpha = w/\|\mathbf{V}\|$,

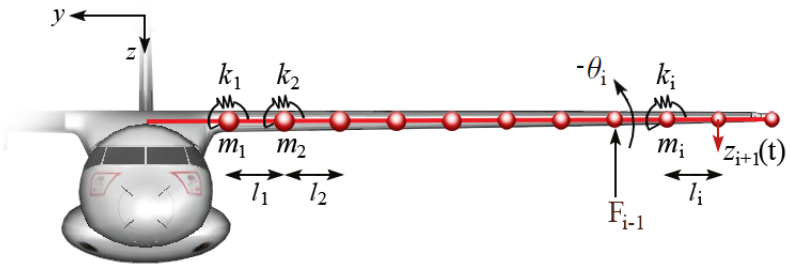


Figure 4.7: Beam/lumped-mass model of the wing.

$\dot{\alpha} = \dot{w}/\|\mathbf{V}\|$, $\beta = v/\|\mathbf{V}\|$, p, q, r, θ, ϕ, h, and controls δ_e, δ_f, δ_r, δ_a, δ_T, δ_{sp1} and δ_{sp2}. The calculation of aerodynamic force and moment coefficients derivative in (4.7), (4.8), and (4.9) can be done using geometric data from Section 4.1.1. The detailed calculations can be found in [82].

4.1.3 Structural Dynamics

In aeroelastic modeling, beam elements have traditionally been used to represent the wing structure. The partial differential equation governing the transverse deformation in the z (vertical) direction of the beam (wing) along the y direction is given as

$$\frac{\partial^2}{\partial y^2}\left(EI(y)\frac{\partial^2 z(y,t)}{\partial y^2}\right) + m(y)\frac{\partial^2 z(y,t)}{\partial t^2} = 0, \tag{4.11}$$

where E is the elastic modulus of the beam material, I is the moment of inertia of the beam cross section about its neutral axis, and $m(y)$ is the mass distribution. Using the separation of variable technique, the solution of (4.11) is of the form $z(y,t) = \sum_{i=1}^{\infty} \nu_i(y)\eta_i(t)$ where $\nu_i(y)$ is the space-dependent function or *mode shapes* and $\eta_i(t)$ is a time-dependent function or *modal coordinates*. This infinite sum can be approximated by truncating it into finite number of terms which is analogous to idealizing the beam into a finite number of lumped-masses and torsional springs. Each mass is connected by one or more massless sticks. We use a beam-and-stick finite element model of the GTA provided by ZAERO user's manual [80] for calculating mode shapes using MSC/NASTRAN. The structural dynamic of the wing can be derived with Lagrangian method using the beam/lumped-mass model along the wing span similar to Figure 4.7. For the beam with N lumped masses, the potential

energy of the beam maybe written as

$$U = \frac{1}{2}\sum_{i=1}^{N} k_i \theta_i^2 = \frac{1}{2}\begin{Bmatrix} \theta_1 \\ \vdots \\ \theta_N \end{Bmatrix}^T \begin{bmatrix} k_1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & k_N \end{bmatrix} \begin{Bmatrix} \theta_1 \\ \vdots \\ \theta_N \end{Bmatrix}, \tag{4.12}$$

while the kinetic energy of the beam maybe written as

$$T = \frac{1}{2}\sum_{i=1}^{N} m_i \dot{z}_i^2 = \frac{1}{2}\begin{Bmatrix} \dot{z}_1 \\ \vdots \\ \dot{z}_N \end{Bmatrix}^T \begin{bmatrix} m_1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & m_N \end{bmatrix} \begin{Bmatrix} \dot{z}_1 \\ \vdots \\ \dot{z}_N \end{Bmatrix}. \tag{4.13}$$

If we choose $\mathbf{q} = [\theta_1, ..., \theta_N]^T$ as a generalized coordinate vector, we can write the relationship between $\dot{\theta}_i$ and $\dot{z}_i$ as

$$\begin{Bmatrix} \dot{z}_1 \\ \vdots \\ \dot{z}_N \end{Bmatrix} = \begin{bmatrix} l_1 & \dots & 0 \\ l_1 + l_2 & \ddots & \vdots \\ l_1 + ... + l_N & \dots & l_N \end{bmatrix} \begin{Bmatrix} \dot{\theta}_1 \\ \vdots \\ \dot{\theta}_N \end{Bmatrix}. \tag{4.14}$$

The wing structural equation is then derived using the Lagrange's equation,

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{\mathbf{q}}_i}\right) - \frac{\partial T}{\partial \mathbf{q}} + \frac{\partial U}{\partial \mathbf{q}} = \mathcal{F}_i. \tag{4.15}$$

By substituting (4.12) and (4.13) into (4.15), the wing equation can be formulated as

$$[\mathbf{M}]\{\ddot{\mathbf{q}}\} + [\mathbf{K}]\{\mathbf{q}\} = \{\mathcal{F}\}, \tag{4.16}$$

where $[\mathbf{M}]$ is the mass matrix and $[\mathbf{K}]$ is the stiffness matrix. Both $[\mathbf{M}]$ and $[\mathbf{K}]$ are real, symmetric, and positive definite indicating that an inverse always exists and all eigenvalues are strictly positive. The wing equation maybe reformulated using the relationship $\{\mathbf{q}\} = [\mathbf{\Phi}]\{\eta\}$ where matrix $[\mathbf{\Phi}] = [\nu_1, \nu_2..., \nu_N]$ is constructed from eigenvectors ν_i of $[\mathbf{M}]^{-1}[\mathbf{K}]$, and $\{\eta\}$ is a new *modal coordinates*,

yielding

$$\begin{aligned} [\mathbf{\Phi}]\{\ddot{\eta}\} + [\mathbf{M}]^{-1}[\mathbf{K}][\mathbf{\Phi}]\{\eta\} &= [\mathbf{M}]^{-1}\{\mathcal{F}\}, \\ \{\ddot{\eta}\} + [\mathbf{\Phi}]^{-1}[\mathbf{M}]^{-1}[\mathbf{K}][\mathbf{\Phi}]\{\eta\} &= [\mathbf{\Phi}]^{-1}[\mathbf{M}]^{-1}\{\mathcal{F}\}. \end{aligned} \tag{4.17}$$

We choose $[\mathbf{\Phi}]$ such that it diagonalizes matrix $[\mathbf{M}]^{-1}[\mathbf{K}]$ and $[\mathbf{\Phi}]^{-1}[\mathbf{M}]^{-1}[\mathbf{K}][\mathbf{\Phi}] = [\mathbf{\Lambda}]$=$[\mathbf{diag}(\omega_i^2)], i = 1, \cdots, N$. The wing equation then becomes

$$\{\ddot{\eta}\} + [\mathbf{\Lambda}]\{\eta\} = [\mathcal{M}]^{-1}\{\mathbf{Q}\}, \tag{4.18}$$

where $\{\mathbf{Q}\}$ is the generalized force vector and $[\mathcal{M}]$ is the generalized mass matrix. These N linear differential equations are now uncoupled and each modal coordinate can be described by

$$\ddot{\eta}_i + \omega_i^2 \eta_i = Q_i/\mathcal{M}_i, \tag{4.19}$$

with a solution of the form $\eta_i(t) = A_i \sin(\omega_i t + \gamma_i)$, where the vibration amplitude A_i and phase shift γ_i can be found from the generalized force Q_i and initial conditions. If the structural damping ζ is applied to the wing model, (4.19) becomes

$$\ddot{\eta}_i + \zeta_i \omega_i \dot{\eta}_i + \omega_i^2 \eta_i = Q_i/\mathcal{M}_i, \tag{4.20}$$

and the displacement $\{\mathbf{q}\}$ is a composition of the mode shape matrix $[\mathbf{\Phi}] = [\nu_1, \nu_2 ..., \nu_N]$ scaled by modal coordinates $\{\eta\} = [\eta_1, \eta_2, ..., \eta_N]^T$ or

$$\{\mathbf{q}(t)\} = [\mathbf{\Phi}(x, y, z)]\{\eta(t)\} = \sum_{i=1}^{N} \nu_i(x, y, z)\eta_i(t). \tag{4.21}$$

For the GTA structural model, every lifting component is modeled as a beam: wing, horizontal tail, vertical tail, and fuselage, as shown in Figure 4.8. The beam is allowed to move and rotate in x, y, and z directions but the major effects on the wing will be wing bending (displacement in the z-direction)

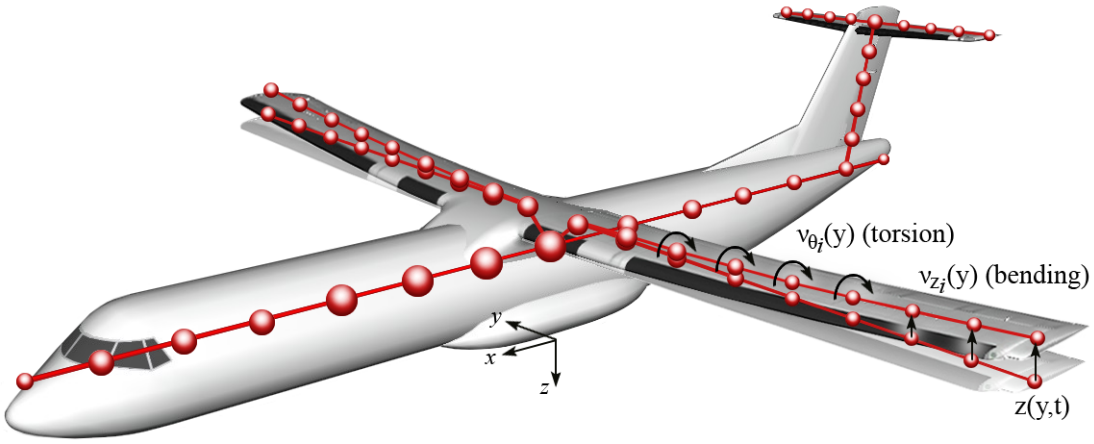


Figure 4.8: Beam/lumped-mass model for GTA with the first wing bending mode demonstrated.

and wing torsion (angular displacement around the body y-axis) of the form

$$z(y,t) = \sum_{i=1}^{N} \nu_{z_i}(y)\eta_i(t) \qquad \text{(bending)}, \tag{4.22}$$

$$\theta(y,t) = \sum_{i=1}^{N} \nu_{\theta_i}(y)\eta_i(t) \qquad \text{(torsion)}, \tag{4.23}$$

where $\nu_{z_i}(y)$ and $\nu_{\theta_i}(y)$ are the mode shapes for the z-direction displacement and the θ angular displacement around the y-axis.

The generalized force Q_i can be derived from the virtual work δW and virtual displacement $\delta\eta$, where $\delta W = Q_i\delta\eta$. Figure 4.9a demonstrates GTA wing bending and torsion displacements whereas Figure 4.9b depicts aerodynamic forces (span-wise 2D lift $l(y)$ and drag $d(y)$) and moment (span-wise 2D moment $m(y)$) that cause the movements. It is noted that generally the wing aerodynamic center where $l(y)$ and $d(y)$ are acted upon is not necessarily the same location as the elastic axis where the stick-beam model is placed. For simplicity, we assume they are the same for the GTA model. The virtual work due to aerodynamic force on the wing can now be written as

$$\begin{aligned}\delta W_{\text{wing}} = \int_{-b_w/2}^{b_w/2} & -(l(y)\cos\alpha + d(y)\sin\alpha)\sum_{i=1}^{N}\nu_{z_i}(y)\delta\eta_i \\ & +(l(y)\sin\alpha - d(y)\cos\alpha)\sum_{i=1}^{N}\nu_{x_i}(y)\delta\eta_i + m(y)\sum_{i=1}^{N}\nu_{\theta_i}(y)\delta\eta_i dy.\end{aligned} \tag{4.24}$$

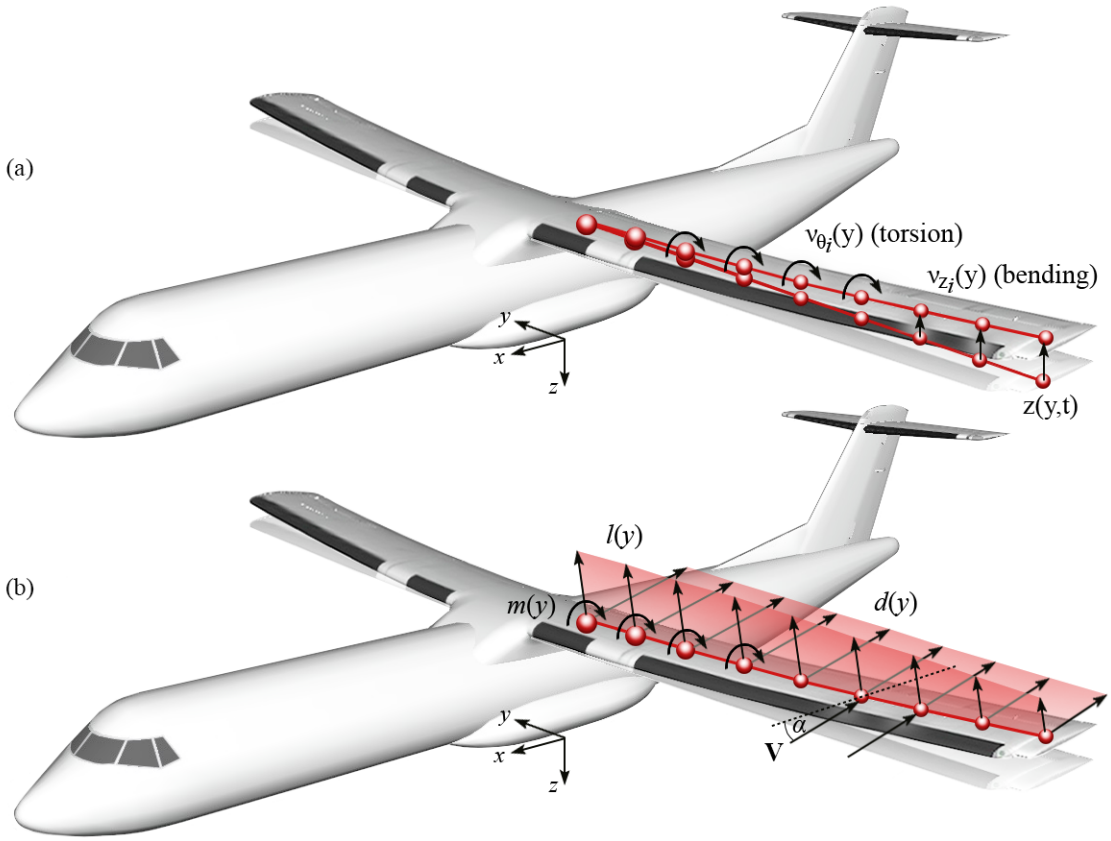


Figure 4.9: (a) Wing bending and torsion displacements. (b) Wing span-wise aerodynamic forces and moments.

The shear movement in the x-direction of the wing is usually so small and neglected ($\nu_{x_i}(y) \approx 0$). The angle of attack α is also small in most flight conditions so that $\sin\alpha \approx 0$ and $\cos\alpha \approx 1$ and (4.24) simplifies to

$$\delta W_{\text{wing}} = \int_{-b_w/2}^{b_w/2} -l(y) \sum_{i=1}^{N} \nu_{z_i}(y)\delta\eta_i + m(y) \sum_{i=1}^{N} \nu_{\theta_i}(y)\delta\eta_i dy, \tag{4.25}$$

and the wing generalized force can be written as

$$Q_{i_{\text{wing}}} = \int_{-b_w/2}^{b_w/2} -l(y)\nu_{z_i}(y) + m(y)\nu_{\theta_i}(y)dy. \tag{4.26}$$

By including the generalized forces from the virtual work terms over all flexible components of the GTA (wing, horizonal tail, and vertical tail), we can write the structural dynamic equation for GTA

as

$$\ddot{\eta}_i + \zeta_i \omega_i \dot{\eta}_i + \omega_i^2 \eta_i = \frac{Q_i}{\mathcal{M}_i} = \frac{1}{\mathcal{M}_i}\left[\int_{-b_w/2}^{b_w/2} -l_w(y)\nu_{z_{i_w}}(y) + m_w(y)\nu_{\theta_{i_w}}(y)dy + \int_{-b_H/2}^{b_H/2} -l_H(y)\nu_{z_{i_H}}(y)dy + \int_0^{b_V} -s_V(z)\nu_{y_{i_V}}(y)dy\right], \tag{4.27}$$

where $l_w(y)$ and $l_H(y)$ are the 2D lift profiles along the wing and the tail, respectively, $m_w(y)$ is the 2D moment profile along the wing, and $s_V(z)$ is the 2D side force profile at the vertical tail. The effect of the wing moment is small compare to the one caused by the lift at the tail so $m_w(y) \approx 0$. The term $\nu_{z_{i_w}}$ and $\nu_{z_{i_H}}$ are the mode shapes representing the z-movements for the wing and the tail, respectively, $\nu_{\theta_{i_w}}$ is the mode shape representing θ angular displacement of the wing, and $\nu_{y_{i_V}}$ is the mode shape representing the y-displacement for the vertical tail.

4.1.4 The Effects of Rigid Body States and Controls on Generalized Force

From the generalized force in (4.27), a 2D lift of the wing can be derived as

$$\begin{aligned} l_w(y) &= C_{l_\alpha}(y)[\alpha(y) - \alpha_0 + \tau_e\delta_e + \tau_f\delta_f + \tau_{sp1}\delta_{sp1} + \tau_{sp2}\delta_{sp2}]q_\infty \bar{c}_w \\ &= C_{l_\alpha}(y)\left[\frac{w + py - q(x_{AC_w} - x_{CG})}{\|\mathbf{V}\|} + i_w + \left(\theta_E(y) + \frac{w_E(y)}{\|\mathbf{V}\|}\right) + \tau_e\delta_e + \tau_f\delta_f + \tau_{sp1}\delta_{sp1} + \tau_{sp2}\delta_{sp2}\right] q_\infty \bar{c}_w, \end{aligned} \tag{4.28}$$

where x_{AC_w} is the x coordinate of the local wing aerodynamic center, x_{CG} is the x coordinate of the center of gravity, $\theta_E(y)$ is the wing twist angle due the elastic effect, and $w_E(y)$ is the vertical speed due to elastic effect. The elastic effects on the 2D airfoil section is depicted in Figure 4.9. Substituting (4.28) into (4.27) yielding

$$\begin{aligned} Q_i = &-\int_{-b_w/2}^{b_w/2} C_{l_\alpha}(y)\left[\frac{w + py - q(x_{AC_w} - x_{CG})}{\|\mathbf{V}\|} + i_w + \tau_e\delta_e + \tau_f\delta_f + \tau_{sp1}\delta_{sp1}\right. \\ &\left. + \tau_{sp2}\delta_{sp2} + \left(\theta_E(y) + \frac{w_E(y)}{\|\mathbf{V}\|}\right)\right] q_\infty \bar{c}_w \nu_{z_i}(y)\, dy + Q_{i_{\text{H.Tail}}} + Q_{i_{\text{V.Tail}}}, \end{aligned} \tag{4.29}$$

where $Q_{i_{\text{H.Tail}}}$ and $Q_{i_{\text{V.Tail}}}$ are generalized force at the horizontal and vertical tail, respectively. Now the generalized force can be expanded in a Taylor series as

$$\begin{aligned} Q_i &= Q_{i_0} + Q_{i_u}u + Q_{i_\alpha}\alpha + Q_{i_\beta}\beta + Q_{i_p}p + Q_{i_q}q + Q_{i_r}r + \sum_{j=1}^{N} Q_{i_{\eta_j}}\eta_j + \sum_{j=1}^{N} Q_{i_{\dot{\eta}_j}}\dot{\eta}_j \\ &\quad + Q_{i_{\delta_e}}\delta_e + Q_{i_{\delta_f}}\delta_f + Q_{i_{\delta_r}}\delta_r + Q_{i_{\delta_a}}\delta_a + Q_{i_{\delta_{sp1}}}\delta_{sp1} + Q_{i_{\delta_{sp2}}}\delta_{sp2}, \end{aligned} \tag{4.30}$$

where Q_{i_0} is the force at trim condition, Q_{i_u}, Q_{i_α}, Q_{i_β}, Q_{i_p}, Q_{i_q}, and Q_{i_r} are the derivative of Q_i with respect to states u, α, β, p, q, and r, respectively, $Q_{i_{\eta_j}}$ and $Q_{i_{\dot{\eta}_j}}$ are the derivative of Q_i with respect to elastic states η_j and $\dot{\eta}_j$, and $Q_{i_{\delta_e}}$, $Q_{i_{\delta_f}}$, $Q_{i_{\delta_a}}$, $Q_{i_{\delta_r}}$, $Q_{i_{\delta_{sp1}}}$, and $Q_{i_{\delta_{sp1}}}$ are the derivative of Q_i with respect to control δ_e, δ_f, δ_a, δ_r, δ_{sp1}, and δ_{sp2}, respectively. Now that we include everybody parts, N represents the number of lumped mass or node of the whole airplane.

The effects of rigid body states and controls on generalized forces are in the form of these derivative, for example, Q_{i_α} can be derived by taking derivative of Q_i in (4.29) with respect to α (it is shown as $w/\|\mathbf{V}\|$ in (4.29)), yielding

$$Q_{i_\alpha} = -q_\infty \bar{c}_w \int_{-b_w/2}^{b_w/2} C_{l_{\alpha_w}}(y)\nu_{z_{i_w}}(y)\,dy - q_H \bar{c}_H \int_{-b_H/2}^{b_H/2} C_{l_{\alpha_H}}(y)\left(1 - \frac{d\epsilon}{d\alpha}\right)\nu_{z_{i_H}}(y)\,dy, \tag{4.31}$$

where $C_{l_{\alpha_w}}(y)$ and $C_{l_{\alpha_H}}(y)$ specify the 2D lift profile along the wing and horizontal tail and $\nu_{z_{i_w}}(y)$ and $\nu_{z_{i_H}}(y)$ are the i-th mode shape for the z-direction displacement for the wing and the horizontal tail. Similarly, the effects from the controls are derived by taking the derivative of Q_i in (4.29) with respect control deflections. For example,

$$Q_{i_{\delta_{sp2}}} = 2q_\infty \bar{c}_w \int_{y_{sp2_{in}}}^{y_{sp2_{out}}} C_{l_{\alpha_w}}(y)\tau_{sp2}\nu_{z_{i_w}}(y)\,dy, \tag{4.32}$$

assuming that spoiler set 2 deflect symmetrically from both wing. The rest of the derivatives in (4.30) can also be derived by the same method. In the equation above, $y_{sp2_{in}}$ and $y_{sp2_{out}}$ indicate the inboard and outboard locations of spoiler set 2 along the wing span. Therefore, the effects of the control deflections on the generalized force depend on the locations of where the surfaces are placed.

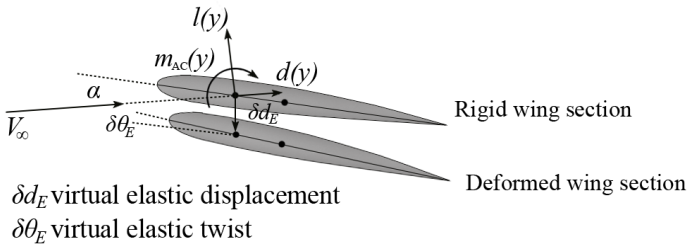


Figure 4.10: Forces, moments, and displacement of the 2D airfoil section.

4.1.5 Elastic Effects on Generalized Force

Structural movements of the aircraft can also affect its generalized forces. These effects are in the form of derivative $Q_{i_{\eta_j}}$ and $Q_{i_{\dot{\eta}_j}}$ that appear in (4.30). The elastic effects on the angle of attack α_E at the wing in (4.29) can be written as functions of mode shapes and modal coordinates as

$$\alpha_E(y) = \theta_E(y) + \frac{w_E(y)}{\|\mathbf{V}\|} = \left(\sum_{j=1}^{N} \nu_{\theta_{j_w}}(y)\eta_j + \frac{\Sigma_{j=1}^{N} \nu_{z_{j_w}}(y)\dot{\eta}_j}{\|\mathbf{V}\|} \right) \tag{4.33}$$

where $\nu_{\theta_{j_w}}(y)$ is the j-th mode shape for the wing torsion. The derivative $Q_{i_{\eta_j}}$ and $Q_{i_{\dot{\eta}_j}}$ can thus be derived by substituting (4.33) as well as similar expressions for the tails into (4.29) and differentiate Q_i with respect to η_j and $\dot{\eta}_j$, respectively.

4.1.6 Elastic Effects on Aerodynamic Forces and Moments

Structural movements such as wing bending or twisting can cause deviations of angle of attack or sideslip angle. Therefore, they can change the distribution of the aerodynamic forces and moments. Figure 4.10 demonstrates the elastically deformed wing section. We may write the local angle of attack of a wing section as

$$\alpha(y) = \alpha_R(y) + \theta_E(y) + \frac{w_E(y)}{\|\mathbf{V}\|} = \alpha_R(y) + \left(\sum_{j=1}^{N} \nu_{\theta_{j_w}}(y)\eta_j + \frac{\Sigma_{j=1}^{N} \nu_{z_{j_w}}(y)\dot{\eta}_j}{\|\mathbf{V}\|} \right) \tag{4.34}$$

where $\alpha_R(y)$ is the angle of attack of the undeformed rigid wing, $\theta_E(y)$ is the wing twist angle due the elastic effect, and $w_E(y)$ is the vertical speed due to elastic effect. From (4.8), the lift coefficient

of the wing can be written as

$$\begin{aligned} C_{L_w} &= C_{L_{\alpha_w}}(\alpha + i_w - \alpha_{0_w} + \tau_{\delta_f}\delta_f + \tau_{\delta_{sp1}}\delta_{sp1} + \tau_{\delta_{sp2}}\delta_{sp2}), \qquad (4.35) \\ &= \int_{-b_w/2}^{b_w/2} C_{l_{\alpha_w}}(y)\left[\left(\sum_{i=1}^{N}\nu_{\theta_{i_w}}(y)\eta_i + \frac{\Sigma_{i=1}^{N}\nu_{z_{i_w}}(y)\dot{\eta}_i}{\|\mathbf{V}\|}\right) + i_w(y) - \alpha_{0_w} + \tau_{\delta_f}(y)\delta_f + ...\right] dy. \end{aligned}$$

Therefore, the derivative of lift coefficient with respect to modal coordinates $C_{L_{\eta_i}}$ and $C_{L_{\dot{\eta}_i}}$ can be derived by differentiate equation (4.35) and similar expressions for the tails with respect to η_i and $\dot{\eta}_i$, yielding

$$C_{L_{\eta_i}} = \int_{-b_w/2}^{b_w/2} C_{l_{\alpha_w}}(y)\nu_{\theta_{j_w}}(y)\,dy + \int_{-b_H/2}^{b_H/2} C_{l_{\alpha_H}}(y)\left(\nu_{\theta_{j_H}}(y) - \frac{d\epsilon}{d\alpha}\nu_{\theta_{j_w}}(y)\right) dy \qquad (4.36)$$

and

$$C_{L_{\dot{\eta}_i}} = \frac{1}{\|\mathbf{V}\|}\int_{-b_w/2}^{b_w/2} C_{l_{\alpha_w}}(y)\nu_{z_{j_w}}(y)\,dy + \int_{-b_H/2}^{b_H/2} C_{l_{\alpha_H}}(y)\left(\nu_{z_{j_H}}(y) - \frac{d\epsilon}{d\alpha}\nu_{z_{j_w}}(y)\right) dy \qquad (4.37)$$

The elastic effects on drag are usually very small and neglected i.e. $C_{D_{\eta_i}} = C_{D_{\dot{\eta}_i}} = 0$ while $C_{S_{\eta_i}}$, $C_{S_{\dot{\eta}_i}}$, $C_{\hat{L}_{\eta_i}}$, $C_{\hat{L}_{\dot{\eta}_i}}$, $C_{M_{\eta_i}}$, $C_{M_{\dot{\eta}_i}}$, $C_{N_{\eta_i}}$, and $C_{N_{\dot{\eta}_i}}$ can be derived using the similar approach.

4.1.7 Full Flexible Nonlinear Model for Generic Transport Aircraft

From the GTA mode shapes provided by MSC/NASTRAN, we select the first 15 modes with the lowest frequencies ω_i for analysis. The first structural mode corresponds to the wing bending which has the frequency of 5.6Hz. With the approximation that the elastic displacements of aircraft are small and the practical mean-axis constraints are met, the rigid body and vibration mode are orthogonal.

Hence, the full flexible nonlinear model for GTA can then be written as

$$
\begin{aligned}
\dot{u} &= -qw + vr - g\sin\theta + \frac{F_{A_x} + F_{P_x}}{m}, \\
\dot{v} &= -ru + pw + g\cos\theta\sin\phi + \frac{F_{A_y} + F_{P_y}}{m}, \\
\dot{w} &= -pv + qu + g\cos\theta\cos\phi + \frac{F_{A_z} + F_{P_x}}{m}, \\
\dot{q} &= \frac{1}{I_{yy}}[(I_{zz} - I_{xx})pr + I_{xz}(r^2 - p^2) + M_A + M_P], \\
\begin{bmatrix} \dot{p} \\ \dot{r} \end{bmatrix} &= \left(\frac{1}{I_{xx}I_{zz} - I_{xz}^2}\right) \begin{bmatrix} I_{zz} & I_{xz} \\ I_{xz} & I_{xx} \end{bmatrix} \begin{bmatrix} I_{xz}pq + (I_{yy} - I_{zz})rq + \hat{L}_A + \hat{L}_P \\ -I_{xz}qr + (I_{xx} - I_{yy})pq + N_A + N_P \end{bmatrix}, \\
\ddot{\eta}_i + \zeta_i\omega_i\dot{\eta}_i + \omega_i^2\eta_i &= \frac{1}{\mathcal{M}_i}\Bigg(Q_{i_0} + Q_{i_u}u + Q_{i_\alpha}\alpha + Q_{i_\beta}\beta + Q_{i_p}p + Q_{i_q}q + Q_{i_r}r + \sum_{j=1}^{15} Q_{i_{\eta_j}}\eta_j + \\
&\quad \sum_{j=1}^{15} Q_{i_{\dot{\eta}_j}}\dot{\eta}_j + Q_{i_{\delta_e}}\delta_e + Q_{i_{\delta_f}}\delta_f + Q_{i_{\delta_r}}\delta_r + Q_{i_{\delta_a}}\delta_a + Q_{i_{\delta_{sp1}}}\delta_{sp1} + Q_{i_{\delta_{sp2}}}\delta_{sp2}\Bigg), i = 1, ..., 15
\end{aligned}
\tag{4.38}
$$

where the formulations of $F_{A_x}, F_{A_y}, F_{A_z}$ are described in (4.4) and they are functions of aerodynamic forces L, S, D. These forces and moments $\hat{L}_A, M_A, N_A$ may also be expressed as a function of rigid body states, elastic states, and controls, e.g.

$$
\begin{aligned}
L = q_\infty \bar{c}_w \Bigg(&C_{L_0} + C_{L_\alpha}\alpha + C_{L_{\dot{\alpha}}}\dot{\alpha} + C_{L_u}u + C_{L_q}q + \sum_{i=1}^{15} C_{L_{\eta_i}}\eta_i + \\
&\sum_{i=1}^{15} C_{L_{\dot{\eta}_i}}\dot{\eta}_i + C_{L_{\delta_e}}\delta_e + C_{L_{\delta_f}}\delta_f + C_{L_{\delta_{sp1}}}\delta_{sp1} + C_{L_{\delta_{sp2}}}\delta_{sp2}\Bigg),
\end{aligned}
$$

The formulation of $F_{P_x}, F_{P_y}, F_{P_z}, \hat{L}_P, M_P, N_P$ are described in (4.5) and (4.6) and kinematic equation (4.2) and (4.3) are also included. The system state thus becomes $\mathbf{x} = [u\ \alpha\ \beta\ p\ q\ r\ \theta\ \phi\ h\ \eta_1\ \dot{\eta}_1 ... \eta_{15}\ \dot{\eta}_{15}]^T$ and control $\mathbf{u} = [\delta_e\ \delta_f\ \delta_a\ \delta_r\ \delta_T\ \delta_{sp1}\ \delta_{sp2}]^T$. The nonlinear equation of motion (4.38) can be written into the form $\dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u})$.

4.1.8 Linearized Flexible Model for Generic Transport Aircraft

The nonlinear model of GTA (4.38) can be linearized around different flight conditions. For application in GLA systems, we focus our study on steady level flight where $\dot{u}_0 = \dot{v}_0 = \dot{w}_0 = \dot{p}_0 = \dot{q}_0 = \dot{r}_0 = \dot{\phi}_0 = \dot{\theta}_0 = \dot{\psi}_0 = \dot{h}_0 = \dot{\eta}_0 = \ddot{\eta}_0 = 0$ and $v_0 = p_0 = q_0 = r_0 = \phi_0 = \psi_0 = \delta_{f_0} = \delta_{a_0} = \delta_{r_0} = \delta_{sp1_0} = \delta_{sp2_0} = 0$.

The subscript "0" indicates the trim value of the corresponding quantity. By substituting these conditions into (4.38), we can solve for the trimmed surge velocity u_0, trimmed angle of attack $\alpha_0 = \frac{w_0}{u_0}$, trimmed pitch angle θ_0, trimmed elevator deflection δ_{e_0}, trimmed throttle setting δ_{T_0}, trimmed modal coordinates η_{i_0}, and trimmed generalized force Q_{i_0}. For GTA, we have $u_0 = 449.7$ ft/sec, $w_0 = 17.7$ ft/sec, $\alpha_0 = \theta_0 = 2.25$ deg, $\delta_{e_0} = -1$ deg, and $\delta_{T_0} = 0.159$ (η_{i_0} and Q_{i_0} are not shown).

By substituting (4.38), (4.2), and (4.3) with the values $u = u_0 + \delta u$, $v = \delta v$, $w = w_0 + \delta w$ and so on, where δu, δv, and δw indicate the perturbation from the trim condition, and use the small perturbation assumption (e.g. the product of perturbation quantities ≈ 0), we can write the linearized model of GTA about the steady level flight condition as

$$
\begin{aligned}
\dot{u} &= -w_0 q - (g\cos\theta_0)\theta + \frac{F_{A_x} + F_{P_x}}{m}, \\
\dot{v} &= -u_0 r + w_0 p + (g\cos\theta_0)\phi + \frac{F_{A_y} + F_{P_y}}{m}, \\
\dot{w} &= -u_0 q + (g\sin\theta_0)\theta + \frac{F_{A_z} + F_{P_x}}{m}, \\
\dot{p} &= \left(\frac{1}{I_{xx}I_{zz} - I_{xz}^2}\right)\left(I_{zz}(\hat{L}_A + \hat{L}_P) - I_{xz}(N_A + N_P)\right), \\
\dot{q} &= (M_A + M_P)/I_{yy}, \\
\dot{r} &= \left(\frac{1}{I_{xx}I_{zz} - I_{xz}^2}\right)\left(I_{xz}(\hat{L}_A + \hat{L}_P) - I_{xx}(N_A + N_P)\right), \\
\ddot{\eta}_i + \zeta_i\omega_i\dot{\eta}_i + \omega_i^2\eta_i &= \frac{1}{\mathcal{M}_i}\Bigg(Q_{i_u}u + Q_{i_\alpha}\alpha + Q_{i_\beta}\beta + Q_{i_p}p + Q_{i_q}q + Q_{i_r}r + \sum_{j=1}^{15} Q_{i_{\eta_j}}\eta_j + \sum_{j=1}^{15} Q_{i_{\dot{\eta}_j}}\dot{\eta}_j \\
&\quad + Q_{i_{\delta_e}}\delta_e + Q_{i_{\delta_f}}\delta_f + Q_{i_{\delta_r}}\delta_r + Q_{i_{\delta_a}}\delta_a + Q_{i_{\delta_{sp1}}}\delta_{sp1} + Q_{i_{\delta_{sp2}}}\delta_{sp2}\Bigg), i = 1, ..., 15
\end{aligned}
\tag{4.39}
$$

with the kinematic equations

$$
\begin{aligned}
\dot{\phi} &= p + (\tan\theta_0)r, \\
\dot{\theta} &= q, \\
\dot{\psi} &= (\sec\theta_0)r, \\
\dot{h} &= (u_0\cos\theta_0 + w_0\sin\theta_0)\theta + (\sin\theta_0)u - (cos\theta_0)w.
\end{aligned}
\tag{4.40}
$$

The state variables $u, v, w, p, q, r, \phi, \theta, \psi, h, \eta_i, \dot{\eta}_i$, forces and moments, F_{A_x}, F_{P_x}, F_{A_y}, F_{P_y}, F_{A_z}, F_{P_z},

$\hat{L}_A$, $\hat{L}_P$, M_A, M_P, N_A, N_P, and controls δ_e, δ_f, δ_a, δ_r, δ_{sp1}, δ_{sp2} in equation (4.39), (4.40) are in the perturbation form with the term "δ" omitted. While forces and moments are formulated similarly to that of nonlinear model, equation (4.39) and (4.40) maybe written into the form $\dot{\mathbf{x}} = A\mathbf{x} + B\mathbf{u}$.

To include the effects of the wind, let u_g, v_g, and w_g be the gust speed in x, y, and z directions, respectively, while $\dot{w}_g$ is the gust acceleration in z direction. By substituting the term u, v, w, and $\dot{w}$ with $u + u_g$, $v + v_g$, $w + w_g$, and $\dot{w} + \dot{w}_g$, respectively, (4.39) and (4.40) can be reformulated into

$$\dot{\mathbf{x}} = A\mathbf{x} + B\mathbf{u} + B_w\mathbf{w_g}, \tag{4.41}$$

where $\mathbf{w_g}$ represents the wind gust vector of the form $\mathbf{w_g} = [u_g\ \beta_g\ \alpha_g\ \dot{\alpha}_g]^T$, $\beta_g = v_g/\|\mathbf{V}\|$, $\alpha_g = w_g/\|\mathbf{V}\|$, and $\dot{\alpha}_g = \dot{w}_g/\|\mathbf{V}\|$. Equation (4.41) can be decoupled in longitudinal and lateral dynamics,

$$\dot{\mathbf{x}}_{\text{long}} = A_{\text{long}}\mathbf{x}_{\text{long}} + B_{\text{long}}\mathbf{u}_{\text{long}} + B_{w_{\text{long}}}\mathbf{w_g}_{\text{long}}, \tag{4.42}$$

$$\dot{\mathbf{x}}_{\text{lat}} = A_{\text{lat}}\mathbf{x}_{\text{lat}} + B_{\text{lat}}\mathbf{u}_{\text{lat}} + B_{w_{\text{lat}}}\mathbf{w_g}_{\text{lat}}, \tag{4.43}$$

where $\mathbf{x}_{\text{long}} = [u\ \alpha\ \theta\ q\ h\ \eta_1\ \dot{\eta}_1\ \eta_4\ \dot{\eta}_4\ \eta_5\ \dot{\eta}_5\ \eta_9\ \dot{\eta}_9\ \eta_{10}\ \dot{\eta}_{10}\ \eta_{12}\ \dot{\eta}_{12}\ \eta_{14}\ \dot{\eta}_{14}]^T$ is the longitudinal state, $\mathbf{x}_{\text{lat}} = [\beta\ \phi\ p\ r\ \eta_2\ \dot{\eta}_2\ \eta_3\ \dot{\eta}_3\ \eta_6\ \dot{\eta}_6\ \eta_7\ \dot{\eta}_7\ \eta_8\ \dot{\eta}_8\ \eta_{11}\ \dot{\eta}_{11}\ \eta_{13}\ \dot{\eta}_{13}\ \eta_{15}\ \dot{\eta}_{15}]^T$ is the lateral state, $\mathbf{u}_{\text{long}} = [\delta_e\ \delta_f\ \delta_T\ \delta_{sp1}\ \delta_{sp2}]^T$ is the longitudinal control, $\mathbf{u}_{\text{lat}} = [\delta_a\ \delta_r]^T$ is the lateral control, $\mathbf{w_g}_{\text{long}} = [u_g\ \alpha_g\ \dot{\alpha}_g]^T$ is the longitudinal gust input, and $\mathbf{w_g}_{\text{lat}} = \beta_g$ is the lateral gust input. The state space matrix A_{long}, B_{long}, $B_{w_{\text{long}}}$ can be found in the Appendix. It is noted that spoilers can be used asymmetrically to create lateral forces and moments. This is however beyond the scope of our work so they are not included in $\mathbf{u}_{\text{lat}}$.

4.2 Control System Setup for GLA

Gust load alleviation (GLA) is the aircraft system design that uses the control surface deflections to attenuate the load due to gust turbulence. This can be formulated as

$$\min_{\mathbf{u}(t)} J = \|\mathbf{y}_{\text{load}}(t)\|, \tag{4.44}$$

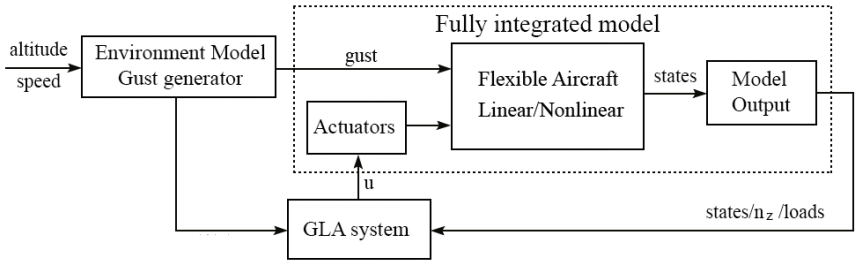


Figure 4.11: Block diagram of fully integrated model and gust model for the GLA design.

subject to

$$\dot{\mathbf{x}} = A\mathbf{x} + B\mathbf{u} + B_w\mathbf{w_g},$$

where $\mathbf{y}_{\text{load}}$ is the structural load vector that will be elaborated later in the section. The norm $\|\cdot\|$ can be chosen to be two norm or infinity norm based on the focus of the GLA study. The two norm is used to represent fatigue loads while the infinity norm is used for static loads. The GLA system can be drawn as a block diagram in Figure 4.11. Fully integrated model consists of the flexible GTA model developed in Section 4.1, actuator model, and output model. Actuators translate the control commands into control surfaces deflections. The output model represents the sensors and gyros readings as well as structural loads. The gust signal is generated by the gust generator function with the scale length that is based on the altitude and speed of the flight envelop.

4.2.1 Gust Types

We consider both discrete and continuous gust profiles as required by airworthiness regulation $CS-25$ [84]. The discrete gust is in the form of $(1-\cos)$ gust profile and can be written as

$$w_g(t) = \begin{cases} \frac{\bar{w}_g}{2}\left(1-\cos\frac{2\pi t}{L_g}\right) & 0 < t < L_g, \\ 0 & t \geq L_g, \end{cases} \tag{4.45}$$

where w_g is the vertical gust velocity, $\bar{w}_g$ is the gust amplitude, and L_g is the gust period. For continuous gust, we use the Dryden model to create stochastic gust excitations. It is modeled as a random Gaussian noise processes through the prescribed power spectral density. The Dryden gust

filter has the form

$$G_w(s) = \bar{w}_g \sqrt{\frac{L}{\pi \|\mathbf{V}\|}} \frac{1 + \sqrt{3}Ls/\|\mathbf{V}\|}{(1 + Ls/\|\mathbf{V}\|)^2}, \tag{4.46}$$

where L represents the gust scale length and $\|\mathbf{V}\|$ represents the vehicle flying speed. The transfer function $G_w(s)$ is used to generate the numerical values of the gust in the vertical direction $w_g(s) = G_w(s)n(s)$, where $n(s)$ is Gaussian noise. The gust then enters the GTA model (4.41) in the form of $\alpha_g(t) = w_g(t)/\|\mathbf{V}\|$. It is noted that if the trim speed is high, the gust energy absorbed to the aircraft will be less. The signal is tapered off at infinity in the time domain. Hence this is representative of real wind gusts acting on an aircraft passing through a turbulence.

4.2.2 Actuator Model

Actuators translate the control commands to control surface deflections. The actuator stroke limitations translate into lower and upper deflection limitations of the control surfaces. We select the deflection limits for the elevator, flaps, ailerons, and rudder to be ±30 degree while the spoilers can move from 0 to 30 degree. For MPC design, these limits will be included in the optimization routine for each control step. The limitation of oil flow through the valves results in maximum deflection rate of the control surfaces. This is modeled by a low pass filter. However, for MPC design this is included directly into optimization routine. The rate limits for elevator, rudder, and other surfaces are typically different. The elevator and rudder are larger surfaces and rate limits on hydraulic actuators are more restrictive. For GTA we choose the cut off frequency to be 60 rad/sec for elevator and rudder and 75 rad/sec for flaps, ailerons, and spoilers.

4.2.3 Output Function

For the GLA design, the model output $\mathbf{y}$ should contain the quantities of interest that need to be optimized. First of which will be standard measurements such as systems states and vertical accelerations around the critical points. The second virtual output contains other quantities that are important for the GLA design but usually not measured. These will be the load: bending moment, torsion moment, and shear, around different aircraft station. For GTA, the output vector for longitudinal motion will

have the form

$$\mathbf{y}_{\text{long}} = [u\ \alpha\ \theta\ q\ h\ n_{z_{\text{CP}}}\ n_{z_{\text{CG}}}\ n_{z_{\text{Tail}}}\ WR_b\ WR_t\ MW_b\ MW_t\ TR_b\ TR_t]^T. \tag{4.47}$$

The first five outputs are the system states. The term $n_{z_{\text{CP}}}$, $n_{z_{\text{CG}}}$, and $n_{z_{\text{Tail}}}$ represent vertical accelerations at the cockpit, the CG, and the tail section, respectively. The term WR_b, MW_b, and TR_b represent bending moment loads at the wing root, the mid-wing station, and the horizontal tail root, respectively. Similarly, the term WR_t, MW_t, and TR_t represent torsion moment loads at the wing root, the mid-wing station, and the horizontal tail root, respectively. Noted that the aircraft is normally designed to withstand very high shear forces so they are not included in this study. The load stations for the GTA is depicted in Figure 4.5.

Vertical Acceleration

Vertical accelerations at different airplane stations maybe measured by accelerometers. We model this signal using

$$n_z(x, y, z, t) = n_{z_R}(x, y, z, t) + \sum_{i=1}^{15} \nu_{z_i}(x, y, z)\ddot{\eta}_i(t), \tag{4.48}$$

where $n_z(x, y, z, t)$ represents vertical acceleration at location (x, y, z) on the fuselage reference axis system and $n_{z_R}(x, y, z, t)$ represents the rigid body component of the acceleration. The rigid body acceleration $\mathbf{a}_R = [n_{x_R}, n_{y_R}, n_{z_R}]^T$ at position $\mathbf{p} = [x, y, z]^T$ can be derived from

$$\mathbf{a}_R = \frac{d\mathbf{V}}{dt} + (\omega \times \mathbf{V}) + (\omega \times \omega \times \mathbf{p}) + (\frac{d\omega}{dt} \times \mathbf{p}). \tag{4.49}$$

For steady level flight condition, the term n_z along the body x-axis can be simplified to

$$n_z(x, t) = \dot{w} - u_0 q - x\dot{q} + \sum_{i=1}^{15} \nu_{z_{if}}(x)\ddot{\eta}_i(t), \tag{4.50}$$

where $\nu_{z_{if}}(x)$ represents mode shape of the z-direction displacement for the fuselage and $n_z(x_{\text{CP}}, t)$, $n_z(x_{\text{CG}}, t)$, and $n_z(x_{\text{Tail}}, t)$ represent the vertical acceleration at the cockpit, the CG, and the tail respectively.

Load Calculation

From the lumped mass wing model in Figure 4.7, we can see that the wing root bending moment is $WR_b(t) = k_1\theta_1(t)$ which is the first entry of the vector $[\mathbf{K}]\mathbf{q}(t) = [\mathbf{K}][\mathbf{\Phi}]\eta(t)$. For the full GTA structural beam model, the loads (bending or torsion) at different locations may be found by picking the appropriate row of the matrix $[\mathbf{K}][\mathbf{\Phi}]$ multiplied by modal coordinate $\eta(t)$. This can be formulated as

$$\mathbf{y}_{\text{load}} = C_{\text{load}}\mathbf{x},$$

where C_{load} represents the appropriate row of the matrix $[\mathbf{K}][\mathbf{\Phi}]$.

After the vertical acceleration and loads have been calculated, the output vector $\mathbf{y}_{\text{long}}$ maybe formulated as

$$\mathbf{y}_{\text{long}} = C_{\text{long}}\mathbf{x}_{\text{long}} + D_{\text{long}}\mathbf{u}_{\text{long}} + D_{w_{\text{long}}}\mathbf{w}_{\mathbf{g}_{\text{long}}}.$$

For the GLA design, we want to reduce the load at the wing root because wing mass is susceptible to this quantity, especially the bending moment. The aspect of static load reduction is reflected by the peak value or $\mathcal{L}_\infty$ norm while the aspect of fatigue load reduction can be captured by the $\mathcal{L}_2$ norm. Both $\|WR_b(t)\|_\infty$ and $\|WR_b(t)\|_2$ will be used as a major criteria for assessing the GLA system.

4.3 Model Predictive Control for GLA

Model Predictive Control (MPC), or receding horizon control, is a discrete method based on repeated online optimization. We calculate the control signal by performing a constrained optimization over a finite prediction horizon indicated by the number of future control steps N_p at each sampling time. From the resulting sequence of optimal control vectors, only the first vector is applied. The optimization is repeated at the next time step when a new measurement is available. The high performance computing along with fast and reliable quadratic programming techniques allow us to apply MPC to problems with fast dynamics such as the GLA.

The advantages of MPC are the ability to work with nonlinear or time-varying systems as well as the case with control constraints. The successive linearization maybe performed for the nonlinear

aircraft dynamics, however the GLA problem only focuses on the aircraft motions around the steady trimmed flight condition so we will use the linearized model of the GTA (4.41) for the control design.

Since MPC is a discrete control method, the state space representation in (4.41) must be discretized to a discrete version A^D, B^D and B_w^D. For the cleanliness of the expressions in our optimization models, the superscript "D" will be omitted from this point forward. The MPC for GLA systems at time step k can be formulated as

$$\min_{\tilde{\mathbf{u}}} J(\tilde{\mathbf{u}}, \mathbf{x}_k, \tilde{\mathbf{w}}_g) = \sum_{i=k}^{k+N_p} \mathbf{x}_i^T Q_i \mathbf{x}_i + \mathbf{u}_i^T R_i \mathbf{u}_i, \tag{4.51}$$

subject to

$$\begin{aligned} \mathbf{x}_{i+1} &= A\mathbf{x}_i + B\mathbf{u}_i + B_w \mathbf{w}_{g_i}, i = k, ..., k+N_p \\ \mathbf{u}_{\min} &\leq \mathbf{u}_i \leq \mathbf{u}_{\max}, \\ \Delta\mathbf{u}_{\min} &\leq \Delta\mathbf{u}_i \leq \Delta\mathbf{u}_{\max}, \end{aligned}$$

where $\mathbf{x}_k$, $\mathbf{u}_k$, and $\mathbf{w}_{g_k}$ are the state, control input, and wind gust vectors, respectively, at time step k. The term $\mathbf{u}_{\min}$ and $\mathbf{u}_{\max}$ are the control deflection constraints while $\Delta\mathbf{u}_{\min}$ and $\Delta\mathbf{u}_{\max}$ are the control rate constraints. The matrices Q_k and R_k are the state and control weighting matrices, respectively. The term $\mathbf{x}_i^T Q_i \mathbf{x}_i$ in the cost function indicates the desire for the model-predictive controller to minimize the state quantities to stay close to the trimmed conditions along the prediction horizon. For the design purpose, the term $\mathbf{x}_i^T Q_i \mathbf{x}_i$ can be formulated such that the loads of interest are in the quadratic form as

$$\mathbf{x}_i^T Q_i \mathbf{x}_i = \begin{bmatrix} \mathbf{x}_i & \mathbf{y}_{\text{load}_i} \end{bmatrix} \begin{bmatrix} Q_{\text{trim}_i} & 0 \\ 0 & Q_{\text{load}_i} \end{bmatrix} \begin{bmatrix} \mathbf{x}_i \\ \mathbf{y}_{\text{load}_i} \end{bmatrix} \tag{4.52}$$

where $Q_i = Q_{\text{trim}_i} + C_{\text{load}}^T Q_{\text{load}_i} C_{\text{load}}$. Therefore the design is done by the selections of weighting matrices $Q_{\text{trim}_i} \geq 0$, $Q_{\text{load}_i} \geq 0$ and $R_i > 0$.

The vector $\tilde{\mathbf{u}} = [\mathbf{u}_k; \mathbf{u}_{k+1}; ...; \mathbf{u}_{k+N_p}]$ is the control signal sequence over the prediction horizon N_p. Similarly, we can write $\tilde{\mathbf{w}}_g = [\mathbf{w}_{g_k}; \mathbf{w}_{g_{k+1}}; ...; \mathbf{w}_{g_{k+N_p}}]$ as the "preview" gust information along the prediction horizon. The state sequence $\tilde{\mathbf{x}} = [\mathbf{x}_k; \mathbf{x}_{k+1}; ...; \mathbf{x}_{k+N_p}]$ can be expressed as a function of

future control sequence $\tilde{\mathbf{u}}$, the current state $\mathbf{x}_k$, and future gust information $\tilde{\mathbf{w}}_g$ as

$$\tilde{\mathbf{x}} = \tilde{C}\tilde{\mathbf{u}} + \tilde{D}_1\mathbf{x}_k + \tilde{D}_2\tilde{\mathbf{w}}_g \tag{4.53}$$

where

$$\begin{aligned}
\tilde{C} &= \begin{bmatrix} 0 & & & & \emptyset \\ B & 0 & & & \\ AB & B & 0 & & \\ \vdots & \vdots & \ddots & 0 & \\ A^{N_p-2}B & A^{N_p-3}B & \dots & B & 0 \end{bmatrix}, \tilde{D}_1 = \begin{bmatrix} I \\ A \\ A^2 \\ \vdots \\ A^{N_p-1} \end{bmatrix}, \\
\tilde{D}_2 &= \begin{bmatrix} 0 & & & & \emptyset \\ B_w & 0 & & & \\ AB_w & B_w & 0 & & \\ \vdots & \vdots & \ddots & 0 & \\ A^{N_p-2}B_w & A^{N_p-3}B_w & \dots & B_w & 0 \end{bmatrix},
\end{aligned} \tag{4.54}$$

where $\tilde{C}\tilde{\mathbf{u}}$ represents the forced term due to the control sequence $\tilde{\mathbf{u}}$, $\tilde{D}_1\mathbf{x}_k$ is the unforced term due to initial condition $\mathbf{x}_k$, and $\tilde{D}_2\tilde{\mathbf{w}}_g$ is the unforce term due to the gust sequence $\tilde{\mathbf{w}}_g$. The preceding constraints can also be expressed as

$$\tilde{E}\tilde{\mathbf{u}} \leq \tilde{F} \tag{4.55}$$

where

$$\tilde{E} = \begin{bmatrix} I_{N_p} \otimes I \\ -I_{N_p} \otimes I \\ \breve{E} \\ -\breve{E} \end{bmatrix}, \breve{E} = \begin{bmatrix} I & & & \emptyset \\ -I & I & & \\ & \ddots & \ddots & \\ \emptyset & & -I & I \end{bmatrix}, \tilde{F} = \begin{bmatrix} \mathbf{1}_{N_p} \otimes \mathbf{u}_{\max} \\ -\mathbf{1}_{N_p} \otimes \mathbf{u}_{\min} \\ \mathbf{1}_{N_p} \otimes \Delta\mathbf{u}_{\max} + \breve{F} \\ -\mathbf{1}_{N_p} \otimes \Delta\mathbf{u}_{\min} - \breve{F} \end{bmatrix}, \breve{F} = \begin{bmatrix} \mathbf{u}_{k-1} \\ \emptyset \\ \vdots \end{bmatrix}, \tag{4.56}$$

where I represents the identity matrix of the size equals to the number of control inputs (for GTA it is 7, or for $\mathbf{u}_{\text{long}}$ of GTA, it is 5), $\mathbf{1}_{N_p}$ is the one's vector of size N_p, and $\mathbf{u}_{k-1}$ is the control vector of

the previous time step. Using (4.53) and (4.55), the MPC at time step k (4.51) becomes a constrained optimization problem

$$\min_{\tilde{\mathbf{u}}} J(\tilde{\mathbf{u}}, \mathbf{x}_k, \tilde{\mathbf{w}}_g) = \tilde{\mathbf{u}}^T[\tilde{C}^T\tilde{Q}\tilde{C} + \tilde{R}]\tilde{\mathbf{u}} + 2[\mathbf{x}_k^T\tilde{D}_1^T + \tilde{\mathbf{w}}_g^T\tilde{D}_2^T]\tilde{Q}\tilde{C}\tilde{\mathbf{u}}, \tag{4.57}$$

subject to

$$\tilde{E}\tilde{\mathbf{u}} \leq \tilde{F},$$

where $\tilde{Q} = \text{diag}[Q, Q, ..., Q_f]$ and $\tilde{R} = \text{diag}[R, R, ..., R]$ or $Q_k = Q_{k+1} = ... = Q_{k+N_p-1} = Q$ and $R_k = R_{k+1} = ... = R_{k+N_p}$. For a positive semi-definite Q and and positive definite R, it follows that $\tilde{C}^T\tilde{Q}\tilde{C} + \tilde{R}$ is positive definite and the optimization problem (4.57) becomes a convex quadratic programming (QP) problem with a unique solution $\tilde{\mathbf{u}}^* = [\mathbf{u}_k^*; \mathbf{u}_{k+1}^*; ...; \mathbf{u}_{k+N_p}^*]$. Only the first input $\mathbf{u}_k^*$ will be applied to the aircraft and finally the optimization is repeated at the next time instant $k+1$ with a shifted horizon forward. It is noted that at each time step k, the state $\mathbf{x}_k$ along with environment signal $\tilde{\mathbf{w}}_g$ (a preview of gust information) are used to calculate the control $\mathbf{u}_k^*$, hence the procedure is a form of feedback control.

4.3.1 Stability of MPC

Unlike infinite-horizon linear quadratic optimal control problem (LQR), the finite horizon MPC is not always asymptotically stable. The stability of finite-horizon MPC problems have been extensively studied, and the techniques to ensure the stability of these systems are stated in the following lemma.

Lemma 4.3.1 *A linear quadratic MPC is stable if and only if*

1. *the pair (A, B) is stabilizable,*
2. *the pair $(A, Q^{1/2})$ is observable, and*
3. *the cost function satisfies the cost monotonicity condition i.e. $J(\tilde{\mathbf{u}}^*, \mathbf{x}_k, \tilde{\mathbf{w}}_g, N+1) \leq J(\tilde{\mathbf{u}}^*, \mathbf{x}_k, \tilde{\mathbf{w}}_g, N)$.*

Proof. See [96, 97]. ■

Lemma 4.3.2 *The cost monotonicity condition can be guaranteed using terminal weighting while the*

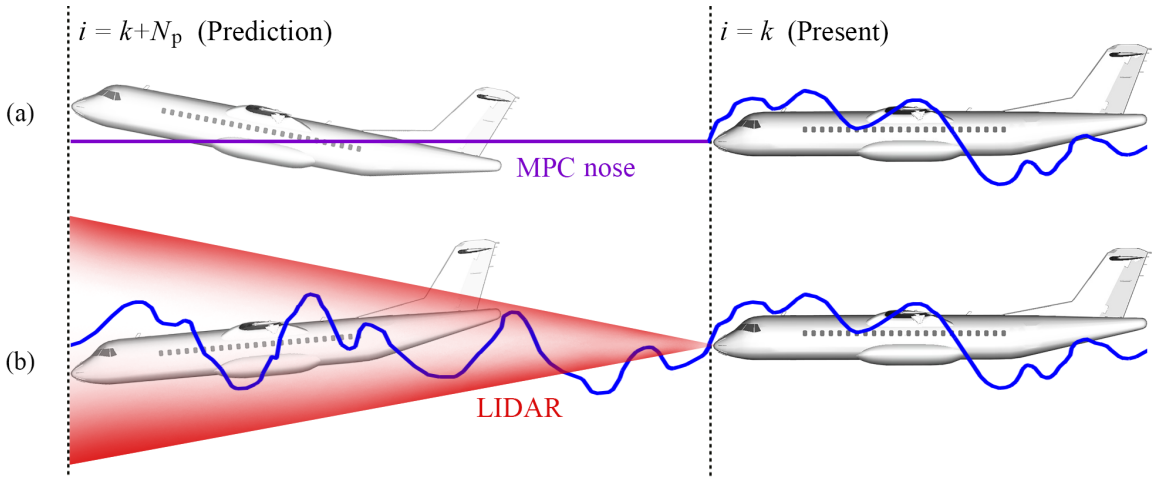


Figure 4.12: (a) MPC with nose measurement, (b) MPC with LIDAR system

penalty function satisfies

$$Q_f \geq Q + K^T R K + (A - BK)^T Q_f (A - BK) \quad \textit{for some } K \in \mathbb{R}^{m \times n}, \tag{4.58}$$

where m and n are dimensions of state and control vector respectively.

Proof. See [98]. ■

For our application, the stability of MPC can be guaranteed by picking the terminal weighting Q_f according to (4.58) while K is chosen to be the LQR gain.

4.4 The Improved MPC for GLA

In this section, we introduce two methods to improve the performance of the GLA system. Since the GLA controller deals with very fast dynamics, we can refine the prediction process by the enhanced sensing capability via the preview of gust information. The other form of improvement comes from the actuation side, where the flap-array is introduced to help reducing the effects of gust onto the different vibration modes.

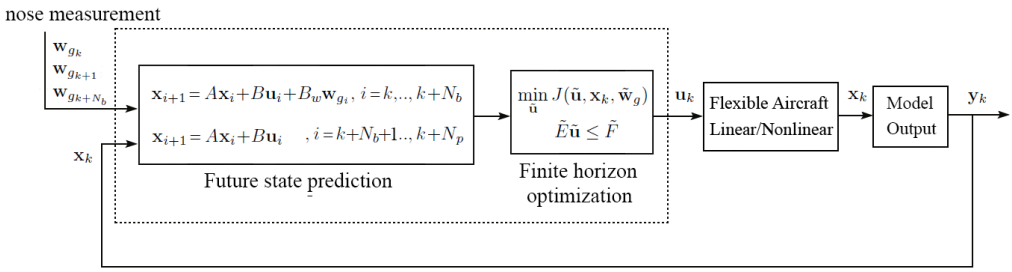


Figure 4.13: Block Diagram of the MPC with nose measurement.

4.4.1 MPC with Preview Gust Information

It is shown that the MPC formulation in (4.51) or (4.57) allows us to include the preview of gust information $\tilde{\mathbf{w}}_g = [\mathbf{w}_{g_k}; \mathbf{w}_{g_{k+1}}; ...; \mathbf{w}_{g_{k+N_p}}]$ to help predicting the future states and counteract disturbances before they arrive at the aircraft. We consider two MPC scenarios as follows:

Aircraft with Nose Gust Sensor Measurement

The gust is measured and stored at the current time step k while the gust in front of the aircraft nose is still unknown and assume to be zero. In this scenario, the preview gust sequence that enters the aircraft will have the form $\tilde{\mathbf{w}}_g = [\mathbf{w}_{g_k}; \mathbf{w}_{g_{k+1}}; ...; \mathbf{w}_{g_{k+N_b}}; 0; ...; 0]$ where N_b is the number of time steps the gust takes to travel from the nose to the location where it affects the aircraft dynamics (4.41). For the GTA, this is the number of steps the gust takes to travel from the nose to the CG. Figure 4.12a depicts the MPC with nose measurement. The block diagram of this method is shown in Figure 4.13.

Aircraft with Gust Measurement via LIDAR

Light Detection And Ranging (LIDAR) system allows us to acquire gust speed information ahead of the aircraft nose. We choose the look-ahead distance for GTA with LIDAR to be 67.5 meters. The prediction horizon N_p is also defined such that over the whole horizon, the gust shape is known. This means that the vector $\tilde{\mathbf{w}}_g$ contains the accurate gust information from $\mathbf{w}_{g_k}$ to $\mathbf{w}_{g_{k+N_p}}$. Figure 4.12b demonstrates the predicted gust measurement for MPC with the LIDAR system. The block diagram of this method is shown in Figure 4.14.

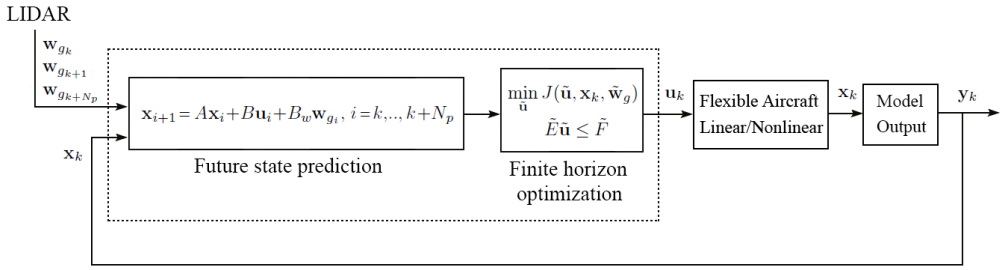


Figure 4.14: Block Diagram of the MPC with LIDAR.

4.4.2 Flap-Array

The goal of the GLA system is to reduce the load that the aircraft structure is subjected to in flight. The most critical load station is the wing root because the wing mass is susceptible to it. As mentioned in Section 4.2.3, the peak and fatigue loads can be expressed in forms of $\mathcal{L}_\infty$ and $\mathcal{L}_2$ norms, respectively, e.g., $\|WR_b(t)\|_\infty$ or $\|WR_b(t)\|_2$. In discrete time domain approach such as MPC, the reductions of these terms reflect in the cost function $J(\tilde{\mathbf{u}}^*, \mathbf{x}_k, \tilde{\mathbf{w}}_g)$ while we try to minimize the output sequence $\|\tilde{\mathbf{y}}(t)\|_2$ over the prediction horizon. It is then possible to reduce the wing load further by adding flexibility to the wing controls. This maybe done by adding "*flap-array*" to the wing control surfaces. A flap-array composes of small trailing edge flaps throughout the entire span of the wing as depicted in Figure 4.15. It can be used in conjunction with distributed spoilers where we similarly divided them evenly throughout the wingspan. This opens up the opportunity to alter the wing lift and drag distributions more delicately. For the GTA, we first choose to add five separate flaps that span evenly from the wing root to the location of the aileron. Each of these flaps takes 10% of the wing chord. Both set of spoilers are also divided into two pieces for the total of four spoilers per each wing.

The introduction of new flaps and spoilers will change both rigid body and structural dynamics. For rigid body, the lift force maybe composed as

$$\begin{aligned} L &= q_\infty \bar{c}_w \left(C_{L_0} + C_{L_\alpha}\alpha + C_{L_{\dot{\alpha}}}\dot{\alpha} + C_{L_u}u + C_{L_q}q + \sum_{i=1}^{15} C_{L_{\eta_i}}\eta_i + \sum_{i=1}^{15} C_{L_{\dot{\eta}_i}}\dot{\eta}_i + C_{L_{\delta_e}}\delta_e \right. \\ &\quad \left. + \sum_{i=1}^{5} C_{L_{\delta_{f_i}}}\delta_{f_i} + \sum_{i=1}^{4} C_{L_{\delta_{sp_i}}}\delta_{sp_i} \right), \end{aligned}$$

where the lift coefficient derivatives with respect to the new flaps and spoilers are derived similarly

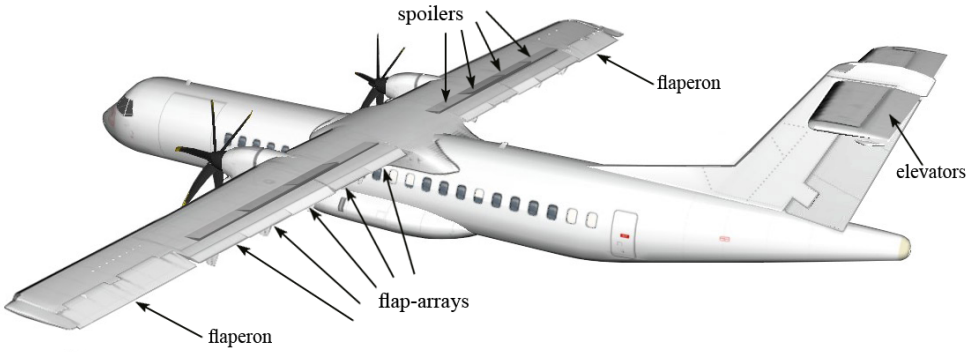


Figure 4.15: Flap-arrays and distributed spoilers

to (4.8), e.g., $C_{L_{\delta_{f3}}} = C_{L_{\alpha_w}} \tau_{\delta_{f3}}$. The effects on the drag are also added similarly. For structural dynamics, the controls can affect the generalized force as

$$\begin{aligned} Q_i &= Q_{i_0} + Q_{i_u} u + Q_{i_\alpha} \alpha + Q_{i_\beta} \beta + Q_{i_p} p + Q_{i_q} q + Q_{i_r} r + \sum_{j=1}^{15} Q_{i_{\eta_j}} \eta_j + \sum_{j=1}^{15} Q_{i_{\dot{\eta}_j}} \dot{\eta}_j \\ &+ Q_{i_{\delta_e}} \delta_e + Q_{i_{\delta_r}} \delta_r + Q_{i_{\delta_a}} \delta_a + \sum_{j=1}^{5} Q_{i_{\delta_{f_j}}} \delta_{f_j} + \sum_{j=1}^{4} Q_{i_{\delta_{sp_j}}} \delta_{sp_j}, i = 1, ..., 15, \end{aligned}$$

where the generalized force derivatives with respect to the new flaps and spoilers are derived similarly to (4.32). It is shown that the effects of the controls on the generalized force depends on the location of where the surfaces are placed. Each flap deflection contributes to the generalized force excitation to various modes differently. At each time step, the MCP will solve for the control commands to which the combined generalized forces yielding the minimal loads.

This alteration of the aircraft model changes the state space representation from (A, B, B_w, C, D, D_w) to $(A, B^{\text{new}}, B_w, C, D^{\text{new}}, D_w)$ while the new control vector is

$$\mathbf{u}^{\text{new}} = [\delta_e \ \delta_a \ \delta_r \ \delta_T \ \delta_{f1} \ \delta_{f2} \ \delta_{f3} \ \delta_{f4} \ \delta_{f5} \ \delta_{sp1} \ \delta_{sp2} \ \delta_{sp3} \ \delta_{sp4}]^T. \tag{4.59}$$

Similar to (4.53), B^{new}, D^{new}, and $\mathbf{u}^{\text{new}}$ lead to the new state sequence vector

$$\tilde{\mathbf{x}}^{\text{new}} = \tilde{C}^{\text{new}} \tilde{\mathbf{u}}^{\text{new}} + \tilde{D}_1 \mathbf{x}_k + \tilde{D}_2 \tilde{\mathbf{w}}_g, \tag{4.60}$$

where $\tilde{C}^{\text{new}}$ is expressed similarly to (4.54) with the substitution of B with B^{new} and D with D^{new}.

The preceding constraint is also altered where $\tilde{E}^{\text{new}}$, $\breve{E}^{\text{new}}$, $\tilde{F}^{\text{new}}$, $\breve{F}^{\text{new}}$ have the similar formulation as (4.56) with the appropriate size of the identity matrix I and control constraints $\mathbf{u}_{\min}^{\text{new}}$, $\mathbf{u}_{\max}^{\text{new}}$, $\Delta\mathbf{u}_{\min}^{\text{new}}$, and $\Delta\mathbf{u}_{\max}^{\text{new}}$. The deflection limits for flap-arrays are chosen to be ± 15 degrees whereas deflection rate limits are 75 rad/sec. The updated MPC at time step k can be then be written as

$$\min_{\tilde{\mathbf{u}}^{\text{new}}} J(\tilde{\mathbf{u}}^{\text{new}}, \mathbf{x}_k, \tilde{\mathbf{w}}_g) = \tilde{\mathbf{u}}^{\text{new}^T}[\tilde{C}^{\text{new}^T}\tilde{Q}\tilde{C}^{\text{new}} + \tilde{R}^{\text{new}}]\tilde{\mathbf{u}}^{\text{new}} + 2[\mathbf{x}_k^T\tilde{D}_1^T + \tilde{\mathbf{w}}_g^T\tilde{D}_2^T]\tilde{Q}\tilde{C}^{\text{new}}\tilde{\mathbf{u}}^{\text{new}}, \quad (4.61)$$

subject to

$$\tilde{E}^{\text{new}}\tilde{\mathbf{u}}^{\text{new}} \leq \tilde{F}^{\text{new}},$$

where $\tilde{R}^{\text{new}} = \text{diag}[R^{\text{new}}, R^{\text{new}}, ..., R^{\text{new}}]$ is the new control weighting matrix that includes the flap-arrays and additional spoilers.

4.5 Simulation Results

We now apply the model predictive controller (4.51) to the linearized flexible GTA model developed in Section 4.1.8. The model parameters are given in table 4.1 and the corresponding longitudinal state space representation is given in Appendix B. The control limitations for the elevator and flap (or the aileron deflects in the same direction) are chosen to be ± 30 degrees. The spoiler has deflection range from 0 to 30 degree. The deflection rate for spoiler and flap are chosen to be 75 rad/sec while the deflection rate for elevator is 60 rad/sec. The throttle setting will not be utilized for GLA design thus the second column of B_{long} and D_{long} will be eliminated. The output weighting matrix Q has the structure

$$Q = Q_{\text{trim}} + C_{\text{load}}^T Q_{\text{load}} C_{\text{load}} \quad (4.62)$$

where

$$\begin{aligned} Q_{\text{trim}} &= \text{diag}[Q_u, Q_\alpha, Q_\theta, Q_q, Q_h], \text{ and} \quad (4.63) \\ Q_{\text{load}} &= \text{diag}[Q_{WRb}, Q_{WRt}, Q_{MWb}, Q_{MWt}, Q_{Tb}, Q_{Tt}], \end{aligned}$$

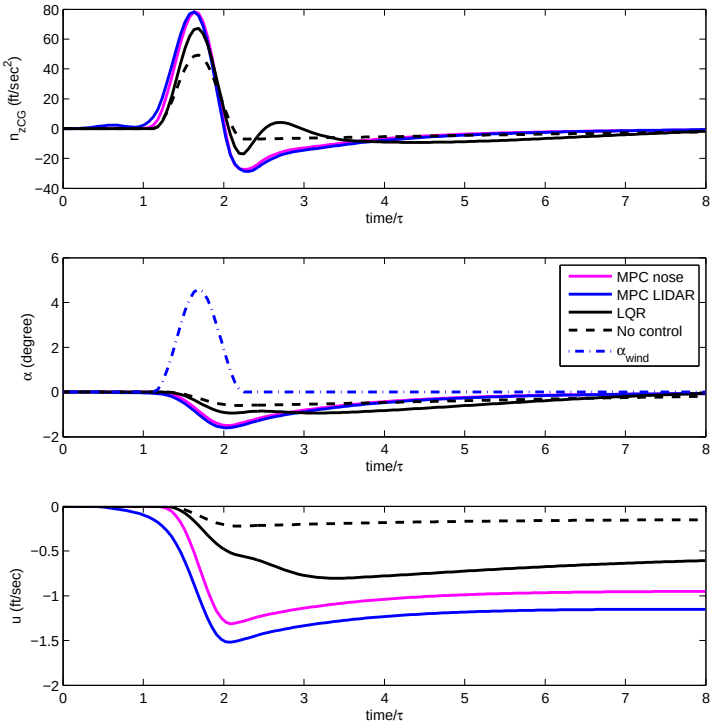


Figure 4.16: From top to bottom: the vertical acceleration, angle of attack, and speed responses to the discrete up-gust comparing the case without GLA, the LQR controller, the MPC with nose measurement, and the MPC with LIDAR.

where we choose the state weighting to be $Q_u = Q_\alpha = Q_\theta = Q_q = Q_h = 10^5$, and the load weighting to be $Q_{WRb} = Q_{WRt} = Q_{MWb} = Q_{MWt} = Q_{Tb} = Q_{Tt} = 0.1$. The state and the load weighting are different in scale because the matrix C_{load} contains large numbers. The control weighting R also has the form

$$R = \text{diag}[R_{\delta_e}, R_{\delta_f}, R_{\delta_{sp1}}, R_{\delta_{sp1}}], \tag{4.64}$$

and we choose $R = 1.6(10^4)I_4$. In order to guarantee the stability of MPC method, we pick the terminal weighting Q_f according to (4.58) by choosing K to be an LQR gain.

We will explore the GLA responses to two different gust types required by the airworthiness regulation $CS-25$ [84] as described in Section 4.2.1. For discrete gust we will look into "up" gust

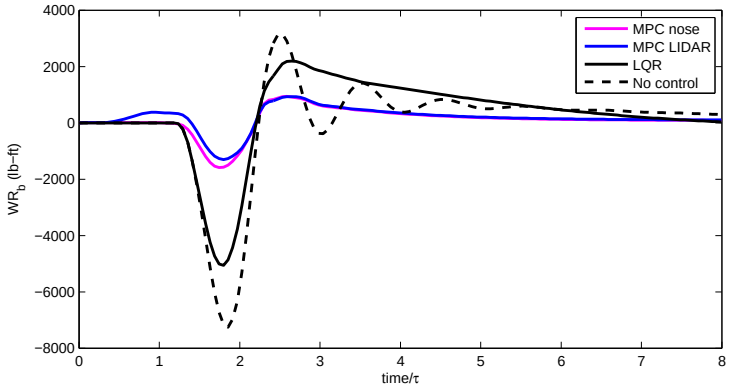


Figure 4.17: The wing root bending moment response to the discrete up-gust comparing the case without GLA, the LQR controller, the MPC with nose measurement, and the MPC with LIDAR.

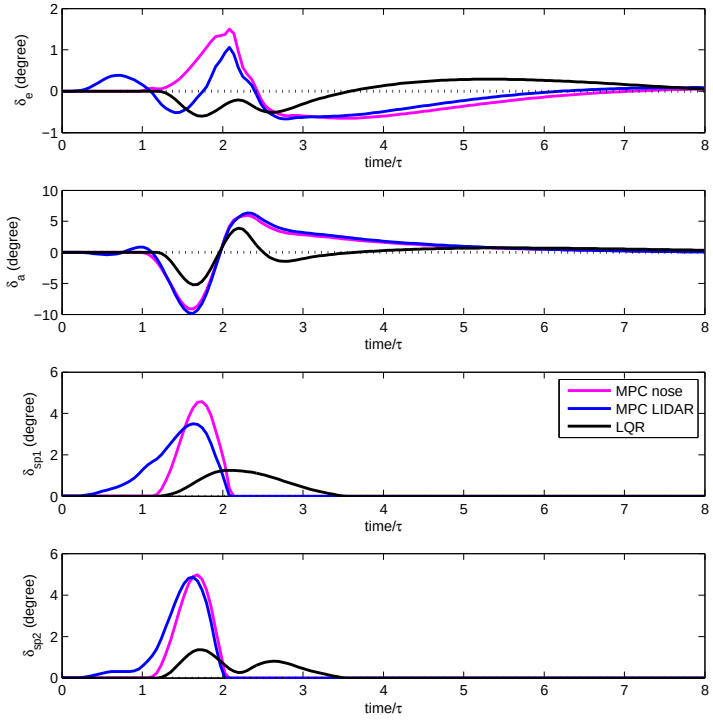


Figure 4.18: From top to bottom: the elevator, aileron, spoiler set 1, and spoiler set 2 responses to the discrete up-gust comparing the case with LQR controller, the MPC with nose measurement, and the MPC with LIDAR.

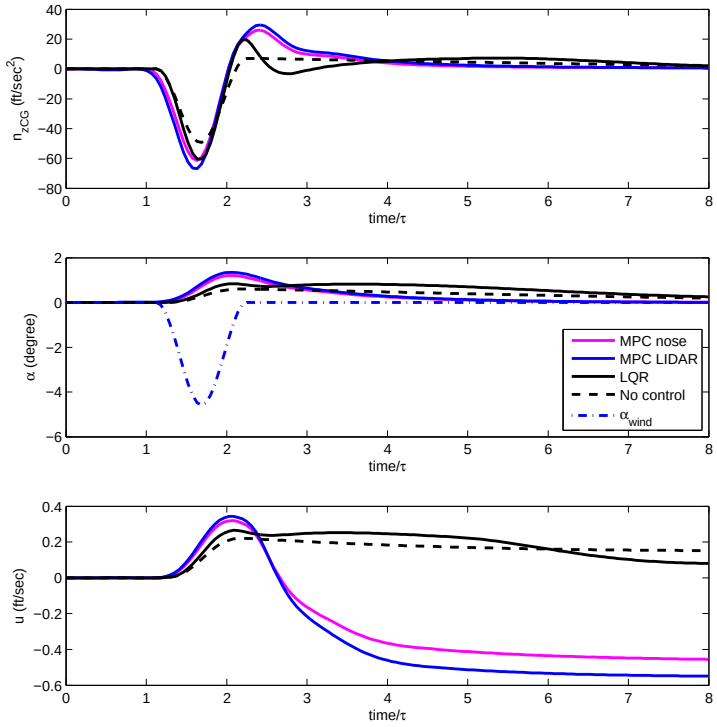


Figure 4.19: From top to bottom: the vertical acceleration, angle of attack, and speed responses to the discrete down-gust comparing the case without GLA, the LQR controller, the MPC with nose measurement, and the MPC with LIDAR.

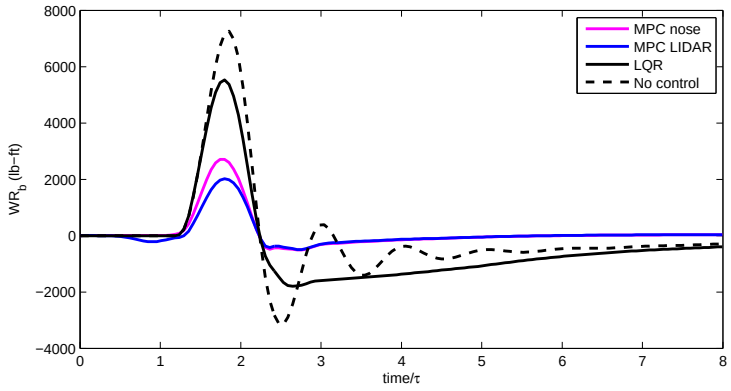


Figure 4.20: The wing root bending moment response to the discrete down-gust comparing the case without GLA, the LQR controller, the MPC with nose measurement, and the MPC with LIDAR.

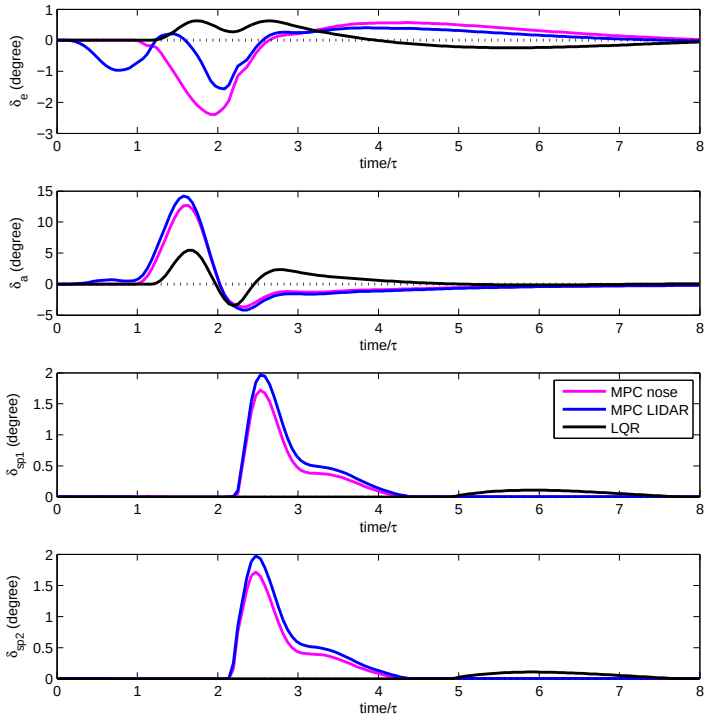


Figure 4.21: From top to bottom: the elevator, aileron, spoiler set 1, and spoiler set 2 responses to the discrete down-gust comparing the case with LQR controller, the MPC with nose measurement, and the MPC with LIDAR.

as well as "down" gust, then we will proceed to see the continuous gust responses.

Figures 4.16-4.18 show the results of the GTA flying through a discrete up-vertical gust (4.45) with gust length $L_g = 0.34$ sec and amplitude $\bar{w}_g = 18$ ft/sec. The plots compare the case when no GLA is applied, the LQR controller, the MPC with nose measurement, and the MPC with LIDAR. In LQR controller here, we impose the control constraints after the control of each step is computed. In order to generalize the gust response characteristic to that of the larger aircraft, the time scale for these plots are normalized by the period of the first structural bending mode. The look-ahead for LIDAR is set to be 90 feet. Figure 4.16 shows the time history of the vertical acceleration, the angle of attack, and the forward speed. The dot-dash line in the second plot indicates the gust angle of attack while the effective angle of attack α_{eff} is the difference between the dot-dash line and other lines. Figure 4.17 shows the bending moment at wing root. It is shown that the LQR controller reduces

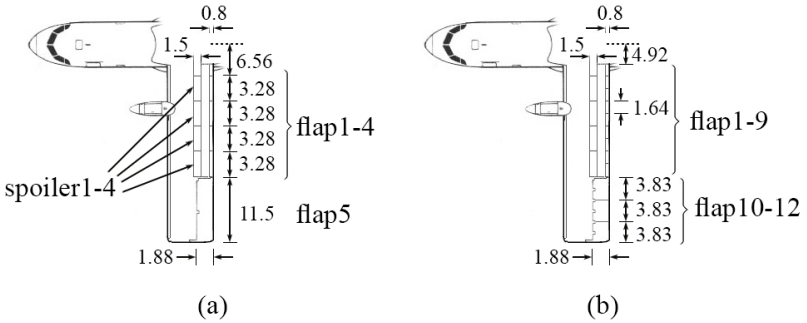


Figure 4.22: (a) flap-array set 1, (b) flap-array set 2 (dimension in feet).

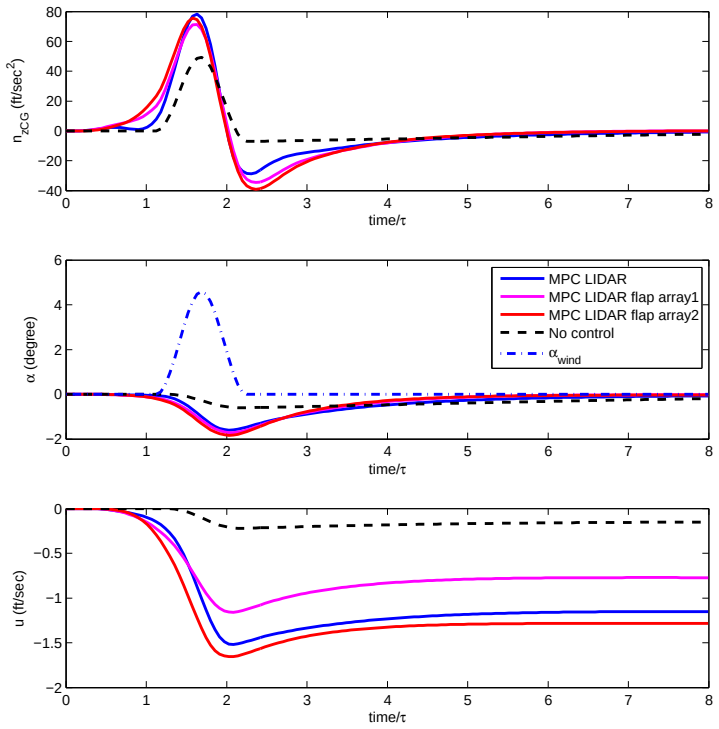


Figure 4.23: From top to bottom: the vertical acceleration, angle of attack, and speed responses to the discrete up-gust comparing the case without GLA, the MPC with LIDAR and flap-array set 1, and the MPC with LIDAR and flap-array set 2.

the peak load from 7400 lb-ft to 5500 lb-ft, the MPC with nose measurement reduces to 2700 lb-ft, and the MPC with LIDAR reduces to 2000 lb-ft. Figure 4.18 compares the elevator, aileron, spoiler

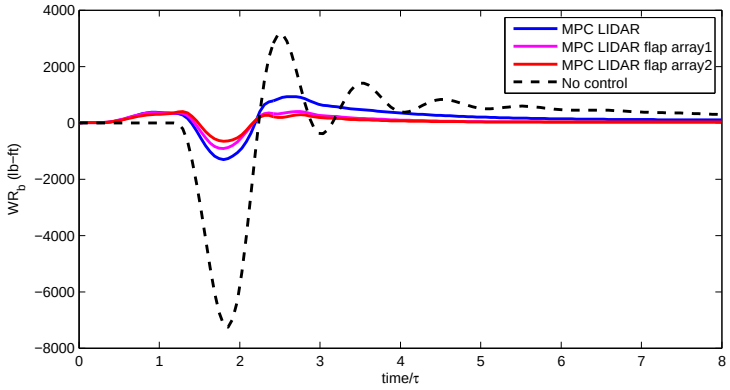


Figure 4.24: The wing root bending moment response to the discrete up-gust comparing the case without GLA, the MPC with LIDAR and flap-array set 1, and the MPC with LIDAR and flap-array set 2.

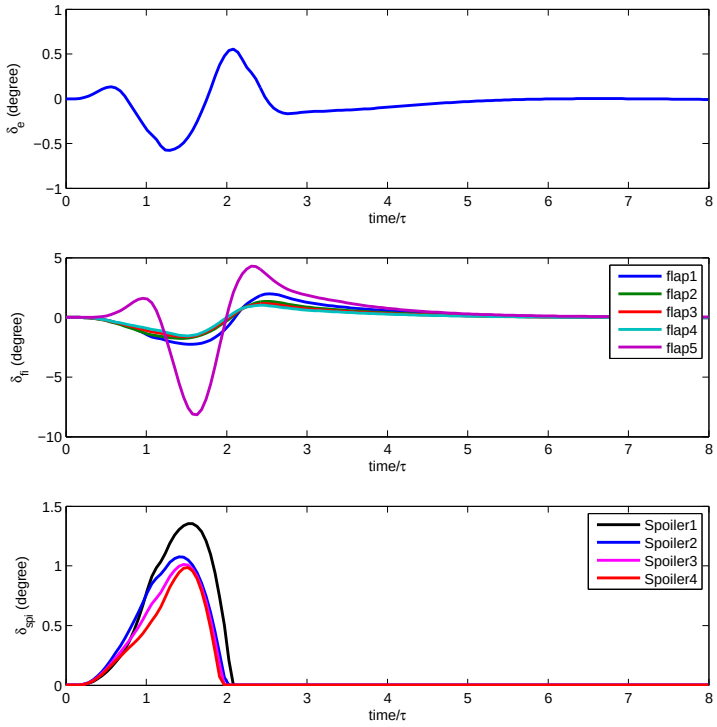


Figure 4.25: The elevator, flap-array, and spoiler responses to the discrete up-gust for the MPC case with LIDAR and flap-array set 1.

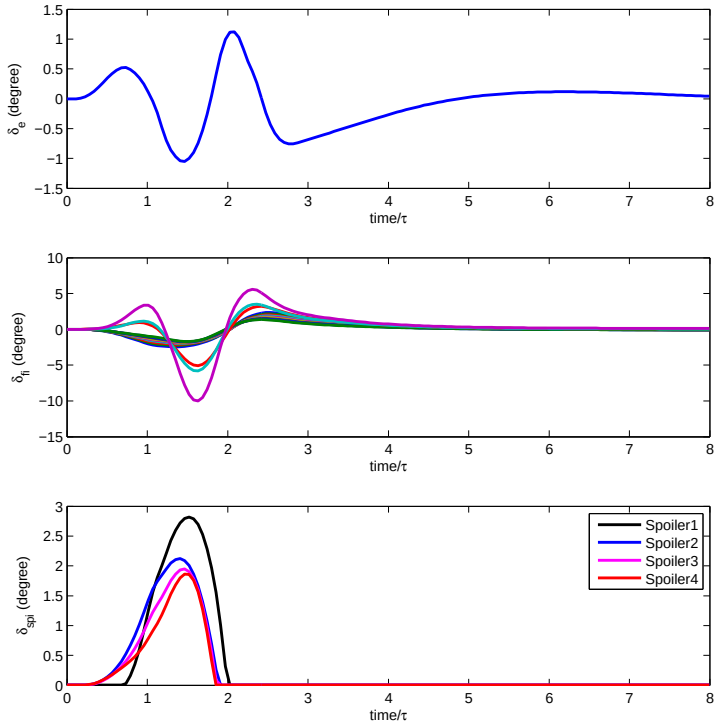


Figure 4.26: The elevator, flap-array, and spoiler responses to the discrete up-gust for the MPC case with LIDAR and flap-array set 2.

set 1 and spoiler set 2 deflections. It is shown that when the gust hits in the upward direction, it increases the effective angle of attack and hence product more lift. The GLA controller responds with the spoiler and aileron deflections to kill the extra lift directly. The elevator is deflected to the positive (down) direction to create a positive pitch moment to compensate the effects from the gust and wing control responses. It is noticed that the LQR controller responds mildly while the MPC with nose measurement gives enough response to reduce the wing bending load. For the MPC with LIDAR, both spoiler and elevator are deflected before the gust actually reaches the aircraft at $t/\tau = 1$. This is because the preview gust information is applied. It is noted from the third plot of Figure 4.16 that the forward speed is dropped due to the application of spoilers. The speed can not be maintained by increasing the weight Q_u. This is because the throttle is not a part of the GLA controller. In practice, the speed control will be run in parallel to the GLA controller, but with different time scale.

Figures 4.19-4.21 depict the responses due to the discrete down-gust comparing the case with no

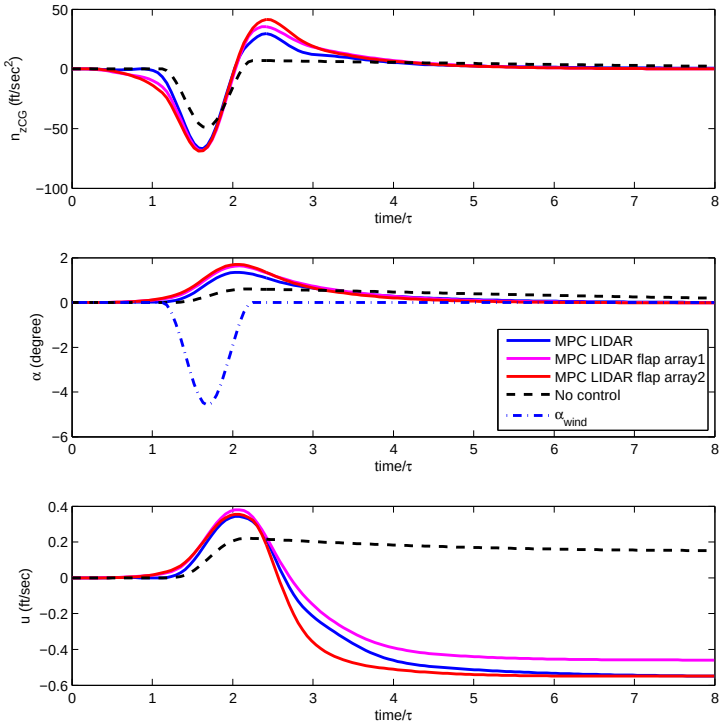


Figure 4.27: From top to bottom: the vertical acceleration, angle of attack, and speed responses to the discrete down-gust comparing the case without GLA, the MPC with LIDAR and flap-array set 1, and the MPC with LIDAR and flap-array set 2.

GLA, the LQR controller, the MPC with nose measurement, and the MPC with LIDAR. Figure 4.19 shows the time history of the vertical acceleration, the angle of attack, and the forward speed, Figure 4.20 shows the bending moment at wing root, and Figure 4.21 compares the elevator, aileron, spoiler set 1 and spoiler set 2 deflections. We notice that in this case when the gust hit downward, it decreases the effective angle of attack. The GLA controller tries to gain the lift back by pitching the aircraft up with elevator and applying the aileron deflection to increase lift. For the MPC with LIDAR case, the elevator is deflected long before the gust arrives in order to prepare extra lift before the gust comes to kill it. The spoilers are applied mildly after the gust exits. This is done in response to the excessive lift from the GLA maneuver. The speed drops for this case are less than the up-gust due to the smaller spoiler deflections. The load alleviation results are quite similar to the case with up-gust. The MPC with LIDAR gives the best result.

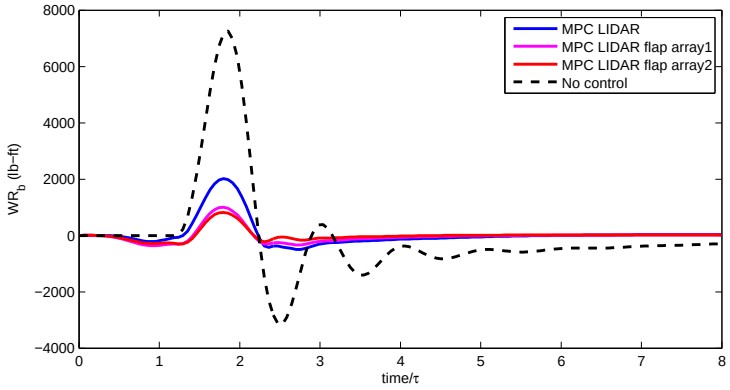


Figure 4.28: The wing root bending moment response to the discrete down-gust comparing the case without GLA, the MPC with LIDAR and flap-array set 1, and the MPC with LIDAR and flap-array set 2.

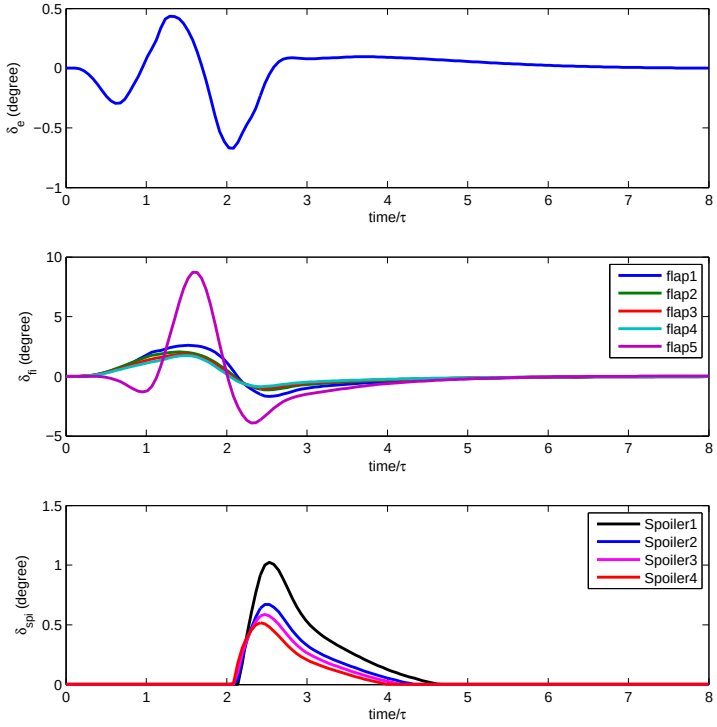


Figure 4.29: The elevator, flap-array, and spoiler responses to the discrete down-gust for the MPC case with LIDAR and flap-array set 1.

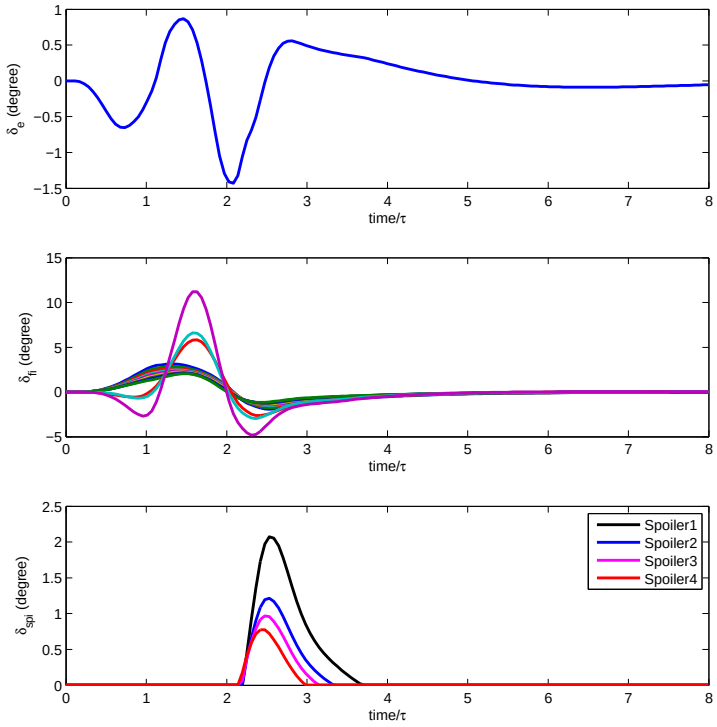


Figure 4.30: The elevator, flap-array, and spoiler responses to the discrete down-gust for the MPC case with LIDAR and flap-array set 2.

Next, we proceed to explore the GLA enhancement with the introduction of a flap-array. Figure 4.22 defines two sets of flap-array placements that will be simulated in the following. The flap-array set 1 composes of four flaps and four spoilers that span evenly up to the location of the aileron. The aileron is notified as flap number five. In flap-array set 2, the trailing edge flap is divided into nine small flaps whereas aileron is divided in the three pieces: inboard, center, and outboard.

Figures 4.23-4.26 show the simulation results of the GTA flying through a discrete up-vertical gust. The plots compare the cases of the MPC with LIDAR on original GTA, the one with flap-array set 1, and the one with flap-array set 2. It is shown in Figure 4.24 that the flap-array set 1 and 2 can reduce the bending moment to 1000 lb-ft and 810 lb-ft, respectively. We notice that the controller commands the aileron (flap number five from the second plots of Figure 4.25) or the outermost aileron (flap number twelve from the second plots of Figure 4.26) to deflect the largest among all other flaps in order to reduce the wing bending moment of the first mode. However, all other flaps deflections

have significant contribution in the overall load reduction. The flaps closer to the wing root can help reducing the load corresponding to the mode shape with high wing root movement. These surfaces are not available on GTA before the introduction of flap-array.

Figures 4.27-4.30 depict the responses due to the discrete down-gust comparing the case of MPC with LIDAR on original GTA, the one with flap-array set 1, and the one with flap-array set 2. The analysis of the simulation results are similar to that of the up-gust where the case with more refined flap-array (set 2) gives the best load reduction.

Table 4.2: Percentage of the peak load value relative to that of the uncontrolled (no GLA) case for discrete down-gust

GLA Type:	LQR	MPC/nose	MPC/LIDAR	Flap-array1	Flap-array2
Angle of attack α	117%	171%	192%	232%	241%
$n_{z_{CG}}$	121%	123%	134%	136%	137%
Wing Root Bending	74%	36.5%	27%	13%	11%
Wing Root Torsion	73%	32.2%	22.7%	10%	7.3%
Mid Wing Bending	75%	35.3%	22.2%	11.67%	9.72%
Mid Wing Torsion	75%	34.67%	25%	11.67%	8.67%
Tail Root Bending	61%	20%	25.56%	23.33%	28.8%
Tail Root Torsion	61.3%	20.67%	25.67%	22.67%	28.67%

In Table 4.2, we compare the alleviation factor by look at the percentage of the peak value to that of the uncontrolled (no GLA) case for the discrete down-gust case. This is the $\mathcal{L}_\infty$ norm of GLA method divide by $\mathcal{L}_\infty$ norm of the uncontrolled case. We can see that the LQR controller, the MPC with nose measurement, the MPC with LIDAR, the MPC with flap-array 1, and the MPC with flap-array 2 reduce the wing root bending to 74%, 36.5%, 27%, 13%, and 11%, respectively. For the wing root and mid wing station, the loads are reduced by around 90% when the flap-array is used. The loads at the tail root remain around the same among all cases because no alteration is made to the tail control surface. The angle of attack and vertical acceleration are increased due to the load alleviation maneuver. However, this is not necessarily critical, as long as it can be covered by the envelope values. For example, the angle of attack only changes by less than two degree at the peak.

The other type of gust to be explored is the continuous turbulence. The continuous gust is defined in the airworthiness regulations *CS*-25 [84] and can be modeled using Dryden spectrum as discussed in § 4.2.1. Since the MPC is the discrete time control method, the simulation is done by generating

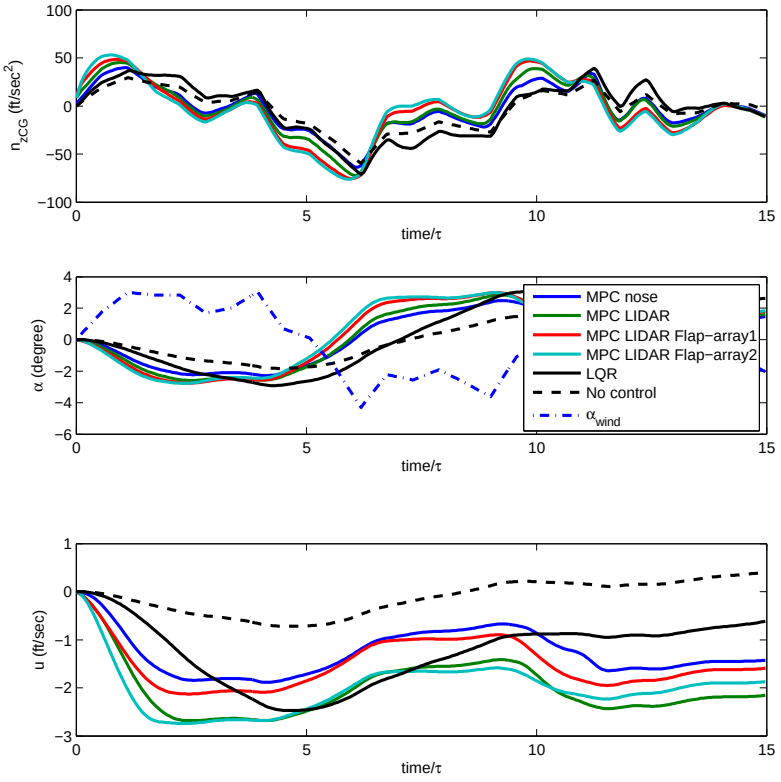


Figure 4.31: From top to bottom: the vertical acceleration, angle of attack, and speed responses to the continuous gust comparing the case without GLA, the LQR controller, the MPC with nose measurement, the MPC with LIDAR and flap-array set 1, and the MPC with LIDAR and flap-array set 2.

a sample of continuous gust speeds. Despite the stochastic nature of the sample, we perform our analysis in the time domain.

Figures 4.31-4.35 show the simulation results of the GTA flying through a continuous vertical gust (4.46) with $\bar{w}_g = 18$ ft/sec and $L = 1$ sec. The plots compare the case when no GLA is applied, the LQR controller, the MPC with nose measurement, the MPC with LIDAR, the MPC with LIDAR and flap-array set 1, and the MPC with LIDAR and flap-array set 2. Figure 4.31 shows the time history of the vertical acceleration, angle of attack, and forward speed. The dot-dash line indicates the angle of attack of the gust. It is shown that at some time instant, the vertical acceleration and the angle of attack from MPC may exceed the uncontrolled case but they are not critical. Figure 4.32 shows the

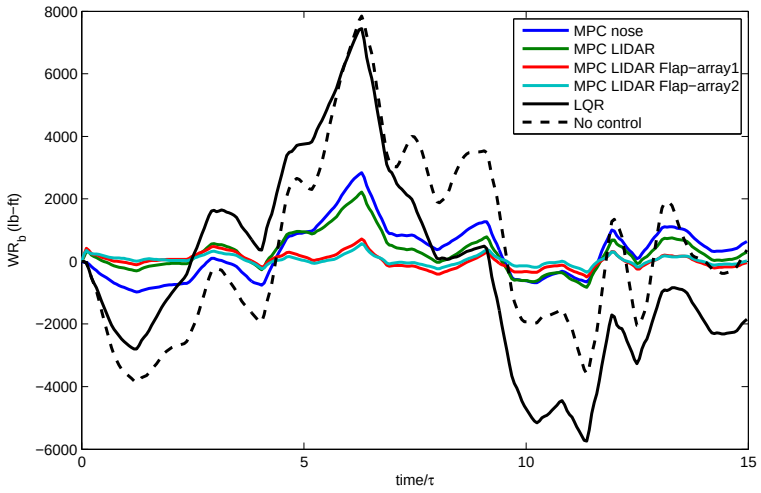


Figure 4.32: The wing root bending moment due to the continuous gust comparing the case without GLA, the LQR controller, the MPC with nose measurement, the MPC with LIDAR and flap-array set 1, and the MPC with LIDAR and flap-array set 2.

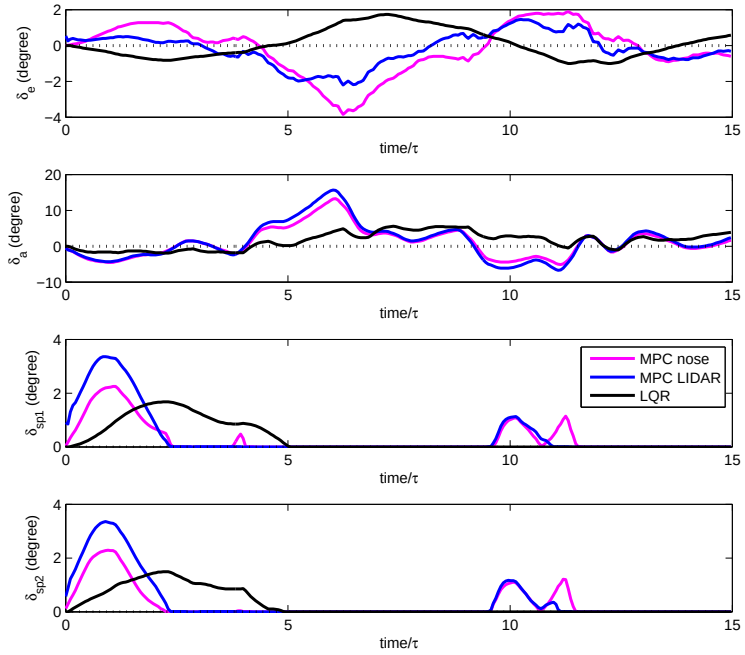


Figure 4.33: The elevator, aileron, spoiler set 1, and spoiler set 2 responses to the continuous gust comparing the case of LQR controller, the MPC with nose measurement, and the MPC with LIDAR.

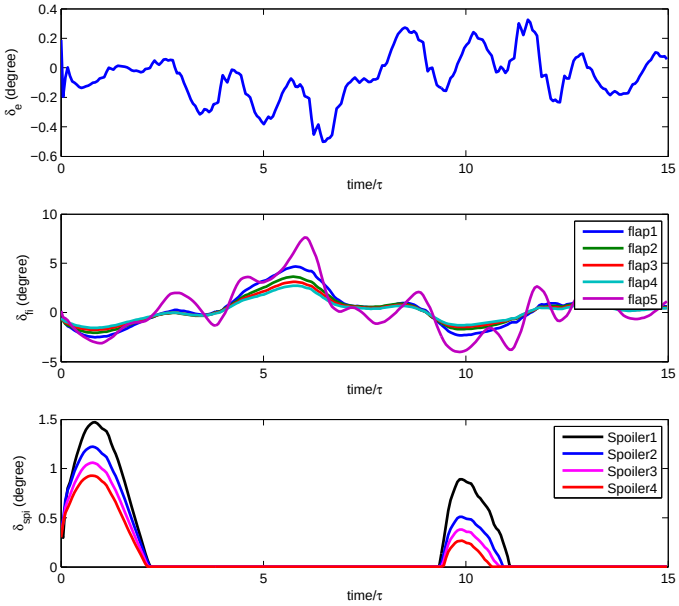


Figure 4.34: The elevator, flap-array, and spoiler responses to the continuous gust for the MPC with LIDAR and flap-array set 1.

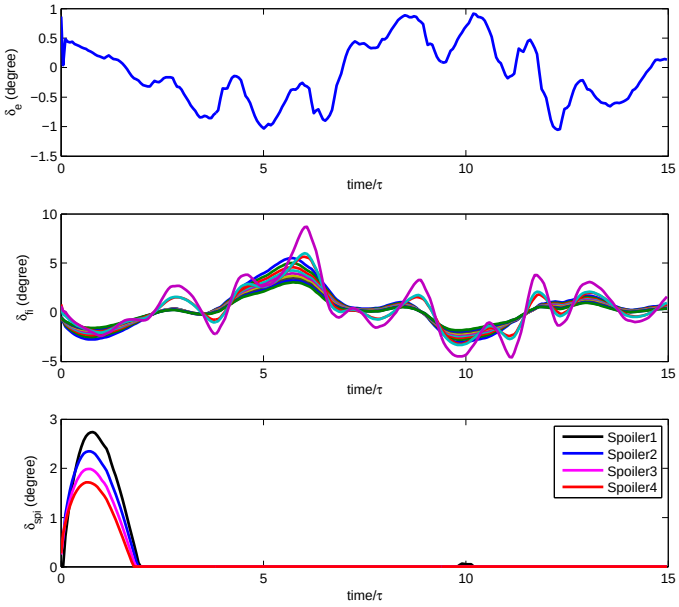


Figure 4.35: The elevator, flap-array, and spoiler responses to the continuous gust for the MPC with LIDAR and flap-array set 2.

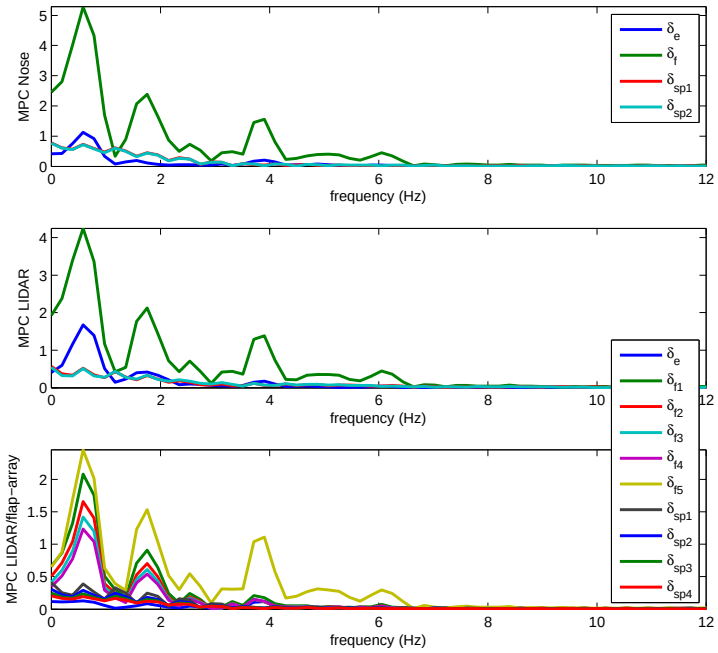


Figure 4.36: The control deflections power spectral density for the MPC with nose measurement, the MPC with LIDAR, and the MPC with LIDAR and flap-array.

bending moment at wing root for all of the GLA methods in this chapter. The LQR controller merely reduces the load at some peaks in this case while all of the MPC controllers yield much better results. The mode refined flap-array (set 2) yield the best load reduction at peaks. Figure 4.33 compares the control surface deflections for the LQR controller, the MPC with nose measurement, and the MPC with LIDAR. Figure 4.34 and 4.35 depict the deflections of elevator, flap-array and distributed spoilers for the MPC with LIDAR and flap-array set 1 and 2, respectively. Finally, Figure 4.36 depicts the power spectral density of the control deflection signals for each MPC case proposed in this chapter. We see that the controllers of these methods mostly activate GLA at lower frequency under 10 Hz and will not interfere with flutter frequency range.

4.6 Remarks

In this chapter, a full six-degrees-of-freedom rigid body and structural dynamics for the generic transport aircraft (GTA) has been developed. The developed equations of motion can effectively integrate large rigid body motion with structural deflections. The model predictive control (MPC) framework

is introduced to address the GLA for a flexible aircraft. The method allows us to use linearized model about the steady trimmed flight and directly consider the load in the cost function. Being a time-domain approach, MPC can avoid the hidden peaks of load signals that the frequency-domain methods such as $\mathcal{H}_\infty$ control can not capture. The stability of the method can be guaranteed by picking the right terminal weighting matrix. The key advantages of using MPC for GLA systems as opposed to other time-domain methods are the possibility to include preview gust information into the aircraft dynamics as well as the the direct consideration of state, control, and output constraints.

We improved the GLA performance by adding sensing capability with the preview of the future gust information. This updated MPC has better prediction accuracy and thus yields the significant load reduction. In contrary, the GLA can be improved with better actuation systems. With the introduction of the flap-array, the GLA controller is able to reduce the wing load from the different bending modes more efficiently. The location, size, and number of flap-array also play important roles in the load reduction as demonstrated in the simulations.

Chapter 5

Conclusion Remarks and Future Directions

This dissertation aimed to develop control platforms that deal with three types of environmental flight uncertainties. We explored the scenarios when an aircraft flies through areas with complex terrain, flies in coordination with other vehicles, and flies through the rough air. A deconfliction algorithm is necessary when UAVs operate in close proximity of each other. A network establishment method is required when relative distances between UAVs fall over the communication range threshold. And finally, a gust load alleviation system is needed when UAVs or other aerial vehicles fly through the turbulence. In this chapter, we mention the concluding remarks, open problems, and future directions.

5.1 Concluding Remarks

In the first part, we provide an approach for the problem when a UAV flies through uncertain environments that contain both stationary and moving obstacles. We develope a deconfliction algorithm that guarantees collision avoidance between a pair of constant speed unicycle-type UAVs as well as convergence to the desired destination for each UAV in presence of static obstacles. The proposed navigation function has been judiciously landscaped by adjusting the swirling effect such that the vehicles operate within their turn rate limitations. The collision avoidance analysis has been performed both for the case where the swirling effect is dominant, as well as for the scenario when it

is combined with the navigation function. Finally, the performance of the algorithm has then been validated with respect to key algorithmic parameters and the mission specific restrictions such as turn rate limitations, constant UAV velocities, and the desire to stay close to the nominal trajectories.

In the second part, the communication uncertainty among the proximity network of cooperative UAVs has been examined. We provide two approaches to establish and maintain the connectivity of UAV networks throughout their entire missions. Optimization and nonlinear programming methods are used to determine the shortest trajectories that bring the UAVs to a connected formation where they are in the range of detection of one another and oriented in the same direction to maintain their connectivity. The control law and path synthesis are derived using necessary conditions by Pontryagin minimum principle. The algorithm for finding the optimal solution for the fixed final network topology is first given, followed by a nonlinear programming method in which the final connected configuration corresponding with the optimal paths emerges from the optimization routine. The first method provides the exact global minimal solution but the computational time gets prohibitive due to the exhaustive search among many optimal candidates. This is not suitable for applying to more than five vehicles. On the other hand, the nonlinear programming method may give a suboptimal solution but it is more suitable for large UAV networks.

In the last part, we focus our research on the uncertainty due to wind gusts. The introduction of an appropriate control scheme to actively alleviate the gust loads can result into weight saving and lower the fuel cost. Gust alleviation systems are dependent on how the structural flexibility of the aircraft affects its dynamics. We thus developed a full six-degrees-of-freedom flexible model for the generic transport aircraft. These equations of motion can effectively integrate large rigid body motion with structural deflections. A model predictive control (MPC) technique is chosen to address the load alleviation for a flexible aircraft. This method allows us to use linearized model about the steady trimmed flight and directly consider the load in the cost function. The key advantages for using MPC as oppose to other time-domain approaches are the possibility of including preview gust information into the aircraft dynamics as well as the the direct consideration of state, control, and output constraints. The stability of MPC can be guaranteed by selecting the appropriate final cost weighting matrix. We demonstrated via simulations that MPC with the preview of look-ahead gust information has better prediction accuracy and thus yields a better load alleviation performance. Finally, we modified the aircraft by adding flap-arrays and corresponding distributed spoilers. Flap-arrays are the new control surfaces that have not been included in conventional aircraft. The simulations shows

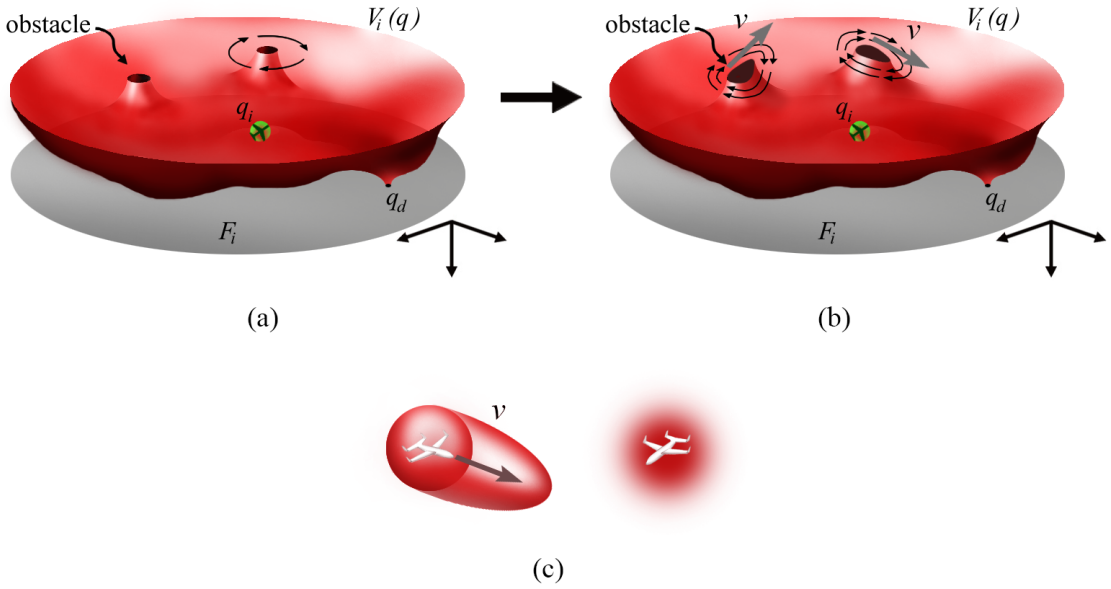


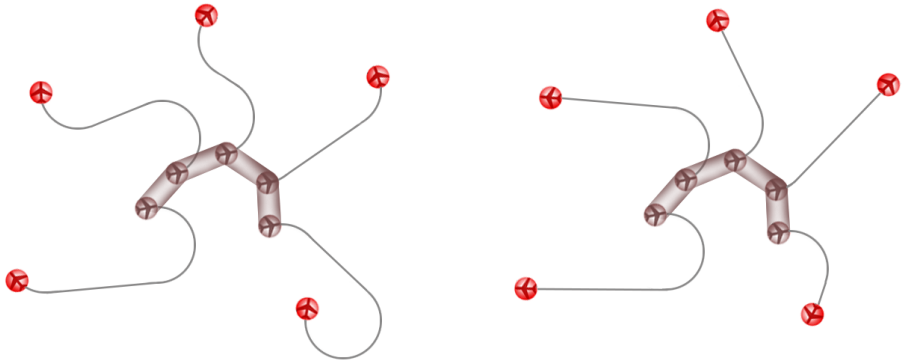
Figure 5.1: (a) the original navigation function in our work, (b) both potential and swirling function are adjusted by the velocity vector of the moving obstacles, and (c) the adjusted protected zone in 3D.

that for both discrete and continuous gust, the MPC with flap-array can reduce the wing bending moment better than the original aircraft. This is due to the availability of control surfaces closer to the wing roots.

5.2 Future Directions and Open problems

In this section, we discuss the open problems and potential future directions for the three types of flight uncertainty managements presented in this dissertation: UAV deconfliction, UAV network acquisition, and gust load alleviation for flexible aircraft.

For UAV deconfliction, we assume to have the ability to detect the locations and velocity of the moving obstacles within the detection range. However, only the location are taken into the consideration to build the potential function and swirling function while the velocity information are used to calculate the safety angle in order to reduce the swirling effect. It is possible that the performance of the algorithm maybe improved if the velocity information of the sensed obstacles can be included in the potential and swirling function construction processes. This is analogous to having a



(a) (b)

Figure 5.2: (a) optimal dispersion, (b) optimal dispersion when final orientations are not specified.

preview of future information similar to the method in Section 4.3. The protected zone that defines the safety for collision avoidance may also be adjusted similarly. The idea is depicted in Figure 5.1. The protected zone can become convex that has a parabolic shape on the direction that vehicle flies to while has a sphere shape on the opposite side. The potential function can be created based on these shapes. The swirling vector field can have the similar shape but in two dimensions. The second extension to this work involves allowing UAV speed to vary between its stall and maximum speed. This can give more flexibility to the controller. In the current setting, we only use the direction of the gradient of the navigation function in conjunction with the swirling effect and give it as a command desired heading. If the UAV velocity is allowed to change, we can then include the magnitude information of the gradients as a desired velocity profile. This can possibly give faster and more efficient avoidance response. Another extension is to formulate the proof for collision avoidance when n vehicles are in the range of detection at the same time and superposition of swirling effect are applied. We demonstrated this by simulation but did not provide the required mathematical analysis. Other extensions for UAV deconfliction include the derivation of higher fidelity models for UAVs that include drift or dynamic coupling terms. Path-length estimating the distances to the goal point can also be added in order to minimize the effect of the deconfliction maneuver on the time of arrival at a particular way point. Finally, the robust performance of the proposed deconfliction algorithm should be examined as it is the significant measure for actual multiple UAV missions.

For UAV optimal path planning, there are several possible extensions of this work. First, the assumption that the initial locations of the UAVs form a convex polygon should be relaxed. This may

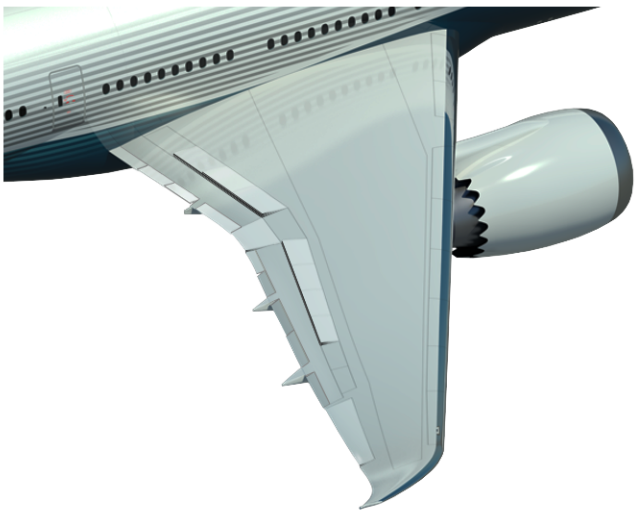


Figure 5.3: The wing flap-array on the larger airplane (Boeing 777-9X), noted that the spoilers distribution is similar to the one presented in Chapter 4.

require a set of commands that allow some vehicles to loiter while other vehicles arrive into the area of interest. Same set of control laws and path synthesis still can apply. One possible extension is to apply the algorithm in the opposite direction, instead of bringing the UAVs into connection, we can disperse the formation to the specified area with the minimum time as shown in Figure 5.2a. The UAVs may not be required to head into the certain direction upon arrival at the dispersed location, hence their optimal paths are only composed of two segments. The idea is shown in Figure 5.2b. The other extension involves allowing the UAV speed to vary between stall and maximum speeds with some vehicle traveling faster than others if necessary. This could give more flexibility for the problem and open the opportunity to specify the location where the UAVs acquire a connected network. In addition, this setup provides the possibility of developing an algorithm that not only brings the UAVs to a connected network, but also forms a desired formation shape. On other extension is to explore the design of the graph topology to determine the best prescribed formation that has the minimum connecting path-length. Finally, we can consider the scenario that combine the optimal path-planning for network connectivity, maintaining the formation, optimal network dispersion as well as UAV deconfliction.

For the work of gust load alleviation (GLA), the generic transport aircraft (GTA) model was developed based on the vehicle description in the ZAERO user's manual [80] that provide the finite

element model for the structural dynamics. This is the aircraft with the same size and configuration to the turbo-prop ATR-42 or Bombardier Dash-8-100. It is sufficient to demonstrate various control designs for GLA as well as to be modified with the addition of flap-array. However, the structural flexibility and the interaction between the rigid body and flexible body are increased with respect to the size of the aircraft. The separation of structural frequencies for GTA are wide enough that the model may not provide control-structure interaction that is of interest in the future research objectives.

Today, the wingspan of the Airbus A380, which entered into service in 2007, is over twice the distance (79.75 meters) to that of the comet (35 meters) in 1952. The increase in size and weight has brought a unique set of aeroelastic and handling quality issues. Currently, control law usually includes roll-off or notch filters to avoid aeroelastic/aeroservoelastic interactions. Our future research interest on this topic lies upon the large interaction between the control commands, structural modes, and rigid body dynamics. It is thus necessary to develop the GLA system for large flexible aircraft. The extension of this work is to develop the structural and aerodynamic model for the aircraft with the similar size as Boeing 777-9X. Many similar approaches can still be used such as stick-and-beam model for the wings and fuselage. Figure 5.3 depicts the flap-array and spoilers distribution for GLA on larger swept wing. The goal is to bring the first structural mode frequency down from 5.6Hz for the GTA to a value between 1 and 2Hz for the large transport aircraft. The notch filters could become ineffective. There should be 20 to 30 modes below 10Hz on an aircraft like the Boeing 777. Separation of structural modes should not be large so that we can capture the possible behavior of control energy leaking into nearby modes. The model should also include sufficient details to ensure that the control law does not excite instabilities in either rigid body or structural dynamics such as flutter. Satisfying this requirements may mean including structural modes of much higher frequency in the structural analysis and control synthesis than typically considered.

Appendix A

Proof of Proposition 3.2: The Hessian of $V_i(\mathbf{q})$ is:

$$\nabla^2_{\mathbf{q}_i} V_i(\mathbf{q}) = \Big[(\Gamma_i(\mathbf{q})^k + \beta_i(\mathbf{q}))\left[k\nabla_{\mathbf{q}_i}\beta_i(\mathbf{q})\nabla_{\mathbf{q}_i}\Gamma_i(\mathbf{q})^T - \nabla_{\mathbf{q}_i}\Gamma_i(\mathbf{q})\nabla_{\mathbf{q}_i}\beta_i(\mathbf{q})^T + k\beta_i(\mathbf{q})\nabla^2_{\mathbf{q}_i}\Gamma_i(\mathbf{q}) - \Gamma_i(\mathbf{q})\nabla^2_{\mathbf{q}_i}\beta_i(\mathbf{q})\right]$$
$$-(\frac{1}{k}+1)\left[k\beta_i(\mathbf{q})\nabla_{\mathbf{q}_i}\Gamma_i(\mathbf{q}) - \Gamma_i(\mathbf{q})\nabla_{\mathbf{q}_i}\beta_i(\mathbf{q})\right]\left[k\Gamma_i^{k-1}\nabla_{\mathbf{q}_i}\Gamma_i(\mathbf{q}) + \nabla_{\mathbf{q}_i}\beta_i(\mathbf{q})\right]^T\Big] \Big/ \Big[k(\Gamma_i(\mathbf{q})^k + \beta_i(\mathbf{q}))^{1/k+2}\Big]. \tag{5.1}$$

When $\nabla_{\mathbf{q}_i}\Gamma_i(\mathbf{q}) = 0$ and $\nabla_{\mathbf{q}_i}\beta_i(\mathbf{q}) = 0$, the Hessian becomes

$$\nabla^2_{\mathbf{q}_i} V(\mathbf{q}) = \frac{k\beta_i\nabla^2_{\mathbf{q}_i}\Gamma_i(\mathbf{q}) - \Gamma_i(\mathbf{q})\nabla^2_{\mathbf{q}_i}\beta_i(\mathbf{q})}{k(\Gamma_i(\mathbf{q})^k + \beta_i(\mathbf{q}))^{1/k+1}}. \tag{5.2}$$

From (2.7), let $\tilde{q} = \left(\frac{\|\mathbf{q}_i - \mathbf{q}_{d_i}\|}{r_p}\right)$ and we have

$$\nabla_{\mathbf{q}_i}\Gamma_i(\mathbf{q}) = \begin{cases} 2(\mathbf{q}_i - \mathbf{q}_{d_i}) + K_p\nabla_{\mathbf{q}_i}d_i^2 & \text{when} \quad \|\mathbf{q}_i - \mathbf{q}_{d_i}\| \geq r_p, \\ \\ 2(\mathbf{q}_i - \mathbf{q}_{d_i}) + K_p\left[f(\tilde{q})\nabla_{\mathbf{q}_i}d_i^2 \right. & \\ \quad \left. + f'(\tilde{q})\frac{d_i^2(\mathbf{q}_i - \mathbf{q}_{d_i})}{r_p\|\mathbf{q}_i - \mathbf{q}_{d_i}\|}\right] & \text{when} \quad \|\mathbf{q}_i - \mathbf{q}_{d_i}\| < r_p, \end{cases}$$

and

$$\nabla^2_{\mathbf{q}_i}\Gamma_i(\mathbf{q}) = \begin{cases} 2I + K_p T \mathbf{diag}(2,2,0) T^{-1} & \text{when} \quad \|\mathbf{q}_i - \mathbf{q}_{d_i}\| \geq r_p, \\ \\ 2I + K_p \left[\beta(\tilde{q}) T \mathbf{diag}(2,2,0) T^{-1} + \left(\frac{f'(\tilde{q}) d_i^2}{r_p \|\mathbf{q}_i - \mathbf{q}_{d_i}\|} \right) I \right. \\ \quad + f'(\tilde{q}) \left(\frac{(\nabla_{\mathbf{q}_i} d_i^2)(\mathbf{q}_i - \mathbf{q}_{d_i})^T + (\mathbf{q}_i - \mathbf{q}_{d_i})(\nabla_{\mathbf{q}_i} d_i^2)^T}{r_p \|\mathbf{q}_i - \mathbf{q}_{d_i}\|} \right) \\ \quad \left. + \left(\frac{f''(\tilde{q})}{r_p^2} - \frac{f'(\tilde{q})}{r_p \|\mathbf{q}_i - \mathbf{q}_{d_i}\|} \right) d_i^2 \left(\frac{\mathbf{q}_i - \mathbf{q}_{d_i}}{\|\mathbf{q}_i - \mathbf{q}_{d_i}\|} \right) \left(\frac{\mathbf{q}_i - \mathbf{q}_{d_i}}{\|\mathbf{q}_i - \mathbf{q}_{d_i}\|} \right)^T \right] \\ \\ & \text{when} \quad \|\mathbf{q}_i - \mathbf{q}_{d_i}\| < r_p, \end{cases}$$

where I is the identity matrix in $\mathbf{R}^{3\times 3}$, T is a coordinate transformation matrix that transforms a unit vector in the z-axis to the vector $\mathbf{q}_{d_i} - \mathbf{q}_{o_i}$, and f' and f'' are the first and second derivatives of the function f defined in (2.12), respectively. Let $\hat{\mathbf{q}}$ be a unit vector along the vector difference $\mathbf{q}_i - \mathbf{q}_{d_i}$ and ξ be the angle between $\mathbf{q}_i - \mathbf{q}_{d_i}$ and $\mathbf{q}_{o_i} - \mathbf{q}_{d_i}$; then $\nabla^2_{\mathbf{q}_i}\Gamma_i(\mathbf{q})$ becomes

$$\nabla^2_{\mathbf{q}_i}\Gamma_i(\mathbf{q}) = \begin{cases} 2I + K_p T \mathbf{diag}(2,2,0) T^{-1} & \text{when} \quad \|\mathbf{q}_i - \mathbf{q}_{d_i}\| \geq r_p, \\ \\ 2I + K_p \left[\beta(\tilde{q}) T \mathbf{diag}(2,2,0) T^{-1} \right. \\ \quad + \frac{f'(\tilde{q})}{r_p} \left(d_i \sin\xi I + (\nabla_{\mathbf{q}_i} d_i^2) \hat{\mathbf{q}}^T + \hat{\mathbf{q}} (\nabla_{\mathbf{q}_i} d_i^2)^T \right) \\ \quad \left. + \left(\frac{f''(\tilde{q}) d_i^2}{r_p^2} - \frac{f'(\tilde{q}) d_i \sin\xi}{r_p} \right) \hat{\mathbf{q}} \hat{\mathbf{q}}^T \right] & \text{when} \quad \|\mathbf{q}_i - \mathbf{q}_{d_i}\| < r_p. \end{cases}$$

When the i-th vehicle reaches its destination that is also far from all other vehicles, we also have $\Gamma_i(\mathbf{q}) = 0, \beta_i(\mathbf{q}) = 1$, and $\nabla^2_{\mathbf{q}_i}\beta_i(\mathbf{q}) = 0$ simplifies (5.2) to $\nabla^2_{\mathbf{q}_i} V_i(\mathbf{q}) = \nabla^2_{\mathbf{q}_i}\Gamma_i(\mathbf{q})$.

Now, by examining the above equation for the case when $\|\mathbf{q}_i - \mathbf{q}_{d_i}\| = 0 < r_p$, we have $\beta(\tilde{q} = 0) = 0$, $d_i = 0$, and the high order polynomial are chosen such that $f'(0) = f''(0) = 0$ for the potential function to be twice differentiable throughout the configuration space. Hence, $\nabla^2_{q_i} V_i(\mathbf{q}_i)$ evaluated at $\mathbf{q}_{d_i}$ equals to $2I$ (positive definite) and hence, the equilibrium point is a minimum. ■

Appendix B

Linear Longitudinal Dynamics

$$\begin{aligned}\dot{\mathbf{x}}_{\text{long}} &= A_{\text{long}}\mathbf{x}_{\text{long}} + B_{\text{long}}\mathbf{u}_{\text{long}} + B_{w_{\text{long}}}\mathbf{w}_{\mathbf{g}_{\text{long}}} \\ \mathbf{y}_{\text{long}} &= C_{\text{long}}\mathbf{x}_{\text{long}} + D_{\text{long}}\mathbf{u}_{\text{long}} + D_{w_{\text{long}}}\mathbf{w}_{\mathbf{g}_{\text{long}}}\end{aligned}$$

where

$$\begin{aligned}\mathbf{x}_{\text{long}} &= [u\ \alpha\ \theta\ q\ h\ \eta_1\ \dot{\eta}_1\ \eta_4\ \dot{\eta}_4\ \eta_5\ \dot{\eta}_5\ \eta_9\ \dot{\eta}_9\ \eta_{10}\ \dot{\eta}_{10}\ \eta_{12}\ \dot{\eta}_{12}\ \eta_{14}\ \dot{\eta}_{14}]', \\ \mathbf{y}_{\text{long}} &= [u\ \alpha\ \theta\ q\ h\ n_{z_{\text{CP}}}\ n_{z_{\text{CG}}}\ n_{z_{\text{Tail}}}\ WR_b\ WR_t\ MW_b\ MW_t\ TR_b\ TR_t]', \\ \mathbf{u}_{\text{long}} &= [\delta_e\ \delta_f\ \delta_T\ \delta_{sp1}\ \delta_{sp2}]', \\ \mathbf{w}_{\mathbf{g}_{\text{long}}} &= [u_g\ \alpha_g\ \dot{\alpha}_g]'.\end{aligned}$$

$A_{\text{long}} =$

$$\begin{bmatrix}
-0.0095 & 29.41 & -32.15 & -17.15 & 0 & -0.002 & -8.4e^{-5} & 0.0067 & -3.4e^{-6} & 0.014 & -7.3e^{-5} & 0.048 & 1.29e^{-5} & -0.043 & 1.37e^{-5} & -0.0015 & 1.45e^{-5} & 0.0029 & 3.77e^{-5} \\
-1.9e^{-6} & -1.392 & 0.0028 & 0.96 & 0 & 1.36e^{-4} & 4.74e^{-6} & -3.81e^{-4} & 1.93e^{-7} & -7.88e^{-4} & 4.12e^{-6} & 2.7e^{-4} & 7.29e^{-7} & 0.0024 & -7.72e^{-7} & 8.3e^{-5} & -8.21e^{-7} & -1.64e^{-4} & -2.13e^{-6} \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
8.67e^{-5} & -1.167 & 2.65e^{-4} & -0.303 & 0 & -9.21e^{-5} & -1.73e^{-6} & -0.003 & 9.36e^{-7} & 2.97e^{-4} & -9.9e^{-7} & -4.89e^{-5} & 3.48e^{-6} & -8.99e^{-4} & -2.02e^{-7} & 2.58e^{-4} & -1.06e^{-5} & -3.55e^{-5} & -3.83e^{-8} \\
0.0393 & -449.3 & 450 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1007.1 & 0 & 2.38 & 0 & -1250.98 & -14.15 & -0.031 & 1.1e^{-4} & 1.027 & -0.0062 & -0.418 & -3.67e^{-4} & -3.01 & 0.001 & -0.0073 & -7.85e^{-6} & -0.333 & 1.07e^{-4} \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 33.94 & 0 & -2.55 & 0 & 0.002 & 1.1e^{-4} & -3816.54 & -24.71 & -0.006 & 7.33e^{-5} & 0.005 & 4.79e^{-5} & 0.012 & -8.2e^{-6} & -0.012 & 2.93e^{-5} & 0.025 & 1.16e^{-4} \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 874 & 0 & 0.318 & 0 & -0.159 & -0.0062 & -5.89e^{-4} & 7.21e^{-5} & -6563.8 & -32.41 & -0.34 & -2.14e^{-4} & -2.5 & 9.02e^{-4} & -0.012 & 3.56e^{-5} & -0.237 & 2.85e^{-4} \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & -214.07 & 0 & -2.94 & 0 & 0.0028 & -3.88e^{-4} & 0.056 & 7.25e^{-5} & -0.045 & -2.13e^{-4} & -20805 & -57.697 & 0.189 & 1.013e^{-5} & 0.035 & 5.24e^{-5} & -0.29 & -0.0016 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & -157.01 & 0 & 0.321 & 0 & 0.029 & 0.001 & -0.0066 & -1.14e^{-5} & -0.156 & 9.02e^{-4} & 0.059 & 1.19e^{-5} & -22122.5 & -59.495 & -0.0042 & -1.51e^{-5} & 0.018 & -2.07e^{-4} \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & -53.01 & 0 & 20.88 & 0 & -0.012 & -3.73e^{-5} & -0.42 & -9.69e^{-7} & -0.055 & -2.55e^{-5} & -0.0011 & 7.29e^{-4} & -0.19 & -1.35e^{-5} & -27906.7 & -66.823 & 0.104 & 5.75e^{-4} \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & -454.62 & 0 & 1.985 & 0 & 0.024 & 9e^{5} & -0.047 & 1.12e^{-4} & -0.212 & 2.73e^{-4} & -0.012 & -0.0015 & 0.72 & -0.0002 & 0.058 & 3.51e^{-4} & -35780.6 & -75.67
\end{bmatrix}$$

$B'_{\text{long}} =$

$$\begin{bmatrix}
4.067 & -0.2032 & 0 & -1.79 & 0 & 0 & -22.49 & 0 & 21.86 & 0 & -4.41 & 0 & 25.71 & 0 & -2.47 & 0 & -179.68 & 0 & -16.22 \\
9.75e^{-4} & -1.5e^{-23} & 0 & 1.4e^{-24} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
7.25 & -0.62 & 0 & -0.037 & 0 & 0 & -289.15 & 0 & 4.04 & 0 & -235.7 & 0 & -16.72 & 0 & 42.61 & 0 & 4.77 & 0 & 8.67 \\
-68.65 & 0.6155 & 0 & -0.059 & 0 & 0 & 2.62 & 0 & 7.99 & 0 & 12.06 & 0 & -59.91 & 0 & 2.51 & 0 & 16.61 & 0 & -40.01 \\
-68.65 & 0.616 & 0 & -0.059 & 0 & 0 & 78.6 & 0 & 4.9 & 0 & 73.22 & 0 & -69.36 & 0 & -18.24 & 0 & 27.4 & 0 & -172.72
\end{bmatrix}$$

$B'_{w_{\text{long}}} =$

$$\begin{bmatrix}
-0.0095 & -1.92e^{-6} & 0 & 8.668e^{-5} & 0.0393 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
29.402 & -1.39 & 0 & -1.17 & -449.3 & 0 & 1007.1 & 0 & 33.93 & 0 & 874 & 0 & 214.07 & 0 & -157 & 0 & -53.01 & 0 & -454.61 \\
-0.075 & -0.01 & 0 & 0.037 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}$$

$C_{\text{long}} =$

$$\begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-2.94e^{-4} & -633.28 & 1.252 & -17.43 & 0 & -0.208 & -9.1e^{-4} & -0.304 & -6.23e^{-4} & -0.736 & -5.21e^{-5} & 6.25 & 0.0167 & 0.63 & -0.0015 & -0.155 & -0.0009 & 1.348 & 0.002 \\
0.0014 & -656.35 & 1.257 & -23.39 & 0 & -0.192 & -7.5e^{-4} & 0.19 & 0.003 & -0.568 & 7.33e^{-4} & -13.627 & -0.0384 & 3.69 & 0.0067 & 3.69 & 0.008 & 10.77 & 0.022 \\
0.0048 & -702.13 & 1.268 & -35.33 & 0 & -0.22 & 0.001 & 1.054 & 0.0095 & -0.597 & 0.0005 & 1.843 & 0.0047 & 0.873 & -7.7e^{-4} & 15.486 & 0.0359 & 9.951 & 0.0202 \\
0 & 0 & 0 & 0 & 0 & 60937.4 & 0 & -2.35e^{5} & 0 & 100129.7 & 0 & -5.55e^{6} & 0 & 7.48e^{5} & 0 & -1.64e^{6} & 0 & 2.21e^{6} & 0 \\
0 & 0 & 0 & 0 & 0 & 22.35e^{6} & 0 & -1.19e^{8} & 0 & 3.86e^{6} & 0 & -1.5e^{9} & 0 & 3.11e^{8} & 0 & -5.36e^{8} & 0 & 4.66e^{8} & 0 \\
0 & 0 & 0 & 0 & 0 & -3323.8 & 0 & 16707.2 & 0 & -2647.69 & 0 & 261134.8 & 0 & -4.44e^{4} & 0 & 85935.2 & 0 & -93926 & 0 \\
0 & 0 & 0 & 0 & 0 & 55.39e^{4} & 0 & -2.78e^{6} & 0 & 440071.7 & 0 & -4.35e^{7} & 0 & 7403393 & 0 & -1.43e^{7} & 0 & 1.56e^{7} & 0 \\
0 & 0 & 0 & 0 & 0 & 102.78 & 0 & -4.0758 & 0 & 83.261 & 0 & -20.539 & 0 & -30.3466 & 0 & 18.872 & 0 & -216.83 & 0 \\
0 & 0 & 0 & 0 & 0 & 1.71e^{4} & 0 & -678.3 & 0 & 13854.08 & 0 & -3412.34 & 0 & -5048.76 & 0 & 3137.75 & 0 & -36062 & 0
\end{bmatrix}$$

$D'_{\text{long}} =$

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & -103.132 & -138.238 & -208.5378 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -6.7e^{-21} & -6.67e^{-21} & -6.61e^{-21} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -279.47 & -280.23 & -281.72 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 276.4 & 275.2 & 272.93 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 276.43 & 275.25 & 272.98 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$D'_{w_{\text{long}}} =$

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & -2.95e{-}4 & 0.0014 & 0.0048 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -633.285 & -656.353 & -702.127 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -4.5474 & -3.813 & -2.3435 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Bibliography

[1] J. Kuchar and L. Yang. A review of conflict detection and resolution modeling methods, *IEEE Transactions on intelligent transportation system*, (1) 4: 179-189, 2000.

[2] I. Hwang and C. Tomlin. Protocol-based conflict resolution for air traffic control, Standford University, Tech. Report SUDAAR-762, 2002.

[3] K. Bilimoria, B. Sridhar, and G. Chatterji. Effects of conflict detection methods for air traffic management, *AIAA Guidance, Navigation, and Control Conference*, 1996.

[4] C. Tomlin, G. Pappas, and S. Sastry. Conflict resolution for air traffic management: a study in multiagent hybrid systems, *IEEE Transactions on Automatic Control*, (43) 4: 509-521, 1998.

[5] P. Fiorini and Z. Shiller. Motion planning in dynamic environments using velocity obstacles, *International Journal of Robotics Research*, (17) 7: 760-772, 1998.

[6] E. Frazzoli, Z. H. Mao, J. H. Oh, and E. Feron. Resolution of conflicts involving many aircraft via semidefinite programming, *Journal of Guidance, Control, and Dynamics*, (24) 1: 79-86, 2001.

[7] S. M. Malaek and A. Abbasi. Near-optimal terrain collision avoidance trajectories using elevation maps, *IEEE Transactions on Aerospace and Electronic Systems*, (47) 4: 2490-2501, 2011.

[8] B. Ajith Kumar. and D. Ghose. Radar-assisted collision avoidance guidance strategy for planar flight, *IEEE Transactions on Aerospace and Electronic Systems*, (37) 1: 77-90, 2001.

[9] E. Lalish, K.A. Morgansen, and T. Tsukamaki. Decentralized reactive collision avoidance for multiple unicycle-type vehicles, *American Control Conference*, 2008.

[10] C. Carbone, U. Ciniglio, F. Corraro, and L. Luongo. A novel 3D geometric algorithm for aircraft autonomous collision avoidance, *IEEE Conference of Decision and Control*, 2006.

[11] A. Chakravarthy and D. Ghose. Obstacle avoidance in a dynamic environment: A collision cone approach, *IEEE Transactions on System, Man, and Cybernetics*, (28) 5: 562- 574, 1998.

[12] K. D. Bilimoria. A geometric optimization approach to aircraft conflict resolution, *AIAA Guidance Navigation and Control Conference*, 2000.

[13] O. Khatib. Real-Time obstacle avoidance for manipulators and mobile robots, *International Journal of Robotics Research*, (5) 1: 90-98, 1986.

[14] O. Brock and O. Khatib. High-Speed navigation using the global dynamic window approach, *Proceedings of IEEE International Conference on Robotics and Automation*, Detroit, MI, pp. 341-346, 1999.

[15] E. Rimon and D. E. Koditschek. Exact robot navigation using artificial potential functions, *IEEE Transactions on Robotics and Automation*, (8) 5: 501-518, 1992.

[16] E. Rimon and D. E. Koditschek. Robot navigation function on manifold with boundary, *Advanced in Applied Mathematics*, (11) 4: 412-442, 1990.

[17] R. Olfati Saber and R. M. Murray. Flocking with obstacle avoidance: cooperation with limited communication in mobile agents, *IEEE Conference of Decision and Control*, 2003.

[18] H. G. Tanner, A. Jadbabaie, and G. J. Pappas. Stable flocking of mobile agents, *IEEE Conference of Decision and Control*, 2003.

[19] D. V. Dimarogonas and K. J. Kyriakopoulos. Decentralized navigation function for multiple robotic agents with limit sensing capabilities, *Journal of Intelligent Robotic Systems*, (48) 3: 411-433, 2007.

[20] D. V. Dimarogonas and K. J. Kyriakopoulos. Decentralized motion control of multiple agents with double integrator dynamics, *IFAC*, 2005.

[21] A. Rahmani, K. Kosuge, T. Tsukamaki, and M. Mesbahi. Multiple UAV deconfliction via navigation functions, *AIAA Guidance Navigation and Control Conference*, 2008.

[22] M. C. De Gennaro and A. Jadbabaie. Formation control for a cooperative multi-agent system using decentralized navigation functions, *American Control Conference*, June, 2006.

[23] M. Abramowitz and I. A. Stegun (Editors), *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables*, New York, Dover, 1972.

[24] I. Mitchell. Game of two identical vehicles, Department of Mathematics, Technical Report, Stanford University, 2001.

[25] J. W. Andrews. A relative motion analysis of horizontal collision avoidance, *SAFE Journal*, (8) 2: 7-8, 1978.

[26] P. Panyakeow and M. Mesbahi. Decentralized deconfliction algorithms for unicycle UAVs, *American Control Conference*, 2010.

[27] P. Panyakeow and M. Mesbahi. Deconfliction algorithms for a pair of constant speed unmanned aerial vehicles, *IEEE Transactions on Aerospace and Electronic Systems*, 2014.

[28] G. Roussos, D. V. Dimarogonas, and K. J. Kyriakopoulos. 3D navigation function and collision avoidance for a nonholonimic vehicle, *American Control Conference*, 2008.

[29] G. Roussos, D. V. Dimarogonas, and K. J. Kyriakopoulos. 3D navigation and collision avoidance for nonholonomic aircraft-like vehicles, *International Journal Adaptive Control and Signal Processing*, (24) 10: 900-920, 2009,

[30] M. Aicardi, G. Cannata, G. Casalino, and G. Indiveri. Guidance of 3D underwater non-holonomic vehicle via projection on holonomic solutions, *Symposium on Underwater Robotic Technology, SURT 2000*, June 2000.

[31] M. Aicardi, G. Cannata, and G. Indiveri. Closed loop control of underactuated vehicles via velocity field tracking, *Proceedings of IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, 2001.

[32] A. Bicchi and L. Pallottino. On optimal cooperative conflict resolution for air traffic management systems, *IEEE Transactions on Intelligent Transportation Systems*, (1) 4: 221-231, 2000.

[33] G. Inalhan, D. Stipoanovic, and C. Tomlin. Decentralized optimization, with application to multiple aircraft coordination, *IEEE Conference of Decision and Control*, 2002.

[34] T. Gandhi, M. T. Yang, R. Kasturi, O. Camps, L. Coraor, and J. McCandless Detection of obstacles in the flight path of an aircraft, *IEEE Transactions on Aerospace and Electronic Systems*, (39) 1: 176-191, 2003.

[35] R. W. Osborne, Y. Bar-Shalom, P. Willett, G. Baker. Design of an adaptive passive collision warning system for UAVs, *IEEE Transactions on Aerospace and Electronic Systems*, (47) 3: 2169-2189, 2011.

[36] H. Yu and R. W. Beard, A vision-based collision avoidance technique for micro air vehicles using local-level frame mapping and path planning, *Journal of Autonomous Robots*, (34) 1-2: 93-109, 1 2013

[37] H. Yu, R. W. Beard, A, and J. Byrne, vision-based collision navigation frame mapping and planning for collision avoidance for miniature air vehicles, *Control Engineering Practice*, (18) 7: 824-836, 7 2010

[38] R. Sharma, J. Saunders, C. N. Taylor, and R. W. Beard, Reactive collision avoidance for fixed-wing MAVs flying in urban terrain, *AIAA Guidance, Navigation, and Control Conference and Exhibit*, 2009

[39] D. B. Kingston, R. W. Beard, and R. Holt, Decentralized Perimeter Surveillance Using a Team of UAVs *IEEE Transactions on Robotics*, (24) 6: 1394-1404, 12, 2008.

[40] D. A. Anisi, Optimal motion control of a ground vehicle, Technical Report FOI-R-0961-SE Swedish Defence Agency, Stockholm, 2003.

[41] L. E. Dubins, On curves of minimal length with a constraint on average curvature and with described initial and terminal positions and tangents, *American Journal of Mathematics*, (79) 3: 497-516, 1957.

[42] D. J. Klein, P. Lee, K. A. Morgansen, and T. Javidi, Integration of Communication and Control using Discrete Time Kuramoto Models for Multivehicle Coordination over Broadcast Networks, *IEEE Journal of Selected Areas in Communications, Special Issue on Control and Communications*, 2007

[43] Y. Kim and M. Mesbahi, On maximizing the second smallest eigenvalue of a state-dependent graph Laplacian, *IEEE Transactions on Automatic Control*, (51) 1: 116-120, 2006.

[44] M. Mesbahi and F.Y Hadegh, Formation flying of multiple spacecraft via graphs, *AIAA Journal of Guidance, Control, and Dynamics*, (24) 2: 369-377, 2000.

[45] A. Fax and R. M. Murray, Information flow and cooperative control of vehicle formations, *IEEE Transactions on Automatic Control*, (49) 9: 1465-1476, 2004.

[46] Z. Lin, B. Francis, and M. Maggiore, Necessary and sufficient graphical conditions for formation control of Unicycles, *IEEE Transactions on Automatic Control*, (50) 1: 121-127, 2005.

[47] R. Sepulchre, D.A. Paley, and N.E. Leonard. Stabilization of planar collective motion: all-to-all communication, *IEEE Transaction on Automatic Control*, (52) 5: 2007.

[48] R. Sepulchre, D.A. Paley, and N.E. Leonard. Stabilization of planar collective motion with limited communication, *IEEE Transaction on Automatic Control*, (53) 3: 2008.

[49] A. G. K. Holmstrom and M. M. Edvall, (2005) User's Guide for tomlab/snopt, [Online]. Available: http://tomlab.biz.

[50] R. A. Horn and C. R. Johnson, *Matrix Analysis*, Cambridge University Press, 2007.

[51] M. De Gennaro and A. Jadbabaie, Decentralized control of connectivity for multi-agent systems, *in Decision and Control, 45th IEEE Conference on*, Sandiego, CA, pp. 3628-3633, 2006.

[52] A. V. Savkin and H. Teimoori, Decentralized formation flocking and stabilization for networks of unicycles, *in Decision and Control, 48th IEEE Conference on*, Shianghai, P.R.China, pp. 984-989, 2009.

[53] D. P. Spanos and R. M. Murray, Robust connectivity of networked vehicles, *in Decision and Control, 43th IEEE Conference on*, Bahamas, pp. 2893-2898, 2004.

[54] M. M. Zavlanos and G. J. Papas, Potential fields for maintaining connectivity of mobile networks, *IEEE Transactions on Robotics*, (23) 4: 812-816, 2007.

[55] R. Dai, J. Maximoff, and M. Mesbahi, Formation of connected network for fractionated spacecraft, *AIAA Guidance, Navigation, and Control Conference*, 2012.

[56] Q. Qiu and K. A.E., Coverage and connectivity control of wireless sensor networks under mobility, *in High Performance Switching and Routing, 2005. HPSR. 2005 Workshop on*, pp. 177-181, 2005.

[57] D. Mclean, Automatic flight control system, Prentice Hall International, Series in Systems and Control Engineering, 1990.

[58] D. Mclean, Gust load alleviation control for aircraft, *Proc. IEE*, (125) 7: 675-685, 1978.

[59] N.Aouf, B. Boulet, and R.M. Botez, $\mathcal{H}_2$ and $\mathcal{H}_\infty$ optimal gust load alleviation for a flexible aircraft, *American Control Conference*, pp. 1872-1876, 2000.

[60] N.Aouf, B. Boulet, and R.M. Botez, Robust gust load alleviation for a flexible aircraft, *Canadian Aeronautics and Space Journal*, (46) 3: 131-139, 2000.

[61] R.M. Botez, I. Boustani, N. Vayani, P.Bigras, and T. Wong Optimal control law for gust load alleviation, *Canadian Aeronautics and Space Journal*, (47) 1, 2001.

[62] J.H. Vincent, A. Emami-Naeini, and N.M. Khraishi Case study comparison of linear quadratic regulator and $\mathcal{H}_\infty$ control synthesis, *AIAA Journal of Guidance, Control, and Dynamics*, (17) 5: 958-965, 1994.

[63] R.J. Niewoehner and I.I. Kaminar Design ofan autoland controllerfor an F-14 aircraft using $\mathcal{H}_\infty$ synthesis, *AIAA Journal of Guidance, Control, and Dynamics*, (19) 3: 656-663, 1996.

[64] D. Mclean, *Automatic flight control system*, Prentice Hall International, Series in Systems and Control Engineering, 1990.

[65] D. Mclean, Gust load alleviation control for aircraft, *Proc. IEE*, (125) 7: 675-685, 1978.

[66] D. Mclean and R. A. Prasad A structural load alleviation control system for a large aircraft, *Transaction of the institute of measurement and control (London)*, (1) 1: 25-37, 1980.

[67] N.Aouf, B. Boulet, and R.M. Botez, $\mathcal{H}_2$ and $\mathcal{H}_\infty$ optimal gust load alleviation for a flexible aircraft, *American Control Conference*, pp. 1872-1876, 2000.

[68] N.Aouf, B. Boulet, and R.M. Botez, Robust gust load alleviation for a flexible aircraft, *Canadian Aeronautics and Space Journal*, (46) 3: 131-139, 2000.

[69] R.M. Botez, I. Boustani, N. Vayani, P.Bigras, and T. Wong, Optimal control law for gust load alleviation, *Canadian Aeronautics and Space Journal*, (47) 1, 2001.

[70] J.H. Vincent, A. Emami-Naeini, and N.M. Khraishi, Case study comparison of linear quadratic regulator and $\mathcal{H}_\infty$ control synthesis, *AIAA Journal of Guidance, Control, and Dynamics*, (17) 5: 958-965, 1994.

[71] R.J. Niewoehner and I.I. Kaminar, Design ofan autoland controllerfor an F-14 aircraft using $\mathcal{H}_\infty$ synthesis, *AIAA Journal of Guidance, Control, and Dynamics*, (19) 3: 656-663, 1996.

[72] S. Suzuki, Simultaneous structure/control design synthesis for aero-servo-elastic system, *Finite Elements in Analysis and Design*, (14), 2-3: 197-208, 1993.

[73] S. Gaulocher, C. Roos, and C. Cumer, Aircraft load alleviation during maneuvers using optimal control surface combinations, *Journal of Guildance, Control, and Dynamics*, (30), 2: 591-600, 2007.

[74] S. Zink, D. Raveh, and D. Mavris, Integrated trim and structural design process for active aeroelastic wing technology, *Journal of Aircraft*, (40), 3: 523-531, 2003.

[75] M. Karpel, Design for active flutter suppression and gust alleviation using state-space aeroelastic modeling, *Journal of Aircraft*, (19), 3: 221-2271, 1982.

[76] O. Eisele, G. Pechlivanoglou, C. N. Nayeri, and C. O. Paschereit, Experimental investigation of dynamics load control strategies using active microflaps on wind turbine blades, *EWEA 2011 Scientific Proceedings* , 3, 2011.

[77] M. Karpel, B. Moulin, and P. C. Chen, Dynamic response of aeroservoelastic system to gust excitation, *Journal of Aircraft*, (42) 5: 1264-1272, 2005.

[78] B. Moulin and M. Karpel Gust load alleviation using special control surfaces, *Journal of Aircraft*, (44) 1: 17-25, 2007.

[79] W. P. Rodden, and E. H. Johnson, MSC/NASTRAN Aeroelastic Analysis User's Guide, *The MacNeal-Schwendler Corp.*, Los Angeles, 1994.

[80] ZAERO, *Version 6.2 Theoretical Manual*, Zona Techgnology, Inc., ZONA 02-12.4, Scottsdale, AZ, 2002.

[81] A. S. Pototzky and B. Perry III, New and existing techniques for dynamic loads analyses of flexible airplanes *Journal of Aircraft*, (23) 4: 340-347, 1986.

[82] D. K. Schmidt, *Modern flight dynamics* McGraw-Hill, New York, 2010.

[83] M. Mashud, M. Ferdous, and S. H. Omee, Effect of spoiler position on aerodynamics characteristic of an airfoil, *International Journal of Mechanical and Mechatronics Engineering IJMME-IJENS*, (12) 6, 2012.

[84] Joint aviation regulations-25, airworthiness standards: Transport category airplanes Vol. Subpart C - Structure.

[85] A. Zole and M. Karpel, Continuous gust response and sensitivity derivatives using state-space models *Journal of Aircraft*, (31) 5: 1212-1214, 1994.

[86] J. Zeng, B. Moulin, R. de Callafon, and M. J. Brenner, Adaptive feed-forward control for gust load alleviation, *Journal of Guidance Control, and Dynanmics*, (33) 3: 862-872, 2010.

[87] A. Wildschek and R. Maier, Integrated adaptive feed-forward control of atmospheric turbulence excited rigid body motions and structural vibrations on a large transport aircraft, *American Control Conference*, Seattle, 2008.

[88] J. W. Stickle and R. C. Henry, Wind-tunnel study to explore the use of slot spoilers to modulate the flap-induced lift of a wing, *NASA Technical Report*, TN D-4664, Langley Research Center, 7, 1968.

[89] D. R. Croom, R. D. Vogler, and J. A. Thelander, Low-speed wind tunnel investigation of flight spoilers as trailing vortex-alleviation devices on an entended-range wide-body tri-jet airplane model, *NASA Technical Report*, TN D-8373, Langley Research Center, 12, 1976.

[90] P. T. Soderman and T. N. Aiken, Full-scale wind-tunnel tests of a small unpowered jet aircraft with a t-tail, *NASA Technical Note*, TN D-6573, Ames Research Center, 11, 1971.

[91] P. Panyakeow, J. Quenzer, K. Morgansen, and M. Mesbahi, Model-predictive gust load alleviation for flexible aircraft using flap-Array, to be submitted.

[92] H. G. Giesseler, M. Kopf, T. Faulwasser, P. Varutti, and R. Findeisen, Gust load alleviation based on model predictive control, *International Forum on Aeroelasticity and structural Dynamics*, 24A, 2013.

[93] H. G. Giesseler, M. Kopf, P. Varutti, T. Faulwasser, and R. Findeisen, Model Predictive control for gust load alleviation, *4th IFAC Nonlinear Model Predict Control Conference*, Noordwijkerhout, NL, pp.27-32, 2012.

[94] S. Haghighat, H. H. T. Liu, and J. R. R. A. Martins, Model-predictive gust load alleviation controller for highly flexible aircraft, *Journal of Guidance, Control, and Dynamics*, (35) 6, 2012.

[95] I. Mitchell. Game of two identical vehicles, Department of Mathematics, Technical Report, Stanford University, 2001.

[96] W. Kwon and S. Han *Receding horizon control: model predictive control for state models*, Springer, New York, 2005.

[97] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. M. Scokaert, Constrained model predictive control: stability and optimality, *Automatica*, (36) 6: 789-814, 2000.

[98] A. Richards and J. P. How, Analytical performance prediction for robust constrained model predictive control, *AIAA Guidance, Navigation, and Control Conference*, AIAA Paper 2004-5110, Providence, 2004.