Title
Optimal Control of Hybrid Systems in Air Traffic Applications

Author
Kamgarpour, Maryam

Publication Date
2011

Peer reviewed|Thesis/dissertation

Optimal Control of Hybrid Systems in Air Traffic Applications

by

Maryam Kamgarpour

A dissertation submitted in partial satisfaction of the
requirements for the degree of
Doctor of Philosophy

in

Mechanical Engineering

in the

Graduate Division

of the

University of California, Berkeley

Committee in charge:

Professor Claire J. Tomlin, Co-chair
Professor John K. Hedrick, Co-chair
Professor Francesco Borrelli
Professor Laurent El Ghaoui
Professor John Lygeros

Fall 2011

Optimal Control of Hybrid Systems in Air Traffic Applications

Maryam Kamgarpour

Abstract

Optimal Control of Hybrid Systems in Air Traffic Applications

by

Maryam Kamgarpour

Doctor of Philosophy in Mechanical Engineering

University of California, Berkeley

Professor Claire J. Tomlin, Co-chair

Professor John K. Hedrick, Co-chair

Growing concerns over the scalability of air traffic operations, air transportation fuel emissions and prices, as well as the advent of communication and sensing technologies motivate improvements to the air traffic management system. To address such improvements, in this thesis a hybrid dynamical model as an abstraction of the air traffic system is considered. Wind and hazardous weather impacts are included using a stochastic model. This thesis focuses on the design of algorithms for verification and control of hybrid and stochastic dynamical systems and the application of these algorithms to air traffic management problems.

In the deterministic setting, a numerically efficient algorithm for optimal control of hybrid systems is proposed based on extensions of classical optimal control techniques. This algorithm is applied to optimize the trajectory of an Airbus 320 aircraft in the presence of wind and storms. In the stochastic setting, the verification problem of reaching a target set while avoiding obstacles (reach-avoid) is formulated as a two-player game to account for external agents' influence on system dynamics. The solution approach is applied to air traffic conflict prediction in the presence of stochastic wind. Due to the uncertainty in forecasts of the hazardous weather, and hence the unsafe regions of airspace for aircraft flight, the reach-avoid framework is extended to account for stochastic target and safe sets. This methodology is used to maximize the probability of the safety of aircraft paths through hazardous weather.

Finally, the problem of optimization of arrival air traffic and runway configuration in dense airspace subject to stochastic weather data is addressed. This problem is formulated as a hybrid optimal control problem and is solved with a hierarchical approach that decouples safety and performance. As illustrated with this problem, the large scale of air traffic operations motivates future work on the efficient implementation of the proposed algorithms.

Contents

List of Figures

Acknowledgments

Let the beauty we love be what we do

Rumi, 1207 AD

To begin, I thank my adviser Claire Tomlin who supported me in every step of my career with her positive attitude and openness in allowing and encouraging me to pursue my interests. From her I learned to be patient and to value every small idea while keeping in mind the big picture. I leave with a wonderful memory that every time I left her office after a meeting, I felt inspired. I thank Prof. Hedrick for a great experience in his lab during one of my early semesters in Berkeley and for teaching me about nonlinear controllability and observability, a topic that I still find very interesting. I am grateful to Prof. Lygeros for his guidance during my visit at ETH Zurich. This short visit at his lab was very beneficial for my academic development. I express my gratitude to Prof. El Ghaoui for always being available to discuss various aspects of research and teaching and for always expressing his attitude and advice openly and truthfully. I thank Prof. Borrelli for teaching me about model predictive control and for inspiring me with his focus and high standards in work. I am grateful to Prof. O'Reilly for his support since the start of my graduate studies and for setting a great example of an amazing teacher through his commitment to clarity and his extreme care for every student. I thank John Robinson, my mentor at NASA Ames, for his continuous encouragement and support.

My PhD research was a collaboration with several amazing colleagues and I would like to express my appreciation to them. Chapter 2 is a joint work with Humberto Gonzalez, Ram Vasudevan and Manuel Soler. Chapter 3 is a joint work with Jerry Ding and Sean Summers. Chapter 4 is a joint work with Vera Dadok and Sean Summers. Chapter 5 is a joint work with Wei Zhang. Several other colleagues enriched my PhD experience with various academic discussions. I thank Allison Ryan, Edgar Lobaton, Carolyn Sparrey, Carolyn White, Bonnie Zhu, Anil Aswani, Gireeja Ranade, Jack Tisdale, Matthias Göerner, George Hines, Jeremy Gillula, Galina Schwartz, Frauke Oldewurtel and Ryo Takei. I also thank Jana Hiraga with whom I volunteered weekly at several schools in Oakland and from whom I learned a lot on how to mentor children.

I acknowledge the Natural Sciences and Engineering Research Council of Canada for five years of scholarship. From the University of Waterloo, I thank Prof. Morris and Prof. Campbell for mentoring me during a wonderful undergraduate research semester on controlling an inverted pendulum. I thank Prof. Davison for an excellent introduction to multivariable control. I am grateful to Prof. McPhee for his support and teaching an interesting and challenging course on dynamics. I appreciate Prof. Jernigan for a very interesting and inspiring course on signals and systems.

In Berkeley, I had the opportunity to live in several cooperative houses and I have many beautiful memories of the spaces and the people I shared the spaces. In particular, I thank Samara and Ryan, for many amazing experiences and for always reminding me to have an open heart and follow it. As I was leaving Iran for Berkeley, my uncle told me that good friends are more important than books. I think I did well!

I thank my family for always giving me the freedom to pursue my desires. I am grateful for my brother who has always been available despite geographical distances, to discuss technical and non-technical subjects. I always admire his optimism and faith in life. I thank my mother who has consistently provided love and reminded me of importance of a well-balanced life. I thank my father for his love, for nurturing a sense of interest and care for all places and people and for his incredible sense of humor in every situation. I am grateful to Zohreh, Ala, Marmar and Hamid, my family in the United States, for their warm and loving homes where I spent many holiday breaks. I thank my dear friend from undergraduate years, Priya, and her warm family with whom I found a second home in Canada. I am fortunate to still receive their love and support from afar.

Last but not least, I thank the Berkeley hills for being there for me every single day of my studies. After a jog or a bike ride on the hills, I would always feel a sense of gratitude, inspiration, relief and excitement to face the remainder of the day and the next...

Chapter 1

Introduction

This thesis is concerned with development of control synthesis tools for hybrid dynamical systems. Several applications from the air traffic control domain motivated the formulations of the theoretical research presented here and have been solved using the numerical techniques developed.

1.1 Air Traffic Control Challenges

Air Traffic Management (ATM) is responsible for safe, efficient and sustainable operation in civil aviation. Since its birth in the 1920s, the ATM system has gradually evolved from its primitive form that consisted of a set of simple operation rules to its current version that is a complex network of sensing, communication and control subsystems. Although various automation systems have been continually introduced, the backbone of the current system was formed during the 1950s when the introduction of radar surveillance and radio communication technologies revolutionized the way the system was operating [1]. After more than half a century, a paradigm shift in the current ATM system is being pursued in order to address the continuous growth of air traffic demand, skyrocketing fuel prices and growing concerns over the environmental impact of air transportation. The proposed developments to accommodate this shift are being addressed in Europe within the framework of the Single European Sky ATM Research (SESAR) [2], and in the United States within the Next Generation (NextGen) Air Transportation systems [3].

It is proposed that the ATM system can be considerably improved by properly incorporating modern sensing and information technologies to enable reliable communication, real-time common situational awareness for pilots and air traffic controllers, and prompt provably safe decision support systems. The NextGen concept advocates for an evolution from the current ground-based navigation system to a satellite-based ATM system, where verbal communi-

cations and ground radar systems are replaced with more reliable and accurate data-link communications and Global Positioning Systems (GPS), so that many traffic control tasks can be handled (semi)-automatically [4]. In addition, the increased automation in conflict detection and resolution would facilitate 4D (space-time) Trajectory Based Operations (TBO), in which individual flights would have the freedom to adjust their trajectories according to real time traffic and weather conditions.

Currently, ATM imposes certain trajectory restrictions, such as flying through a rigid airway structure, in order to guarantee safety and ease the task of air traffic controllers. Some of these restrictions result in non-minimal fuel consumptions and hence higher operative costs and emissions. The new concept of TBO allows optimization of individual aircraft trajectories while ensuring that the airspace is used safely and efficiently. An important problem in implementing the TBO concept is designing trajectories which are optimal with respect to a cost function determined by the pilot or the operating airline and are provably safe in the presence of wind, hazardous weather and other aircraft. A natural modeling framework for aircraft dynamics is a hybrid system abstraction consisting of discrete and continuous states [5, 6, 7]. In this framework, the discrete states represent the flight modes and operating procedures, while the continuous states describe the evolution of aircraft motion. Hence, the task of optimal trajectory design can be formulated as an optimal control problem for a hybrid system subject to constraints of collision and hazardous weather avoidance. It is also important to consider the uncertainty due to wind and weather forecasts in the trajectory planning problem. While a deterministic robust approach should account for the worst-case of the uncertainty and consequently may result in sub-optimal and very conservative trajectory design, a stochastic approach accounts for the uncertainty by designing trajectories that have a probability of safety above a desired threshold.

A second area of improvement proposed by NextGen is regulation of air traffic operations in super dense airspace in the proximity of airports. In airports with multiple intersecting runways, such as the John F. Kennedy International Airport (JFK) in New York city, the set of active runways, referred to as a runway configuration for arrival or departure, is chosen from a larger set based on factors including the crosswind and tailwind magnitudes, visibility, traffic flow and noise abatement laws. The choice of runway configuration in a dense airspace affects the arrival routes of incoming traffic to all nearby airports. For example, the arrival routes of LaGuardia and Newark airport may be modified to accommodate for the JFK runway configuration. Currently, unanticipated runway configuration switches not only increase the workload of air traffic controllers and pilots, but also result in many approaching aircraft being put into holding patterns. Consequently, the capacity lost during the transitional period of a runway configuration switch is referred to as "perishable capacity" and the delays are propagated into the airspace far beyond the terminal area. It is a vision of NextGen to use available weather and traffic forecast data in order to optimally schedule the runway configuration, anticipate the required switching time between the configurations and control the arrival traffic to minimize delays. For the problem of configuration planning, an

appropriate mathematical model is a hybrid system abstraction in which the runway configuration represents the discrete modes of the system and the aircraft configuration in the airspace of interest represents the continuous state. Optimal runway scheduling subject to weather constraints can then be posed as a hybrid optimal control problem.

1.2 Background on Control of Hybrid Systems

Hybrid dynamical models arise in systems in which discrete events interact with the continuous state evolution. One of the motivating application domains for the introduction of hybrid models is the field of embedded systems in which software, represented as a finite-state machine, interacts with a continuous physical process. Ever since their introduction, hybrid models have been used for a large set of engineering problems, including air traffic management [8, 5, 9], autonomous vehicle motion planning [10, 11], automotive control [12], robotics [13], manufacturing systems [14], systems biology [15, 16, 17] and bipedal walking [18]. In these applications, the behavior of the system can be described in terms of an abstraction in which the discrete state, also referred to as a mode, can capture qualitative behavior, for example the operating modes of a flight management system or the foot impact of a bipedal walker, while the continuous state can capture quantitative characteristics such as the velocity and heading of the aircraft or the joint angles of a biped.

The hybrid modeling framework combines the discrete event dynamical modeling approach with the differential or difference equation modeling approach from classical control theory. As such, analysis and control synthesis methods can be categorized based on the approaches typically used for these systems. In the first approach, model checking and deductive theorem proving from automata theory have been effective for certain classes of hybrid systems with simple continuous dynamics, such as timed automata and linear hybrid automata [19, 20, 21, 22]. In order to apply these tools to more complex dynamics, approximate abstractions of the system may be required [23]. In the second approach, tools for stability analysis, reachability and control synthesis from classical control theory have been extended in order to address verification and control of systems with multiple modes of operation [24]. In general, additional assumptions on the discrete dynamics are imposed, such as the assumption of a finite number of switches in finite time, and numerical tools have been restricted to certain classes of hybrid systems, such as those with linear continuous dynamics in each discrete mode.

Deterministic Hybrid Optimal Control

In order to address optimal control of hybrid systems from the control theoretic approach, several researchers have extended the optimality conditions which were developed by Bell-

man and Pontryagin for optimal control of nonlinear dynamical systems [25, 26, 27]. These optimality conditions originated from the fields Calculus of Variations and Mathematical Optimization during the space exploration era of 1950's and have formed the backbone of the field of optimal control theory. While in the former Soviet Union, the work of Pontryagin resulted in extensions of the Calculus of Variations into a set of necessary optimality conditions [28], in the United States, dynamic programming developed by Bellman extended the earlier work of Hamilton and Jacobi on classical physics and provided a set of sufficient optimality conditions for the discrete-time optimal control problems [29], while Kalman provided the continuous-time counterpart [30] and Isaac addressed the game formulation [31]. Although theoretical results have been developed based on extensions of these necessary and sufficient optimality conditions for hybrid systems, in practice, unless additional assumptions such as linearity are made on the continuous dynamics, computation of an optimal control based on these optimality conditions is difficult. Hence, development of efficient numerical methods for optimal control of general hybrid dynamical systems is an active area of research.

Stochastic Hybrid Optimal Control & Verification

In many physical processes there are uncertainties in the evolution of the system dynamics, either due to the presence of random noise or due to modeling imperfections. In addition, the evolution of the discrete and continuous state may be modeled probabilistically through analysis of statistical data. For such cases, a natural modeling framework is that of a Stochastic Hybrid System (SHS) [32, 33, 6, 34]. In a continuous-time stochastic hybrid system, research has focused on establishing certain desired properties of the model such as existence of the solution process and the Markov property. These properties have been derived for various classes of stochastic hybrid systems such as piecewise deterministic Markov processes [35, 36], switching diffusions [37] and general stochastic hybrid systems [38, 39]. The optimal control problem in the stochastic setting can be formulated in terms of optimization of the expectation of an objective function of the state and inputs. Methods based on dynamic programming have been shown to hold for a general class of stochastic hybrid systems and the objective function has been characterized as a solution of a coupled Hamilton-Jacobi-Bellman equation [39]. Computing numerical solutions of these equations and obtaining an optimal control input remain as open problems.

In addition to optimization of an objective function, an important problem in systems analysis and control is verifying safety, that is, the trajectory of the system remains inside a safe set, and reachability, that is, the trajectory reaches a desired target set. This problem has been well-studied for deterministic dynamical systems and its connection to optimal control theory has been established [40, 41]. In addition, numerical tools have been developed to compute the set of initial conditions which satisfy the reachability and safety properties [42, 43]. In a stochastic framework, given that the state trajectory is probabilistic, safety and target attainability are also characterized probabilistically. Thus, in this case the control

synthesis problem is concerned with maximizing the probability of safety or reachability of system trajectories. While for certain classes of stochastic hybrid systems the reachability problem has been addressed either analytically [38] or computationally [44, 45], for a general stochastic hybrid model in continuous-time, reachability results are not yet well-understood [38]. Given that measurability results are easier to establish in discrete-time, research has focused on discrete-time stochastic hybrid systems. Recently, research has explored safety, and reachability verification, and control synthesis for this class of systems [46, 47]. Extensions of results to more general dynamical models, developing fast numerical methods and application of these methods to realistic problems are topics of ongoing research.

1.3 Contributions and Organization

In this thesis, we develop algorithms for optimal control of deterministic hybrid systems (Chapter 2) and stochastic hybrid systems (Chapter 3 and 4). We also develop a framework for optimizing arrival traffic and runway configuration planning (Chapter 5). The details of the content of each chapter are as follows.

In Chapter 2, we focus on deterministic hybrid systems. Having a general and unified viewpoint of hybrid systems makes development of an efficient numerical algorithm challenging. Thus, we focus on nonlinear switched dynamical systems. These represent a class of hybrid systems in which the continuous state does not exhibit jumps during discrete mode switches. We assume the switches between discrete modes are controlled, that is, a control input can choose the mode of operation. In our optimal control framework we include constraints on inputs and states which may arise due to the physical limitations or requirements on the system. Given that we account for constraints and nonlinear dynamics, our formulation is general enough to apply to several engineering problems. We develop a computationally feasible algorithm for addressing the optimal control of this class of problems. We illustrate the applicability of the algorithm with a trajectory planning problem motivated by the TBO concept. In our approach, different flight modes and operational procedures are combined with the continuous dynamics of a realistic aircraft model of Airbus 320 in order to design fuel optimal safe trajectories. The theoretical material in this chapter was presented in [48, 49, 50, 51], while the aircraft trajectory planning case study appeared in [52].

In Chapter 3 we consider a more general hybrid modeling framework in which we account for stochastic disturbances in the evolution of the continuous and discrete states. In addition, we account for deterministic disturbances in the model. The motivation is that while some classes of uncertainties, such as those by nature, are best modeled stochastically, some other classes of uncertainties, such as those due to presence of agents with competing objectives, are best modeled in the deterministic worst-case approach. For example, in a collision avoidance scenario between two aircraft, on the one hand, wind affects the dynamics of aircraft and the uncertainties in wind may be best accounted for through a stochastic framework.

On the other hand, in the absence of communication between the aircraft, from the perspective of each aircraft the trajectory must be safe in the worst-case performance of the other aircraft. Hence, a robust approach should be considered. We formulate a stochastic hybrid dynamic zero-sum game between the control and the disturbance to address both classes of uncertainties. To alleviate measurability difficulties, we consider our formulation in discrete time. We develop an algorithm for synthesizing a control law that maximizes the probability of safety and target attainability subject to worst-case deterministic disturbance performance. We then address the collision prediction and resolution between two aircraft using this proposed algorithm. The result is characterization of the maximum probability of safety for any initial relative state of the two aircraft and a control policy for each aircraft which achieves this probability under any physically realizable trajectory of the other aircraft. The material in this chapter appeared partially in [53] and is in preparation for submission [54].

In Chapter 4 we consider several extensions of the stochastic hybrid dynamic game framework. First, we account for uncertain obstacles, such as those arising from forecasts of hazardous weather, by modeling them as stochastic sets. We show how the verification and control synthesis methodology for stochastic hybrid systems can be generalized to address this problem. This theory is used to solve an aircraft trajectory planning problem in which the objective is maximizing the probability of reaching a waypoint in the airspace while avoiding hazardous weather. Next, we consider reachability in infinite horizon and develop theoretical conditions under which the algorithm proposed in the previous chapter converges. We apply this analysis to a pursuit-evasion game between a quadrotor helicopter and a ground vehicle in which the objective of the helicopter is target attainability while maintaining its position and velocity within certain safety bounds. The material in this chapter is based on the papers presented in [55, 56] and a paper in preparation [57].

In Chapter 5 we visit the problem of runway configuration management in super dense terminal airspaces. Here, our objective is determination of optimal runway configuration sequences and switching times to ensure safety of landing aircraft and to minimize delays and holding patterns. We develop an accurate model for the arrival traffic dynamics in terminal airspace that takes into account weather uncertainties and runway configuration changes. We model the air traffic control problem as an optimal control of a constrained hybrid system. Then, we introduce a hierarchical algorithm for solving this particular problem which can reduce complexity. We apply our results to a runway scheduling problem in the JFK airspace. The material in this chapter was presented in [58, 59].

Finally, in Chapter 6 we outline some of our future work on the algorithm developments based on the needs of realistic air traffic management scenarios.

Chapter 2

Optimal Control of Deterministic Constrained Switched Systems

We develop a numerical method for addressing the optimal control problem for constrained switched nonlinear systems. First, we describe the problem and review the related work. Then, we present our solution approach and an application of our resulting algorithm in an aircraft trajectory planning problem. The material in this chapter was presented in [48, 49, 50, 51, 52].

2.1 Background

Switched systems consist of a finite number of dynamical subsystems and a switching law that describes which subsystem is active at a given time. In each subsystem, also referred to as a mode, the evolution of the state is described by a set of differential or difference equations. Switched systems usually refer to the class of hybrid systems in which the discrete mode transitions are either triggered by an external input, or by the continuous states reaching certain thresholds, and in which there are no discontinuities (jumps) in the state at the switch times. Many hybrid models encountered in practice, such as automobiles and locomotives with different gears [12, 60, 61, 62], DC-DC converters [63, 64], and biological systems [16, 65] may be modeled as switched systems. In addition, complex nonlinear dynamics can be decomposed into modes of operation, hence giving rise to switched systems, so that analysis and controller design is simplified [11, 10].

In the most general form, optimal control of a switched system involves finding a mode sequence, switch times between the modes and an input for each mode such that a cost function is optimized while certain constraints on states and inputs are satisfied. Existence of solutions to the optimal control problem has been addressed for various formulations of

cost functions [66, 26]. Branicky's seminal work, which presented many of the theoretical underpinnings of hybrid systems in their most general form, also included a set of sufficient conditions for the optimal control of such systems under an infinite horizon discounted cost formulation [26]. However, computation of an optimal control based on these sufficient conditions is difficult because it requires solving for a continuously differentiable value function which satisfies a set of quasi-variational inequalities. Due to computational complexity, other approaches that are based on the dynamic programming principle are applied to switched systems either through approximations [67] or by making assumptions, such as linear or affine continuous-state dynamics in each discrete mode [68, 69, 70].

Several researchers developed necessary optimality conditions for various classes of switched systems based on variational analysis such as the Maximum Principle [71, 72, 25]. However, numerical computation of the optimal control based on these necessary conditions is difficult. For a fixed mode sequence, the determination of optimal switch times and input to each mode can be addressed with classical Maximum Principle tools [27], or through extension of the state to include switch times as part of an extended state [73]. Thus, one can formulate an iterative two-stage algorithm in which at one stage, the optimal control problem for a fixed mode sequence is solved and then at the other stage, the mode sequence is varied. The process would be repeated until some desired convergence is met. The challenge then would be in defining variation of the mode sequence in a computationally feasible way. While variations based on Hamming distance [27] or enumeration of mode sequences [74] are proposed, the trajectories generated from modification of a given mode sequence with such approaches are not analytically comparable unless the optimal control problem is solved for each of the candidate mode sequences; a task that is computationally complex. In [75], a method is proposed to alleviate this problem for autonomous unconstrained switched systems. Here, instead of resorting to brute force search, the sequence is updated by inserting a single mode and computing the variation of the cost function as the duration of the inserted mode approaches zero. This work inspired the development of our algorithm.

We consider a constrained nonlinear switched system with controlled switches, that is, switches which are determined by a control input rather than by the state reaching prescribed regions of state space. We develop a two-stage algorithm that divides the problem into two nonlinear constrained optimization problems. Given a fixed mode sequence at one stage, the mode duration and the input in each mode is optimized, through formulating a conventional optimal control problem. At another stage, the mode sequence is varied through inserting a single mode. We analytically characterize the variation of the cost and the constraint functions due to this mode insertion. Consequently we are able to analytically compare the cost resulted from the new mode sequence to that of the original one, through solving an optimization problem, rather than solving the optimal control problem associated with each mode sequence. The algorithm is used for realistic trajectory planning for an Airbus 320 aircraft. Here, the discrete modes represent maneuvers and the constraints are due to airspace safety requirements.

2.2 Problem Statement and Solution Approach

We consider a switched dynamical system consisting of a set of subsystems, indexed by $Q = \{1, 2, \ldots, M\}$, such that for each $q \in Q$, $f_q : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$ is a vector field that describes the system dynamics in mode q. The system undergoes N switches in the time interval $[0, t_f]$ at switch times $s_1, s_2, \ldots, s_N$. The evolution of the state $x \in \mathbb{R}^n$ is given by

$$\begin{aligned} \dot{x}(t) &= f_{\sigma_i}(x(t), u(t)), \quad t \in [s_i, s_{i+1}) \\ x(0) &= x_0. \end{aligned} \tag{2.1}$$

In the above, $\sigma_i \in Q$, $i = 0, 1, \ldots, N$, and we have defined $s_0 = 0$, $s_{N+1} = t_f$. We denote the feasible switch times, also referred to as the switch time vector, as

$$\mathcal{S} = \{s \in \mathbb{R}^N : 0 \leq s_1 \leq s_2 \cdots \leq s_N \leq t_f\}. \tag{2.2}$$

We assume no jump in the state during a switch, that is, $\lim_{t \uparrow s_i} x(t) = \lim_{t \downarrow s_i} x(t)$ for $i = 1, 2, \ldots, N$. Let U be a compact subset of $\mathbb{R}^m$. The input u belongs to set $\mathcal{U}$ defined as

$$\mathcal{U} = \{\mu : [0, t_f] \to U \,|\, \mu \text{ is measurable}\}.$$

In contrast to the mode sequence $\sigma = (\sigma_0, \sigma_1, \ldots, \sigma_N)$ which is a discrete input, μ is usually referred to as a continuous input. To avoid confusion on continuity requirements of μ, we refer to μ as a non-discrete or modal input.

The hybrid optimal control problem of our interest is stated as follows.

Problem 2.1. Given the switched system (2.1) whose state and inputs are subject to a set of N_c constraints for $t \in [0, t_f]$ defined by

$$h_j(x(t)) \leq 0, \quad j = 1, 2, \ldots, N_c. \tag{2.3}$$

Find the number of modes N, the sequence of modes σ, the switch time vector $s \in S$ and the input $\mu \in \mathcal{U}$ in order to minimize the cost function defined as

$$J(\sigma, s, \mu) = \sum_{i=0}^{N} \int_{s_i}^{s_{i+1}} l_{\sigma_i}(x(\tau), \mu(\tau))\, d\tau + g(x(t_f)). \tag{2.4}$$

In the above, $l_{\sigma_i} : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$, $i = 0, 1, \ldots, N$ are mode-dependent running costs and $g : \mathbb{R}^n \to \mathbb{R}$ is the final cost. We assume that f_q, h_j, l_q, and g are Lipschitz for all $q \in Q$, $j = 1, 2, \ldots, N_c$. This assumption is sufficient to ensure existence and uniqueness of solutions to the differential equations. In addition, we assume these functions are differentiable and their derivatives are also Lipschitz in their arguments. This assumption is sufficient for deriving necessary optimality conditions.

Optimality Criteria

We show that the two main analytical approaches for solving the optimal control problem, the Maximum Principle and the Hamilton-Jacobi-Bellman Partial Differential Equation, apply to the optimal control of unconstrained switched nonlinear systems formulated above. In order to apply these results in our problem setting, we combine the discrete and non-discrete inputs by introducing the input $\alpha = (\beta, \mu)$, in which the function $\beta : [0, t_f] \to Q$ returns the mode of the system at a given time, that is, $\beta(t) = \sigma_i$ for $t \in [s_i, s_{i+1})$. Let $A = Q \times U$ and denote the admissible input space as $\mathcal{A} = \{\alpha : [0, t_f] \to A | \, \alpha \text{ is measurable}\}$. The dynamics in Equation (2.1) can be written as

$$\begin{aligned} \dot{x}(t) &= f(x(t), \alpha(t)), \\ x(0) &= x_0. \end{aligned} \tag{2.5}$$

Here, $f : \mathbb{R}^n \times A \to \mathbb{R}^n$. The cost in the new control variable can be written as

$$J(\alpha) = \int_0^{t_f} l(x(\tau), \alpha(\tau)) \, d\tau + g(x(t_f)). \tag{2.6}$$

For expressing the optimality conditions, to simplify notation, we consider a non-constrained optimal control problem with the objective of minimizing (2.6) subject to dynamics (2.5).

Maximum Principle

The Maximum Principle provides necessary conditions that control variables and the resulting optimal state trajectory must satisfy. The proof is based on the idea of needle-type variation of the input (as opposed to a first-order variation) and hence is applicable to input spaces with finitely many elements such as the space Q considered here. The optimality conditions are described by first defining the control Hamiltonian $H(x, p, a)$, for $x, p \in \mathbb{R}^n$ and $a \in A$, as follows:

$$H(x, p, a) = p^T f(x, a) + l(x, a). \tag{2.7}$$

Theorem 2.1 (Pontryagin Maximum Principle)**.** *Assume $\alpha(t)$ is an optimal input for problem (2.6), subject to dynamics (2.5), and that $x(t)$ is the corresponding optimal state trajectory. Then, there exists a function $p : [0, t_f] \to \mathbb{R}^n$ that satisfies the following dynamics for almost all $t \in [0, t_f]$:*

$$\begin{aligned} \dot{p}(t) &= -\frac{\partial H}{\partial x}^T (x(t), p(t), \alpha(t)) p(t), \\ p(t_f) &= \frac{\partial g}{\partial x}^T (x(t_f)). \end{aligned} \tag{2.8}$$

In addition, the optimal input satisfies

$$\alpha(t) = \arg\min_{a \in A} H(x(t), p(t), a). \tag{2.9}$$

The proof of the above result appears in [28, 76]. For a survey on historical development of the Maximum Principle and extensions to state constraints please see [77] and [78], respectively. Equations (2.5) and (2.8) form a two point boundary value problem which must be satisfied by the optimal input and the optimal state trajectory. Equation (2.9) reduces the problem of functional minimization for finding $\alpha : [0, t_f] \to A$ to a pointwise minimization for finding $\alpha(t)$ at each instant of time. In certain problems, these necessary conditions are sufficient to uniquely define the input. In some other cases, they can be used as an *a posteriori* check on optimality of a candidate solution.

Computationally, if the discrete mode sequence is absent, then an iterative method can be employed in which a candidate input is used to solve the forward differential equation (2.5) and then the backward differential equation (2.8) is solved. Alternatively, (2.5) and (2.8) may be solved simultaneously by taking an initial guess on final or initial value of the state or costate, respectively. The input is updated either based on the gradient of the cost function (gradient methods), or through pointwise minimization of (2.9) (shooting methods) [79]. In general, for convergence, the gradient methods are slow while the shooting methods are very sensitive to initial guess. Although numerical tools have been proposed to take into account discrete modes [71], such methods must *a priori* assign the mode switch times to discretized time points. In addition, in the absence of gradient of cost function with respect to the mode sequence, only shooting methods can be used in this case. Additionally, similar to the non-hybrid case, accounting for state constraints is numerically difficult.

Hamilton-Jacobi-Bellman equation

The idea here is to transform the optimal control problem into a larger class of problems in which the initial state x_0 and the initial time t_0 are also optimization variables. Then, if there exists a function $v : \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}$, referred to as the value function, which satisfies the so-called Hamilton-Jacobi-Bellman (HJB) partial differential equation, this function is the optimal cost for the more general optimal control problem starting at time $t \in [t_0, t_f]$ with $x(t) = x \in \mathbb{R}^n$. For $t \geq t_0$, define $A_t = \{\alpha : [t, t_f] \to A | \, \alpha \text{ is measurable}\}$ and the value function as

$$v(x, t) := \inf_{\alpha \in A_t} \int_t^{t_f} l(x(\tau), \alpha(\tau)) \, d\tau + g(x(t_f)).$$

Theorem 2.2 (Hamilton-Jacobi-Bellman optimality condition)**.** *Assume that v is a continuously differentiable function of the variables (x, t). Then v solves the nonlinear partial differential equation*

$$\frac{\partial v}{\partial t}(x, t) + \min_{a \in A} \{ \frac{\partial v}{\partial x}(x, t) f(x, a) + l(x, a) \} = 0, \tag{2.10}$$

initialized with $v(x, t_f) = g(x)$. In addition, if $\alpha(x, t)$ achieves the minimum above, then α is an optimal input.

Several books derive these sufficient conditions for various classes of optimal control problems, see for example, [29, 30]. In contrast to the Maximum Principle which results in open loop inputs, the input obtained from solving the above optimization problem is in feedback form. In addition, the costate found from the Maximum Principle is identified with the partial derivative of the value function along the optimal trajectory: $\frac{\partial v}{\partial x}(x, t) = p(t)$.

In general, it is very difficult to solve for the value function v, analytically. In addition, there may not exists a continuously differentiable function that satisfies the HJB equations. In these cases, a relaxed solution to the partial differential equations above, referred to as a viscosity solution, is defined [80, 81] and shown to be the appropriate notion of solution for the optimal control problem. Although for certain classes of problems such as the Linear Quadratic Regulator (LQR), analytical results for the value function and the optimal input exist, once the discrete input variables are introduced, it is no longer easy to characterize the value function or the optimal input even for the switched LQR models. Hence, approximations of the HJB sufficient conditions are derived [82]. In addition, accounting for state constraints remains a challenge in numerical application of the HJB methods.

Two-Stage Solution Approach

The challenge with determining the discrete modes in the hybrid optimal control problem is that the trajectories obtained from variations of a given mode sequence may be far from the nominal one and not comparable in a computationally efficient manner. However, if one considers a variation in which the modified sequence differs from the original one by modes whose durations are sufficiently small, one can then analyze the differences between the resulting trajectory and the original trajectory and consequently their associated cost and constraint functions. Thus, starting with a nominal sequence of modes, additional modes may be added in such way that we can prove every additional mode will reduce the cost while maintaining feasibility. This defines our two-stage solution approach as stated below:

Stage 1. Given a mode sequence, $\sigma = (\sigma_0, \sigma_1, \ldots, \sigma_N)$, find the optimal input μ and the optimal switch time vector $s = (s_1, \ldots, s_N)$.

Stbge 2. Form a new sequence $\hat{\sigma}$ as a result of insertion of a mode $q \in Q$ into the original sequence σ, which would decrease the cost while maintaining feasibility. If such a mode cannot be found, stop. Else, repeat Stage a using $\hat{\sigma}$.

The above procedure leads to suboptimal solutions since only certain variations of the discrete mode sequence, that is, mode insertions, are considered. However, it provides a systematic

and computationally efficient manner of examining candidate mode sequences without solving the optimal control problem for each candidate mode sequence. In the next two sections, we describe how to address each of the above two stages of the algorithm.

2.3 Stage 1 - Fixed Mode Sequence

In many switched systems the sequence of modes is either fixed *a priori* or is determined from knowledge of the dynamical system and the cost function. Moreover, as described previously, in many approaches for solving the optimal control problem for switched systems, first a fixed mode sequence is assumed and then methods for varying this mode sequence to decrease the cost are used. With a fixed mode sequence, Problem 2.1 is simplified as follows:

Problem 2.2. Given the switched system (2.1) subject to constraints (2.3) for $t \in [0, t_f]$, and a nominal mode sequence $\sigma = (\sigma_1, \sigma_2, \ldots, \sigma_N)$, find the switch time vector $s \in S$ and the input $\mu \in \mathcal{U}$ in order to minimize the cost function

$$J(s, \mu) = \sum_{i=0}^{N} \int_{s_i}^{s_{i+1}} l_{\sigma_i}(x(\tau), \mu(\tau))\, d\tau + g(x(t_f)). \tag{2.11}$$

Developing analytical or numerical solutions for the optimal control of switched systems with a fixed mode sequence has been explored in several previous research projects [83, 84, 85, 86, 87]. We present two of the main approaches for solving the above optimal control problem. First, we briefly review the approach for transforming the problem into a conventional optimal control problem as proposed in [88, 73]. Next, we present our approach, which is based on characterizing the variation of the cost function with respect to variation of the switch times. This latter approach serves as a starting point for Stage 2 of the hybrid optimal control procedure described above, in which variation of the cost function with respect to a mode insertion is derived.

Transformation Method

The idea here is to convert the optimal control problem with unknown switch times into an equivalent optimal control problem with an extended state and known switch times. As in [88], we describe the general case in which the final time may also be an optimization variable.

The first step is to introduce the new states, $x_{n+1}, \ldots, x_{n+N+1}$, corresponding to the switch times s_i for $i = 1, 2, \ldots, N$, and the final time $s_{N+1} = t_f$. Then, $x_{n+i} = s_i$, and $\dot{x}_{n+i} = 0$, for $i = 1, 2, \ldots, N+1$. We also introduce a new independent variable $\tau \in [0, N+1]$. The relation

between τ and t is as follows: $t = x_{n+1}\tau$ for $\tau \in [0,1]$, and $t = x_{n+i+1}(\tau - i) - x_{n+i}(\tau - i - 1)$ for $\tau \in [i, i+1]$, $1 \leq i \leq N$.

For a function $f(t)$, let f' denote the derivative with respect to the new independent variable, τ. Next, define $\hat{f}_{\sigma_0} = x_{n+1} f_{\sigma_0}$, and $\hat{f}_{\sigma_i} = (x_{n+i+1} - x_{n+i}) f_{\sigma_i}$ for $i = 1, \ldots, N$.

The equivalent optimal control problem in the extended state space is as follows:

$$\begin{aligned} \min \quad & \sum_{i=0}^{N} \int_{i}^{i+1} l_{\sigma_i}(x(\tau), \mu(\tau))\, d\tau + g(x(N+1)) \\ \text{s.t.} \quad & x'(\tau) = \hat{f}_{\sigma_i}(x(\tau), \mu(\tau)),\ \tau \in [i, i+1] \\ & x'_{n+1+i}(\tau) = 0,\ i = 0, \ldots, N \\ & h_j(x(\tau)) \leq 0,\ j = 1, \ldots, N_c \end{aligned} \tag{2.12}$$

Define $\hat{x} := (x_1, \ldots, x_n, x_{n+1}, \ldots, x_{n+1+N})$ as the extended state. In the optimal solution of the above problem, $(\hat{x}^*, u^*)$, the last $N+1$ components of the state $\hat{x}^*$, are the N optimal switch times and the final time. Since the duration of each mode is constant in the introduced transformation, the new equivalent problem is a conventional optimal control problem, that is, an optimal control problem without varying switch times.

In order to numerically solve the conventional optimal control problem, the Maximum Principle may be used to formulate necessary optimality conditions resulting in a two point boundary value problem (indirect method) [79]. Alternatively, by appropriate discretization of control input and constraints, the optimal control problem may be approximated as a finite dimensional nonlinear program (direct method) [89, 90, 91, 88, 79]. Even though direct methods only ensure local optimality, due to their computational efficiency they have been widely used for solving optimal control engineering problems, such as aircraft and aerospace trajectory planning [92].

First-order Method

An alternative method for finding the switch times and the modal input is through formulating first-order necessary optimality conditions for the switch times by finding the derivative of the cost function with respect to the switch times. As such, here we develop an analytic formula for this derivative.

The problem of finding the derivative with respect to switch times was considered for autonomous switched systems in [86]. Approximations of the derivative were derived by assuming certain variations of input [83] or formulating differential algebraic equations [85].

Let $D_s J(s, \mu)$ denote the derivative of the cost function with respect to the switch times. Since in many switched systems, μ may not be continuous at switch times, we allow for

the discontinuity of input in deriving the formula. As such, for any $w : \mathbb{R} \to \mathbb{R}^n$ we define $w(t^-) = \lim_{\tau \uparrow t} w(\tau)$ and $w(t^+) = \lim_{\tau \downarrow t} w(\tau)$ as the limits, from the left and the right respectively, of the signal w at time t. In addition, at a switch time s_i we define

$$\begin{aligned} \Delta f(s_i^-) &= f_{\sigma_{i-1}}(x(s_i), \mu(s_i^-)) - f_{\sigma_i}(x(s_i), \mu(s_i^-)), \\ \Delta f(s_i^+) &= f_{\sigma_{i-1}}(x(s_i), \mu(s_i^+)) - f_{\sigma_i}(x(s_i), \mu(s_i^+)). \end{aligned} \tag{2.13}$$

Similarly, we define $\Delta l(s_i^-)$ and $\Delta l(s_i^+)$.

To proceed with the derivative formula, we first assume variation of only one switch time, s_i, in the switch time vector $s = (s_1, s_2, \ldots, s_N)$. We denote $\nabla_{e_i} J(s, \mu)$ as the directional derivative of $J(s, \mu)$ in the direction of unit vector $e_i \in \mathbb{R}^N$:

$$\nabla_{e_i} J(s, \mu) = \lim_{\lambda \downarrow 0} \frac{J(s + \lambda e_i, \mu) - J(s, \mu)}{\lambda}.$$

The main result is that the directional derivative is characterized as follows:

Proposition 2.1. *The directional derivative of the cost function in direction e_i for $i = 1, 2, \ldots, N$ exists when $s_i < s_{i+1}$ and is given as*

$$\nabla_{e_i} J(s, u) = p^T(s_i) \Delta f(s_i^+) + \Delta l(s_i^+),$$

where the costate $p(t)$, $t \in [0, t_f]$ satisfies the following switched system dynamics:

$$\begin{aligned} \dot{p}(t) &= -\frac{\partial f_{\sigma_i}}{\partial x}^T (x(t), \mu(t)) p(t) - \frac{\partial l_{\sigma_i}}{\partial x}^T (x(t), \mu(t)), \quad t \in [s_{i+1}, s_i) \\ p(t_f) &= \frac{\partial g^T(x(t_f))}{\partial x}, \end{aligned} \tag{2.14}$$

for $i = 0, 1, \ldots, N$ and at the switch times $p(s_i^-) = p(s_i^+)$.

Proof. The proof proceeds by first considering only a final cost, $J(s, \mu) = g(x(t_f))$, and then generalizing the result for the running cost and the final cost.

The directional derivative of the final cost in direction e_i is found using the Chain Rule as $\nabla_{e_i} g(x(t_f)) = \frac{\partial g^T(x(t_f))}{\partial x} \nabla_{e_i} x(t_f)$. In order to find $\nabla_{e_i} x(t_f)$, we make the dependence of the state on s explicit by introducing the flow $\psi(t, s) : \mathbb{R} \times \mathbb{R}^N \to \mathbb{R}^n$ as the solution of the differential equation at time $t \in [0, t_f]$ for a switch time vector $s \in S$. As such, $\psi(t, s + \lambda e_i)$ denotes the solution of the differential equation with the same initial condition, however, with the switch from mode σ_{i-1} to mode σ_i occurring at time $s_i + \lambda \leq s_{i+1}$ instead of s_i, where $\lambda > 0$ is sufficiently small. Now, since $\psi(t, s)$ and $\psi(t, s + \lambda e_i)$ satisfy the same initial condition and the same differential equation for $t \in [0, s_i)$, they are equal on $[0, s_i)$ by

uniqueness of the solution. For $t \in [s_i, s_i + \lambda)$, $\psi(t, s)$ is governed by f_{σ_i} while $\psi(t, s + \lambda e_i)$ is governed by $f_{\sigma_{i-1}}$. Hence,

$$\psi(s_i + \lambda, s + \lambda e_i) = \psi(s_i + \lambda, s) + \int_{s_i}^{s_i+\lambda} \big(f_{\sigma_{i-1}}(x(\tau), \mu(\tau)) - f_{\sigma_i}(x(\tau), \mu(\tau))\big) d\tau. \quad (2.15)$$

For $t \in [s_i + \lambda, t_f]$, $\psi(t, s)$ and $\psi(t, s + \lambda e_i)$ satisfy the same differential equation again. Define $\Delta x_0 = \psi(s_i + \lambda, s + \lambda e_i) - \psi(s_i + \lambda, s)$. For a fixed u the vector field $f_q(x(t), \mu(t))$ can be considered as an autonomous vector field $f_q(x(t), t)$. As such, we apply the theorem on perturbation analysis of differential equations in Appendix B.2 of [93] to find variation of the flow for $t \in [s_i + \lambda, t_f]$ with respect to variation in the initial condition as

$$\psi(t, s + \lambda e_i) - \psi(t, s) = \phi(t, s_i + \lambda)\Delta x_0 + o(\Delta x_0),$$

where $o(\Delta x_0)$ denotes the higher order term:

$$\lim_{\|\Delta x_0\| \to 0} \frac{o(\Delta x_0)}{\|\Delta x_0\|} = 0, \quad (2.16)$$

and $\phi(t, s_i + \lambda) \in \mathbb{R}^{n \times n}$ satisfies the following matrix differential equation for $t \in [t_0, t_f]$:

$$\begin{aligned} \dot{\phi}(t, t_0) &= \frac{\partial f_{\sigma_i}}{\partial x}(x(t), \mu(t))\phi(t, t_0), \quad t \in [s_i, s_{i+1}) \\ \phi(t_0, t_0) &= I_{n \times n}, \end{aligned}$$

with $t_0 = s_i + \lambda$. We compute the directional derivative of $\psi(t, s)$:

$$\nabla_{e_i}\psi(t, s) = \lim_{\lambda \downarrow 0} \frac{\psi(t, s + \lambda e_i) - \psi(t, s)}{\lambda} = \lim_{\lambda \downarrow 0} \frac{\phi(t, s_i + \lambda)\Delta x_0}{\lambda} + \lim_{\lambda \downarrow 0} \frac{o(\Delta x_0)}{\lambda}. \quad (2.17)$$

The term $\phi(t, s_i + \lambda)$ approaches $\phi(t, s_i)$ as $\lambda \downarrow 0$ from continuous dependence of $\phi(t, t_0)$ on the initial time t_0 [93]. From the definition of Δx_0, we have,

$$\frac{\Delta x_0}{\lambda} = \frac{\int_{s_i}^{s_i+\lambda} \big(f_{\sigma_{i-1}}(x(\tau), \mu(\tau)) - f_{\sigma_i}(x(\tau), \mu(\tau))\big) d\tau}{\lambda},$$

which approaches $\Delta f(s_i^+) = f_{\sigma_{i-1}}(x(s_i), \mu(s_i^+)) - f_{\sigma_i}(x(s_i), \mu(s_i^+))$ as $\lambda \downarrow 0$. Next, we have

$$\lim_{\lambda \downarrow 0} \frac{o(\Delta x_0)}{\lambda} = \lim_{\lambda \downarrow 0} \frac{o(\Delta x_0)}{\|\Delta x_0\|} \lim_{\lambda \downarrow 0} \frac{\|\Delta x_0\|}{\lambda}.$$

The first limit in the right hand side equals 0 since $\lim_{\lambda \downarrow 0} \Delta x_0 = 0$ and due to (2.16), while the second equals $\Delta f(s_i^+)$. Hence, the above expression is 0, and from (2.17) we have $\nabla_{e_i}\psi(t, s) = \phi(t, s_i)\Delta f(s_i^+)$. Let $y(t) = \phi(t, s_i)\Delta f(s_i^+)$. We can verify that

$$\begin{aligned} \dot{y}(t) &= \frac{\partial f_{\sigma_i}}{\partial x}(x(t), \mu(t))y(t), \quad t \in [s_i, s_{i+1}) \\ y(s_i) &= \Delta f(s_i^+). \end{aligned} \quad (2.18)$$

Also, $\nabla_{e_i}\psi(t_f, s) = \phi(t_f, s_i)\Delta f(s_i^+) = y(t_f)$. Hence,

$$\nabla_{e_i} g(x(t_f)) = \frac{\partial g(x(t_f))}{\partial x} y(t_f). \tag{2.19}$$

Let $p \in \mathbb{R}^n$ be the adjoint of the linear system (2.18), whose dynamics would satisfy

$$\begin{aligned} \dot{p}(t) &= -\frac{\partial f_{\sigma_i}}{\partial x}^T (x(t), \mu(t))p(t), \quad t \in [s_i, s_{i+1}) \\ p(t_f) &= \frac{\partial g^T(x(t_f))}{\partial x}. \end{aligned}$$

We can verify that for all $t \in [0, t_f]$, $D_t(p^T(t)y(t)) = 0$. Hence, $p^T(t_f)y(t_f) = p^T(s_i)y(s_i)$ and we use this result in (2.19) and the boundary values of y and p to get

$$\nabla_{e_i} g(x(t_f)) = p^T(s_i)\Delta f(s_i^+). \tag{2.20}$$

To account for the running cost we introduce a new state $x_{n+1} \in \mathbb{R}$ whose dynamics are given by

$$\begin{aligned} \dot{x}_{n+1}(t) =& l_{\sigma_i}(x(t), \mu(t)), \quad t \in [s_i, s_{i+1}) \\ x_{n+1}(0) =& 0. \end{aligned}$$

Let $\bar{x} = (x_1, \ldots, x_n, x_{n+1}) \in \mathbb{R}^{n+1}$ be an extended state. The cost $J(s, u)$ in (2.4) is equivalent to a new final cost defined as $\bar{g}(\bar{x}(t_f)) = g(x(t_f)) + x_{n+1}(t_f)$. Let the extended costate be $\bar{p} = (p_1, \ldots, p_n, p_{n+1}) \in \mathbb{R}^{n+1}$. It is easy to verify that $(p_1, p_2, \ldots, p_n)$ now satisfies the dynamics given in (2.14), while $\dot{p}_{n+1}(t) = 0$ for $t \in [0, t_f]$. In addition, $p_{n+1}(t_f) = D_{x_{n+1}}\bar{g}(\bar{x}(t_f)) = 1$, and hence $p_{n+1}(s_i) = 1$. Using this and applying (2.20) to the extended state we have: $\nabla_{e_i} J(s, \mu) = \nabla_{e_i}\bar{g}(\bar{x}(t_f)) = p^T(s_i)\Delta f(s_i^+) + \Delta l(s_i^+)$ as desired. □

The partial derivative of $J(s, \mu)$ with respect to s_i, $\frac{\partial J}{\partial s_i}$, exists if and only if $\nabla_{-e_i} J(s, \mu) = -\nabla_{e_i} J(s, \mu)$. It is easy to verify that $\nabla_{-e_i} J(s, \mu)$ is given as $\nabla_{-e_i} J(s, \mu) = -p^T(s_i)\Delta f(s_i^-) - \Delta l(s_i^-)$, for $s_{i-1} < s_i$. Based on the above, we can show that if

$$p^T(s_i)\Delta f(s_i^+) + \Delta l(s_i^+) = p^T(s_i)\Delta f(s_i^-) + \Delta l(s_i^-), \tag{2.21}$$

then the partial derivative of the cost function exists and is given as $\frac{\partial J}{\partial s}(s, \mu) = (\frac{\partial J}{\partial s_1}, \ldots, \frac{\partial J}{\partial s_N})$. A sufficient condition for the equality above and thus for the existence of this derivative is that the subsystems are autonomous or that the input μ is continuous at $t = s_i$ for $i = 1, 2, \ldots, N$. In addition, we have the following first-order optimality condition when the derivative is well-defined:

Corollary 2.1. If $s = (s_1, s_2, \ldots, s_N)$ is locally optimal, then $p^T(s_i)\Delta f(s_i) + \Delta l(s_i) = 0$.

We next provide two examples to illustrate the use of the variational analysis.

Example 2.1 (Car with two gears)**.** Optimal switching of automobile gears is a classic control problem and has been addressed in numerous previous works. The particular formulation of this example is motivated by [82]. Consider a car with two gears whose dynamics is modeled as

$$\begin{aligned} \dot{x}_1(t) &= x_2(t) \\ \dot{x}_2(t) &= h_q(x_2(t))\mu(t) \end{aligned} \tag{2.22}$$

In the above x_1 and x_2 denote the position and velocity of the car respectively, $h_q : \mathbb{R} \to \mathbb{R}_+$ represents the effect of being in gear q, $q \in \{1, 2\}$, with h_1 being a decreasing function and h_2 being an increasing function. The throttle input is $\mu : t \mapsto [-1, 1]$. The problem is to choose the input μ and the switching strategy so that the car moves from point $(x_0, 0)$ to $(x_f, 0)$, while minimizing the cost $J(s, \mu) = \int_0^{t_f} l(x_2(\tau))\, d\tau$, where $l : \mathbb{R} \to \mathbb{R}$ penalizes the speed. The final time t_f is free.

The mode dependent Hamiltonian as per Equation (2.7) in this case is given as

$$H(x, p, q, \mu) = l(x_2) + p_1 x_2 + p_2 h_q(x_2)\mu$$

From the Maximum Principle and the convexity of the Hamiltonian in input μ, we find that the optimal input is bang-bang: $\mu(t) = -\text{sgn}(p_2(t))$, for any mode of operation. The derivative of the cost function with respect to a switch time from mode 1 to mode 2 is then given by $-|p_2|\big(h_1(x_2) - h_2(x_2)\big)$ from the variational analysis of this section. Thus, the necessary optimality condition for a switch between the gears, based on Corollary 2.1, is $h_1(x_2(t)) = h_2(x_2(t))$ or $p_2(t) = 0$. The first case corresponds to the speed at which the gears reach the same efficiency and hence is the correct solution for switching the gears. Note that this result is independent of the cost function, initial and final desired point. In addition, the result is consistent with that given in [82] based on a numerical simulation of an approximate dynamic programming algorithm developed for hybrid systems.

Example 2.2 (Switched Linear Quadratic Regulator)**.** Consider a switched Linear Quadratic Regulator (LQR) in which the dynamics for mode $q \in \{1, 2\}$ is $\dot{x}(t) = A_q x(t) + B_q \mu(t)$, $A_q \in \mathbb{R}^{n\times n}$ and $B_q \in \mathbb{R}^{n\times m}$. We assume a single switch at $s \in [0, t_f]$. The cost function is

$$\begin{aligned} J(s, \mu) = &\int_0^s \big(x^T(\tau)Q_1 x(\tau) + \mu^T(\tau)R\mu(\tau)\big)\, d\tau + \\ &\int_s^{t_f} \big(x^T(\tau)Q_2 x(\tau) + \mu^T(\tau)R\mu(\tau)\big)\, d\tau + x^T(t_f)Qx(t_f). \end{aligned}$$

In the above, $Q_1, Q_2, Q \in \mathbb{R}^{n\times n}$ are positive semidefinite and $R \in \mathbb{R}^{m\times m}$ is positive definite. From the solution to the LQR problem with time-varying dynamics [93], it is known that for a given switch time s, the optimal input μ_s is a linear state-feedback law

$$\mu_s(t) = -R^{-1}B^T(t)K(t)x(t), \tag{2.23}$$

where $K(t) \in \mathbb{R}^{n\times n}$ is positive semi-definite and is governed by a differential Riccati equation

$$-\dot{K}(t) = K(t)A(t) + A^T(t)K(t) - K(t)B^T(t)R^{-1}B(t)K(t) + Q(t),$$

with boundary condition $K(t_f) = Q$, and $A(t) = A_2$, $Q(t) = Q_2$ for $t \in [s, t_f]$, $A(t) = A_1$, $Q(t) = Q_1$ for $t \in [0, s)$. It follows from (2.23) that for any problem in which $B_1 = B_2$ the input μ_s is continuous. Thus, the derivative of the cost function with respect to the switch time is well-defined. Also the costate satisfies $p(t) = K(t)x(t)$ [93]. Using Corollary 2.1, we find the first-order necessary optimality condition for the switch time $s \in [0, t_f]$ as

$$x^T(s)\big(K(s)(A_1 - A_2) + Q_1 - Q_2\big)x(s) = 0.$$

Note that although we considered switched LQR with two modes and one switch time, the results can easily be extended to multiple modes and switch times.

In the next section, we use similar variational analysis methods to define the variation of the cost and constraint functions with respect to a mode insertion.

2.4 Stage 2 - Variable Mode Sequence

The approach in optimization with respect to the mode sequence is based on considering a specific variation of the mode sequence for which we can derive analytic expressions for the resulting variation in the cost and the constraint functions. Using this variational analysis, we can derive necessary conditions for optimality of a given mode sequence and design an algorithm based on these optimality conditions. First, we conceptually describe the optimality conditions without discussing the mathematical details, and next we go into details on deriving the analytic expressions for implementing the algorithm.

Optimality Condition

Consider the hybrid optimal control problem described in Section 2.2. We compactly denote the control input, which is the optimization variable consisting of the mode sequence, switch times and model input, by a tuple $\xi := (\sigma, s, \mu)$. The cost as a function of this tuple is $J(\xi)$. We also define a function $\zeta : \xi \mapsto \mathbb{R}$ to compactly denote the constraints as follows:

$$\zeta(\xi) = \max_{j\in\{1,\dots,N_c\}} \max_{t\in[0,t_f]} h_j(x(t)). \tag{2.24}$$

The constraints $h_j(x(t)) \le 0$, $\forall t \in [0, t_f]$, $j = 1, 2, \dots, N_c$ can be represented by $\zeta(\xi) \le 0$.

Define insertion of mode $\hat{q} \in Q$ at a time $\hat{t} \in (t_0, t_f)$ to be a modification to ξ such that the subsystem $f_{\hat{q}}(x, \hat{u})$, $\hat{u} \in U$, would be active in the interval $(\hat{t} - \frac{\lambda}{2}, \hat{t} + \frac{\lambda}{2}) \subset [0, t_f]$ for

λ sufficiently small. If $\hat{t} = 0$, we insert the mode $\hat{q}$ in the interval $[0, \lambda]$ and if $\hat{t} = t_f$ we insert this mode in the interval $[t_f - \lambda, t_f]$. The mode insertion can be characterized by three parameters $(\hat{q}, \hat{t}, \hat{u}) \in Q \times [0, t_f] \times U$ and compactly denoted by $\eta := (\hat{q}, \hat{t}, \hat{u})$. The resulting modified input to the system is denoted as $\hat{\xi} := (\hat{\sigma}, \hat{s}, \hat{\mu})$. We define $\rho^{(\xi,\eta)}$ to be a function that describes this insertion, that is, $\rho^{(\xi,\eta)} : \lambda \mapsto \hat{\xi}$. In the discussion below, a fixed control input ξ and insertion η is considered. Hence, for simplicity in notation, we drop the dependence of ρ on ξ and η and simply write $\rho(\lambda)$.

If an input ξ is optimal then there would not be any mode insertion η into ξ such that the cost would be reduced while the constraints remain feasible. Thus, let us study effects of the mode insertion on the cost and the constraint functions. The variation of the cost function with respect to the mode insertion η can be analytically quantified by considering the directional derivative

$$\nabla_\eta J(\xi) := \lim_{\lambda \downarrow 0} \frac{J(\rho(\lambda)) - J(\xi)}{\lambda}. \tag{2.25}$$

Since modes of zero duration do not change the trajectory, we have $J(\rho(0)) = J(\xi)$. As such, for small $\lambda > 0$, we have the following first-order approximation:

$$J(\rho(\lambda)) \approx J(\xi) + \nabla_\eta J(\xi)\lambda. \tag{2.26}$$

Intuitively, if $\nabla_\eta J(\xi) < 0$ it would be possible to decrease the cost function through the mode insertion. Additionally, we need to ensure that after the mode insertion the constraints will not be violated. For this, we need to consider the directional derivative of the constraint function with respect to the mode insertion defined as

$$\nabla_\eta \zeta(\xi) := \lim_{\lambda \downarrow 0} \frac{\zeta(\rho(\lambda)) - \zeta(\xi)}{\lambda}. \tag{2.27}$$

Consider a strictly feasible point, that is, $\zeta(\xi) < 0$. Due to continuous dependence of $\zeta(\rho(\lambda))$ on $\lambda > 0$ (follows from the fact that (2.27) is well-defined), for sufficiently small $\lambda > 0$, we have $\zeta(\rho(\lambda) \leq 0$ and hence the constraints remain feasible. On the other hand, if $\zeta(\xi) = 0$ then a similar first-order approximation as in Equation (2.26) holds for the constraint function and hence a sufficient condition for constraint feasibility after the mode insertion is $\nabla_\eta \zeta(\xi) < 0$.

Based on the above variational analysis, we can define a set of necessary conditions for optimality of a control input $\xi = (\sigma, s, u)$.

Proposition 2.2. *If ξ is feasible and optimal then the following conditions hold:*

$$\textit{If } \zeta(\xi) < 0 \textit{ then } \forall \eta,\ \nabla_\eta J(\xi) \geq 0. \tag{2.28}$$

$$\textit{If } \zeta(\xi) = 0 \textit{ then } \forall \eta,\ \nabla_\eta J(\xi) \geq 0 \quad \textit{or } \nabla_\eta \zeta(\xi) \geq 0. \tag{2.29}$$

Proof. By contradiction. Let ξ be feasible and optimal. Suppose that the first condition does not hold. That is, $\zeta(\xi) < 0$ and there exists an insertion η such that $\nabla_\eta J(\xi) < 0$. Then, by definition of $\nabla_\eta J(\xi)$, there exists $\bar{\lambda} > 0$ such that for all $0 < \lambda \leq \bar{\lambda}$, $J(\rho(\lambda)) - J(\xi) \leq \frac{\bar{\lambda}}{2}\nabla_\eta J(\xi) < 0$, while $\zeta(\rho(\lambda)) - \zeta(\xi) \leq \frac{1}{2}\zeta(\xi) < 0$. Hence, for all $\lambda \leq \bar{\lambda}$, the new mode sequence $\hat{\xi} = \rho(\lambda)$ has a lower cost while the constraints remain feasible. This contradicts that ξ is an optimal input. Now, suppose the second condition does not hold. That is, $\zeta(\xi) = 0$ and there exists an insertion η such that $\nabla_\eta J(\xi) < 0$ and $\nabla_\eta \zeta(\xi) < 0$. Similarly, we can find $\bar{\lambda}$ such that for all $0 < \lambda < \bar{\lambda}$, $\hat{\xi} = \rho(\lambda)$ has a lower cost while $\zeta(\hat{\xi}) - \zeta(\xi) \leq \frac{\bar{\lambda}}{2}\nabla_\eta \zeta(\xi) < 0$. Thus, ξ cannot be feasible and optimal. □

Once we have an optimality condition for the hybrid optimal control problem, we have to determine a computationally feasible method for verification of the optimality condition. Let $\gamma > 0$ and consider the function $\theta : \xi \mapsto \mathbb{R}$ defined as a solution of the following optimization problem:

$$\theta(\xi) = \min_{\eta} \max \left\{ \nabla_\eta J(\xi), \zeta(\xi) + \gamma \nabla_\eta \zeta(\xi) \right\}. \tag{2.30}$$

First we can show that θ is a non-positive function as follows: Given a feasible $\xi = (\sigma, s, u)$, if for any time $\hat{t}$ we insert the mode in the sequence σ that is active at $\hat{t}$ and keep the input $\mu(t)$ during the insertion, then ξ remains unchanged and hence $\nabla_\eta J(\xi) = 0$ and $\nabla_\eta \zeta(\xi) = 0$ for this insertion. Consequently, there exists an η such that the function θ_0 defined as

$$\theta_0(\xi, \eta) := \max \left\{ \nabla_\eta J(\xi), \zeta(\xi) + \gamma \nabla_\eta \zeta(\xi) \right\} \tag{2.31}$$

is zero. It follows that $\theta(\xi) = \min_\eta \theta_0(\xi, \eta)$ is a non-positive function. Another important property of θ is that its zeros include inputs ξ which are feasible and optimal.

Proposition 2.3. *If ξ is feasible and optimal then $\theta(\xi) = 0$.*

Proof. By contradiction. Suppose ξ is feasible and optimal but $\theta(\xi)$ is not zero. Since θ is non-positive, then $\theta(\xi) < 0$. Thus, there exists η such that $\nabla_\eta J(\xi) < 0$ and $\zeta(\xi) + \gamma \nabla_\eta \zeta(\xi) < 0$. Now, if $\zeta(\xi) < 0$, then by first-order analysis we can insert η for sufficiently small λ and reduce the cost while remaining feasible. On the other hand, if $\zeta(\xi) = 0$, then we can ensure feasibility due to $\zeta(\xi) + \gamma \nabla_\eta \zeta(\xi) < 0$ and reduce cost due to $\nabla_\eta J(\xi) < 0$ for sufficiently small λ. This contradicts optimality of ξ. □

Verifying optimality of ξ by formulating an optimization problem of (2.30) is constructive because if $\theta(\xi) < 0$ we can use the insertion η for which $\theta_0(\xi, \eta) < 0$, to reduce the cost function while maintaining feasibility.

Intuitively, similar to the finite-dimensional optimization terminology, the mode insertion η for which $\theta_0(\xi, \eta) < 0$ defines a descent direction for the hybrid optimal control problem in the discrete mode sequence space, and λ defines the step size. In addition, $\theta(\xi) = 0$ could be used as a stopping criteria for the two-stage hybrid optimal control solution approach.

Algorithm Implementation

In order to implement Stage 2 of the hybrid optimal control algorithm, we derive analytic expressions for the directional derivative terms (2.25) and (2.27).

First, we need to introduce some notation. Let $\hat{t} \in (0, t_f)$ be an insertion time so that we insert the mode $\hat{q}$ in the interval $[\hat{t} - \frac{\lambda}{2}, \hat{t} + \frac{\lambda}{2}]$, $\lambda > 0$. Let σ_- be the mode immediately before $\hat{t}$, that is σ_- is the mode active at $t_- = \lim_{\lambda \to 0} \hat{t} - \lambda$ and σ_+ be the mode immediately after $\hat{t}$, that is the mode at $t_+ = \lim_{\lambda \to 0} \hat{t} + \lambda$. In addition, let $\mu(\hat{t}^-)$ and $\mu(\hat{t}^+)$ denote the inputs immediately before and after the insertion time, respectively. If $\hat{t} = 0$, we insert the new mode $\hat{q}$ in the interval $[0, \lambda]$ and we define $\sigma_+ = \sigma_- = \sigma_0$. If $\hat{t} = t_f$ we insert the new mode $\hat{q}$ in the interval $[t_f - \lambda, t_f]$ and define $\sigma_- = \sigma_+ = \sigma_N$. We define $\Delta_\eta f$ and $\Delta_\eta l$ as

$$\Delta_\eta f = f_{\hat{q}}(x(\hat{t}), \hat{u}) - \frac{1}{2} f_{\sigma_-}(x(\hat{t}), \mu(\hat{t}^-)) - \frac{1}{2} f_{\sigma_+}(x(\hat{t}), \mu(\hat{t}^+)), \tag{2.32}$$

$$\Delta_\eta l = l_{\hat{q}}(x(\hat{t}), \hat{u}) - \frac{1}{2} l_{\sigma_-}(x(\hat{t}), \mu(\hat{t}^-)) - \frac{1}{2} l_{\sigma_+}(x(\hat{t}), \mu(\hat{t}^+)). \tag{2.33}$$

To make the dependence of state $x(t)$ on the input $\xi = (\sigma, s, u)$ clear, we denote $\psi(t, \xi)$ as the solution of the differential equation evaluated at time t for a given control input ξ. We then define the directional derivative of the state with respect to the mode insertion as:

$$\nabla_\eta \psi(t, \xi) := \lim_{\lambda \downarrow 0} \frac{\psi(t, \rho(\lambda)) - \psi(t, \xi)}{\lambda}. \tag{2.34}$$

Proposition 2.4. *Consider an insertion $\eta = (\hat{q}, \hat{t}, \hat{u})$ into a given control input $\xi = (\sigma, s, u)$.*

(a) The directional derivative of the state is given as

$$\begin{aligned} \nabla_\eta \psi(t, \xi) &= 0, && \text{if } t \le \hat{t} \\ \nabla_\eta \psi(t, \xi) &= \phi(t, \hat{t}) \Delta_\eta f, && \text{if } t > \hat{t} \end{aligned}$$

In the above, $\phi(t, \hat{t}) \in \mathbb{R}^{n \times n}$ satisfies the following matrix differential equation:

$$\begin{aligned} \phi(t, \hat{t}) &= \frac{\partial f_{\sigma_i}}{\partial x}(x(t), \mu(t)) \phi(t, \hat{t}), \quad t \in [s_i, s_{i+1}) \\ \phi(\hat{t}, \hat{t}) &= I_{n \times n}. \end{aligned} \tag{2.35}$$

(b) The directional derivative of the cost function is given as

$$\nabla_\eta J(\xi) = p^T(\hat{t}) \Delta_\eta f + \Delta_\eta l, \tag{2.36}$$

where $p(t) \in \mathbb{R}^n$ satisfies the following backward differential equation for $t \in [\hat{t}, t_f]$:

$$\begin{aligned} \dot{p}(t) &= -\frac{\partial f_{\sigma_i}}{\partial x}^T (x(t), \mu(t)) p(t) - \frac{\partial l_{\sigma_i}}{\partial x}^T (x(t), \mu(t)), \quad t \in [s_i, s_{i+1}) \\ p(t_f) &= \frac{\partial g}{\partial x}^T (x(t_f)). \end{aligned}$$

(c) Let $\mathcal{H} := \{(j,t) \mid h_j(x(t)) = \zeta(\xi)\}$. The directional derivative of the constraint function is given as

$$\nabla_\eta \zeta(\xi) = \max_{(j,t)\in\mathcal{H}} \frac{\partial h_j}{\partial x}(x(t))\nabla_\eta \psi(t,\xi). \tag{2.37}$$

Proof. (a) First, the mode insertion does not affect the state trajectory before the insertion time $\hat{t}$ and hence $\nabla_\eta \psi(t,\xi) = 0$. For the state variation after the insertion time, consider $\hat{t} \in (0, t_f)$. Define $\psi(t,\hat{\xi})$ as the solution to the differential equation with control variable $\hat{\xi} = \rho(\lambda)$. In addition, let $\tilde{\xi}$ be the control input for which we make the mode insertion during the interval $[\hat{t} - \frac{\lambda}{2}, \hat{t}]$ and define $\psi(t,\tilde{\xi})$ as the trajectory corresponding to this mode insertion. We can write

$$\psi(t,\hat{\xi}) - \psi(t,\xi) = \psi(t,\hat{\xi}) - \psi(t,\tilde{\xi}) + \psi(t,\tilde{\xi}) - \psi(t,\xi). \tag{2.38}$$

Similar to the method of proof in Proposition 2.1 of Section 2.3 it can be shown that

$$\begin{aligned}
\lim_{\lambda\downarrow 0} \frac{\psi(t,\tilde{\xi}) - \psi(t,\xi)}{\lambda} =& \frac{1}{2}\phi(t,\hat{t})\big(f_{\hat{q}}(x(\hat{t}),\hat{u}) - f_{\sigma_-}(x(\hat{t}),\mu(\hat{t}^-))\big),\\
\lim_{\lambda\downarrow 0} \frac{\psi(t,\hat{\xi}) - \psi(t,\tilde{\xi})}{\lambda} =& \frac{1}{2}\phi(t,\hat{t})\big(f_{\hat{q}}(x(\hat{t}),\hat{u}) - f_{\sigma_+}(x(\hat{t}),\mu(\hat{t}^+))\big),
\end{aligned}$$

where $\phi(t,\hat{t})$ is the state-transition matrix and satisfies (2.35). Hence, using Equations (2.32), (2.33) and (2.38) we conclude that

$$\nabla_\eta \psi(t,\xi) := \lim_{\lambda\downarrow 0} \frac{\psi(t,\hat{\xi}) - \psi(t,\xi)}{\lambda} = \phi(t,\hat{t})\Delta_\eta f.$$

The cases for which the insertion is at $\hat{t} = 0$ or $\hat{t} = t_f$ can also be shown similarly.

(b) The result on directional derivative of the cost function then follows from the application of the Chain Rule and defining the adjoint dynamics as was done in proof of Proposition 2.1.

(c) From Corollary 5.4.6 and proof of Theorem 5.4.7 in [88] we conclude that

$$\nabla_\eta \zeta(\xi) = \max_{(j,t)\in\mathcal{H}} \nabla_\eta h_j(x(t_j)).$$

Now, for a given constraint function $h_j(x(t))$, by the Chain Rule

$$\nabla_\eta h_j(x(t)) = \frac{\partial h_j}{\partial x}(x(t))\nabla_\eta \psi(t,\xi).$$

Hence, we have the desired result. □

Given the analytical expressions for the directional derivatives of the cost and the constraint functions, we can compute $\theta_0(\xi, \eta)$ for every η. Once a mode insertion η with $\theta_0(\xi, \eta) < 0$ is found, we need to determine the insertion duration λ such that the cost is reduced. In analogy to finite-dimensional optimization problems the insertion η is a descent direction but we need to determine the step size to move along the descent direction. For this, we define the Armijo step size rule [94] for our problem as follows: Choose $\alpha, \beta \in (0, 1)$ and let the step size $\lambda^{(\xi,\eta)}$ be defined as

$$\lambda^{(\xi,\eta)} = \max_{k \in \mathbb{N}} \left\{ \beta^k \mid \ \zeta(\rho(\beta^k)) \leq 0, J(\rho(\beta^k)) - J(\xi) \leq \alpha \beta^k \theta_0(\xi, \eta) \right\}. \tag{2.39}$$

Lemma 2.1. *If $\theta_0(\xi, \eta) < 0$, then $\lambda^{(\xi,\eta)}$ defined in Equation (2.39) is strictly positive.*

Proof. By definition of the directional derivative, we have that as $k \to \infty$,

$$\frac{1}{\beta^k} \left(J(\rho(\beta^k)) - J(\xi) \right) \to \nabla_\eta J(\xi).$$

Since $\nabla_\eta J(\xi) \leq \theta_0(\xi, \eta)$ and $\theta_0(\xi, \eta) < 0$, it follows that $\nabla_\eta J(\xi) < \alpha \theta_0(\xi, \eta)$. Hence, there exists k_1 such that for $k \geq k_1$ we have

$$\frac{1}{\beta^k} \left(J(\rho(\beta^k)) - J(\xi) \right) < \alpha \theta_0(\xi, \eta).$$

Also, if $\zeta(\xi) < 0$ then there exists $k_2 > 0$ such that for $k > k_2$, $\zeta(\rho(\beta^k)) < 0$ since $\zeta(\rho(\beta^k)) \to \zeta(\xi)$. On the other hand, $\zeta(\xi) = 0$ and $\theta_0(\xi, \eta) < 0$ imply that $\nabla_\eta \zeta(\xi) < 0$. Hence, there exists k_2 such that for $k \geq k_2$ we have $\zeta(\rho(\beta^k)) \leq 0$. Consequently, β^{k_0} with $k_0 := \max\{k_1, k_2\}$ is a lower bound for the maximum in Equation (2.39) and $\lambda^{(\xi,\eta)}$ is strictly positive. □

Numerical implementation

Algorithm 2.1 details the two-stage solution approach of Section 2.2. In the algorithm, $\bar{N}$ is a parameter denoting the desired maximum number of modes.

To solve Stage 1 (Seps 0.a and 2.c of the algorithm), the methods discussed in Section 2.3 may be used. To find the mode insertion (Step 2.a of the algorithm), for every candidate mode $\hat{q} \in Q$, let $\eta = (\hat{q}, t, u)$. Then, the following optimization problem needs to be solved:

$$\min_{t \in [0, t_f], u \in U} \max \left\{ \nabla_\eta J(\xi), \zeta(\xi) + \gamma \nabla_\eta \zeta(\xi) \right\}. \tag{2.40}$$

Problem (2.40) can be implemented by employing the epigraph transformation to obtain a standard constrained minimization problem. Since this optimization problem is not necessarily convex, the resulting solution for a given $\hat{q}$, denoted as $(\hat{t}, \hat{u})$, may only be locally

Algorithm 2.1 Optimization Algorithm for the Hybrid Optimal Control Problem

Step 0. (Initialization) Let $\xi^0 := (\sigma^0, s^0, \mu^0)$.

a. Let (s^1, μ^1) be solution to Stage 1 with initial condition ξ^0.

b. Set $\sigma^1 = \sigma^0$, define $\xi^1 = (\sigma^1, s^1, \mu^1)$ and set $j = 1$.

Step 1. If $\theta(\xi^j) = 0$ or $N = \bar{N}$ then stop.

Step 2. Define ξ^{j+1} as follows:

a. (Stage 2) $\hat{\eta} = \arg\min_\eta \theta(\xi^j, \eta)$.

b. Find insertion duration λ from Armijo rule (2.39), and set $\hat{\xi} = (\hat{\sigma}, \hat{s}, \hat{\mu}) = \rho^{(\xi^j, \hat{\eta})}(\lambda)$.

c. (Stage 1) Let (s^{j+1}, μ^{j+1}) be solution to (2.11) with initial condition $\hat{\xi}$.

d. Define $\sigma^{j+1} = \hat{\sigma}$, $\xi^{j+1} = (\sigma^{j+1}, s^{j+1}, \mu^{j+1})$.

Step 3. Replace j by $j + 1$ and go to step 1.

optimal. However, if the objective function $\theta_0(\xi, \eta)$ is negative for some $\hat{\eta} = (\hat{q}, \hat{t}, \hat{u})$, then $\hat{\eta}$ is a mode insertion which decreases the cost while maintaining constraint feasibility. Hence, it is not necessary to find a global minimum of (2.40), rather any $\hat{\eta}$ which would result in $\theta_0(\xi, \eta) < 0$ would be sufficient. Analogous to the finite-dimensional optimization problems, while a global minimum would give a steepest descent direction, any $\hat{\eta}$ for which $\theta_0(\xi, \hat{\eta}) < 0$ would give a valid descent direction. In addition, note that γ is a parameter chosen to be small to ensure that if $\zeta(\xi) < 0$ for some ξ, then as desired by condition (2.28) in Proposition 2.2, the algorithm would return $\hat{\eta}$ with $\nabla_{\hat{\eta}} J(\xi) < 0$ as a feasible mode insertion, regardless of the value of $\nabla_{\hat{\eta}} \zeta(\xi)$.

We have implemented Algorithm 2.1 on Matlab. In our implementation, Stage 1 is solved through first transforming the problem into a conventional optimal control problem and then into a nonlinear program, as described in Chapter 4 of [88] and reviewed in Section 2.3. Both Stage 1 and Stage 2 optimizations are solved using SNOPT[1], a sparse nonlinear programming solver, provided by the TOMLAB optimization package. The algorithm has been employed in trajectory planning for several dynamical systems including quadrotor helicopter and bevel-tip surgical needle [49].

2.5 Aircraft Trajectory Design

The flight dynamics of an aircraft intrinsically has the characteristics of a switched hybrid system due to the coupling of the discrete flight modes with the continuous aircraft dynamics

[1]SNOPT: an SQP algorithm for Large-Scale Constrained Optimization, www.sbsi-sol-optimize.com

[5, 6, 7]. Switches between flight modes can be autonomous or controlled. Autonomous switches take place when the continuous state hits prescribed regions of the state space. For example, when the aircraft reaches a prescribed altitude, a switch from climb mode to cruise mode of flight would occur, while controlled switches occur due to a control command.

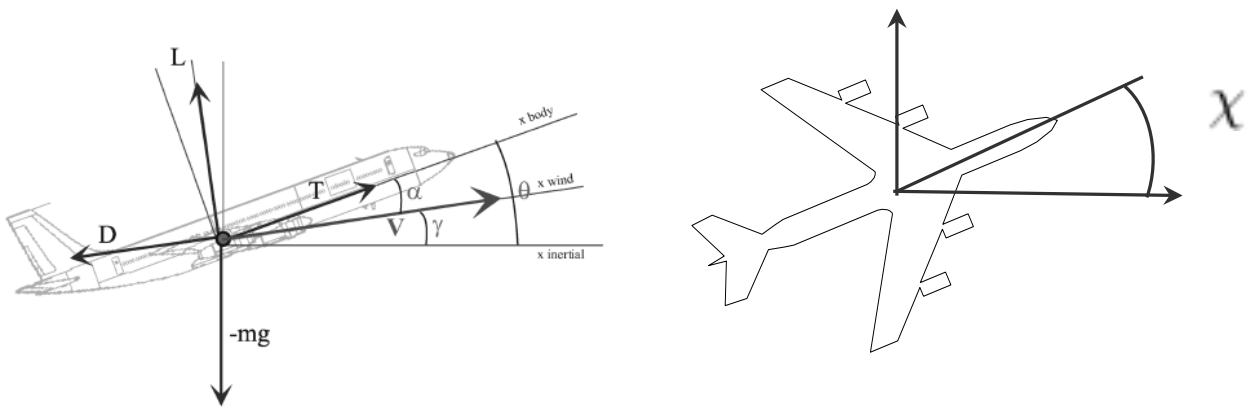
One of the NextGen concepts of operation, referred to as Trajectory Based Operations (TBO), envisions optimization of individual aircraft trajectories while ensuring safety and efficiency of air traffic. The TBO concept can be studied by formulating and solving the trajectory design using a hybrid optimal control framework.

In aerospace engineering, hybrid optimal control problems with a fixed mode sequence have been frequently formulated as multi-phase problems, in which, a phase refers to a mode of the hybrid dynamics [95]. The multi-phase problems are usually solved using pseudospectral methods and some have been applied to spacecraft missions [96]. However, none of the above research has focused on commercial aircraft. Recently, optimization with respect to fuel consumption of vertical profiles [97] and 3D profiles [98] of a commercial aircraft were formulated as hybrid optimal control problems. The solution method was based on assuming a fixed mode sequence and transforming the problem into a conventional optimal control problem as discussed in Section 2.3.

Although the sequence of modes in the current paradigm of flight is fixed *a priori* by the pilots or the air traffic controllers, variation of this sequence may improve the objective defined by the pilots or the airlines, which is for instance minimization of fuel consumption. In addition, given some unanticipated phenomenon such as storms, there may be a need to update the original sequence of flight modes in order to tackle the uncertainties in a safe and optimal way. Motivated by the possible gains of varying the flight mode sequence, this study applies the hybrid optimal control algorithm described in the previous sections to address commercial aircraft trajectory optimization. In our framework, we consider the Airbus 320 dynamic model, include effects of wind in the aircraft dynamics and model locations of storm obtained from forecast as constraints in airspace.

Hybrid Dynamics

In order to design fuel optimal aircraft trajectories, it is common to consider a 3 Degree Of Freedom (DOF) dynamic model that describes the point variable-mass motion of the aircraft over a flat earth model. To support a 3 DOF model, the translational equations are uncoupled from the rotational equations by assuming that the airplane rotational rates are small and that control surface deflections do not affect forces. Let the states of aircraft be denoted by V, χ, γ, referring to the true airspeed, heading angle, and flight path angle respectively, as shown in Figure 2.1; x, y, h, referring to the aircraft 3D position; and m, referring to the aircraft mass. We further assume symmetric flight so that there is no sideslip, and disregard earth rotation and curvature to obtain the equations of motion as follows:



(a) Vertical view (b) Horizontal view

Figure 2.1: Illustration of aircraft states

Aircraft equations of motion

$$
\begin{aligned}
m\dot{V} &= T - D - mg\sin\gamma \\
mV(\dot{\chi}\cos\gamma\cos\mu - \dot{\gamma}) &= mg\sin\mu\cos\gamma \\
mV(\dot{\chi}\cos\gamma\sin\mu + \dot{\gamma}) &= L - mg\cos\mu\cos\gamma \\
\dot{x} &= V\cos\gamma\cos\chi + V_{windx_h} \\
\dot{y} &= V\cos\gamma\sin\chi + V_{windy_h} \\
\dot{h} &= V\sin\gamma + V_{windz_h} \\
\dot{m} &= -T\eta
\end{aligned}
\tag{2.41}
$$

In the above the three dynamic equations are expressed in an aircraft-attached reference frame, while the three kinematic equations are expressed in a ground based reference frame. Wind is included due to its considerable effects on fuel consumption. V_{windx_h}, V_{windy_h}, V_{windz_h} are components of the wind, T is the thrust, and μ is the bank angle. Lift $L = C_L S\hat{p}$ and drag $D = C_D S\hat{p}$ are the components of the aerodynamic force, where S is the reference wing surface area, $\hat{p} = \frac{1}{2}\rho V^2$ is the dynamic pressure and ρ is the air density. A parabolic drag polar $C_D = C_{D0} + KC_L^2$ and a standard atmosphere are assumed.

In our model, the bank angle μ, the engine thrust T, and the coefficient of lift C_L are the inputs. The coefficient of lift C_L is a known function of the angle of attack α and the Mach number. The path constraints are based on the aircraft's flight envelope and can be found in the BADA manual [99]. For further details on aircraft dynamics see, for instance [100].

Flight modes

A 3D flight plan can be subdivided into a sequence of modes pertaining to flight in a vertical or horizontal plane. In both cases, we consider symmetric flight, that is, we assume there is no sideslip and all forces lie in the plane of symmetry of the aircraft. Also, we neglect the vertical component of the wind V_{windz_h} due to its low influence.

Horizontal 2D flight

In horizontal flight, $\dot{h}$ and γ are set to zero. Consequently, the following algebraic constraint is present: $L = mg\cos\mu$. We consider two modes in the horizontal flight. In mode 1, *control speed*, it is assumed that the aircraft flies with constant heading but with variable speed. The engine thrust T is the input and the bank angle μ is set to zero. In mode 2, *control heading*, the speed is set to a constant value and the input is μ.

Climb/Descent flight

In this mode the bank angle μ is set to zero. Without loss of generality, we consider $\chi = 0$, $\dot{y} = 0$. The engine thrust T, and the coefficient of lift C_L are the inputs of the aircraft. We refer to this mode as mode 3, the *control altitude* mode.

Trajectory Optimization

We consider en-route portion of the aircraft flight. In general, in this portion of the flight aircraft fly straight line segments connecting waypoints. In order to avoid hazardous weather, the aircraft may be required to deviate from their nominal paths. In terms of air traffic control, these deviations are characterized by maneuvers which may consist of heading, speed, or altitude changes. In our analysis, we consider flight maneuvers as modes of the switched system and consider maneuvers characterized by the three modes of *control speed*, *control heading*, and *control altitude* as introduced above. These types of maneuvers are routinely used in the air traffic control practice since they are easily communicated to the pilots and are easily implemented by auto-pilots [5].

We assume a region of airspace is unsafe to fly through due to weather storms. In the weather forecast data, storms may be characterized as regions with high values of Vertically Integrated Liquid (VIL) [101]. Although the VIL forecast are provided for a gridded airspace, a minimum-volume bounding ellipsoid can be used to capture these no-fly zones as obstacles, so that they can be used as constraints in an optimization algorithm [102].

To include wind in aircraft dynamics, we use the wind forecast data of July 6th, 2010, from the Rapid Update Cycle (RUC) by National Oceanic Atmospheric Administration (NOAA)[2]. A 4th degree polynomial, with the appropriate study of the residual and the regression coefficient statistics was fitted to the wind data.

Given a nominal path for the aircraft and an obstacle along the path which represents a storm, we formulate the problem of obstacle avoidance while minimizing fuel consumption as an optimal control problem for a constrained switched nonlinear system. In this set-up, a mode (or equivalently a maneuver) needs to be inserted in the original flight plan in order to avoid the obstacle while minimizing the cost function.

For the following two case studies, we solved the trajectory optimization problem using Algorithm 2.1. To solve Stage 1, a nonlinear program was formulated with a fixed number of sample points, $N_s = 40$, for each mode. The discretization scheme was chosen to be Euler for case study 1 and Simpson for case study 2. The equations of motion were enforced at each sample point for each mode. For example, for Euler discretization, the nonlinear equality constraint $x(k+1) - x(k) - \delta_i f_{q_i}(x(k), \mu(k)) = 0$ was enforced at the sampling points. The step size δ_i was scaled based on duration of mode i, that is, $\delta_i = \frac{s_{i+1} - s_i}{N_s}$. The resulting sparse nonlinear programming problem was solved using TOMLAB SNOPT optimization software. To solve Stage 2, the optimization problem expressed in (2.40) was converted to epigraph form and was solved using SNOPT optimization software. The stopping criteria in Step 1 of Algorithm 2.1 was set to $\theta(\xi^j) < 10^{-3}$.

Case study 1 - Obstacle avoidance in horizontal 2D flight

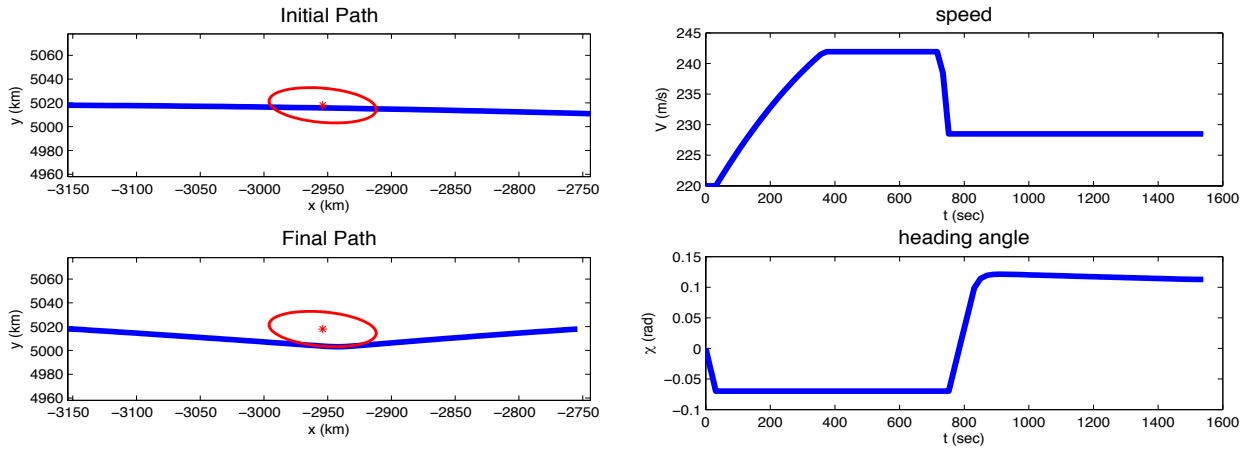
We assume the aircraft is cruising at a constant altitude of 11000 meters. The equations of motion are presented in (2.41) with the horizontal 2D flight hypothesis. There are two modes for the horizontal flight. In mode 1, *control speed* mode, the aircraft is flying with constant heading angle and hence the input μ is set to zero and the control input is the thrust T. For this mode the states with dynamics are V, x, y, and m. In mode 2, *control heading*, the speed is held constant by setting the thrust equal to the drag, $T = D$, and the control input is μ. The states with dynamics are χ, x, y, m. The aircraft needs to reach a target point $x_d \in \mathbb{R}^2$ while avoiding the hazardous weather obstacle.

Let x_{pos} be the 2D position of the aircraft. The objective is formulated as a final cost function which is a weighted sum of the distance from the target point, the cost of fuel consumption, and the final time to reach the target point. The cost function is given as:

$$J(\sigma, s, u) = K_d ||x_{pos}(t_f) - x_d||_2^2 - K_m m(t_f) + K_t t_f.$$

The weights were $K_d = 10$, $K_m = 0.5$, and $K_t = 0.1$. The obstacle was an ellipse centered at $(-3054 \text{ km}, 5018 \text{ km})$ with major and minor axis lengths of 42.2 and 14.4 km, respectively.

[2]http://www.noaa.gov/



(a) 2D aircraft Path

(b) States: speed V, and heading χ

Figure 2.2: Optimal 2D aircraft trajectory and states

	Iteration 1	Iteration 2
Mode sequence	(2)	(2,1,2)
Switch times	(1622)	(29.52, 379.68, 1504.2)
Cost	214.23	202.97

Table 2.1: Optimization results for 2D flight

The aircraft path was initialized as a straight line segment connecting the initial position of $(-3154 \text{ km}, 5018 \text{ km})$ to the final desired position of $(-2754 \text{ km}, 5018 \text{ km})$ and the mode sequence was initialized in mode 2. In the first iteration, Stage 1 of the algorithm returned an optimal path in which the obstacle was avoided by flying around it. Next, Stage 2 of the algorithm determined that an insertion of mode 1 at time 121 seconds would result in reduction of cost while ensuring feasibility of the path. The second iteration of Stage 1 of the algorithm, now initialized with mode sequence $(2, 1, 2)$ resulted in a reduced cost and a modified path. Figure 2.2 shows the aircraft path, its speed and heading angle after this iteration. The numerical results are summarized in Table 2.1.

This case study indicates that given a pre-defined aircraft path that is designed to avoid the obstacle using only a turn maneuver, the cost function can be reduced by including a straight flight maneuver, through the application of speed maneuver at an appropriate time, and by increasing the speed to an optimal value for an optimal duration of time.

Case study 2 - Obstacle avoidance in variable altitude flight

In this case study, it is assumed that the aircraft can be in three possible modes of *control speed*, *control heading*, *control altitude* as defined previously. In the first two modes where the altitude is held constant the horizontal 2D flight hypothesis hold. In *control altitude* mode, the inputs and equations of motion are modified based on the climb/descent flight hypothesis. In this case study, for simplicity in numerical optimization, wind is not taken into account.

Let $x_{pos} = (x, y, h)$ denote the aircraft position in 3D and $x_d \in \mathbb{R}^3$ denote the desired aircraft position. The cost function is defined similar to the previous case study:

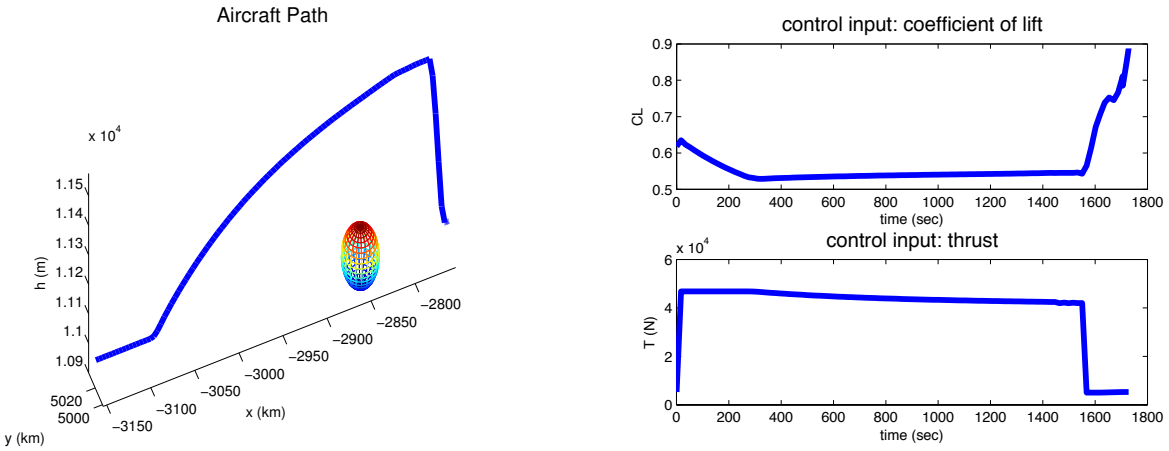
$$J(\sigma, s, u) = K_d||x_{pos}(t_f) - x_d||_2^2 - K_m m(t_f) + K_t t_f.$$

The weights in the cost function and the initial and final state of the aircraft were set to that of the previous case study. The weather obstacle was an ellipsoid in 3 dimensions, centered at $(-2854 \text{ km}, 5018 \text{ km}, 11 \text{ km})$, with an axis length of 20 km in the horizontal plane and 100 meters in the vertical plane.

Due to nonlinearities in the climb/descent flight dynamics, Euler integration did not provide good results. Consequently, a Simpson collocation method, as described in [90], was used to formulate the nonlinear program in Stage 1. The aircraft path and the mode sequence were initialized as the previous case study. In the first iteration of Stage 1, the algorithm resulted in an optimal solution in which the aircraft avoided the obstacle by flying around it in the horizontal plane, similar to the maneuver in the previous case study. Stage 2 of the algorithm found that an insertion of mode 3 at time index of 36 seconds would reduce the cost while maintaining feasibility. In the second iteration of Stage 1, initialized with mode sequence $(2, 3, 2)$, the aircraft gradually climbed to the maximum allowable altitude of 11500 meters to avoid the obstacle. At the very last portion of flight, it quickly descended to the desired final point. Figure 2.3 shows the aircraft path and the inputs. The inputs for mode 2 are not shown due to the small duration of this mode. The numerical results are summarized in Table 2.2.

The results here are consistent with the knowledge that there is less drag at higher altitudes due to reduced air density and hence it is optimal to avoid the obstacle by a climb maneuver. In addition, the gradual climb to the maximum altitude is in agreement with the concept of cruise climb in which the aircraft, ideally, would increase its altitude steadily as its weight is decreased. This concept, however, is not currently implemented due to Air Traffic Control safety requirements.

The running time for both case studies were below 2 minutes on a 2.56 GHz laptop with 4 GB RAM. Thus, both examples could be computed onboard.



(a) 3D aircraft path

(b) inputs: C_L and T

Figure 2.3: Optimal 3D aircraft trajectory

	Iteration 1	Iteration 2
Mode sequence	(2)	(3,2)
Switch times	(1828)	(1705, 1728)
Cost	241.61	227.09

Table 2.2: Optimization results for 3D flight

2.6 Conclusions

We described the problem of hybrid optimal control for a general class of hybrid systems referred to as switched nonlinear systems. For such systems, the control parameter has both a discrete component, the sequence of modes, and two continuous components, the duration of each mode and the input to each mode. We discussed the challenges in application of necessary and sufficient optimality conditions for solving such problems and reviewed the pertinent research on addressing these challenges. We developed a two-stage numerical algorithm to determine locally optimal control parameters for constrained nonlinear switched systems. The proposed algorithm divides the problem into two nonlinear constrained optimization problems; one associated with the continuous input components and the other associated with the discrete input component. During the continuous optimization, the mode sequence is fixed and the optimal mode duration and non-discrete input are constructed. During the discrete optimization, the mode sequence is varied by inserting a single mode. Through analytically characterizing the variation of the cost and the constraint functions due to this variation, we were able to analytically compare the optimality and feasibility of the mode

insertion, without needing to solve the optimal control problem associated with the new mode sequence.

We formulated the problem of fuel-efficient aircraft trajectory design subject to hazardous weather constraints as an optimal control problem for a constrained switched nonlinear system. Two applications on aircraft trajectory optimization were formulated in this framework and successfully solved. Based on the case studies, we propose several possible air traffic management applications for the hybrid optimal control formulation and the two-stage algorithm. At the strategic level, given a predefined sequence of modes that define the flight plan, the algorithm can be utilized to provide modifications to the mode sequence such that the gate to gate 4D trajectory is optimized. At the operational level, the modification of planned trajectories due to appearance of hazardous storms, potential collision, or the appropriate sequencing of aircraft at top of descent for starting a continuous descent approach, is currently addressed by an ad-hoc redefinition of the flight plans. Conceptually, the hybrid optimal control algorithm will be able to tackle such modifications through optimal maneuver insertions. However, to address the complexity in such realistic scenarios which include presence of multiple aircraft, it is necessary to improve the efficiency of the algorithm through further exploring different integration schemes, Non Linear Programming (NLP) solvers, and programing languages.

Chapter 3

Robust Control Synthesis for Stochastic Hybrid Systems

We develop a framework for analysis and control synthesis for safety and reachability of stochastic hybrid systems. In order to account for the presence of agents with uncertain influence on system dynamics, we formulate the reachability and safety objectives as a zero-sum stochastic game between two players, the control and the adversary. It will be shown that the maximum probability of the control reaching a target set while remaining inside the safe set, subject to the worst-case adversary behavior, can be computed through a suitable dynamic programming algorithm. The algorithm is applied to aircraft conflict detection in the presence of stochastic wind. The material in this chapter is based on our work in [53, 54].

3.1 Background

While mathematical models may in certain cases exactly describe the system under study, in most engineering and physical systems, such as air traffic or biological gene networks, models are abstractions of the behavior of the system, hopefully to an extent that makes analysis, prediction and control possible. Hence, it is natural to expect some discrepancy between the behavior of the system and that of the model representing it. In addition, even if the model truly represents the system, the interactions of the environment in which the system operates, such as wind effects on aircraft motion, may not be well-understood. Usually this discrepancy is acknowledged by including uncertainties in the model class or parameters. There are several methods for dealing with this uncertainty in the control design. Control under the so-called robust framework assumes the uncertain parameters belong to a bounded set around a nominal value and the objective should be satisfied under the worst-case performance of the uncertainty. As such, a control law that satisfies the

performance may be overly conservative or may not even exist. At a more fundamental level, no information about the uncertainty is used in the design process. For example, it may be known that the uncertain parameter often lies near a nominal value. As such, an alternative approach, based on assuming a probabilistic model of uncertainty, could be more appropriate. The control specification then may be given on the average rather than the worst-case performance of the system. The Stochastic Hybrid System (SHS) [32] framework is a powerful modeling technique that generalizes the hybrid dynamical models to include uncertainty in evolution of both the discrete and the continuous states.

For a controlled SHS, the performance of the closed-loop system can be evaluated in terms of the probability that the system trajectory obeys certain desired specifications. Of interest to safety-critical applications are probabilistic safety and reachability in which the control objective is to maximize the probability of remaining within a certain safe set or reaching a desired target set. When these two objectives are coupled the problem is referred to as *reach-avoid*. Early contributions in this domain for continuous-time SHS include [32, 103]. In [104] it is shown that the reach-avoid probability is the solution of an appropriate Hamilton-Jacobi-Bellman equation. To address the computational issues associated with probabilistic reachability analysis, the authors in [105, 106] propose a Markov chain approximation of the SHS, and apply the results to air traffic control studies. The safety probability for autonomous SHS has also been analyzed using a Lyapunov-like technique, referred to as the barrier certificates method [107] and a lower bound for probability of safety, albeit a very conservative one, is derived. For Discrete-Time Stochastic Hybrid Systems (DTSHS), a theoretical framework for the study of probabilistic safety problem is established in [46]. These results are generalized in [47] to address the reach-avoid problem, with considerations for time-varying and stochastic target sets and safe sets given in [56].

In this chapter, we extend the results on probabilistic safety and reachability of DTSHS [46, 47], to a zero-sum stochastic game setting. In particular, we consider a scenario where the evolution of the system state is affected not only by the actions of the control, as in previous work [47], but also by the actions of a rational adversary, whose objectives are opposed to that of the control. This consideration is motivated by practical applications such as conflict resolution between pairs of aircraft and control of networked systems subject to external attacks, in which the uncertainty in the decisions of the external agent may not obey any *a priori* known probability distribution, and the decisions may rather depend in a rational fashion on the current state of the system and possibly also on the actions of the control. Thus, we combine the previous modeling frameworks to account for two sources of uncertainty: one affects the system with a probabilistic model, and the other, in the absence of any additional knowledge, affects the system in a worst-case model. We call this modeling framework a Discrete-Time Stochastic Hybrid Game (DTSHG).

This Chapter is organized as follows: The DTSHG is described mathematically in Section 3.2. Then, we develop a stochastic game formulation of the reach-avoid problem in Section 3.3 and show that under certain standard continuity/compactness assumptions [108, 109]

on the underlying stochastic kernels and player action spaces, there exists: (a) a dynamic programming algorithm for determining the maximal probability of satisfying the reach-avoid objective, subject to the worst-case adversary behavior, called the maxmin reach-avoid probability; (b) a maxmin control policy which achieves the maxmin reach-avoid probability under the worst-case adversary strategy. Throughout, in order to explain the terminology developed and the solution methodology, we provide a tutorial example in which both the maxmin reach-avoid probability and the maxmin control policy can be calculated in an analytic fashion. We conclude with an application of pairwise aircraft conflict detection from air traffic management. We consider the possibility of lack of communication between the two aircraft, and include uncertainty in wind as a stochastic disturbance.

3.2 Discrete-Time Stochastic Hybrid Game Model

The model we propose for a Discrete-Time Stochastic Hybrid Dynamic Game (DTSHG) is an extension of the Discrete-Time Stochastic Hybrid System (DTSHS) model [46, 47] to a two-player stochastic game setting. Following standard conventions, we will refer to the control as player 1 and to the adversary as player 2. First, let us recall the definition of a Borel σ-algebra and Borel space.

Definition 3.1. Let X be a topological space. The Borel σ-algebra of X, denoted as $\mathcal{B}(X)$, is the smallest set of subsets of X which contains all the open sets of X and satisfies the three axioms of σ-algebra: it is (a) non-empty, (b) closed under complementation and (c) closed under countable union.

A space equipped with the Borel σ-algebra is called a Borel space and the memebers of the Borel σ-algebra are referred to as Borel subsets.

Definition 3.2. A Discrete-Time Stochastic Hybrid Game (DTSHG) between two players is a tuple $\mathcal{H} = (Q, n, A, D, \tau_v, \tau_q, \tau_r)$ as described below.

- *Discrete state space* $Q := \{1, 2, \ldots, M\}$, with $M \in \mathbb{N}$;
- *Dimension of continuous state space* $n : Q \to \mathbb{N}$: a map which assigns to each discrete state $q \in Q$ the dimension of the continuous state space $\mathbb{R}^{n(q)}$. The hybrid state space is given by $X := \bigcup_{q \in Q} \{q\} \times \mathbb{R}^{n(q)}$;
- *Player 1 control space* A: a nonempty, compact Borel space;
- *Player 2 control space* D: a nonempty, compact Borel space;
- *Continuous state stochastic kernel* $\tau_v : \mathcal{B}(\mathbb{R}^{n(\cdot)}) \times X \times A \times D \to [0, 1]$: a Borel-measurable stochastic kernel on $\mathbb{R}^{n(\cdot)}$ given $X \times A \times D$ which assigns to each $x = (q, v) \in X$, $a \in A$ and $d \in D$ a probability measure $\tau_v(\cdot|x, a, d)$ on the Borel space $(\mathbb{R}^{n(q)}, \mathcal{B}(\mathbb{R}^{n(q)}))$;

- *Discrete state stochastic kernel* $\tau_q : Q \times X \times A \times D \to [0, 1]$: a discrete stochastic kernel on Q given $X \times A \times D$ which assigns to each $x \in X$, $a \in A$ and $d \in D$ a probability distribution $\tau_q(\cdot|x, a, d)$ over Q;
- *Reset stochastic kernel* $\tau_r : \mathcal{B}(\mathbb{R}^{n(\cdot)}) \times X \times A \times D \times Q \to [0, 1]$: a Borel-measurable stochastic kernel on $\mathbb{R}^{n(\cdot)}$ given $X \times A \times D \times Q$ which assigns to each $x \in X$, $a \in A$, $d \in D$ and $q' \in Q$ a probability measure $\tau_r(\cdot|x, a, d, q')$ on the Borel space $(\mathbb{R}^{n(q')}, \mathcal{B}(\mathbb{R}^{n(q')}))$.

Few remarks about the model above are described below.

In this chapter and the subsequent chapter, the hybrid state is represented by $x = (q, v)$, in which q denotes the discrete state and v denotes the continuous state. This is in contrast with the deterministic formulation of Chapter 2, in which, due to absence of discrete state dynamics, the only dynamic state was the continuous state and was represented by x.

Note that the discrete stochastic kernel $\tau_q(.|x, a, d)$ can capture a very general class of hybrid systems as it allows for both state dependent (autonomous) and control dependent (controlled) switches. This includes piecewise deterministic Markov processes [35, 36] and discretized version of switching diffusions considered in [37].

The Borel subsets represent the events of our interest in X which we would like to assign a measure. For example, one can assign a probability measure to a Borel set $S \subset X$ in order to find the probability of the state being inside this set. The measurability requirements on the stochastic kernels are thus needed for the formal characterization of the probability that the state remains within or reaches desired subsets of the state space as will be shown in the next section. On the other hand, the input spaces are considered to be Borel in the model above so that randomized inputs can be defined on the space. Although in this thesis we do not consider randomized inputs, this is a topic that we would like to explore in future.

Within a non-cooperative dynamic game setting it is important to define the *information pattern*, namely the knowledge that each player has about the state of the system and the actions of the other player. With different information patterns, one may arrive at different formulations of the stochastic game, along with correspondingly different algorithms for computing the payoff functions for each player [110]. We consider here an information pattern which gives an advantage to player 2: at each time step, player 1 is allowed to select inputs based upon the current state of the system, while player 2 is allowed to select inputs based upon both the system state and the control input of player 1. This information pattern is common in robust control problems where the intentions of the adversarial agents are not known, and the control selects inputs in anticipation of the worst-case behavior of the adversary. For example in a network security application the network administrator (acting as player 1) implements certain security measures at each time step, while an external agent (acting as player 2) initiates network attacks after observing these security measures [111]. A mathematical description of this information pattern is given below.

Definition 3.3. A Markov policy for player 1 is a sequence $\mu = (\mu_0, \mu_1, \ldots, \mu_{N-1})$ of Borel measurable maps $\mu_k : X \to A$, $k = 0, 1, \ldots, N-1$. The set of all admissible Markov policies for player 1 is denoted by $\mathcal{M}_a$.

Definition 3.4. A Markov strategy for player 2 is a sequence $\gamma = (\gamma_0, \gamma_1, \ldots, \gamma_{N-1})$ of Borel measurable maps $\gamma_k : X \times A \to D$, $k = 0, 1, \ldots, N-1$. The set of all admissible Markov strategies for player 2 is denoted by Γ_d.

For a given initial condition $x_0 = (q_0, v_0) \in X$, player 1 policy $\mu \in \mathcal{M}_a$, and player 2 strategy $\gamma \in \Gamma_d$, the evolution of a DTSHG can be described as follows: At the beginning of each time step k, each player obtains a measurement of the current system state $x_k = (q_k, v_k) \in X$. Using this information, player 1 selects his/her controls as $a_k = \mu_k(x_k)$. Following this, player 2 selects his/her controls as $d_k = \gamma_k(x_k, a_k)$. The discrete state is then updated according to the discrete stochastic kernel as $q_{k+1} \sim \tau_q(\cdot|x_k, a_k, d_k)$. If the discrete state remains the same, namely $q_{k+1} = q_k$, then the continuous state is updated according to the continuous state stochastic kernel as $v_{k+1} \sim \tau_v(\cdot|x_k, a_k, d_k)$. On the other hand, if there is a discrete jump, the continuous state is instead updated according to the reset stochastic kernel as $v_{k+1} \sim \tau_r(\cdot|x_k, a_k, d_k, q_{k+1})$.

Following this description, we can use a similar approach as in [46] to compose the stochastic kernels τ_v, τ_q, and τ_r and form a hybrid state stochastic kernel $\tau : \mathcal{B}(X) \times X \times A \times D \to [0, 1]$ which describes the evolution of the hybrid state under the influence of player 1 and player 2 inputs and can be used to define the system execution compactly. Let $x = (q, v) \in X$, then

$$\tau\big((q', dv')|(q, v), a, d, q'\big) := \begin{cases} \tau_v(dv'|(q, v), a, d)\tau_q(q|(q, v), a, d), & \text{if } q' = q \\ \tau_r(dv'|(q, v), a, d, q')\tau_q(q'|(q, v), a, d), & \text{if } q' \neq q. \end{cases}$$

We can now define the execution of the DTSHG based on the hybrid stochastic kernel.

Definition 3.5. Let $\mathcal{H}$ be a DTSHG and $N \in \mathbb{N}$ be a finite time horizon. A stochastic process $\{x_k,\ k = 0, \ldots, N\}$ with values in X is an execution of $\mathcal{H}$ associated with a Markov policy $\mu \in \mathcal{M}_a$, a Markov strategy $\gamma \in \Gamma_d$, and an initial condition $x_0 \in X$ if its sample paths are obtained according to Algorithm 3.1.

As the player 1 policy μ and player 2 strategy γ are in general time-varying, the execution $\{x_k, k = 0, \ldots, N\}$ of the DTSHG is a time inhomogeneous stochastic process on the sample space $\Omega = X^{N+1}$, endowed with the canonical product topology $\mathcal{B}(\Omega) := \prod_{k=1}^{N+1} \mathcal{B}(X)$. In particular, the evolution of the closed-loop hybrid state trajectory can be described in terms of the stochastic kernels $\tau^{\mu_k, \gamma_k}(\cdot|x) := \tau\big(\,\cdot\,|x, \mu_k(x), \gamma_k(x, \mu_k(x))\big)$, $k = 0, 1, \ldots, N$. By Proposition 7.28 of [112], for a given $x_0 \in X$, $\mu \in \mathcal{M}_a$, $\gamma \in \Gamma_d$, these stochastic kernels

Algorithm 3.1 DTSHG Execution

Input Initial hybrid state $x_0 \in X$, Markov policy $\mu = (\mu_0, \mu_1, \ldots, \mu_{N-1}) \in \mathcal{M}_a$, Markov strategy $\gamma = (\gamma_0, \gamma_1, \ldots, \gamma_{N-1}) \in \Gamma_d$
Output Sample Path $\{x_k, k = 0, \ldots, N\}$
  Set $k = 0$;
  while $k < N$ **do**
    Set $a_k = \mu_k(x_k)$;
    Set $d_k = \gamma_k(x_k, a_k)$;
    Extract from X a value x_{k+1} according to $\tau(\cdot|x_k, a_k, d_k)$;
    Increment k;
  end while

induce a unique probability measure $P_{x_0}^{\mu,\gamma}$ on Ω as defined by

$$P_{x_0}^{\mu,\gamma}(X_0 \times X_1 \times \cdots \times X_N) = \int_{X_0} \int_{X_1} \cdots \int_{X_N} \tau^{\mu_{N-1},\gamma_{N-1}}(dx_N|x_{N-1}) \times \cdots \times \tau^{\mu_0,\gamma_0}(dx_1|x_0')\delta_{x_0}(dx_0'), \tag{3.1}$$

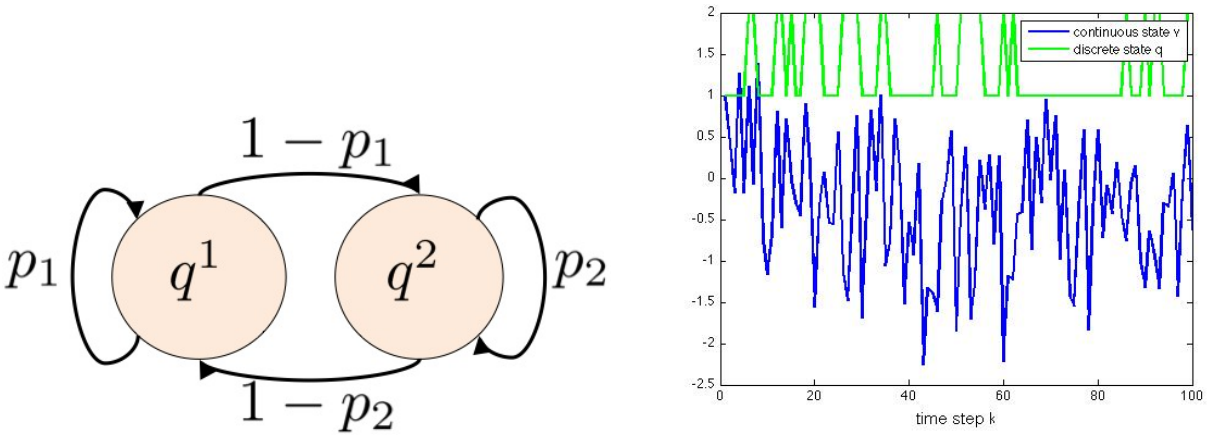
where $X_0, X_1 \ldots, X_N \in \mathcal{B}(X)$ are Borel subsets and δ_{x_0} denotes the probability measure on X which assigns mass one to the point $x_0 \in X$.

Example 3.1 (2-mode DTSHG)**.** In order to illustrate the definitions given so far, we provide a simple example. Consider a discrete-time stochastic hybrid system with two modes of operation $Q = \{q^1, q^2\}$, as shown in Figure 3.1(a). The transitions between the discrete modes are modeled probabilistically, with the probability of dwelling in mode q^i given by p^i, $i = 1, 2$. In mode q^i, the continuous state $v \in \mathbb{R}$ evolves according to a stochastic difference equation $v_{k+1} = f_q(v_k, a_k, d_k, \eta_k)$, defined as

$$\begin{aligned} v_{k+1} &= f_1(v_k, a_k, d_k, \eta_k) = 2v_k + a_k + d_k + \eta_k, \\ v_{k+1} &= f_2(v_k, a_k, d_k, \eta_k) = \frac{1}{2}v_k + a_k + d_k + \eta_k, \end{aligned} \tag{3.2}$$

where a_k and d_k are the inputs of player 1 and player 2, respectively, and η_k is a random variable modeling the effect of noise upon the system dynamics. It is assumed that the players have identical capabilities, with a_k and d_k taking values in $[-1, 1]$. The noise is modeled by a uniform distribution $\eta_k \sim U(-1, +1)$. A sample execution for initial condition $x_0 = (q_0, v_0) = (1, 1)$, $\mu_k = -\text{sgn}(v_k)$ and $\gamma_k = \frac{v_k a_k}{|2v_k a_k|}$ is shown in Figure 3.1(b).

Under the formal modeling framework defined previously, the hybrid state space is $X = \{q^1, q^2\} \times \mathbb{R}$, and the player input spaces are $A = D = [-1, 1]$. The discrete stochastic kernel τ_q is derived from the mode transition diagram of Figure 3.1(a) as $\tau_q(q^1|(q^1, v), a, d) = p_1$, $\tau_q(q^2|(q^1, v), a, d) = 1 - p_1$, $\tau_q(q^1|(q^2, v), a, d) = 1 - p_2$, $\tau_q(q^2|(q^2, v), a, d) = p_2$. The continuous stochastic kernel τ_v can be derived from the continuous state dynamics (3.2) as



(a) Discrete modes and transitions

(b) An execution of the DTSHG

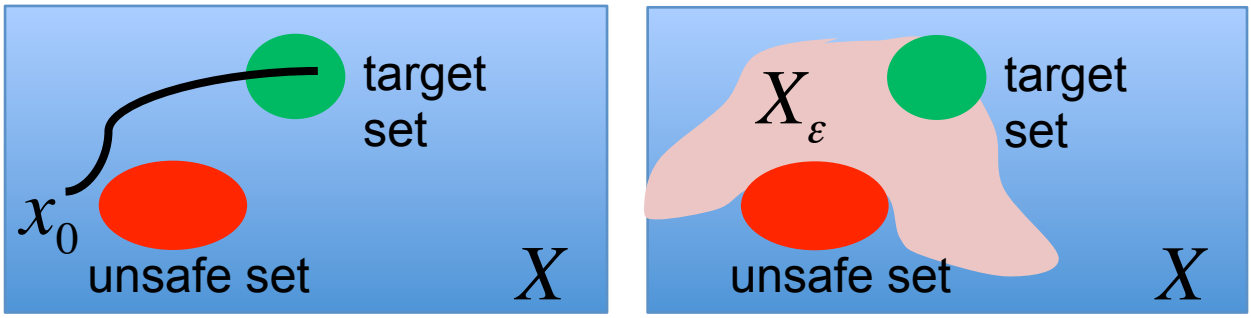
Figure 3.1: Discrete-time stochastic hybrid game with two modes

$\tau_v(dv'|(q^1,v),a,d) \sim U(2v+a+d-1, 2v+a+d+1)$, $\tau_v(dv'|(q^2,v),a,d) \sim U(\frac{1}{2}v+a+d-1, \frac{1}{2}v+a+d+1)$. With the assumption that the continuous state v is not reset during a discrete mode transition, the reset kernel is given by $\tau_r(dv'|(q,v),a,d,q') = \tau_v(dv'|(q,v),a,d)$. It is easy to see that the stochastic kernels are all Borel measurable.

3.3 Reach-Avoid Problem and Solution Approach

In the setting of the DTSHG, the reach-avoid problem becomes a stochastic game in which the objective of player 1 (the control) is to steer the hybrid system state into a desired target set while avoiding a set of unsafe states, as shown in Figure 3.2(a). On the other hand, the objective of player 2 (the adversary) is to either steer the state into the unsafe set or prevent it from reaching the target set.

Our reach-avoid problem formulation follows closely the formulation in [47]. Suppose that Borel sets $K, K' \in \mathcal{B}(X)$ are given as the desired target set and safe set, respectively, with $K \subseteq K'$. Then the probability that the state trajectory $(x_0, x_1, \ldots, x_N)$ reaches K while



(a) An execution of the stochastic trajectory (b) Probabilistic backward reach-avoid set

Figure 3.2: Reach-avoid problem for stochastic hybrid systems

staying inside K' under fixed choices of $\mu \in \mathcal{M}_a$ and $\gamma \in \Gamma_d$ is given by

$$\begin{aligned} r^{\mu,\gamma}_{x_0}(K, K') &:= P^{\mu,\gamma}_{x_0}\left(\bigcup_{j=0}^{N}(K' \setminus K)^j \times K \times X^{N-j}\right) \\ &= \sum_{j=0}^{N} P^{\mu,\gamma}_{x_0}\big((K' \setminus K)^j \times K \times X^{N-j}\big), \end{aligned} \tag{3.3}$$

where the second equality in (3.3) follows by the fact that the union is disjoint. By Equality (3.1) this probability can be computed as

$$r^{\mu,\gamma}_{x_0}(K, K') = E^{\mu,\gamma}_{x_0}\left[\mathbf{1}_K(x_0) + \sum_{j=1}^{N}\left(\prod_{i=0}^{j-1}\mathbf{1}_{K'\setminus K}(x_i)\right)\mathbf{1}_K(x_j)\right], \tag{3.4}$$

where $E^{\mu,\gamma}_{x_0}$ denotes the expectation with respect to the probability measure $P^{\mu,\gamma}_{x_0}$. Now define the worst-case reach-avoid probability under a choice of Markov policy μ as

$$r^{\mu}_{x_0}(K, K') = \inf_{\gamma\in\Gamma_d} r^{\mu,\gamma}_{x_0}(K, K'). \tag{3.5}$$

Our control objective is to maximize this worst-case probability over the set of Markov policies. The precise problem statement is as follows:

Problem 3.1. Given a DTSHG $\mathcal{H}$, target and safe sets $K, K' \in \mathcal{B}(X)$, $K \subseteq K'$, and $x_0 \in X$:

(a) Compute the *maxmin* reach-avoid probability $r^*_{x_0}(K, K') := \sup_{\mu\in\mathcal{M}_a} r^{\mu}_{x_0}(K, K')$;

(b) Find a *maxmin* policy $\mu^* \in \mathcal{M}_a$ such that $r^*_{x_0}(K, K') = r^{\mu^*}_{x_0}(K, K')$.

We show that the maxmin reach-avoid probability and the maximin control policy can be computed using an appropriate dynamic programming algorithm. For our theoretical derivations, we require the following assumptions on the stochastic kernels.

Assumption 3.1.

(a) For each $x = (q, v) \in X$ and $q' \in Q$, the function $\tau_q(q'|x, a, d)$ is continuous on $A \times D$;

(b) For each $x \in X$ and $E_1 \in \mathcal{B}(\mathbb{R}^{n(q)})$, the function $\tau_v(E_1|x, a, d)$ is continuous on $A \times D$;

(c) For each $x \in X$, $q' \in Q$, and $E_2 \in \mathcal{B}(\mathbb{R}^{n(q')})$, the function $\tau_r(E_2|x, a, d, q')$ is continuous on $A \times D$.

Note that we only assume continuity of the stochastic kernels in the actions of player 1 and player 2, but not necessarily in the system state. Thus, our Borel-measurable model still allows for stochastic hybrid systems where transition probabilities change abruptly with changes in the system state. Furthermore, if the action spaces A and D are finite or countable, then the above assumptions are clearly satisfied under the discrete topology on A and D. Also, if $\tau_v(\cdot|(q, v), a, d)$ has a density function $f_v(v'|(q, v), a, d), v' \in \mathbb{R}^{n(q)}$ for every $q \in Q$, and $f_v(v'|(q, v), a, d)$ is continuous in a and d, it can be checked that the assumption for τ_v is satisfied. A similar condition can also be stated for the reset kernel τ_r.

The compactness of the players' input spaces and the continuity assumptions on the stochastic kernels are sufficient to ensure that optimal policies and strategies exist at each step of the dynamic programming algorithm as will be shown. If optimal or ϵ-optimal policies do not exist, then the dynamic programming recursion may not hold in general. Please see Examples 1 and 3 in Chapter 8 of [112] for counter examples.

Let $\mathcal{F}$ denote the set of Borel measurable functions from X to $[0, 1]$. For the statement of the algorithm, define $H : X \times A \times D \times \mathcal{F} \to [0, 1]$ as

$$H(x, a, d, J) = \int_X J(y)\tau(dy|x, a, d). \tag{3.6}$$

Next, define the dynamic programming operator $T : \mathcal{F} \to \mathcal{F}$:

$$T[J](x) = \sup_{a \in A} \inf_{d \in D} \mathbf{1}_K(x) + \mathbf{1}_{K' \setminus K}(x) H(x, a, d, J) \quad x \in X. \tag{3.7}$$

Our main result is as follows:

Theorem 3.1. *Let $\mathcal{H}$ be a DTSHG satisfying Assumption 3.1. Let $K, K' \in \mathcal{B}(X)$ be Borel sets such that $K \subseteq K'$. Let the operator T be defined as in (3.7). Then the composition $T^N = T \circ T \circ \cdots \circ T$ (N times) is well-defined and*

(a) $r^*_{x_0}(K, K') = T^N[\mathbf{1}_K](x_0), \forall x_0 \in X$;

(b) There exists a player 1 policy $\mu^* \in \mathcal{M}_a$ and player 2 strategy $\gamma^* \in \Gamma_d$ satisfying

$$r_{x_0}^{\mu,\gamma^*}(K, K') \leq r_{x_0}^*(K, K') \leq r_{x_0}^{\mu^*,\gamma}(K, K'), \tag{3.8}$$

for all $x_0 \in X$, $\mu \in \mathcal{M}_a$, and $\gamma \in \Gamma_d$. In particular, μ^* is a maxmin policy for player 1.

(c) If $\mu^* = (\mu_0^*, \mu_1^*, \ldots, \mu_{N-1}^*) \in \mathcal{M}_a$ is a Markov policy which satisfies

$$\mu_k^*(x) \in \arg\sup_{a \in A} \inf_{d \in D} H(x, a, d, J_{k+1}),\ x \in K' \setminus K, \tag{3.9}$$

where $J_k = T^{N-k}[\mathbf{1}_K]$, $k = 0, 1, \ldots, N$, then μ^* is a maxmin policy for player 1. In addition, if $\gamma^* = (\gamma_0^*, \gamma_1^*, \ldots, \gamma_{N-1}^*) \in \Gamma_d$ is a Markov strategy which satisfies

$$\gamma_k^*(x, a) \in \arg\inf_{d \in D} H(x, a, d, J_{k+1}),\ x \in K' \setminus K,\ a \in A, \tag{3.10}$$

then γ^* is a worst-case strategy for player 2.

Although there is a large number of previous results in the field of non-cooperative stochastic games [113, 108, 109, 114, 115], the direct application of these results to our formulation is difficult, for several reasons. First, the pay-off functions for the safety and reach-avoid problems are sum-multiplicative, which prevents the use of results from the more common additive cost problems [108, 114]. In addition, although there is previous work on more general utility functions which depend on the entire history of the game [109, 115], the results are primarily for the existence of randomized policies under a symmetric information pattern. Due to practical implementation and robustness concerns, we are more interested in the existence of nonrandomized policies under a non-symmetric information pattern. Finally, an important feature of hybrid systems is that the dynamics in the continuous state space can change abruptly across switching boundaries. This requires a relaxation of the continuity assumptions in the continuous state space such as those given in [113].

The proof of this theorem proceeds through a sequence of lemmas and propositions which generalize the dynamic programming algorithms given in [47] and [46] for the single player case. First, it is shown that the operator T preserve measurability properties, and so the sequential composition of T is well-defined. Furthermore, using the continuity properties given in Assumption 3.1, it is shown that there exist Borel measurable functions which achieve the supremum and infimum in (3.7) at each step of the dynamic programming recursion. Next, it is shown that for fixed $\mu \in \mathcal{M}_a$ and $\gamma \in \Gamma_d$, the reach-avoid probability $r_{x_0}^{\mu,\gamma}(K, K')$, can be computed using a recursive formula. Finally, it is shown that the function $T^N[\mathbf{1}_K]$ simultaneously upper bounds and lower bounds $r_{x_0}^*(K, K')$ and hence is equal to $r_{x_0}^*(K, K')$. In the course of proving this last result, the existence of a maxmin policy for player 1 and a worst-case strategy for player 2 is also established.

Properties of the dynamic programming operator T

Here, we will prove some properties of T. First, we state a special case of Corollary 1 given in [116]. This result allows us to show that the operator T preserves Borel measurability and that it is sufficient to consider Borel measurable selectors.

Lemma 3.1. *Let X, Y be complete separable metric spaces such that Y is compact, and f be a real-valued Borel measurable function defined on $X \times Y$ such that $f(x, \cdot)$ is lower semicontinuous with respect to the topology on Y. Define $f^* : X \to \mathbb{R} \cup \{\pm\infty\}$ by*

$$f^*(x) = \inf_{y \in Y} f(x, y).$$

(a) The set $I := \{x \in X : \text{ for some } y \in Y, f(x, y) = f^*(x)\}$ is Borel measurable.

(b) For every $\epsilon > 0$, there exists a Borel measurable function $\phi : X \to Y$, satisfying

$$f(x, \phi(x)) = f^*(x), \text{ if } x \in I,$$
$$f(x, \phi(x)) \leq \begin{cases} f^*(x) + \epsilon, & \text{if } x \notin I,\ f^*(x) > -\infty, \\ -1/\epsilon, & \text{if } x \notin I,\ f^*(x) = -\infty. \end{cases}$$

In order to prove that the supremum and infimum in the expression for T is achieved, we will need the operator H to produce functions continuous in A and D. For this purpose, we state the following technical result from [108].

Lemma 3.2. *Let f be a bounded real-valued Borel measurable function on a Borel space Y, and τ be a Borel measurable transition probability from a Borel space X into Y such that $\tau(B|\cdot)$ is continuous on X for each $B \in \mathcal{B}(Y)$. Then the function $x \to \int f(y)\tau(dy|x)$ is continuous on X.*

We are now ready to prove that the operator T preserves Borel measurability, and that the infimum and supremum in (3.7) can be achieved by Borel measurable selectors. For notational convenience, we introduce an operator G which takes a real-valued Borel measurable function on X and produces a real-valued function on $X \times A$:

$$G[J](x, a) = \inf_{d \in D} \mathbf{1}_K(x) + \mathbf{1}_{K' \setminus K}(x) H(x, a, d, J). \tag{3.11}$$

Proposition 3.1.

(a) $\forall J \in \mathcal{F}$, $T[J] \in \mathcal{F}$.

(b) For any $J \in \mathcal{F}$, there exists a Borel measurable function $\lambda^* : X \times A \to D$ such that, $\forall (x, a) \in X \times A$,

$$\lambda^*(x, a) \in \arg\inf_{d \in D} \mathbf{1}_K(x) + \mathbf{1}_{K' \setminus K}(x) H(x, a, d, J). \tag{3.12}$$

(c) For any $J \in \mathcal{F}$, there exists a Borel measurable function $\pi^* : X \to A$, such that $\forall x \in X$,

$$\pi^*(x) \in \arg\sup_{a \in A}\{\inf_{d \in D} \mathbf{1}_K(x) + \mathbf{1}_{K' \setminus K}(x) H(x, a, d, J)\}. \tag{3.13}$$

Proof. For any $J \in \mathcal{F}$, define a function $f_J : X \times A \times D \to \mathbb{R}$ as

$$f_J(x, a, d) = H(x, a, d, J).$$

From the definition of H in Equation (3.6), it follows that the range of f_J lies in $[0, 1]$. By the Borel measurability of J and τ, Proposition 7.29 of [112] implies that f_J is Borel measurable. Furthermore, for each $x \in X$, Lemma 3.2 implies that $f_J(x, a, d)$ is continuous in a and d. Now consider $\tilde{f}_J : X \times A \times D \to \mathbb{R}$,

$$\tilde{f}_J(x, a, d) = \mathbf{1}_K(x) + \mathbf{1}_{K' \setminus K}(x) f_J(x, a, d).$$

Clearly, $0 \leq \tilde{f}_J \leq 1$. Furthermore, given that Borel measurability is preserved under summation and multiplication (see for example Proposition 2.6 of [117]), $\tilde{f}_J$ is also Borel measurable. Finally, it is clear that $\tilde{f}_J(x, a, d)$ is continuous in a and d for each $x \in X$. We observe that

$$G[J](x, a) = \inf_{d \in D} \tilde{f}_J(x, a, d). \tag{3.14}$$

Since the range of $\tilde{f}_J$ is $[0, 1]$, the range of $G[J]$ is also $[0, 1]$. By assumption, A and D are Borel spaces and hence metrizable. Thus, A can be endowed with a metric d_1 consistent with the topology on A, while D can be endowed with a metric d_2 consistent with the topology on D. Furthermore, as shown in [118], the hybrid state space X can be endowed with a metric equivalent to the standard Euclidean metric when restricted to each continuous domain $\mathbb{R}^{n(q)}$, $q \in Q$. Under the assumptions on the DTSHG model, the spaces X, A, and D are also complete and separable. Now for each $(x, a) \in X \times A$, we have by the previous derivations that $\tilde{f}_J(x, a, \cdot)$ is continuous on D. By the compactness of D, the infimum in Equation (3.14) is achieved for each fixed (x, a) (see for example Theorem 4.16 in [119]). Thus, applying Lemma 3.1, we have that there exists a Borel measurable function $\lambda^* : X \times A \to D$ for which (3.12) holds.

For the outer supremum, note that the composition of Borel measurable functions remains Borel measurable and thus, $G[J]$ is a Borel measurable function. Since the infimum in Equation (3.14) is achieved and $\tilde{f}_J$ is continuous on A, we conclude that $G[J]$ is also continuous on A. Observe that

$$T[J](x) = -\inf_{a \in A} -G[J](x, a), \ x \in X. \tag{3.15}$$

By the compactness of A, the infimum in (3.15) is achieved for each $x \in X$. Thus, a repeated application of Lemma 3.1 shows that there exists a Borel measurable function $\pi^* : X \to A$ such that $-T[J](x) = -G[J](x, \pi^*(x))$, $\forall x \in X$. Hence, $T[J]$ is composition of Borel measurable functions and is Borel measurable. Finally, clearly, range of $T[J]$ lies in $[0, 1]$, and so $T[J] \in \mathcal{F}$. □

Next, motivated by the expression for $r^{\mu,\gamma}_{x_0}(K, K')$ in (3.3), for fixed $\mu \in \mathcal{M}_a$ and $\gamma \in \Gamma_d$, we define the functions $V^{\mu,\gamma}_k : X \to [0,1]$, $k = 0, \ldots, N$

$$\begin{aligned} V^{\mu,\gamma}_N(x) =& \mathbf{1}_K(x), \\ V^{\mu,\gamma}_k(x) =& \mathbf{1}_K(x) + \mathbf{1}_{K'\setminus K}(x) \int_{X^{N-k}} \sum_{j=k+1}^{N} \prod_{i=k+1}^{j-1} \mathbf{1}_{K'\setminus K}(x_i)\mathbf{1}_K(x_j) \\ & \prod_{j=k+1}^{N-1} \tau^{\mu_j,\gamma_j}(dx_{j+1}|x_j)\tau^{\mu_k,\gamma_k}(dx_{k+1}|x). \end{aligned} \tag{3.16}$$

In the above, we use the convention that $\prod_{i=k}^{j}(.) = 1$ for $k > j$. From the definition above and expanding the expectation in (3.3), it is clear that $V^{\mu,\gamma}_0(x_0) = r^{\mu,\gamma}_{x_0}(K, K')$, $\forall x_0 \in X$.

The task of computation of reach-avoid probability for a given policy and strategy becomes equivalent to finding a method for computing $V^{\mu,\gamma}_k(x)$. Consider a recursion operator $T_{\pi,\lambda} : \mathcal{F} \to \mathcal{F}$, parameterized by Borel measurable functions $\pi : X \to A$ and $\lambda : X \times A \to D$:

$$T_{\pi,\lambda}[J](x) = \mathbf{1}_K(x) + \mathbf{1}_{K'\setminus K}(x)H(x, \pi(x), \lambda(x, \pi(x)), J), \ x \in X \tag{3.17}$$

where H is defined in (3.6). The following result shows that the functions $V^{\mu,\gamma}_k$ can be computed using backwards recursion under the operator $T_{\pi,\lambda}$.

Lemma 3.3. *Let $\mu \in \mathcal{M}_a$ and $\gamma \in \Gamma_d$. For $k = 0, 1, \ldots, N-1$, the following identity holds:*

$$V^{\mu,\gamma}_k = T_{\mu_k,\gamma_k}[V^{\mu,\gamma}_{k+1}]. \tag{3.18}$$

Proof. For $k = N-1$, the definition of $V^{\mu,\gamma}_N$ implies that for any $x \in X$,

$$\begin{aligned} V^{\mu,\gamma}_{N-1}(x) &= \mathbf{1}_K(x) + \mathbf{1}_{K'\setminus K}(x) \int_X \mathbf{1}_K(x_N)\tau^{\mu_{N-1},\gamma_{N-1}}(dx_N|x) \\ &= T_{\mu_{N-1},\gamma_{N-1}}[V^{\mu,\gamma}_N]. \end{aligned}$$

For $0 \leq k < N-1$, we have by the expression for $V^{\mu,\gamma}_k$ in (3.16) that for any $x \in X$,

$$\begin{aligned} V^{\mu,\gamma}_k(x) =& \mathbf{1}_K(x) + \mathbf{1}_{K'\setminus K}(x) \int_X \mathbf{1}_K(x_{k+1}) + \mathbf{1}_{K'\setminus K}(x_{k+1}) \\ & \left(\int_{X^{N-k-1}} \sum_{j=k+2}^{N} \prod_{i=k+2}^{j-1} \mathbf{1}_{K'\setminus K}(x_i)\mathbf{1}_K(x_j) \right) \prod_{j=k+1}^{N-1} \tau^{\mu_j,\gamma_j}(dx_{j+1}|x_j)\tau^{\mu_k,\gamma_k}(dx_{k+1}|x) \\ =& \mathbf{1}_K(x) + \mathbf{1}_{K'\setminus K}(x) \int_X V^{\mu,\gamma}_{k+1}(x_{k+1})\tau^{\mu_k,\gamma_k}(dx_{k+1}|x). \end{aligned}$$

It follows from the definition of $T_{\pi,\lambda}$ that the last expression above is $T_{\mu_k,\gamma_k}[V^{\mu,\gamma}_{k+1}]$ as desired. □

For the proofs of the next two propositions, we use the fact the operator $T_{\pi,\lambda}$ satisfies a monotone property: for any Borel measurable functions J, J' from X to $[0,1]$ such that $J(x) \leq J'(x), \forall x \in X$, $T_{\pi,\lambda}[J](x) \leq T_{\pi,\lambda}(J')(x), \forall x \in X$. It is straightforward to check this property using the definition of H in Equation (3.6) and the properties of integrals.

Proposition 3.2.

(a) $\forall x_0 \in X$, $T^N[\mathbf{1}_K](x_0) \leq r^*_{x_0}(K, K')$.

(b) There exists $\mu^* \in \mathcal{M}_a$ *such that, for any* $\gamma \in \Gamma_d$, $T^N[\mathbf{1}_K](x_0) \leq r^{\mu^*,\gamma}_{x_0}(K, K'), \forall x_0 \in X$.

Proof. For notational convenience, for $k = 0, 1, \ldots, N$ we define $J_{N-k} := T^k[\mathbf{1}_K]$.

First, we prove the following claim: There exists $\mu^*_{N-k\to N} := (\mu^*_{N-k}, \mu^*_{N-k+1}, \ldots, \mu^*_{N-1})$ such that, for any $\gamma_{N-k\to N} := (\gamma_{N-k}, \gamma_{N-k+1}, \ldots, \gamma_{N-1})$, $J_{N-k}(x) \leq V^{\mu^*_{N-k\to N},\gamma}_{N-k}(x)$, $\forall x \in X$.

Let $\gamma = (\gamma_0, \gamma_1, \ldots, \gamma_{N-1}) \in \Gamma$ be arbitrary. The case of $k = 0$ is trivial. For the inductive step, assume that this holds for $k = h$. By the induction hypothesis, there exists a policy $\mu^*_{N-h\to N} = (\mu^*_{N-h}, \mu^*_{N-h+1}, \ldots, \mu^*_{N-1}) \in \mathcal{M}_a$ such that, for any $\gamma \in \Gamma_d$, $J_{N-h}(x) \leq V^{\mu^*_{N-h\to N},\gamma}_{N-h}(x), \forall x \in X$. Furthermore, by Proposition 3.1(c), there exists a Borel measurable function $\pi^* : X \to A$ such that $G[J_{N-h}](x, \pi^*(x)) = T[J_{N-h}](x), \forall x \in X$. Choose a policy $\mu^*_{N-h-1\to N} = (\pi^*, \mu^*_{N-h}, \mu^*_{N-h+1}, \ldots, \mu^*_{N-1})$. Then by the monotonicity of the operator $T_{\pi,\lambda}$ and Lemma 3.3 we have:

$$\begin{aligned}
V^{\mu^*_{N-h-1\to N},\gamma}_{N-h-1}(x) &= T_{\pi^*,\gamma_{N-h-1}}[V^{\mu^*_{N-h\to N},\gamma}_{N-h}](x) \\
&\geq T_{\pi^*,\gamma_{N-h-1}}[J_{N-h}](x) \\
&= \mathbf{1}_K(x) + \mathbf{1}_{K'\setminus K}(x) H(x, \pi^*(x), \gamma_{N-h-1}(x, \pi^*(x)), J_{N-h}) \\
&\geq \inf_{d\in D} \mathbf{1}_K(x) + \mathbf{1}_{K'\setminus K}(x) H(x, \pi^*(x), d, J_{N-h}) \\
&= G[J_{N-h}](x, \pi^*(x)) \\
&= T[J_{N-h}](x) = J_{N-h-1}(x),
\end{aligned}$$

which holds for each $x \in X$ and thus concludes the proof of the claim.

This result implies that there exists $\mu^*_{0\to N} = (\mu^*_0, \mu^*_1, \ldots, \mu^*_{N-1}) \in \mathcal{M}_a$ such that, for any $\gamma = (\gamma_0, \gamma_1, \ldots, \gamma_{N-1}) \in \Gamma_d$, $T^N[\mathbf{1}_K](x_0) = J_0(x_0) \leq V^{\mu^*_{0\to N},\gamma}_0(x_0) = r^{\mu^*_{0\to N},\gamma}_{x_0}(K, K')$, $\forall x_0 \in X$. Hence, $\mu^*_{0\to N}$ is the Markov policy satisfying statement (b) of the propsoition. Also, since γ is arbitrary, $T^N[\mathbf{1}_K](x_0) \leq \inf_{\gamma\in\Gamma_d} r^{\mu^*_{0\to N},\gamma}_{x_0}(K, K')$, $\forall x_0 \in X$. Thus, $T^N[\mathbf{1}_K](x_0) \leq r^*_{x_0}(K, K')$, $\forall x_0 \in X$ as desired. □

Proposition 3.3.

(a) $\forall x_0 \in X$, $r^*_{x_0}(K, K') \leq T^N[\mathbf{1}_K](x_0)$.

(b) There exists $\gamma^* \in \Gamma_d$ such that, for any $\mu \in \mathcal{M}_a$, $r_{x_0}^{\mu,\gamma^*}(K,K') \leq T^N[\mathbf{1}_K](x_0), \forall x_0 \in X$.

Proof. As in the proof of Proposition 3.2, we first prove the following claim by induction on k: There exists $\gamma^*_{N-k\to N} = (\gamma^*_{N-k}, \gamma^*_{N-k+1}, \ldots, \gamma^*_{N-1}) \in \Gamma_d$ such that, for any $\mu = (\mu_{N-k}, \mu_{N-k+1}, \ldots, \mu_{N-1}) \in \mathcal{M}_a$, $V_{N-k}^{\mu,\gamma^*_{N-k\to N}}(x) \leq J_{N-k}(x)$, $\forall x \in X$.

Let $\mu = (\mu_0, \mu_1, \ldots, \mu_{N-1}) \in \mathcal{M}_a$ be arbitrary. The case of $k = 0$ is trivial. For the inductive step, assume that this holds for $k = h$. By the induction hypothesis, there exists a strategy $\gamma^*_{N-h\to N} = (\gamma^*_{N-h}, \gamma^*_{N-h+1}, \ldots, \gamma^*_{N-1}) \in \Gamma_d$ such that, for any $\mu \in \mathcal{M}_a$, $V_{N-h}^{\mu,\gamma^*_{N-h\to N}}(x) \leq J_{N-h}(x)$, $\forall x \in X$. Furthermore, by Proposition 3.1(b), there exists a Borel measurable function $\lambda^* : X \times A \to D$ such that for all $(x,a) \in X \times A$ the following holds:

$$\mathbf{1}_K(x) + \mathbf{1}_{K'\setminus K}(x) H(x, a, \lambda^*(x,a), J_{N-h}) = G[J_{N-h}](x,a).$$

Choose a Markov strategy, $\gamma^*_{N-h-1\to N} = (\lambda^*, \gamma^*_{N-h}, \gamma^*_{N-h+1}, \ldots, \gamma^*_{N-1})$. Then by the monotonicity of the operator $T_{\pi,\lambda}$ and Lemma 3.3, we have for each $x \in X$:

$$\begin{aligned}
V_{N-h-1}^{\mu,\gamma^*_{N-h-1\to N}}(x) &= T_{\mu_{N-h-1},\lambda^*}[V_{N-h}^{\mu,\gamma^*_{N-h\to N}}](x) \\
&\leq T_{\mu_{N-h-1},\lambda^*}[J_{N-h}](x) \\
&= \mathbf{1}_K(x) + \mathbf{1}_{K'\setminus K}(x) H(x, \mu_{N-h-1}(x), \lambda^*(x, \mu_{N-h-1}(x)), J_{N-h}) \\
&= G[J_{N-h}](x, \mu_{N-h-1}(x)) \\
&\leq \sup_{a\in A} G[J_{N-h}](x,a) \\
&= T[J_{N-h}](x) = J_{N-h-1}(x),
\end{aligned}$$

which concludes the proof of the claim.

This result implies that there exists $\gamma^*_{0\to N} = (\gamma^*_0, \gamma^*_1, \ldots, \gamma^*_{N-1}) \in \Gamma_d$ such that, for any $\mu = (\mu_0, \mu_1, \ldots, \mu_{N-1}) \in \mathcal{M}_a$, $r_{x_0}^{\mu,\gamma^*_{0\to N}}(K,K') = V_0^{\mu,\gamma^*_{0\to N}}(x_0) \leq J_0(x_0) = T^N[\mathbf{1}_K](x_0)$, $\forall x_0 \in X$. Thus, $\gamma^*_{0\to N}$ is the Markov strategy satisfying statement (b) and $r_{x_0}^{\mu}(K,K') = \inf_{\gamma\in\Gamma_d} r_{x_0}^{\mu,\gamma}(K,K') \leq T^N[\mathbf{1}_K](x_0)$, for any $\mu \in \mathcal{M}_a$ and $x_0 \in X$. Since μ is arbitrary, $r^*_{x_0}(K,K') \leq T^N[\mathbf{1}_K](x_0)$, $\forall x_0 \in X$, proving statement (a). □

Combining the results of Proposition 3.2 and 3.3, we can now prove Theorem 3.1.

Proof. Statement (a) of Theorem 3.1 follows directly from the inequalities in Proposition 3.2(a) and Proposition 3.3(a). By Proposition 3.2(b) and statement (a) of the theorem, there exists a Markov policy $\mu^* \in \mathcal{M}_a$ such that, for any $\gamma \in \Gamma_d$, $r^*_{x_0}(K,K') \leq r_{x_0}^{\mu^*,\gamma}(K,K')$, $\forall x_0 \in X$. This implies that $r^*_{x_0}(K,K') \leq r_{x_0}^{\mu^*}(K,K')$, $\forall x_0 \in X$. On the other hand, the reverse inequality always holds: $r_{x_0}^{\mu^*}(K,K') \leq r^*_{x_0}(K,K')$, $\forall x_0 \in X$. This shows that μ^* is a maxmin policy. Similarly, by Proposition 3.3(b) and statement (a) of the theorem, there exists a Markov strategy $\gamma^* \in \Gamma_d$ such that, for any $\mu \in \mathcal{M}_a$, $r_{x_0}^{\mu,\gamma^*}(K,K') \leq r^*_{x_0}(K,K'), \forall x_0 \in X$. Thus, we

have statement (b). Finally, statement (c) follows directly from the proof of Proposition 3.2 and Proposition 3.3. □

Implications of the theorem

(a) Robust optimal policy: By statement (b) of Theorem 3.1, if the control were to choose μ^* and the adversary were to deviate from γ^*, then the reach-avoid probability will be at least $r^*_{x_0}(K, K')$. On the other hand, if the control were to deviate from the maxmin policy and the adversary were to choose the worst-case Markov strategy, then the reach-avoid probability will be at most $r^*_{x_0}(K, K')$. Thus, μ^* can be interpreted as a robust control policy which optimizes the worst-case probability for achieving the reach-avoid objective.

(b) Control synthesis: Equations (3.9) and (3.10) provide us with sufficient conditions for optimality of the players' policies and strategies. In particular, Equation (3.9) can be used to synthesize a maxmin control policy for player 1 from the value functions computed through the dynamic programming recursion. To illustrate, suppose that the input ranges A and D along with the state space X have been appropriately discretized, for example according to the method suggested in [120]. Then for each system state $x \in K' \setminus K$ in the grid, at the k-th iteration of the dynamic programming algorithm, we can compute and store an optimal control input

$$a^* \in \arg\sup_{a \in A} \inf_{d \in D} H(x, a, d, J_{k+1}).$$

This provides us with a discretized representation of the one-step maxmin control policy μ^*_{N-k} on a grid of the continuous state space within each mode. Storing these values as lookup tables then allows us to select control inputs in an optimal fashion as state measurements are received.

(c) Probabilistic reach-avoid set: Consider the case in which it is required from the system design perspective to have a reach-avoid probability greater than some threshold $(1 - \epsilon)$, for $\epsilon \in [0, 1)$. The set of initial conditions X_ϵ for which this specification is feasible, under the worst-case adversary behavior, can be derived from the maxmin reach-avoid probability as

$$X_\epsilon = \{x_0 \in X : r^*_{x_0}(K, K') \geq (1 - \epsilon)\}.$$

In other words, X_ϵ is the $(1 - \epsilon)$-sublevel set of the reach-avoid probability map $r^*_{x_0}(K, K')$, $x_0 \in X$. A conceptual illustration of such a set is shown in Figure 3.2(b).

(d) Numerical computation: For a few problems, such as the example in the next section, analytic computation of $r^*_{x_0}(K, K')$ may be possible. In general, there may not be a closed-form expression for the operator T. The computation of the dynamic programming can be done on a discretized grid of the continuous state and input spaces for each mode. For a given grid point x_g, inputs a_g and d_g in the discretized input spaces A and D respectively, the hybrid transition probability $\tau(x'_g | x_g, a_g, d_g)$ for each x'_g in the grid can be approximated by

integration of $\tau(dx|x_g, a_g, d_g)$ over the grid volume. Alternatively, Monte Carlo simulation may be used to approximate these transition probabilities. Regardless of the method for evaluating the transition probabilities, it can be observed that the computational cost of a discretization approach scales exponentially with the dimensions of the continuous state space and player input spaces, which currently limits the application of our approach to problems with continuous state dimensions of $n \leq 4$.

Example 3.2 (A reach-avoid example with analytic solution)**.** In order to illustrate the procedure for computing the reach-avoid probability, the maxmin player 1 policy and the worst-case player 2 strategy, we describe a simple reach-avoid problem for which an analytic solution can be obtained. Specifically, consider the system dynamics given in Example 3.1, and a regulation problem where the objective of player 1 is to drive the continuous state into a neighborhood of the origin, while staying within some safe operating region. In this case, the target set and safe set are chosen to be $K = \{q^1, q^2\} \times [-\frac{1}{4}, \frac{1}{4}]$ and $K' = \{q^1, q^2\} \times [-2, 2]$, respectively. The time horizon is chosen to be $N = 1$.

First, we observe that the stochastic kernels τ_v and τ_r are continuous in a and d, while τ_q is independent of the players' inputs. Thus, Assumption 3.1 is satisfied. For a given a function $J : X \to \mathbb{R}$, the value of the map $H(x, a, d, J)$ for a hybrid state x, in discrete mode 1, that is, $x = (q^1, v)$, is derived as

$$\begin{aligned} H\big((q^1, v), a, d, J\big) &= \int_X J(x')\tau(dx'|(q^1, v), a, d) \\ &= \tau_q(q^1|(q^1, v), a, d) \int_{\mathbb{R}} J(q^1, v')\tau_v(dv'|(q^1, v), a, d) \\ &+ \tau_q(q^2|(q^1, v), a, d) \int_{\mathbb{R}} J(q^2, v')\tau_r(dv'|(q^1, v), a, d, q^2) \\ &= p_1 \int_{-1}^{1} J(q^1, 2v + a + d + \eta)d\eta + (1 - p_1) \int_{-1}^{1} J(q^2, 2v + a + d + \eta)d\eta. \end{aligned} \tag{3.19}$$

Similarly, we can derive $H(x, a, d, J)$ for $x = (q^2, v)$. Given the form of the target set K, the dynamic programming recursion is initialized by the function

$$\mathbf{1}_K(q, v) = \begin{cases} 1, & |v| \leq \frac{1}{4},\ q = q^1, q^2 \\ 0, & |v| > \frac{1}{4},\ q = q^1, q^2 \end{cases}$$

By Theorem 3.1, the maxmin reach-avoid probability $r^*_{x_0}(K, K')$ for an initial condition $x_0 = (q_0, v_0)$ can be computed as

$$T[\mathbf{1}_K](q_0, v_0) = \begin{cases} 1, & |v_0| \leq \frac{1}{4},\ q_0 = q^1, q^2 \\ 0, & |v_0| > 2,\ q_0 = q^1, q^2 \\ \sup_{a \in A} \inf_{d \in D} H\big((q_0, v_0), a, d, \mathbf{1}_K\big), & \frac{1}{4} < |v_0| \leq 2,\ q_0 = q^1, q^2 \end{cases} \tag{3.20}$$

From the above, we see that the dynamic programming step only needs to be carried out on the set $K' \setminus K = \{q^1, q^2\} \times [-2, -\frac{1}{4}) \cup (\frac{1}{4},\ 2]$. From equation (3.19), it can be verified that for $q_0 = q^1$

$$H\big((q^1, v_0), a, d, \mathbf{1}_K\big) = \begin{cases} \frac{1}{4}, & 0 \le |2v_0 + a + d| \le \frac{3}{4} \\ \frac{5}{8} - \frac{1}{2}|2v_0 + a + d| & \frac{3}{4} < |2v_0 + a + d| \le \frac{5}{4} \\ 0, & |2v_0 + a + d| > \frac{5}{4} \end{cases} \tag{3.21}$$

Combining equations (3.20) and (3.21), the maxmin reach-avoid probability for the initial condition $x_0 = (q^1, v_0)$ can be derived as

$$r^*_{x_0}(K, K') = T[\mathbf{1}_K](q^1, v_0) = \begin{cases} 1 & |v_0| \le \frac{1}{4} \\ \frac{1}{8} & \frac{1}{4} < |v_0| \le \frac{1}{2} \\ \frac{5}{8} - |v_0| & \frac{1}{2} < |v_0| \le \frac{5}{8} \\ 0 & |v_0| > \frac{5}{8} \end{cases}$$

In the process of performing the dynamic programming step in (3.20), we also obtain a maxmin player 1 policy μ^*_0, and a worst-case player 2 strategy γ^*_0, in mode q^1 satisfying the sufficient conditions for optimality in (3.9) and (3.10) as follows:

$$\mu^*_0(q^1, v_0) = \begin{cases} 1, & |v_0| > \frac{1}{2} \\ -2v_0, & |v_0| \le \frac{1}{2} \end{cases} \qquad \gamma^*_0((q^1, v_0), a) = \begin{cases} -1, & 2v_0 + a < 0 \\ 1, & 2v_0 + a \ge 0 \end{cases}$$

Similarly, we can compute the maxmin reach-avoid probability for $x_0 = (q^2, v_0)$ as

$$r^*_{x_0}(K, K') = T[\mathbf{1}_K](q^2, v_0) = \begin{cases} 1, & |v_0| \le \frac{1}{4} \\ \frac{1}{8}, & \frac{1}{4} \le |v_0| \le 2 \\ 0, & |v_0| > 2 \end{cases}$$

Furthermore, a maxmin player 1 policy and a worst-case player 2 strategy satisfying the sufficient conditions for optimality in mode q^2 can be derived and are given as

$$\mu^*_0(q^2, v_0) = \begin{cases} 1, & |v_0| > 2 \\ -\frac{1}{2}v_0, & |v_0| \le 2 \end{cases} \qquad \gamma^*_0((q^2, v_0), a) = \begin{cases} -1, & \frac{1}{2}v_0 + a < 0 \\ 1, & \frac{1}{2}v_0 + a \ge 0 \end{cases}$$

Specialization to Stochastic Safety Problem

Consider the probabilistic safety problem described in [46], in which the objective of player 1 is to keep the system state within a given safe set $S \in \mathcal{B}(X)$ over some finite time horizon

$[0, N]$, while the objective of player 2 is again opposed to that of player 1. Similar to the reach-avoid derivation, it can be shown that the probability that the hybrid state trajectory $(x_0, x_1, \ldots, x_N)$ remains in S under fixed choices of $\mu \in \mathcal{M}_a$ and $\gamma \in \Gamma_d$ can be formulated as a sum-multiplicative cost

$$p_{x_0}^{\mu,\gamma}(S) := P_{x_0}^{\mu,\gamma}(S^{N+1}) = E_{x_0}^{\mu,\gamma}\left[\prod_{k=0}^{N} \mathbf{1}_S(x_k)\right].$$

The connection between the safety problem and reach-avoid problem is established by the observation that the hybrid state remains inside a set S for all $k = 0, 1, \ldots, N$ if and only if it does not reach $X \backslash S$ for any $k = 0, 1, \ldots, N$. Mathematically speaking, for any $\mu \in \mathcal{M}_a$ and $\gamma \in \Gamma_d$

$$p_{x_0}^{\mu,\gamma}(S) = 1 - r_{x_0}^{\mu,\gamma}(X \backslash S, X). \tag{3.22}$$

The solution to the probabilistic safety problem can be obtained from a complementary reach-avoid problem. In particular, consider a reach-avoid problem with the value function

$$\bar{r}_{x_0}^{*}(X \setminus S, X) = \inf_{\mu \in \mathcal{M}_a} \sup_{\gamma \in \Gamma_d} r_{x_0}^{\mu,\gamma}(X \backslash S, X),\ x_0 \in X.$$

Then the maxmin probability of safety is given by

$$p_{x_0}^{*}(S) = \sup_{\mu \in \mathcal{M}_a} \inf_{\gamma \in \Gamma_d} p_{x_0}^{\mu,\gamma}(S) = 1 - \bar{r}_{x_0}^{*}(X \setminus S, X),\ x_0 \in X. \tag{3.23}$$

By minor modifications of the proof for Theorem 3.1, it is not difficult to see that $\bar{r}_{x_0}^{*}(X \backslash S, X)$ can be computed by the dynamic programming recursion

$$\bar{r}_{x_0}^{*}(X \setminus S, X) = T_S^N[\mathbf{1}_{X \setminus S}](x_0),\ x_0 \in X,$$

where the operator T_S is defined as

$$T_S[J](x) = \inf_{a \in A} \sup_{d \in D} \mathbf{1}_{X \setminus S}(x) + \mathbf{1}_S(x) H(x, a, d, J),\ x \in X. \tag{3.24}$$

The corresponding maxmin probability of safety can then be obtained through (3.23).

For completeness, we note that there exists an equivalent dynamic programming recursion to compute the safety probability, similar to the one given in [46] for the single player case. Specifically, consider an operator $\tilde{T}_S$ defined as

$$\tilde{T}_S[J](x) = \sup_{a \in A} \inf_{d \in D} \mathbf{1}_S(x) H(x, a, d, J),\ x \in X. \tag{3.25}$$

The relation between $\tilde{T}_S$ and T_S is established through the following lemma.

Lemma 3.4. *For every $x \in X$ and $k = 0, 1, \ldots, N$, $\tilde{T}_S^k[\mathbf{1}_S](x) = 1 - T_S^k[\mathbf{1}_{X\setminus S}](x)$.*

Proof. We prove this result by induction on k. The case of $k = 0$ is established by the fact that $\mathbf{1}_S = 1 - \mathbf{1}_{X\setminus S}$. Now suppose the identity holds for $k = h$, then $\forall x \in X$,

$$\begin{aligned}
\tilde{T}_S^{h+1}[\mathbf{1}_S](x) &= \tilde{T}_S(\tilde{T}_S^h(\mathbf{1}_S))(x) = \tilde{T}_S[1 - T_S^h[\mathbf{1}_{X\setminus S}]](x) \\
&= \sup_{a\in A} \inf_{d\in D} \mathbf{1}_S(x) H(x, a, d, 1 - T_S^h[\mathbf{1}_{X\setminus S}]) \\
&= \sup_{a\in A} \inf_{d\in D} \mathbf{1}_S(x)(1 - H(x, a, d, T_S^h[\mathbf{1}_{X\setminus S}])) \\
&= \mathbf{1}_S(x) + \sup_{a\in A} \inf_{d\in D} -\mathbf{1}_S(x) H(x, a, d, T_S^h[\mathbf{1}_{X\setminus S}]).
\end{aligned}$$

It then follows that for every $x \in X$, we have:

$$\begin{aligned}
1 - \tilde{T}_S^{h+1}(\mathbf{1}_S)(x) &= 1 - \mathbf{1}_S(x) - \sup_{a\in A} \inf_{d\in D} -\mathbf{1}_A(x) H(x, a, d, T_S^h[\mathbf{1}_{X\setminus S}]) \\
&= \mathbf{1}_{X\setminus S}(x) + \inf_{a\in A} \sup_{d\in D} \mathbf{1}_S(x) H(x, a, d, T_S^h[\mathbf{1}_{X\setminus S}]) \\
&= T_S[T_S^h(\mathbf{1}_{X\setminus S}])(x) = T_S^{h+1}[\mathbf{1}_{X\setminus S}](x),
\end{aligned}$$

which completes the proof. □

Thus, an equivalent recursion for computing the maxmin safety probability is given by

$$p_{x_0}^*(S) = \tilde{T}_S^N[\mathbf{1}_S](x_0),\ x_0 \in X. \tag{3.26}$$

Using either the operator T_S or the operator $\tilde{T}_S$, we can also derive sufficient conditions of optimality for player 1 and player 2, similar to those given in Equations (3.9) and (3.10).

Consideration of Alternative Information Patterns

In the previous sections, we considered a non-symmetric information pattern which gives an advantage to player 2, namely the intent of player 1 is available to player 2 at each time step. For the discussions in this section, we refer to this as *Scenario I*. Suppose instead that the actions of the adversary are observed. For example in a patrol and surveillance application the actions of an intruder are assumed to be captured by a surveillance system. Then one can reasonably consider an alternative information pattern in which player 1 selects Markov strategies and player 2 selects Markov policies. We refer to such cases as *Scenario II*.

In Scenario II, player 1 chooses a Borel measurable Markov strategy at each step, $\mu_k : X \times D \to A$, $k = 0, 1, ..., N-1$. The set of such strategies is denoted by Γ_a. Player 2 on the other hand, chooses a Borel measurable Markov policy $\gamma_k : X \to D$, $k = 0, 1, ..., N-1$. The

set of such policies is denoted by $\mathcal{M}_d$. We briefly note that Markov policies are a subclass of Markov strategies, namely they consist of the set of Markov strategies which do not explicitly depend on the input of the other player. Thus, we have that $\mathcal{M}_a \subset \Gamma_a$ and $\mathcal{M}_d \subset \Gamma_d$.

Using a similar construction as in Section 3.2, we can define for a given Markov strategy $\mu \in \Gamma_a$ and a given Markov policy $\gamma \in \mathcal{M}_d$, a closed-loop stochastic kernel at time step k by $\tilde{\tau}^{\mu_k,\gamma_k}(\cdot|x_k) := \tilde{\tau}(\cdot|x_k, \mu_k(x_k, \gamma_k(x_k)), \gamma_k(x_k))$. As before, this induces a probability measure, denoted by $\tilde{P}_{x_0}^{\gamma,\mu}$, on the sample space Ω. Note that if $\mu^1 \in \mathcal{M}_a$ and $\mu^2 \in \mathcal{M}_d$ are Markov policies for both players, then the probability measures in Scenario I and II are equivalent: $\tilde{P}_{x_0}^{\mu^1,\mu^2} \equiv P_{x_0}^{\mu^1,\mu^2}$.

Let $\tilde{E}_{x_0}^{\mu,\gamma}$ denote the expectation with respect to the probability measure $\tilde{P}_{x_0}^{\mu,\gamma}$ on the sample space Ω. Under Scenario II, the reach-avoid probability for a given Markov strategy μ, and Markov policy γ, is

$$\tilde{r}_{x_0}^{\mu,\gamma}(K, K') = \tilde{E}_{x_0}^{\mu,\gamma}\left[\mathbf{1}_K(x_0) + \sum_{j=1}^{N}\left(\prod_{i=0}^{j-1}\mathbf{1}_{K'\setminus K}(x_i)\right)\mathbf{1}_K(x_j)\right]. \tag{3.27}$$

In this scenario, we are interested in computing the *minmax* value function defined as

$$\tilde{r}_{x_0}^{*}(K, K') := \inf_{\gamma\in\mathcal{M}_d}\sup_{\mu\in\Gamma_a}\tilde{r}_{x_0}^{\mu,\gamma}(K, K'),\ \forall x_0 \in X.$$

In addition, we are interested in finding the *minmax* strategy $\mu^* \in \Gamma_a$, if it exists, such that $\tilde{r}_{x_0}^{*}(K, K') = \inf_{\gamma\in\mathcal{M}_d}\tilde{r}_{x_0}^{\mu^*,\gamma}(K, K')$, $\forall x_0 \in X$.

By a proof analogous to that of Theorem 3.1, the minmax value function can be computed by a suitable dynamic programming recursion. More precisely, consider a dynamic programming operator $\tilde{T}$ acting on $\mathcal{F}$, the Borel measurable functions from X to $[0, 1]$:

$$\tilde{T}[J](x) = \inf_{d\in D}\sup_{a\in A}\mathbf{1}_K(x) + \mathbf{1}_{K'\setminus K}(x)H(x, a, d, J),\ x \in X. \tag{3.28}$$

Then main result for Scenario II is then as follows:

Theorem 3.2. *Let $\mathcal{H} = (Q, n, A, D, \tau_v, \tau_q, \tau_r)$ be a DTSHG satisfying Assumption 3.1, with the information pattern for Scenario II. Let $K, K' \in \mathcal{B}(X)$ be Borel subsets such that $K \subseteq K'$. Then the composition $\tilde{T}^N = \tilde{T} \circ \tilde{T} \circ \cdots \circ \tilde{T}$ is well-defined and $\forall x_0 \in X$:*

(a) $\tilde{r}_{x_0}^{*}(K, K') = \tilde{T}^N(\mathbf{1}_K)(x_0)$;

(b) There exists a Markov strategy $\mu^* \in \Gamma_a$ such that $\tilde{r}_{x_0}^{*}(K, K') = \inf_{\gamma\in\mathcal{M}_d}\tilde{r}_{x_0}^{\mu^*,\gamma}(K, K')$;

(c) There exists a Markov policy $\gamma^* \in \mathcal{M}_d$ such that $\tilde{r}_{x_0}^{*}(K, K') = \tilde{r}_{x_0}^{\mu^*,\gamma^*}(K, K')$.

Given the forms of the recursions in (3.7) and (3.28), it is easy to verify that

$$T^N[\mathbf{1}_K] \leq \tilde{T}^N[\mathbf{1}_K],$$

which agrees with the intuition that player 1 should do better under the information pattern in Scenario II as compared with Scenario I.

In certain applications, one may be more interested in a symmetric information pattern in which both players make decisions based only upon the state of the system at each time step and the intent of neither player is available to the opposing player. More formally, player 1 is constrained to choose policies $\mu^1 \in \mathcal{M}_a$, while player 2 is constrained to choose policies $\mu^2 \in \mathcal{M}_d$. By the fact that the set of Markov policies is a subset of the set of Markov strategies, this case can be viewed as a subset of either Scenario I or Scenario II. In addition, in this case, we have by the equivalence of $\tilde{P}_{x_0}^{\mu^1,\mu^2}$ and $P_{x_0}^{\mu^1,\mu^2}$ that $\tilde{r}_{x_0}^{\mu^1,\mu^2}(K,K') = r_{x_0}^{\mu^1,\mu^2}(K,K')$, $\forall x_0 \in X$. For this symmetric information pattern, the value functions $\inf_{\mu^1\in\mathcal{M}_a}\sup_{\mu^2\in\mathcal{M}_d} r_{x_0}^{\mu^1,\mu^2}(K,K')$ and $\sup_{\mu^2\in\mathcal{M}_d}\inf_{\mu^1\in\mathcal{M}_a} r_{x_0}^{\mu^1,\mu^2}(K,K')$ are referred to as the upper and lower value functions respectively. It is not too difficult to show that the lower value function is less than or equal to the upper value function (as consistent with the naming). In general, conditions under which these value functions are equal are of interest but difficult to obtain.

3.4 Pairwise Aircraft Conflict Detection

The collision detection scenario considered here involves two aircraft with possibly intersecting nominal trajectories. From the perspective of the first aircraft, the task is to detect the possibility of conflict given the current position of another aircraft, and design a collision avoidance trajectory in case a potential conflict is detected. This problem has been studied with significant detail in [5] within a deterministic setting. Motivated by practical concerns of wind influence on aircraft trajectories and consequently, on accuracy of conflict detection [121], we consider the stochastic wind model proposed in [106]. Conflict detection becomes a probabilistic safety problem involving two players (aircraft 1 and 2), in which the unsafe set is all aircraft states closer than an allowable distance.

Let $(x^1, x^2, x^3) \in \mathbb{R}^2 \times [0, 2\pi]$ denote, respectively, the 2D position and the heading angle of aircraft 2 in the reference frame of aircraft 1. We model each aircraft as a unicycle and discretize the dynamics in relative coordinates [5] to obtain the deterministic part of the dynamics as follows:

$$\begin{bmatrix} x_{k+1}^1 \\ x_{k+1}^2 \\ x_{k+1}^3 \end{bmatrix} = f(x_k, a_k, d_k) := \begin{bmatrix} x_k^1 + \Delta t(-s^1 + s^2\cos(x_k^3) + a_k x_k^2) \\ x_k^2 + \Delta t(s^2\sin(x_k^3) - a_k x_k^1) \\ x_k^3 + \Delta t(d_k - a_k) \end{bmatrix}. \tag{3.29}$$

In the above, Δt is the discretization step, s^i is the speed of aircraft i, which is assumed to be constant in this case study, a and d are the angular velocities of aircraft 1 and 2 respectively and are the inputs of each aircraft.

In order to develop a realistic model for the effects of wind in relative coordinates, we consider the stochastic model of the wind as described in [106]. In this work, wind is modeled as having a deterministic known component and a stochastic component. In this scenario, for simplicity, we ignore the deterministic known component of the wind. The stochastic wind component is modeled, in continuous time, as a time dependent random field over the $2D$ space. In particular, at each planar position $(x^1, x^2) \in \mathbb{R}^2$, the stochastic wind component has the distribution $\sigma dB(x^1, x^2, t)$ in which B is a position dependent Brownian motion and σ is a positive constant. As such, the aircraft positions are correlated due to the presence of wind. It is then shown in [106] that the wind in relative coordinates has the distribution

$$\omega^1(t) = \sigma\sqrt{2(1 - h(x^1, x^2))}W^1(t),$$
$$\omega^2(t) = \sigma\sqrt{2(1 - h(x^1, x^2))}W^2(t),$$

where $W(t) = (W^1(t), W^2(t))$ is a standard Brownian motion and $h : \mathbb{R}^2 \to \mathbb{R}$ is referred to as the spatial correlation function. A choice for h which is suitable for air traffic applications is $h(x^1, x^2) = \exp(-\beta\|(x^1, x^2)\|)$, where β is a positive constant. Consequently, the random variable modeling wind in discrete-time relative coordinates has a Gaussian distribution with zero mean and position dependent covariance matrix $\Sigma(x^1, x^2) = 2(\Delta t\sigma)^2(1 - h(x^1, x^2))I_2$,

$$(\omega_k^1, \omega_k^2) \sim \mathcal{N}\big(0, \Sigma(x^1, x^2)\big). \tag{3.30}$$

Whereas in [106] conflict was predicted assuming a nominal aircraft trajectory perturbed by wind, here, we consider detection and resolution of conflict by including aircraft inputs in the model. Thus, for any initial relative position of aircraft, we associate both the minimum probability of conflict, under appropriate communication and coordination schemes, and also an optimal policy for the aircraft that achieves this minimum. In addition, we capture the effects of actuator noise on the angular velocity of either aircraft through introducing the random variable ω^3, with a Gaussian distribution $\omega_k^3 \sim \mathcal{N}(0, \sigma_a)$.

Let $\omega_k \sim \mathcal{N}(0, \Sigma(x)) \in \mathbb{R}^3$ denote the stochastic uncertainty due to wind and actuator noise, where $\Sigma(x) = \text{diag}(\Sigma(x^1, x^2), \sigma_a) \in \mathbb{R}^{3\times 3}$ is a block diagonal covariance matrix. The stochastic equations of motion are then given as $x_{k+1} = f(x_k, a_k, d_k) + \omega_k$.

In air traffic management, a conflict between two aircraft is defined if the aircraft get closer than a critical distance, R_c. Hence, the safe set in 2D can be defined as

$$S = \{(x^1, x^2) \in \mathbb{R}^2 \; s.t. \; \|(x^1, x^2)\|_2 \geq R_c\}.$$

In a collision detection and resolution, the choice of communication protocol becomes an important parameter. Here, we assume that the position of each aircraft is available to both

aircraft 1 and 2, for example through Automatic Dependent Surveillance-Broadcast (ADS-B) network at each time step. For the conflict resolution scenario, we consider two control models as described below.

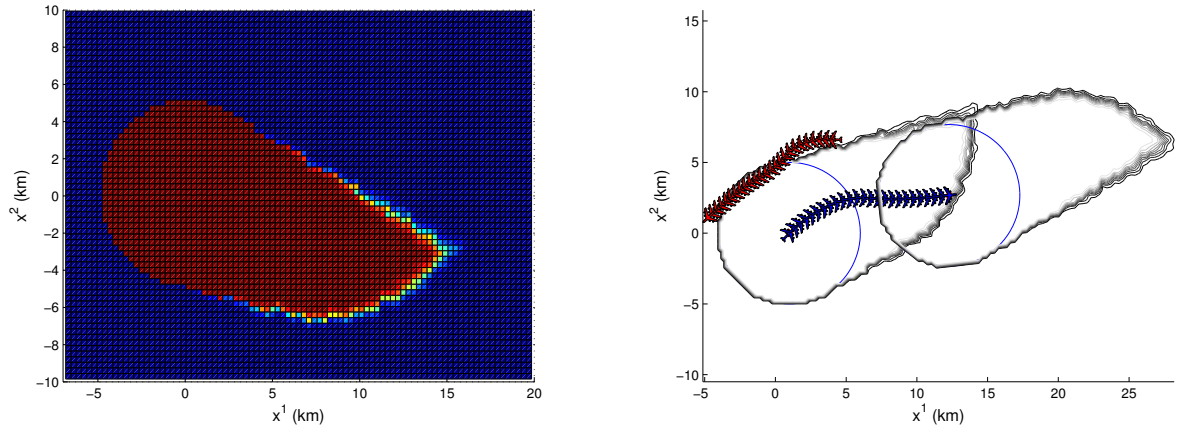
Model 1: We assume that the control of the two aircraft is decentralized. Namely, there are no air traffic controllers coordinating the choice of input of each aircraft. Furthermore, in the absence of further information on the decision algorithms, each aircraft assumes that the other aircraft could potentially make choices of inputs that endanger safety. Thus, the conflict resolution problem is also non-cooperative. Conflict detection and resolution from perspective of aircraft 1 then becomes probabilistic safety under the worst-case aircraft 2 input. Hence, the maxmin probability of safety, $p^*_{x_0}(S)$, and aircraft 1's maxmin control policy, $\mu^* \in \mathcal{M}_a$, which achieves this probability must be computed. Based on Section 3.3, the solution to this problem can be obtained either through the specialized recursion (3.26) for the safety problem or from an equivalent reach-avoid problem and the resulting recursion defined in Theorem 3.1.

Model 2: In the second model, we assume that the control of the two aircraft is centralized. This can be either due to the presence of an air traffic controller or direct communication via the ADS-B communication network between the aircraft. Then we can assume that the aircraft are cooperating to avoid collision. As such, both aircraft want to maximize probability of safety. The problem is then to compute the *maxmax* probability of safety, as well as optimal control policies for the two aircraft which achieve this probability. Since both players are optimizing the same objective, maximizing probability of safety can be formulated using the single-player verification formulation of [47].

Motivated by discrete maneuvers currently used in air traffic management, we consider a scenario in which at any given time, each aircraft can choose to be in one of the three flight maneuvers: straight, right turn, or left turn, corresponding to the angular velocity of $\bar{a} \in \mathbb{R}$. As such, we consider the input sets $A = D = \{0, -\bar{a}, \bar{a}\}$ for the aircraft.

For the numerical results included here, the parameters of the problem are chosen as follows: The sampling time is set to $\Delta t = 0.1$ minute, the time horizon to 2.5 minutes, $R_c = 5$ km, the aircraft speed to $s^1 = s^2 = 5$ km per minute and the angular velocity to $\bar{a} = 1$ radians per minute. The covariance parameters are set to $\sigma = 0.73$ and $\sigma_a = 0.26$. The constant β in the function h is chosen as 0.1. Computation is performed over a subset of the state space given by $[-7, 20] \times [-10, 10] \times [0, 2\pi]$, on a grid size of $90 \times 67 \times 65$.

For this problem, the minmax probability of collision, $\bar{r}^*_{x_0}(X \setminus S, X)$, which is equivalent to $1 - p^*_{x_0}(S)$, with $p^*_{x_0}(S)$ being the maxmin probability of safety, is computed. Figure 3.3(a) shows $\bar{r}^*_{x_0}(X \setminus S, X)$ for the set of initial conditions with relative heading of $\frac{3\pi}{4}$ radians. The interpretation of this probability map is as follows: Consider an initial condition of (6.79 km, 2.55 km, $\frac{3\pi}{4}$ rad). From the value function we obtain $\bar{r}^*_{x_0}(X \setminus S, X) \approx 0.013$. This means that if aircraft 1 selects flight maneuvers according to the minmax policy μ^* and aircraft 2 selects maneuvers according to the worst-case strategy γ^*, then the probability of



(a) Minmax probability of collision

(b) Aircraft trajectories

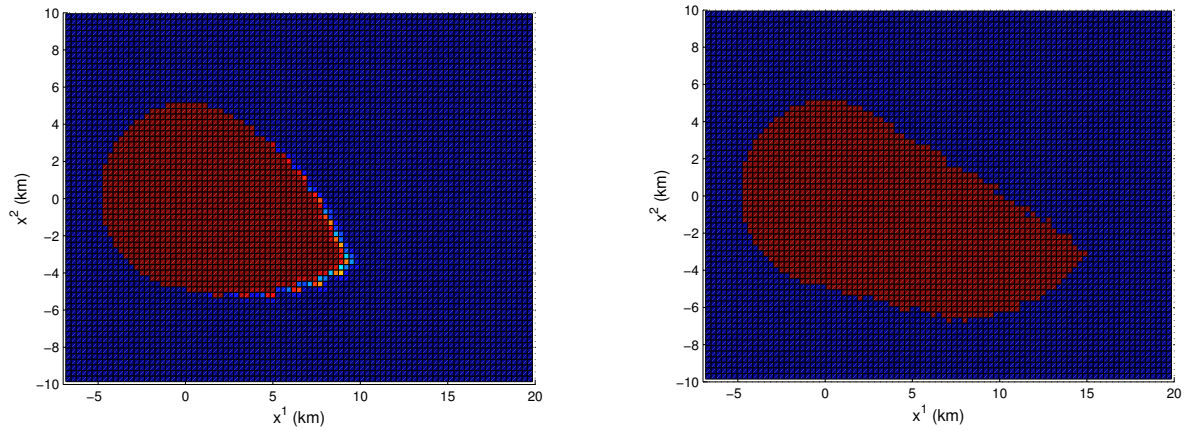
Figure 3.3: Minmax probability of collision

collision within a 2.5 minute time horizon is approximately 1.3%. Furthermore, if aircraft 2 were to deviate from the worst-case strategy γ^*, while aircraft 1 selected maneuvers according to μ^*, then the probability of collision would remain at most 1.3%. On the other hand, if aircraft 1 were to deviate from the minmax policy μ^*, while aircraft 2 selected headings according to γ^*, then the probability of collision may be greater than 1%. Thus, aircraft 1 has an incentive for choosing the minmax policy as a robust control policy to counter the worst-case behavior by aircraft 2.

In Figure 3.3(b) an execution of the aircraft trajectories, with aircraft 1 initial condition at $(1\text{ km}, 0\text{ km}, \frac{\pi}{4}\text{ rad})$ and aircraft 2 initial condition at $(4\text{ km}, 6.6\text{ km}, \pi\text{ rad})$, based on the maxmin control policy and worst-case strategy is shown. The contours of the probability map $\bar{r}^*_{x_0}(X \setminus S, X)$, at the initial condition are also drawn. In addition, the boundary of the unsafe set is shown as a circle, centered at aircraft 1 initial and final positions. In this particular execution, aircraft 1 is able to avoid collision due to its choice of policy.

In contrast, the minmin probability of collision, equivalent to the maxmax probability of safety, associated with *Model 2* is shown in Figure 3.4(a). Notice that, as expected, when the aircraft are cooperating, the probability of safety is greater than the previous case in which the worst-case aircraft 2 behavior was assumed.

For comparison, we computed the maxmin backward reachable sets under the assumption of no stochastic noise in the dynamics. This deterministic backward reachable set for the set of initial conditions with relative heading of $\frac{3\pi}{4}$ rad is shown in Figure 3.4(b). In this figure, we see the set of initial conditions such that for any control policy for aircraft 1, there exists a control strategy for aircraft 2 that leads to a conflict.



(a) Minmin probability of collision

(b) Deterministic reachable set

Figure 3.4: Cooperative and deterministic collision avoidance

3.5 Conclusions

In this chapter, we motivated and formulated the probabilistic safety and reach-avoid problem for discrete-time stochastic hybrid systems as a zero-sum stochastic game between a control (player 1) and an adversary (player 2). Under certain assumptions on the input spaces of the two players, it was shown that there always exists a Markov control policy which guarantees a maxmin probability of achieving the reach-avoid objectives, regardless of the adversarial strategy. Furthermore, this worst-case probability can be computed via an appropriate dynamic programming recursion. We discussed how various information patterns related to players' knowledge of the game and their communication abilities can be accounted for in the formulation and solution approach.

A practical example from the air traffic collision detection and resolution domain was provided to illustrate the application of the proposed approach in the current air traffic practice, under two communication and coordination models. A stochastic model of the wind influence on aircraft dynamics based on [106] was considered. In order to consider more realistic air traffic management scenarios, a good model of nominal and stochastic wind components obtained from the forecast data must be included in aircraft dynamics. The NextGen vision of improved operations requires ADS-B equipped aircraft to be able to operate with the non-equipped aircraft so that transitions to improved air traffic management system occur smoothly and gradually. As such, our proposed algorithm for conflict detection and resolution needs to be tested under various communication protocols between the aircraft and the ground station. Additionally, the algorithm needs to account for the cases in which measurements of aircraft state may not be available due to possible failures of sensing or com-

munication. To address such failures, formulations of the safety and reach-avoid problems with imperfect state information must be considered.

It is important to apply the algorithm to collision detection and resolution in multiple aircraft scenarios which may arise in en-route air traffic centers, with inclusion of altitude, heading and speed change maneuvers. However, as stated previously, current implementation of the algorithm is restricted to dimensions of less than 4, due to exponential scaling of the dynamic programming with respect to state dimensions. As such, first, it is required to establish convergence results on the approximation of the maxmin reach-avoid probability and optimal strategies through the state space discretization approach employed here, similar to those given in [122]. Second, numerical techniques need to be explored for speeding up the computation or breaking down large problems into a sequence of smaller sized problems which can be solved with the currently available numerical tools.

Finally, in many scenarios, there may be performance specifications on the system state, in addition to safety and reachability. For example, one may be interested in designing a collision avoidance maneuver while fuel consumption is minimized. We are currently working on extension of our methods to account for such cases through chance constrained and multi objective optimization frameworks.

Chapter 4

Extensions of Control Synthesis for Stochastic Hybrid Systems

We extend the reach-avoid problem of the previous chapter in two directions. First, motivated by the presence of uncertainty in weather forecast data for aircraft trajectory planning, we consider probabilistic models of the target and unsafe sets. We develop tools for verification and control synthesis for this class of problems. Next, motivated by applications in which the target set must be reached at any time prior to entering the unsafe set, we consider the reach-avoid problem in infinite horizon and prove convergence of the dynamic programming algorithm under appropriate assumptions. The material in this chapter is based on the papers presented in [55, 56] and a paper in preparation [57].

4.1 Random Sets in Reachability and Safety problems

In many applications, such as aircraft trajectory planning through hazardous weather, maneuvering a vehicle over an unknown terrain for exploration and disaster response, or steering a needle in a tissue for surgical procedures, the environment is only partially known. As such, the locations of the obstacles as well as the target sets are known with uncertainty and may change as more information is gathered about the environment. One method for ensuring obstacle avoidance and target acquisition constraints is by considering a robust approach in which the constraints need to be satisfied for any instance of uncertainty ranging over a specified domain. This worst-case approach, in general, may lead to overly conservative plans. In addition, the resulting robust optimization problems are difficult to solve unless assumptions are made about the robot dynamics and the environment uncertainty. Another method is to formulate a probabilistic model of the environment and then require that the constraints are satisfied with a desired probability.

The problem of chance-constrained programming was introduced as a general framework for decision making under probabilistic uncertainty [123]. Research has addressed this class of problems by making assumptions on the uncertainty model and the objective function or employing randomized algorithms [124]. Linear objective functions with probabilistic linear matrix inequality constraints are formulated as a convex optimization problem through sampling the constraints [125]. It is then shown that these probabilistic constraints can be converted to convex second order cone constraints for a wide class of probability distributions and hence solved efficiently [126]. Given that non-Gaussian distributions, in general, do not lead to an analytic constraint formulation, they may be approximated using a particle filter approach, and in certain instances, the resulting optimal control problem may be formulated as a Mixed Integer Linear Program (MILP) [127]. Since this approach is intractable with increasing the number of samples, various conservative approximations of the chance constraints, for example, using Boole's inequality [127, 128, 129] or ellipsoidal relaxations [130], have been introduced. Additionally, for certain classes of models randomized algorithms have been applied [131, 132]. Other methods include variations of the Probabilistic Roadmap and Rapidly-exploring Random Tree (RRT) to account for the probabilistic obstacles in the environment [133, 134, 135].

To account for the presence of uncertainty in the environment, we formulate the reach-avoid problem from the previous chapter with consideration of stochastic safe and target sets. We assume the sets can be modeled by a stochastic parameter together with a set-valued map. Based on certain assumptions for the set-valued stochastic process, we derive a dynamic programming algorithm for maximizing the reach-avoid probability which has a computational complexity independent of the dimensions of the parameters of the set.

Stochastic Set Model with Stochastic Hybrid System

As in the previous chapter, we let the system state space be represented by a hybrid set $X := \bigcup_{q \in Q} \{q\} \times \mathbb{R}^{n(q)}$, where the discrete state space is $Q := \{1, 2, \ldots, M\}$, $M \in \mathbb{N}$ and the map $n : Q \to \mathbb{N}$ assigns to each discrete state $q \in Q$ the dimension of the continuous state space $\mathbb{R}^{n(q)}$. Let d be a metric on X. Let $\mathcal{K}$ denote the set of all closed subsets of the hybrid state space X and d_H denote the Hausdorff metric. It follows that $(\mathcal{K}, d_H)$ is also a complete separable metric space and the open subsets corresponding to d_H generate the Borel σ-algebra $\mathcal{B}(\mathcal{K})$.

To account for set uncertainty we introduce the random closed set model. This model was introduced for applications in which the shape of the uncertain set is important in analyzing the properties of the set, for example in tumor characterization through medical imaging [136]. Please refer to [136, 137] for further details.

Definition 4.1. A random closed set is a measurable function $\Xi : \Omega \to \mathcal{K}$ from a probability space $(\Omega, \mathcal{F}, P)$ into the measure space $(\mathcal{K}, \mathcal{B}(\mathcal{K}))$.

The distribution of a random closed set Ξ is specified by $P\{\omega \mid F \cap \Xi(\omega) \neq \emptyset\}$, $\forall F \in \mathcal{K}$. For $F = \{x\} \in X$, the probability $P\{\omega \mid x \cap \Xi(\omega) \neq \emptyset\}$ is equivalent to $P\{x \in \Xi\} := P\{\omega \mid x \in \Xi(\omega)\}$. We refer to $p_\Xi(x) := P\{\omega \mid x \in \Xi(\omega)\}$ as the covering function. It follows that, $p_{X\setminus\Xi}(x) = 1 - p_\Xi(x)$.

It has been shown that the set of random closed sets has several desirable properties, for example, closure with respect to certain set transformations, such as convexification, dilation and erosion [136]. However, in general, the characterization of a random closed set and the computation of associated functions, such as the covering function, are difficult due to the size of $\mathcal{K}$. As such, methods have been suggested in the literature that alleviate these complexities [136, 138]. For example, random closed sets are often characterized by families of closed subsets of $\mathcal{K}$ which are parametrized [137]. For computational purposes, in the remainder of this section, we consider parameterized random closed sets and their associated covering functions.

In our work, we define a parameterized stochastic set-valued process as follows: Let $Y \subset \mathbb{R}^o$ for $o \in \mathbb{N}$ denote a parameter space. For $k = 0, 1, 2, \ldots, N$, let ζ_k be a Borel-measurable stochastic kernel on Y given Y, $\zeta_k : \mathcal{B}(Y) \times Y \to [0, 1]$, which assigns to each $y \in Y$ a probability measure $\zeta_k(\cdot|y)$ on the Borel space $(Y, \mathcal{B}(Y))$. Then, a discrete-time time-inhomogeneous Markov process $\{y_k, k = 0, 1, \ldots, N\}$, $y_k \in Y$ can be described by the stochastic kernels ζ_k.

Definition 4.2. A parameterized stochastic set process is a Borel measurable function $\phi : Y \to \mathcal{K}$ together with a Markov process $\{y_k, k = 0, 1, \ldots, N\}$ in the parameter space Y, with transition probability functions $\zeta_k : \mathcal{B}(Y) \times Y \to [0, 1]$, for $k = 0, 1, \ldots, N - 1$.

As in Chapter 3, let A be a compact Borel set representing the control space and define the controlled transition probability function $\tau : \mathcal{B}(X) \times X \times A \to [0, 1]$, that is, $\tau(\cdot|x, a)$ assigns a probability measure on $\mathcal{B}(X)$ for each $x \in X$, $a \in A$. Note that in this chapter, for notational simplicity, we do not consider presence of an adversary. Let $\bar{x} = (x, y)$ be the augmented state in $\bar{X} = X \times Y$, the augmented state space. Further, let us define the stochastic stochastic kernels $\bar{\tau}_k : \mathcal{B}(\bar{X}) \times \bar{X} \times A \to [0, 1]$ as

$$\bar{\tau}_k(d\bar{x}'|\bar{x}, a) = \tau(dx'|x, a)\zeta_k(dy'|y). \tag{4.1}$$

We call the resulting stochastic process an Augmented Discrete-Time Stochastic Hybrid System (ADTSHS) $\bar{\mathcal{H}}$.

Note that the stochastic kernels associated with the stochastic set parameters are independent of the control. On the other hand, the control may, in general, depend on the stochastic set parameters. For this general case, we define a Markov policy as follows:

Definition 4.3. A Markov policy for $\bar{\mathcal{H}}$ is a sequence $\mu = (\mu_0, \mu_1, ..., \mu_{N-1})$ of universally measurable maps $\mu_k : \bar{X} \to A$, $k = 0, 1, ..., N - 1$. The set of all admissible Markov policies is denoted by $\bar{\mathcal{M}}$.

Remark: In the previous chapter, we defined policies as Borel measurable maps. The Borel measurability property, together with the assumption on the continuity of the stochastic kernels in the input spaces, were sufficient conditions for proving existence of the optimal policies and strategies for the control and the adversary, respectively. In the single player case considered here, we enlarge the set of controls to universally measurable maps and relax the continuity assumptions on the stochastic kernels. While we can still prove the dynamic programming result, we only guarantee existence of ϵ suboptimal policies. Additionally, due to these alternative assumptions, we present the proof of the main theorem, Theorem 4.1, with a different method from that of Theorem 3.1, which is instructive.

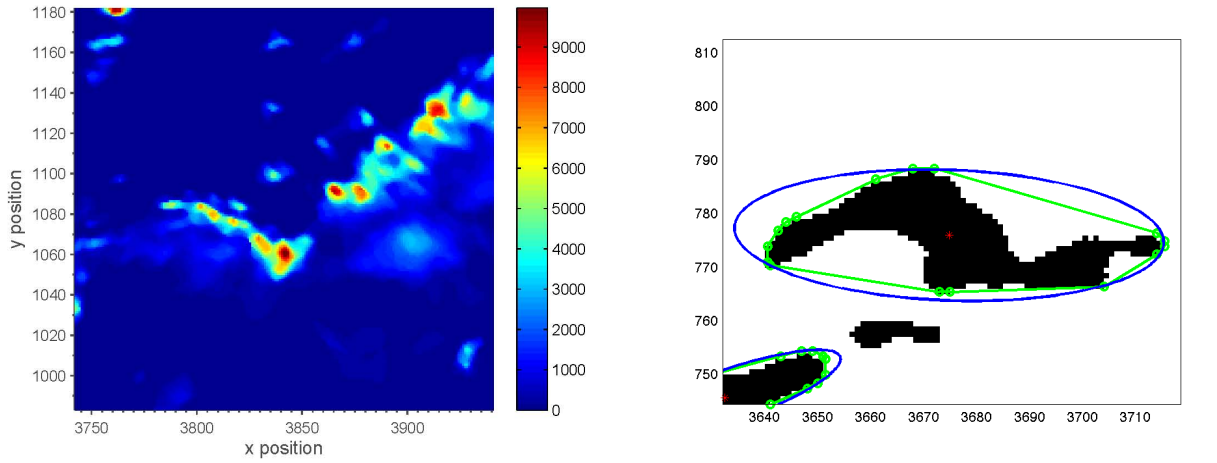
Given a Markov policy $\mu \in \bar{\mathcal{M}}$ and initial state $(x_0, y_0) \in X \times Y$, the execution of the augmented process denoted by $\{(x_k, y_k), k = 0, 1, \ldots, N\}$ is a stochastic process defined on the canonical sample space $\Omega := \bar{X}^{N+1}$, endowed with its product $\sigma-$algebra $\mathcal{B}(\Omega)$. The probability measure $P^{\mu}_{(x_0,y_0)}$ on Ω is uniquely defined by the stochastic kernels $\bar{\tau}_k$, the Markov policy $\mu \in \bar{\mathcal{M}}$, and the initial state $(x_0, y_0) \in \bar{X}$ [112].

In the following example, we illustrate the terminology developed and the application of the stochastic set-valued process with the problem of safe aircraft trajectory planning using the uncertain hazardous weather forecast data.

Example 4.1 (Stochastic forecast for aircraft trajectory planning)**.** In aircraft trajectory planning, the ability to identify and characterize regions of hazardous weather is vitally important. One factor to determine the safety of a region of the airspace for an aircraft to fly through is the Vertically Integrated Liquid (VIL) water content measurement [139, 140] which represents the level of precipitation in a column of the airspace. This measurement has proven useful in the detection of severe storms and short-term rainfall forecasting [140], and hence can be used as an indicator for establishing a no-fly zone for aircraft. The Corridor Integrated Weather System (CIWS) product [141] provides actual and forecast VIL numbers for a 1 km by 1 km grid of the United States airspace and with a 5-minute resolution in time. These measurements can be quantized into 6 levels, with levels 3 and higher indicating a recommended no-fly zone. The forecast horizon is 2 hours and the forecast is updated at 5 minute intervals. Figure 4.1(a) shows the VIL measurements for a 300 km $\times$200 km grid.

Clearly, there is uncertainty associated with the forecast data, and the uncertainty increases with the forecast horizon. One of the early works that accounted for uncertainty in no-fly zones due to hazardous weather was that of [142]. Although a stochastic Markov chain model was proposed to describe evolution of the storms, this model was not verified with any real forecast data.

To account for the uncertainty in CIWS forecast data, one can introduce a random set Ξ, for a no-fly zone and describe the covering function $p_{\Xi}(x)$, for every x in the airspace region of interest. This will then indicate the probability of hazard at every location of interest. However, the CIWS forecast data is deterministic and the covering function is not available. Thus, we propose a method for describing the stochastic no-fly zones through parametrizing



(a) VIL measurements

(b) Enclosing no-fly zones by geometric shapes

Figure 4.1: Hazardous weather regions from CIWS forecast product

the regions with high VIL levels as follows. First, for a given deterministic forecast, we enclose the regions with VIL values greater than the safety threshold with a minimum-volume bounding polygon or ellipsoid [102]. The uncertainty in the no-fly zone can then be associated with the randomness in the parameters of the polygon or the ellipsoid. Figure 4.1(b) shows minimum-volume bounding ellipses and polygons for regions with high VIL levels in 2D. In Figure 4.2(a) we compare the ellipses enclosing the no-fly zones extracted from the forecast and actual VIL measurements, for every 5 minutes in a 15-minute horizon.

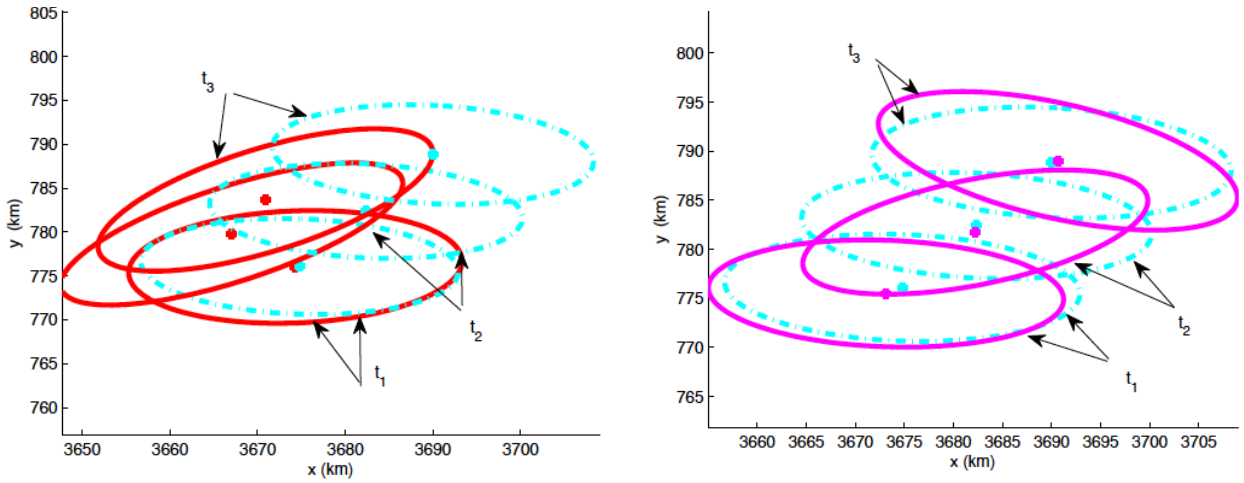
For simplicity in representation, here we focus on minimum-volume bounding ellipsoidal enclosure of the no-fly zones. Each ellipse $\mathcal{E}$ is parameterized by its center $m \in \mathbb{R}^2$ and its positive definite eccentricity matrix $M \in \mathbb{R}^{n\times n}$:

$$x \in \mathcal{E}(m, M) \iff (x-m)^T M(x-m) \leq 1.$$

For a given ellipse, let $m(t)$ and $m(t+\delta t)$ be the center of the ellipse at time t and $t+\delta t$ respectively, where $\delta t = 5$ minutes indicates the forecast time resolution. Define $b_t \in \mathbb{R}^2$ as $b_t = \frac{m(t+\delta t)-m(t)}{\delta t}$ to capture the incremental motion of the center per minute. To model the uncertainty in the forecast, we assume that the true center is a random variable c_k with dynamics:

$$c_{k+1} = c_k + b_k + \eta_k, \tag{4.2}$$

where $\eta_k \sim N(0, \Sigma_\eta)$, $\Sigma_\eta \in R^{2\times 2}$ is the covariance matrix of the noise associated with the forecast, $b_k = b_t$ for $k = t, t+\Delta t, \ldots, t+\delta t$, $\Delta t = 1$ minute indicates the discretization time for trajectory planning, and $c_t = m_t$ since we have the true weather data at time t.



(a) Forecast (dashed lines) and actual

(b) Forecast and an execution of obstacle process

Figure 4.2: Trajectory of the no-fly zones

Let $M(t)$ and $M(t+\delta t)$ be the eccentricities obtained from the forecast at times t and $t+\delta t$ respectively. We model the randomness in the eccentricity through introducing C_k:

$$C_k = R(\theta_k)\big((1-\frac{k-t}{\delta t})M(t) + \frac{k-t}{\delta t}M(t+\delta t)\big)R(\theta_k)^T, \tag{4.3}$$

where R is a rotation matrix, and the angle of rotation, θ_k, is zero for $k=t$ and is a random variable with uniform distribution over an interval $[-a,a]$ for $k=t+\Delta t,\ldots,t+\delta t$.

The random ellipse at time k is hence given by $\mathcal{E}(c_k,C_k)$. The stochastic parameters Σ_η and a are determined from the comparison of forecast and actual weather data as well as the rate of movement of the storms. For the data associated with Figure 4.2(a) these parameters were set to $\Sigma_\eta = I_{2\times 2}$ and $a=\frac{\pi}{6}$. A realization of the random ellipses for a 15 horizon is compared with the ellipses obtained from the forecast data in Figure 4.2(b).

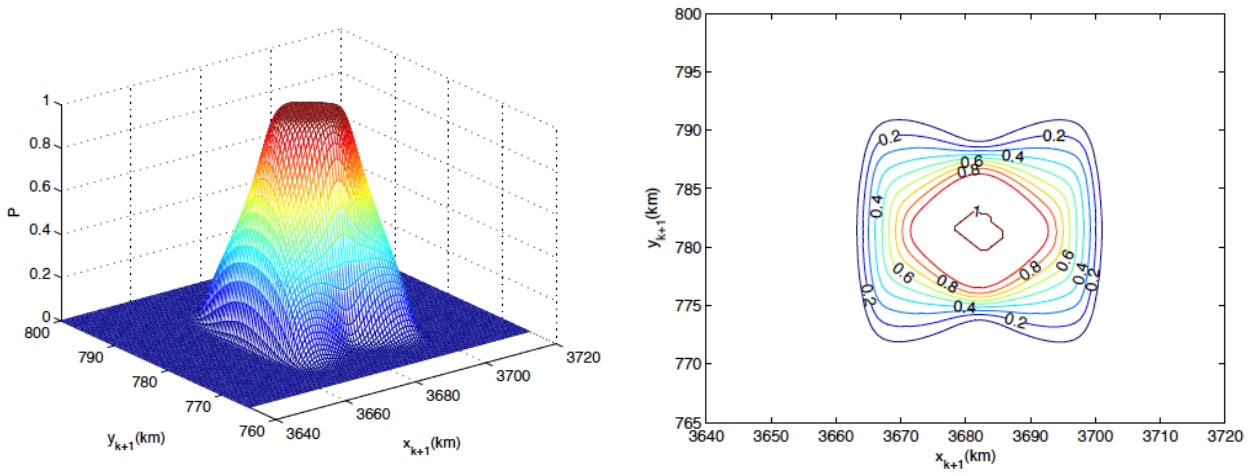
The no-fly zone stochastic set process is parameterized by $c_k \in \mathbb{R}^2$, $C_k \in \mathbb{R}^{2\times 2}$, together with the set-valued map $\phi : \mathbb{R}^2 \times \mathbb{R}^{n\times n} \to \mathcal{K}$, with $\mathcal{K}$ being the set of closed subsets of $\mathbb{R}^2$:

$$\phi(c_k,C_k) := \{x \in \mathbb{R}^2 \mid (x-[c^1,c^2])^T C_k (x-[c^1,c^2]) \le 1\}.$$

For an aircraft position, $x_k \in \mathbb{R}^2$, the probability of being in the hazardous region Ξ_k is

$$P_{\Xi_k}(x_k) = P(\omega \mid (x_k-[c_k^1(\omega),c_k^2(\omega)])^T C_k(\omega)(c_k^3(\omega))(x_k-[c_k^1(\omega),c_k^2(\omega)]) \le 1).$$

Unfortunately, an analytic formula for computing the above covering function does not exist. If the eccentricity of the ellipse is assumed to be deterministic, the above probability



(a) Probability distribution of the random set

(b) Contours of the probability distribution

Figure 4.3: Monte Carlo simulation of the covering function

obeys a Chi-squared distribution and can be approximated using statistical computational tools. For the more general case, one can use Monte Carlo simulations to approximate the probability. For the forecast data with $m_k = (3675 \text{ km}, 775 \text{ km})$, $b_k = (7.1 \text{ km}, 6.4 \text{ km})$, and $M_k = (0.0028, 0; 0, 0.0278)$, the covering function $p_{\Xi_{k+1}}(x_{k+1})$ at time $k+1$ is approximated over the region $[3640, 3720] \times [765, 800]$ using 10^4 Monte Carlo samples and 101×101 grid discretization. The results are displayed in Figure 4.3.

Reach-Avoid with Stochastic Set Processes

Consider the stochastic kernels $\zeta_k : \mathcal{B}(Y) \times Y \to [0, 1]$ and the parameter process $\{y_k\}$, for $k = 0, 1, \ldots, N$ distributed according to these kernels, along with two Borel measurable functions $\phi : Y \to \mathcal{K}$ and $\phi' : Y \to \mathcal{K}$, such that $\phi(y) \subseteq \phi'(y)$, $\forall y \in Y$. Define $K_k := \phi(y_k)$ and $K'_k := \phi'(y_k)$ as stochastic target and safe sets, respectively. We assume that the initial parameter state y_0, and hybrid state x_0, are known. Our goal is to evaluate and subsequently maximize the probability that the execution of the Markov control process $\{x_k\}$, for $k = 0, 1, \ldots, N$ will reach the target set at some time in the horizon while remaining safe at all prior times.

The probability that the system initialized at $x_0 \in X$, $y_0 \in Y$, with control policy $\mu \in \bar{\mathcal{M}}$ reaches K_k while avoiding $X \setminus K'_k$ using our mathematical notation is:

$$r^{\mu}_{(x_0,y_0)}(\phi, \phi') := P^{\mu}_{(x_0,y_0)}\{\exists j \in [0, N] \mid x_j \in \phi(y_j) \wedge \forall i \in [0, j-1] \;\; x_i \in \phi'(y_i) \setminus \phi(y_i)\}. \quad (4.4)$$

Note that while we assume knowledge of the initial state and parameters of the sets, the consideration of a probabilistic initial condition for each is straightforward.

Let $\bar{K} = \{(x, y) \in X \times Y \mid x \in \phi(y)\}$ and $\bar{K}' = \{(x, y) \in X \times Y \mid x \in \phi'(y)\}$. From Borel measurability of ϕ and ϕ', it follows that $\bar{K}$ and $\bar{K}'$ are graphs of Borel measurable functions and are thus Borel measurable [143]. The reach-avoid probability in Equation (4.4) can be characterized based on the probability measure on $\Omega = \bar{X}^{N+1}$, that is $P_{(x_0,y_0)}$, as

$$\begin{aligned} r^{\mu}_{(x_0,y_0)}(\bar{K}, \bar{K}') &= P^{\mu}_{(x_0,y_0)}\left(\bigcup_{j=0}^{N} (\bar{K}' \setminus \bar{K})^j \times \bar{K} \times X^{N-j}\right) \\ &= \sum_{j=0}^{N} P^{\mu}_{(x_0,y_0)}\big((\bar{K}' \setminus \bar{K})^j \times \bar{K} \times X^{N-j}\big), \end{aligned}$$

where the second equality follows by the fact that the union is disjoint. Let $\bar{x}_k = (x_k, y_k)$. By definition of expectation on this probability measure, $r^{\mu}_{(x_0,y_0)}(\bar{K}, \bar{K}')$ can be computed as

$$r^{\mu}_{(x_0,y_0)}(\bar{K}, \bar{K}') = E^{\mu}_{(x_0,y_0)}\left[\mathbf{1}_{\bar{K}}(\bar{x}_0) + \sum_{j=1}^{N}\left(\prod_{i=0}^{j-1} \mathbf{1}_{\bar{K}' \setminus \bar{K}}(\bar{x}_i)\right)\mathbf{1}_{\bar{K}}(\bar{x}_j)\right]. \tag{4.5}$$

In the above, $E^{\mu}_{(x_0,y_0)}$ denotes the expectation with respect to the probability measure $P^{\mu}_{(x_0,y_0)}$.

Our objective is to design an efficient algorithm for evaluating the reach-avoid probability $r^{\mu}_{(x_0,y_0)}(\bar{K}, \bar{K}')$ for a given Markov policy and for optimizing this probability over the set of Markov policies. The precise problem statement is as follows:

Problem 4.1. Given an ADTSHS $\bar{\mathcal{H}}$, with stochastic set parameters $y \in Y$, and set-valued maps ϕ and ϕ', $\phi'(y) \subseteq \phi(y)$ for all $y \in Y$, representing the target and safe sets respectively:

(a) Compute the optimal value function $r^{*}_{\bar{x}_0}(\bar{K}, \bar{K}') := \sup_{\mu \in \bar{\mathcal{M}}} r^{\mu}_{\bar{x}_0}(\bar{K}, \bar{K}')$, $\forall \bar{x}_0 \in \bar{X}$;

(b) Find an optimal Markov policy $\mu^* \in \bar{\mathcal{M}}$, if it exists, such that $r^{*}_{\bar{x}_0}(\bar{K}, \bar{K}') = r^{\mu^*}_{\bar{x}_0}(\bar{K}, \bar{K}')$, $\forall \bar{x}_0 \in \bar{X}$.

From Equation (4.5), we observe that the probabilistic reach-avoid problem with stochastic sets is transformed into a probabilistic reach-avoid problem with deterministic sets in an extended state space. Hence, reach-avoid methods for deterministic safe and target sets as described in the previous chapter for two players and in [47] for a single player can be applied to the problem at hand. As discussed in the previous chapter, this method becomes computationally intractable for any hybrid and parameter spaces with combined dimensions above 4 or 5, due to the Curse of Dimensionality [29]. In the remainder of this section, we introduce an approximation which greatly reduces the computational burden and has also been successfully applied in our work [56].

Tractable Solution of the Reach-Avoid Problem

To simplify evaluation and maximization of $r^{\mu}_{(x_0,y_0)}(\bar{K},\bar{K}')$ we make the following assumption:

Assumption 4.1. The Markov parameters describing the stochastic sets are given as or can be fairly approximated by independent time-varying probability distributions. That is, $\zeta_{k+1}(dy_{k+1}|y_k) = \zeta_{k+1}(dy_{k+1})$, $k = 0, 1, \ldots, N$.

Recall that the Markov policy for the ADTSHS $\bar{\mathcal{H}}$, was given in Definition 4.3 as a sequence of maps $\mu_k : X \times Y \to A$. Due to the independence of the probability distribution ζ_{k+1} from the parameter y_k, without loss of generality, we consider the Markov policy also being independent from the parameter y_k. Thus, we define the Markov policy as sequence of universally measurable maps $\mu_k : X \to A$, $k = 0, 1 \ldots, N$. Let $\mathcal{M}$ denote the set of all such policies. Note that due to this assumption, the closed loop stochastic kernels $\bar{\tau}_k(.|\bar{x_k}, \mu_k(\bar{x_k})$ become equivalent to product of two decoupled stochastic kernels $\tau(.|x_k, \mu_k(x_k))$ and $\zeta_k(y_k)$.

Based on the set-valued maps ϕ and ϕ', representing the target and safe sets, we define the following covering functions:

$$p_{K_k}(x) = \int_Y \mathbf{1}_{\bar{K}}(x, y_k)\zeta_k(dy_k) = \int_Y \mathbf{1}_{\phi(y_k)}(x)\zeta_k(dy_k) = E\left[\mathbf{1}_{\phi(y_k)}(x)\right], \tag{4.6}$$
$$p_{K'_k}(x) = \int_Y \mathbf{1}_{\bar{K}'}(x, y_k)\zeta_k(dy_k) = \int_Y \mathbf{1}_{\phi'(y_k)}(x)\zeta_k(dy_k) = E\left[\mathbf{1}_{\phi'(y_k)}(x)\right].$$

Since $\phi(y) \subseteq \phi'(y)$, $\forall y \in Y$, we get $p_{K'_k \setminus K_k}(x) = p_{K'_k}(x) - p_{K_k}(x)$. In addition, since $\bar{K}$ and $\bar{K}'$ are Borel measurable sets and $\mathbf{1}_{\bar{K}}$, $\mathbf{1}_{\bar{K}'}$ are indicator functions of Borel measurable sets, they are Borel measurable. From Borel measurability of ζ_k and Proposition 7.29 of [112], it follows that the covering functions p_{K_k} and $p_{K'_k}$, are also Borel measurable.

The main consequence of the independent assumption is an equivalent expression of the reach-avoid probability based on the above covering functions as stated below.

Lemma 4.1. *For an ADTSHS $\bar{\mathcal{H}}$ with independent stochastic set process, the reach-avoid probability in Equation* (4.5) *can be expressed as*

$$r^{\mu}_{(x_0,y_0)}(\bar{K},\bar{K}') = E^{\mu}_{x_0}\left[\sum_{j=0}^{N}\left(\prod_{i=0}^{j-1} p_{K'_i \setminus K_i}(x_i)\right) p_{K_j}(x_j)\right], \tag{4.7}$$

where $E^{\mu}_{x_0}$ is the expectation with respect to the canonical probability measure on X^{N+1}, and we use the convention that $\prod_{i=k}^{j}(.) = 1$ for $k > j$. .

Proof. A proof by Fubini's Theorem [119] is as follows:

$$
\begin{aligned}
r^{\mu}_{(x_0,y_0)}(\bar{K},\bar{K}') &= E^{\mu}_{(x_0,y_0)}\left[\sum_{j=0}^{N}\left(\prod_{i=0}^{j-1}\mathbf{1}_{\bar{K}'\setminus\bar{K}}(\bar{x}_i)\right)\mathbf{1}_{\bar{K}}(\bar{x}_j)\right] \\
&= \int_{X^N}\int_{Y^{N+1}}\left[\sum_{j=0}^{N}\left(\prod_{i=0}^{j-1}\mathbf{1}_{\phi'(y_i)\setminus\phi(y_i)}(x_i)\right)\mathbf{1}_{\phi(y_j)}(x_j)\right]\prod_{j=0}^{N}\zeta_j(dy_j)\prod_{j=0}^{N-1}\tau^{\mu_j}(dx_{j+1}|x_j) \\
&= E^{\mu}_{x_0}\left[\sum_{j=0}^{N}\left(\int_{Y^j}\prod_{i=0}^{j-1}(\mathbf{1}_{\phi'(y_i)}(x_i)-\mathbf{1}_{\phi(y_i)}(x_i))\mathbf{1}_{\phi(y_j)}(x_j)\prod_{i=0}^{j}\zeta_i(dy_i)\right)\right] \\
&= E^{\mu}_{x_0}\left[\sum_{j=0}^{N}\left(\prod_{i=0}^{j-1}\int_{Y}((\mathbf{1}_{\phi'(y_i)}(x_i)-\mathbf{1}_{\phi(y_i)}(x_i))\zeta_i(dy_i))\right)\int_{Y}\mathbf{1}_{\phi(y_j)}(x_j)\zeta_j(dy_j)\right] \\
&= E^{\mu}_{x_0}\left[\sum_{j=0}^{N}\left(\prod_{i=0}^{j-1}p_{K_i'\setminus K_i}(x_i)\right)p_{K_j}(x_j)\right].
\end{aligned}
$$

□

Now, we derive a recursion for computing the reach-avoid probability expressed in Equation (4.7) for a given Markov policy $\mu \in \mathcal{M}$. Let us define the functions $V^{\mu}_k : X \to [0,1]$, $k = 0,\ldots,N$, with a backwards recursion as

$$
\begin{aligned}
&V^{\mu}_N(x) = p_{K_N}(x), \\
&V^{\mu}_k(x) = p_{K_k}(x) + \\
&\qquad p_{K_k'\setminus K_k}(x)\int_{X^{N-k}}\sum_{j=k+1}^{N}\left(\prod_{i=k+1}^{j-1}p_{K_i'\setminus K_i}(x_i)\right)p_{K_j}(x_j)\prod_{j=k+1}^{N-1}\tau^{\mu_j}(dx_{j+1}|x_j)\tau^{\mu_k}(dx_{k+1}|x).
\end{aligned}
$$

From the definition above and Equation (4.7), it is clear that $V^{\mu}_0(x_0) = r^{\mu}_{(x_0,y_0)}(\bar{K},\bar{K}')$, $\forall x_0 \in X$.

Let $\mathcal{F}$ denote the set of universally measurable functions from X to $\mathbb{R}$. Define the operator $H : X \times A \times \mathcal{F} \to \mathbb{R}$ as

$$
H(x,a,J) := \int_X J(y)\tau(dy|x,a). \tag{4.8}
$$

The following lemma shows that $V^{\mu}_k(x)$ can be computed via a backwards recursion.

Lemma 4.2. *Fix a Markov policy $\mu = (\mu_0,\mu_1,\ldots\mu_{N-1}) \in \mathcal{M}$. The functions $V^{\mu}_k : X \to [0,1]$, $k = 0,1,\ldots,N-1$ can be computed by the backward recursion:*

$$
V^{\mu}_k(x) = p_{K_k}(x) + p_{K_k'\setminus K_k}(x)H(x,\mu_k(x),V^{\mu}_{k+1}), \tag{4.9}
$$

initialized with $V^{\mu}_N(x) = p_{K_N}(x)$, $x \in X$.

Proof. The proof is by induction. First, due to the definition of V_k^μ we have that

$$V_{N-1}^\mu(x) = p_{K_{N-1}}(x) + p_{K'_{N-1}\setminus K_{N-1}}(x) \int_X V_N^\mu(x_N)\tau^{\mu_{N-1}}(dx_N|x),$$

so that (4.9) is proven for $k = N-1$. For $k < N-1$ we can separate the terms associated with x_{k+1} from those associated with $k+2, \ldots, N$ as follows:

$$\begin{aligned} V_k^\mu(x) =& p_{K_k}(x) + \\ & p_{K'_k\setminus K_k}(x) \int_X p_{K_{k+1}}(x_{k+1}) + p_{K'_{k+1}\setminus K_{k+1}}(x_{k+1}) \left(\int_{X^{N-k-1}} \sum_{j=k+2}^{N} \prod_{i=k+2}^{j-1} p_{K'_i\setminus K_i}(x_i) \right. \\ & \left. p_{K_j}(x_j) \prod_{j=k+2}^{N-1} \tau^{\mu_j}(dx_{j+1}|x_j)\tau^{\mu_{k+1}}(dx_{k+2}|x_{k+1}) \right) \tau^{\mu_k}(dx_{k+1}|x) \\ =& p_{K_k}(x) + p_{K'_k\setminus K_k}(x) \int_X V_{k+1}^\mu(x_{k+1})\tau^{\mu_k}(dx_{k+1}|x), \end{aligned}$$

which concludes the proof. □

The above two lemmas show that given a Markov policy $\mu \in \mathcal{M}$, we can evaluate the reach-avoid probability with a recursion which scales exponentially with respect to only the hybrid state space and is independent of dimensions of the stochastic sets' parameters. In the following theorem, we prove that the same conclusion can be drawn for maximizing the reach-avoid probability. In addition, in the process of maximizing the probability, the optimal Markov policy can be synthesized.

Theorem 4.1. *Let $\bar{\mathcal{H}}$ be an ADTSHG with independent stochastic set process. Define $V_N^*(x) = p_{K_N}(x)$ and $V_k^* : X \to [0,1]$, $k = 0, 1, ..., N-1$, by the backward recursion*

$$V_k^*(x) = \sup_{a\in A}\{p_{K_k}(x) + p_{K'_k\setminus K_k}(x)H(x, a, V_{k+1}^*)\}, \tag{4.10}$$

(a) The optimal reach-avoid probability is given as $r_{\bar{x}_0}^*(\bar{K}, \bar{K}') = V_0^*(x_0), \forall x_0 \in X$;

(b) If $\mu^* \in \mathcal{M}$ is a Markov policy which satisfies

$$\mu_k^*(x) \in \arg\sup_{a\in A} H(x, a, V_{k+1}^*), \tag{4.11}$$

$\forall x \in X$, for $k = 0, 1, \ldots, N-1$, then μ^* is an optimal Markov policy.

The proof follows the results used in Proposition 8.2 of [112] for discrete-time stochastic optimal control problems. The main steps are as follows: First, we show that the iterative procedure in Equation (4.9) can be written with a dynamic programming operator $T_{k,\pi}$:

$\mathcal{F} \to \mathcal{F}$, where $\pi : X \to A$ is a universally measurable map. Since this operator preserves measurability properties (Lemma 4.3), the sequential composition of $T_{k,\pi}$ is well-defined. From the fact that the operator $T_k : \mathcal{F} \to \mathcal{F}$, formed after optimizing $T_{k,\pi}$ with respect to policies, preserves lower semi-analyticity of functions (Lemma 4.5), existence of ϵ suboptimal policies follows from Proposition 7.50 of [112]. Finally, the proof is completed by showing that T_k preserves monotone property (Lemma 4.4).

To facilitate application of the theoretical results in [112] we first define an equivalent minimization problem. Given a Markov policy $\mu \in \mathcal{M}$, define the value function $J_k^\mu := -V_{N-k}^\mu$, $k = 0, 1, \ldots, N$. Further, for notational convenience, define the functions $p_k(x) = p_{K_{N-k}}(x)$ and $p'_k(x) = p_{K'_{N-k} \setminus K_{N-k}}(x)$ for $x \in X$ and $k = 0, 1, \ldots, N$. It follows that J_k^μ may be written with a forward recursion as:

$$J_k^\mu(x) = -p_k(x) + p'_k(x) H(x, \mu_{N-k}(x), J_{k-1}^\mu), \tag{4.12}$$

initialized with $J_0^\mu(x) = -p_0(x)$, $x \in X$. Consequently, from Lemma 4.1

$$J_N^\mu(x_0) = E_{x_0}^\mu \left[-\sum_{j=0}^{N} \left(\prod_{i=0}^{j-1} p'_{N-i}(x_i) \right) p_{N-j}(x_j) \right] = -r_{(x_0,y_0)}^\mu(\bar{K}, \bar{K}').$$

We next define optimal one-step cost and input as

$$\begin{aligned} J_k^*(x) &= \inf_{a \in A} \{-p_k(x) + p'_k(x) H(x, a, J_{k-1}^*)\}, & (4.13) \\ \mu_k^*(x) &= \arg\inf_{a \in A} \{-p_{N-k}(x) + p'_{N-k}(x) H(x, a, J_{N-k-1}^*)\}, & (4.14) \end{aligned}$$

whenever the infimum is attained, with $J_0^*(x) = -p_0(x)$, $x \in X$. Next, define the function $G_k : X \times A \times \mathcal{F} \to \mathbb{R}$, and the map $T_{k,\pi} : \mathcal{F} \to \mathcal{F}$ for $\pi : X \to A$ as

$$\begin{aligned} G_k(x, a, J) &= -p_k(x) + p'_k(x) H(x, a, J), \\ T_{k,\pi}[J](x) &= G_k(x, \pi(x), J). \end{aligned}$$

Lemma 4.3. *The map $T_{k,\pi}$ preserves universal measurability, that is, if $J \in \mathcal{F}$, then, $T_{k,\pi}[J] \in \mathcal{F}$.*

Proof. Due to Borel measurability of the stochastic kernel τ and since $J \in \mathcal{F}$, we can apply Proposition 7.46 of [112] to conclude that $H(x, a, J)$ is universally measurable. As discussed before, the covering functions p_{K_k} and $p_{K'_k}$, are Borel measurable and thus p_k and p'_k are also Borel measurable for $k = 0, 1, \ldots, N$. The result follows since universal measurability property is preserved through multiplication and addition of Borel measurable functions. □

Given that the recursion (4.12) can be rewritten as $J_k^\mu = T_{k,\mu_{N-k}}[J_{k-1}^\mu]$ and $J_0^\mu \in \mathcal{F}$, by Lemma 4.3 we conclude $J_k^\mu \in \mathcal{F}$, for $k = 1, 2, ..., N$.

Lemma 4.4. *The map $T_{k,\pi}$ satisfies the following properties:*

(a) $\forall J, J' \in \mathcal{F}$ if $J(x) \leq J'(x)$, $\forall x \in X$, then $T_{k,\pi}[J](x) \leq T_{k,\pi}[J'](x)$, $\forall x \in X$, $k \in \mathbb{N}$.
(b) For any $J \in \mathcal{F}$, $x \in X$, $k \in \mathbb{N}$, and any real number $r > 0$,

$$T_{k,\pi}[J](x) \leq T_{k,\pi}[J + r](x) \leq T_{k,\pi}[J](x) + r. \tag{4.15}$$

Proof. Part (a) immediately follows from the definition of $T_{k,\pi}$. For Part (b), note that

$$\begin{aligned} G_k(x, a, J + r) &= -p_k(x) + p'_k(x)H(x, a, J + r) \\ &= -p_k(x) + p'_k(x)H(x, a, J) + rp'_k(x) \int_X \tau(dy|x, a). \end{aligned}$$

Since $p'_k(x) \int_X \tau(dy|x, a) = p'_k(x)$ is bounded between 0 and 1, we conclude that

$$G_k(x, a, J) \leq G_k(x, a, J + r) \leq G_k(x, a, J) + r,$$

for $x \in X$, $k \in \mathbb{N}$, $a \in A$. Thus, the result for $T_{k,\pi}$ then follows. □

We now define the map $T_k : \mathcal{F} \to \mathcal{F}$ as $T_k[J](x) = \inf_{a \in A} G_k(x, a, J)$, $x \in X$, $k \in \mathbb{N}$. The recursion in (4.13) can be re-expressed as $J^*_k = T_k[J^*_{k-1}]$. It follows that $J^*_k = T^k[J^*_0]$, $k = 0, 1, \ldots, N$, where $T^0[J] = J$ and $T^k[J] = T_k \circ T^{k-1}[J]$.

Let $\mathcal{F}^* \subset \mathcal{F}$ denote the set of lower semi-analytic functions.

Lemma 4.5. *The map T_k preserves the lower semi-analyticity, that is, if $J \in \mathcal{F}^*$, then $T_k[J] \in \mathcal{F}^*$.*

Proof. From Propositions 7.48 in [112], for any $k = 0, 1, \ldots, N$, $a \in A$ and $J \in \mathcal{F}^*$, $G_k(x, a, J)$ is lower semi-analytic. It follows from Proposition 7.47 in [112] that $T_k[J] = \inf_{a \in A} G_k(x, a, J)$ is lower semi-analytic as desired. □

Since $J^*_k = T_k[J^*_{k-1}]$ and $J^*_0 \in \mathcal{F}^*$, by the above lemma, we conclude that $J^*_k \in \mathcal{F}^*$, for $k = 1, 2, \ldots, N$. In addition, due to the lower semi-analytic property, we can apply Proposition 7.50 in [112] to show existence of ϵ suboptimal policies at every stage of the minimization in recursion (4.14). This property is used to show our first dynamic programming result:

Proposition 4.1. *The optimal cost $J^*_N(x)$, can be defined in terms of the operator T^N:*

$$\inf_{\mu} E^{\mu}_x \left[-\sum_{j=0}^{N} \left(\prod_{i=0}^{j-1} p'_{N-i}(x_i) \right) p_{N-j}(x_j) \right] = T^N[J^*_0](x).$$

Proof. We show that $T^N[J_0^*](x)$ simultaneously upper bounds and lower bounds the optimal cost. Due to lower semi-analyticity of $G_k(x, a, J_k^*)$ for $k = 0, 1, \ldots, N$, Proposition 7.50 in [112] implies that for any $\epsilon > 0$ and $k = 0, 1, \ldots, N$, there exists a universally measurable function $\pi_{N-k}^\epsilon : X \to A$ such that $G_k(x, \pi_{N-k}^\epsilon, J_{k-1}^*) \leq \inf_{a \in A} G_k(x, a, J_{k-1}^*) + \frac{\epsilon}{N}$. This in turn implies that $T_{k,\pi_{N-k}^\epsilon} \circ T^{k-1}[J_0^*] \leq T^k[J_0^*] + \frac{\epsilon}{N}$. If we consider $\mu^\epsilon = (\pi_0^\epsilon, \pi_1^\epsilon, \ldots, \pi_{N-1}^\epsilon)$, by Lemma 4.4, we obtain

$$\begin{aligned} J_N^{\mu^\epsilon} &= T_{N,\pi_0^\epsilon} \circ T_{N-1,\pi_1^\epsilon} \circ \cdots \circ T_{1,\pi_{N-1}^\epsilon}[J_0^*] \leq T_{N,\pi_0^\epsilon} \circ T_{N-1,\pi_1^\epsilon} \circ \cdots \circ T^1[J_0^*] + \frac{\epsilon}{N} \\ &\leq T_{N,\pi_0^\epsilon} \circ T_{N-1,\pi_1^\epsilon} \circ \cdots \circ T^2[J_0^*] + 2\frac{\epsilon}{N} \\ &\leq \ldots \leq T^N[J_0^*] + N\frac{\epsilon}{N}. \end{aligned}$$

We conclude that $T^N[J_0^*]$ upper bounds the infimum:

$$J_N^*(x) = \inf_\mu E_x^\mu \left[-\sum_{j=0}^{N} \left(\prod_{i=0}^{j-1} p'_{N-i}(x_i) \right) p_{N-j}(x_j) \right] \leq T^N[J_0^*](x).$$

On the other hand, for any $\mu \in \mathcal{M}$

$$J_N^\mu = T_{N,\mu_0} \cdots T_{1,\mu_{N-1}}[J_0^\mu] \geq T_{N,\mu_0} \cdots T_{2,\mu_{N-1}} T^1[J_0^\mu] \geq \ldots \geq T^N[J_0^\mu].$$

Taking the infimum over $\mu \in \mathcal{M}$, and since $J_0^\mu = J_0^*$ we obtain

$$T^N[J_0^*](x) \leq \inf_\mu E_x^\mu \left[-\sum_{j=0}^{N} \left(\prod_{i=0}^{j-1} p'_{N-i}(x_i) \right) p_{N-j}(x_j) \right].$$

Since T^N upper bounds and lower bounds the optimal cost, we get the desired result. □

Finally, with the lemmas and proposition above, we can prove Theorem 4.1.

Proof. It directly follows from the definition of the functions V_k^* (4.10), J_k^* (4.13), and Proposition 4.1 that the dynamic programming algorithm holds, that is, for any $x_0 \in X$

$$V_0^*(x_0) = -J_N^*(x_0) = -T^N[J_0^*](x_0) = \sup_{\mu \in \mathcal{M}} r^\mu_{(x_0,y_0)}(\bar{K}, \bar{K}').$$

Thus, Part (a) of the theorem is proven.

For part (b), we show that a control policy defined by $\mu^* = (\mu_0^*, \mu_1^*, \ldots, \mu_N^*)$ satisfying (4.14) is Markov and optimal. Suppose μ_k^* satisfies (4.14). Then, using the introduced notations:

$$T_{N-k,\mu_k^*}[J_{N-k-1}^*](x) = \inf_{a \in A} G_{N-k}(x, a, J_{N-k-1}^*) = J_{N-k}^*(x),$$

for $x \in X$. Since $G_{N-k}(x, a, J^*_{N-k})$ is lower-semianalytic by Lemma 4.5, if its infimum with respect to $a \in A$ is attained for $x \in X$, the resulting function $\mu^*_k : X \to A$, is universally measurable by part (b) of Proposition 7.50 in [112]. By Proposition 4.1, $\inf_\mu E^\mu_x \left[-\sum_{j=0}^{N} \left(\prod_{i=0}^{j-1} p'_{N-i}(x_i) \right) p_{N-j}(x_j) \right] = J^*_N(x)$, for all $x \in X$. Also,

$$\begin{aligned} J^*_N(x) =& T_{N,\mu^*_0}[J^*_{N-1}](x) \\ =& T_{N,\mu^*_0} \circ T_{N-1,\mu^*_1}[J^*_{N-2}](x) = \ldots = T_{N,\mu^*_0} \circ T_{N-1,\mu^*_1} \circ \cdots \circ T^*_{1,\mu_{N-1}}[J^*_0](x) \\ =& J^{\mu^*}_N(x) = E^{\mu^*}_x \left[-\sum_{j=0}^{N} \left(\prod_{i=0}^{j-1} p'_{N-i}(x_i) \right) p_{N-j}(x_j) \right]. \end{aligned}$$

Now, since $J^*_{N-k-1} = -V^*_{k+1}$, we have

$$\begin{aligned} G_{N-k}(x, a, J^*_{N-k-1}) =& - p_{N-k}(x) + p'_{N-k}(x) H(x, a, J^*_{N-k-1}) \\ =& - p_{N-k}(x) - p'_{N-k}(x) H(x, a, V^*_{k+1}). \end{aligned}$$

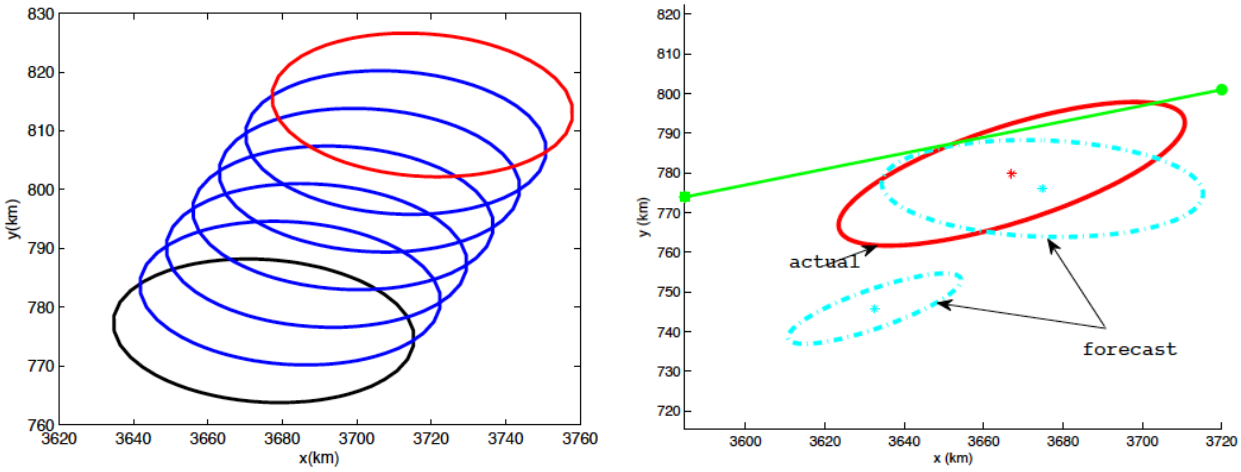
Thus, the optimizers in (4.14) are the same as those in (4.11). Consequently, $\mu^* = (\mu^*_0, \mu^*_1, \ldots, \mu^*_{N-1})$ is an optimal reach-avoid Markov policy. □

We end the theoretical development noting that although in the course of the proof, we are only able to guarantee existence of ϵ suboptimal policies, a sufficient condition for existence of an optimal policy μ^* is that the level sets $\{a \in A | H(x, a, V^*_{k+1}) \geq \lambda\}$ are compact $\forall x \in X$, $\lambda \in \mathbb{R}$, $k \in [0, N-1]$. In addition, it would be interesting to explore whether optimal randomized policies always exist.

Remark: While the computation of the reach-avoid probability with this result is independent of the dimension of the set parameters, in contrast to the state extension method, we now need to have a model of the stochastic sets over the time horizon *a priori* in order to compute $p_{K_k}(x)$ and $p_{K'_k}(x)$ in the backward recursion for all x in the hybrid state space.

4.2 Aircraft Trajectory Planning through Stochastic Hazardous Weather

We consider maximization of aircraft safety and reachability given hazardous weather forecast obtained for a section of airspace centered at latitude 30^o and longitude 86.5^o, near the gulf coast of Florida, on 01/07/2009, a day in which storms were observed in the region under consideration. We extracted a thirty minute forecast comprising VIL levels, resulting in no-fly zones represented as bounding ellipses, at 5 minute increments from the CIWS forecast data [141]. Figure 4.4(a) represents the forecast over a thirty minute period. It is interesting to note that the forecasts at time steps $5, 10, \ldots, 30$ appear to be translations of



(a) No-fly zones from 30 minute forecast

(b) Aircraft trajectory

Figure 4.4: Forecasted no-fly zones and an aircraft trajectory

the forecast at time step 0. Figure 4.4(b) shows an aircraft path over a 10 minute period which avoids the forecast but it intersects the hazardous region obtained from the actual weather data of the same day.

For the objective of trajectory planning, we model the aircraft as a unicycle with three modes of operation; straight flight, right turn, and left turn. Let (x^1, x^2) denote the aircraft position in $2D$, $x^3 \in [-\pi, \pi]$ denote its heading angle, s denote the speed, a denote the heading angle command and Δt denote the sampling time. The equations of motion of the aircraft are

$$\begin{aligned} x^1_{k+1} =& x^1_k + \Delta t s_k \cos(x^3_k) + \omega^1_k, \\ x^2_{k+1} =& x^2_k + \Delta t s_k \sin(x^3_k) + \omega^2_k, \\ x^1_{k+1} =& x^3_k + \Delta t a_k + \omega^3_k. \end{aligned} \tag{4.16}$$

In the above, $\omega = (\omega^1, \omega^2, \omega^3) \sim \mathcal{N}(0, \Sigma_\omega)$ denotes the stochastic disturbance due to presence of wind and actuator noise. The linear velocity of the aircraft is assumed to be constant and its angular velocity takes three values based on the mode of the system, that is, $a \in \{0, -\bar{a}, \bar{a}\}$, where $\bar{a} \in \mathbb{R}$ is the angular velocity of the aircraft when in turning mode.

In the following examples, we model the motion of the no-fly zones based on the stochastic ellipse representation described in Example 4.1. In the first case study, we augment the state of the aircraft with stochastic parameters of the ellipse and maximize the probability of safety of the aircraft trajectory. In the second case study, we model the stochastic ellipse motion as an independent stochastic set process. We maximize the probability of the aircraft attaining a target region while avoiding the uncertain unsafe locations.

For numerical results, we consider a sampling time of $\Delta t = 1$ minute, aircraft speed of $s_k = 7.1$ km per minute, angular speed of $\bar{a} = 0.3$ radians per minute, and disturbance variances defined by $\Sigma_\omega(1,1) = \Sigma_\omega(2,2) = 0.25$, $\Sigma_\omega(3,3) = 0.05$, and $\Sigma_\omega(i,j) = 0$ if $i \neq j$.

Case study 1 - Reach-avoid on extended state space

Recall that the parameters describing the stochastic ellipse were $c_k \in \mathbb{R}^2$, denoting the ellipse center and $C_k \in \mathbb{R}^{n\times n}$, denoting its eccentricity. Augmenting the 3D state of the aircraft with these parameters would prevent us from doing any numerical computation due to the high dimensions. Hence, we consider the following simplifications. First, we assume that $\theta_k = 0$ and $M_k = M$ for all k, thereby removing the eccentricity C_k as a state. Additionally, we form a new state corresponding to the relative coordinate of the aircraft and the obstacle process location $(x^1, x^2) - (c_1, c_2) \in \mathbb{R}^2$. The resulting state of the coupled process is denoted by $\bar{x} := (x^1 - c^1, x^2 - c^2, x^3) \in \mathbb{R}^2 \times [-\pi, \pi]$.

Combining equations of the ellipse movement (4.2) with aircraft equations of motion (4.16), the equations of motion of the augmented state are given as

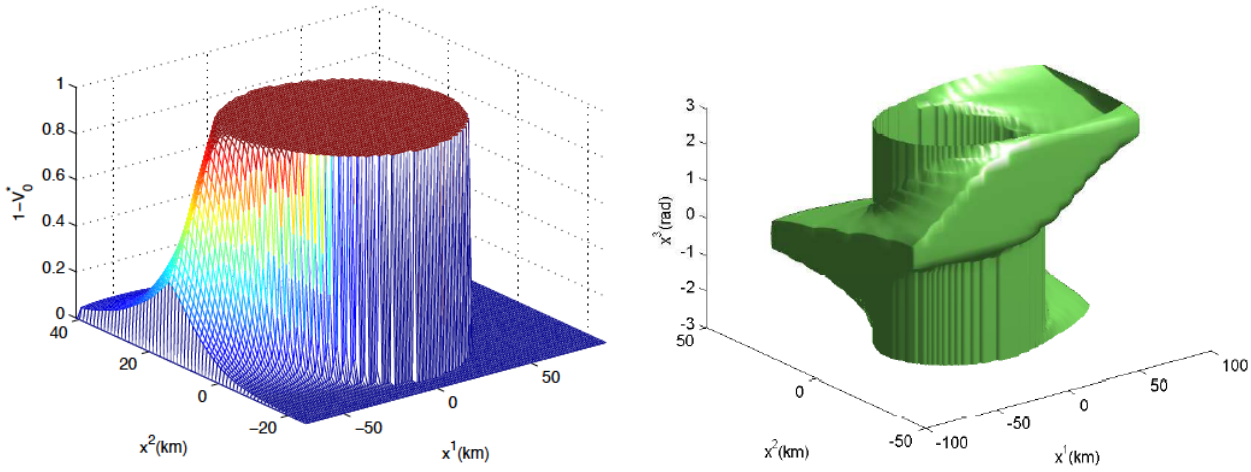
$$\begin{aligned}
\bar{x}^1_{k+1} =& \bar{x}^1_k + \Delta t s_k \cos(\bar{x}^3_k) + \omega^1_k - b^1_k - \eta^1_k,\\
\bar{x}^2_{k+1} =& \bar{x}^2_k + \Delta t s_k \cos(\bar{x}^3_k) + \omega^2_k - b^2_k - \eta^2_k,\\
\bar{x}^3_{k+1} =& \bar{x}^3_k + \Delta t a_k + \omega^3_k.
\end{aligned}$$

The reach-avoid objective now has to be expressed in relative coordinates. For this, consider $K_1 = [-90, 90] \times [-40, 60] \times [-\pi, \pi]$ and $K_2 = [-69, 89] \times [-24, 40] \times [-\pi, \pi]$. We set the target region to $K = K_1 \setminus K_2$, that is, a region that has sufficient distance from the hazardous weather, and the safe set to $K' = K_1 \setminus E$, with E being:

$$E = \mathcal{E}(0, M) \oplus \mathcal{C}(0, 8) \times [-\pi, \pi].$$

Here, $\mathcal{C}(c, r)$ represents a circle with center $c \in \mathbb{R}^2$ and radius r. This circle denotes the protected zone of the aircraft and $\oplus$ represent the Minkowski sum. The reach-avoid objective is to maximize the probability that the aircraft attains K while staying inside the safe set over a horizon of thirty minutes.

Let V_0^* be the optimal value function, which represents the maximum probability of attaining the target region safely at some time during the thirty minute horizon. In Figure 4.5(a) $1 - V_0^*$ (for better visualization) is shown for initial heading angle of $x^3 = -0.1571$ over the 2D relative coordinate space. The interpretation of the figure is as follows: For the relative state initialized at $\bar{x} = (-60$ km, -10 km, -0.1571 rad) the maximum probability of success is 83.08 percent. The set of states with the optimal value function satisfying $V_0^* \geq 0.95$ is shown in Figure 4.5(b). All initial conditions which start outside this level set have a success probability greater than 95 percent. The numerical computations were performed on a $161 \times 67 \times 20$ grid.



(a) Probability map for $x_0^3 = -0.1571$

(b) Set of states with $V_0^* \geq 0.95$

Figure 4.5: Optimal probability of reach-avoid in relative coordinates

Case study 2 - Independent stochastic set process

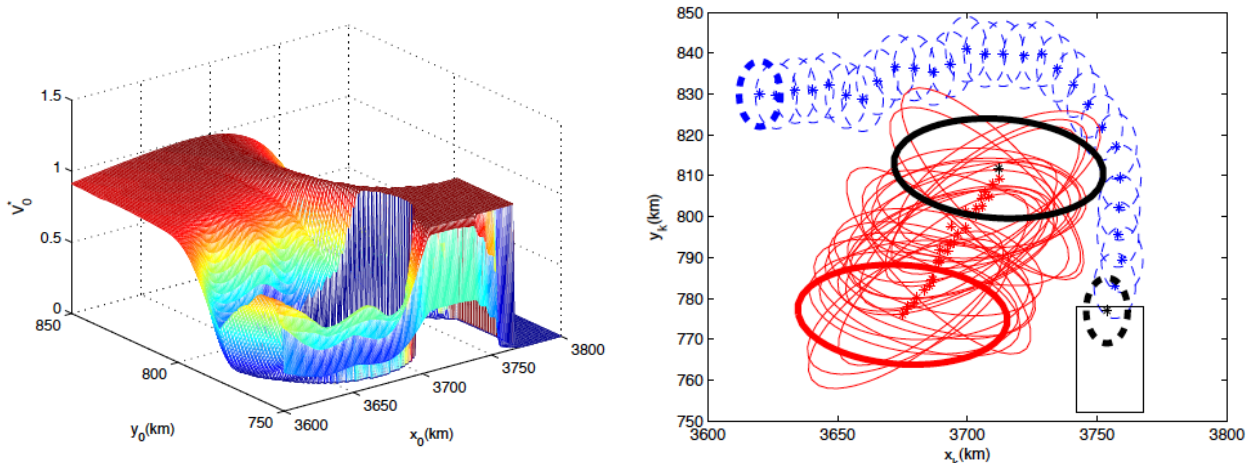
In the previous model of the stochastic set process in the augmented state space, several problem specific simplifying assumptions had to be introduced, in order to reduce the problem dimensionality and to use the nominal numerical methods for probabilistic reach-avoid with deterministic sets. Here, we consider the independence set process assumptions and use the result of Theorem 4.1 to design optimally safe aircraft trajectories.

We model the unsafe regions as random ellipses $\mathcal{E}(c_k, C_k)$ where the motion models for the center c_k and eccentricity C_k were described in Equations (4.2) and (4.3) respectively. The target is a squared rectangular region around a waypoint at 3700×800km, defined by $K = [3742, 3768] \times [752, 778] \times [-\pi, \pi]$. The safe set is set to $K'_k = [3600, 3800] \times [750, 850] \times [-\pi, \pi] \setminus \Xi_k$, where Ξ_k denotes the following random closed set:

$$\Xi_k = \mathcal{E}(c_k, C_k) \oplus \mathcal{C}(0, 8) \times [-\pi, \pi]. \tag{4.17}$$

For a given initial condition of the ellipse center and eccentricity, and the forecast data available at 5-minute increments, we pre-computed the covering function $p_{K'_k \setminus K}(x)$ over the thirty minute horizon using Monte Carlo simulations as described in Example 4.1. We used the dynamic programming algorithm defined by Theorem 4.1 to optimize the probability that the aircraft attains the target set while avoiding the unsafe set over the 30 minute horizon and to synthesize an optimal Markov policy that achieves this probability.

The optimal value function, V_0^*, is shown in Figure 4.6(a) for an initial heading angle of $x^3 = -0.785$ radians. For example, according to this value function, the aircraft initial



(a) Probability map for $x_0^3 = -0.785$

(b) Maximally safe trajectory

Figure 4.6: Maximal reach-avoid probability and an execution of stochastic processes

position of $x_0 = (3620 \text{ km}, 830 \text{ km}, -0.0785 \text{ rad})$, has a maximum probability of success of 93.3 percent. An example execution of the process from this initial condition is shown in Figure 4.6(b). The numerical computations were performed on a $201 \times 101 \times 40$ grid.

Conclusions

We introduced the model of parameterized stochastic set-valued process to address uncertainty arising in locations of the target and safe sets. We showed that evaluating and maximizing the reach-avoid probability for this formulation of the problem can be addressed with the tools developed in the previous chapter for deterministic sets, through extending the state space appropriately. Due to computational limitations in an extended state dimension, we introduced an approximation of the stochastic set process which allowed for an efficient dynamic programming algorithm.

Throughout, we applied our formulation and solution approaches with a practical aircraft trajectory planning problem. In particular, we modeled hazardous weather regions obtained from forecast as stochastic obstacles for the trajectory planning. We then designed maximally safe trajectories, first using the state extension method and then under the Assumption 4.1. As an objective of the NextGen, in the near future, the aircraft would be equipped with sensing and communication technologies such that they could receive information about local weather forecasts. Thus, there are several potential directions for application of this framework to realistic air traffic scenarios.

4.3 Reach-Avoid Problem in Infinite Horizon

In this section, we extend the finite horizon reach-avoid problem for the Discrete-Time Hybrid Dynamic Game (DTSHG) described in Chapter 3 to the case where the time horizon N tends toward infinity. The problem becomes one of characterizing the probability that the system trajectory $(x_0, x_1, \dots)$ enters the target set K at some time $k \geq 0$ before exiting the safe set K'.

Let $\mu = (\mu_0, \mu_1, \dots) \in \mathcal{M}_a$ be an infinite horizon Markov policy for player 1 and let $\gamma = (\gamma_0, \gamma_1, \dots) \in \Gamma_d$ be an infinite horizon Markov strategy for player 2. Then by Proposition 7.28 of [112], the stochastic kernels τ^{μ_k, γ_k}, $k = 0, 1, ..$ induce a unique probability measure $P_{x_0}^{\mu,\gamma}$ on the sample space $\Omega = \bigcup_{k=0}^{\infty} X$. Under a given player 1 policy μ and a player 2 strategy γ, the infinite horizon reach-avoid probability is defined as

$$r_{x_0}^{\mu,\gamma}(K, K') := P_{x_0}^{\mu,\gamma}\{\exists j \geq 0 : x_j \in K \wedge \forall i \in [0, j-1] \;\; x_i \in K' \setminus K\}.$$

The above expression can be equivalently written as

$$\begin{aligned} r_{x_0}^{\mu,\gamma}(K, K') &= P_{x_0}^{\mu,\gamma}\left(\bigcup_{j=0}^{\infty}(K' \setminus K)^j \times K\right) = \sum_{j=0}^{\infty} P_{x_0}^{\mu,\gamma}((K' \setminus K)^j \times K) \\ &= \lim_{N\to\infty} \sum_{j=0}^{N} E_{x_0}^{\mu,\gamma}\left[\left(\prod_{i=0}^{j-1} \mathbf{1}_{K'\setminus K}(x_i)\right)\mathbf{1}_K(x_j)\right] \\ &= \lim_{N\to\infty} r_{x_0}^{\mu_{0\to N}, \gamma_{0\to N}}(K, K'), \end{aligned}$$

where $\mu_{0\to N} = (\mu_0, \dots, \mu_{N-1})$ and $\gamma_{0\to N} = (\gamma_0, \dots, \gamma_{N-1})$ denote the player 1 policy and player 2 strategy, respectively, over time horizon $[0, N]$. In other words, under a fixed infinite horizon policy μ, and a fixed infinite horizon strategy γ, the infinite horizon reach-avoid probability is the limit of the finite horizon reach-avoid probability as $N \to \infty$. In the following, it will be shown that this identity is preserved even as we optimize over player 1 policies and player 2 strategies. Specifically, consider the following definitions of the maxmin reach-avoid probability:

$$r_{x_0}^{N}(K, K') := \sup_{\mu_{0\to N}\in\mathcal{M}_a} \inf_{\gamma_{0\to N}\in\Gamma_d} r_{x_0}^{\mu_{0\to N}, \gamma_{0\to N}}(K, K'), \;\; N \in \mathbb{N} \tag{4.18}$$

$$r_{x_0}^{\infty}(K, K') := \sup_{\mu\in\mathcal{M}_a} \inf_{\gamma\in\Gamma_d} r_{x_0}^{\mu,\gamma}(K, K'). \tag{4.19}$$

The main result of this section is that the infinite horizon maxmin probability can be computed by the limit of the finite horizon maxmin probability:

$$r_{x_0}^{\infty}(K, K') = \lim_{N\to\infty} r_{x_0}^{N}(K, K'), \;\; \forall x_0 \in X. \tag{4.20}$$

By the conclusion of Theorem 3.1 of Chapter 3, it then follows that

$$r^{\infty}_{x_0}(K, K') = \lim_{N\to\infty} T^N[\mathbf{1}_K](x_0),\ \forall x_0 \in X,$$

where T is the dynamic programming operator defined in Chapter 3 as

$$T[J](x) = \sup_{a\in A} \inf_{d\in D} \mathbf{1}_K(x) + \mathbf{1}_{K'\setminus K}(x) H(x, a, d, J), \tag{4.21}$$

$$H(x, a, d, J) = \int_X J(y)\tau(dy|x, a, d).$$

Furthermore, it will be shown that the function $V^* : X \to [0, 1]$ defined by $V^*(x_0) := r^{\infty}_{x_0}(K, K')$, $\forall x_0 \in X$, is the fixed point of the operator T, that is, $V^* = T[V^*]$.

We begin by showing that the limit on the right hand side in Equation (4.20) in fact exists.

Lemma 4.6. *For each $x_0 \in X$, the sequence $\left\{r^N_{x_0}(K, K')\right\}_{N=1}^{\infty}$ converges.*

Proof. For each $N \geq 1$, $r^N_{x_0}(K, K')$ is the finite horizon maxmin reach-avoid probability over $[0, N]$ as defined in the previous chapter and computed in Theorem 3.1. Thus, for every $x_0 \in X$ and $N \geq 1$, $r^N_{x_0}(K, K') \in [0, 1]$. In addition, by Theorem 3.1, for each $N \geq 1$ we have that $r^N_{x_0}(K, K') = T^N[\mathbf{1}_K](x_0)$. From the definition of T in equation (3.7), it is clear that $\mathbf{1}_K \leq T[\mathbf{1}_K]$. Furthermore, by the properties of integrals, it follows directly that the operator T satisfies a monotonicity property: if J, J' are Borel-measurable functions such that $J \leq J'$, then $T[J] \leq T[J']$. Thus, $T^k[\mathbf{1}_K] \leq T^{k+1}[\mathbf{1}_K]$ for every $k \geq 0$. We conclude that, $\forall x_0 \in X$, the sequence $\left\{r^N_{x_0}(K, K')\right\}_{N=1}^{\infty}$ is bounded and monotonically increasing, and hence converges (see for example Theorem 3.14 of [119]). □

From Theorem 3.1, it follows that $\lim_{N\to\infty} r^N_{x_0}(K, K')$ is the limit of a sequence of Borel-measurable functions, and hence is also Borel-measurable (see for example Proposition 2.7 of [117]).

In order to establish the equality in (4.20) we prove that the following two equalities hold:

$$\sup_{\mu\in\mathcal{M}_a} \lim_{N\to\infty} \inf_{\gamma\in\Gamma_d} r^{\mu_{0\to N},\gamma_{0\to N}}_{x_0}(K, K') = \lim_{N\to\infty} \sup_{\mu\in\mathcal{M}_a} \inf_{\gamma\in\Gamma_d} r^{\mu_{0\to N},\gamma_{0\to N}}_{x_0}(K, K') \tag{4.22}$$

$$\sup_{\mu\in\mathcal{M}_a} \lim_{N\to\infty} \inf_{\gamma\in\Gamma_d} r^{\mu_{0\to N},\gamma_{0\to N}}_{x_0}(K, K') = \sup_{\mu\in\mathcal{M}_a} \inf_{\gamma\in\Gamma_d} \lim_{N\to\infty} r^{\mu,\gamma}_{x_0}(K, K') = r^{\infty}_{x_0}(K, K') \tag{4.23}$$

The derivation of the above two equalities is based on [112] which establishes conditions under which the limit and minimization commute. Let us define $J^{\mu,\gamma}_N(x) := -r^{\mu_{0\to N},\gamma_{0\to N}}_x(K, K')$ and J^{**}, J_∞ and define J^* associated with the three limits in Equations (4.22) and (4.23) as

$$J^{**}(x) := \inf_{\mu\in\mathcal{M}_a} \lim_{N\to\infty} \sup_{\gamma\in\Gamma_d} J^{\mu,\gamma}_N(x),$$

$$J_\infty(x) := -\lim_{N\to\infty} r^N_x(K, K'),$$

$$J^*(x) := -r^{\infty}_{x_0}(K, K').$$

We proceed by proving that $J_\infty = J^{**}$ and hence conclude the equality (4.22). To show this, first define the operator $\tilde{T}_{\pi,\lambda} : \mathcal{F} \to \mathcal{F}$ for $\pi : X \to A$ and $\lambda : X \times A \to D$ as:

$$\tilde{T}_{\pi,\lambda}[J](x) = -\mathbf{1}_K(x) + \mathbf{1}_{K'\setminus K}(x) H(x, \pi(x), \lambda(x, \pi(x)), J).$$

In addition, define the map $\tilde{G} : X \times A \times \mathcal{F} \to \mathbb{R}$, and the operator $\tilde{T}_\pi : \mathcal{F} \to \mathcal{F}$, for $\pi : X \to A$ as

$$\begin{aligned} \tilde{G}(x, a, J) &= \sup_{d \in D} -\mathbf{1}_K(x) + \mathbf{1}_{K'\setminus K}(x) H(x, a, d, J), \\ \tilde{T}_\pi(J) &= \tilde{G}(x, \pi(x), J). \end{aligned}$$

From Theorem 3.1 of Chapter 3, for a fixed Markov policy μ,

$$\sup_{\gamma \in \Gamma_d} J_N^{\mu,\gamma} = \tilde{T}_{\mu_0} \tilde{T}_{\mu_1} \dots \tilde{T}_{\mu_{N-1}}[-\mathbf{1}_K]. \tag{4.24}$$

Proposition 4.2. *Define $J_0(x) = -\mathbf{1}_K(x)$. The map $\tilde{G}$ satisfies the following properties:*

(a) $\tilde{G}(x, a, J_0) \le J_0$, $\forall a \in A$

(b) If $(J_k) \in \mathcal{F}$ is a sequence satisfying $J_{k+1} \le J_k \le J_0$, then $\forall a \in A$

$$\lim_{k\to\infty} \tilde{G}(x, a, J_k) = \tilde{G}(x, a, \lim_{k\to\infty} J_k).$$

(c) There exists a scalar $\alpha > 0$ such that for all scalars $r > 0$ and functions $J \in \mathcal{F}$ with $J \le J_0$ we have the following inequalities $\forall a \in A$:

$$\tilde{G}(x, a, J) - \alpha r \le \tilde{G}(x, a, J - r) \le \tilde{G}(x, a, J)$$

Proof. (a) Since τ is a stochastic kernel we get that $-1 \le H(x, a, d, -\mathbf{1}_K) \le 0$, and as such $-1 \le \mathbf{1}_{K'\setminus K} \sup_{d\in D} H(x, a, d, -\mathbf{1}_K) \le 0$. The result on $\tilde{G}$ follows by adding $-\mathbf{1}_K(x)$ to both sides of the right hand side inequality.

(b) Since $J_{k+1} \le J_k \le 0$, by the Monotone Convergence Theorem, $\lim_{k\to\infty} J_k(x)$ exists. Then,

$$\begin{aligned} &\sup_{d\in D} \int_X \lim_{k\to\infty} J_k(y) \tau(dy|x, a, d) = \sup_{d\in D} \lim_{k\to\infty} \int_X J_k(y) \tau(dy|x, a, d) \\ &= \lim_{k\to\infty} \sup_{d\in D} \int_X J_k(y) \tau(dy|x, a, d) = \lim_{k\to\infty} \sup_{d\in D} H(x, a, d, J_k) \end{aligned}$$

In the above, the first equality follows from the Monotone Convergence Theorem. For the next equality, note that $\int_X J_k(y)\tau(dy|x, a, d)$ is continuous on D by Assumption 3.1 and D is compact. Hence, an application of Proposition 10.1 of [144] gives the desired result. Since $\tilde{G}(x, a, J_k) = -\mathbf{1}_K(x) + \mathbf{1}_{K'\setminus K} \sup_{d\in D} H(x, a, J_k)$ the desired result follows.

(c) It is easy to verify that the inequalities hold for $\alpha = 1$. □

By the above proposition, the assumptions of Proposition 5.3 in [112] hold. Thus, from (4.24) and definition of J^{**} it follows that

$$J^{**} = \inf_{\mu\in\mathcal{M}_a} \lim_{k\to\infty} (\tilde{T}_{\mu_0}\tilde{T}_{\mu_1}\dots\tilde{T}_{\mu_{k-1}})[-\mathbf{1}_K],$$

is a fixed point of the dynamic programming operator $\tilde{T} : \mathcal{F} \to [0,1]$, defined as

$$\tilde{T}[J](x) = \inf_{a\in A} \tilde{G}(x,a,J) = \inf_{a\in A} \sup_{d\in D} -\mathbf{1}_K(x) + \mathbf{1}_{K'\setminus K}(x)H(x,a,d,J).$$

That is, $J^{**} = \tilde{T}[J^{**}]$. In addition, by the result in [112], J^{**} can be computed by the infinite horizon dynamic programming recursion. That is, $J^{**} = J_\infty$ as desired. It follows that equality (4.22) hold. Let $V_\infty = \lim_{N\to\infty} r_x^N(K,K') = -J_\infty$. From the definitions of $\tilde{T}$ and T, it also follows that $T[V_\infty] = V_\infty$.

Now, we proceed by showing the equality (4.23). In order to use the result of [112] which considers a single player minimization problem, we show that for a stationary policy $\mu = (\pi,\pi,\dots)$, $\inf_{\gamma\in\Gamma_d} \lim_{N\to\infty} V_0^{\mu,\gamma} = \lim_{N\to\infty} \inf_{\gamma\in\Gamma_d} V_0^{\mu,\gamma}$, where $V_k^{\mu,\gamma} = -J_{N-k}^{\mu,\gamma}$, as was also defined in Chapter 3 through a backwards recursion. First, we need to make an assumption on existence of optimal stationary policies:

Assumption 4.2. There exists a Borel measurable stationary policy μ^*, such that

$$\sup_{\mu\in\mathcal{M}_a} \lim_{k\to\infty} \inf_{\gamma\in\Gamma_d} V_k^{\mu,\gamma} = \lim_{k\to\infty} \inf_{\gamma\in\Gamma_d} V_k^{\mu^*,\gamma}.$$

For the stationary policy $\mu = (\pi,\pi,\dots)$, define the map $G : X\times D\times\mathcal{F}\to\mathbb{R}$ and the operator $T_\mu : \mathcal{F}\to\mathcal{F}$ as:

$$\begin{aligned} G(x,d,J) &= \mathbf{1}_K(x) + \mathbf{1}_{K'\setminus K}(x)H(x,\mu(x),d,J),\\ T_\mu[J] &= \inf_{d\in D} \mathbf{1}_K(x) + \mathbf{1}_{K'\setminus K}(x)H(x,\pi(x),d,J). \end{aligned}$$

Let $J_0(x) = \mathbf{1}_K(x)$. Similar to Proposition 4.2, the following properties for G can be verified:

(a) $G(x,d,J_0) \geq J_0$, $\forall d\in D$

(b) If $(J_k)\in\mathcal{F}$ is a sequence satisfying $J_{k+1}\geq J_k\geq J_0$ then for all $d\in D$

$$\lim_{k\to\infty} G(x,d,J_k) = G(x,d,\lim_{k\to\infty} J_k)$$

(c) There exists a scalar $\alpha > 0$ such that for all scalars $r > 0$ and functions $J\in\mathcal{F}$ with $J\leq J_0$ we have the following inequalities for all $d\in D$, $G(x,d,J)\leq G(x,d,J+r)\leq G(x,d,J)+\alpha r$.

Next, for $k > 0$, consider the level-set $\Omega_{x,k} := \{d \in D \mid G(x, d, T_\mu^k[J_0]) \leq b\}$. Since the stochastic kernel τ is continuous in players' inputs, $G(x, d, J)$ is continuous in d for any $J \in \mathcal{F}$. As such, $\Omega_{x,k}$ is pre-image of $[0, b]$ under a continuous map and is closed. From the fact that $\Omega_{x,k}$ is closed and properties (a), (b), (c) above we conclude that the assumptions of Proposition 5.10 of [112] hold and it follows that $\inf_{\gamma\in\Gamma_d} \lim_{N\to\infty} V_0^{\mu,\gamma} = \lim_{N\to\infty} \inf_{\gamma\in\Gamma_d} V_0^{\mu,\gamma}$. In particular, this equality holds when the maximizing stationary policy of Assumption 4.2 is used and we obtain:

$$\inf_{\gamma\in\Gamma_d} \lim_{N\to\infty} V_0^{\mu^*,\gamma} = \sup_{\mu\in\mathcal{M}_a} \inf_{\gamma\in\Gamma_d} \lim_{N\to\infty} V_0^{\mu,\gamma} = \sup_{\mu\in\mathcal{M}_a} \lim_{N\to\infty} \inf_{\gamma\in\Gamma_d} V_0^{\mu,\gamma}.$$

Thus, equality (4.23) holds. Finally, since both Equations (4.22) and (4.23) hold we conclude the desired result:

$$V_\infty := \lim_{N\to\infty} r_{x_0}^N(K, K') = r_{x_0}^\infty(K, K') = V^*.$$

In addition, by the fact that $V_\infty = T[V_\infty]$, it follows that $T[V^*] = V^*$ as desired.

Remark: Admittedly, Assumption 4.2 may seem strong. Thus, we are working on providing conditions under which this assumption holds. Additionally, we are working towards an alternate proof of the convergence of the infinite horizon dynamic programming algorithm which does not require this assumption.

In the following section, we describe a target tracking game between an aerial and a ground vehicle. The finite horizon target tracking game was illustrated in [53]. Here, the objective of the aerial vehicle is to gain coverage of the ground vehicle at some time $k \geq 0$ before violating its velocity bounds.

Target Coverage Game

The scenario is as follows: An autonomous quadrotor helicopter, considered as player 1, wants to achieve coverage of a moving ground vehicle, considered as player 2. The experimental set up is based on the Stanford Testbed of Autonomous Rotorcraft for Multi-Agent Control (STARMAC), an unmanned aerial vehicle platform consisting of six quadrotor helicopters each equipped with onboard computation, sensing, and control capabilities [145]. The quadrotor helicopter is shown in Figure 4.7(a) and the target coverage experiment is shown in Figure 4.7(b).

In many aerial robotic platforms, there are unmodeled high order dynamics, aerodynamic effects, and actuator and measurement noise. For example, at high speeds several aerodynamic effects impact the flight characteristics of the STARMAC quadrotors and these effects are difficult to model [146]. In order to account for these uncertainties, the authors in [147] addressed the target coverage problem with a deterministic robust approach. Accounting for

(a) STARMAC quadrotor (b) STARMAC tracking a ground vehicle

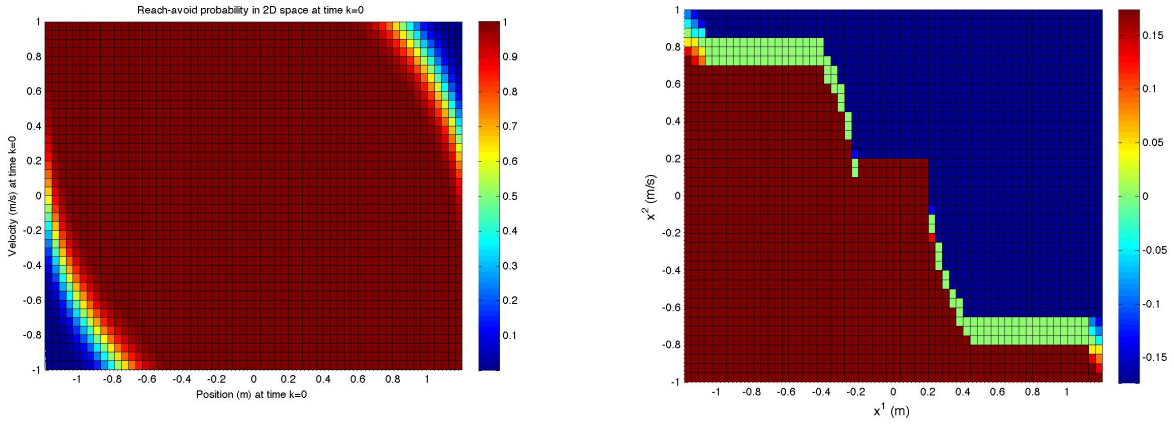
Figure 4.7: STARMAC quadrotor helicopter hardware and experiment

all of the uncertainties deterministically, using a robust control approach may lead to very conservative control laws, or even to the lack of a control law that can achieve the reach-avoid objective. As such, we introduce a stochastic framework to capture the modeling uncertainties and formulate a probabilistic reach-avoid problem.

Let x^1, x^2, y^1, y^2 denote the position and velocity of the quadrotor relative to the ground vehicle in the x axis and y axis respectively. Under a previously designed inner control loop, the position-velocity dynamics in the planar x and y directions can be assumed to be decoupled, with pitch θ and roll ϕ angles as the respective control inputs. Then from the point of view of the high level controller the dynamics of the vehicle under pitch and roll commands can be approximated as

$$\begin{aligned} x^1_{k+1} &= x^1_k + \Delta t x^2_k + \frac{\Delta t^2}{2}(g\sin(\theta_k) + d^x_k) + \eta^1_k \\ x^2_{k+1} &= x^2_k + \Delta t(g\sin(\theta_k) + d^x_k) + \eta^2_k \\ y^1_{k+1} &= y^1_k + \Delta t y^2_k + \frac{\Delta t^2}{2}(g\sin(-\phi_k) + d^y_k) + \eta^3_k \\ y^2_{k+1} &= y^2_k + \Delta t(g\sin(-\phi_k) + d^y_k) + \eta^4_k \end{aligned}$$

In the above, Δt is the discretization time step, g is the gravitational acceleration constant, and d^x and d^y are the unknown terms corresponding to the acceleration of the ground vehicle. The terms η^i_k, for $i = 1, \ldots, 4$, represent noise arising from measurement and actuation. The noise is assumed to have a Gaussian distribution, with $\eta^i \sim \mathcal{N}(0, {\sigma^i}^2 \Delta t^2)$. Based upon experimental trials, the bounds for the accelerations, d^x and d^y, of the ground vehicle are chosen to be $[-.4, .4]$ m/s^2 corresponding to about 30% of the maximum allowable



(a) Maxmin probability of reach-avoid

(b) Maxmin policy for the quadrotor

Figure 4.8: Infinite horizon reach-avoid target coverage game

acceleration of the ground vehicle [147]. The roll and pitch commands are assumed to be in the range $[-10, 10]$ degrees. The variances of the noise are set to $\sigma^i = 0.4$, for $i = 1, 2, \ldots, 4$.

The target set is chosen to be a square centered on the ground vehicle, with some tolerance on the relative velocity. In the x^1, x^2 coordinates this set is specified by

$$K = [-0.2, 0.2]m \times [-0.2, 0.2]m/s.$$

The unsafe set is the set of all positions in which the quadrotor would be too far from the ground vehicle to maintain observation, and the set of velocities violating the velocity bounds. Hence, the safe set in the x^1, x^2 coordinates is given as

$$K' = [-1.2, 1.2]m \times [-1, 1]m/s.$$

The target set and the safe set in the y axis are the same as those in the x axis. The infinite horizon reach-avoid objective is then defined as maximizing the probability of reaching the coverage region at some time while remaining inside the safe set at all prior times. That is, we need to find $r^{\infty}_{x_0}(K, K')$.

Given that the dynamics, the target and safe sets in the x and y axis are decoupled, we can perform the reach-avoid analysis for each axis separately and reduce the problem dimension from four to two. By the conclusions of the previous section, under Assumption 4.2, $r^{\infty}_{x_0}(K, K') = \lim_{N\to\infty} T^N[\mathbf{1}_K](x_0)$. Thus, based on the dynamic programming algorithm given in Theorem 3.1, we performed the recursion $T^N[\mathbf{1}_K](x_0)$ to find the infinite horizon maxmin reach-avoid probability.

The computation was performed on a discretized grid of the state space. The grid size was 61×41 in position-velocity plane for each axis. The input spaces for both players were also discretized. On the one hand, the discretization was required for the numerical computation of the algorithm. On the other hand, quantization of the input levels also results from the fact that the quadrotor helicopter is controlled digitally by an on-board computer. As such, the continuous range for the roll and pitch commands were discretized at a 2.5 degree discretization step. For numerical computation, the disturbance range was discretized at 0.1 m/s^2 intervals.

After $N = 60$ iterations, the maxmin reach-avoid probability converged to a stationary value. Convergence was defined by the fact that $\|V_N^* - V_{N+1}^*\|_\infty \leq e^{-8}$. Similarly, it was found that the maxmin optimal quadrotor policy converged to a stationary policy $\mu^* : X \to A$. The maxmin reach-avoid probability for the set of initial conditions insie the safe set K', is shown in Figure 4.8(a). The optimal stationary feedback quadrotor policy that achieves the maxmin probability is shown in Figure 4.8(b).

Conclusions

We considered the probabilistic reach-avoid problem in infinite horizon. Under an assumption on the existence of stationary optimal policies, we showed that the infinite horizon maxmin reach-avoid probability can be computed as limit of a finite horizon reach-avoid probability. In addition, this optimal value function is a fixed point of the dynamic programming operator. It is interesting to explore how can one solve for the fixed point of the dynamic programming operator, in a method that is computationally more efficient than doing a large number of iteration of the dynamic programming algorithm. In addition, it is important to find conditions under which stationary optimal policies and strategies exist.

We applied the result to a target tracking application by a quadrotor helicopter in which the objective was to obtain coverage of a ground vehicle at some point in the horizon. In future, we would like to test the optimal maxmin policy obtained from the dynamic programming algorithm on the experimental platform and to compare the performance of the probabilistic method used here with that of the deterministic one presented in [147].

Chapter 5

Air Traffic Optimization During Runway Configuration Switch

We develop a hybrid dynamical model as a detailed abstraction of runway configuration modes and arrival traffic in a terminal airspace. The problem of optimization of arrival traffic together with runway configuration planning is posed as a hybrid optimal control problem. Probabilistic constraints due to the presence of hazardous weather and wind are posed. We develop a hierarchical algorithm in which, at the top stage runway configuration and aircraft paths are determined such that safety with respect to weather conditions is addressed and at the bottom stage the optimization of arrival traffic is addressed. The material in this chapter was presented in [58, 59].

5.1 Background

Air transportation in the United States is regulated in a hierarchical and distributed manner, in which the airspace is divided into subregions, such as air route traffic centers, sectors, and terminal areas. The different subregions are controlled by different groups of air traffic controllers so that the overall responsibility is decoupled. A terminal area refers to a region of airspace that is within approximately a 50 nautical mile (nmi) radius of an airport and its control is divided between the Terminal Radar Approach Control (TRACON) and the Tower Control. It is the most crowded portion of the airspace and is often the throughput bottleneck of the airspace system. Due to its importance in safety and capacity of the airspace, a concept of operations for the NextGen terminal airspace, referred to as Super-Density Operations, has been proposed.

Super-Density Operations envision the use of advanced ground and flight deck automation, efficient Area Navigation (RNAV) and Required Navigation Performance (RNP) routes,

optimized vertical profiles and delegated interval management to maintain efficient utilization of terminal airspace even in adverse weather conditions. The main challenge in achieving highly efficient operations lies in weather and traffic uncertainties as well as configuration changes in runways or airways necessitated by adverse weather conditions.

The problem of runway configuration management introduced in Chapter 1 is as follows: In airports with multiple intersecting runways a set of active runways, referred to as a runway configuration, for arrival or departure are chosen based on factors including the crosswind and tailwind magnitudes, visibility, traffic flow and noise abatement laws. The choice of runway configuration in a major airport affects the arrival routes of incoming traffic to the airport, as well as to other nearby airports in the same TRACON airspace. Conversations with air traffic controllers in New York and Boston TRACON[1] indicate that unanticipated runway switching, usually due to weather conditions, increases the workload of the air traffic controllers and pilots. In order to deal with the changes in the airspace route structure during these unanticipated switches, several incoming aircraft are put in holding patterns. As such, there is a lost capacity during the transitional periods of configuration switches which can result in large delays, specifically in super dense airspaces.

This chapter is motivated by the vision of Super-Density Operations in better planning of runway configuration switching and the arrival traffic management during switching.

Previous Work

The previous work on terminal airspace air traffic management falls into three categories depending on whether air traffic control, airspace management or runway configuration management is addressed. In the first category, optimization of aircraft landing times given fixed arrival routes and runways is addressed [148, 149, 150]. Bayen et al. [148] formulate a Mixed Integer Linear Program (MILP) to minimize aircraft delay given fixed arrival routes and approximate the MILP with a polynomial-time algorithm. Balakrishnan et al. [150] determine aircraft arrival sequence using Constrained Position Shifting in order to reduce delays while satisfying the arrival spacing requirements. These works do not consider the complexity arising due to switching of the runway configuration nor the weather effects on the arrival route availability.

Next, research has studied effects of hazardous weather on the availability of predefined routes in the terminal area. The Route Availability Planning Tool (RAPT) has been developed based on the Corridor Integrated Weather System (CIWS) product [141] to help air traffic controllers assess the availability of departure routes in adverse weather [151]. Michalek et al. [152] use machine learning algorithms to determine routes that are robustly safe to fly through under weather uncertainties. Reconfiguration of airspace by designing routes that

[1] JFK TRACON visit, 2007; Boston Logan airport visit, 2010

are safe with respect to hazardous weather is considered by Krozel et al. and Michalek et al. [153, 154]. These works do not consider control of the arrival traffic or runway configuration management in adverse weather.

Recently, research has begun to consider the problem of runway configuration planning. Roach [155] discusses configuration planning based on wind data and analyzes air traffic delays caused by non-prevailing wind conditions at Dallas/Fort Worth airport. In the work of Leihong et al. [156] wind forecast data is used in order to determine feasibility of runway configurations in a given future time horizon. The authors then formulate a dynamic programming algorithm to address runway configuration selection in order to maximize the throughput of the landing aircraft. Ramanujam et al. [157] determine a set of factors that are used in choosing a runway configuration and then apply machine learning in order to model the air traffic controllers' decisions in choosing runway configurations. These works do not consider the determination of the optimal switch times between the configurations and the control of the arrival traffic during the switching. In reality, in many instances the configuration sequence may be known to the air traffic controllers, due to availability of weather forecasts and previously established procedures, while the switch times between configuration changes and the management of arrival traffic need to be determined optimally in order to minimize delays resulting from the transitional periods of the configuration switches.

Current work

Our goal here is to develop (a) an accurate model for the arrival traffic dynamics in terminal airspace that takes into account weather uncertainties and runway configuration changes; (b) a traffic control algorithm based on the model that can reduce delays or other desired cost factors. In Section 5.2, we mathematically define the problem of runway configuration and aircraft scheduling and develop a hybrid system model to describe the dynamics of the arrival traffic. In this model, the discrete modes represent the runway configurations and the continuous states represent the locations of the aircraft in the terminal airspace. The runway switching problem is formulated as an optimal control problem of a hybrid system that requires minimization of the total delay subject to the separation constraints between the aircraft as well as configuration and path constraints due to weather. In Section 5.3, we define our solution approach for addressing the problem. Even though the hybrid optimal control problem has a large scale, simplified dynamics of aircraft as well as existence of certain hard constraints facilitate a simplified hierarchical solution approach. In Section 5.4, we apply the model and solution approach to an instance of configuration planning at JFK airport. Finally, we summarize our results and directions for future work in Section 5.5.

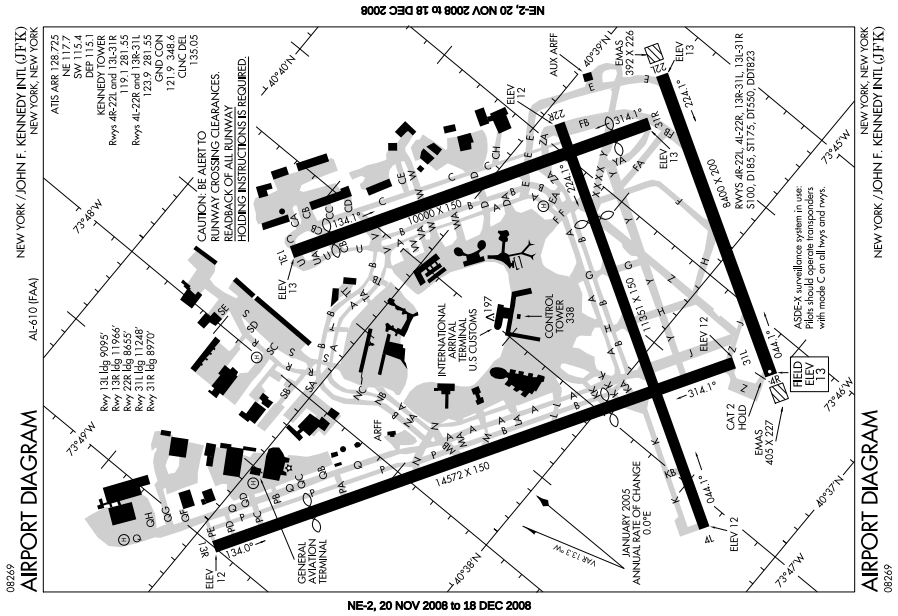


Figure 5.1: JFK airport diagram

5.2 Problem Model

Figure 5.1 shows JFK Airport diagram obtained from the Federal Aviation Administration (FAA) website[2]. The main purpose of these Airport Diagrams is information about the runways. A runway is labeled based on its direction relative to the magnetic North. For example, an arrival on $22L$ indicates that aircraft will be landing with a heading of approximately 220 degrees measured from the magnetic North along this runway. JFK airport has two pairs of parallel, labeled as L for left and R for right, runways. This leads to arrivals from four possible directions and to several possible arrival runway configurations. For example, a common configuration in periods of high traffic is $\{22L, 22R, 31L\}$.

The set of runways that are selected for landing at any airport is referred to as the arrival runway configuration and will simply be referred to as the runway configuration in the rest of this paper. The configuration may change several times in a day because it is selected by considering various factors such as wind direction and magnitude, noise level, visibility and air traffic patterns. The choice of configuration affects the air traffic routes in the terminal airspace. The airspace model developed here captures the air traffic routes and the runway configurations, while the hybrid dynamic model of the aircraft captures the motion of aircraft on these arrival routes.

[2] `http://www.faa.gov/airports/runway_safety/diagrams`

Hybrid Model of Arrival Traffic

Airspace model

Aircraft are often required to enter and leave the terminal airspace through certain fixed locations in airspace called meter fixes. For each meter fix, there are usually several predefined paths leading to different runways in the airport. This pathway structure within the terminal area can be easily described by a directed graph $G = (V, E)$, where each node $v \in V$ in this graph represents a waypoint, including runways, in the terminal airspace, and each edge $e = (v_1, v_2) \in E \subset V \times V$ represents a directed aircraft route from node v_1 to node v_2. Each edge is associated with a length l_e as well as a set of neighbors $\mathcal{N}(e)$ consisting of the edges that share a common node with e. A node is called an *entry node* if it connects the terminal airspace to the en-route airspace and is called a *sink node* if it corresponds to a runway. The set of entry and sink nodes are denoted by V_e and V_s, respectively. For example, in JFK, $V_s = \{22L/R, 31L/R, 4L/R, 13L/R\}$ as shown in Figure 5.1. A path is a connected set of edges that starts at an entry node and ends at a sink node. The set of edges $e = (v_1, v_2)$ with $v_2 \in V_s$ is denoted by E_s, referred to as the set of final edges.

A mode of the graph is characterized by a set of sink nodes $\sigma \subset V_s$. There is a one-to-one correspondence between the graph modes and the runway configurations. We will use the terms mode and runway configuration interchangeably in the rest of this chapter. If the graph is in mode σ then the configuration includes runways which are represented by nodes in σ. In this case, the edge $e = (v_1, v_2) \in E_s$ is available as a route if and only if $v_2 \in \sigma$.

The control input for the graph is the choice of graph mode over a time horizon. This choice over an interval of time $[t_0, t_f]$ is represented as:

$$[(t_{s_0}, \sigma_0), (t_{s_1}, \sigma_1), \ldots, (t_{s_N}, \sigma_N)], \tag{5.1}$$

where $0 < N < \infty$, $t_{s_0} \leq t_{s_1} \cdots \leq t_{s_N} \leq t_f$ and $\sigma_k \subset V_s$ for $k = 0, 1, \ldots, N$. In this sequence, the pair (t_{s_0}, σ_0) is the initial condition and the pair (t_{s_k}, σ_k), $k \geq 1$, indicates that at time t_{s_k} the graph mode changes from β_{k-1} to β_k. As a consequence, in the time interval $[t_{s_k}, t_{s_{k+1}})$ the graph mode is given by σ_k. Thus, during this time interval, all arrival aircraft must use a runway $v \in \sigma_k$.

Aircraft dynamics

Let $[t_0, t_f]$ be a time interval of interest for optimizing arrival traffic. Suppose there are N_a scheduled arrivals during this interval with the i-th aircraft crossing one of the entry nodes at time t_0^i. Once aircraft enter the terminal airspace, they should travel along the pathways defined by graph G. For aircraft i, let $q^i(t) \in E$ be the edge it is on at time t, $x^i(t) \in \mathbb{R}_+$ be its current distance from the first node of edge $q^i(t)$, $y^i(t) \in \mathbb{R}_+$ be the total distance it has traveled since time t_0^i and $z^i(t) = (q^i(t), x^i(t), y^i(t))$ be its hybrid state.

The evolution of the hybrid state $z^i(t)$ is controlled by air traffic controllers through speed adjustment, edge selection and holding pattern assignment. We assume speed changes and holding patterns occur only when the aircraft is at one of the nodes in the graph. We denote a generic air traffic control command as $u = (s, \eta)$ where $s \in \mathbb{R}_+$ is the speed magnitude assignment and $\eta \in E \cup \{\text{hold}\}$ is the discrete control command specifying whether the aircraft needs to travel along the new edge specified by η (when $\eta \in E$) with speed s or enter a holding pattern (when $\eta = \text{hold}$) with speed s at the current node.

Suppose that aircraft i is at node $v \in V$ at some time $\hat{t} \geq t_0^i$ and receives a control $u = (s, \eta)$. If $\eta = \text{hold}$, then evolution of the hybrid state is given by:

$$\begin{bmatrix} q^i(t) \\ \dot{x}^i(t) \\ \dot{y}^i(t) \end{bmatrix} = \begin{bmatrix} q^i(\hat{t}) \\ 0 \\ s \end{bmatrix}.$$

On the other hand, if $\eta \in \mathcal{N}(q^i(\hat{t}))$, then the hybrid state first undergoes an instantaneous reset to $z^i(\hat{t}_+) = (\eta, 0, y^i(\hat{t}))$, where $\hat{t}_+$ denotes the time immediately after $\hat{t}$. The dynamics after time $\hat{t}_+$ is:

$$\begin{bmatrix} q^i(t) \\ \dot{x}^i(t) \\ \dot{y}^i(t) \end{bmatrix} = \begin{bmatrix} \eta \\ s \\ s \end{bmatrix}.$$

The above two evolutions continue until the aircraft finishes the number of holding patterns assigned or reaches the next node, at which time it will receive a new control command and the process repeats. The evolution stops once the aircraft reaches one of the sink nodes, which can be determined through the hybrid state by checking whether the edge $q^i(t)$ is a final edge and $x^i(t) = l_{q^i(t)}$. The time aircraft i reaches a sink node is denoted by t_f^i.

Let n^i denote the number of edges in the aircraft path from the source to the sink node. The set of all controls for aircraft i is given by $u^i = (s^i, \eta^i)$, where the continuous control is $s^i = (s_1^i, \ldots, s_{n^i}^i)$ and the discrete control is $\eta^i = (\eta_1^i, \ldots, \eta_{n^i}^i)$.

Constraints

There are constraints on the airspace due to weather conditions and on the aircraft due to separation requirements between the aircraft.

Airspace constraints

The weather can affect the dynamics of the graph by affecting availability of the edges. If a significant portion of an edge is blocked by a storm or hazardous weather, then no aircraft

can be assigned to that edge. In addition, if a runway is prohibited from landing due to strong wind or other environmental conditions, then no aircraft can be assigned to any of the final edges leading to the sink node corresponding to the runway.

First, we discuss runway feasibility. We focus on the effects of wind on runway selection and do not consider other factors such as noise abatement which are dependent on the particular airport and procedures. The wind direction and magnitude is a major factor in determining whether a runway is safe for landing because aircraft cannot safely land if the component of the wind perpendicular to the landing direction, referred to as the crosswind, or parallel to the landing direction, referred to as the tailwind, are above certain thresholds.

The wind forecast data is provided in terms of wind magnitude and direction at regular intervals. Comparison of the historical forecast data with actual wind data could provide reasonable models for the probabilities of the wind magnitude and direction being within certain bounds of the predicted values. Thus, from forecast data, we assume the probabilities that the crosswind and tailwind to runway v are below the required thresholds are estimated. Denote these probabilities by $p_{v,cw}$ and $p_{v,tw}$ respectively. We define a configuration or graph mode feasible if the probabilities of crosswind and tailwind threshold satisfaction are above a desired level $\lambda_w \in (0, 1]$ for each runway in the configuration. The constraint for the graph mode sequence of Equation (5.1) over the time horizon $t \in [t_{s_k}, t_{s_{k+1}}]$ can then be written as

$$\sigma_k \in \{c \subset V_s \mid \forall v \in c, \;\; p_{v,cw}(t) \geq \lambda_w \;\wedge\; p_{v,tw}(t) \geq \lambda_w\}. \tag{5.2}$$

Next, we discuss the edge feasibility. Research in the past has used the weather forecast data in order to identify routes that will remain open in the forecast horizon and has provided the uncertainty associated with this prediction [152]. Thus, we assume that we have the probability $p_e(t)$ of edge $e = (v_1, v_2)$ being open at time t in the planning horizon. In order to assign aircraft to edge e, we require that the edge is open with high enough probability, that is, $p_e(t) \geq \lambda_e$, where $\lambda_e \in (0, 1]$ is a parameter determined by the safety requirements. Consequently, the discrete aircraft control η at time t and at node v_1 has to satisfy

$$\eta \in \{e \in E \mid p_e(t) \geq \lambda_e\} \cup \{0, 1, \ldots, H\}. \tag{5.3}$$

In the above, the maximum number of allowable holding patterns at a node is denoted by H. A final edge $e = (v_1, v_2) \in E_s$, must additionally satisfy $v_2 \in c$ where c is a feasible runway according to (5.2).

State constraints

For safety requirements the aircraft on the same edge or neighboring edges must be separated by a given distance d. In addition, for safety due to wake vortex of aircraft, there are runway separation distance requirements based on leading and trailing aircraft types. Let the type

of aircraft i be denoted by a^i. The runway separation distance between aircraft i and j is denoted by $D(a^i, a^j)$. Let matrix $D \in \mathbb{R}_+^{n_t \times n_t}$ represent the runway separation requirement between all pairs of aircraft, in which n_t denotes the number of different aircraft types. In general, the separation requirement and hence the matrix D would also depend on the landing runway of each pair of aircraft. For simplicity in notation, we drop this dependence here. The separation constraints along the edges and runways are encoded with the constraint $h(z^i, z^j, t) \leq 0$, where the function h is defined as:

$$h(z^i, z^j, t) = \begin{cases} x^j(t) - x^i(t) + d, & \text{if } q^j(t) = q^i(t), \\ x^j(t) - x^i(t) + d, & \text{if } q^j(t) \in \mathcal{N}(q^i(t)) \ \wedge \ x^i(t) = l_{q^i(t)}, \\ x^j(t) + D(a^i, a^j) - l_{q^j(t)}, & \text{if } q^i(t), q^j(t) \in E_s \ \wedge \ x^i(t) = l_{q^i(t)}. \end{cases} \tag{5.4}$$

The first constraint denotes the separation requirement for two aircraft on the same edge, the second denotes the separation requirement for aircraft on neighboring edges and the third denotes the separation requirement for landing aircraft. In all cases, it is assumed that aircraft j precedes aircraft i. The constraint on the final state is the requirement that aircraft land at a runway by some time $t \in [t_0, t_f]$ in the planning horizon:

$$z^i(t) \in \{(q, x, y) \mid q \in E_s \ \wedge \ x(t) = l_q\}. \tag{5.5}$$

Optimization

For each aircraft, we penalize a function of the aircraft state by defining a running cost function $L(z^i)$. This function could for example denote the total distance or travel time of the aircraft and hence in general is a function of the discrete state representing edges and holding patterns in aircraft path and the continuous state representing the distance travelled. We associate a cost due to switching from graph mode σ_i to mode σ_j, $S(\sigma_i, \sigma_j)$, due to overhead associated with switching runway configuration. Let $u = (u^1, u^2, \ldots, u^{N_a})$ denote the sequence of inputs to all the aircraft. The cost function to be minimized is formulated as:

$$J(\sigma, u) = \sum_{i=1}^{N_a} \int_{t^i{}_0}^{t_f^i} L(z^i(t))dt + \sum_{k=1}^{N-1} S(\sigma_k, \sigma_{k+1}). \tag{5.6}$$

The constraints of the optimization are those on the graph mode sequence (5.2), the aircraft edges (5.3), the aircraft state (5.4), (5.5) and the range of allowable aircraft speed. The optimization problem formulated above is a constrained hybrid optimal control problem. There are discrete control inputs consisting of the runway sequence selection, aircraft path and holding pattern assignments, and continuous inputs including the switch times between the runway configurations and the speed assignment along the edge for each aircraft.

In such a general formulation, it is very difficult to encode the state constraints (5.4) and the discrete optimization variables into an optimization solver. Hence, we describe a hierarchical solution approach to simplify the optimization.

5.3 Hierarchical Solution Approach

The hierarchical approach consists of two stages: In the first stage, the optimal runway sequence and the aircraft paths are determined. In the second stage, the optimal switch times and the speed and holding pattern control inputs along the paths for each aircraft are determined. The hierarchical approach is motivated by the fact that weather and wind conditions pose hard constraints on the feasibility of the runway, while this feasibility is not affected by individual aircraft behavior. On the other hand the choice of runway configuration does affects the arrival paths and hence the control of the individual aircraft.

Stage 1

Here, the optimization variables are the runway mode sequence and the sequence of edges that describe the path of each aircraft. These variables are determined by taking into account weather forecast data and established arrival procedures as described below.

Mode sequence determination

Given that it is not feasible to switch configurations frequently, we can always choose the planning horizon small enough, such as one to two hours, so that there is one runway configuration switch. Consequently, we assume there are only two graph modes during the planning horizon $[t_0, t_f]$. The initial condition for the mode is σ_0. Due to wind or traffic demand, the initial mode becomes infeasible and hence a switch to another mode σ_1 is required. The new mode is chosen such that it is feasible with respect to wind, that is, σ_1 satisfies (5.2) for all $t \in [t_s, t_f]$, where t_s is the switch time to be determined. If there are multiple modes that are feasible with respect to wind, the configuration that accommodates the traffic demand is selected. While in this stage the graph mode sequence is determined, the switch time t_s will be determined in Stage 2.

Aircraft path determination

In most airports, the path the aircraft travels prior to landing is chosen based on predefined arrival routes, such as those prescribed in established Standard Arrival Routes (STARs). We use the established procedures to determine the edges that need to be selected for the aircraft path. If with high probability an edge is infeasible due to weather as described in Equation (5.3) the aircraft will be assigned to a new edge which is not blocked due to weather. These new edges could be determined or designed from the forecast data [152, 153]. The remaining

control inputs for the aircraft are the speed and holding patterns along each edge which are determined in Stage 2.

Stage 2

The wind magnitude and direction from the forecast data are uncertain and have low resolution in time, for example, hourly predictions. As such we do not have an exact time at which infeasibility of a runway configuration occurs and there is some flexibility in choosing the switch time between the configurations in order to minimize the overhead in the configuration switch. Let λ denote the first time the infeasibility due to wind based on the forecast is encountered. If the forecast interval is δ_w minutes, we define $\lambda_1 = \lambda - \delta_w$ and $\lambda_2 = \lambda + \delta_w$. We assume that the configuration switch must occur at $t_s \in [\lambda_1, \lambda_2] \subset [t_0, t_f]$. Since t_s affects arrival paths of aircraft and hence individual aircraft control inputs, our objective is to determine t_s along with the aircraft inputs such that the cost of interest is optimized while aircraft separation constraints are satisfied.

In order to impose the state constraints (5.4) and (5.5) we formulate an equivalent characterization of these constraints based on conversion of a separation constraint in terms of distance to a separation constraint in terms of time by integrating aircraft equations of motion.

Separation constraints along edges

Consider the first separation constraint in Equation (5.4). Suppose aircraft i and j fly on an edge $e = (v_1, v_2)$, with aircraft i preceding aircraft j. The aircraft fly with constant speeds of s^i and s^j respectively along the edge. Let $x = x^i - x^j$ and t_1^i, t_1^j be the time at which aircraft i, j depart from node v_1 respectively. Suppose $x(t_1^j) \geq d$, that is, the distance between the two aircraft at the time aircraft j crosses node v_1 is greater than the minimum required distance. Then, in order to ensure separation constraint along the edge, due to constant aircraft speed along the edge it is sufficient to impose $x(t_2^i) \geq d$, where t_2^j denotes the time aircraft j crosses node v_2. By integrating equations of motion of the aircraft, this constraint can be converted to a constraint on the time of crossing node v_2 as $t_2^j \geq t_2^i + \frac{l_e}{s^j}$. For aircraft on neighboring edges, the second constraint in Equation (5.4) must hold. Similar argument shows that separation distance in this case can be converted to a separation requirement for time of crossing the common node of the neighboring edges.

Next, we find bounds on feasible times of arriving at the nodes along the path of each aircraft. Consider aircraft i flying through edge $e = (v_1, v_2)$ with constant speed $s^i \in [s_l^i, s_u^i]$. Let t_1^i and t_2^i denote the time of arrival of aircraft at nodes v_1 and v_2, respectively. Then, $t_2^i \in I_0^i = [\tau_l^i, \tau_u^i]$, where $\tau_l^i = t_1^i + \frac{l_e}{s_u^i}$ and $\tau_u^i = t_1^i + \frac{l_e}{s_l^i}$. In addition, if the aircraft is to

perform a number $n_H \geq 0$ of holding patterns at node v_1, each for a duration of t_{hp}, then $t_2^i \in I = \cup_{k=0}^{n_H} I_k^i$, where $I_k^i = kt_{hp} + [\tau_l^i, \tau_u^i]$. Based on this analysis, we find upper and lower bounds on the arrival time of aircraft at each node along the path of the aircraft. In addition, given an arrival time $t_2^i \in I$ we can uniquely determine the minimum number of holding patterns required at node v_1 and the speed of aircraft along the edge e. Hence, the assignment of speed along edge $e = (v_1, v_2)$ and holding pattern at node v_1 can be converted to the assignment of time of arrival of aircraft at node v_2.

Let the arrival time at a non sink node $v \notin V_s$, for aircraft i be written as $t_v^i \in I_v^i$, where I_v^i may be a union of disjoint intervals due to presence of holding patterns as discussed above. Let $d_t = \frac{l_e}{s_l}$, where s_l is the minimum aircraft speed through node v. For aircraft i and j flying through the same edge or neighboring edges, the separation constraint can be written as $t_v^j \geq t_v^i + d_t$, where t_v is the time of crossing the common node v of the aircrafts' corresponding edges. Let N_v denote the number of aircraft that fly through node v in the planning horizon. We assume aircraft are ordered according to the nominal time of arrival at node v. To account for the first two separation constraints in (5.4) we minimize deviation of arrival time of each aircraft from its nominal arrival time $\bar{t}_v^i$ at this node through formulating the constrained optimization problem:

$$\begin{aligned} \min \quad & \sum_{i=1}^{N_v} t_v^i - \bar{t}_v^i \\ \text{s.t.} \quad & t_v^i \in I_v^i, \quad i = 1, \ldots, N_v, \\ & t_v^i - t_v^{i+1} + d_t \leq 0, \quad i = 1, \ldots, N_v - 1. \end{aligned} \tag{5.7}$$

Optimal switch time and separation constraints at runways

Consider the last constraint in Equation (5.4) which is the runway spacing requirement for aircraft i and j landing at times t_f^i and t_f^j respectively with $t_f^j > t_f^i$. Let s_0^j be the minimum arrival speed of aircraft j and $D_t(a^i, a^j) = \frac{D(a^i, a^j)}{s_0^j}$ denote the runway separation requirement in units of time. This separation requirement can be written as a constraint on the landing times of the two aircraft: $t_f^j \geq t_f^i + D_t(a^i, a^j)$. The switch time between the configurations, t_s, affects the landing runway and consequently the path of the aircraft. Here, we assume that if aircraft i's time of arrival at TRACON entry node is before the configuration switch time then the aircraft takes the final edge $e_0 \in E_s$ to a runway in the first configuration σ_0, and else it takes the final edge $e_1 \in E_s$ to a runway in the new mode σ_1. Since the length of the edges are different, the feasible landing time interval becomes dependent on the switch time. That is, landing time of aircraft i must be inside the set of feasible intervals: $t_f^i \in I^i(t_s) = \cup_{k=0}^{H} kt_{hp} + [\tau_l^i(t_s), \tau_u^i(t_s)]$. In order to determine the optimal switch time and aircraft landing times, we formulate the cost function as the total differences between the nominal landing time $\bar{t}_f^i$ and the actual landing time t_f^i for all

aircraft, that is, $J(t_s, u_f) = \sum_{i=1}^{N_a} t_f^i - \bar{t}_f^i$. Here, u_f denotes the vector of speed and holding pattern assignments for all aircraft along the final edge of their paths. We assume aircraft are ordered according to their nominal landing times. For a given switch time, to determine optimal aircraft input we formulate:

$$\begin{aligned} \min \quad & \sum_{i=1}^{N_a} t_f^i - \bar{t}_f^i \\ \text{s.t.} \quad & t_f^i \in I^i(t_s),\ i = 1, \ldots, N_a, \\ & t_f^i - t_f^{i+1} + D_t(a^i, a^{i+1}) \leq 0,\ i = 1, \ldots, N_a - 1. \end{aligned} \tag{5.8}$$

Note that it is easy to formulate a cost function which would penalize both early and late landing times using $J(t_s, u_f) = \sum_{i=1}^{N_a} |t_f^i - \bar{t}_f^i|$. Let $u_f(t_s)$ denote the optimal input for a given switch time and define $J(t_s) = J(t_s, u_f(t_s))$. Then, the optimal switch time is $t_s^* = \arg\min_{t_s \in [\lambda_1, \lambda_2]} J(t_s)$ and the optimal speed and number of holding patterns on the last edge of the aircraft is given by $u_f(t_s^*)$.

Numerical solution of the optimization problems

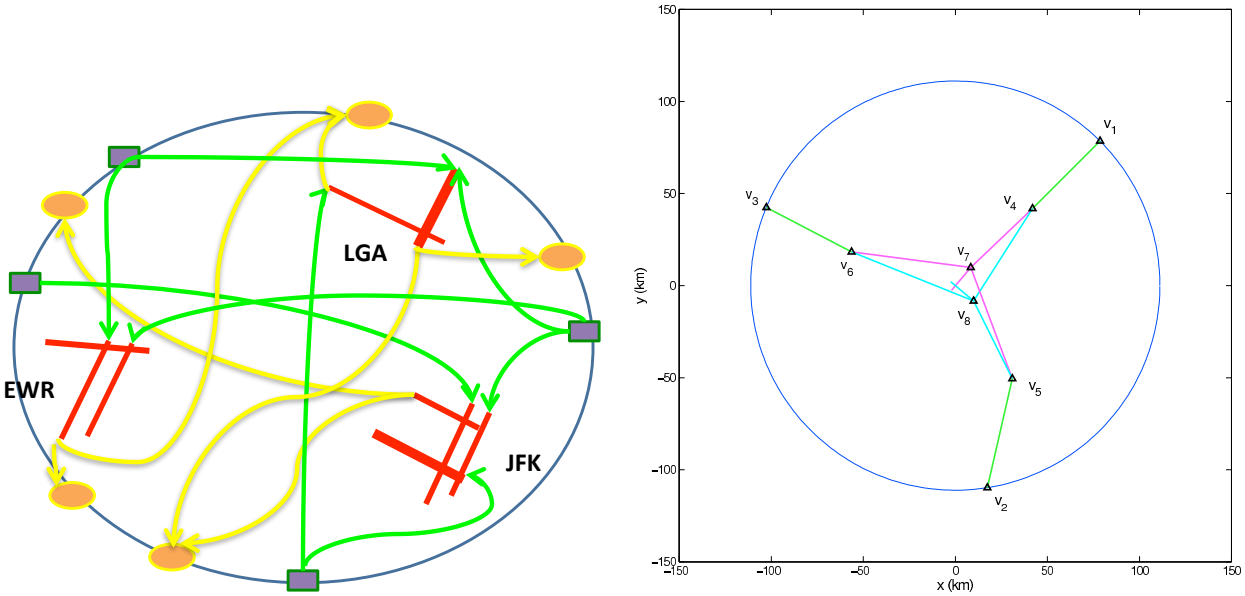
The hybrid optimal control problem has been reduced to a set of optimization problems; Problem (5.7) at each node that is not a runway and Problem (5.8) at the runway nodes. In order to solve these optimization problems, we can formulate a Mixed Integer Linear Program (MILP) as follows [148]: Let $\delta_k \in \{0, 1\}$ for $k = 0, 1, \ldots, H$ denote binary variables for each discrete interval I_k. Then, the constraint $t \in \cup_{k=0}^{H} I_k$ can be equivalently written as $t \in \sum_{k=0}^{H} \delta_k I_k$ with $\sum_{k=0}^{H} \delta_k = 1$. If for any aircraft, the upper and lower bounds on arrival time at a node satisfy $\tau_u - \tau_l \geq t_{hp}$, then, the intervals I_k overlap. In this case, the feasible arrival time becomes one connected interval and this constraint can simply be cast as a Linear Program (LP) without the need for binary variables.

In summary, a MILP, or at best a LP, for each node needs to be solved to find optimal arrival times that satisfy separation constraints. As for landing times, Problem (5.8) is a MILP for a given switch time t_s. Since the allowable range of t_s, $[\lambda_1, \lambda_2]$, is usually small (less than one or two hours) and precisions to order of minutes for determination of the switch time is sufficient, we can discretize $[\lambda_1, \lambda_2]$ and solve a MILP at each discrete value to determine optimal switch time, aircraft landing times and aircraft control inputs along their final edges.

The number of binary variables would be at most $N_a \times H \times V_a$ where N_a is the number of aircraft in the planning horizon and V_a is the maximum number of nodes along each aircraft path. For a typical problem horizon of 2 hours, the number of binary variables would be of order 10^3 which is manageable by current numerical optimization solvers.

5.4 Case Study for JFK Arrival

We consider an instance of optimally planning the John F. Kennedy (JFK) airport runway configurations and aircraft arrival schedules. The airspace graph is derived based on abstraction of the airspace structure of the JFK airport. The aircraft arrival rates and aircraft types are generated according to the counts and probabilities observed in practice.



(a) JFK TRACON with arrival and departure paths

(b) JFK airspace graph

Figure 5.2: Airspace graph model abstraction

Airspace Graph

The JFK TRACON consists of a region of approximately 60 nmi radius centered on the JFK airport and includes many airports in the region, the major ones being LaGuardia (LGA) and Newark (EWR). A representation of the arrival and departure paths and the runway directions for these airports is shown in Figure 5.2(a).

There are a number of Standard Arrival Routes (STARs) leading to the JFK airport. An example of such route from the West direction, referred to as LENDY FIVE Arrival, obtained from the FAA website, is shown in the Figure 5.3[3]. These diagrams describe the routes aircraft take for arrival and locations for assigning holding patterns. For example,

[3] `http://aeronav.faa.gov/index.asp?xml=aeronav/applications/d_tpp`

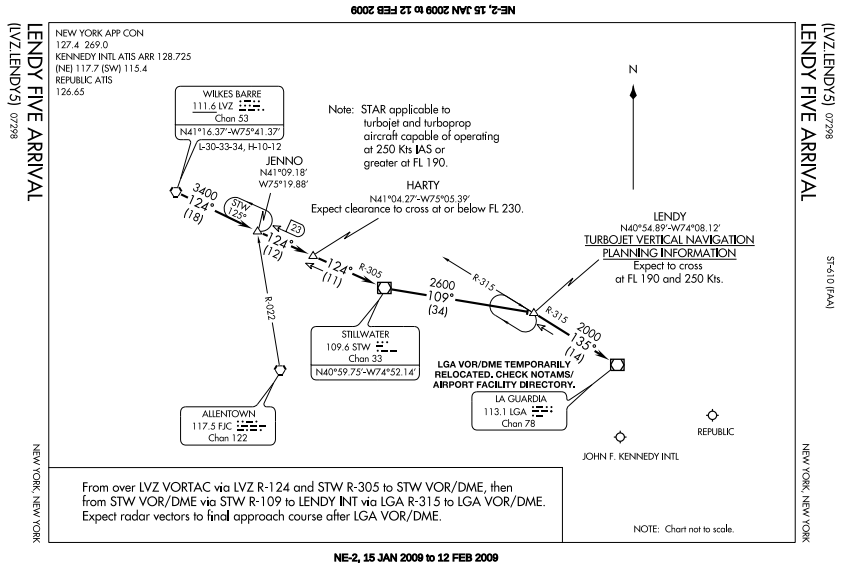


Figure 5.3: JFK Standard Arrival Route LENDY FIVE.

in LENDY FIVE STAR we see that aircraft may enter a holding pattern at JENNO or LENDY meter fixes. Once the aircraft reach the last meter fix in the arrival path, they may follow verbal commands from the air traffic controllers or follow an Instrument Approach Plate (IAP) to make their final descent to a JFK runway as shown in Figure 5.1. There are several possibilities for arrival runway configurations. For this case study, we consider the configuration $\sigma_0 = \{22L, 22R\}$, which is a common arrival configuration in low traffic, and $\sigma_1 = \{31L, 31R\}$, which may be used if landing in σ_0 is not possible due to wind. Based on studying JFK runway configurations, STAR files and our discussions with the air traffic controllers at the JFK airport, we identify three main arrival directions to JFK and create a graph which models the arrival airspace structure. This graph is shown in Figure 5.2(b). In this figure we superimposed the left and right runways in each set of parallel runways for simplicity, that is, the runway set $\{22L, 22R\}$ is shown as one runway and the runway set $\{31L, 31R\}$ is shown as one runway. The entry nodes are $\{v_1, v_2, v_3\}$ corresponding to three major entry meter fixes. Two of the sink nodes v_7 and v_8, corresponding to runways $\{22L, 22R\}$ and $\{31L, 31R\}$ respectively, are shown. The paths from each entry meter fix to each of the two runway configurations are depicted.

Data for Scenario Set-up

We use the ASPM[4] database in order to instantiate wind forecast data and aircraft arrival times. The day under consideration is 07/01/2009 and the time interval is $[19.00, 24.00]$

[4]Aviation Systems Performance Metric: http://www.aspm.faa.gov

during which high wind magnitudes were recorded. The data in ASPM is recorded at 15-minute intervals. Although the wind data is the recordings of actual wind magnitude and direction, we treat this as an uncertain wind forecast for our problem. Note that the data is used as a guideline for setting up the simulation rather than for comparison of performance of our algorithm with current procedures.

The aircraft arrival times at the entry meter fixes are generated randomly but with the number of arrivals in each 15-minute interval set according to the ASPM arrival counts. The probability of arrival of aircraft types $\{\text{Heavy}, \text{B757}, \text{Large}, \text{Small}\}$ and the landing speed of these aircraft types were set to $\{0.390, 0.066, 0.179, 0.365\}$ and $\{150, 130, 130, 90\}$ knots respectively [156]. The aircraft were assigned an entry node randomly, with equal probability for each entry node. The required runway separation distance in minutes is shown in Table 5.1 and is derived based on the data on required separation distance in nautical miles and the average landing speed of aircraft [156]. In this table, the leading/trailing aircraft are represented by rows/columns respectively.

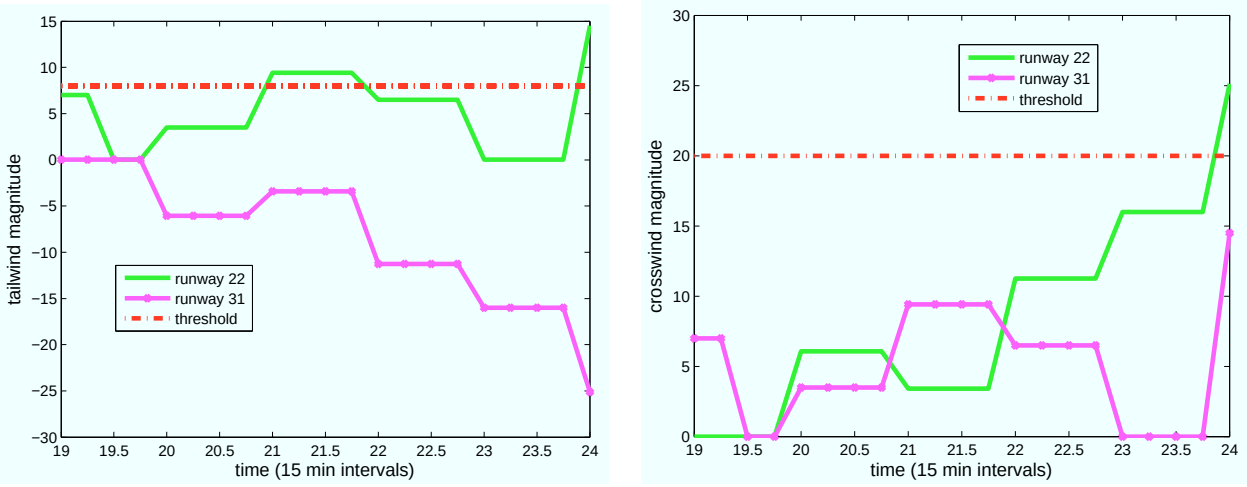
	Heavy	B757	Large	Small
Heavy	1.60	2.31	2.31	4.00
B757	1.60	1.85	1.85	3.33
Large	1.00	1.15	1.15	2.67
Small	1.00	1.15	1.15	1.67

Table 5.1: Runway separation requirement in minutes

Stage 1 optimization

We computed the crosswind and tailwind for the two modes, $\sigma_0 = \{22L, 22R\}$ and $\sigma_1 = \{31L, 31R\}$ as shown in Figure 5.4. The crosswind and tailwind thresholds were set to 20 and 8 knots respectively. The initial mode is σ_0. This mode becomes infeasible due to large tailwind at approximately hour 21.00, while runway configuration σ_1 remains feasible. Due to consideration of uncertainty in wind data, we choose the range of allowable switch time as $[\lambda_1, \lambda_2] = [20.30, 21.30]$. We aim to choose the switch time in this interval so that aircraft delay is minimized. We consider all aircraft in the JFK airspace in a two-hour planning horizon of $[20.00, 22.00]$ hour.

Aircraft paths were set based on the airspace graph and the entry nodes of the aircraft. The paths to the two runway configurations under study from each entry node are shown in Figure 5.2(b). For example, an aircraft arriving at entry node v_1 would take $\{(v_1, v_4), (v_4, v_7)\}$ to land on either $22L$ or $22R$, and would take $\{(v_1, v_4), (v_4, v_8)\}$ to land on either $31L$ or $31R$.



(a) Tailwind (b) Crosswind

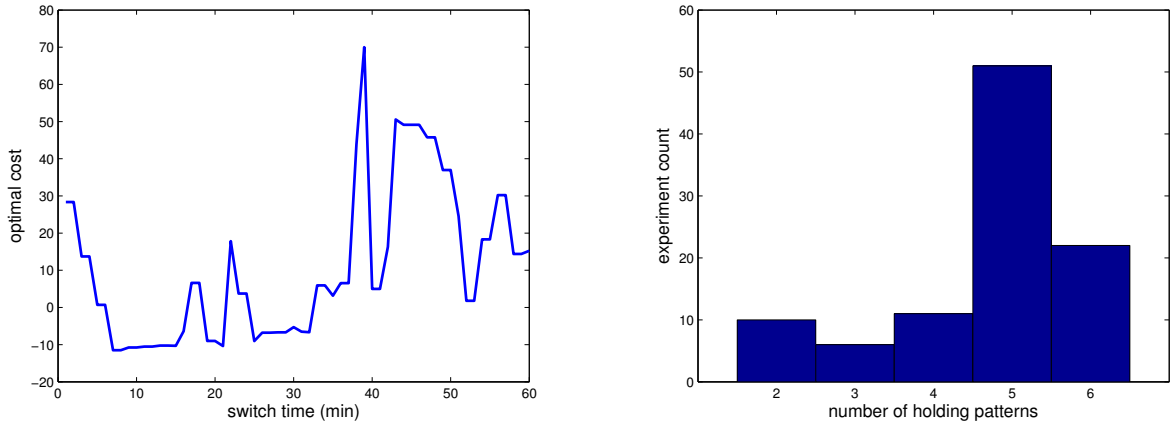
Figure 5.4: Wind impact on runway configurations

In order to determine the speed and holding pattern on the edges along the path for each aircraft and the optimal switch time we used Stage 2 solution approach.

Stage 2 optimization

Problems (5.7) was solved for each of the nodes v_4, v_5, v_6, to determine arrival times at these nodes such that aircraft separation constraint is maintained along the first three edges (v_1, v_4), $(v_2, v_5), (v_3, v_6)$. The spacing requirement at these nodes was set to 2 minutes which results in a separation distance greater than 5 nmi in the 2D plane. Then, Problem (5.8) was solved for each switch time in the interval $[20.30, 21.30]$ in order to find the optimal switch time and to determine the optimal speed and holding pattern assignment for each aircraft along its final edge.

The parameters were set as follows: The upper and lower bounds on speed of each aircraft were set to 12% above and below the nominal speed of the aircraft along an edge, respectively. The nominal speed was determined based on aircraft type and its distance from the runway. Each holding pattern had a duration of 3 minutes [148] and the maximum number of holding patterns at each node for each aircraft was set to $H = 2$. Hence, for aircraft i at each of the two nodes along its path (excluding the runway node), there were 3 binary variables δ_k^i, $k = 0, 1, 2$ associated with zero, one and two holding patterns. Additionally, to minimize the number of holding patterns, we penalized each holding pattern by including a cost term $w(\delta_1^i + \delta_2^i)$, with $w > 0$ a weight which was set to 10 in the following simulation.



(a) Cost as a function of switch time

(b) Holding patterns for each experiment

Figure 5.5: Optimal cost function and its sensitivity

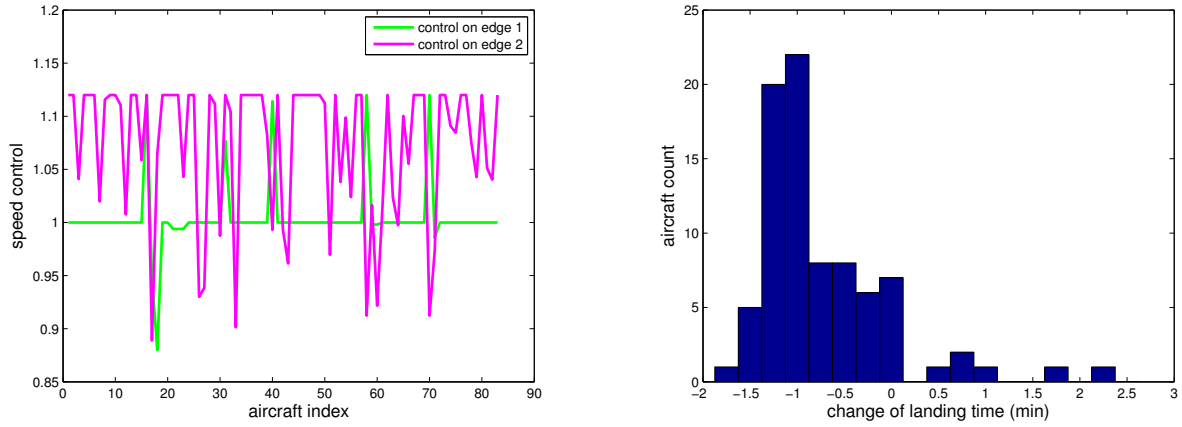
Simulation Results

The result of Stage 2 optimization for an instance of randomly generated arrival data is summarized as follows: In the 2-hour planning horizon there were 85 aircraft in the JFK airspace. We used CPLEX optimization software package to solve Problem (5.7) for the arrival times at nodes v_4, v_5, v_6 and then used CPLEX to solve Problem (5.8) by discretizing the time interval $[20.30, 21.30]$ into 60 minutes and solving a MILP for each of the 60 possible switch times to determine the one with the lowest cost function.

The optimized cost $J(t_s)$, which is the total deviations from the nominal landing time summed with the cost of holding patterns, as a function of the switch time t_s (shown here in the interval $[0, 60]$, with 0 indicating 20.30) is plotted in Figure 5.5(a). From this computation, we find the optimal switch time to be $t_s^* = 20.37$. For this optimal solution 5 aircraft are put on hold at node v_4, each for one holding pattern. The large variations of the cost with respect to switch time indicates that by properly choosing the switch time and optimizing aircraft inputs based on the anticipated switch time, delays can be reduced significantly.

We note that the ASPM database had recorded a switch time of 23.00 hour. However, the actual configuration switch may have occurred earlier, as the ASPM runway configuration data is written manually and is known to have delays or errors. In addition, given that different aircraft types can operate under slightly different tailwind thresholds, it is possible that during the time horizon of interest, the tailwind threshold used in practice was larger than the one we determined.

Given that in practice the aircraft may deviate from their nominal arrival times at a meter fix,



(a) Aircraft speed input

(b) Delay or early arrival of each aircraft

Figure 5.6: Aircraft optimized control inputs and arrival times

we did a sensitivity analysis of our results. We computed the optimal switch time associated with perturbing the initial arrival times of the aircraft at the entry meter fixes by a uniform random distribution of $U[-0.5, 0.5]$ minutes. 81% of the experiments had an optimal switch time in the interval $[20.35, 20.38]$. There were a few experiments in which the switch time was significantly different. This could be due to the particular separation of the arrival times of the aircraft. Despite the differences in the optimal switch time, the total savings at the optimal switch time remained at least 10 minutes in all experiments. The histogram of the number of holding patterns for these experiments is shown in Figure 5.5(b) and indicates that the number of holding patterns remained less than 6 in all experiments.

For the optimal switch time in the nominal experiment, the fraction of decrease or increase of the speed of each aircraft from its nominal value along the first and second edges of aircraft path are shown in Figure 5.6(a). As can be seen, most speed control commands are assigned on the last edges of the aircraft path. This is due to the aircraft spacing requirements at the landing runways. The histogram of the changes of landing times from the nominal landing times is shown in Figure 5.6(b). At the optimal switch time, 9 out of 83 aircraft were delayed with a maximum delay of less than 2.25 minutes.

In terms of computation complexity, the number of binary variables in the CPLEX problem was equal to $N_a \times H \times V_a$ where $N_a = 85$ was the number of aircraft, $H = 3$ and $V_a = 2$ were three possibilities for holding pattern at each of the 2 nodes along the path of the aircraft. The average running time for the CPLEX solver was 0.99 seconds on a processor with 2.66 GHz processing speed and 4 GB memory and running with MATLAB interface.

5.5 Conclusions

We developed a hierarchical approach to plan a runway configuration which is safe with respect to wind, and to optimize the switch time between the runway configurations as well as the traffic control strategies during the transitional period of a configuration switch in order to minimize the overall traffic delays. The framework is based on deriving an accurate model of arrival airspace structure and is consistent with the TRACON and Tower Control procedures commonly used. We illustrated our modeling and solution approach with a case study inspired by the JFK airspace geometry, arrival and wind data. The case study shows the use of the framework in real scenarios and the benefits gained by optimizing aircraft arrival during transitional periods of a configuration switch.

In our formulation, for simplicity, various control strategies, such as path stretching or Vector For Spacing, were not considered. We expect the number of holding patterns to decrease if additional control strategies are taken into account. In addition, currently the optimization problems solved at different nodes along the aircraft path are decoupled and hence aircraft cannot anticipate large separation requirements that may arise in the downstream edges of their path. We are currently extending the solution approach to account for all the nodes along the path simultaneously. There are several other considerations we have to address before implementing the proposed modeling and solution approach. First, we need to test our approach with more realistic air traffic scenarios, with the use of historical data for aircraft arrival times and the consideration of more complex airspaces which include multiple airports. In addition, we need to formulate and solve the algorithm within a receding horizon optimization framework, such that the optimization is repeated as aircraft or weather forecast data is updated.

Finally, we note that the problem of configuration planning combines interesting aspects of human and automation. While in theory it is not too difficult to determine the set of runways that are safe with respect to wind conditions and switch runways to best accommodate traffic conditions, there are considerations in configuration switch due to human operators. For example, the air traffic controllers at the airport tower have a mental model of the arrival traffic routes. As a runway switch occurs, the normal traffic patterns get disturbed. Usually the air traffic controllers require all traffic in the old routes to land before they start the operation in the new runways. Although an automation would not face this problem, if this automation is to be an aid for humans, and for transition to automated system to occur smoothly, such problems should be considered.

Chapter 6

Conclusions and Future Work

In this thesis, we developed algorithms for optimal control of deterministic and stochastic hybrid systems. We also developed a framework for optimizing arrival traffic and runway configuration planning. In the deterministic setting, our algorithm was used to plan aircraft trajectories that are safe with respect to hazardous weather, while minimizing fuel consumption and flight time by considering wind as a deterministic known disturbance acting on the system. Due to the presence of various sources of uncertainty in air traffic operations, we introduced a stochastic hybrid game model. This model allowed for the presence of both stochastic uncertainties arising from nature, such as wind and hazardous storms, and deterministic uncertainty arising due to the presence of other agents, such as nearby aircraft. Constraint satisfaction and target reachability in this formulation became probabilistic problems. Our proposed algorithm addressed maximizing the probability of target attainability while satisfying constraints. This algorithm was applied to aircraft conflict detection in the presence of stochastic wind and maximally safe trajectory planning in the presence of stochastic hazardous weather. The case studies in this thesis illustrated the potential of our algorithms for addressing the TBO concept of the NextGen air transportation system, which consists of taking into account individual aircraft objectives while ensuring safety.

As illustrated in Chapter 5, one of the main challenges in implementing the TBO concept in realistic air traffic scenarios is the large-scale nature of the air traffic system. Individual aircraft behavior must take into account global objectives such as capacity and spacing requirements in various airspaces. For example, while it is known that a continuous descent is the most fuel efficient aircraft trajectory, implementation of a continuous descent in dense arrival regions is not possible due to the need to maneuver aircraft for collision avoidance and landing separation [158]. This challenge motivates several directions of future work.

In the deterministic setting, while our proposed two-stage algorithm can improve individual aircraft operations, the computation time of the nonlinear programming optimization problems formulated in the algorithm do not scale well as the state dimension increases. Thus,

to address realistic air traffic scenarios we are exploring efficient numerical implementations of our algorithm through various discretization schemes, such as pseudo-spectral methods, as well as various nonlinear programming solvers. Additionally, given advances in the field of mixed integer nonlinear programming, we are working on formulating the hybrid optimal control problem as a mixed integer optimal control problem and combining Branch & Bound techniques with nonlinear programming to find the discrete and continuous components of the hybrid optimal control input.

In the stochastic setting, our proposed dynamic programming algorithm enables quantifying and maximizing safety probability of aircraft trajectories in the presence of uncertain forecast data. This algorithm suffers from the curse of dimensionality and thus it requires an efficient implementation in order to apply the algorithm online to realistic air traffic settings. We are exploring how to adapt fast numerical methods in solving the deterministic Hamilton-Jacobi equations such as Fast Marching methods [159].

To apply our algorithms in a realistic setting, a receding horizon implementation would be required so that the safety verification and trajectory planning are repeated as updates about the forecast and nearby aircraft states are provided. Thus, analysis of constraint satisfaction in a moving horizon and proper choice of the look-ahead horizon become important so that dangerous events can be accounted for early enough to be able to plan a safe maneuver. While in the deterministic case the constraint satisfaction (persistent feasibility) could potentially be addressed by introducing final state constraints, in the stochastic case the receding horizon formulation and solution are open problems that we would like to explore.

The implementation of the ADS-B inter aircraft communication networks and System Wide Information Management for sharing information amongst aircraft would enable semi decentralized control of aircraft, such that air traffic controllers would only need to intervene if a safety critical scenario occurs. Thus, we aim to develop a decentralized implementation of our algorithms. However, as technological changes occur gradually, our optimization framework needs to account for aircraft not equipped with new technologies and it must be safe with respect to failures in communication or control. Another area of work is the inclusion of multi-objective optimization methods in order to plan safe aircraft trajectories while taking into account efficiency objectives or objectives of a number of different aircraft [160].

Finally, the techniques developed in this thesis are important and useful for other safety- and efficiency-critical application domains. For example, in smart grid applications, the energy input of various power sources such as wind and solar energy, as well as the demand on the grid, are predicted and have uncertainty. Our stochastic verification and control methods have the potential to provide a systematic way of ensuring safety and optimal performance. To address such important problems, the next steps would be defining a mathematical model abstraction of the application which would be amenable to our analysis tools, and extending our tools to account for large scale operations of realistic applications.

Bibliography

[1] G. Gilbert, "Historical development of the air traffic control system," *IEEE Transactions on Communications*, vol. 21, no. 5, pp. 364–375, 1973.

[2] SESAR Joint Undertaking, "SESAR (Single European Sky ATM Research)," 2007. [Online]. Available: http://www.sesarju.eu/

[3] Joint Planning and Development Office, "NEXTGEN. Concept of Operations for the Next Generation Air Transportation System, Version 2.0," 2007. [Online]. Available: http://www.jpdo.gov/library/nextgen_v2.0.pdf

[4] W. Zhang, M. Kamgarpour, D. Sun, and C. Tomlin, "A Hierarchical Flight Planning Framework for Air Traffic Management," *Proceedings of the IEEE*, 2011, to appear.

[5] C. Tomlin, G. Pappas, and S. Sastry, "Conflict resolution for air traffic management: A study in multiagent hybrid systems," *IEEE Transactions on Automatic Control*, vol. 43, no. 4, pp. 509–521, 2002.

[6] W. Glover and J. Lygeros, "A stochastic hybrid model for air traffic control simulation," in *Hybrid Systems: Computation and Control*, ser. Lecture Notes in Computer Science, R. Alur and G. J. Pappas, Eds., 2004, vol. 2993, pp. 372–386.

[7] I. Ross and C. D'Souza, "Hybrid optimal control framework for mission planning," *Journal of Guidance, Control and Dynamics*, vol. 28, no. 4, p. 686, 2005.

[8] S. Sastry, G. Meyer, C. Tomlin, J. Lygeros, D. Godbole, and G. Pappas, "Hybrid control in air traffic management systems," in *IEEE Conference on Decision and Control*, vol. 2, 1995, pp. 1478–1483.

[9] M. Soler, A. Olivares, and E. Staffetti, "Hybrid Optimal Control Approach to Commercial Aircraft Trajectory Planning," *Journal of Guidance, Control and Dynamics*, vol. 33, no. 3, pp. 985–991, 2010.

[10] J. Gillula, G. Hoffmann, H. Huang, M. Vitus, and C. Tomlin, "Applications of hybrid reachability analysis to robotic aerial vehicles," *International Journal of Robotics Research*, vol. 30, no. 3, pp. 335–354, 2011.

[11] E. Frazzoli, M. Dahleh, and E. Feron, "Robust hybrid control for autonomous vehicle motion planning," in *IEEE Conference on Decision and Control*, 2000, pp. 821–826.

[12] A. Balluchi, L. Benvenuti, M. Di Benedetto, C. Pinello, and A. Sangiovanni-Vincentelli, "Automotive engine control and hybrid systems: Challenges and opportunities," *Proceedings of the IEEE*, vol. 88, no. 7, pp. 888–912, 2000.

[13] J. de Sousa, A. Girard, J. Hedrick, and F. Kretz, "Real-time hybrid control of mobile offshore base scaled models," in *Proceedings of American Control Conference*, vol. 1, no. 6, 2000, pp. 682–686.

[14] C. Cassandras, D. Pepyne, and Y. Wardi, "Optimal control of a class of hybrid systems," *IEEE Transactions on Automatic Control*, vol. 46, no. 3, pp. 398–415, 2001.

[15] R. Alur, C. Belta, F. Ivančić, V. Kumar, M. Mintz, G. Pappas, H. Rubin, and J. Schug, "Hybrid modeling and simulation of biomolecular networks," in *Hybrid Systems: Computation and Control*, ser. Lecture Notes in Computer Science, M. Di Benedetto and A. Sangiovanni-Vincentelli, Eds. Springer, 2001, vol. 2034, pp. 19–32.

[16] R. Ghosh and C. Tomlin, "Symbolic reachable set computation of piecewise affine hybrid automata and its application to biological modelling: Delta-Notch protein signalling," *Systems Biology*, vol. 1, no. 1, pp. 170–183, 2004.

[17] P. Lincoln and A. Tiwari, "Symbolic systems biology: Hybrid modeling and analysis of biological networks," in *Hybrid Systems: Computation and Control*, ser. Lecture Notes in Computer Science, R. Alur and G. Pappas, Eds. Springer, 2004, pp. 147–165.

[18] A. Ames, R. Sinnet, and E. Wendel, "Three-dimensional kneed bipedal walking: A hybrid geometric approach," in *Hybrid Systems: Computation and Control*, ser. Lecture Notes in Computer Science, R. Majumdar and P. Tabuada, Eds. Springer, 2009, pp. 16–30.

[19] R. Alur, C. Courcoubetis, T. Henzinger, and P. Ho, "Hybrid automata: An algorithmic approach to the specification and verification of hybrid systems," in *Hybrid Systems: Computation and Control*, ser. Lecture Notes in Computer Science, R. L. Grossman, A. Nerode, A. P. Ravn, and H. Rischel, Eds. Springer-Verlag, 1993, pp. 209–229.

[20] T. Henzinger, "The Theory of Hybrid Automata," in *Annual Symposium on Logic in Computer Science*, jul 1996, pp. 278 –292.

[21] A. Puri and P. Varaiya, "Decidability of hybrid systems with rectangular differential inclusions," in *Computer Aided Verification*, ser. Lecture Notes in Computer Science. Springer, 1994, vol. 818, pp. 95–104.

[22] O. Maler, A. Pnueli, and J. Sifakis, "On the synthesis of discrete controllers for timed systems," in *Annual Symposium on Theoretical Aspects of Computer Science*, ser. Lecture Notes in Computer Science, 1995, pp. 229–242.

[23] P. Tabuada, *Verification and control of hybrid systems: a symbolic approach*. Springer-Verlag New York Inc, 2009.

[24] M. Branicky, "Multiple Lyapunov functions and other analysis tools for switched and hybrid systems," *IEEE Transactions on Automatic Control*, vol. 43, no. 4, pp. 475–482, 1998.

[25] H. J. Sussmann, "A maximum principle for hybrid optimal control problems," in *IEEE Conference on Decision and Control*, 1999, pp. 425–430.

[26] M. S. Branicky, V. S. Borkar, and S. K. Mitter, "A unified framework for hybrid control: model and optimal control theory," *IEEE Transactions on Automatic Control*, vol. 43, no. 1, pp. 31–45, 1998.

[27] M. S. Shaikh and P. E. Caines, "On the optimal control of hybrid systems: Optimization of trajectories, switching times, and location schedules," *Lecture Notes in Computer Science*, vol. 2623, pp. 466–481, 2003.

[28] L. S. Pontryagin, V. G. Boltyanskii, R. V. Gamkrelidze, and E. Mishchenko, *The mathematical theory of optimal processes (International series of monographs in pure and applied mathematics)*. Interscience Publishers, 1962.

[29] R. E. Bellman, *Dynamic Programming*. Princeton University Press, 1957.

[30] R. Kalman, in *Mathematical optimization techniques*, R. Bellman, Ed. University of California Press, 1963, ch. The theory of optimal control and the calculus of variations, pp. 309–331.

[31] R. Isaacs, *Differential games: A mathematical theory with applications to warfare and pursuit, control and optimization*. Wiley (New York), 1965.

[32] J. Hu, J. Lygeros, and S. Sastry, "Towards a theory of stochastic hybrid systems," in *Hybrid Systems: Computation and Control*, ser. Lecture Notes in Computer Science, N. A. Lynch and B. H. Krogh, Eds. Springer, 2000, vol. 1790, pp. 160–173.

[33] J. P. Hespanha, "Stochastic hybrid systems: Application to communication networks," in *Hybrid Systems: Computation and Control*, ser. Lecture Notes in Computer Science, R. Alur and G. J. Pappas, Eds. Springer, 2004, vol. 2993, pp. 47–56.

[34] J. Hespanha and A. Singh, "Stochastic models for chemically reacting systems using polynomial stochastic hybrid systems," *International Journal of Robust and nonlinear control*, vol. 15, no. 15, pp. 669–689, 2005.

[35] M. Davis, "Piecewise-deterministic Markov processes: A general class of non-diffusion stochastic models," *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 353–388, 1984.

[36] O. V. Costa, M. D. Fragoso, and R. Marques, *Discrete-time Markov jump linear systems*. Springer Verlag, 2005.

[37] M. Ghosh, A. Arapostathis, and S. Marcus, "Ergodic control of switching diffusions," *SIAM Journal on Control and Optimization*, vol. 35, no. 6, pp. 1952–1988, 1997.

[38] M. Bujorianu, "Extended stochastic hybrid systems and their reachability problem," in *Hybrid Systems: Computation and Control*, ser. Lecture Notes in Computer Science, R. Alur and G. J. Pappas, Eds. Springer, 2004, pp. 234–249.

[39] M. Bujorianu and J. Lygeros, "General stochastic hybrid systems: Modelling and optimal control," in *IEEE Conference on Decision and Control*, vol. 2, 2004, pp. 1872–1877.

[40] J. Lygeros, C. Tomlin, and S. Sastry, "Controllers for reachability specifications for hybrid systems," *Automatica*, vol. 35, pp. 349–370, 1999.

[41] J. Lygeros, "On reachability and minimum cost optimal control," *Automatica*, vol. 40, no. 6, pp. 917–927, 2004.

[42] C. Tomlin, I. Mitchell, A. M. Bayen, and M. Oishi, "Computational techniques for the verification of hybrid systems," in *Proceedings of the IEEE*, no. 7, Jul. 2002, pp. 986–1001.

[43] K. Margellos and J. Lygeros, "Hamilton-Jacobi formulation for Reach-Avoid Differential Games," *IEEE Transactions on Automatic Control*, sep 2011, to appear.

[44] J. Katoen, "Stochastic model checking," in *Stochastic Hybrid Systems*, ser. 24, C. Cassandras and J. Lygeros, Eds. CRC Press, 2006, pp. 79–106.

[45] M. Prandini and J. Hu, "A stochastic approximation method for reachability computations," in *Stochastic Hybrid Systems*, H. A. Blom and J. Lygeros, Eds. Springer, 2006, pp. 107–139.

[46] A. Abate, M. Prandini, J. Lygeros, and S. Sastry, "Probabilistic reachability and safety for controlled discrete time stochastic hybrid systems," *Automatica*, vol. 44, no. 11, pp. 2724 – 2734, 2008.

[47] S. Summers and J. Lygeros, "Verification of discrete time stochastic hybrid systems: A stochastic reach-avoid decision problem," *Automatica*, vol. 46, no. 12, pp. 1951 – 1961, 2010.

[48] M. Kamgarpour and C. Tomlin, "Optimal control of switched systems," in *SIAM Conference on Optimization and Control*, Denver, Colorado, 2009.

[49] H. González, R. Vasudevan, M. Kamgarpour, S. S. Sastry, R. Bajcsy, and C. J. Tomlin, "A descent algorithm for the optimal control of constrained nonlinear switched dynamical systems," in *Hybrid Systems: Computation and Control*, ser. Lecture Notes in Computer Science, K. H. Johansson and W. Yi, Eds., 2010, pp. 51–60.

[50] H. Gonzalez, R. Vasudevan, M. Kamgarpour, S. Sastry, R. Bajcsy, and C. Tomlin., "A Numerical Method for the Optimal Control of Switched Systems," in *IEEE Conference on Decision and Control*, 2010, pp. 7519–7526.

[51] M. Kamgarpour and C. Tomlin, "On Optimal Control of Non-Autonomous Switched Systems with a Fixed Mode Sequence," *Automatica*, 2011, to appear.

[52] M. Kamgarpour, M. Soler, C. Tomlin, A. Olivares, and J. Lygeros, "Hybrid optimal control for aircraft trajectory design with a variable sequence of modes," in *Proceedings of IFAC World Congress*, Aug. 2011, pp. 7238–7243.

[53] M. Kamgarpour, J. Ding, S. Summers, , A. Abate, J. Lygeros, and C. Tomlin, "Discrete Time Stochastic Hybrid Dynamic Games: Verification & Controller Synthesis," in *IEEE Conference on Decision and Control*, Dec 2011, to appear.

[54] ——, "Discrete Time Stochastic Hybrid Dynamic Games: Verification & Controller Synthesis," *IEEE Transactions on Automatic Control*, 2011, in preparation.

[55] M. Kamgarpour, V. Dadok, and C. Tomlin, "Trajectory Generation for Multiple Aircraft Subject to Dynamic Weather Uncertainty," in *IEEE Conference on Decision and Control*, 2010, pp. 2063–2068.

[56] S. Summers, M. Kamgarpour, J. Lygeros, and C. Tomlin, "A Stochastic Reach-Avoid Problem with Random Obstacles," in *Hybrid Systems: Computation and Control*, E. Frazzoli and R. Grosu, Eds. ACM, 2011, pp. 251–260.

[57] S. Summers, M. Kamgarpour, C. Tomlin, and J. Lygeros, "Verification of Stochastic Hybrid Systems with Stochastic Sets," *Automatica*, 2011, in preparation.

[58] M. Kamgarpour, W. Zhang, and C. Tomlin, "Modeling and Optimization of Air Traffic in Terminal Airspace," in *Transportation Research Forum, Long Beach, CA*, mar 2011.

[59] ——, "Modeling and Optimization of Terminal Airspace and Aircraft Arrival subject to Weather Uncertainties," in *Proceedings of AIAA Guidance, Navigation and Control Conference*, aug 2011.

[60] M. Rinehart, M. Dahleh, D. Reed, and I. Kolamnovsky, "Suboptimal control of switched systems with an application to the disc engine," *IEEE Transactions on Control Systems Technology*, vol. 16, no. 2, pp. 189–201, Mar. 2005.

[61] R. Johansson and A. Rantzer, *Nonlinear and hybrid systems in automotive control*. Springer, 2003.

[62] P. Howlett, P. Pudney, and X. Vu, "Local energy minimization in optimal train control," *Automatica*, vol. 45, no. 11, pp. 2692–2698, 2009.

[63] T. Geyer, G. Papafotiou, and M. Morari, "On the optimal control of switch-mode DC-DC converters," in *Hybrid Systems: Computation and Control*, ser. Lecture Notes in Computer Science, R. Alur and G. J. Pappas, Eds. Springer, 2004, pp. 77–85.

[64] R. Loxton, K. Teo, V. Rehbock, and W. Ling, "Optimal switching instants for a switched-capacitor DC/DC power converter," *Automatica*, vol. 45, no. 4, pp. 973–980, 2009.

[65] M. Kamgarpour and C. Tomlin, "Modeling and analysis of cell differentiation using hybrid systems," in *Proceedings of the American Control Conference*, 2010.

[66] T. I. Seidman, "Optimal Control for Switching Systems," in *Annual Conference on Information Science and Systems*, 1987, pp. 485–489.

[67] B. Lincoln and A. Rantzer, "Relaxing Dynamic Programming," *IEEE Transactions on Automatic Control*, no. 8, Aug. 2006.

[68] A. Bemporad and M. Morari, "Control of systems integrating logic, dynamics, and constraints," *Automatica*, vol. 35, pp. 407–428, 1999.

[69] F. Borrelli, M. Baotić, A. Bemporad, and M. Morari, "Dynamic programming for constrained optimal control of discrete-time linear hybrid systems," *Automatica*, vol. 41, pp. 1709–1721, Jun. 2005.

[70] C. Seatzu, D. Corona, A. Giua, and A. Bemporad, "Optimal control of continuous-time switched affine systems," *IEEE Transactions on Automatic Control*, vol. 51, pp. 726–741, 2006.

[71] M. Alamir and S.-A. Attia, "An efficient algorithm to solve optimal control problems for nonlinear switched hybrid systems," in *Proceedings of IFAC symposium*, Sep. 2004, pp. 417–422.

[72] J. Chudoung and C. Beck, "The Minimum Principle for deterministic impulsive control systems," in *IEEE Conference on Decision and Control*, Dec. 2001, pp. 3569– 3574.

[73] X. Xu and P. J. Antsaklis, "Optimal control of switched systems via nonlinear optimization based on direct differentiations of value functions," *International Journal of Control*, no. 8, pp. 1406–1426, Nov. 2002.

[74] X. Xu and P. Antsaklis, "Results and perspectives on computational methods for optimal control of switched systems," in *Hybrid Systems: Computation and Control*, ser. Lecture Notes in Computer Science, O. Maler and A. Pnueli, Eds. Springer, 2003, vol. 2623, pp. 540–555.

[75] H. Axelsson, Y. Wardi, M. Egerstedt, and E. Verriest, "Gradient Descent Approach to Optimal Mode Scheduling in Hybrid Dynamical Systems," *Journal of Optimization Theory and Applications*, vol. 136, no. 2, pp. 167–186, 2008.

[76] R. Vinter, *Optimal control, Systems and Control: Foundations and Applications.* Birkhauser Boston Inc., Boston, MA, 2000.

[77] H. J. Sussmann and J. C. Willems, "300 years of optimal control: from the Brachystochrone to the maximum principle," *Control Systems Magazine, IEEE*, vol. 17, pp. 32–44, 1997.

[78] R. Hartl, S. Sethi, and R. Vickson, "A survey of the maximum principles for optimal control problems with state constraints," *SIAM review*, vol. 37, no. 2, pp. 181–218, 1995.

[79] A. E. Bryson, *Dynamic Optimization.* Addison Wesley Longman Inc., 1999.

[80] M. Crandall and P. Lions, "Viscosity solutions of Hamilton-Jacobi equations," *Transactions of the American Mathematical Society*, pp. 1–42, 1983.

[81] M. Bardi and I. Dolcetta, *Optimal control and viscosity solutions of Hamilton-Jacobi-Bellman equations.* Springer, 1997.

[82] S. Hedlund and A. Rantzer, "Optimal control of hybrid systems," in *IEEE Conference on Decision and Control*, Dec. 1999, pp. 3972–3977.

[83] X. Xu and P. Antsaklis, "Optimal control of switched systems via nonlinear optimization based on direct differentiations of value functions," *International Journal of Control*, vol. 75, no. 8, pp. 1406–1426, Nov. 2002.

[84] E. Khmelnitsky, "A combinatorial, graph-based solution method for a class of continuous-time optimal control problems," *Mathematics of Operations Research*, pp. 312–325, 2002.

[85] X. Xu and P. Antsaklis, "Optimal control of switched systems based on parametrization of the switching instants," *IEEE Transactions on Automatic Control*, vol. 49, pp. 2–16, 2004.

[86] M. Egerstedt, Y. Wardi, and F. Delmotte, "Optimal control of switching times in switched dynamical systems," in *IEEE Conference on Decision and Control*, Dec. 2003, pp. 2138– 2143.

[87] R. Loxton, K. Teo, and V. Rehbock, "Computational method for a class of switched system optimal control problems," *IEEE Transactions on Automatic Control*, vol. 54, no. 10, pp. 2455–2460, 2009.

[88] E. Polak, *Optimization: Algorithms and Consistent Approximations*. Springer, 1997.

[89] O. von Stryk and R. Bulirsch, "Direct and indirect methods for trajectory optimization," *Annals of Operations Research*, vol. 37, no. 1, pp. 357–373, 1992.

[90] C. R. Hargraves and S. W. Paris, "Direct trajectory optimization using nonlinear programming and collocation," *Journal of Guidance, Control, and Dynamics*, vol. 10, no. 4, pp. 338–342, 1987.

[91] A. L. Herman and B. A. Conway, "Direct optimization using collocation based on high order Gauss-Lobatto quadrature rules," *Journal of Guidance, Control, and Dynamics*, vol. 19, no. 3, pp. 592–599, 1996.

[92] J. T. Betts, *Practical Methods for Optimal Control Using Nonlinear Programming*. SIAM, 2001.

[93] F. M. Callier and C. A. Desoer, *Linear System Theory*. Springer-Verlag, 1991.

[94] L. Armijo, "Minimization of functions having Lipschitz continuous first partial derivatives," *Pacific Journal of Mathematics*, vol. 16, 1996.

[95] G. Huntington and A. Rao, "Optimal Configuration of Spacecraft Formations via a Gauss Pseudospectral Method," *Advances in the Astronautical Sciences*, vol. 120, pp. 33–50, 2005.

[96] I. Ross and F. Fahroo, "Pseudospectral knotting methods for solving optimal control problems," *Journal of Guidance, Control, and Dynamics*, vol. 27, no. 3, pp. 397–405, 2004.

[97] M. Soler, D. Zapata, A. Olivares, E. Staffetti, and J.Cegarra, "Comparative Analysis of Commercial Aircraft Trajectory Performance," in *International Conference on Engineering and Optimization*, 2010.

[98] M. Soler, A. Olivares, and E. Staffetti, "Hybrid Optimal Control Approach to Commercial Aircrafts 3D Multiphase Trajectory Optimization," in *AIAA Guidance, Navigation and Control Conference*, 2010.

[99] A. Nuic, *User Manual for the Base of Aircraft Data (BADA) Revision 3.6*, Eurocontrol Experimental Center, 2005.

[100] D. G. Hull, *Fundamentals of Airplane Flight Mechanics*. Springer, 2007.

[101] M. Wolfson, B. Forman, K. Calden, W. Dupree, R. Johnson Jr, R. Boldi, C. Wilson, P. Bieringer, E. Mann, and J. Morgan, "Tactical 0-2 hour convective weather forecasts for FAA," in *Conference on Aviation, Range and Aerospace Meteorology*, 2004.

[102] J. Pannequin, A. Bayen, I. Mitchell, H. Chung, and S. Sastry, "Multiple aircraft deconflicted path planning with weather avoidance constraints," in *AIAA Guidance, Navigation and Control Conference*, 2007.

[103] M. L. Bujorianu and J. Lygeros, "Reachability questions in piecewise deterministic Markov processes," in *Hybrid Systems: Computation and Control*, ser. Lecture Notes in Computer Science, O. Maler and A. Pnueli, Eds. Springer, 2003, pp. 126–140.

[104] X. D. Koutsoukos and D. Riley, "Computational methods for reachability analysis of stochastic hybrid systems," in *Hybrid Systems: Computation and Control*, ser. Lecture Notes in Computer Science, J. P. Hespanha and A. Tiwari, Eds. Springer, 2006, pp. 377–391.

[105] J. Hu, M. Prandini, and S. Sastry, "Probabilistic safety analysis in three dimensional aircraft flight," in *IEEE Conference on Decision and Control*, vol. 5, December 2003, pp. 5335–5340.

[106] ——, "Aircraft conflict prediction in the presence of a spatially correlated wind field," *IEEE Transactions on Intelligent Transportation Systems*, vol. 6, no. 3, pp. 326–340, 2005.

[107] S. Prajna, A. Jadbabaie, and G. Pappas, "A framework for worst-case and stochastic safety verification using barrier certificates," *IEEE Transactions on Automatic Control*, vol. 52, no. 8, pp. 1415 –1428, aug. 2007.

[108] A. S. Nowak, "Universally measurable strategies in zero-sum stochastic games," *The Annals of Probability*, vol. 13, no. 1, pp. pp. 269–287, 1985.

[109] U. Rieder, "Non-cooperative dynamic games with general utility functions," in *Stochastic Games and Related Topics*, T. Raghavan, T. S. Ferguson, T. Parthasarathy, and O. J. Vrieze, Eds. Kluwer Academic Publishers, 1991, pp. 161 –174.

[110] T. Başar and G. Olsder, *Dynamic noncooperative game theory*. Society for Industrial Mathematics, 1999.

[111] S. Amin, A. Cardenas, and S. Sastry, "Safe and secure networked control systems under denial-of-service attacks," in *Hybrid Systems: Computation and Control*, ser. Lecture Notes in Computer Science, R. Majumdar and P. Tabuada, Eds. Springer, 2009, vol. 5469, pp. 31–45.

[112] D. P. Bertsekas and S. E. Shreve, *Stochastic Optimal Control: The Discrete Time Case*. Academic Press, 1978.

[113] J. I. Gonzalez-Trejo, O. Hernandez-Lerma, and L. F. Hoyos-Reyes, "Minimax control of discrete-time stochastic systems," *SIAM Journal on Control and Optimization*, vol. 41, no. 5, pp. 1626–1659, 2002.

[114] P. R. Kumar and T. H. Shiau, "Existence of value and randomized strategies in zero-sum discrete-time stochastic dynamic games," *SIAM Journal on Control and Optimization*, vol. 19, no. 5, pp. pp. 617–634, 1981.

[115] A. Maitra and W. Sudderth, "Finitely additive stochastic games with Borel measurable payoffs," *International Journal of Game Theory*, vol. 27, pp. 257–267, 1998.

[116] L. D. Brown and R. Purves, "Measurable selections of extrema," *The Annals of Statistics*, vol. 1, no. 5, pp. 902–912, 1973.

[117] G. B. Folland, *Real Analysis*. John Wiley & Sons, 1999.

[118] M. Davis, *Markov Models and Optimization*. London: Chapman & Hall, 1993.

[119] W. Rudin, *Principles of Mathematical Analysis*. McGraw-Hill New York, 1976.

[120] A. Abate, S. Amin, M. Prandini, J. Lygeros, and S. Sastry, "Computational approaches to reachability analysis of stochastic hybrid systems," in *Hybrid Systems: Computation and Control*, ser. Lecture Notes in Computer Science, A. Bemporad, A. Bicchi, and G. C. Buttazzo, Eds. Springer, 2007, vol. 4416, pp. 4–17.

[121] R. A. Paielli and H. Erzberger, "Conflict probability estimation for free flight," *AIAA Journal of Guidance, Control and Dynamics*, vol. 20, no. 3, pp. 588–596, 1997.

[122] A. Abate, J. Katoen, J. Lygeros, and M. Prandini, "Approximate model checking of stochastic hybrid systems," *European Journal of Control*, no. 6, pp. 624–641, 2010.

[123] A. Charnes and W. Cooper, "Chance-constrained programming," *Management Science*, pp. 73–79, 1959.

[124] A. Prékopa, *Stochastic programming*. Springer, 1995, vol. 324.

[125] G. Calafiore and M. Campi, "The scenario approach to robust control design," *IEEE Transactions on Automatic Control*, vol. 51, no. 5, pp. 742–753, 2006.

[126] G. Calafiore and L. El Ghaoui, "Linear programming with probability constraints - part 1," in *American Control Conference*, 2007, pp. 2636–2641.

[127] L. Blackmore, "A probabilistic particle control approach to optimal, robust predictive control," in *Proceedings of the AIAA Guidance, Navigation and Control Conference*, 2006.

[128] M. Ono, L. Blackmore, and B. Williams, "Chance constrained finite horizon optimal control with nonconvex constraints," in *American Control Conference*, 2010, pp. 1145–1152.

[129] M. P. Vitus and C. J. Tomlin, "Belief Space Planning For Linear, Gaussian Systems In Uncertain Environments," in *Proceedings of IFAC World Congress*, 2011.

[130] D. van Hessem and O. Bosgra, "Closed-loop stochastic dynamic process optimization under input and state constraints," in *Proceedings of the American Control Conference*, vol. 3, 2002, pp. 2023 – 2028.

[131] Y. Oishi and H. Kimura, "Randomized algorithms to solve parameter-dependent linear matrix inequalities and their computational complexity," in *IEEE Conference on Decision and Control*, 2001.

[132] S. Kanev, B. De Schutter, and M. Verhaegen, "An ellipsoid algorithm for probabilistic robust controller design," *Systems and Control Letters*, vol. 49, no. 5, pp. 365–375, 2003.

[133] N. Melchior and R. Simmons, "Particle RRT for path planning with uncertainty," in *IEEE International Conference on Robotics and Automation*, 2007.

[134] P. Missiuro and N. Roy, "Adapting probabilistic roadmaps to handle uncertain maps," in *IEEE International Conference on Robotics and Automation*, 2006.

[135] R. Alterovitz, T. Siméon, and K. Goldberg, "The stochastic motion roadmap: A sampling framework for planning with markov motion uncertainty," in *Robotics: Science and Systems*, 2007, pp. 246–253.

[136] N. Cressie and G. M. Laslett, "Random set theory and problems of modeling," *SIAM Review*, pp. 557–574, 1987.

[137] G. Matheron, *Random Sets and Integral Geometry*. New York: Wiley, 1975.

[138] D. Stoyan, "Random sets: Models and statistics," *International Statistical Review*, vol. 66, no. 1, pp. pp. 1–27, April 1998.

[139] S. A. Amburn and P. L. Wolf, "VIL density as a hail indicator," *Weather and Forecasting*, vol. 12, no. 3, pp. 473–478, 1997.

[140] B. Boudevillain and H. Andrieu, "Assessment of vertically integrated liquid (VIL) water content radar measurement," *Journal of Atmospheric and Oceanic Technology*, vol. 20, no. 6, pp. 807–819, 2003.

[141] J. Evans, K. Carusone, M. Wolfson, B. Crowe, D. Meyer, and D. Klingle-Wilson, "The Corridor Integrated Weather System (CIWS)," in *Aviation, Range, and Aerospace Meteorology Conference*, 2001.

[142] A. Nilim, L. El Ghaoui, V. Duong, and M. Hansen, "Trajectory-based air traffic management under weather uncertainty," in *USA/Europe Air Traffic Management Research and Development Seminar*, 2001.

[143] J. J. Buckley, "Graphs of Measurable Functions," *Proceedings of the American Mathematical Society*, 1974.

[144] M. Schäl, "Conditions for optimality in dynamic programming and for the limit of n-stage optimal policies to be optimal," *Probability Theory and Related Fields*, vol. 32, no. 3, pp. 179–196, 1975.

[145] G. Hoffmann, H. Huang, S. Waslander, and C. J. Tomlin, "Quadrotor helicopter flight dynamics and control: Theory and experiment," in *AIAA Conference on Guidance, Navigation and Control*, Aug. 2007.

[146] H. Huang, G. Hoffmann, S. Waslander, and C. Tomlin, "Aerodynamics and control of autonomous quadrotor helicopters in aggressive maneuvering," in *IEEE International Conference on Robotics and Automation*, 2009, pp. 3277–3282.

[147] J. Ding, E. Li, H. Huang, and C. J. Tomlin, "Reachability-based synthesis of feedback policies for motion planning under bounded disturbances," in *IEEE International Conference on Robotics and Automation*, may 2011, pp. 2160 –2165.

[148] A. Bayen, J. Zhang, C. Tomlin, and Y. Ye, "MILP Formulation and Polynomial Time Algorithm for an Aircraft Scheduling Problem," in *IEEE Conference on Decision and Control*, 2003, pp. 5003–5010.

[149] L. A. Meyn and H. Erzberger, "Airport Arrival Capacity Benefits Due to Improved Scheduling Accuracy," in *AIAA Aviation, Technology, Integration, and Operations Conference*, sep 2005.

[150] H. Balakrishnan and B. Chandran, "Scheduling aircraft landings under constrained position shifting," in *AIAA Guidance, Navigation and Control Conference*, 2006.

[151] M. Robinson, R. DeLaura, and N. Underhill, "The Route Availability Planning Tool (RAPT): Evaluation of Departure Management Decision Support in New York during the 2008 Convective Weather Season," in *USA/Europe Air Traffic Management Research and Development Seminar*, 2009.

[152] D. Michalek and H. Balakrishnan, "Identification of Robust Routes Using Convective Weather Forecasts," in *USA/Europe Air Traffic Management Research and Development Seminar*, 2009.

[153] J. Krozel, S. Penny, J. Prete, and J. S. B. Mitchell, "Comparison of Algorithms for Synthesizing Weather Avoidance Routes in Transition Airspace," in *AIAA Guidance, Navigation, and Control Conference*, 2004.

[154] D. Michalek and H. Balakrishnan, "Dynamic Reconfiguration of Terminal Airspace during Convective Weather," in *IEEE Conference on Decision and Control*, 2010.

[155] K. Roach, "Procedures and Issues of a Restrictive Runway Configuration at Dallas/Fort Worth International Airport," in *AIAA Modeling and Simulation Technologies Conference*, aug 2007.

[156] L. Leihong, J.-P. Clarke, C. C. Hui-Han, and T. Melconian, "A probabilistic decision-making model for runway configuration planning under stochastic wind conditions," in *Digital Avionics Systems Conference*, 2009.

[157] V. Ramanujam and H. Balakrishnan, "Estimation of Maximum-Likelihood Discrete-Choice Models of the Runway Configuration Selection Process," in *American Control Conference*, 2011.

[158] J. Robinson and M. Kamgarpour, "Benefits of Continuous Descent Operations in High-Density Terminal Airspace under Scheduling Constraints," in *AIAA Aviation Technology, Integration and Operations Conference*, 2010.

[159] J. Sethian, *Level Set Methods and Fast Marching Methods: Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision, and Materials Science*. Cambridge University Press, 1999, no. 3.

[160] G. Inalhan, D. Stipanovic, and C. Tomlin, "Decentralized optimization, with application to multiple aircraft coordination," in *IEEE Conference on Decision and Control*, 2002, pp. 1147–1155.