Real-Time Probabilistic Collision Avoidance for Autonomous Vehicles, Using Order Reductive Conflict Metrics

by

Thomas Jones

M.Sc. University of Stellenbosch (1999)

B.Eng. University of Stellenbosch (1997)

Submitted to the Department of Aeronautics and Astronautics
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy
at the
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
June 2003

Author .....
Department of Aeronautics and Astronautics
June, 2003

Certified by .....
John J. Deyst
Professor of Aeronautics and Astronautics

Certified by .....
James K. Kuchar
Associate Professor of Aeronautics and Astronautics
Thesis Committee Member

Certified by .....
Brent Appleby
Principal Member of the Technical Staff, Draper Laboratory
Thesis Committee Member

Certified by .....
Lorraine M. Fesq
Principal Engineer, NASA JPL
Thesis Committee Member

Certified by .....
Emilio Frazzoli
Assistant Professor of Aeronautics and Astronautics, UIUC
Thesis Committee Member

Accepted by .....
Edward M. Greitzer
H.N. Slater Professor of Aeronautics and Astronautics
Chairman, Committee on Graduate Students

Real-Time Probabilistic Collision Avoidance for Autonomous Vehicles, Using Order Reductive Conflict Metrics

by

Thomas Jones

Submitted to the Department of Aeronautics and Astronautics
on June, 2003, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

Abstract

Contemporary collision avoidance systems such as the Traffic Alert and Collision Avoidance System (TCAS) have proven their effectiveness in the Commercial Aviation (CA) industry within the last decade. Yet, TCAS and many systems like it represent attempts at collision avoidance that do not fully recognize the uncertain nature of a conflict event. Most systems circumvent probabilistic representation through simplifying approximations and pre-compiled notions of hazard space, since probabilistic representation of collision in three dimensions is considered to be an intractable problem.

Recent developments by Kuchar and Yang[70] and Paielli and Erzberger[50] have shown that collision avoidance may be cast as a probabilistic state-space problem. Innovative solution approaches may then allow systems of this nature to probe collision risk in real-time, based on real-time state estimates. The research documented in this thesis further develops the probabilistic approach for the non-cooperative, two-vehicle problem as applied in real-time to autonomous aircraft. The research is kept in a general form, thereby warranting application to a wide variety of multi-dimensional collision avoidance applications and scenario geometries.

The work primarily improves the state of the art through the creation of order reductive collision metrics in order to simplify the intractable problem of multi-dimensional collision risk calculation. As a result, a tractable, real-time, probabilistic algorithm is developed for the calculation of collision risk as a function of time.

The collision avoidance problem is contextualized not only within the realm of recent research within the CA industry, but is also likened to such concepts as the first passage time problem encountered in physics, and the field of reliability theory often encountered in civil and mechanical engineering problems. Yang's method of solution, a piece-wise straight-line Monte-Carlo approach to state propagation, is extended with a model-predictive, finite horizon risk accumulation algorithm. Through this extension we are capable of modelling collision risk for linear(-ized), time-variant, dynamic vehicle models and control strategies. A strategy is developed whereby the advantage of delayed collision avoidance action is calculated and it is framed as an extension of the notion of system operating characteristics (SOCs). The complexity of the probabilistic representation is reduced by application of quadratic conflict metrics. The numerical complexity can be reduced from $\mathcal{O}(N^{2n})$ to

$\mathcal{O}(Nlog_2(N))$ at each time step within a finite horizon time interval.

Risk calculation errors due to numerical and stochastic approximations are quantified. An applicability test is also devised whereby a vehicle's dynamic model and control characteristics may be used to calculate risk error estimates before implementing the bulk of the algorithmic solution. Various other applications of the work, outside the scope of collision avoidance, are also identified.

Thesis Supervisor: John J. Deyst
Title: Professor of Aeronautics and Astronautics

Acknowledgments

This thesis is not the culmination of my work at MIT, but rather a testimony to the support, good nature and spirit of a number of incredible people without whom I could not have gotten by.

Let me first thank the MDPP group with whom I have spent the past 4 years. They are my family away from home. Sanghyuk "San-Seng" Park (Anyoong Hase-Yo!), Francois "Ou Perd" Urbain (Los Tabernakos!), Sarah Saleh, Damien Jourdan, Damian Toohey, Jason Kepler, Richard Poutrel, Anand Srinivas and Alexander Omelchenko. Special thanks to Sanghyuk and Francois, for their wisdom, friendship and kindred spirits.

Then there is my thesis committee, each of whom have played a valuable role in my education at MIT: Prof. John Deyst, Dr. Brent Appleby, Dr. Lorraine Fesq, Prof. James Kuchar and Prof. Emilio Frazzoli. Special thanks to Lorraine, who arrived at MIT at the same time as I did, for your open door, friendship and the skills you have taught me. Also a very special thanks to Prof. Deyst, my mentor, who took many chances on me and provided me with so many opportunities, again and again. I trust that I have not disappointed you.

There are also a number of silent partners who surfaced at the most critical times to provide support and guidance. I wish to thank Prof. Danielle Veneziano, for sound advice and his willingness to help; Dr. Lee Yang, for proof reading the thesis and his valuable inputs; Sean George, for his assistance in my day to day project research; Donald ("The Don") Wiener, for his kindness, wisdom and support; Dick Perdichizzi, for his constant and easily offered advice; and Phyllis Collymore, for sorting out anything that could possibly go wrong.

I need to thank a number of individuals who believed in me enough to support my coming to MIT in the first place. Prof. Jan du Plessis, dankie vir sewe jaar van ondersteuning, aansporing, vertroue en advies; Dr. Jan Roodt, for his wisdom and support; and the NRF, in the personage of Rose Robertson, for the financial support which sustained me when I needed it the most.

Then there are my friends in Boston: Ernst Scholtz, Amanda Hollenbaugh, Tom Reynolds, Halyley Davison, Blake and Jeanine Booysen, Simon Nolet (Je me souviens Quebecois, mon ami!), Xueen Yang and Lodewyk Steyn, sources of constant support and unforgettable times. Also my friends back home, Japie and Madri Engelbrecht, Nelius Goosen, Gerhard le Roux and P.C. Visagie, for their warm friendship and for their roles in motivating me to study at MIT.

I wish to thank my mom and dad for believing in me and for having brought me up to be the man I am today. Also many thanks to my new family, Lex and Andy de Vries, soon to be called mom and dad too.

Then there is Lisa Ann de Vries, or simply Leesie. I thought I'd be going to the States for a Ph.D, but I also found you. Thank you for your love and kindness, constant support and for all those adventures past and those yet to share. I love you dearly and this thesis is dedicated to you. Leesie, I'm coming home now!

Contents

List of Figures

List of Abbreviations

ATM	Air Traffic Management
CA	Commercial Aviation
CD	Correct Detection
CR	Correct Rejection
CSOC	Condensed System Operating Characteristic
DMOD	Distance Modification (see TCAS)
FA	False Alarm
FAA	Federal Aviation Authority
FFT	Fast Fourier Transformation
FMC	Flight Management Computer
FI	Fourier Inversion
FOSM	First Order, Second Moment; As superscript indicates corrected to FOSM
IDFT	Inverse Discrete Fourier Transformation
IFFT	Inverse Fast Fourier Transform
IID	Independent, Identically Distributed
lb	Lower Bound; As superscript indicates upper bound value
LTI	Linear, Time Invariant
MD	Missed Detection
MDPP	MIT/Draper Technology Development Partnership Program
MPT	Mean Hazard Passage Time
NGM-II	New Generation Mini Vehicle Two (a PCUAV Aircraft)
PCUAV	Parent-Child Unmanned Aerial Vehicle Project
PDF	Probability Density Function
SA	Successful Avoidance
SOC	System Operating Characteristic
TCAS	Traffic Alert and Collision Avoidance System
UA	Unnecessary Avoidance
UAV	Un-crewed Aerial Vehicle
ub	Upper Bound; As superscript indicates upper bound value
UCAV	Unmanned Combat Aerial Vehicle

List of Nomenclature

Symbol	Meaning
Δt	Time discretization step size
$\delta(t-\tau)$	Dirac delta function occurring at $t=\tau$
$\Delta E[a_k]$	Change in expected value of a because of truncation after $t=t_k$
$\Delta\gamma$, $\Delta\dot{\gamma}$, $\Delta\Gamma$, $\Delta\dot{\Gamma}$	Equivalent to $\Delta t\cdot\gamma$, $\Delta t\cdot\dot{\gamma}$, $\Delta t\cdot\Gamma$, $\Delta t\cdot\dot{\Gamma}$ respectively
γ	$\gamma = x^T Bx$ quadratic collision metric
γ_0	Boundary of D_f such that $\gamma<\gamma_0$ constitutes collision
$\phi_{\Gamma_k}(\tau)$	Characteristic function of γ
Ω_{a_k}	FOSM error in calculation of a at $t=t_k$
A_i	Avoidance maneuver delayed by $i.t_{AD}$
$D_a(t)$, $D_f(t)$, $D_s(t)$	Domains of alerting, failure and safety respectively
DCT_i	i'th Discrete Time Compound (Avoidance) Trajectory
$E[a]$	Expected value of some random variable a
$f_{X(t)}(x)$	PDF of x at time t
$f_{X(t_a)}(x)\|B_{[b,c]}$	PDF of x at time t_a given no collision within $[t_b,t_c]$
$f_{\Gamma(t_a)}(\gamma)\|B_{[b,c]}$	PDF of γ at time t_a given no collision within $[t_b,t_c]$
$f_{X_k}^{B_k}(x)$	$f_{X(t_k)}(x)\|B_{[0,k-1]}$, zeroed where $x\in D_f$, and Bayes normalized
$f_{\Gamma_k}^{B_k}(\gamma)$	$f_{\Gamma(t_k)}(\gamma)\|B_{[k-1,k-1]}$, zeroed where $\gamma<\gamma_0$
$g_{A_k}(x)$, $g_{B_k}(x)$	$[1-u(x-x_k)]f_{X(t_k)}(x)\|B_{[0,k-1]}$ and $u(x-x_k)f_{X(t_k)}(x)\|B_{[0,k-1]}$ respectively
$g_{A_k}(\gamma)$, $g_{B_k}(\gamma)$	$[1-u(\gamma-\gamma_k)]f_{\Gamma(t_k)}(\gamma)\|B_{[k-1,k-1]}$ and $u(\gamma-\gamma_k)f_{\Gamma(t_k)}(\gamma)\|B_{[k-1,k-1]}$ respectively
K	Number of discrete simulation time steps
$max_{[t_a,t_b]}[f(t)]$	Maximum value of $f(t)$ between t_a and t_b, $t_a<t_b$
$I\equiv t_h/t_{AD}+1$	Number of Discrete Time Compound Trajectories
$N(m,\Sigma)$	Normal distribution with mean m and covariance Σ
$P(A)$	Probability of occurrence of event A
$P_c(t_a)$	Total probability of collision accumulated over the interval $[t_0,t_a]$
$P_c(k)$	$P_c(k)\equiv P_c(t_k)$

$P_c^{lb}(k)$, $P_c^{ub}(k)$, $P_c^{true}(k)$	Lower bound, Upper Bound and True probabilities of collision respectively, accumulated over the interval $[t_0, t_k]$
$P_{c_{max}}$	Maximum acceptable safe probability of collision for a vehicle
P_c^A, P_c^N	$P_c(t_h)$ given that avoidance action is taken or not, respectively
R^n	n-dimensional Real space
T_1, T_2	$P(SA)$ and $P(UA)$ Alerting Thresholds respectively
t	Time
t_0, t_h	Simulation start time and finite horizon (end)time respectively
t_c	Maximum simulation time length, bound by real-time tractability
t_d	Unforeseen intent change implementation delay interval
t_k	Time at k'th discrete time step
t_w	Finite horizon simulation time window length
$u(x - x_a)$	Unit step function at $x = x_a$
$v(t)$	White measurement noise
$w(t)$	White process noise
$x_i(t)$	i'th state variable
$x(t)$	State vector, $dim(x(t)) = n$
$x_{OL}(t)$	Open loop state vector

Chapter 1

Introduction and Background

1.1 Background

More than a century ago we saw the dawn of an era of crewed flight. We have now embarked upon a new era, where the wonders of aviation technology have become commonplace and technological advances, especially in electronics and software, have changed the face of the aviation industry over the last twenty years[38].

The 1980's and 90's saw the introduction of so-called glass cockpits and later flight directors and traffic alert systems, increasingly pushing towards automation in the commercial aviation (CA) cockpit. Even though the human flight crew does play a critical role, few passengers know that autopilots and flight management computers (FMCs) are responsible for flying them most of the way between airport destinations. Recently, various issues such as loss of situational awareness have been under scrutiny and have been linked to human pilots' automated cockpit systems[36]. This opens the door for automated collision avoidance and air traffic management (ATM) systems that would be capable of playing the role of not only an alerting system, but also would automatically solve inter-vehicle conflict.

Military aviation has also undergone a dramatic transformation during the end of the 20th century. More and more Un-crewed Aerial Vehicles (UAVs)[47] are being developed and many of those such as Predator and Global Hawk have become household names. The state of current technology and a political climate, where risk of the loss of human life is of increasing concern, are fuelling the inevitable growth of the UAV industry. Some

of the advances so far include un-crewed combat vehicles (UCAVs), un-crewed reconnaissance vehicles and multi-role UAVs like Predator[47][7][1][13][46]. Recent developments at MIT, such as the Parent-Child Un-crewed Aerial Vehicle (PCUAV)[63][51][37] system have shown that multi-tier co-operative UAV systems are prime candidates for the collection of up-close reconnaissance information. This is especially true when dealing with cluttered and inhospitable terrain such as caves and forests where high-altitude surveillance proves to be of limited use. These missions were always considered to be too risky for piloted flight.

Many UAVs are still piloted by human operators, but the added separation between the pilot and the vehicle and limited sensory perception (esp. visual) only increases the frequency and severity of loss of situational awareness. Add to this the expense of having highly skilled operators for each independent vehicle and large vehicle support teams and we see that the opportunities for added autonomy abound. As vehicles become increasingly self-sufficient, one such opportunity arises for the creation of automated collision avoidance systems for autonomous vehicles.

Most collision avoidance systems are developed for the CA industry, in an attempt to boost safety while relaxing the in-trail spacing currently observed to maintain adequate traffic separation, usually during piloted flight[67][56]. There is a need to create automated collision avoidance systems, especially for autonomous vehicles, that are on par with the most recent developments in alerting systems for the CA sector[43][70][69].

At present, UAVs and especially autonomous UAVs are required to remain outside of FAA controlled airspace. Except in very special circumstances, air traffic control authorities do not allow autonomous, or even un-crewed, aircraft general access to such airspace, since they are considered to pose a threat to civilian traffic. Vehicles would need not only trusted control systems and trajectory scheduling capability, in order for this ban to be lifted, but would also require the ability to adapt to unforseen hazardous situations. Automated collision avoidance therefore brings us one step closer towards opening a door to commercial airspace for autonomous UAVs. Until then, the UAV industry will be constrained to specialized missions, especially of a military nature.

There are a number of reasons why UAVs provide a remarkable testing ground for automated collision avoidance systems. Foremost is the lack of human presence aboard a UAV, thereby allowing us to attempt a large number of solutions without putting life in danger. The resulting development rate of systems that integrate onto UAVs and the reduced certification requirements make this platform an excellent breeding ground for the development of new avionics systems. Also, there is an increasing need for the interaction between UAVs and crewed vehicles, a prospect that can naturally make any pilot nervous. Automated collision avoidance systems may make such interaction feasible in the near future. Finally, there is the issue of using UAVs as a proving ground for avoidance systems that might eventually make their way into a crewed cockpit.

1.2 Uniqueness of the Autonomous UAV Problem

Inter-vehicle collision avoidance systems are naturally designed to be most useful under conditions of increased air traffic density, where there is an associated increase in the risk of conflict. The traditional school of thought dictates that avoidance systems be limited to acting primarily as alerting systems, since pilots are especially vigilant or fully in control of an aircraft under conditions of increased traffic. Pilot response then dictates the measure of conflict resolution. This view still holds for the CA industry, and rightly so. Humans can be skilled operators and may evaluate measures of safety and apply measures of evasion and control that are far beyond the capabilities of traditional automated systems. A more probabilistic view suggests that pilots also add a measure of decision-making capability to the collision avoidance problem that is at least not fully correlated with the existing alerting system. This may improve the avoidance system's performance. We have grown accustomed to their proven safety record and thus airlines, passengers and authorities feel more comfortable when a human supervises and/or controls a vehicle.

It is partly the complexity of the nature of a human operator that also hampers the development of adequate collision avoidance systems. There is a very definite uncertainty associated with the intent of a human operator. Collision avoidance by definition includes the capability to project future vehicle state in order to be able to predict future inter-vehicle conflict. It is the uncertainty of intent that hampers the accuracy of conflict detection and

most alerting systems need to be designed around this problem [41][42][69][70].

Autonomous UAVs provide a unique opportunity to create conflict detection systems where more certainty exists about vehicle intent. Most auto-pilots are deterministic and models of their operation may be created with an accuracy far exceeding that of human operator models. There remains a measure of uncertainty though and its effects on the avoidance system are discussed in Section 4.3.1. Figure 1-1 illustrates the typical conceptual difference in future state prediction uncertainty between manually and autonomously piloted vehicles in the presence of a hazardous area. Such a hazardous area is referred to as a domain of failure $D_f(t)$.

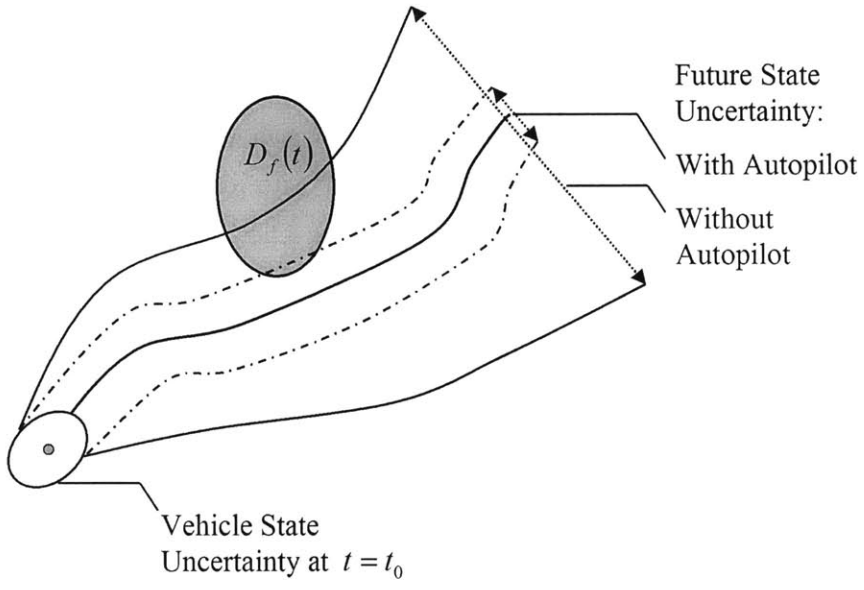

Figure 1-1: Future State Uncertainty as a Result of Uncertain Intent

Another troublesome aspect of alerting systems is that there is an interaction between the performance of collision avoidance and an alerting system's tendency to issue false alarms. Section 2.1 provides more detail about this concept which was also analyzed by Kuchar[41]. In short, false alarms lead to mistrust of a system by human operators. It is this mistrust of a system that reduces its effectiveness, in accordance with such principles as risk homogenization and negative reinforcement[8]. Pilots tend to become less inclined to

take action after false alerts have been issued, thereby increasing the risk of collision when an alert turns out to be valid.

Autonomous UAVs do not suffer from the same adverse consequences of interaction between false alarm rates and system performance. In a well-segmented system, one sub-system, namely the conflict alerting software, would decide whether a collision is imminent and alert when a threshold is exceeded. The vehicle's guidance and control sub-system would then implement an avoidance trajectory suggested by the alerting system. Even though false alarm rates may increase the number of trajectory changes and the associated trajectory cost, the autonomous control system would not normally distrust the validity of alerts, since it does not usually have this capability. After a flight we may analyze the performance of an alerting system and adjust it as best possible, but in-flight alerts are always heeded. The result is that we may design a system where the cost associated with false alarms is much smaller than that for piloted platforms.

1.3 Research Objectives

This thesis details the development of a real-time, automated collision avoidance system for autonomous vehicles. We focus on a non-cooperative two-vehicle conflict scenario. First we will show how this problem is different from conflict resolution and alerting systems usually developed for crewed flight. It will be shown that these differences may be exploited at a basic level of system design and that a new breed of collision avoidance systems may be developed that is based on the creation of collision risk estimates within a finite time horizon interval.

The work is primarily developed from the probabilistic framework and principles laid down by Kuchar and Yang [42] [70] [41] [69]. The initial development also draws on methods described by Paielli and Erzberger[49][50]. An emphasis is placed on progressing from basic probability theory and we show that the solution is computationally intractable. It is, after all, this intractability that usually leads researchers to apply ad-hoc methods of approximation and probabilistic solutions such as the Monte-Carlo method. Instead, we show how the complexity of the solution space may be reduced considerably through the application of

order reductive collision metrics, such as describing collision hazards as multi-dimensional quadratic forms. We then apply well-founded approximations to the remaining problem in order to develop a tractable probabilistic solution to real-time collision avoidance.

It is also shown how the System Operating Characteristic (SOC) representation developed by Kuchar [42][41] may be expanded to evaluate the non-trivial probabilistic effects of delaying collision avoidance action.

Two example applications are simulated, one for a small PCUAV vehicle and one for large commercial transports. A number of future avenues of research are discussed, including application to automated landing systems and other automated maneuvers where there is a need to evaluate whether vehicles remain within safe bounds of state space, not just position-space.

1.4 Thesis Overview

In Chapter 1 we focus on background aspects highlighting the need for collision avoidance systems, especially as applied to autonomous vehicles. It is further illustrated how the collision avoidance problem for autonomous vehicles differs from that of piloted vehicles. A short overview of the thesis is then provided.

Chapter 2 elaborates on the basic concepts of collision avoidance as described and viewed by contemporary researchers and focusses on the state space representation of these problems developed by Kuchar[41]. Different approaches to collision avoidance are discussed, culminating in the probabilistic approach upon which the research detailed in this thesis is founded. The collision avoidance problem is then compared to similar problems described in the field of reliability theory and also to the so-called first passage time problem.

The next step is to start building on the probabilistic collision avoidance problem in Chapter 3. It is shown how reduced uncertainty associated with autopilot intent may be combined into a closed loop state space representation of the vehicle under autonomous control. The state space representation may then be used to predict state distribution

within a finite horizon interval through model-predictive mean and covariance propagation. A conflict risk calculation algorithm is then designed using the predicted state distribution and it is shown to be an intractable problem. The final section of the chapter describes how the notion of system operating characteristics (SOCs) may be expanded upon in order to handle the effects of time-delayed collision avoidance action.

Chapter 4 focusses on the development of a tractable algorithm for the determination of collision risk as a function of time. The algorithm is empowered by a order reductive collision metric and simple statistical approximations. An approximation analysis is completed and calculation errors are characterized and quantified. The chapter also addresses a number of practical issues associated with, for example, the sizes of time horizon intervals and simulation steps.

After development of the theory up to Chapter 4, Chapter 5 applies the research and analyzes the performance of simulated collision probabilities as a function of time. The research is applied to two application examples: A 2m wingspan autonomous UAV encountering a tree trunk-sized hazard; and two large commercial transports conflicting with each other while under autopilot control.

Chapter 6 summarizes the contributions of the research and conclusions drawn, culminating in a discussion of future avenues of development and applications of the research.

1.5 Chapter Summary

It is illustrated that automatic collision avoidance systems for autonomous vehicles need to be developed: In order to allow UAVs to operate in FAA controlled airspace; in order to increase the effectiveness of commercial aircraft collision avoidance when under autopilot control; and to ensure safety when piloted and autonomous vehicles are required to interact.

Differences between piloted and autonomous vehicles that influence conflict avoidance design are discussed. The effectiveness of alerting systems for autonomous vehicles is shown to be less sensitive to the occurrence of false alarms. Uncertainties usually associated with

future pilot intent can be reduced dramatically by deterministic autopilots, thereby opening the door to more precise probabilistic conflict detection. It is the hope of this author that the research presented in this thesis will help facilitate the implementation of more autonomous flight systems, while making the skies safer for all.

Chapter 2

Alerting System Concepts

2.1 General Concepts

Kuchar [41] showed that conflict alerting systems are but a part of several safety components in complex vehicle systems. Yang[69] illustrated how conflict detection and resolution can further be represented in a closed loop framework. An illustration of this concept, adapted from Yang's work, is shown in Figure 2-1. The adaptation generalizes the concept to both crewed and un-crewed vehicles.

Together, the environment, conflict detection and conflict resolution form a collision avoidance system. For autonomous vehicles such a system monitors the environment, predicts vehicle-relative state, issues alerts and executes an avoidance trajectory (conflict resolution). This is slightly different from traditional conflict avoidance systems where alerts are issued, but resolution is accomplished by a human. The major elements of collision avoidance will be discussed briefly.

First, ownship[1] and air-traffic state must be monitored. Various sensors may be used to accomplish this, but measurements are all combined in estimator architectures to provide the best state estimate and a corresponding estimate of the state distribution at time $t = t_0$. As with most alerting systems, we always assume that the estimated state variables are of a multi-variate normal distribution[69][19][71]. Measurements might result in full- or

[1] "ownship" is defined as the vehicle running the collision avoidance software

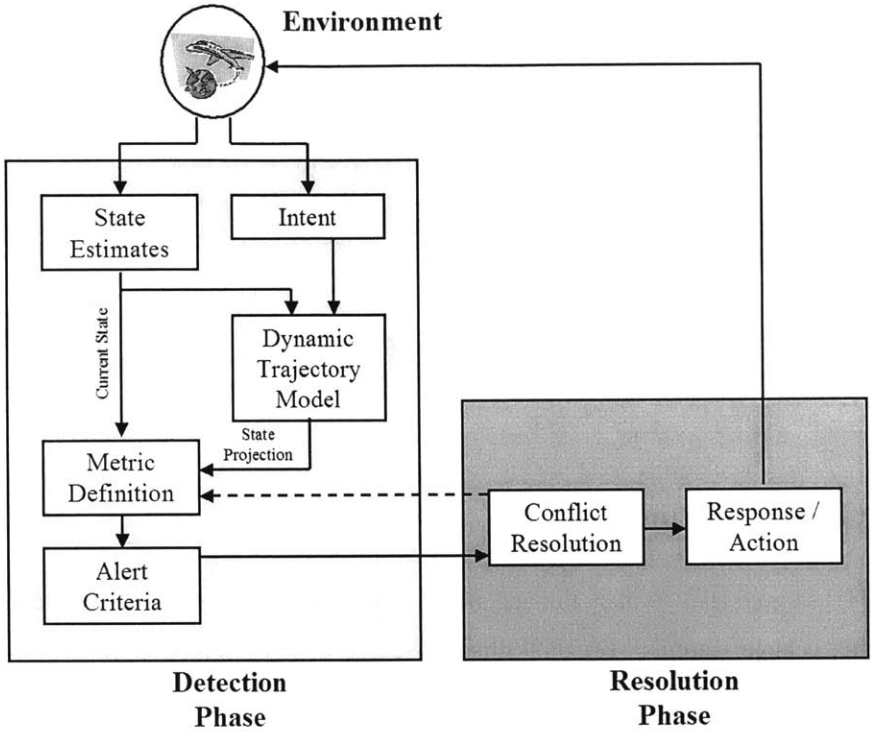

Figure 2-1: Block Diagram of the Conflict Detection and Resolution Process

partial-state estimation.

Intent must next be established. This includes models of the auto-pilot and trajectory scheduling systems of vehicles and also knowledge of flight plans and waypoints. We will refer to the known flight path, before avoidance action is taken, as N, the nominal trajectory. As discussed earlier, knowledge of intent is essential when attempting to increase prediction accuracy. A closed loop dynamic model of the vehicle must be known and when combined with intent information we may propagate current state estimates into the future.

We next define alerting parameter(s) in order to judge whether collisions are likely. Such parameters are known as alerting metrics or collision metrics. Part of the focus of this thesis is in defining a collision metric in such a way as to reduce the state order, and thus complexity, of a specific collision avoidance problem.

After describing a collision metric, we may test it against a number of pre-defined safety thresholds or alert criteria. For probabilistic systems we usually test for the probability of crossing a specific metric threshold and test this probability against a pre-defined acceptable measure of collision risk. The latter is one way to ensure that the alerting system is robust against uncertainties in the calculation of collision metrics. An example of a simple singular alert criterium would be to alert if the minimum estimated vehicle separation will be less than 100m some time within the next 30s (TCAS[67][69]).

Collision avoidance systems may decide to give an alert based on the degree of certainty of a conflict event and also the predicted time of occurrence of the conflict. Yang assumes that when a conflict is detected one time, it is both predicted to occur and deemed appropriate to alert an operator [69]. We will soon show how the work may be extended to predict not only the probability of collision within a finite time horizon, but to estimate the last possible time when it is essential to alert (and automatically avoid). The system we propose will also provide the avoidance maneuver of choice at the time of the alert, from amongst a pre-defined set of avoidance maneuvers.

Once an alert is issued, we may enter the conflict resolution phase. Such alerts may also be in the form of advisories, generating an advised resolution action or trajectory. We will especially focus on creating such a system. More precisely, we will aim to create an advised avoidance trajectory that can be implemented by an autopilot trajectory planner. This means that we will not only issue advisories, such as"turn left", but we will also go as far as mapping out the specific avoidance trajectory. An autopilot may then follow this avoidance trajectory and for crewed flight a flight director may instruct a pilot how to proceed. On a higher level, a suggested avoidance trajectory may become the first section (maneuver) of a newly planned flight path in a system such as that developed by Frazzoli[27] at MIT.

The above discussion provides some insight into the multitude of estimates, metrics, thresholds and conflict resolution factors, all dependent on one another, which are found in the literature. It is this complicated dependence which makes the problem an interesting one, resulting in various solutions being proposed by a number of different researchers (See survey in [43]). Most of these methods are however ad-hoc approaches, with little proba-

bilistic foundation. Varying levels of success are reported, usually under specific conditions and for specific conflict scenarios. In this thesis we attempt to create a well-founded probabilistic approach, partially based on the work of Kuchar[42][41] and Yang[70][69], for a more general avoidance scenario.

2.2 Methodologies

This section is intended to ground the research within the realm of previous research on the topic of collision avoidance. First we show how to represent the problem within an established framework of the field of collision avoidance and then we illustrate a method of approach which leads up to an efficient solution space.

2.2.1 State Space Representation

In [42] and [41] Kuchar developed a state space representation of collision avoidance problems. Figure 2-2 summarizes the state space representation.

First we define the time dependent state vector

$$x(t) = [x_1(t)...x_n(t)]^T, \tag{2.1}$$

where n is the order of the system represented in state space and each state variable $x_i(t)$ represents a parameter that may in some way be employed by the alerting logic. We specify that that $x(t) \in R^n$. The state of the vehicle may change within a period of time, and the history of $x(t)$ within $t = [t_1, t_2]$ will be referred to as the *state trajectory* of the system within $[t_1, t_2]$.

In general there are regions of, or subsets in, the state space R^n that are hazardous to the vehicle. These regions are usually defined as *hazard space*, sometimes denoted by $H(t)$. In this development we will refer to such a region as a *domain of failure*, denoted by $D_f(t) \subset R^n$, in accordance with the principles of reliability theory[5].

It is clearly undesirable to have $x(t) \in D_f(t)$ and if $x(t)$ enters $D_f(t)$ we say that a *missed detection* has occurred. A missed detection constitutes a failure of the collision

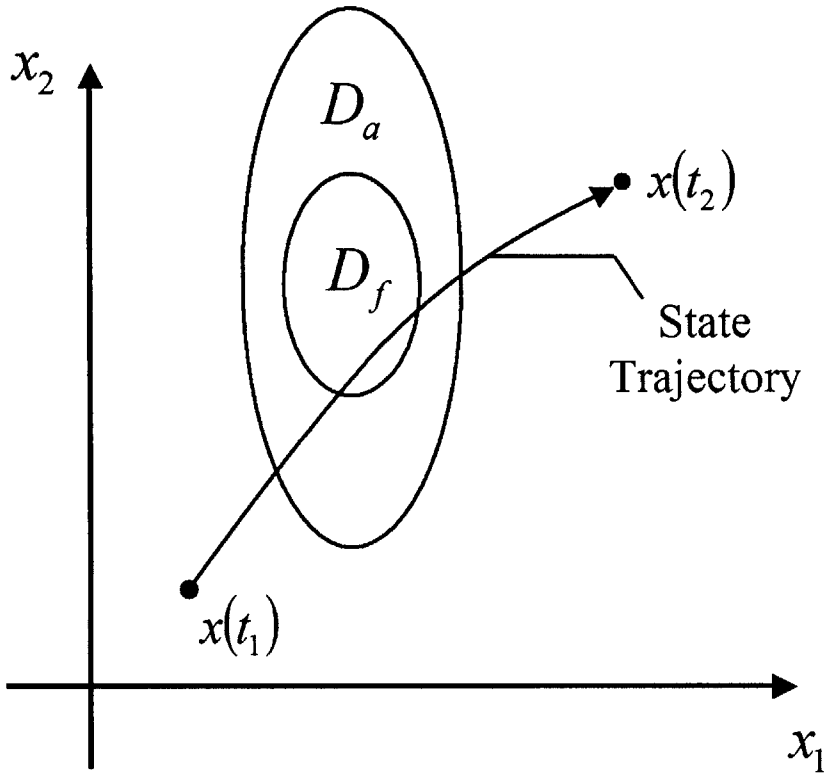

Figure 2-2: A 2-Dimensional State Space Representation of the Avoidance Problem

avoidance system.

The dynamics, state and relevant avoidance trajectories of a vehicle require that an alert be issued reasonably far outside of $D_f(t)$ so as to ensure that a missed detection does not occur. We therefore define the *alert space* as $D_a(t)$, such that $D_f(t) \subset D_a$. The boundary between $D_a(t)$ and $D_a^c(t)$ (the complement of $D_a(t)$) represents the alert threshold. When $x(t) \in D_a(t)$ we issue an alert in an attempt to avoid a collision, i.e., avoid $x(t) \in D_f(t)$. It is therefore prudent to alert well enough before entering $D_f(t)$ in order to resolve a predicted conflict. When vehicles are operated by humans, we have to account for avoidance action delays, caused by the operator's reaction time. Autonomous vehicles will react more predictably and probably also with less delay.

2.2.2 Standard Definitions and Framework

At this point it is useful to specifically define a number of different outcomes to a conflict scenario. These are illustrated in Figures 2-3 and 2-4. The outcomes are split among intuitive boundaries, and have become a standard in the field of conflict avoidance[42][68][30][20]. Yang[69] and Kuchar[41] describe the definitions and framework in a similar way.

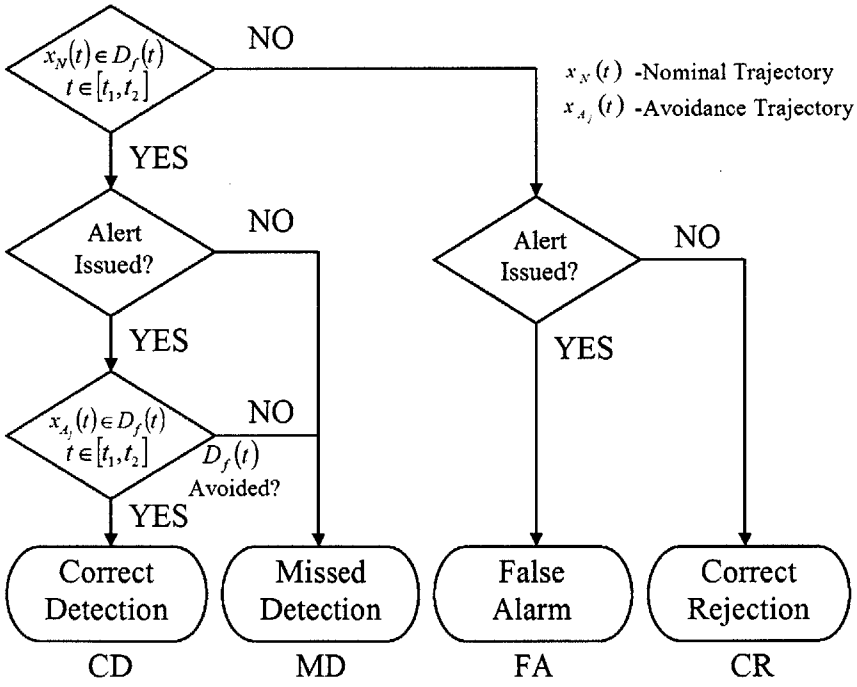

Figure 2-3: Flowchart of Different Alerting Outcomes

For conflict prediction we first need to establish if a vehicle would inevitably enter $D_f(t)$ if no action is taken.

If no collision will occur, issuing an alert would constitute a *false alarm* (FA). In such a case, taking avoidance action that induces collision also constitutes an FA. Refraining from issuing an alert, the correct action to take, is referred to as *correct rejection* (CR).

When however $x_N(t) \in D_f(t)$ will inevitably be true if no action is taken, then we should issue an alert to avoid collision. Issuing an alert in time to avoid the collision constitutes

a *correct detection* (CD). Any other eventuality, such as failing to issue the necessary alert or alerting too late, constitutes a *missed detection* (MD).

The only truly positive outcomes are correct detections and correct rejections. False alarms may result in a loss of confidence in a system and, even worse, induced collision. Missed detection always results in entrance onto $D_f(t)$, usually because of taking late avoidance action, since $D_f(t) \subset D_a(t)$. State uncertainty allows us to express these outcomes only as probabilities. For example, $P(CD)$ would be the probability of correct detection and similarly for $P(FA)$, $P(MD)$ and $P(CR)$.

The collision avoidance problem may now be stated as follows: Given a nominal trajectory $x_N(t)$, a domain of failure $D_f(t)$ and J avoidance trajectories $x_{A_j}(t)$, find an algorithm that minimizes a balance of both $P(MD)$ and $P(FA)$.

In other words, the goal is to create an alert space which strikes the most effective balance between $P(FA)$ and $P(MD)$. For example, if $D_a(t)$ is enlarged so that fewer missed detections occur and $P(MD)$ is reduced, this results in an increased number of false alarms and $P(FA)$ will be increased. This is a conservative approach where alerts will be issued more often. Conversely, reducing the size of $D_a(t)$ will decrease the number of false alarms and decrease $P(FA)$, while increasing the number of missed detections and increasing $P(MD)$. A fundamental and interesting trade-off thus results between $P(FA)$ and $P(MD)$ which Kuchar showed is commonly encountered in signal detection theory[42][41]. In particular, there will always be uncertainty in the outcome of alerting decisions, and a tradeoff between $P(FA)$ and $P(MD)$.

2.2.3 Methods of Approach

Traditionally, collision avoidance is solved through prediction of conflict events by means of *single path* and *worst case* approaches. Cited examples of these approaches are primarily taken from a survey paper by Kuchar and Yang [43].

The *single path* approach assumes that a vehicle follows a fixed trajectory, often a straight line, and that conflict occurs when a safety buffer around one vehicle is projected

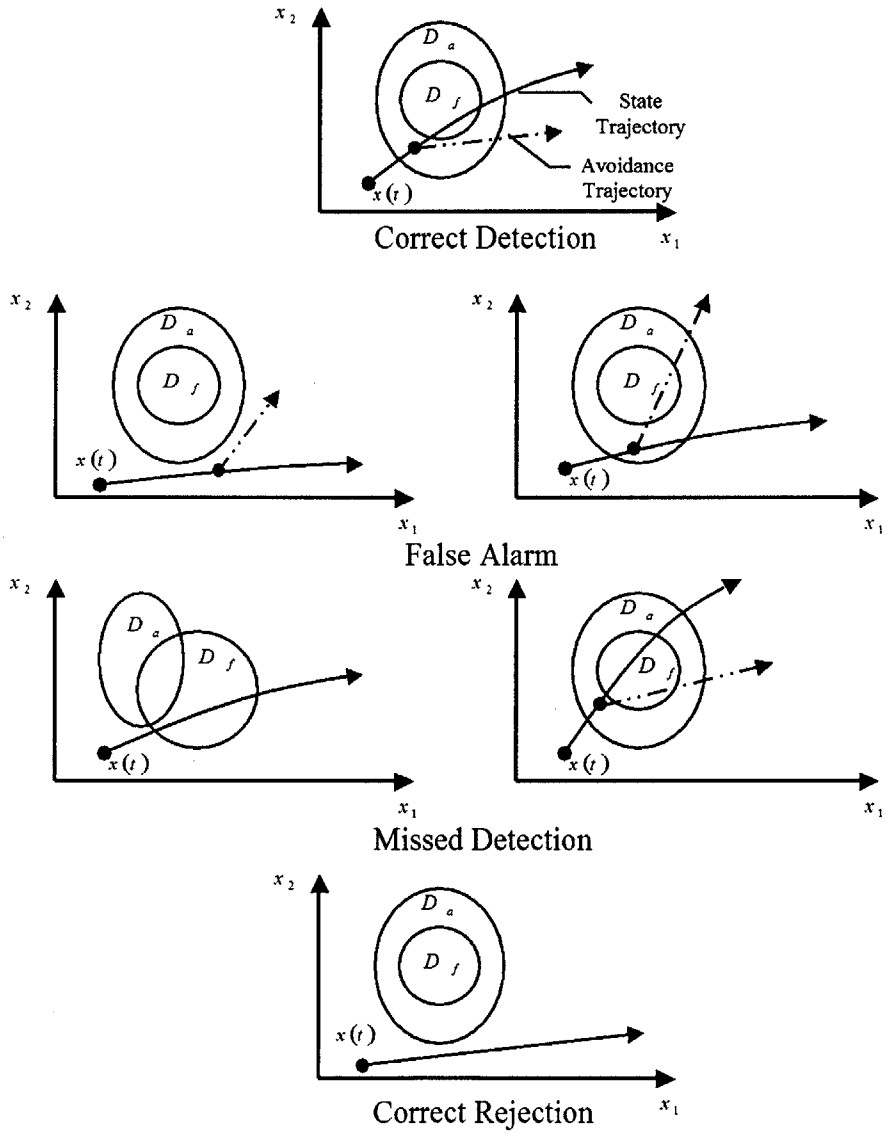

Figure 2-4: Possible Different Outcomes of an Alert in a State-Space Representation

to intersect with $D_f(t)$. This is a deterministic approach with a definite hit or miss result. Some examples of this approach may be found in [40], [4] and [21].

The *worst case* approach propagates a vehicle's state trajectory according to its maximum dynamic capabilities. An alert is issued when it is predicted that $x(t) \in D_f(t)$ will occur within a time frame of interest. This is still a deterministic approach with definite hits and misses being predicted. The method was originally developed in an attempt to model the probabilistic nature of heading estimates. It is most useful for short durations of conflict prediction, since trajectory deviations may be severe when projected at maximal rates. Examples of this approach may be found in [59] and [26], among others and also includes the application of viability or "reachability" theory as applied by Teo and Tomlin[62].

Most single path and worst case approaches may be implemented in real-time, since this is partly the purpose of their creation. These are tractable, approximate, deterministic solutions to a generally intractably probabilistic problem. Such methods do however suffer from inaccuracies brought on by approximation. These inaccuracies often manifest themselves as sub-optimal false alarm and missed detection rates and unpredictable behavior. Most of these systems are applicable to very specialized collision avoidance problems [43][69] and are highly sensitive to a vehicle's control strategy and dynamic model.

The probabilistic approach forms a middle ground between single path and worst case methods. In this case, we determine the likelihood of collision as the likelihood of $x(t) \in D_f(t)$ within a given time horizon. Yang [69] views it as weighing all possible trajectories with their probability of occurrence and then adding up the contribution of those that cross into $D_f(t)$. The probability of collision accumulated over the interval $[t_0, t]$, as a function of time, will be denoted by $P_c(t)$. In this way the amount of collision risk accumulated from the start to the end of the time horizon, i.e. within the interval $t = [t_0, t_h]$, becomes $P_c(t_h)$.

The reason why the probabilistic approach has been used very sparingly is because, in general, the problem is intractable[26][70]. The single path and worst case methods were created as approximate solutions, in an attempt to avoid having to solve for probabilities of collision. Yang and Kuchar [69][42][70] and Paielli and Erzberger[49][50] have been at

the forefront of re-stating the problem in its probabilistic form and this thesis partly builds upon, and expands the scope of, their work. Other probabilistic approaches include [66], [54], [58] and [33].

2.3 A Probabilistic Approach

2.3.1 Concerning Alert Space

The probabilistic approach essentially redefines the notion of alert space. In this section, we show how a probabilistic alert space results and that entry into this space is equivalent to crossing the alerting threshold (see Figure 2-6).

In Section 2.2.1 we explained how the alert space $D_a(t)$ can be defined as a region of state space where alerts are issued so as to avoid the domain of failure $D_f(t)$. Much of the development effort of collision avoidance systems such as TCAS is expended in an attempt to define $D_a(t)$[56][20]. In this process it soon becomes clear that $D_a(t)$ needs to be tailored to specific scenarios and approach geometries in order to ensure that both false alarms and missed detections are minimized. The shape of $D_a(t)$ is usually optimized across multitudes of representative conflict scenarios, and for various vehicle applications. We will refer to this operation as the creation of pre-compiled alert spaces.

A pre-compiled alert space ensures that relatively little processing is required to perform real-time alert calculations. Estimation of the most likely trajectory of a vehicle is required and is then checked for intersection with $D_a(t)$. Pre-compilation needs to be a robust solution, representing in some sense the global optimal $D_a(t)$ across a number of different vehicle platforms and for different conflict scenarios. It is this robustness that also curtails its performance, settling for sub-optimal instantaneous relative values of $P(FA)$ and $P(MD)$. Such systems usually err on the side of false alarms, since these are usually considered to be safer than missed detections.

Another way to view the problem is to define two trajectories: The nominal trajectory without avoidance, denoted by N, and the avoidance trajectory, denoted by A. Then let us also define the probabilities of collision along each of these trajectories as P_c^N and P_c^A

respectively. In this way we may define

$$P(UA) \equiv 1 - P_c^N \tag{2.2}$$

and

$$P(SA) \equiv 1 - P_c^A \tag{2.3}$$

as the probabilities of unnecessary avoidance and successful avoidance respectively. Figure 2-5 illustrates the two concepts.

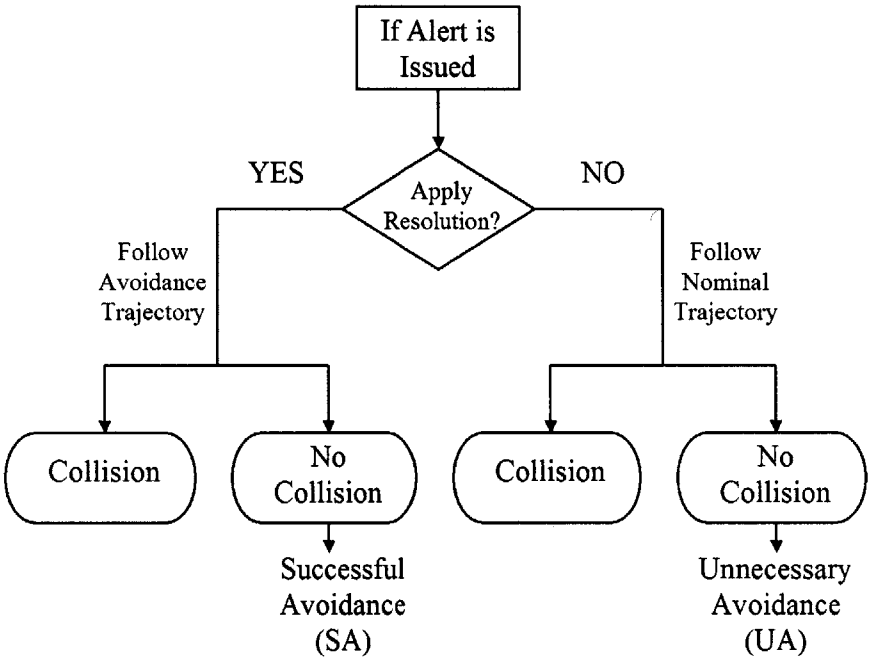

Figure 2-5: Flowchart Representation of Successful Avoidance and Unnecessary Alerts

It is useful to view a truly probabilistic approach from a different perspective. Instead of defining $D_a(t)$ in state space, we may define it in terms of probability space, as illustrated in Figure 2-6. The remainder of this section is devoted to illustrating this concept. Simply put, an alert may be issued when it is deemed probable that a collision would occur along the existing (nominal) trajectory, while there still remains a high probability of effecting safe collision avoidance.

Kuchar[41][42] shows how decision theory may be used to define an optimal region within the space $P(SA)$ vs. $P(UA)$ in an attempt to homogenize the performance of an alerting system over all encounters. $P(SA)$ vs. $P(UA)$ maps out a system operating characteristic curve[41] as a function of time. This concept is discussed in more detail in Section 3.3 and is briefly illustrated in Figure 2-6.

We may thus decide on acceptable levels of false alarms and probabilities of safe avoidance for a specific vehicle and then alert when such thresholds have been transgressed. Such thresholds are represented by T_1 and T_2 in Figure 2-6. Probabilities of collision are then determined using $D_f(t)$, a current state estimate and knowledge of our vehicle's dynamic model and tested against the aforementioned thresholds of $P(SA)$ and $P(UA)$. $P(SA)$ and $P(UA)$ therefore plot out a curve over time. Even though $D_a(t)$ does still exist, we do not attempt to create this ever changing space. We rather make use of the fact that the region inside $D_a(t)$ represents a certain safety risk and then ascertain whether we are inside or outside of this risk space.

We will soon show that the main objective of this work is to find $P_c(t)$ through application of a more rigorous probabilistic approach. Kuchar's work may then be applied in order to determine the necessary alerting thresholds.

2.3.2 Similarity of Collision Avoidance to Alternative Problems

The probabilistic collision avoidance problem can be likened to a number of other research interests in many different fields of study. We will briefly discuss two of these fields, since some of the underlying concepts used to create the approach described in this thesis are borrowed from them. The similarities of the problems are also uncanny and this realization leads to a useful perspective.

Reliability Theory

Reliability theory is employed by a wide range of disciplines, not the least of which is the field of civil engineering[5]. We will discuss how reliability theory may be likened to the collision avoidance problem along the lines of a civil engineering problem.

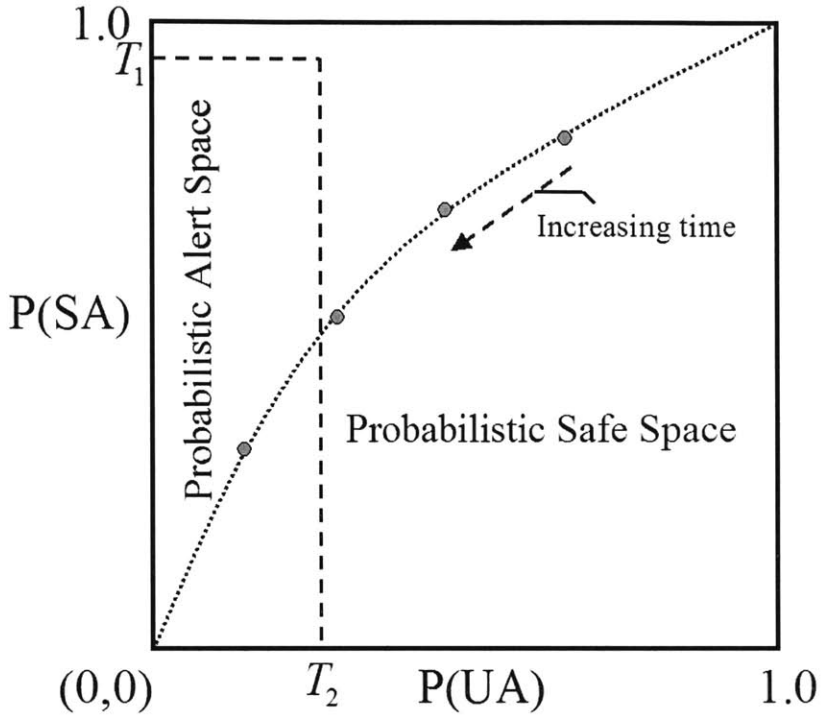

Figure 2-6: An Illustration of the Probabilistic Alert Space

Say that a building is constructed using a multitude of steel and concrete pillars and beams and that these serve as its major load bearing elements. Even though these elements are constructed to exacting specifications, their compression and tensile strengths are characterized in terms of probability distributions. Given that members of different distributions are distributed randomly throughout the construction process, we wish to determine the risk of catastrophic failure due to wind load within the next one-hundred years. Wind speed is usually described as an "Extreme Type" I or II distribution [5].

This is by no means a simple problem and clearly very similar to that of determining collision risk within a finite horizon time window. The problem is solved by creating a state vector containing such information as the locations of load bearing members, the force applied by the wind, soil density and many other relevant parameters. A safe region

within the state space is then defined, called the domain of safety D_s and the complement is referred to as the domain of failure $D_f = D_s^c$. In this case it is the domain of safety that is bounded, as illustrated in Figure 2-7.

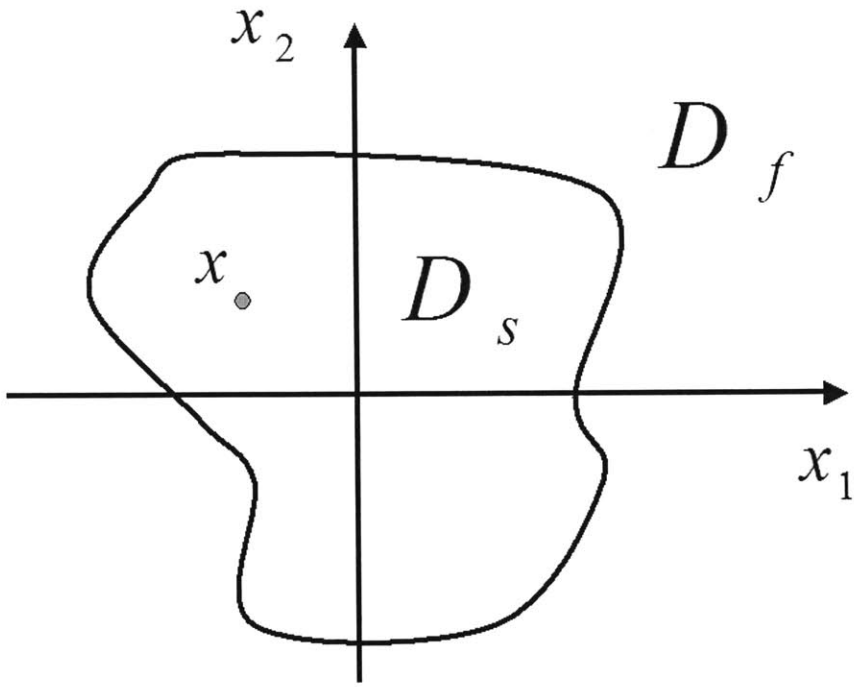

Figure 2-7: A 2-Dimensional Representation of Hazard Space in Reliability Theory

Given the state probability density function (PDF) $f_X(x)$, the probability of catastrophic failure is

$$P_f = 1 - \int_{x \in D_s} f_X(x) dx. \tag{2.4}$$

It is from reliability theory that we borrow such concepts as the domain of failure and some of our notation[5]. It is however no trivial exercise to find $f_X(x)$, as is the case with collision avoidance. The collision avoidance problem is more complex though, since an aircraft does not only possess a state distribution, but also a non-trivial state trajectory. Part of our method of solution, but especially the point of view adopted in this thesis, stems from dividing the collision avoidance problem into a multiple number of reliability theory problems, one at each simulation time step.

The First Passage Time Problem

The first passage time problem was first developed to describe Brownian motion[10] in particle physics. Physicists are interested in understanding the motion of a molecule under the influence of uncorrelated inter-particle collisions. The collisions are usually modelled as transfer of momentum from a Wiener excitation process, and the major interest, as applied to a single dimension of a homogeneous process, is to determine the probability of the molecule remaining within an interval (a, b), or

$$a \leq x \leq b, \tag{2.5}$$

where a and b are absorbing bounds, illustrated in Figure 2-8. An absorbing bound is similar to the boundary of D_f, since entry into this domain negates the possibility of exit from the domain. The developments in Chapter 3 further clarify this statement.

Here we are dealing with a system described by an Ito stochastic differential equation of the form

$$dx(t) = A[x(t), t]dt + \sqrt{B[x(t), t]}dW(t), \tag{2.6}$$

where W(t) is a Wiener process. In Chapter 3 we show how the collision avoidance problem can be described by a multi-dimensional form of Equation 2.6.

Solving for the probability of leaving the interval as a function of time is then similar to solving for the probability of an aircraft entering into D_f, denoted by $P_c(t)$. This is especially evident when one bound, say b, is considered to be very far away and the particle's initial position is much closer to a. $P_c(t)$ can then be solved through application of the backward *Fokker-Planck (Kolmogorov)* equation[28]. This solution, even in one dimension, is of excessive complexity. Desilles[19] provides an example of how this process may be applied to the two-vehicle collision avoidance problem.

The most interesting aspect of the first passage time representation is that it likens the calculation of collision risk to the flow of probability space across a boundary, into a domain of failure, D_f. The risk calculation strategy in Chapters 3 and 4 is developed from the very

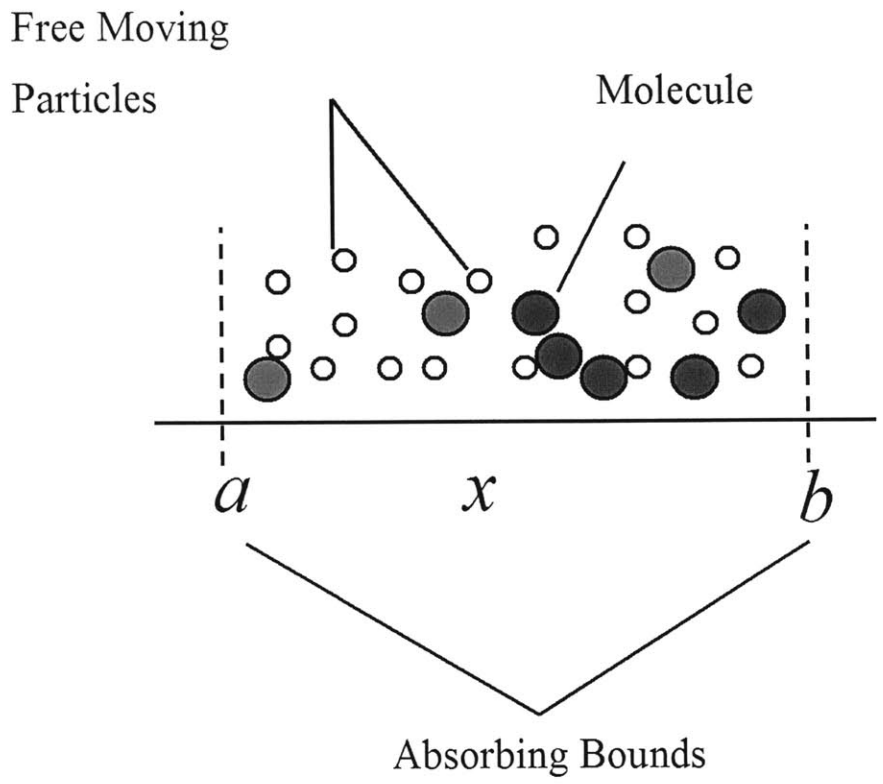

Figure 2-8: An Illustration of the First Passage Time Problem for Two Absorbing Bounds

different standpoint of reliability theory, but also results in a solution which calculates the amount of probability flow into D_f over time.

2.4 Chapter Summary

The autonomous conflict detection and collision avoidance problem is cast into a state space problem, similar to that developed by Kuchar[41]. As with most probabilistic approaches(e.g. [69][19][71]), we assume that the vehicle state may be described by a multivariate normal distribution. This representation culminates with the introduction of probabilities of safe avoidance, $P(SA)$, and unnecessary avoidance, $P(UA)$, respectively.

The calculation of collision risk within a finite time horizon is shown to be equivalent to problems encountered in a number of separate fields of research. Two such fields are:

Reliability theory, especially as applied to civil engineering problems where catastrophic failures are characterized as a function of time; and the first passage time problem described by Kolmogorov equations, or as special cases thereof, such as the Fokker-Planck equation often encountered in physical problems and stochastic control.

Chapter 3

Expanding the Probabilistic State Space Approach

We have shown how the collision avoidance problem may be solved in several ways, with the probabilistic approach being the most accurate representation. In this chapter we show how the probabilistic approaches of Yang and Kuchar [69][41] may be expanded upon in a number of ways. At the same time, we show why the complete probabilistic approach is considered to be intractable for real-time applications. Three major developments are discussed, namely:

1. Real-time future conflict probing may be model-predictive instead of approximating the trajectory as piecewise straight-line segments. We may also treat a vehicle's control strategy and dynamic model as time-variant. These extensions stem from the fact that autonomous vehicles do not suffer from the same extent of uncertainty in future vehicle intent as encountered in piloted flight. The gain lies in propagating conditional mean and variance, thereby reducing the state uncertainty envelope. This aspect is described in Section 3.1.

2. The use of a repeated outcome approach such as Monte-Carlo simulation in the risk probing process suggested by Yang[69] may be replaced by a calculation approach such as propagation of state distribution through a combination of approximation and convolution. Such direct approaches are usually avoided because of the associated high calculation complexity. Paielli and Erzberger avoid this complexity by, for example, assuming constant vehicle velocity and constant state distribution over the conflict

interval. Section 3.1.2 instead deals with solving the problem through a more rigorous calculation approach.

3. After detection of a potential collision, we may delay taking avoidance action until it clearly becomes necessary. The probabilistic effect of such time-delayed avoidance action may be incorporated to reduce the incidence of false alarms by establishing avoidance trajectories that circumnavigate risk peaks in probability space, as described in Section 3.3. This extension builds upon the concept of system operating characteristics (SOCs).

The methods described in this chapter operate within a finite horizon time window from an instantaneous realtime point of implementation, i.e. from the *now*, denoted by t_0 to the *horizon* denoted by t_h, with $t_w = t_h - t_0$ being the time window to the horizon.

3.1 Estimation

3.1.1 State-Space Formulation

Model-prediction allows us to narrow down risk probabilities based on a vehicle's dynamics and control models. Truly effective model-predictive state propagation therefore begs knowledge of a vehicle's closed loop dynamics model and also of the intended trajectory. For these reasons, this extension should be made when a vehicle is under autonomous control along a pre-planned flight path. The research becomes especially applicable to most commercial and military vehicles, especially UAVs, under autopilot control.

Assume that we are dealing either with time-variant linear dynamics or with non-linear dynamics, as long as a linearized system model is available at any time step, and linearization errors are quantifiably small[1]. The following research focusses on describing statistics for a vector valued, linear, time-variant system of the form

$$\begin{aligned} \dot{x}_{OL} &= A(t)x_{OL} + B(t)U_{FC}(t) + B_w(t)w(t) \\ y &= C(t)x_{OL} + D(t)U_{FC}(t) + D_v(t)v(t) \end{aligned} \tag{3.1}$$

[1] An extended non-linear estimation approach may also be used to describe state distribution for highly non-linear systems.

with $w(t)$ and $v(t)$ denoting white process and measurement noise vectors respectively. $U_{FC}(t)$ is the feedback control input to be employed during control loop closure and x_{OL} is the open loop state vector. We will need to estimate the vehicle's state distribution within a finite horizon of interest. At first glance this seems like a standard state estimation problem. There are however some important issues to consider that are discussed in the next few paragraphs.

The vehicle's autopilot, as part of the control system, needs to be included in our representation. Assuming that optimal state estimation is achieved, we may write out the estimator model as

$$\begin{aligned} \dot{\hat{x}}_{OL} &= A(t)\hat{x}_{OL} + B(t)U_{FC}(t) + L(t)(y - \hat{y}) \\ \hat{y} &= C(t)\hat{x}_{OL} + D(t)U_{FC}(t) \end{aligned} \tag{3.2}$$

with $L(t)$ denoting the optimal filter gain. The closed loop system takes on a different form which we will exploit in the next subsection. This form is illustrated using the example of a standard state feedback problem as follows:

Assume that the feedback controller may be modelled as

$$U_{FC}(t) = K(t)\hat{x}_{OL}. \tag{3.3}$$

The complete general model of the closed-loop system may now be constructed from algebraic manipulation and is expressed in terms of its full-state representation

$$\begin{bmatrix} \dot{x}_{OL} \\ \dot{\hat{x}}_{OL} \end{bmatrix} = \begin{bmatrix} A(t) & B(t)K(t) \\ L(t)C(t) & A(t) + B(t)K(t) - L(t)C(t) \end{bmatrix} \begin{bmatrix} x_{OL} \\ \hat{x}_{OL} \end{bmatrix} + \begin{bmatrix} B_w(t) & 0 \\ 0 & L(t)D_v(t) \end{bmatrix} \begin{bmatrix} w(t) \\ v(t) \end{bmatrix}. \tag{3.4}$$

In the above representation, the closed loop system state is described by $x = [x_{OL}, \hat{x}_{OL}]^T$. A more general representation might also include deterministic inputs such as trajectory commands. Such deterministic inputs are by definition uncorrelated with the system states. Dynamic feedback control strategies such as lead and lag networks would also add dimension to the closed loop state vector.

3.1.2 Propagation of State Distributions

We have thus far shown that we are dealing with a describable closed loop system when flying under autopilot control along a planned flight path. The remaining problem is of the form

$$\dot{x} = A(t)x + B(t)U(t) + B_w(t)w(t) \tag{3.5}$$

where $Q_c(t)\delta(t-\tau) \equiv E[w(t)w^T(\tau)]$, $E[w(t)] = 0$, $E[x(0)x^T(0)] = P_0$, $E[x(0)] = 0$, $E[x(0)w^T(t)] = 0$ and $w(t)$ is multi-variate white noise[2]. It is essential to note that, at this stage, $U(t)$ does not represent feedback control, since the control loop was closed in Equation 3.4. Feedback control is denoted slightly differently, by $U_{FC}(t)$. $U(t)$ is a deterministic reference tracking input. According to Gelb[29], with $U(t) = 0\ \forall t$, the mean of x, denoted by $m_X(t)$, is propagated by[29]

$$\dot{m}_{X,U=0}(t) = A(t)m_{X,U=0}(t) \tag{3.6}$$

and the covariance of x, denoted by $P_X(t)$, is propagated by

$$\dot{P}_{X,U=0}(t) = A(t)P_{X,U=0}(t) + P_{X,U=0}(t)A(t)^T + B_w(t)Q_c(t)B_w^T(t). \tag{3.7}$$

The above two propagation equations need to be expanded to account for non-zero reference inputs, i.e., $U(t) \neq 0\ \forall t$. The propagation of the state mean is clearly similar to the solution to $x(t)$, since

$$m_X(t) = m_{X,U=0}(t) + \int_{t_0}^{t} e^{A(\tau)(t-\tau)}B(\tau)U(\tau)d\tau, \tag{3.8}$$

therefore

$$\dot{m}_X(t) = A(t)m_X(t) + B(t)U(t). \tag{3.9}$$

As before, covariance at each time step is expressed as

$$\dot{P}_{X,U\neq 0}(t) = E[(x(t) - m_X(t))(x(t) - m_X(t))^T] \tag{3.10}$$

[2] $E[...]$ is the expected value operator

and since subtracting Equation 3.9 from Equation 3.5 yields

$$\dot{x}(t) - \dot{m}_X(t) = A(t)(x(t) - m_X(t)) + B_w(t)w(t) \tag{3.11}$$

(note the absence of $U(t)$), we may also write

$$\dot{P}_{X,U\neq 0}(t) = A(t)P_{X,U\neq 0}(t) + P_{X,U\neq 0}(t)A(t)^T + B_w(t)Q_c(t)B_w^T(t). \tag{3.12}$$

$U(t)$ becomes the reference path $x_{ref}(t)$, that the vehicle is commanded to follow at any time t. The state will be expressed as a multi-variate normal distribution with mean vector $m_X(t)$ and covariance matrix $P_X(t)$. For this application, true state estimation is not required during the horizon of propagation, since the best estimate of x is not of importance after $t = t_0$, only the instantaneous distribution of x.

In this way, we can now predict the state distribution of the vehicle, based on models of both the autonomous vehicle itself and its scheduled future flight path. This is, by definition, a model-predictive solution. Mean and covariance propagation is a conditional process. Such propagation is not simply an ah-hoc approach to solving the problem of describing the distribution of state over time, but is a natural and proven method in the field of estimation and control. It has also been employed in research by Paielli and Erzberger[49][50] and Prandini et al[54] for collision risk calculation between piloted aircraft. The following deffinition now arises:

Definition: $f_{X(t)}(x)$ is the estimated, time-variant, probability density function of the vehicle state $x(t)$, with t denoting time.

It is very important to realize that the PDF, $f_{X(t)}(x)$, as described in this section, is not dependent on the occurrence (or not) of collisions within the horizon of interest. This is because the described mean and variance propagation processes inherently ignore the occurrence of collisions. We do however employ this propagation method in the reduction process described in Chapter 4. The next order of business is to describe how collision risk may be determined from the knowledge of state distribution (and visa-versa) and the shape of a so-called domain of failure D_f.

3.2 Propagating and Accumulating Risk

3.2.1 The Single-Vehicle-Single-Obstacle Problem

Yang employs a Monte-Carlo approach using piecewise straight-line segments for trajectory models when probing the vehicle state space for possible future conflict. This was necessitated by the calculation complexity involved when attempting to solve for all the complicated probability distributions that arise from the trajectory model, and the computational speed required for real-time application[69][70].

Part of this section describes how complexity arises from the dimensionality of the problem, state-time correlation and especially the conditionality of state distribution on collision events. We recognize that the problem is closely related to the well-known *first passage time* problem of Brownian motion of particles encountered in Physics, as shown in Section 2.3.2. We will show how to perform the required calculations, but Chapter 4 describes how to make the calculations tractable, thereby avoiding the need for Monte-Carlo probes of the probability space. We are constrained to dealing with a two-vehicle problem. Multiple vehicles may for example be dealt with by resolving the problem into two-vehicle pairs, but this concept and the complexities associated with it are considered to be outside the scope of this research.

It is clear that a two-vehicle collision avoidance problem can always be transformed into a single-vehicle-single-obstacle problem. In addition, this obstacle can always be transformed to exist at the state-origin. This can be shown by defining $D_{f_a}(t) \subset I\!R^n$ and $D_{f_b}(t) \subset R^n$ as vehicle-centered domains of failure for two vehicles respectively, and the normally distributed corresponding vehicle states with marginal density functions given by $f_{X_a}(x_a)$ and $f_{X_b}(x_b)$. Defining the relative state difference as $x = x_a - x_b$, the marginal density function for the relative state becomes

$$f_X(x) \sim N(m_{x_a} - m_{x_b}, \Sigma_{x_a} + \Sigma_{x_b}), \tag{3.13}$$

where m_{x_a} and Σ_{x_a} denote the vector mean and covariance matrices of the vector x_a respectively. Paielli and Erzberger[49][50] and Sanders[58] also illustrate this combination of

mean and covariance for the two-vehicle problem. At the same time the domain of failure becomes

$$D_f(t) = D_{f_a}(t) \cup D_{f_b}(t) \subset R^n. \tag{3.14}$$

$D_f(t)$ may be transformed or translated to exist at the state origin. Let us therefore now only solve the problem of having an obstacle of finite size, located at the state origin, called a domain of failure $D_f(t)$ and that an equivalent single vehicle model's marginal state PDF $f_{X(t)}(x)$ can be determined within a time interval of interest, using Equations 3.9 and 3.12.

3.2.2 Accumulation of Risk

The accumulation of risk over time will be explained with the help of the discrete representations in Figures 3-1, 3-2 and 3-3. Figure 3-1 is provided as an illustration of the basic concepts of risk accumulation, and should be kept in mind when later, more mathematical, representations are provided. The state distribution is propagated forward at discrete time intervals, since, as we will soon see, analytical solutions to this procedure do not exist. After every time step, the intersection between the distribution and the hazard is integrated to provide the added risk accumulated over that time step. The section intersecting with the hazard is cut from the distribution, and the remaining distribution is propagated to the next time step, and so forth.

A number of interesting aspects of the risk accumulation procedure should be noted:

1. We are dealing with **flow** of probability space into a hazard, over a short time interval. It is this flow into the hazard that undergoes volumetric integration at each time step and is them added together as time progresses, to provide the total accumulated collision risk up to that time.

2. The hazard absorbs the probability space, since such space that has already collided with the hazard should not be propagated any further. One way to understand the absorbtion (absorbing bound in the First Passage Time problem), is to view the state probability density as a cloud of vehicles and to recognize that once a vehicle within the cloud enters the hazard, it should be removed from the process of counting further collisions. This means that we are dealing with a conditional state distribution at

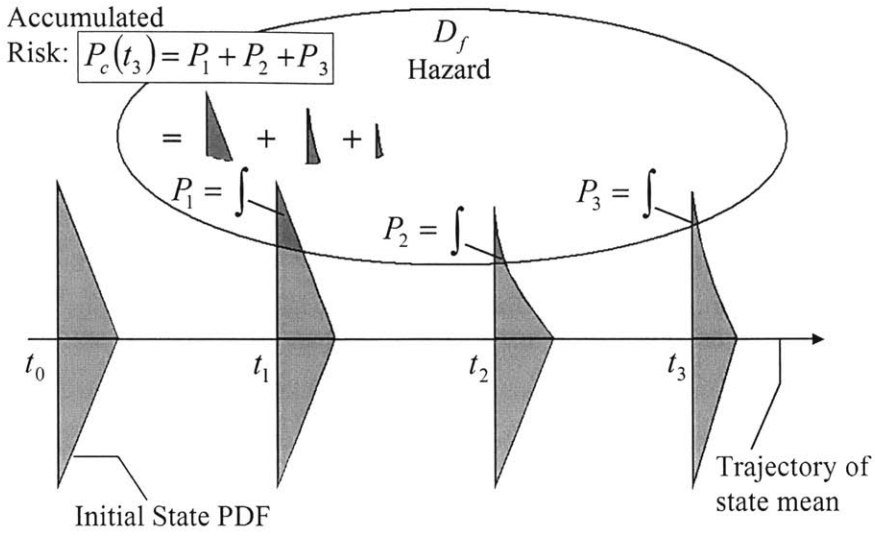

Figure 3-1: A Basic 2-Dimensional Representation of the Risk Accumulation Process, Illustrated With 1-Dimensional Triangular State Distributions, for Simplicty

every time interval and the implications are that: Calculation of the conditional state PDF at every time step becomes a complex task; the probability space diminishes as time progresses; and collision risk increases with exposure time to the hazard

We describe collision risk within the discretized interval $t = [t_0, t_{K-1}]$, where $t_{K-1} = t_h$ is the horizon time and $t = t_0$ is considered to be the present, i.e., the point in time when we need to determine collision risk within the next $t_w = t_h - t_0$ seconds. We now describe the collision probability similarly to the first passage time probability, i.e., the probability of entering the domain of failure by the time $t = t_k$, as

$$\begin{aligned} P_c(k) &= P[x(t) \in D_f(t), \text{ exactly once in } [t_0, t_k]] \\ &= P[x_{t_0} \in D_f(t_0)] + P[x_{t_1} \in D_f(t_1) | x_{t_0} \notin D_f(t_0)] P[x_{t_0} \notin D_f(t_0)] + \\ &\quad P[x_{t_2} \in D_f(t_2) | (x_{t_0} \notin D_f(t_0)) \cap (x_{t_1} \notin D_f(t_1))] P[(x_{t_0} \notin D_f(t_0)) \cap (x(t_0) \notin D_f(t_0))] + \ldots \end{aligned} \tag{3.15}$$

where $x_t \equiv x(t)$ to shorten notation. For the continuous time case, $K \to \infty$ and therefore $\Delta t \to 0$. The continuous case is the multidimensional statement of the first passage time problem discussed in Chapter 2. In Equation 3.15 we assume that the vehicle has not col-

lided before $t = t_0$ and at every time step we need to account for the probability of collision, given that no collision has taken place up to that time.

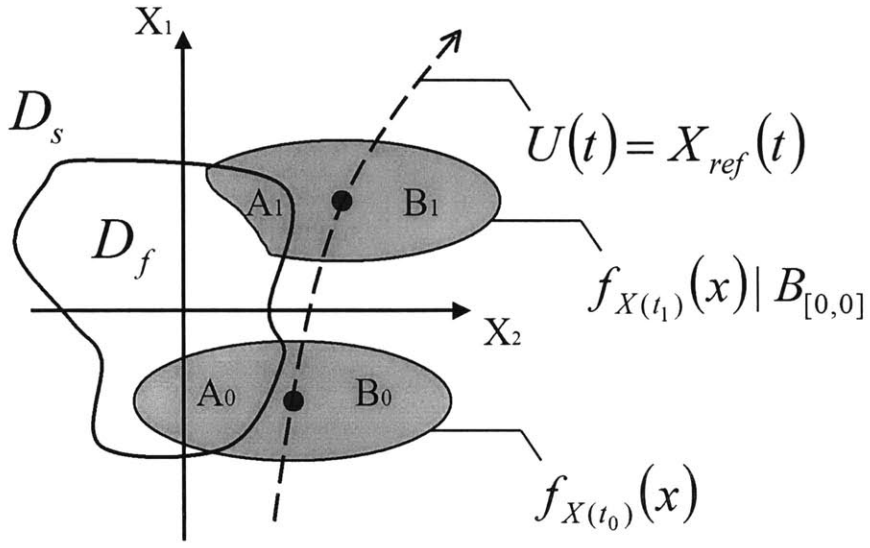

Figure 3-2: Progression of $f_{X_k}(x)|B_{[0,k-1]}$ Over One Time Step from $t = t_0$, in 2 Dimensions

Solving Equation 3.15 in its given form is a complicated task, both intuitively and in terms of calculation complexity. We will write the equation in a different form, by recognizing that we may propagate the probability density function (PDF) describing the state space that remained outside of $D_f(t_k)$ within $[t_0, t_k]$, denoted by $f_{X_k}(x)|B_{[0,k-1]}$. This is the conditional form of $f_{X_k}(x)$, given that no collision has taken place before or on $t = t_k$.

Definition: $f_{X(t_k)}(x)|B_{[0,k-1]}$ is the time-variant PDF of the state $x(t)$ at the discrete time $t = t_k$, conditioned on the fact that no collision has occurred while $t \in [t_0, t_{k-1}]$. The conditional statement is expressed as $B_{[0,k-1]}$.

At time $t = t_0$, the start of the horizon of propagation, the conditional PDF is equivalent to the marginal PDF given by

$$f_{X(t_0)}(x) = g_{A_0}(x) + g_{B_0}(x), \tag{3.16}$$

where $g_{A_0}(x)$ and $g_{B_0}(x)$ are the segments of the PDF inside and outside of the domain of $D_f(t_0)$ respectively. At time $t = t_0 + k.\Delta t$, k discrete time steps later, the equation becomes

$$f_{X(t_k)}(x)|B_{[0,k-1]} = g_{A_k}(x) + g_{B_k}(x). \tag{3.17}$$

When we write Equation 3.15 in terms of Equations 3.16 and 3.17 we can express the probability of collision at the discrete time $t = t_k$ as

$$\begin{aligned} P_c(k+1) &= \int_{x \in D_f(t_0)} g_{A_0}(x)dx + \int_{x \in D_f(t_1)} g_{A_1}(x)dx[1 - P_c(0)] + ... \\ &\quad \int_{x \in D_f(t_k)} g_{A_k}(x)dx[1 - P_c(k-1)] + \int_{x \in D_f(t_{k+1})} g_{A_{k+1}}(x)dx[1 - P_c(k)] \\ &= P_c(k) + \int_{x \in D_f(t_{k+1})} g_{A_{k+1}}(x)dx[1 - P_c(k)]. \end{aligned} \tag{3.18}$$

In this way we have shown that $P_c(k)$ may be determined through an incremental update at each time step from $t = t_{k-1}$ to $t = t_k$, with $P_c(k < 0) = 0$, provided that $g_{A_k}(x)$ can be found for $0 \leq k \leq (K-1)$.

Finding $g_{A_k}(x)$ entails expressing $f_{X(t_k)}(x)|B_{[0,k-1]}$ and salvaging the segment of $f_{X(t_k)}(x)|B_{[0,k-1]}$ inside $D_f(t_k)$, according to Equations 3.16 and 3.17. We therefore focus on calculating $f_{X(t_k)}(x)|B_{[0,k-1]}$.

At time $t = t_0$ we start with the marginal PDF $f_{X(t_0)}(x)$ and then need to find $f_{X(t_1)}(x)|B_{[0,0]}$ and so forth. In general we are calculating $f_{X(t_{k+1})}(x)|B_{[0,k]}$ when $f_{X(t_k)}(x)|B_{[0,k-1]}$ is known and $f_{X(t_0)}(x)|B_{[0,-1]} = f_{X(t_0)}(x)$. To do this we need to understand how x_{k+1} can be realized from x_k. Assuming Euler integration, we can write

$$\begin{aligned} x_{k+1} &= x_k + \Delta t \dot{x}_k \\ &= [I + \Delta t A(k)]x_k + [\Delta t B(k)u(k) + \Delta t B(k)w(k)] \\ &= M_k(x_k) + R_k[u(k), w(k)] \end{aligned} \tag{3.19}$$

where $M_k(x_k)$ and $R_k[u(k), w(k)]$ are independent quantities. This independence allows us

to apply n-dimensional convolution to find

$$f_{X_{k+1}}(x) = f_{M_{t_k}}(m) * f_{R_{t_k}}(r). \tag{3.20}$$

In the same way, we may propagate $f_{X(t_k)}(x)|B_{[0,k-1]}$ by setting

$$\begin{aligned} f_{X(t_{k+1})}(x)|B_{[0,k]} &= \frac{g_{B_k}(x)}{\int_{x \notin D_f(t_k)} g_{B_k}(x)} * f_{R_{t_k}}(r) \\ &= \frac{g_{B_k}(x)}{1 - \int_{x \in D_f(t_k)} g_{A_k}(x)} * f_{R_{t_k}}(r) \\ &= f^{B_k}_{X(t_k)}(x) * f_{R_{t_k}}(r), \end{aligned} \tag{3.21}$$

where

$$f^{B_k}_{X(t_k)}(x) \equiv \frac{g_{B_k}(x)}{1 - \int_{x \in D_f(t_k)} g_{A_k}(x)}, \tag{3.22}$$

and with $P_c(-1) = 0$. In Equation 3.21 $g_{B_k}(x)$ represents the segment of $f_{X(t_k)}(x)|B_{[0,k-1]}$ outside of $D_f(t_k)$ and $\int_{x \notin D_f(t_k)} g_{B_k}(x)$ normalizes $g_{B_k}(x)$ to become a true PDF, in accordance with Bayes' rule. It is essential to realize that only that part of $f_{X(t_k)}(x)|B_{[0,k-1]}$ outside of $D_f(t_k)$ must be propagated, since $f_{X(t_{k+1})}(x)|B_{[0,k]}$ is by definition conditioned on $B_{[0,k]}$. No closed-form representation of the convolution in Equation 3.21 is however known and the computational complexity of a numerical solution is $\mathcal{O}(N^{2n})$, with N being the number of quantizations of each dimension of the space and $n = dim(x)$.

3.2.3 Propagation Procedure

So far we have made assumptions about the nature of the system under consideration, such as it being linear, time-varying. We are also dealing with a one-vehicle-one-obstacle problem with normally distributed states. We have shown that this is equivalent to a two-vehicle problem, and have described the relative state and domain of failure in a way showing similarity to that found in the work of Paielli and Erzberger[49] and Sanders[58]. Given these assumptions and accepting that the vehicle has either undergone no collision before $t = t_0$ or that we can cast probability of collision as the probability of collision within $[t_0, t_h]$, then the derivations so far from Equations 3.13 to 3.21 are accurate discrete time representations.

Figure 3-3 illustrates how the exact discrete time probability of collision accumulated

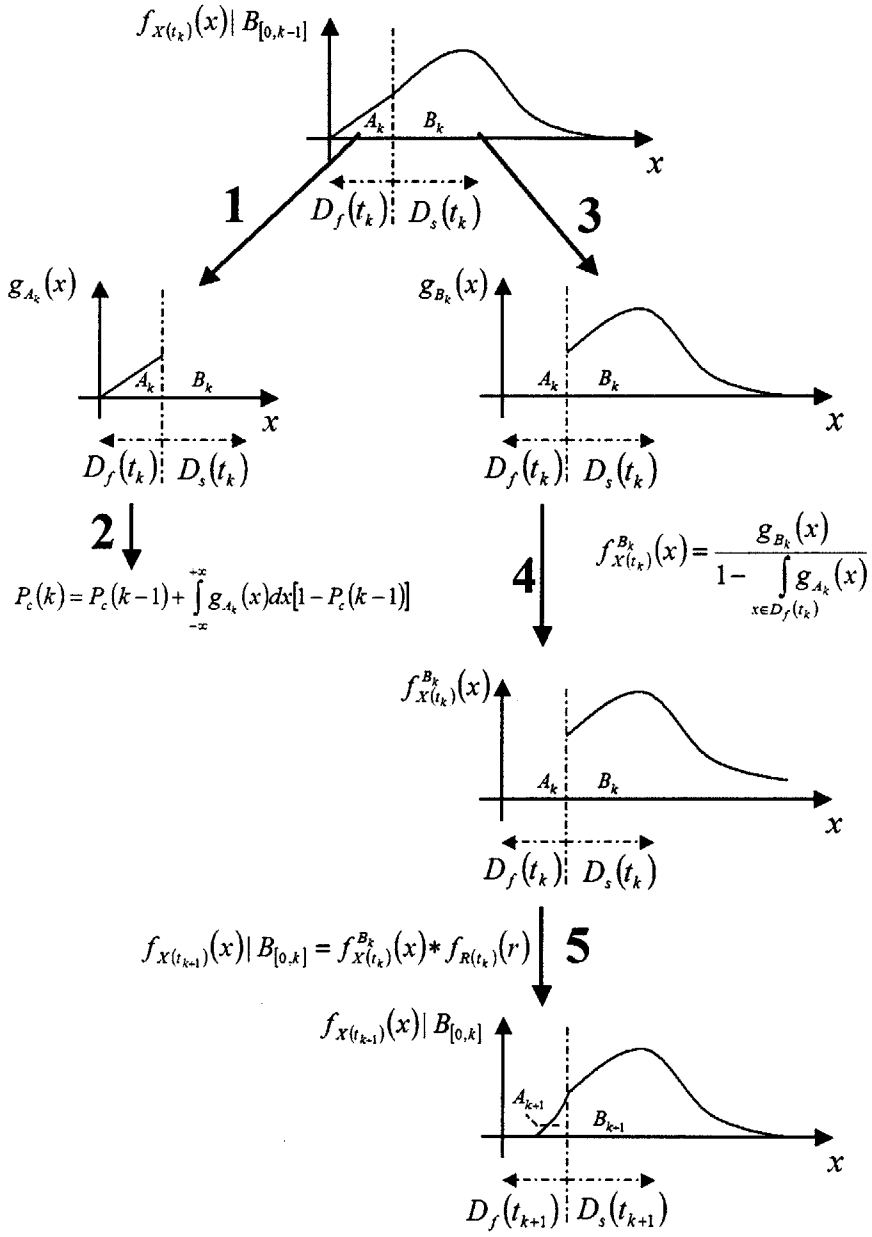

Figure 3-3: 5-Step $f_{X(t_{k+1})}(x)|B_{[0,k]}$ Propagation and Calculation of $P_c(k)$ (1-Dimensional)

within $[t_0, t_k]$, denoted by $P_c(k)$, may be found through a 5-step propagation process. The illustration is provided for a one-dimensional problem in order to reinforce understanding of the process. Notice that the propagation process requires knowledge of $P_c(k-1)$ in order to find $P_c(k)$. This implies that the 5-step process needs to be applied incrementally for each time step in $[t_0, t_k]$, starting at t_0 and in increasing order of k. Recall that $f_{X(t_0)}(x)|B_{[0,-1]} = f_{X(t_0)}(x)$ and $P_c(-1) = 0$. At every time step $t = t_k$,

Step 1: Segment $g_{A_k}(x)$ from $f_{X(t_k)}(x)|B_{[0,k-1]}$ by setting $f_{X(t_k)}(x)|B_{[0,k-1]} = 0 \; \forall \; x \notin D_f(t_k)$.

Step 2: Use $P_c(k) = P_c(k-1) + \int_{x \in D_f(t_k)} g_{A_k}(x)dx[1 - P_c(k-1)]$ to accumulate the probability of collision.

Step 3: Segment g_{B_k} from $f_{X(t_k)}(x)|B_{[0,k-1]}$ by setting $f_{X(t_k)}(x)|B_{[0,k-1]} = 0$ $\forall \; x \in D_f(t_k)$.

Step 4: Normalize g_{B_k} to create the PDF $f^{B_k}_{X(t_k)}(x) \equiv \frac{g_{B_k}(x)}{1 - \int_{x \in D_f(t_k)} g_{A_k}(x)}$.

Step 5: Propagate $f^{B_k}_{X(t_k)}(x)$ to the next time step through the n-dimensional convolution $f_{X(t_{k+1})}(x)|B_{[0,k]} = f^{B_k}_{X(t_k)}(x) * f_{R_{t_k}}(r)$.

Now continue the process by utilizing $f_{X(t_{k+1})}(x)|B_{[0,k]}$ at time $t = t_{k+1}$ to obtain $g_{A_k}(x)$ as in Step 1, and so forth.

The numerical complexity of the propagation phase is dominated by the $\mathcal{O}(N^{2n})$ convolution process indicated in Step 5 and described by Equation 3.21, and is therefore an intractable problem for meaningful values of n and N. Chapter 4 deals with order reduction metrics and approximation techniques that make this calculation tractable.

3.3 Time-delayed Avoidance

3.3.1 System Operating Characteristic

The performance of a collision avoidance system may be characterized as a trade-off between unnecessary alerts and successful alerts. This method of performance evaluation is based on the use of the System Operating Characteristic (SOC) curve, developed by Kuchar[41][42]. The curve facilitates a visual exchange between $P(SA)$ and $P(UA)$ as the time of implementation of a single avoidance maneuver is varied. It is assumed that no collision has

taken place before the time corresponding to any point on the SOC. The SOC curve is always specific to the avoidance maneuver and the geometry of the encounter, see Figure 3-4. The top left corner of the SOC figure represents perfect, safe avoidance strategies with zero probability of false alarm. Curves that approach this corner therefore allow for improved alerting thresholds. The opposite is true for the bottom right corner of the SOC figure.

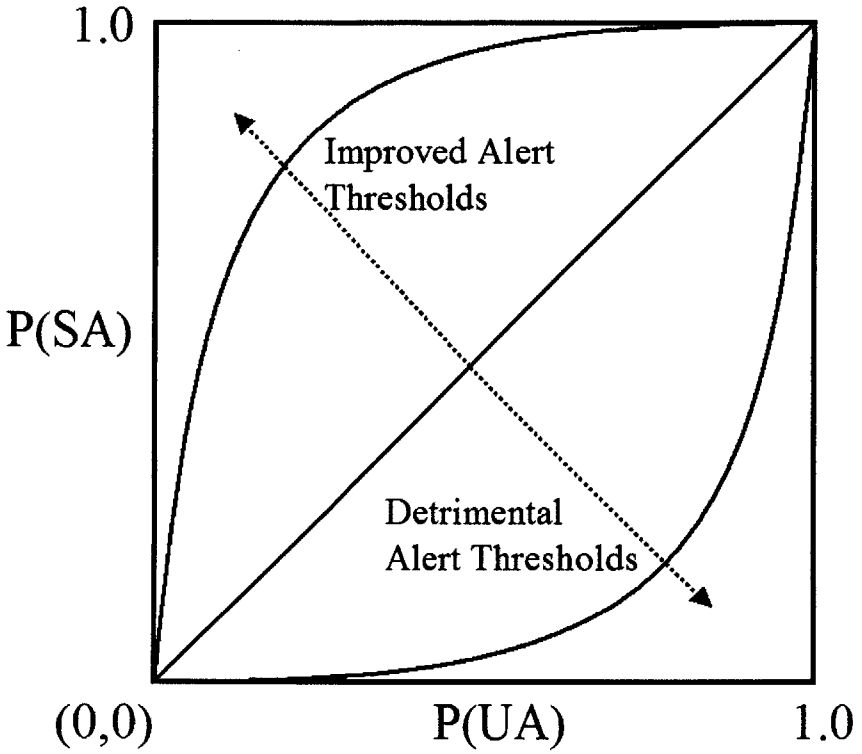

Figure 3-4: Different Shapes of SOC Curves

SOCs are not created for $t > t_0$ when attempting real-time conflict alerting, because of the uncertainty in the future intent of most vehicles and the method of creating estimates of $P(SA)$ and $P(UA)$. This means that an SOC curve is mapped out as time passes, but the most futuristic point on the curve is at $t = t_0$, the present. This point, by definition, evaluates the performance of immediately employing a collision avoidance maneuver. Real-time collision avoidance strategies such as [70] and [69] make use of this most recent point on the SOC curve to decide on an alert eventuality. Earlier in this thesis, we have

shown that vehicles under deterministic and stable autopilot control allow us to propagate their state PDF into the future, without the severe increase in state uncertainty usually associated with unknown pilot intent. Section 4.3.1 describes how the finite horizon time window cannot however exceed t_d, a bound based on our knowledge of intent, since intent is never known exactly within an un-bounded period of time. Therefore, for this discussion, we will predict the shape of the SOC curve for $t_0 \leq t \leq t_0 + t_w$, where $t_w = t_h - t_0 \leq t_d$.

3.3.2 Condensed SOC

Unfortunately the shape of SOC curves do not always follow a simple path to the origin, as usually depicted for vehicles on an inevitable collision course, seen in Figure 3-4. Various examples of collision avoidance exist that result in more complex SOC curve representations. One such example is illustrated in Figure 3-5. In this case, a vehicle is flying directly towards a domain of failure, and employs an avoidance maneuver that only temporarily ventures off the nominal trajectory. The avoidance maneuvers are investigated t_{AD} seconds apart.

A more complex SOC curve results from the example in Figure 3-5, illustrated in Figure 3-6. Typical alerting thresholds such as $(P(SA) < T_1) \cap (P(FA) < T_2)$ would alert at time $t = t_0$ when the curve is only known up to $t \leq t_0$. Without certainty of the curve at $t > t_0$, an alert would be issued. This is the standard approach to application of the SOC curve [41],[70]. A more complicated scenario, such as the given example, can however be envisaged where the curve later exits the alert-space demarcated in Figure 3-6. A finite horizon simulation is able to draw the curve up to $t = t_h$, therefore predicting whether such an exit will occur within the finite time window.

A number of interesting questions now arise. First, should the alert be issued upon entering the alert-space even if an exit is foretold? Second, how long and how far inside the alert-space may a vehicle remain with a degree of safety? It is ventured that an SOC curve alone cannot answer these questions, since future probabilities of collision are time-correlated, as described in the previous section and should therefore be conditional on previous collision risk.

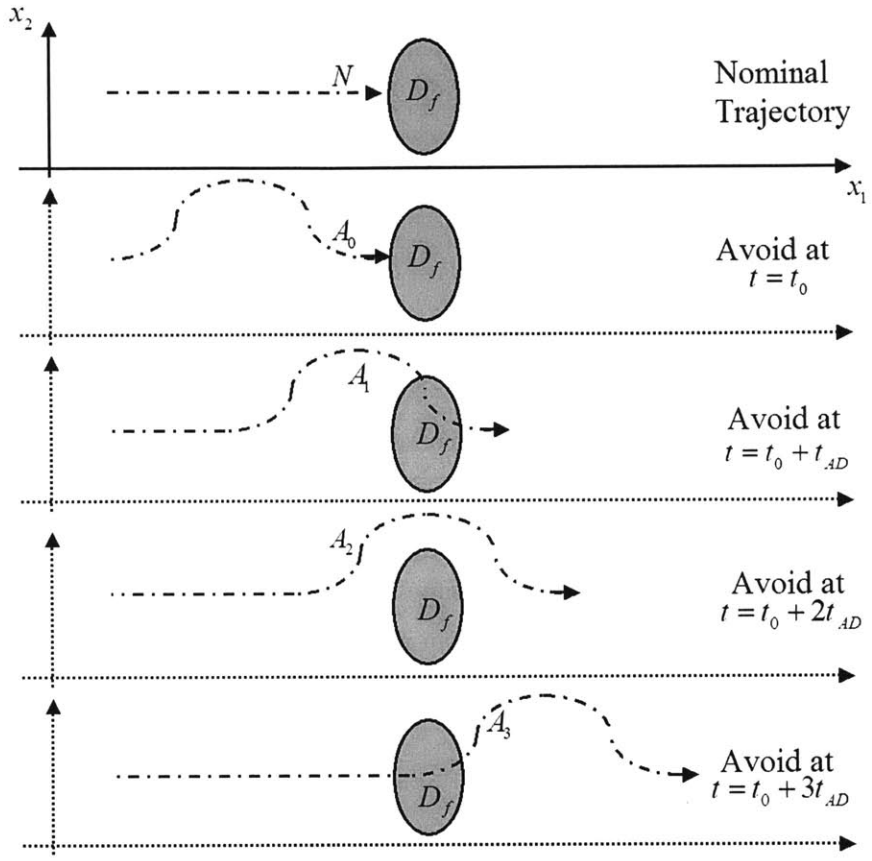

Figure 3-5: An Example Showing the Benefit of Delayed Avoidance Action

Let us assume that a vehicle may either continue along a nominal trajectory N or employ a single avoidance strategy A at any time t. Continuous time delayed avoidance may then be viewed as the creation of an infinite set of compound trajectories where a vehicle continues along, for example, trajectory N for $0 \leq t_N \leq t_d$ seconds and then branches onto A for the next $0 \leq t_A \leq t_d - t_N$ seconds. We now discretize the number of opportunities to transition from N to A, and prevent the vehicle from transitioning back to N. Each transition time is designated as illustrated in Figure 3-7. In this way, any one of the discrete compound trajectories may be defined as

$$DCT_i = [N(0, t_i), A(t_i, t_d)]. \tag{3.23}$$

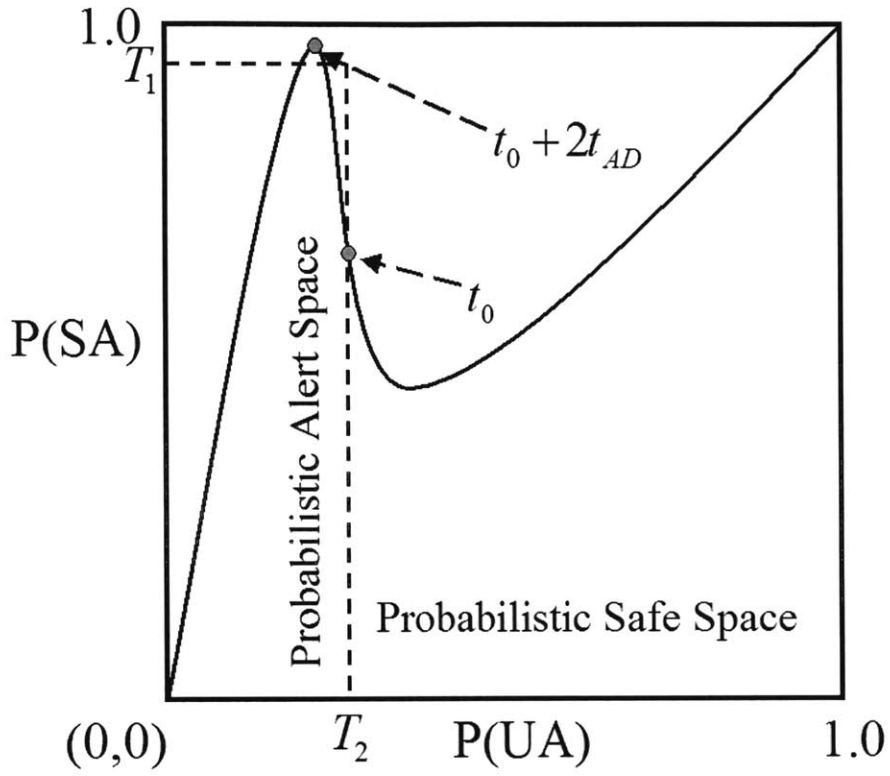

Figure 3-6: The Unpredictable Nature of the SOC Curve for $t > t_0$

The avoidance discretization step size will be denoted by $t_{AD} = t_1 - t_0$ and the number of steps are denoted by $I = t_d/t_{AD} + 1$. In this way we may describe $(I - 1)$ SOCs, where $P(UA)$ is derived from the probability of collision of the nominal path $N = DCT_I$, and each other DCT_i within $i = [0, I - 2]$. The issue of the size of t_{AD} is considered in Section 4.3.2.

Now we may create a new kind of SOC, the Condensed SOC (CSOC), which corresponds to that single one of the $(I - 1)$ SOC curves with the highest $P(SA)$ at $t = t_0$. Notice that the $P(UA)$ of each SOC is exactly equal at $t = t_0$. In this way, the CSOC provides the avoidance option with the highest $P(SA)$ while condensing the outcome of the next t_d seconds into a single point at $t = t_0$. We no longer have to deal with the issues of an SOC possibly entering and exiting the alert-space within a window of time, but simply need to ask if the CSOC is inside or outside of the alert-space when deciding to alert or not. Recall that the "alert" referred to here really is the decision to let the autopilot of

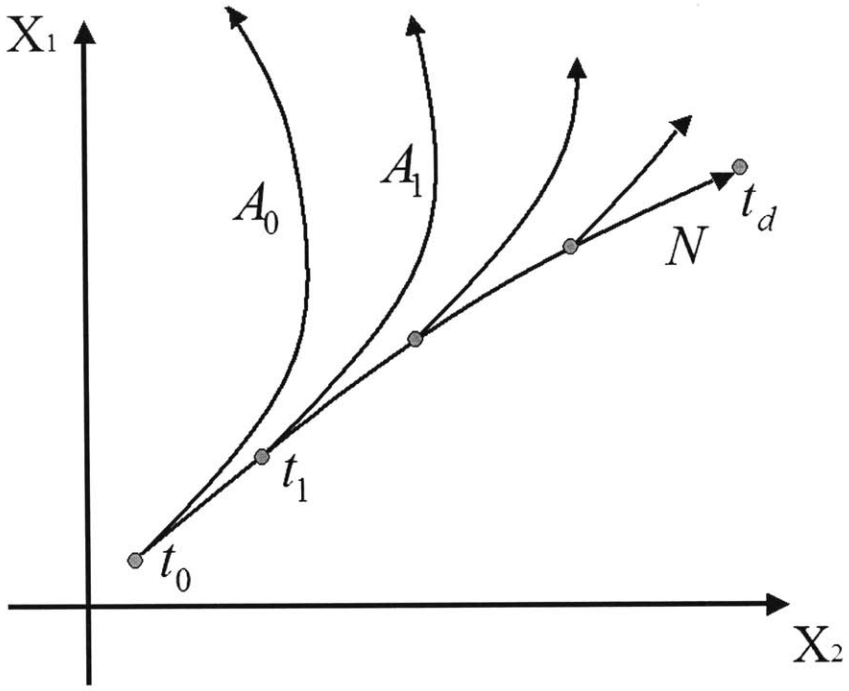

Figure 3-7: Creation of Compound Avoidance Maneuvers

our vehicle implement an avoidance maneuver. Notice also that the avoidance maneuver is delayed by t_{i*} seconds, where $i*$ corresponds to the designation of the selected SOC that became the CSOC. The avoidance maneuver associated with the CSOC now becomes the suggested avoidance maneuver if an alarm is issued.

It is worth reminding the reader that the collision avoidance problem solved in this thesis is a conservative one where only very small amounts of collision probability are tolerated. In this sense the alert-space bounded on the CSOC curve would be described by $(P(SA) < T_1) \cap (P(FA) < T_2)$, where $T_1, T_2 \geq (1 - P_{c_{max}})$ and where $P_{c_{max}}$ is the maximum allowable collision risk that a vehicle may be exposed to within any finite horizon interval. The alerting aspect of the decision making process does not need to know exactly where a point on the SOC curve is, only which side of the alert-space boundary it is on. We therefore only calculate $P_c(k)$ until $P_c(k) \geq P_{c_{max}}$ becomes true.

The CSOC reflects a change in philosophy from the original SOC in terms of what the metrics for decision making should be. An SOC essentially employs the metric of "needing to alert" vs. the "success of alerting now". In the case of the CSOC we employ the metrics "need to alert" vs. "safety", where "safety" accounts for the possible benefit of delaying avoidance action now, while acting later. The CSOC concept can therefore be viewed as an extension of Yang's work on selecting the most beneficial avoidance maneuver from within a set of avoidance maneuvers, at $t = t_0$[69].

The concept of a CSOC is created to link this dissertation with the works of Kuchar and Yang and to be able to apply existing methods of analysis and alerting threshold design to the time-delayed avoidance problem. The rest of the thesis, including application examples, will however focus mainly on the calculation of $P_c(t)$ (in the form of either $P_c^N(t)$ or $P_c^A(t)$) so that SOCs may be created according to Equations 2.2 and 2.3. For trade-off between $P(SA)$ and $P(UA)$ in alerting threshold design we refer the reader to [42],[41] and [69].

It is also worth pointing out that we do not really need to create I number of SOC solutions before being able to find the CSOC. The principles of dynamic programming can be employed to find the SOC with the highest $P(SA)$.

3.4 Chapter Summary

The development in this chapter rests on three major assumptions, namely that: A vehicle may be modelled using time-variant, linear dynamics; collision risk only extends from the present into the future, that is, collision risk only makes sense when calculated after $t = t_0$, the start of the model-predictive, finite horizon simulation window; and Euler integration is used in order to time-discretize the intractable, continuous-time collision risk calculation problem.

The autonomous vehicle problem is extended to a closed loop representation where it is shown that mean and covariance propagation may be used to estimate future vehicle state uncertainty through finite horizon model predictive simulation. This development is a direct consequence of the reduction of uncertainty of pilot intent because of autonomous

control. Paielli and Erzberger[50] and Prandini et al[54], among others, also illustrate a similar development, but for straight line vehicle trajectories.

The calculation of collision risk within a finite and discretized time horizon is cast as a reliability theory problem and developed from first principles. The solution is developed as a set of discrete propagation equations illustrated in Figure 3-3 and it is shown that the calculations are numerically intractable when real-time solutions are required.

The notion of the system operating characteristic (SOC)[42][41] employed as a trade-off between $P(SA)$ and $P(UA)$ when making alerting decisions is extended. The SOC at any point in time is based on the outcome of making an avoidance decision at that time. It is shown that the effects of time-delayed avoidance action may be incorporated into the SOC representation, resulting in a best-case or time condensed SOC (CSOC), for a specific time horizon and a given set avoidance trajectories. This allows us to make use of signal detection theory in order to optimize alerting thresholds, while at the same time investigating the pay-off of delayed avoidance action. As a direct consequence, the work of Kuchar may be employed when optimizing alerting thresholds[41].

Chapter 4

Tractable Risk Propagation Through Quadratic Collision Metrics

4.1 Order Reduction of Convolution

The intractability of the n-dimensional numerical convolution required to propagate a vehicle's state PDF arises mostly from the $\mathcal{O}(N^{2n})$ computational complexity involved in solving Equation 3.21, where N represents the number of samples along one dimension of the discrete numerical solution grid. At first it also seems that the n-dimensional integration of Equation 3.18 could potentially add additional $\mathcal{O}(N^n)$ complexity to the solution. Fortunately this integral may be solved through a gradient search approach of $\mathcal{O}(nN)$[5].

In the event of no collision risk within $[t_0, t_h]$, no convolution would be required and the mean and covariance propagators described in Equations 3.9 and 3.12 would provide an optimal estimate of the vehicle's state PDF at any time within the horizon. Unfortunately, resolving $f_{X(t_k)}(x)|B_{[0,k-1]}$ into $g_{A_k}(x)$ and $g_{B_k}(x)$, segments within and without of $D_f(t_k)$ respectively, adds complexity. This is because only $f^{B_k}_{X(t_k)}(x)$ derived from $g_{B_k}(x)$ (see Equation 3.21) must be propagated forward in time and it is no longer a normal distribution for which special properties might be exploited to negate the need for the application of n-dimensional convolution. Hence, analytical solutions to Equation 3.18 and 3.20 are un-

known. This chapter focusses on the application of intuitively natural collision metrics in order to reduce the $\mathcal{O}(N^{2n})$ complexity of risk calculation to $\mathcal{O}(Nlog_2(N))$ through system order reduction to $n = 1$ and the application of efficient numerical convolution techniques.

4.1.1 Simplifying Assumptions

Existing collision avoidance systems such as TCAS define a so-called Distance Modification (DMOD), a collision safety bound (distance)[67][48] around an aircraft which corresponds to a radius of conflict. TCAS then predicts the amount of time it would take any vehicle to close to within DMOD. It then issues a *Traffic Advisory* (TA) when it is predicted that an aircraft will close to within DMOD, within τ_1 seconds. A second level of alert, a *Resolution Advisory* (RA) is issued when $\tau_2 < \tau_1$ becomes the predicted time to conflict. The DMOD distance essentially defines a equivalent domain of failure around a vehicle.

It is in the spirit of TCAS, its well-established track record and its wide acceptance as a standard for comparison[67][41][69][48][24][25][56] that we choose a similar safety bound or DMOD surrounding a vehicle. Many other systems with this approach are cited in [43]. It will be shown that this tried and tested bound may in fact be shaped into a useful order reductive metric. We also generalize the metric to shapes with varying radius around the vehicle, by defining it as a quadratic function of the vehicle's state described by

$$R = \sqrt{x^T B x}, \tag{4.1}$$

where B is constrained to be a real, symmetric positive definite matrix.

The square of the R metric is more useful, since the square root operator in Equation 4.1 spawns considerable numerical and analytical complexity when dealing with the calculation of PDFs. The metric therefore becomes:

$$\gamma = R^2 = x^T B x. \tag{4.2}$$

For typical spherical buffers around vehicles where position is a simple state, B only retains some unitary diagonal entries. The focus of the next section lies in reducing the required

convolution order of the n-dimensional state distribution by applying this one-dimensional metric. Soon we will need to find the probability distribution of γ and the lack of a square root operator in Equation 4.2 makes this feasible. The collision safety bound (the boundary of $D_f(t)$) of an aircraft must be expressed in the form of Equation 4.2 in order to apply the algorithms developed in this dissertation.

4.1.2 Mathematical Reduction

A Simplified Propagation Equation

Define γ_0 as the safety buffer threshold (the square of the DMOD) or the boundary of D_f, i.e., if

$$\gamma \leq \gamma_0, \tag{4.3}$$

then collision has occurred. Equation 4.3 now becomes the reductive collision metric of interest, since we are dealing only with a one-vehicle-one-obstacle problem as described in Section 3.2. We aim to determine the probability of satisfying this inequality within $[t_0, t_k]$, which is equivalent to evaluating the probability of collision, $P_c(k)$. The full-state representation of Equation 3.19 may be replaced by the single-state equivalent

$$\begin{aligned} \gamma_{k+1} &= \gamma_k + \Delta t \dot{\gamma}_k \\ &= \gamma_k + \Delta \dot{\gamma}_k, \end{aligned} \tag{4.4}$$

where $\Delta t \dot{\gamma}_k$ will be referred to as $\Delta \dot{\gamma}_k$, simply to avoid unnecessarily complicated future notation issues. In addition, it will be shown at a later stage that $\Delta \dot{\gamma}_k$ can also be conveniently expressed as a quadratic form of x (see Subsection 4.1.3).

The simple summation of random variables in Equation 4.4 leads us to also express γ_{k+1} as a random variable. In order to do this we recognize from Figure 4-1 that the cumulative distribution function (CDF) of γ_{k+1} is described by $F_{\Gamma_{k+1}}^{B_{k+1}}(\gamma) = P[\Gamma_{k+1} < \gamma] | B_{[0,k]}$.

In this way, the marginal PDF of γ_{k+1} may be described exactly by[52]

$$f_{\Gamma_{k+1}}(\gamma)|B_{[0,k]} = \frac{d}{d\gamma} \int_{-\infty}^{+\infty} \int_{-\infty}^{\gamma-\psi} f_{\Gamma_k, \Delta\dot{\Gamma}_k}^{B_k}(\psi, \phi) d\phi d\psi, \tag{4.5}$$

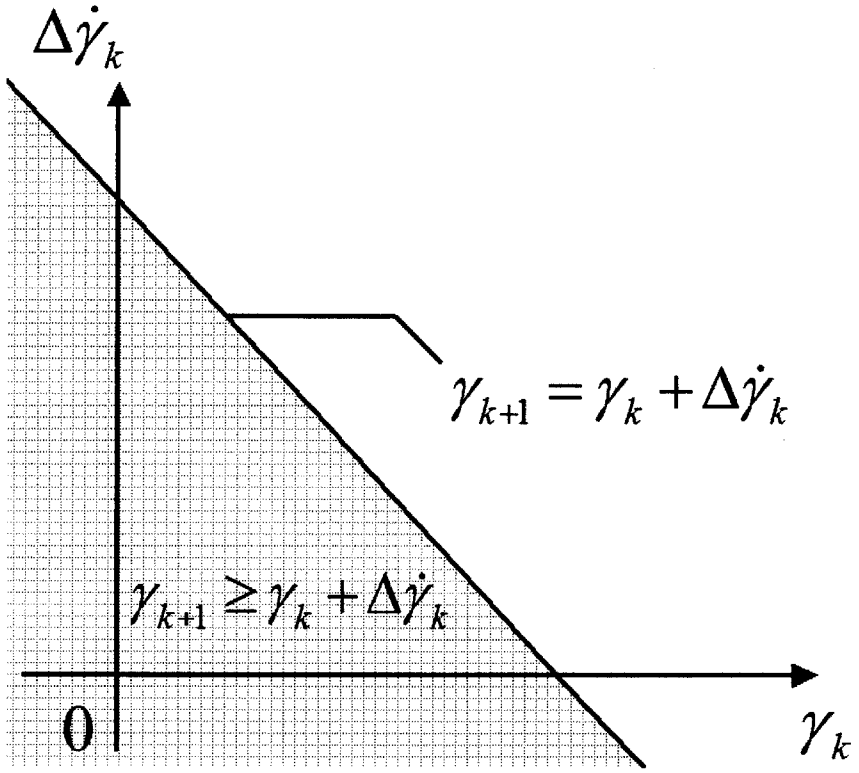

Figure 4-1: Region in the γ_k-$\Delta t\dot{\gamma}_k$ Plane where $\gamma_{k+1} \geq \gamma_k + \Delta t\dot{\gamma}_k$, Used to Describe $F_{\Gamma_{k+1}}(\gamma)$

where

$$f^{B_k}_{\Gamma_k,\Delta\dot{\Gamma}_k}(\gamma, \Delta\dot{\gamma}) \equiv \frac{d^2}{d\gamma d\Delta\dot{\gamma}} F^{B_k}_{\Gamma_k,\Delta\dot{\Gamma}_k}(\gamma, \Delta\dot{\gamma}) \tag{4.6}$$

is the joint PDF of γ and $\Delta\dot{\gamma}$ at $t = t_k$, given that no collision has occurred within $[t_0, t_k]$. Also, ψ and ϕ are simply variables of integration.

4.1.3 Simplifying Approximations

The reductive metric results in Equations 4.4 and 4.5, which are simpler to deal with than their full-state n-dimensional counterparts. There are however two issues that cannot be circumvented without approximation:

1. The double-integral of Equation 4.5 needs to be approximated numerically with complexity of $\mathcal{O}(N^4)$, unless γ and $\Delta\dot{\gamma}$ are independent, in which case complexity of $\mathcal{O}(N^2)$ results.

2. No solution is known to the joint PDF in Equation 4.6. Numerical approximation would be of $\mathcal{O}(N^{2n+2})$ complexity, since we would need to revert back to the full-state representation and employ e.g. a Monte-Carlo method to solve.

Approximation will need to fulfill at least two primary requirements:

1. The calculation order complexity of Equations 4.5 and 4.6 needs to be reduced significantly. Most digital signal processors (DSPs) are optimized for $\mathcal{O}(N^2)$ or less complexity, since this is widely considered to be the feasible extent of tractability and applicability for image processing applications. We will show how such DSP techniques may be used to solve our risk calculation problem and therefore aim at reducing our complexity to $\mathcal{O}(N^2)$ or less.

2. Approximation errors need to be characterized to a sensible extent. This would imply that we need to estimate or exactly determine the sizes of all sign-indefinite[1] errors. Sign-definite[2] errors at least need to be clamped between upper and lower bounds, with an emphasis on the calculation of the upper bound. Upper bounds on, for example, P_c estimates translate into reduced estimates of $P(SA)$ and $P(UA)$ and therefore lead to a more conservative and usually safer alerting approach for autonomous vehicles.

It is clear that the major issue of complexity is the calculation of Equation 4.6. Recall that Equation 3.15 in the previous chapter provides the discrete representation of $P_c(k) \equiv P_c(t_k)$. The aim is to find an upper bound to the calculation performed there-in that would provide an equivalent expression to Equation 4.6, but of considerably reduced complexity. To this effect we approximate

$$\begin{aligned}
P_c(k) &= P[x(t) \in D_f(t), \text{ exactly once in } [t_0, t_k]] \\
&= P[x(t_0) \in D_f(t_0)] + P[x_{t_1} \in D_f(t_1) | x_{t_0} \notin D_f(t_0)] P[x(t_0) \notin D_f(t_0)] + \\
&\quad P[x_{t_2} \in D_f(t_2) | (x_{t_0} \notin D_f(t_0)) \cap (x_{t_1} \notin D_f(t_1))] P[(x(t_0) \notin D_f(t_0)) \cap (x(t_0) \notin D_f(t_0))] + \ldots \\
&\approx P[x(t_0) \in D_f(t_0)] + P[x_{t_1} \in D_f(t_1) | x_{t_0} \notin D_f(t_0)] P[x(t_0) \notin D_f(t_0)] + \\
&\quad P[x_{t_2} \in D_f(t_2) | x_{t_1} \notin D_f(t_1)] P[x(t_1) \notin D_f(t_1)] + \ldots \\
&= P[x(t) \in D_f(t), \text{ at least once in } [t_0, t_k]]
\end{aligned} \tag{4.7}$$

[1] Neither positive or negative definite, i.e., the sign is unknown
[2] Either positive or negative definite, i.e., the sign is known

in accordance with such texts as [65],[61],[60],[15],[16] and [64]. This provides an upper bound approximation to $P_c(k)$, since

$$P[x(t) \in D_f(t), \text{ at least once in } [t_0, t_k]] \geq P[x(t) \in D_f(t), \text{ exactly once in } [t_0, t_k]]. \quad (4.8)$$

The approximation of Equation 4.7 allows us to employ unconditional means to calculate Equation 4.6, such that

$$f^{B_k}_{\Gamma_k, \Delta\dot{\Gamma}_k}(\gamma, \Delta\dot{\gamma}) = \frac{u(\gamma - \gamma_0) f_{\Gamma_k, \Delta\dot{\Gamma}_k}(\gamma, \Delta\dot{\gamma})}{1 - \int_{-\infty}^{+\infty} \int_0^{\gamma_0} f_{\Gamma_k, \Delta\dot{\Gamma}_k}(\gamma, \Delta\dot{\gamma}) d\gamma d\Delta\dot{\gamma}} \quad (4.9)$$

where $u(\gamma - \gamma_0)$ is a unit step function and $f_{\Gamma_k, \Delta\dot{\Gamma}_k}(\gamma, \Delta\dot{\gamma})$ is the joint PDF of γ and $\Delta\dot{\gamma}$ at $t = t_k$, regardless of any collision activity having taken place within $[t_0, t_k]$.
$1 - \int_{-\infty}^{+\infty} \int_0^{\gamma_0} f_{\Gamma_k, \Delta\dot{\Gamma}_k}(\gamma, \Delta\dot{\gamma}) d\gamma d\Delta\dot{\gamma}$ is simply a scalar normalization according to Bayes' rule.

One more approximation is made, this time to reduce the complexity of Equation 4.5: Assume that γ and $\Delta\dot{\gamma}$ are independent within any one time step $[t_k, t_{k+1}]$. It is essential to note that this is very different from assuming than γ and $\dot{\gamma}$ are independent within $[t_0, t_h]$, as will be shown shortly.

Our independence assumption and assumption on the calculation of $P_c(k)$, combined with the application of Leibniz's rule [52] allows us to re-write Equation 4.5 as

$$\begin{aligned} f_{\Gamma_{k+1}}(\gamma)|B_{[0,k]} &\approx f_{\Gamma_{k+1}}(\gamma)|B_{[k,k]} \\ &= f^{B_k}_{\Gamma_k}(\gamma) * f_{\Delta\dot{\Gamma}_k}(\Delta\dot{\gamma}), \end{aligned} \quad (4.10)$$

where

$$f^{B_k}_{\Gamma_k}(\gamma) = \frac{u(\gamma - \gamma_0) f_{\Gamma_k}(\gamma)}{1 - \int_0^{\gamma_0} f_{\Gamma_k}(\gamma) d\gamma} \quad (4.11)$$

and $f_{\Gamma_k}(\gamma)$ and $f_{\Delta\dot{\Gamma}_k}(\Delta\dot{\gamma})$ are the marginal PDFs of γ and $\Delta\dot{\gamma}$ respectively, without conditioning on any collision event. The usual normalization factor is again present in the denominator.

It is Equation 4.10, a one-dimensional convolution, combined with the ability to operate on the unconditional marginal PDFs of γ and $\Delta\dot{\gamma}$ to find $P_c(k)$, that makes the solution to this problem tractable. It is shown in Section 4.1.3 that the required marginal PDFs may be calculated for a quadratic form of normally distributed state variables.

Figure 4-2 shows how to go about solving for $P_c(k)$ at each time step $t = t_k$. This figure may be contrasted to the exact full-state calculation illustrated in Figure 3-3.

Assuming that the unconditional marginal PDFs of γ and $\Delta\dot{\gamma}$ are known, the process of finding $P_c(k)$ and propagating $f_{\Gamma(t_{k+1})}(\gamma)|B_{[k,k]}$ may be divided into the following steps:

Step 1: Find $g_{A_k}(\gamma)$, the segment of $f_{\Gamma_k}(\gamma)|B_{[k-1,k-1]}$ for which $\gamma \leq \gamma_0$.

Step 2: Make use of $P_c(k) = P_c(k-1) + \int_0^{\gamma_0} g_{A_k}(\gamma)d\gamma[1 - P_c(k-1)]$ to accumulate the probability of collision.

Step 3: Apply Equation 4.11 to find $f_{\Gamma_k}^{B_k}(\gamma)$. Note that this step does not rely on steps 1 and 2.

Step 4: Apply Equation 4.10 to propagate $f_{\Gamma_k}^{B_k}(\gamma)$ into $f_{\Gamma_{k+1}}(\gamma)|B_{[k,k]}$.

Now continue by returning to Step 1 for calculation of $P_c(k+1)$ and so forth.

A clear segmentation between the above outlined steps is evident. This is because Equation 4.7 allows us to express Equations 4.8 and 4.11 as functions of unconditional marginal PDFs that are independent of the conditional operations at the previous time $t = t_{k-1}$. The conditional PDF, $f_{\Gamma_{k+1}}(\gamma)|B_{[k,k]}$, in Step 4 is created by assuming independence between γ and $\Delta\dot{\gamma}$ within only one time step. This segmentation leads to a very useful result: Even though errors resulting from the independence assumption do accumulate in the calculation of $P_c(k)$, these errors are not propagated into the creation of $f_{\Gamma_{k+1}}(\gamma)|B_{[k,k]}$ at any next time step $t = t_k$, thereby resulting in a convergent rather than divergent approximation of $P_c(k)$ as $t \to t_h$.

An alternate view of the apparent segmentation is that the independence assumption is only employed over the course of one time step and only to determine the amount of flow of probability into $D_f(t_k)$. The seemingly simpler route of assuming independence within the entire interval $[t_0, t_h]$ and using it to propagate $f_{\Gamma_{k+1}}(\gamma)|B_{[0,k]}$ through the resulting simple

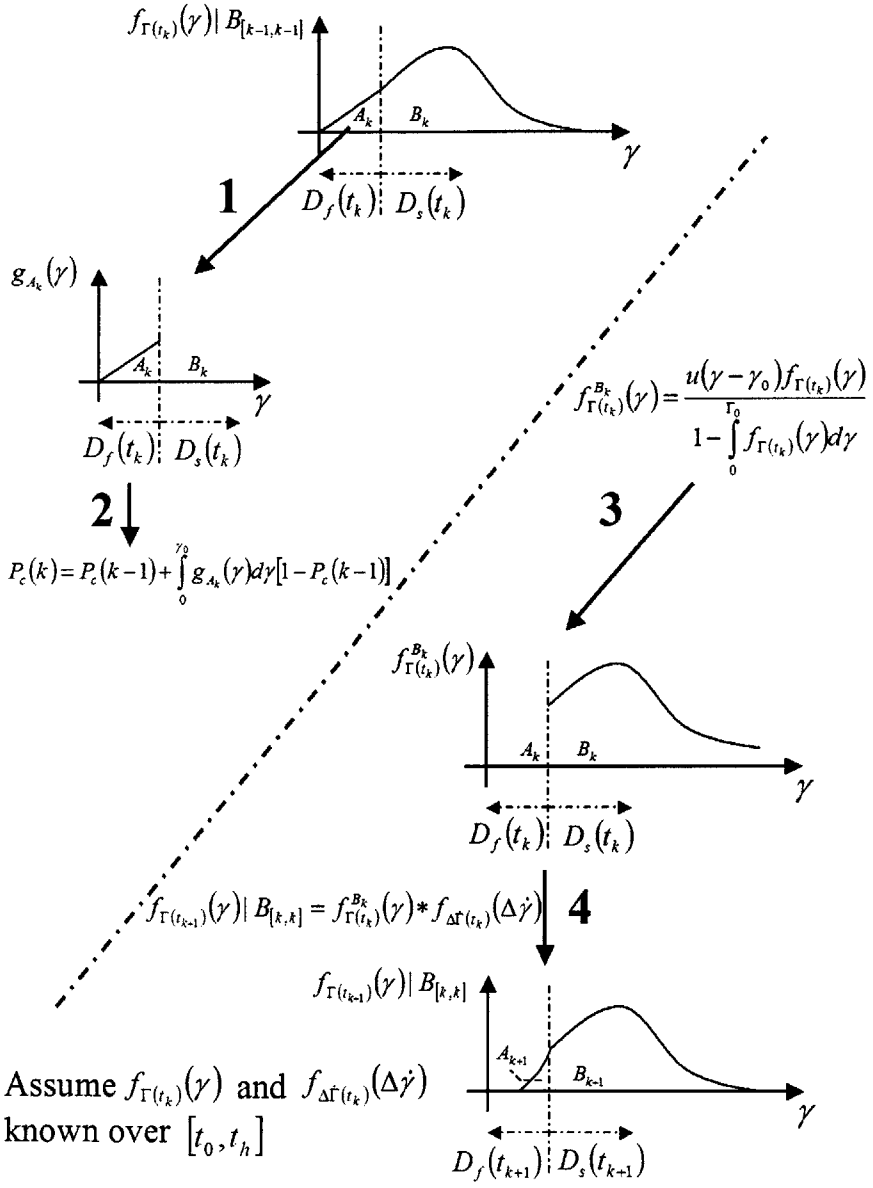

Figure 4-2: 4-Step $f_{\Gamma(t_{k+1})}(\gamma)|B_{[k,k]}$ Propagation and Calculation of $P_c(k)$

one-dimensional convolution, is folly, since convolution always leads to an increase in the variance of the PDF. In reality, the variance of a quadratic metric should reduce drastically when its mean approaches γ_0, because the chi-square distribution narrows when it nears the origin.

Errors resulting from the two approximations made in this section are investigated in Section 4.2.

Methods of Solution

The focus now becomes finding a solution to $f_{\Gamma_k}(\gamma)$ and $f_{\Delta\dot{\Gamma}_k}(\Delta\dot{\gamma})$ within $[t_0, t_h]$. We will only show how to solve for $f_{\Gamma_k}(\gamma)$ and eventually prove that $f_{\Delta\dot{\Gamma}_k}(\Delta\dot{\gamma})$ may be solved in the same way.

Other than the useful special cases discussed in Subsection 4.3.3, the PDF of a general quadratic form of normally distributed random variables needs to be approximated. Methods of approximation include series expansion, numerical Fourier inversion, Monte-Carlo and particle-based filtering simulations and approximations based on moment matching techniques. Moment matching through processes such as *Cornish-Fisher*, *Gram-Charlier* and *Edgeworth* Expansion[35][45][6] do not provide accurate results when tails of PDFs are of value and provide little or no guarantee on rates of convergence relative to the number of cumulants matched[35]. In this case Monte-Carlo techniques are computationally expensive when exploring high-dimensional probability spaces, again because of the high accuracy required when representing the tails of estimated PDFs, as with most problems in risk and reliability theory. It is partly for this reason that Yang [69] also performs a dimensional reduction to vehicle heading, before attempting Monte-Carlo simulation.

Fourier Inversion emerges as the salient choice, since it turns out that the inversion process can be greatly simplified for the quadratic collision metric. When applying Fourier Inversion we would first need to solve for the marginal characteristic function of γ given by

$$\phi_{\Gamma_k}(\tau) = E[e^{j\tau\gamma}] = \int_{-\infty}^{+\infty} e^{j\tau\gamma} f_\Gamma(\gamma) d\gamma \tag{4.12}$$

and then solve for the *Fourier Inversion Integral* (FII)

$$f_{\Gamma_k}(\gamma) = \frac{1}{2\pi}\int_{-\infty}^{+\infty} \phi_{\Gamma_k}(\tau)e^{-jw\gamma}d\tau. \tag{4.13}$$

Characteristic Function

Solving for a characteristic function can in itself be a daunting task. The next few paragraphs however show that a closed form solution to $\phi(\tau)$ does exist in an analytical form, when dealing with the proposed quadratic metric.

First we transform γ into an appropriate form using the substitution $y = (x - m_X)$ where m_X is the vector mean representation of x. In this way we find that

$$\begin{aligned} \gamma &= x^T Bx \\ &= m_X^T Bm_X + 2m_X^T By + y^T By \\ &= \Theta + \Delta^T y + \frac{1}{2}y^T My. \end{aligned} \tag{4.14}$$

When we solve the *generalized eigenvalue problem* such that

$$\begin{aligned} CC^T &= \Sigma_X \\ C^T MC &= \Lambda \end{aligned} \tag{4.15}$$

where $\Lambda = diag(\lambda_1, ..., \lambda_n)$, then we can substitute $y = Cz$ and $\delta = C^T\Delta$ into Equation 4.14 such that

$$\begin{aligned} \gamma &= \Theta + \delta^T z + \frac{1}{2}z^T\Lambda z \\ &= \Theta + \sum_{i=1}^{n}(\delta_i z_i + \frac{1}{2}\lambda_i z_i^2) \end{aligned} \tag{4.16}$$

where $z = (z_1, ..., z_n)^T$ is a standard normal vector and $\delta = (\delta_1, ..., \delta_n)^T \in \Re^n$. Now we reshape $\phi_{\Gamma_k}(\tau)$ into

$$\begin{aligned} \phi_{\Gamma_k}(\tau) &= E[e^{j\tau\gamma}] \\ &= \int_{-\infty}^{+\infty} e^{j\tau\gamma} f_{\Gamma_k}(\gamma)d\gamma \end{aligned}$$

$$= e^{j\tau\Theta} E[e^{j\tau(\gamma-\Theta)}] \tag{4.17}$$

and recognize that $E[e^{j\tau(\gamma-\Theta)}] = \phi_{\chi^2}(\tau)$ where $\phi_{\chi^2}(\tau)$ is the characteristic function of a non-central Chi-square (χ^2) variate in z, where z is in standard normal form. $\phi_{\chi^2}(\tau)$ is known analytically[34] and when substituted into Equation 4.17 $\phi_{\Gamma_k}(\tau)$ becomes

$$\phi_{\Gamma_k}(\tau) = e^{j\tau\Theta} \prod_{i=1}^{n} \frac{1}{\sqrt{1-j\lambda_i\tau}} e^{-\frac{1}{2}\delta_i^2\tau^2/(1-j\lambda_i\tau)}, \tag{4.18}$$

an analytical form to be substituted into Equation 4.13, requiring no numerical approximation. The computational complexity of function evaluations in Equation 4.18 is $\mathcal{O}(nN)$ and that of the eigenvalue decomposition (Equation 4.15) is in general $\mathcal{O}(n^3)$. The latter may be reduced to $\mathcal{O}(n)$, as described in [18].

Fourier Inversion Integral

Various options exist when attempting to solve the inverse Fourier integral (Equation 4.13) using the characteristic function in Equation 4.18. The first is to perform brute force numerical integration and to reduce the discrete step size of t and w until the error on the calculated distribution of γ is negligible. This is not difficult but requires computational cost $\mathcal{O}(N^2)$. Care must however be taken to reduce the error of this calculation to be negligible when compared with the small amount of allowable maximum probability of collision defined in the previous chapter. In this case, the key to error analysis of Equation 4.13 because of discrete integration is the Poisson summation formula[34].

The second numerical method of calculation is more powerful and less intensive and amounts to the use of FFT algorithms in order to reduce computational cost from $\mathcal{O}(N^2)$ to $\mathcal{O}(Nlog_2(N))$. As is the usual practice in engineering, we define the *Fourier Transform* (FT) as [55]

$$\mathcal{F} : f(x) \mapsto F(u) = \int_{-\infty}^{+\infty} e^{-j2\pi ux} f(x)dx. \tag{4.19}$$

Substituting $u = \frac{-t}{2\pi}$ in Equation 4.19 we can show that

$$F(u) = \phi_{\Gamma_k}(-2\pi u). \tag{4.20}$$

We now attempt to find $f(x) = f_{\Gamma_k}(\gamma)$ by *Inverse Fourier Transform* (IFT), which is clearly equivalent to applying Equation 4.13, the FII. All numerical solutions to IFT are approximate, unless $f(x)$ is continuous in x, periodic with period T and band limited to frequencies $|u| < N/2T$ where the integer N represents the number of discrete samples. We do not expect $f(x)$ to be periodic, since, by definition, $\int_{-\infty}^{+\infty} f(x)dx = 1$.

Hughett [32] shows that FFT approximation of $f(x)$ can now be accomplished by first splitting it into $f(x) = f_c(x) + C$, where $f_c(x)$ is absolutely integrable and C is a constant. This split is done to avoid singularity problems when applying the FT. We now focus our attention on

$$f_c(x) = \int_{-\infty}^{+\infty} F_c(u) e^{j2\pi ux} du \tag{4.21}$$

which can be approximated as a sum of rectangles with width $1/T$ to obtain

$$f_c(x) = \sum_{k=-\infty}^{\infty} F_s[k] e^{j2\pi kx/T}, \tag{4.22}$$

the *Fourier Series* (FS) representation of $f_c(x)$, with

$$F_s[k] = \frac{F_c(k/T)}{T}. \tag{4.23}$$

Both Proakis [55] and Hughett [32] show how the solution of FS through

$$f(x) \approx g(x) = \sum_{k=-N/2}^{N/2} G[k] e^{j2\pi kx/T}, \tag{4.24}$$

where $G[k]$ is defined by

$$G[k] = \begin{cases} F(0)/T + C & \text{for } k = 0, \\ F(k/T)/T & \text{for } 0 < |k| < N/2, \\ 0 & \text{for } |k| = N/2, \end{cases} \tag{4.25}$$

provides the most accurate known numerical approximation of $f(x)$. Notice that Equation 4.24 is usually defined [55] as the *Inverse Discrete Fourier Transform* (IDFT) and that it may be solved using the Radix-2 *Inverse Fast Fourier Transform* (IFFT) algorithm, as long as N can be written in the form $N = 2^E$ where E is a positive integer. IFFTs are

accomplished with at most $N/2log_2(N)$ multiplications and $Nlog_2(N)$ additions, thereby reducing computational complexity to $\mathcal{O}(Nlog_2(N))$.

Concerning $\Delta\dot{\gamma}$

In previous sections we have shown that $f_{\Gamma_k}(\gamma)$ can be calculated within $[t_0, t_h]$, since γ can be converted into the required quadratic form. It turns out that $\Delta\dot{\gamma}_k$ can also be expressed in this quadratic form as shown in the following derivation, where B is symmetric:

$$\begin{aligned}
\gamma &= x^T Bx \\
\Delta\dot{\gamma} &= \Delta t\dot{x}^T Bx + \Delta t x^T B\dot{x} \\
&= 2\Delta t\dot{x}^T Bx \\
&= 2\Delta t[Ax + B_u u(k) + B_w w(k)]^T Bx \\
&= 2\Delta t x^T A^T Bx + 2\Delta t u^T(k) B_u^T Bx + 2\Delta t w^T(k) B_w^T Bx \\
&\sim x^T Nx + Sx, \qquad (4.26)
\end{aligned}$$

with $N = 2\Delta t A^T B$ and $S = 2\Delta t(u^T(k)B_u^T B + w^T(k)B_w^T B)$, and where Δt is the time step, such that $\Delta t = t_{k+1} - t_k$.

Equation 4.26 would be in a quadratic form similar to Equations 4.2 and 4.16, but for the term $2\Delta t w^T(k)B_w^T Bx$ which contains the random variable $w(k)$, white noise. This single term would make the dimensional and computational reduction discussed so far intractable again, unless we impose the constraint that

$$B_w^T B = [0]. \qquad (4.27)$$

At first glance this constraint seems excessive, but it simply means that no states that form part of the calculation of γ may be directly excited by white noise. In other words, B may not contain non-zero terms which operate directly on states excited by white noise. Collision bounds on position (as is usually the case) would require that position and velocity be part of the state vector. Even the simplest Newtonian models would include white noise excitation at acceleration or jerk level, not velocity and position. This means that the constraint usually holds for typical vehicle models. If, for some reason, the constraint

does not hold naturally, white noise can be shaped with a high bandwidth low pass filter in order to add another state buffer between any state required in the calculation of $\Delta\dot{\gamma}$ and the noise excitation.

We may therefore calculate $f_{\Delta\dot{\Gamma}_k}(\Delta\dot{\gamma})$ within $[t_0, t_h]$ from the PDF of x described by propagation of Equations 3.9 and 3.12. This is done in the same way as outlined for $f_\Gamma(\gamma)$ in the preceding sections. Also, recall that Equation 4.2 was created for B belonging to the class of real, symmetric, positive definite matrices. $\Delta\dot{\gamma}$ is clearly not sign-definite, since γ may increase and decrease in value. The only constraint on B employed in the calculation of $f_\Gamma(\gamma)$ is that it must be a real matrix. Symmetry and positive definiteness were only exploited while finding $P_c(k)$ and proving that $\Delta\dot{\gamma}$ is a quadratic form.

Lastly, in Section 4.2 we require that $\Delta\dot{\gamma}$ be written exactly in the complete square form described by Equation 4.2 when calculating the covariance between γ and $\Delta\dot{\gamma}$ in Equation 4.39. The covariance calculation affects the estimate of calculation error because of the independence assumption. It is not immediately obvious that we may always write $\Delta\dot{\gamma}$ in the required form and this will not be proven true or false. It will however be stated that $\Delta\dot{\gamma}$ can be written in the complete square quadratic form whenever the only non-zero elements in B are on its diagonal and all the derivatives of states operated upon by non-zero entries in B are themselves inside the state vector.

This will be illustrated with a simple example: Say we have four states

$$x = \begin{bmatrix} x & \dot{x} & y & \dot{y} \end{bmatrix}^T \tag{4.28}$$

and $\gamma = ax^2 + by^2$, a complete square quadratic form

$$\gamma = x^T \begin{bmatrix} a & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & b & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} x, \tag{4.29}$$

where only diagonal elements of B are non-zero. This leads us to write $\dot{\gamma} = 2ax\dot{x} + 2by\dot{y}$,

which is also a complete square quadratic form

$$\dot{\gamma} = x^T \begin{bmatrix} 0 & a & 0 & 0 \\ a & 0 & 0 & 0 \\ 0 & 0 & 0 & b \\ 0 & 0 & b & 0 \end{bmatrix} x. \tag{4.30}$$

This example also illustrates how the $B_w^T B = [0]$ constraint holds. If the vehicle undergoes noise excitation in each dimension at the acceleration level then

$$B_w = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix}, \tag{4.31}$$

and therefore

$$B_w^T B = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}. \tag{4.32}$$

Concerning Convolution

Even though Equation 4.10 only requires one-dimensional convolution, this still results in $\mathcal{O}(N^2)$ computational complexity. Convolution can however be accomplished by computing the characteristic functions of each of γ_k and $\Delta\dot{\gamma}_k$ and multiplying these (not convolving) before applying the FII [52]. $\phi_{\Delta\dot{\Gamma}_k}(\tau)$ is determined analytically by Equation 4.18 already and would require no additional computation. $f^B_{\Gamma_k}(\gamma)$ however needs to be calculated within $[t_0, t_{K-1}]$ by applying Equation 4.11 at each time step. It can be shown that $\phi^B_{\Gamma_k}(\tau)$ (the characteristic function of $f^B_{\Gamma_k}(\gamma)$) can be determined in a similar way to that described when computing $f_{\Gamma_k}(\gamma)$, but with an FFT algorithm [32]. The numerical complexity thus again reduces to $\mathcal{O}(Nlog_2(N))$.

4.2 Induced Error on Probability Calculations

We may differentiate among two different types of errors when calculating $P_c(k)$. The first is approximation error, discussed in Section 4.2.1, where simplifying assumptions lead

us to calculate an approximate solution. The second induced error is that of numerical calculation, discussed in Section 4.2.2, where we solve for the approximate solution to a specific accuracy. Both these types of errors have very different origins and therefore require very different methods of characterization.

4.2.1 Approximation Error

Approximation error may be further sub-divided into two major attributing factors. First we find an upper bound approximation to $P_c(k)$ (Equation 4.8) and second we assume a measure of independence between γ_k and $\Delta\dot{\gamma}_k$ when performing this calculation (Equation 4.10), so our upper bound is itself calculated with an amount of error. We calculate a signed calculation error because of the independence assumption, in order to narrow down the envelope of error and more closely approximate the true value of $P_c(k)$. Figure 4-3 illustrates the relationships between various collision probability calculations and their associated errors.

Two additional, and comparatively minor, sources of error that also require discussion are those induced by quadratic approximation of the collision bound and also that of time discretization. These issues are treated at the end of this section.

Upper bound Probability Error

Section 4.1.3 describes why the calculated $P_c(k) = P_c^{ub}(k)$ is an upper bound to the true value denoted by $P_c^{true}(k)$, with "ub" signifying "upper bound". It is infeasible to attempt to solve directly for the amount of error between the calculation and the true value, since the true value would need to be derived from an intractable calculation. We define the error as

$$P_c^{ube}(k) = P_c^{ub}(k) - P_c^{true}(k) = P_c(k) - P_c^{true}(k) \tag{4.33}$$

with "ube" signifying the "upper bound error".

Instead of venturing along the path of intractability, we rather create a lower bound to $P_c^{true}(k)$ which we shall call $P_c^{lb}(k) \leq P_c^{true}(k)$. Now use the lower bound to re-write

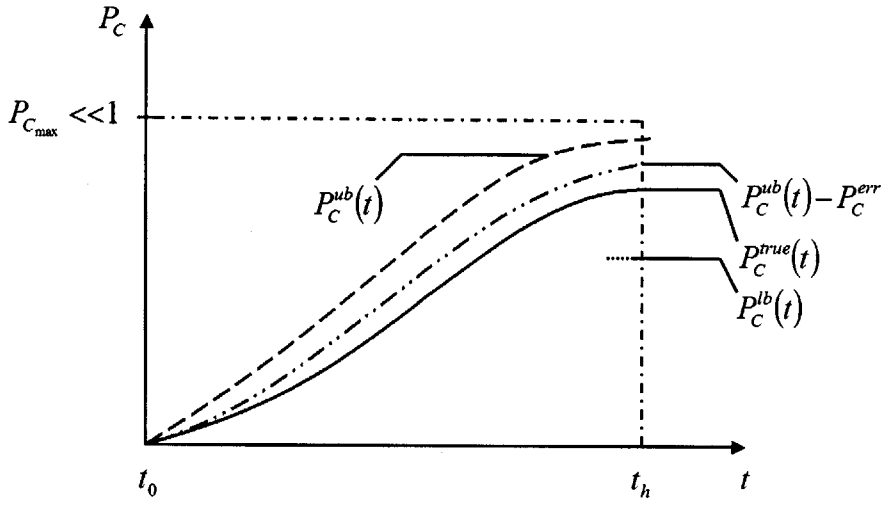

Figure 4-3: Relationships Between $P_c^{true}(t)$, $P_c^{ub}(t)$, $P_c^{err}(t)$ and $P_c^{lb}(t)$

Equation 4.33 as

$$P_c^{ube}(k) \leq P_c(k) - P_c^{lb}(k). \tag{4.34}$$

A feasible lower bound would allow us to clamp the true probability of collision between two calculable limits. One such intuitive and simple lower bound often used in estimation theory and also employed as a measure of probability of collision by texts such as that by Prandini et al[54] can be mathematically described as

$$P_c^{lb}(k) = max_{[0,k]} \int_0^{\gamma_0} f_{\Gamma_k}(\gamma) d\gamma. \tag{4.35}$$

This is simply a matter of finding the largest probability of state space intersecting with $D_f(t)$ within a period of time, as illustrated in Figure 4-4. Various texts exist where Equation 4.35 is used[12][11][2][44][3] to approximate $P_c^{true}(k)$. This is however a dangerous course of action, since there is no guarantee on the lower bound's error, potentially compromising on $P(SA)$ instead of on $P(UA)$. Equation 4.35 is so simple however, that it only adds complexity of $\mathcal{O}(nN)$ to the calculations performed at each time step. All being said, the equation makes for a useful, empirically accurate and much-used lower bound, rather than an estimate of $P_c^{true}(k)$.

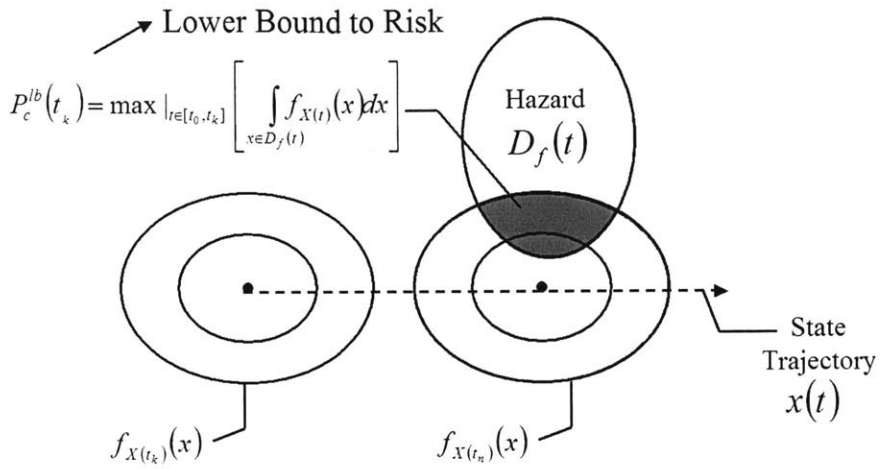

Figure 4-4: Calculating $P_c^{lb}(t)$ Over a Finite Time Horizon Interval

Figure 4-3 illustrates the relationships between the different bounded approximations and their errors and also includes $P_c^{err}(t)$ to be discussed presently.

Calculation Error

Even calculation error because of an independence assumption is a complex problem. Independence between γ and $\Delta\dot{\gamma}$ is equivalent to a combination of two assumptions:

1. $f^{B_k}_{\Delta\dot{\Gamma}_k}(\Delta\dot{\gamma}) = f_{\Delta\dot{\Gamma}_k}(\Delta\dot{\gamma})$, thereby implying that the segmentation of probability space inside and outside of $D_f(t)$ and the subsequent creation of $f^{B_k}_{\Gamma_k}(\gamma)$ has no effect on the conditional distribution of $\Delta\dot{\gamma}$.

2. Convolution may be employed so that we may express

$$f_{\Gamma_{k+1}}(\gamma)|B_{[k,k]} = f^{B_k}_{\Gamma_k}(\gamma) * f_{\Delta\dot{\Gamma}_k}(\Delta\dot{\gamma})|B_{[0,k-1]}, \tag{4.36}$$

   where Equation 4.5 is the combination of items 1 and 2.

The two above enumerations will be referred to as Assumptions 1 and 2 respectively.

Before continuing, we acquire a number of mathematical tools describing the relationship between γ and $\Delta\dot{\gamma}$ in order to characterize error.

First we need to be able to describe expected values, variance and covariance of quadratic forms. According to Jaschke [35][34], we may describe the expected value of a quadratic form $x^T Bx$ of the multi-variate normally distributed random vector x as

$$E(x^T Bx) = tr(B\Sigma_x) + \mu_x^T B\mu_x \tag{4.37}$$

where Σ_x and μ_x are the covariance matrix and mean vector of x respectively. The variance becomes

$$\sigma^2_{x^T Bx} = Var(x^T Bx) = 2tr(B\Sigma_x B\Sigma_x) + 4\mu_x^T B\Sigma_x B\mu_x \tag{4.38}$$

and the covariance between two quadratic forms $y_1 = x^T Bx$ and $y_2 = x^T Px$ can be described by

$$\rho\sigma_{x^T Bx}\sigma_{x^T Px} = Cov(x^T Bx, x^T Px) = 2tr(B\Sigma_x P\Sigma_x) + 4\mu_x^T B\Sigma_x P\mu_x. \tag{4.39}$$

From these equations we may now construct the mean vector of the joint density for $h = [y_1, y_2]^T$ as

$$E[h] = [m_{y_1}, m_{y_1}]^T \tag{4.40}$$

and the covariance matrix

$$\Sigma_h = \begin{bmatrix} \sigma^2_{y_1} & \rho\sigma_{y_1}\sigma_{y_2} \\ \rho\sigma_{y_1}\sigma_{y_2} & \sigma^2_{y_2} \end{bmatrix}, \tag{4.41}$$

where $y_1 = x^T Bx$ and $y_2 = x^T Px$.

The next step is to be able to describe the orthogonal principal axes of the symmetric matrix Σ_h. This is done through eigenvector decomposition and after some algebra the slope of the principal axis with the same sign as ρ is given by

$$m = \frac{[-\sigma^2_{y_2} + \sigma^2_{y_1} + (\sigma^4_{y_2} - 2\sigma^2_{y_1}\sigma^2_{y_2} + \sigma^4_{y_1} + 4\rho^2\sigma^2_{y_1}\sigma^2_{y_2})^{\frac{1}{2}}]}{2\rho\sigma_{y_1}\sigma_{y_2}} \tag{4.42}$$

such that the correlation line $y_1 = my_2 + c$ is created. If we let $\gamma = y_1$ and $\Delta\dot{\gamma} = y_2$ then Figure 4-5 provides additional clarity with a contour plot of the joint density function $f_{\Gamma,\Delta\dot{\Gamma}}(\gamma, \Delta\dot{\gamma})$. We will now describe the First-Order-Second-Moment (FOSM) change

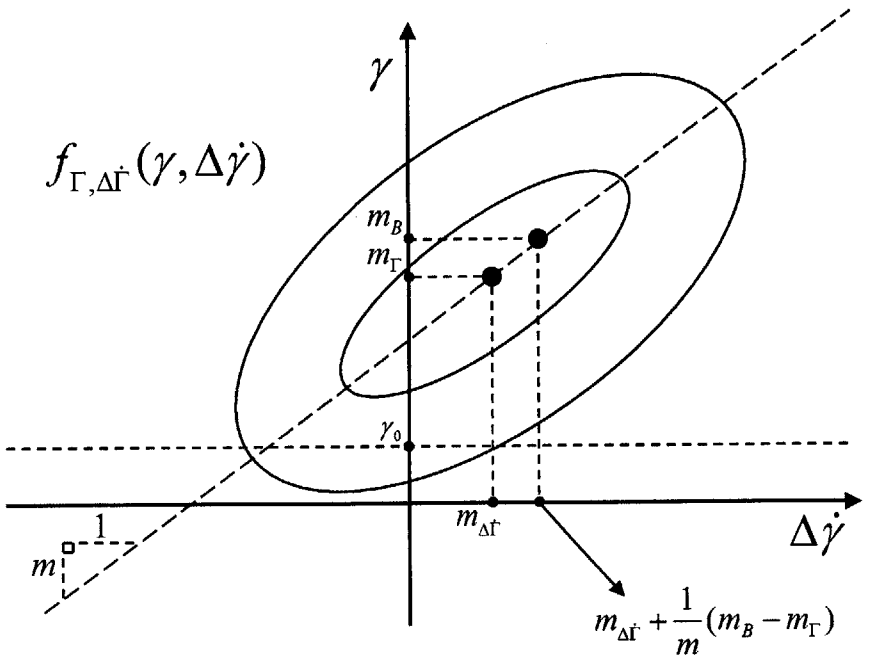

Figure 4-5: Contour Plot of $f_{\Gamma,\dot{\Gamma}}(\gamma,\dot{\gamma})$ with Correlation Line

in $f_{\Delta\dot{\Gamma}_k}(\Delta\dot{\gamma})$ because of an FOSM change in $f_{\Gamma_k}(\gamma)$. This is the characterization of error because of Assumption 1.

$f_{\Gamma_k}(\gamma)$ and $f_{\Delta\dot{\Gamma}_k}(\Delta\dot{\gamma})$ are comprised of well-behaved summed and scaled versions of non-central chi-square distributions. The maximum allowable $P_c(k)$ within $[t_0, t_h]$, denoted by $P_{c_{max}}$, is a very small number (typically $P_{c_{max}} < 0.05$). Furthermore we make the reasonable assumption that we take action along our flight path such that $P_c(k) < P_{c_{max}}$ always holds (or equivalently we don't calculate $P_c(k) > P_{c_{max}}$ as described in Section 3.3.2). Using these assumptions, we will calculate the expected influence on the mean and variance of $f^{B_k}_{\Delta\dot{\Gamma}_k}(\Delta\dot{\gamma})$ because of adjustments to $f_{\Gamma_k}(\gamma)$, such as the removal of ΔP_{c_k} probability space to form $f^{B_k}_{\Gamma_k}(\gamma)$. When we define

$$\Delta P_{c_k} = \int_0^{\gamma_0} f_{\Gamma_k}(\gamma)d\gamma, \tag{4.43}$$

then, as before, we split the PDF of γ into

$$\begin{aligned} f_{\Gamma_k}(\gamma) &= g_{A_k}(\gamma) + g_{B_k}(\gamma) \\ &= \Delta P_{c_k} f_{\Gamma_k}^{A_k}(\gamma) + (1 - \Delta P_{c_k}) f_{\Gamma_k}^{B_k}(\gamma). \end{aligned} \tag{4.44}$$

Here the mean and variance of, for example, $f_{\Gamma}^{A_k}(\gamma)$ is denoted by m_{A_k} and $\sigma^2_{\Gamma_k^A}$ respectively. Now focus on the expected value of γ after the removal of probability space within $D_f(t_k)$, i.e., given that $\gamma > \gamma_0$. This is expressed as

$$E[\gamma]|(\gamma > \gamma_0) \equiv E[\gamma_k^{B_k}] = E[\gamma_k] + \Delta E[\gamma_k], \tag{4.45}$$

where $\Delta E[\gamma_k]$ is the change in the mean of γ after the cut. Notice that $E[\gamma_k^{B_k}] \geq E[\gamma_k]$ and therefore $\Delta E[\gamma_k] \geq 0$.

Equation 4.45 can be re-written as

$$\begin{aligned} \Delta E[\gamma_k] &= E[\gamma_k^{B_k}] - E[\gamma_k] \\ &= \frac{\Delta P_{c_k}}{1 - \Delta P_{c_k}} (E[\gamma_k] - E[\gamma_k^{A_k}]) \\ &= \frac{\Delta P_{c_k}}{1 - \Delta P_{c_k}} (m_{\Gamma_k} - m_{A_k}) = \Delta m_{\Gamma_k}, \end{aligned} \tag{4.46}$$

with $E[\gamma_k^{A_k}] \equiv E[\gamma]|(0 \leq \gamma < \gamma_0)$. So if the mean of γ changes by $\Delta E[\gamma_k]$, then according to our assumptions we may employ Equation 4.42 to show that the mean of $\Delta\dot{\gamma}$ changes by $\frac{1}{m}\Delta E[\gamma_k]$. With $E[\Delta\dot{\gamma}_k^{B_k}] = E[\Delta\dot{\gamma}_k] + \Delta E[\Delta\dot{\gamma}_k]$ we may then express

$$\begin{aligned} \Delta m_{\Delta\dot{\Gamma}_k} &\equiv \Delta E[\Delta\dot{\gamma}_k] \\ &= \frac{2\rho\sigma_\Gamma\sigma_{\Delta\dot{\Gamma}}}{[-\sigma^2_{\Delta\dot{\Gamma}} + \sigma^2_{\dot{\Gamma}} + (\sigma^4_{\Delta\dot{\Gamma}} - 2\sigma^2_{\dot{\Gamma}}\sigma^2_{\Delta\dot{\Gamma}} + \sigma^4_{\dot{\Gamma}} + 4\rho^2\sigma^2_{\dot{\Gamma}}\sigma^2_{\Delta\dot{\Gamma}})^{\frac{1}{2}}]} \frac{\Delta P_{c_k}}{1 - \Delta P_{c_k}} (m_{\Gamma_k} - m_{A_k}). \end{aligned} \tag{4.47}$$

We describe the change in variance $\Delta\sigma^2_{\Gamma_k}$ in a similar way such that

$$\begin{aligned} \Delta\sigma^2_{\Gamma_k} &= \sigma^2_{\Gamma_k^{B_k}} - \sigma^2_{\Gamma_k} \\ &= \frac{1}{1 - \Delta P_{c_k}} [\sigma^2_{\Gamma_k} - 2\frac{\Delta P_{c_k}(m_{\Gamma_k})}{1 - \Delta P_{c_k}} m_{\Gamma_k} + (\frac{m - \Delta P_{c_k} m_{A_k}}{1 - \Delta P_{c_k}})^2 - m^2_{\Gamma_k}] \end{aligned}$$

$$-\frac{\Delta P_{c_k}}{1-\Delta P_{c_k}}[\sigma^2_{\Gamma_k^{A_k}} - 2\frac{m_{\Gamma_k}-m_{A_k}}{1-\Delta P_{c_k}}m_{A_k} + (\frac{m-\Delta P_{c_k}m_{A_k}}{1-\Delta P_{c_k}})^2 - m^2_{A_k}] - \sigma^2_{\Gamma_k} \tag{4.48}$$

and can express the change in the variance of $\dot{\gamma}$ as

$$\Delta\sigma^2_{\Delta\dot{\Gamma}_k} = (\frac{1}{m})^2\Delta\sigma^2_{\Gamma_k}. \tag{4.49}$$

Equations 4.47 and 4.49 are representations of the changes in mean and variance of $\Delta\dot{\gamma}$, given the first order approximation that the mapping of the first two moments from γ to $\Delta\dot{\gamma}$ may be accomplished through the correlation line with slope m. This approximation is only feasible for small values of ΔP_{c_k} associated with small time steps of calculation Δt.

Notice that Equations 4.47 and 4.49 contain no reference to m_{B_k} or $\sigma^2_{\Gamma_B}$, but are expressed in terms of m_{A_k} and $\sigma^2_{\Gamma_A}$. This is the cause of much of the complexity of these equations, but is crucial in order to determine upper bounds to errors at a later stage, since $f^{A_k}_{\Gamma_k}(\gamma)$ is strictly domain bounded within $0 < |\gamma| < \gamma_0$.

We now return to Assumption 2 and form the combined FOSM error estimates because of Assumptions 1 and 2. Recall that the Assumption 2 error is induced by determining the probability density of

$$\gamma_{k+1} = \gamma_k + \Delta\dot{\gamma}_k \tag{4.50}$$

through convolution of the conditional marginal PDF of γ_k and the unconditional marginal PDF $\Delta\dot{\gamma}_k$ instead of operating on the joint density. Elementary statistics dictate that the true mean and variance of γ_{k+1} be described by

$$m^{true}_{\Gamma_{k+1}} = m^{true}_{\Gamma_k} + m^{true}_{\Delta\dot{\Gamma}_k} \tag{4.51}$$

and

$$\sigma^{2\cdot true}_{\Gamma_{k+1}} = \sigma^{2\cdot true}_{\Gamma_k} + \sigma^{2\cdot true}_{\Delta\dot{\Gamma}_k} - 2\rho_k\sigma^{true}_{\Gamma_k}\sigma^{true}_{\Delta\dot{\Gamma}_k} \tag{4.52}$$

respectively. The convolution process maintains Equation 4.51, so the only error in the mean results from Assumption 1. Therefore, the error on $m_{\Gamma_{k+1}}$ accumulated within the

single time step $[t_k, t_{k+1}]$ can be written as

$$\Omega_{m_{\Gamma_{k+1}}} \equiv \Delta m_{\Delta\dot{\Gamma}_k}, \tag{4.53}$$

which is the direct application of Equation 4.46 such that

$$\begin{aligned} m^{ub}_{\Gamma_{k+1}} &\approx m^{FOSM}_{\Gamma_{k+1}} \\ &= m^{calculated}_{\Gamma_{k+1}} + \Omega_{m_{\Gamma_{k+1}}}. \end{aligned} \tag{4.54}$$

The induced variance error over the same time step is described in a similar way, becoming

$$\Omega_{\sigma^2_{\Gamma_{k+1}}} \equiv \Delta\sigma^2_{\Delta\dot{\Gamma}_k} - 2\rho_k\sigma_{\Gamma_k}(\sigma_{\Delta\dot{\Gamma}_k} + \Delta\sigma_{\Delta\dot{\Gamma}_k}), \tag{4.55}$$

with the first term originating from Equation 4.49. The second term, in keeping with our FOSM approximation, is provided by Equation 4.39, because of Assumption 2. In this case

$$\begin{aligned} \sigma^{2\cdot ub}_{\Gamma_{k+1}} &\approx \sigma^{2\cdot FOSM}_{\Gamma_{k+1}} \\ &= \sigma^{2\cdot calculated}_{\Gamma_{k+1}} + \Omega_{\sigma^2_{\Gamma_{k+1}}}. \end{aligned} \tag{4.56}$$

In the spirit of FOSM analysis we can now perform an FOSM matching between $f^{ub}_{\Gamma_{k+1}}(\gamma)|B_{[k,k]} \approx f^{FOSM}_{\Gamma_{k+1}}(\gamma)|B_{[k,k]}$ and $f^{calculated}_{\Gamma_{k+1}}(\gamma)|B_{[k,k]}$ by creating a new random variable through the transformation

$$r_{k+1} = \frac{\sqrt{\sigma^{2\cdot calculated}_{\Gamma_{k+1}} + \Omega_{\sigma^2_{\Gamma_{k+1}}}}}{\sigma^{calculated}_{\Gamma_{k+1}}}\gamma + \Omega_{m_{\Gamma_{k+1}}} \tag{4.57}$$

such that the FOSM approximation of $f^{ub}_{\Gamma_{k+1}}(\gamma)|B_{[k,k]}$ becomes

$$f^{FOSM}_{\Gamma_{k+1}}(r_{k+1})|B_{[k,k]} = \frac{\sigma^{calculated}_{\Gamma_{k+1}}}{\sqrt{\sigma^{2\cdot calculated}_{\Gamma_{k+1}} + \Omega_{\sigma^2_{\Gamma_{k+1}}}}} f^{calculated}_{\Gamma_{k+1}}(\gamma(r_{k+1}))|B_{[k,k]}. \tag{4.58}$$

The FOSM estimate of $P^{ub}_{c_{k+1}}$ then becomes

$$P^{FOSM}_c(k+1) = P^{FOSM}_c(k) + \int_0^{\gamma_0} f^{FOSM}_{\Gamma_k}(\gamma)|B_{[k,k]}dr[1 - P^{FOSM}_c(k)] \tag{4.59}$$

so that the FOSM estimate probability error $P^{err}_{c_{k+1}}$ then becomes

$$P_c^{err}(k+1) = P_c^{FOSM}(k+1) - P_c(k+1), \tag{4.60}$$

where $P_c(k+1)$ is also equivalent to $P_c^{calculated}(k+1)$.

Note that $P_c^{FOSM}(k)$ may always be utilized as an FOSM corrected measure of $P_c(k)$ but then higher order moments need to be determined if we are interested in characterizing the remaining error. A very simple way of applying this FOSM correction, with minimal impact on the solution algorithm for $P_c(k)$, is to adjust γ_0 at each time step to become

$$\bar{\gamma}_0 \equiv \frac{\sqrt{\sigma^{2\cdot calculated}_{\Gamma_k} + \Omega_{\sigma^2_{\Gamma_{k+1}}}}}{\sigma^{calculated}_{\Gamma_{k+1}}}\gamma_0 - \Omega_{m_{\Gamma_{k+1}}}. \tag{4.61}$$

Vehicle Applicability Tests

The preceding discussion focussed on finding $P_c^{err}(k)$, the accumulated FOSM error in the calculation of an upper bound to collision risk by time $t = t_k$. Such a real-time measure of error is very useful when evaluating confidence in the upper bound calculation of $P_c(k)$. It would however be helpful to be able to determine a measure of *a priori* applicability of the research described so far on a vehicle-obstacle problem, before implementing the bulk of the propagation algorithm described so far. We now deal with the creation of a vehicle applicability test, whereby vehicle parameters, dynamics and process noise may be employed to create a measure of the maximum expected $P_c^{err}(k)$ within $[t_0, t_h]$, i.e. $max_{\forall h<\infty}(P^{err}_{c_h})$. It is then up to the developer to decide whether this measure of $P_c^{err}(k)$ is sufficiently small when compared to $P_{c_{max}}$ and the alerting performance required for the application.

It should be made clear that we will not attempt to find the true $max(P^{err}_{c_h})$ without the calculation of $P_c^{err}(k)$ described earlier in this chapter. In the interest of simplicity and the creation of a basic heuristic test, we will quantify an upper bound to the dominant portion of $P_c^{err}(k)$. Eventually, this approach boils down to calculating the number of standard deviations between $E[\gamma]$ and the γ_0 boundary, calculating the maximum change in the number of standard deviations and then translating this into an amount of change in probability.

Inspection of Equations 4.47 to 4.49 and comparison with Equations 4.53 and 4.55 indicate that the so-called Assumption 2 in the previous Subsection is the dominant contributor towards $P_c^{err}(k)$. The dominance is asserted by the relatively small value of $\Delta P_c(k) \leq P_{c_{max}}$ and the amount of error is largely moderated by the covariance between γ and $\dot{\gamma}$, given by Equation 4.39. Assumption 2 only contributes to the second term of $\Omega_{\sigma^2_{\Gamma_{k+1}}}$ (Equation 4.55) so this is where we will focus our attention. As such, and only for this test, we assume that

$$\Omega_{m_{\Gamma_{k+1}}} \approx 0 \tag{4.62}$$

and

$$\Omega_{\sigma^2_{\Gamma_{k+1}}} \approx -2\rho_k \sigma_{\Gamma_k} \sigma_{\Delta\dot{\Gamma}_k}. \tag{4.63}$$

From Equation 4.2, the quadratic relationship between x and γ and also x and $\dot{\gamma}$, we know that σ_{Γ_k} and $\sigma_{\Delta\dot{\Gamma}_k}$ are functions of the vehicle's closed loop dynamics and process noise. We will refer to the combination of these three aspects as the *system model.* For the time being it is best to fix the system model to a representative form, to promote clarity during the course of this explanation. For any one such system model, ρ is always maximized at a point where the vehicle's velocity vector is aimed directly at the state origin, since γ and $\dot{\gamma}$ are then perfectly aligned, i.e.

$$grad(\gamma) = C \cdot grad(\dot{\gamma}), \tag{4.64}$$

where C is scalar and real-valued.

Next we perform a straight-forward state-propagation simulation (propagating closed loop dynamics with a reference path and process noise in Simulink). The simulation is performed in the vicinity of the state origin, where the vehicle is made to fly directly at the origin at a representative speed. This is the encounter where ρ is maximized. During the simulation we also compute $\sigma^2_{\Gamma_k}$, $\sigma^2_{\Delta\dot{\Gamma}_k}$ and $\rho_k \sigma_{\Gamma_k} \sigma_{\Delta\dot{\Gamma}_k}$ using Equations 4.38 and 4.39. Now express $\Omega_{\sigma^2_{\Gamma_k}}$ as a ratio of $\sigma^2_{\Gamma_k}$ such that

$$R_{\sigma^2_k} = \frac{\Omega_{\sigma^2_{\Gamma_k}}}{\sigma^2_{\Gamma_k}}$$

$$= -2\rho\frac{\sigma_{\Delta\dot{\Gamma}_k}}{\sigma_{\Gamma_k}}$$

$$\Rightarrow R_{\sigma_k} = \sqrt{2|\rho|\frac{\sigma_{\Delta\dot{\Gamma}_k}}{\sigma_{\Gamma_k}}}, \tag{4.65}$$

where R_σ is the ratio of the standard deviation of the calculation error of γ with respect to the (unconditional) standard deviation of γ.

Furthermore, as a worst-case (maximum calculation error), find the maximum value of R_{σ_k} and associated $k = k_m$ where $\gamma_{k_m} > \gamma_0$, and assume that all of $P_c(k_m) = P_{c_{max}}$ is accumulated at $k = k_m$, meaning that $P_c(k_m - 1) = 0$ and $P_c(K) = P_c(k_m)$. R_{σ_k} will peak as the vehicle approaches the origin and will then change sign as the vehicle passes through the origin. The shape of R_{σ_k} is dominated by the behavior of ρ_k and we denote the maximum value of R_{σ_k} as $R_{\sigma_{MAX}}$. See Figure 4-6 for a typical illustration of the progression of R_{σ_k} vs. γ.

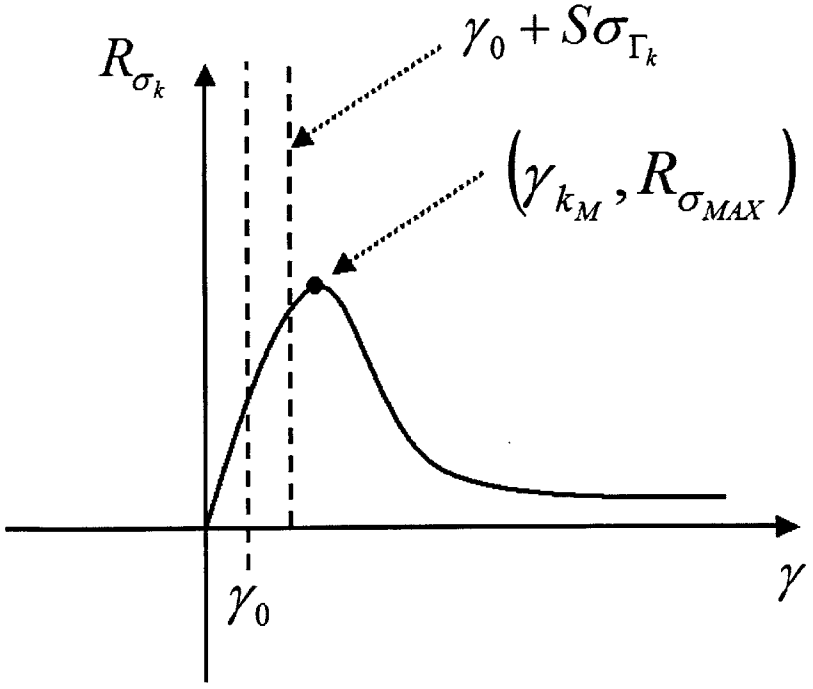

Figure 4-6: R_{σ_k} vs. γ for the Simulation of a Vehicle Flying Towards the Origin

It is $R_{\sigma_{MAX}}$ that allows us to express $max(\frac{P_c^{err}(k)}{P_{c_{max}}})$. Let's look at an example: Say that $P_{c_{max}} = 0.06$, then $P_c(k_m) = P_{c_{max}}$ would be true when $E[\gamma]$ is S standard deviations away from γ_0, or mathematically

$$E[\gamma_k] = \gamma_0 + S\sigma_{\Gamma_k}, \tag{4.66}$$

where S is a positive real number that is dependent on the system model and derived from $f_{\Gamma_{k_m}}(\gamma)$. For example, for the New Generation Mini II (NGM-II) UAV, described in Section 5.1, $S \approx 3$ when $P_{c_{max}} = 0.02$ and a bound on the standard deviation can be expressed as

$$(3 - R_{\sigma_k})\sigma_{\Gamma_k} < \sigma_{\Gamma_{km}}^{true} < (3 + R_{\sigma_k})\sigma_{\Gamma_k}. \tag{4.67}$$

Since $R_{\sigma_k} = 0.15$ for the NGM-II we can use $f_{\Gamma_{k_m}}(\gamma)$ to calculate that

$$max_{\forall t<\infty}|(P_{c_h}^{err})| < 0.013 \tag{4.68}$$

or $max(\frac{P_c^{err}(k)}{P_{c_{max}}}) \approx 0.22$.

We may now iterate across various vehicle models in order to determine a worst-case scenario. Also notice that the outcome of Equation 4.66 and the value γ_{k_m} in Figure 4-6 are seldom equal, but the worst case error results, as described in this section, when they are equal.

At this point, it is worth pointing out that the vehicle applicability test is based on a concept often encountered in reliability theory. The ratio $R_{\sigma_{MAX}}$ is the induced deviation of the so-called maximum safe *Mahalanobis Distance* [5]. The deviation is caused purely by calculation error.

4.2.2 Numerical Approximation

This subsection focusses on quantifying the induced calculation error when determining $f(x)$ through the FFT method described in Equations 4.19 to 4.25. Numerical approximation of errors can be divided into three main contributing factors, namely round-off, sampling and truncation error. Round-off error is usually negligible as long as sufficient numerical precision is used to represent discrete values. Sampling error is a function of discretization

step size and is induced when sampled signals are integrated (or differentiated). Truncation error is a function of the FFT method and is caused by the need to truncate the bounds of the indefinite integral representing the FI integral. We will treat numerical error as a sum of the latter two types of errors. (Aliasing error is negligible as long as we hold to the frequency and sampling constraints described in Section 4.1.3).

Sampling and truncation errors are both functions of the interval between samples and the number of samples taken. Therefore, as a practical matter, it is useful and important to be able to express the sampling interval and number of samples required for a given error in terms of the characteristic function. These errors are usually described in terms of a probability density function [32][17], but such a function is not available to us before starting our calculation. Other approaches, like that of Bohman[9], are to experiment with numerical parameters in order to obtain good convergence, clearly not a feasible approach when calculation time and complexity and calculable risk error bounds are of importance. It is however noteworthy that numerical approximation error may be reduced to an arbitrary value at the cost of computational complexity.

Hughett [32] proved that the minimax ($\epsilon = \epsilon_T + \epsilon_N = L_\infty > 0$) error of a two-sided random variable x can be related to the minimum sampling period T and minimum number of samples N of the density function $f_X(x)$ by bounding $f_X(x)$ and $\phi_X(x)$ with exponential functions. In this way we find that

$$T \geq 2\left(\frac{2A(2\alpha-1)}{\epsilon_T(\alpha-1)}\right)^{1/\alpha} \tag{4.69}$$

and

$$N \geq 2 + 2\left(\frac{2BT^{\beta-1}}{\epsilon_N\sqrt{\beta-\frac{1}{2}}}\right)^{1/(\beta-\frac{1}{2})} \tag{4.70}$$

if

$$|f_X(x)| \leq A|x|^{-\alpha} \quad \forall |x| \geq T/2 \tag{4.71}$$

with $\alpha > 1$ and

$$|F_X(u)| \leq B|u|^{-\beta} \quad \forall |u| \geq N/2T \tag{4.72}$$

and if $\beta > \frac{1}{2}$. For a given maximum allowable error ϵ, Equations 4.69 and 4.70 may be

iterated for different values of $\epsilon = \epsilon_T + \epsilon_N$ to find the minimum values of T and N. It is also useful to note that if $|f_X(x)| = 0\ \forall |x| \geq T/2$ and $|F_X(u)| = 0\ \forall |u| \geq N/2T$, then $\epsilon_T = 0$. Hughett specifically derived Equations 4.69 to 4.72 since Equations 4.71 and 4.72 hold for x expressed as a quadratic form of standard normal random variables. These bounds are therefore directly applicable to our problem.

There is a subtle point to be observed when computing N and T for a given ϵ: Unless x is strongly skewed, it pays to perform the calculation of the IFFT to result in $f_X(x - x_0)$, where $f_X(x - x_0)$ has approximately zero mean or median. This shift decreases the required value of A to make Equation 4.71 hold. The shift is done by exploiting a property of the FT, which is to multiply $F(u)$ by $e^{j2\pi u x_0}$, where x_0 is the amount of shift required to center $f_X(x)$. After the calculation, $f_X(x)$ can be regained from $f_X(x - x_0)$.

A difficulty does however arise in that A and α need to be determined from the shape of the density function $f_X(x)$, not the characteristic function $F(u)$, as promised. Recall that we are attempting to calculate $f_X(x)$, so it is not yet available to substitute into Equation 4.71. Hughett also shows that if we fix $\alpha = 2\eta$, with η being any non-negative integer, then we may take

$$A = (-1)^n \phi^{2\eta}(0), \tag{4.73}$$

where $\phi^{2\eta}$ is the 2η 'th derivative of the characteristic function, and Equation 4.71 will always hold, as long as $\phi^{2\eta}(0)$ actually exists. It pays to be able to increase α, since it lowers the approximation error bound, as is clear from Equation 4.69.

4.2.3 Quadratic Approximation of Collision Bounds

A collision bound may be as simple as a spherical shell surrounding an aircraft, but may also be any general quadratic shape described by Equation 4.2. The contents of $D_f(t)$ inside the collision bound is usually viewed as an unsafe domain around a vehicle. This implies that the boundary is not "skin-tight", i.e. $D_f(t)$ includes space that is not taken up by the vehicle itself (and possibly even *vice versa*). Such a representation is inevitable when describing γ_0 in a quadratic fashion, since no relevant vehicles are truly ellipsoids (airships probably come closest). Collision avoidance systems like TCAS are based on the

premise that vehicles are not to be allowed within a minimum distance of one-another, the DMOD[67][56]. For such systems, prediction of future transgression of this minimum miss distance constitutes an unacceptably high risk of collision and warrants resolution action. The quadratic collision metric presented in this thesis may be used to provide just such a prediction of transgressing a minimum miss distance $\sqrt{\gamma_0}$. $\sqrt{\gamma_0}$ is however only a true DMOD distance when Equation 4.2 is a purely spherical form. In this way, both TCAS and the research documented in this dissertation really define collision risk as the risk of entering domain of failure bubble around a vehicle.

It is worth pointing out that truer collision risk may be obtained (even though this is a higher risk venture) by shaping the quadratic bound such that it more closely approximates the shape of a vehicle. One such optimization may be suggested as

$$B^*(t) = argmin_{B(t)}[\int_{\forall x \in (D_v(t) \cap D_f(t))^c} dx \text{ such that } D_v(t) \subseteq D_f(t)], \tag{4.74}$$

where $D_v(t)$ describes the volume contained within the vehicle and $D_f(t)$ describes the usual $\gamma \leq \gamma_0$. The time dependence is usually dropped, but it may sometimes be useful to account for vehicles changing shape, such as the PCUAV system after docking [63] or vehicles with variable wing sweep. This is left to the reader to implement if it is deemed necessary for a specific problem.

4.2.4 Time Discretization Error

The numerical solution approach to finding $P_c(t)$ described in this thesis requires time discretization in order to propagate vehicle state, state PDFs and the resulting accumulation of $P_c(t)$ within a finite time horizon window $t = [t_0, t_h]$. In discrete time the window is expressed at $t = [t_0, t_{K-1}]$, such that K discrete time samples exist between t_0 and $t_{K-1} = t_h$. The time step is defined as $\Delta t = t_{k+1} - t_k$ and for this discussion is assumed to be constant $\forall\ k = [0, K-1]$, where k is an integer.

When $\Delta t \rightarrow 0$, the discretization most accurately represents the continuous time realm, but at the cost of immense computational complexity. As with most discrete problems, we need to design Δt in order to guarantee a measure of discrete representational accuracy. In

the next few paragraphs we will discuss two aspects that drive the size of Δt.

State and Collision Metric Propagation Accuracy

The solution algorithm to $P_c(t)$ proposed in Figure 4-2 requires propagation of $f_{\Gamma_k}^{B_k}(\gamma)$ in order to find $f_{\Gamma_{k+1}}(\gamma)|B_{[k,k]}$. The propagation process employs convolution, an approximate solution to the problem, to which the approximation error is discussed in Section 4.2.1. During the propagation procedure however, we propagate the collision metric using

$$\begin{aligned} \gamma_{k+1} &= \gamma_k + \Delta\dot{\gamma}_k \\ &= \gamma_k + \Delta t.\dot{\gamma}_k. \end{aligned} \tag{4.75}$$

This is a first order, or Euler, integration approach described in [53] and [72]. The dominant second order error induced at each time step resulting from application of Euler integration can be found from *Taylor* series expansion[22] to be

$$\varepsilon_k^\gamma = \frac{1}{2}(\Delta t)^2\ddot{\gamma}_k. \tag{4.76}$$

For the purposes of this discussion, we assume that the error in Equation 4.76 is representative of the total integration error. In addition, we assume that the vehicle in question is commanded to fly a straight-line nominal trajectory at constant velocity V, at a unit circle domain of failure and that the state covariance matrix $\Sigma_X(t)$ is steady. In this case $\gamma = x^2$, therefore $\ddot{\gamma}_k = 2x\ddot{x} + 2\dot{x}^2$. As a result, the integration error becomes

$$\varepsilon_k^\gamma = (\Delta t)^2 V^2. \tag{4.77}$$

The integration error is an absolute figure that needs to be compared with a relevant measure of accuracy. As in previous sections, we employ the standard deviation σ_{Γ_k} as a reference. In short, we need to ask how large ε_k^γ may be when compared to σ_{Γ_k}, i.e., find

$$R_{max}^\gamma \equiv max_{[0,K-1]}\frac{\varepsilon_k^\gamma}{\sigma_{\Gamma_k}}. \tag{4.78}$$

R_{max}^γ occurs when σ_{Γ_k} is minimized, since ε_k^γ is constant. For every individual application, we need to find an acceptable R_{max}^γ, a process described in detail in Section 4.2.1's discus-

sion on vehicle applicability.

For the NGM-II vehicle example application in Chapter 5, when $\Delta t = 0.002$, then $R^{\gamma}_{max} = 0.002$. The maximum resulting error in the calculation of $P_c(t_h)$ because of discrete time collision metric propagation therefore becomes about 0.8 percent. This is considered to be negligible when compared to $P_c^{err}(t_h)$, especially since the propagation error is an upper bound.

Finally, recall that γ and $\dot{\gamma}$ are calculated from propagation of the vehicle state x within the finite horizon simulation time window. For the constant velocity case described above, the state error is negligible for $\Delta t = 0.002$, in fact, it can be shown that there is always an order of difference between the error incurred when propagating γ and x respectively, with $P_c^{err}(t_h)$ being dominant. Therefore, if a given value of Δt provides a satisfactory value of $P_c^{err}(t_h)$, then there is no need to be concerned with state propagation error.

Risk Probability Accumulation Accuracy

The accumulation of P_c as a function of time is in itself an integration process. At every discrete discretization time we calculate the added collision risk $\Delta P_c(k) \equiv \int_0^{\gamma_0} g_{A_k}(\gamma) dx[1 - P_c(k-1)]$ and $P_c(k) = P_c(k-1) + \Delta P_c(k)$ (Refer to Section 4.1.3). As is the practice with such Euler integration processes, this integration error is approximated by the second order term

$$\begin{aligned} P_c^{acc-err}(t) &\equiv \frac{1}{2}(\Delta t)^2 \ddot{P}(t) \\ \rightarrow P_c^{acc-err}(k) &= \frac{1}{2}[\Delta P_c(k) - \Delta P_c(k-1)]. \end{aligned} \tag{4.79}$$

The value of $P_c^{acc-err}(k)$ relative to $P_c(k)$, needs to be sufficiently small in order to ensure accurate discrete risk accumulation, or

$$max_{[0,K-1]} \frac{P_c^{acc-err}(k)}{P_c(k)} < \xi, \tag{4.80}$$

where $\xi \ll 1$. The maximum relative error in Equation 4.80 accumulates when the time rate of change of $\Delta P_c(k)$ is maximized. This worst-case scenario again occurs when a vehicle is

on a straight-line heading towards the center of D_f, at the maximum possible speed relative to D_f, and when $P_c(k)$ is maximized. Since $P_c(k) \leq P_{c_{max}}$, we should choose simulation time step size such that

$$[\Delta P_c(k) - \Delta P_c(k-1)] \leq \frac{2P_{c_{max}}}{\xi}. \tag{4.81}$$

This process requires selecting a value of Δt, using this Δt to complete a simulation of a vehicle approaching perpendicular collision with D_f, and iterating until Equation 4.81 holds true. Note that the simulation does at least need to start with a small enough Δt to allow a number of non-zero values of $\Delta P_c(k)$ to be calculated.

For the NGM-II for example, if we require $\xi = 0.01$ and $P_{c_{max}} = 0.05$, then $\Delta t = 0.001$ results. Notice that for this example the value of Δt is very close to that calculated for metric propagation accuracy. The smaller of the two values should be chosen.

The values of Δt suggested here are lower bounds in order to strictly ensure required representational accuracy for all possible conflict scenarios. It is worth choosing Δt according to the measures described, but Δt should always be perturbed through the course of a number of simulations in order to determine whether larger discretization steps yield sufficiently accurate results. In this way we might considerably reduce computational effort.

It should also be realized that the head-on collision scenarios often used to characterize worst-case discrete time performance seldom require the same amount of accuracy when calculating $P_c(t)$. In most head-on collision examples $P_c(t)$ accumulates so quickly, and collision is so clearly apparent, that accuracy under this special condition may be sacrificed. The result would be an increase in Δt and therefore reduced computational complexity during most other conflict scenarios, without any visible loss of avoidance system performance.

4.3 Time and Resolution Issues

4.3.1 Finite Horizons

This research focusses on solving for collision risk within a finite horizon interval. Simulations of this nature within infinite horizons are folly because of the obvious infinite cal-

culation resources required. Such calculations might still be intractable for very large, yet finite horizons. The intent is to create a method of solution that would be tractable within a required simulation interval, and we therefore need to characterize both the tractability or complexity of a solution and the size of the required time horizon, $t_w = t_h - t_0$. A required time horizon could be characterized in terms of a minimum, maximum or an optimal duration.

Let us denote the upper and lower bounds on t_w as t_w^{ub} and t_w^{lb} respectively. The lower bound is driven upwards only by the minimum amount of useful forward simulation time, for example, the shortest amount of warning time required to avoid a hazard. t_w^{ub} is however clamped down by both the maximum computationally tractable simulation time window, denoted by t_c, and knowledge of vehicle intent.

Intent is possibly the most interesting of the influences on t_w. It has been pointed out (See Chapter 1) that collision avoidance systems for piloted vehicles suffer from an inability to model pilot intent, and that this intent causes divergence in the extrapolation of state estimation/prediction into the future. (See Figure 1-1). The intent of vehicles under autopilot control may be characterized more exactly, since such systems are usually guided by known control laws and deterministic decision making processes. Even though this is true, automated vehicles are still subject to events for which future intent is uncertain and highly complex. Some of the processes might include unexpected human intervention, required rescheduling of flight paths because of mission changes, complex interactions between numerous automated vehicles and/or system failure. These events make it impossible to predict even the behavior of automated vehicles for an indeterminate period of time, thereby capping t_w. We may however divide such events into two groups, those for which the onset may be delayed by a known time t_d and those that cannot be delayed. t_d may, for example, be the amount of time elapsing between input of a new navigation waypoint and the vehicle acting on the waypoint change. The designer of a UAV may ideally be able to control this time delay. As long as $t_w < t_d$ we are able to make use of finite horizon methods for collision prediction, as stated in this thesis, without loss of generality, since no event of this nature will occur within a given finite horizon simulation window. Additionally, the event is both controlled and expected. Chief amongst the events that we will not attempt to compensate

for are immediate human intervention and unexpected systems failures.

It is up to the designer of a system to determine whether t_w is upper bounded by t_d or by t_c, given the specifics of the system under development. The general tractability of the algorithms developed here allows us to pose this problem as one of expense vs. requirements, where expense drives processing power and requirements drive the nature of a vehicle's guidance and control solution. Figure 4-7 illustrates both the bounds of t_w and the factors driving the bounds. On this figure, "Benefit" describes how useful a simulation time window is, while "Complexity" describes the degree of computational complexity required to solve for collision risk.

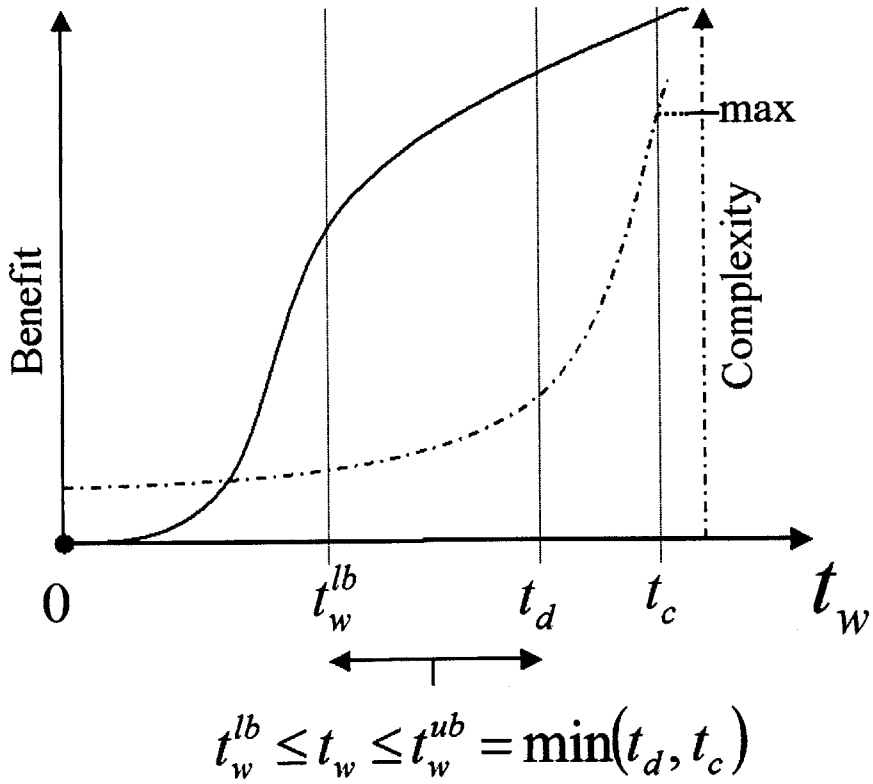

Figure 4-7: Factors Influencing the Size of t_w

Furthermore, it is worth pointing out that the size of t_w has little effect on $P_c^{err}(k)$, the

accumulated FOSM collision risk calculation error, discussed in Section 4.2. The calculation errors are driven by the size of $P_{c_{max}}$, the maximum allowable collision risk for a vehicle, and since we assume that action is always taken to ensure that $P_c(k) < P_{c_{max}}$ within any t_w, the error is both calculable and bounded within such an interval.

Interestingly enough, the upper bound of t_w, given that it is either constrained by t_d or processing power, is also in a sense the optimum value of t_w. The preceding discussion shows that we always benefit by extending t_w, as long as we can afford the expense and ensure that $t_w < t_d$. A larger value of t_w does provide more lead-time to collision event prediction and more closely approximates an infinite horizon problem. The benefit does however approach an asymptotic constant as t_w increases.

The only remaining question is whether the upper bound of t_w is useful, i.e., that the time interval is long enough to provide adequate warning of a collision risk event. This is an issue of adequate control authority and the ability to investigate risk along alternate trajectories capable of avoiding typical collision scenarios. The lower bound of t_w, denoted by t_w^{lb}, is captured from an avoidance scenario that requires immediate evasive action (at $t = t_0$) where the vehicle avoids collision risk of $P_c(t_h) = P_c(K-1) > P_{c_{max}}$. This is again dependent on the specific system under consideration. To clarify, a two-dimensional example is provided:

Let $R_{sm}(t)$ be the semi-major axis of an ellipsoid $D_f(t)$, and $R_{tr}(t)$ be the minimum turning radius of a vehicle, at maximum ground speed V_{max}, where this turning radius is employed as an avoidance trajectory (see Section 3.3). From simple geometry in Figure 4-8 we can show that

$$t_w^{lb} \leq M \frac{\sqrt{(R_{sm}(t) + R_{tr}(t))^2 - R_{tr}(t)^2}}{V_{max}}, \tag{4.82}$$

where M is a real scalar multiplier with $M \geq 1$. M simply provides a safety margin, so that $P_c(h) \leq P_{c_{max}}$.

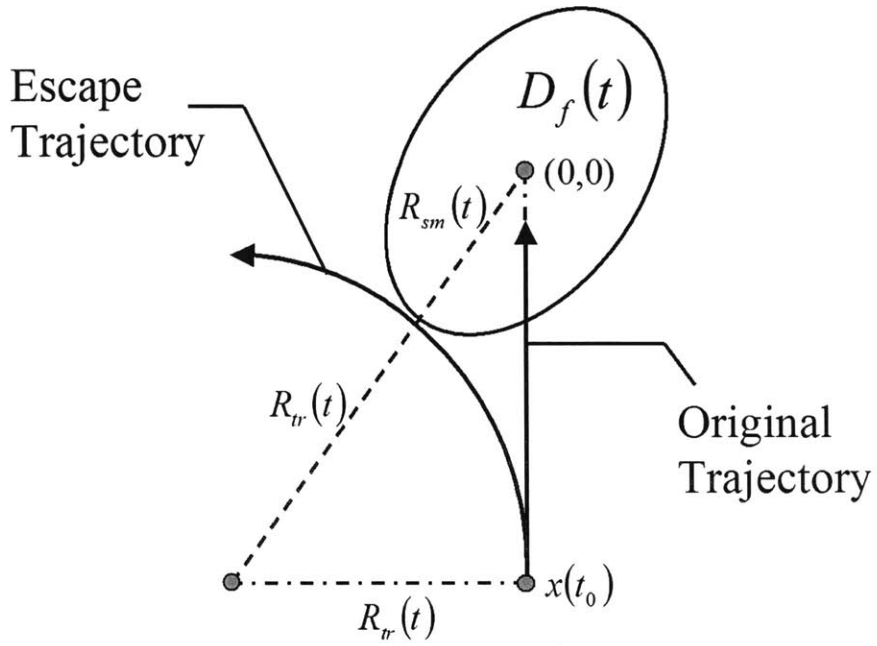

Figure 4-8: 2D Example for Finding the Minimum Safe Propagation Time Window

4.3.2 Avoidance Trajectory Discretization Step Size

In Section 3.3 we describe how an infinite set of avoidance trajectories would need to be created in order to form a continuous CSOC curve. We then discretized the time interval at which avoidance trajectories are investigated, so as to create a finite number of compound avoidance trajectories to examine. The relevant trajectory discretization step size was designated t_{AD}. In this section we investigate the size of t_{AD} and its influence on the avoidance system. We do not attempt to create a general solution to the size of t_{AD}, but rather discuss the issues that should be considered during implementation on a specific platform.

The consideration is one of avoidance resolution vs. computational complexity. A high "avoidance resolution" implies that we may investigate the delayed implementation of avoidance trajectories along very small increments of time. The maximum resolution is reached when $t_{AD} = \Delta t$, the simulation time step size and the minimum resolution results when $t_{AD} = t_w$, the time horizon window size. The latter is equivalent to the traditional method

of solving for an SOC at $t = t_0$, and results in only one SOC, which is automatically the trivial CSOC. On the other hand, having $t_{AD} = \Delta t$ would result in unnecessary computational complexity. This smallest possible t_{AD} may result in the ability to exploit the smallest possible window of avoidance opportunity, and also allows us to wait until the last possible opportunity before taking avoidance action, since the peak of $P(SA)$ is more precisely defined. In this way we reduce $P(FA)$.

There is however little gain in decreasing t_{AD} past the point where the resolution of the resulting value of $P(SA)$ is smaller than our ability to accurately calculate $P(SA)$. Also, intuitively there is very little gain in waiting until the last possible moment to apply an avoidance maneuver when a collision is inevitable along the existing nominal trajectory N. We will focus on the former as being in some sense the optimal value of t_{AD}.

We recall that $P(SA) = 1 - P_c(K-1)$, where $P_c(K-1)$ is the probability of collision accumulated within the duration of t_w for a specific avoidance maneuver. We also recall that the vehicle applicability test in Section 4.2.1 developed a heuristic measure of the maximum calculation error incurred when finding $P_c(K-1)$. A developer may therefore keep reducing t_{AD} until

$$P(SA)|_{t_{AD}+\Delta t_{AD}} + max(P_c^{err}(K-1)) < P(SA)|_{t_{AD}} \tag{4.83}$$

for the CSOC across a large number of representative simulation runs.

Experience with simulation examples and general implementation of the collision avoidance strategy described in this thesis indicate that it is seldom worth reducing t_{AD} to less than

$$t_{AD} \sim \frac{1}{\kappa}\lambda_{AD}, \tag{4.84}$$

where λ_{AD} is the vehicle's largest closed-loop position tracking time constant when commanded to follow a specific avoidance trajectory and κ is a positive real number such that usually $2 < \kappa < 5$, depending on the damping ration of λ_{AD}. For the NGM-II $\kappa \approx 3$ with a damping ratio $\zeta \approx 0.6$. A decrease in ζ would result in an increase in κ. Equation 4.84 provides a good initial guess for t_{AD}, which may then be optimized through iteration of

Equation 4.83, if desired.

4.3.3 Special forms

A number of special mathematical conditions exist which trivialize the calculation of the PDF of a quadratic form of normally distributed random variables.

Special relationships between B and Σ_X

$f_{\Gamma_k}(\gamma)$ will always be a chi-square distribution, more generally a non-centric chi-square distribution, when any one of two equivalent sets of conditions hold [31]:

1. $tr(B\Sigma_X B\Sigma_X) = tr(B\Sigma_X)$, and $m_X^T B\Sigma_X B m_X = m_X^T B m_X$
2. $B\Sigma_X$ is idempotent, i.e. $B\Sigma_X = B\Sigma_X B\Sigma_X$

where Σ_X and m_X are the covariance matrix and mean of x respectively. In such a case, the non-centrality parameter of the distribution is $\lambda = 0.5 m_X^T B\Sigma_X B m_X$, degrees of freedom $f = tr(B\Sigma_X)$, mean $m_\Gamma = f + 2\lambda$ and variance $\sigma_\Gamma^2 = 2f + 8\lambda$.

This special case allows us to create $f_{\Gamma_k}(\gamma)$ numerically from pre-compiled distribution tables when the only non-zero entries in B are unit entries along its diagonal. At the same time Σ_X would need to be an identity matrix. The advantages of this special property is two-fold: Firstly we may somewhat reduce real-time computational complexity by transforming B and Σ_X into the above-mentioned desired format, off-line; and Secondly this format is useful for verifying and testing the implementation of the FI algorithm.

Special relationships between random variables

Another special case arises when all state variables are Independent Identically Distributed (IID) normal random variables. The work of Fisher [23] and Cochran [14] show that when sampling normal universes, the distribution and independence of quadratic forms are related to their degrees of freedom. This result is usually referred to as *Cochran's Theorem.*

In more mathematical terms [39], if

$$x = \begin{bmatrix} x_1 & \dots & x_n \end{bmatrix}^T \tag{4.85}$$

is a random sample from a random universe, with mean and variance denoted by μ and σ^2 respectively and $\gamma = x^T Bx$. Denote a noncentral chi-square distribution with f degrees of freedom and non-centrality parameter δ^2 by $\chi_f(\delta_i^2)$. Let $\lambda_(1) > \lambda_2 > ... > \lambda_l$ be all the distinct eigenvalues of B with multiplicities $r_1, r_2, ..., r_l$ respectively such that $\sum_{i=1}^{l} r_i = n$, then

$$f_{\Gamma_k}(\gamma) = \sum_{i=1}^{l} \lambda_i \chi_{r_i}^2(\Delta_i^2), \tag{4.86}$$

where

$$\Delta_i^2 = \mu^2(1_n^T E_i 1_n)/\sigma^2 \tag{4.87}$$

and

$$E_i = \prod_{j \neq i} \frac{1}{(\lambda_i - \lambda_j)}(I - \lambda_j B),\ i = 1, ..., l. \tag{4.88}$$

In this case 1_n denotes a nx1 column vector of 1's.

This very useful form may be exploited if the state vector can be transformed into an IID format. Such a transformation is not possible in general, but it is well worth exploring this property for a specific application. Real-time computational savings would result at the cost of calculating $\chi_f(\delta_i^2)$ tables off-line.

4.4 Chapter Summary

This chapter is developed from the following major assumptions:
$P[x(t) \in D_f(t),$ exactly once in $[t_0, t_k]] \approx P[x(t) \in D_f(t),$ at least once in $[t_0, t_k]]$; the boundary of a domain of failure may be expressed or approximated as a quadratic function of state space; the quadratic collision metric, γ, and $\dot{\gamma}$ are assumed to be independent over one discrete time step when calculating the amount of collision probability flow into $D_f(t)$ (the error resulting from this assumption is characterized and can be corrected for); and, when the collision avoidance system performs properly, there always exists a flight path along which the probability of collision is less than some pre-defined maximum $P_{c_{max}}$, such that $P_{c_{max}} \ll 1$.

An intuitive collision metric is introduced that is defined as a positive definite quadratic measure of state space. In three-dimensional position space the metric manifests itself as a

spheroid domain of failure or hazard space surrounding vehicles and hazards. The quadratic form is shown to act as a order reductive metric, compressing n-dimensional state space into a one dimensional metric.

It is shown how the quadratic collision metric lends itself to solving the collision risk algorithm illustrated in Figure 3-3 through simple and proven approximations, resulting in the tractable algorithm described in Figure 4-2. The intractable $\mathcal{O}(N^{2n})$ numerical complexity required by the solution in Figure 3-3 at each time step in a finite horizon window of solution is thereby replaced by significantly reduced $\mathcal{O}(N^2)$ complexity.

We also illustrate how research lead by Jaschke[35] and Hughett[32] in the field of econometrics may be employed to efficiently solve for the quadratic PDFs required by the risk solution algorithm in Figure 4-2. The calculations are shown to be solvable through Fourier inversion of known probabilistic characteristic functions, and we show how to employ Inverse Fast Fourier Transforms (IFFT) in order to find the PDFs according to predefined bounds of representation accuracy (error). The bounds are linked to the characteristic function, which is known exactly before inversion takes place. In this way, we may guarantee and control calculation error bounds even before attempting IFFTs. The efficient IFFT solutions of guaranteed accuracy allow further reduction of computational complexity at each time step in the finite horizon to $\mathcal{O}(Nlog_2(N))$ for the algorithm in Figure 4-2.

A number of properties of the time derivative of the collision metric are derived. The risk solution algorithm in Figure 4-2, for example, requires that the PDF of the time derivative of the quadratic collision metric be calculable. It is therefore shown that this derivative is itself a quadratic form for most dynamic models of second order or more. It is also shown that we may always convert the representation of the rate of change into a quadratic form of second order or more. Jaschke and Hughett's methods may thus also be used to solve for the required PDF of $\dot{\gamma}$.

The calculation errors induced by the assumptions made to convert the intractable algorithm in Figure 3-3 to the tractable version displayed in Figure 4-2 are identified. Each type of error is either quantified exactly, upper bounded or approximated through FOSM

methods, depending on the nature and severity of the error. These calculations may be performed during real-time simulation, in order to provide error estimates on each estimation of collision risk.

A vehicle applicability test was devised which can be run off-line, before implementing the bulk of the solution strategy. The test gauges the worst-case error of the solution strategy based on the bulk of the calculation error.

Practical aspects such as simulation step size, the time resolution of investigating avoidance trajectories and the horizon time length are investigated. The major contributing factors influencing these aspects are identified and a number of metrics are suggested in order to constrain and optimize these quantities.

Chapter 5

Application Examples

This chapter focusses on application and analysis of the algorithm presented in Section 4.1, for the determination of collision risk along a given vehicle trajectory. Yang and Kuchar[41][42][70][69] have shown how to employ instantaneous values and thresholds of $P(SA)$ and $P(UA)$ in an alerting system. We will not attempt to redo these algorithms and applications provided in their publications, but will rather focus on finding $P_c(t)$ and analyzing its calculation error, as described throughout Chapter 4. The reader is reminded of the relationships between $P_c(t)$ and $P(SA)$ and $P(UA)$ described by Equations 2.2 and 2.3.

Two simulated application examples are provided in the next two sections, with each further sub-divided into two conflict scenarios. The first example is that of the PCUAV NGM-II vehicle needing to avoid hazards such as tree trunks, while the second example focusses on the calculation of collision risk between two large commercial transports. Both applications are implemented in two dimensions, for the sake of simplicity and in order to be able to distill meaningful conclusions from the simulation results. Results are analyzed relative to the true probability of collision, $P_c^{true}(t_h)$, at the end of each simulation horizon, and percentage changes are described as $\frac{Amount \cdot of \cdot Change}{P_c^{true}(t_h)} \times 100$.

In each simulation example, results are plotted only for part of the true finite horizon time window t_w. This is done in an effort to illustrate the phase of the simulation where risk is accumulated. No significant change in $P_c(t)$ is visible outside of $t = [t_1, t_2]$ in Figures 5-2, 5-3, 5-5 and 5-6, e.g., $P_c(t_h) = P_c(t_2)$.

5.1 PCUAV NGM-II and a Stationary Hazard

5.1.1 Vehicle and Scenarios

Vehicle Description

The PCUAV NGM-II is an autonomous UAV developed within the MIT/Draper Technology Development Partnership Program (MDPP) at MIT[51][63]. The vehicle is part of a cooperative group of UAVs known collectively as the PCUAV system and is operated at medium and low altitudes. At these altitudes, surface obstacles such as trees (or tree trunks), buildings, communication towers and the like are abundant.

The NGM-II is an 8kg, fixed-wing, single-engine aircraft in a pusher configuration, with a wing-span of approximately 2m. It has two tail booms with a vertical stabilizer on each boom and a horizontal stabilizer extending between the booms. The vehicle's length from the tip of its nose to the trailing edge of its rudder is 1.5m. These dimensions and the vehicle's handling capabilities combined with a cruising airspeed of around 20m/s, imply that the NGM-II may be likened to a large sport or aerobatic model remote control (R/C) aircraft.

The vehicle is known for its exceptionally accurate autonomous control system, able to control its vertical and horizontal position to within 2m circular error probability (CEP) (3-D) of a desired tracking input. Onboard sensors include a 5Hz Global Positioning System (GPS), an Inertial Measurement Unit (IMU), pressure transducers for altitude and wind speed measurement and differential GPS capability. The vehicle's control system is most accurate when it locks onto a larger UAV within the PCUAV fleet, the Outboard Horizontal Stabilizer (OHS) Parent vehicle, using an optical tracking system.

A simplified 2-dimensional, linear time-invariant (LTI), closed-loop dynamic model of the aircraft, under free flight conditions, was created with the help of the architect of its control system, Sanghyuk Park[51]. The closed loop vehicle dynamics are represented by

the fourth-order state space model

$$\begin{bmatrix} \dot{x}_1 \\ \ddot{x}_1 \\ \dot{x}_2 \\ \ddot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -0.96 & -0.8 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -0.6 & -1 \end{bmatrix} \begin{bmatrix} x_1 \\ \dot{x}_1 \\ x_2 \\ \dot{x}_2 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0.96 & 0 \\ 0 & 0 \\ 0 & 0.6 \end{bmatrix} U(t) + \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} w(t); \tag{5.1}$$

with

$$E[w(t)w(t)^T] = \begin{bmatrix} 0.5 & 0 \\ 0 & 0.5 \end{bmatrix}. \tag{5.2}$$

The states are illustrated in Figure 5-1.

Conflict Scenarios

Two conflict scenarios where chosen. These scenarios represent typical, simple conflicts where both the expected and worst-case accuracy of the proposed calculation algorithms are probed, as illustrated in Figure 5-1.

The first scenario, on the right in Figure 5-1, is that of the NGM-II flying past a stationary, tree trunk-sized hazard in close proximity, while commanded to hold a straight-line trajectory with a forward airspeed of 20m/s, in a wind free environment. The simulation starts with a mean state distribution of

$$m_X = [2, 0, -21.8, 20]^T \tag{5.3}$$

and state covariance matrix

$$\Sigma_X = P_X = \begin{bmatrix} 0.3255 & 0 & 0 & 0 \\ 0 & 0.3125 & 0 & 0 \\ 0 & 0 & 0.4167 & 0 \\ 0 & 0 & 0 & 0.25 \end{bmatrix} \tag{5.4}$$

at $t = t_0$. We will refer to this as the "fly-by" scenario. This is considered to be a typical scenario for the NGM-II, flying straight line trajectories between waypoints at cruise speed. The proximity with which it passes the obstacle is representative of the maximum

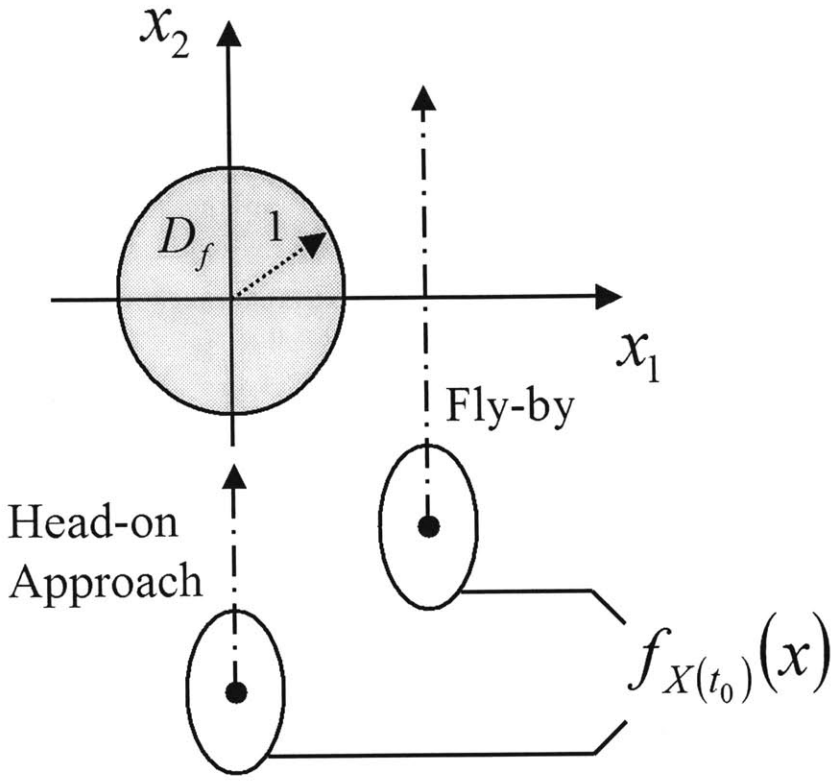

Figure 5-1: The Two PCUAV NGM-II Application Example Approach Scenarios

acceptable collision risk allowed for this vehicle, chosen to be $P_{c_{max}} = 0.05$. The resulting $P_c(t_h)$ will therefore be representative of a level of risk when collision avoidance action will typically be taken.

A mean hazard passage time (MPT) is provided for all fly-by scenarios, and is denoted by $\bar{t}_{MPT}$. The MPT indicates when a vehicle passed the center of the domain of failure. More, precisely, for the straight-line flight paths and circular shapes of D_f in this chapter, $\bar{t}_{MPT}$ is the only time when the mean of $\dot{\Gamma}$ is zero, i.e., it is changing sign. In this instance, $\bar{t}_{MPT} = 1.09$ seconds.

The second scenario, on the left in Figure 5-1, has the NGM-II approach a stationary, tree trunk-sized hazard on a direct collision course for the center of the hazard. The

simulation starts with a mean state distribution of

$$m_X = [0, 0, -24, 20]^T \tag{5.5}$$

and state covariance matrix

$$\Sigma_X = P_X = \begin{bmatrix} 0.3255 & 0 & 0 & 0 \\ 0 & 0.3125 & 0 & 0 \\ 0 & 0 & 0.4167 & 0 \\ 0 & 0 & 0 & 0.25 \end{bmatrix} \tag{5.6}$$

at $t = t_0$. This will be referred to as the "head-on" scenario. Such a conflict represents the worst-case (most calculation error, see Section 4.2.1) performance of the proposed collision probability calculation algorithm. This scenario also represents a clear case where collision is inevitable and where the probability of collision accumulates most rapidly over time. We will make use of the "head-on" simulation in order to validate the vehicle applicability test described in Section 4.2.1, and therefore terminate the simulation when $P_c(t) \approx P_{c_{max}}$.

Notice that no distinction is made between nominal and avoidance trajectories, since the algorithm itself makes no real distinction. It can however be proposed that the first scenario may be the result of either a nominal trajectory, or for example, an avoidance trajectory from a nominal "head-on" trajectory. In both cases, the domain of failure (hazard) D_f is described as a circular space around the vehicle, with unit radius (in meters). This is considered to correspond to an appropriate value of γ_0, given the vehicle's dimensions and a tree trunk of approximately 15cm radius.

5.1.2 Results

Fly-by Simulation

Figure 5-2 illustrates the results of accumulation of risk within a time window $t = [t_1, t_2]$, where $t_0 < t_1 < t_2 < t_h$, within a certain finite horizon time window. In each instance throughout this chapter, $t_1 = t_0 + 1$. The four distinct plots on this figure are related to each other as follows.

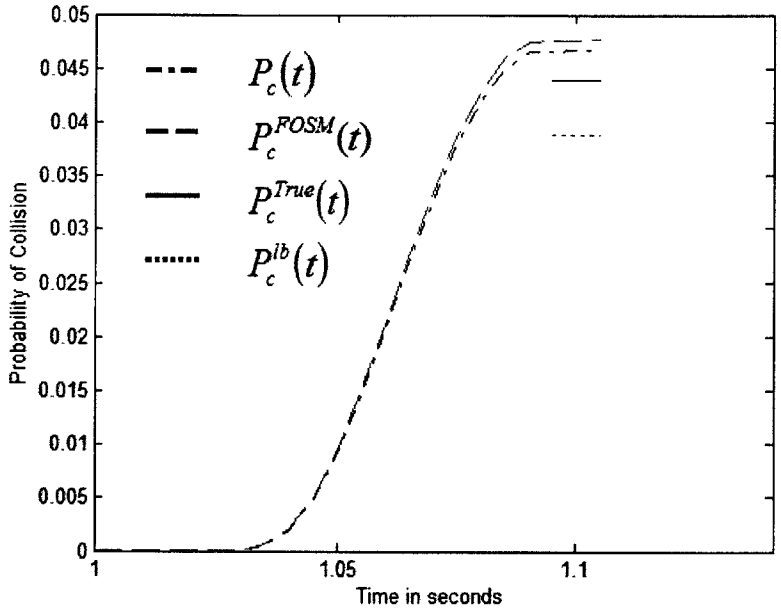

Figure 5-2: Risk Accumulation for PCUAV NGM-II Hazard Fly-by

The solid line represents a close approximation of the true probability of collision at time $t = t_h$. This is denoted by $P_c^{true}(t)$, or the "True" probability of collision, introduced in Section 4.1.3. This close approximate is calculated from an exhaustive and time-consuming, model-predictive Monte-Carlo simulation and is considered to be the benchmark for comparing the performance of the algorithm developed in Chapter 4.

The algorithms for calculation of collision risk, developed in this dissertation, are represented by two plots in Figure 5-2. One is denoted by $P_c(t)$, the calculated upper bound to collision risk, and another is denoted by $P_c^{FOSM}(t)$. The latter is an FOSM error corrected value of $P_c(t)$, accounting for calculation error induced by the independence assumption discussed in Section 4.2.1.

A lower bound probability of collision, $P_c^{lb}(t)$ is determined, as described by Equation 4.35. This representation of probability of collision is suggested by Prandini et al[54] and also by number of other publications on collision avoidance[12][11][2][44][3]. Adding

this plot to the figure allows us to clamp $P_c^{true}(t)$ between upper and lower bounds. In addition, it allows us to compare our work with comparable recent research on probabilistic collision avoidance.

Figure 5-2 illustrates how $P_c(t)$ and $P_c^{FOSM}(t)$ are accumulated over time, as the vehicle passes the hazard, again emphasizing the similarities between collision probability and the first passage time problem discussed in Section 2.3.2. The figure shows the upper-bounded nature of $P_c(t_h) = P_c(t_2)$, since it is larger than $P_c^{true}(t)$ and also shows the lower bounded nature of $P_c^{lb}(t)$, which is clearly smaller than $P_c^{true}(t)$.

$P_c^{FOSM}(t_h) = P_c^{FOSM}(t_2)$ is within eight percent of $P_c^{true}(t)$ and $P_c(t_h)$ is within six percent of $P_c^{true}(t_h)$. The seemingly more accurate $P_c(t)$ is not evidence of failure of the FOSM error correction method. $P_c(t)$ is an approximation of the upper bound of the probability of collision, denoted by $P_c^{ub}(t)$, and $P_c^{FOSM}(t_h)$ is a more exact approximate of $P_c^{ub}(t)$. It is circumstantial that $P_c(t_h) < P_c^{FOSM}(t_h)$, and it can be shown that the opposite inequality holds under different simulation conditions. Specifically, when the finite horizon accumulates more probability of collision while γ and $\dot{\gamma}$ are negatively correlated ($\rho < 0$), then $P_c(t_h) < P_c^{FOSM}(t_h)$. $P_c(t_h) > P_c^{FOSM}(t_h)$ when more risk is accumulated while $\rho > 0$. This is obvious when assessing Equation 4.55.

The independence assumption labelled "Assumption 1" in Section 4.2.1 accounts for the approximately two percent difference between $P_c^{FOSM}(t_h)$ and $P_c(t_h)$. The figure is well below the worst-case upper bound error of twenty-two percent derived during the vehicle applicability test example in Section 4.2.1. This can be expected, since the simulation is not a worst-case scenario, such as the head-on approach. It is under this worst-case assumption that the bound was derived.

$P_c^{lb}(t)$ is of special interest in Figure 5-2, since it is approximately fifteen percent smaller than $P_c^{true}(t_h)$. Using the lower bound alone in order to obtain an estimate of $P_c^{true}(t_h)$ would be much simpler to accomplish than the algorithm used to find $P_c(t_h)$. The danger involved in trusting in a lower bound of collision risk, combined with the associated estimation error, would however make this a risky venture. The difference between $P_c^{lb}(t)$ and

$P_c(t_h)$ may however serve as an insightful method of bounding $P_c^{true}(t_h)$.

It is interesting to note that $P_c^{lb}(t)$ quickly becomes a progressively worse measure of $P_c^{true}(t_h)$, the more time is spent in the vicinity of D_f. One such example is when orbiting a circular D_f at a constant radius. $P_c^{lb}(t)$ will not grow as time progresses, while both $P_c^{true}(t_h)$ and $P_c(t_h)$ would increase.

Head-on Simulation

The head-on approach shown in Figure 5-1 incites the worst-case difference between $P_c^{FOSM}(t_h)$ and $P_c(t_h)$ because of the maximal correlation between γ and $\dot{\gamma}$ when these vectors have co-linear gradients (see Equation 4.64 and the surrounding discussion). Figure 5-3 illustrates the results of the simulation.

Note the exponential increase in $P_c(t_h)$ and $P_c^{FOSM}(t_h)$ over time, as the vehicle inevitably enters D_f. The increase in risk is sudden and evolves from almost zero risk to $P_c^{FOSM}(t_h) = 0.053$ over the course of only about 0.06 seconds. It is this sudden increase in risk that accounts for the minimal difference between $P_c^{lb}(t_h)$ and $P_c^{true}(t_h)$. There is very little probability flow out of D_f, with the commanded cruise speed of 20m/s dominating the flow into D_f. This is therefore a scenario where the vehicle will spend very little time in close vicinity to D_f, resulting in an accurate $P_c^{lb}(t_h)$, approximately four percent smaller than $P_c^{true}(t_h)$.

At the same time, $P_c^{FOSM}(t_h)$ is also accurate, being approximately eight percent larger than $P_c^{true}(t_h)$. $P_c^{FOSM}(t_h)$ is therefore of similar accuracy for both the "fly-by" and "head-on" simulations. $P_c(t_h)$ undershoots $P_c^{true}(t_h)$ by about 10 percent.

$P_c^{FOSM}(t_h)$ is roughly eighteen percent larger than $P_c(t_h)$, which is close to the 22 percent approximate worst case "Assumption 1" calculation error predicted in Section 4.2.1. This simulation (and many similar simulations) therefore validates the approximate vehicle applicability test described in Section 4.2.1.

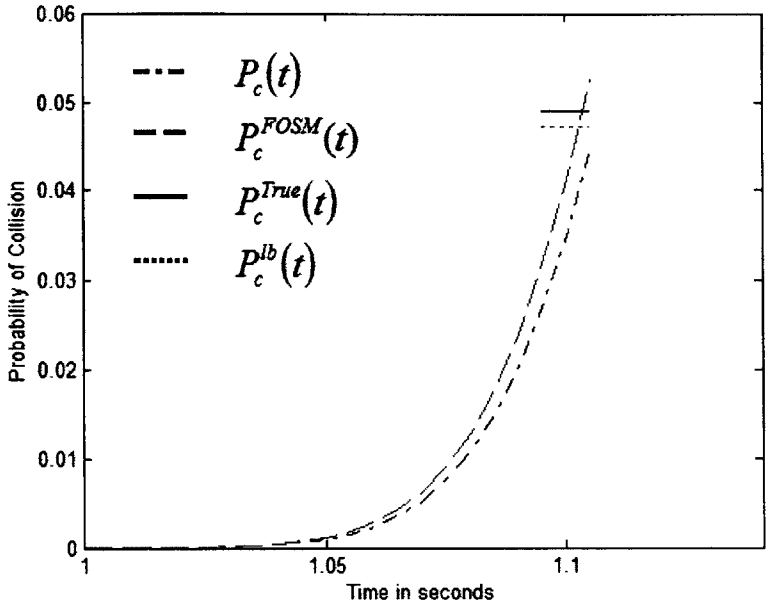

Figure 5-3: Risk Accumulation for PCUAV NGM-II Head-on Hazard Approach

5.2 Two Large Commercial Transports

5.2.1 Vehicle and Scenarios

Vehicle Description

This application example aims to represent conflict scenarios between two large commercial transports such as the Boeing 747-400, or vehicles of equivalent size and dynamics, under autopilot control.

The simplified two-dimensional LTI, closed-loop model of the vehicle, under autopilot control, was developed with the aid of Tom Reynolds at the International Center for Air Transportation (ICAT) at MIT and according to dynamics provided by Roskam[57]. A

fourth order representation can be summarized as

$$\begin{bmatrix} \dot{x}_1 \\ \ddot{x}_1 \\ \dot{x}_2 \\ \ddot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -0.2 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -0.2 & -1 \end{bmatrix} \begin{bmatrix} x_1 \\ \dot{x}_1 \\ x_2 \\ \dot{x}_2 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0.2 & 0 \\ 0 & 0 \\ 0 & 0.2 \end{bmatrix} U(t) + \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} w(t); \quad (5.7)$$

with

$$E[w(t)w(t)^T] = \begin{bmatrix} 0.5 & 0 \\ 0 & 0.5 \end{bmatrix}. \quad (5.8)$$

The model is further transformed to vehicle-relative space as to adhere to the single-vehicle-single-obstacle problem definition described in Section 3.2. The transformation results in a single vehicle model in conflict with a stationary D_f. D_f is described by a circular region with 40m radius, roughly accounting for the dimensions of two large transports.

Conflict Scenarios

Conflict scenarios similar to that described for the PCUAV NGM-II vehicle are investigated. In the case of the two-vehicle transport collision problem however, the fly-by scenario, on the right in Figure 5-4, corresponds roughly to two vehicles crossing each other's flight trajectories at the same altitude. The fly-by simulation starts with a mean state distribution of

$$m_X = [43, 0, -248, 200]^T \quad (5.9)$$

and state covariance matrix

$$\Sigma_X = P_X = \begin{bmatrix} 2.5 & 0 & 0 & 0 \\ 0 & 0.5 & 0 & 0 \\ 0 & 0 & 2.5 & 0 \\ 0 & 0 & 0 & 0.5 \end{bmatrix} \quad (5.10)$$

at $t = t_0$. For this scenario, the MPT is calculated as $\bar{t}_{MPT} = 1.24$ seconds.

The head-on conflict scenario, on the left in Figure 5-4, corresponds directly to two vehicles artificially commanded along trajectories that would result in direct head-on collision.

The simulation starts with a mean state distribution of

$$m_X = [0, 0, -246.5, 200]^T \tag{5.11}$$

and state covariance matrix

$$\Sigma_X = P_X = \begin{bmatrix} 2.5 & 0 & 0 & 0 \\ 0 & 0.5 & 0 & 0 \\ 0 & 0 & 2.5 & 0 \\ 0 & 0 & 0 & 0.5 \end{bmatrix} \tag{5.12}$$

at $t = t_0$.

Both scenarios are typical of situations that might be encountered under straight-line autopilot control between navigation waypoints. In this case however, the head-on simulation is prolonged in order to accumulate very large risk, about five times that of $P_{c_{max}} = 0.05$ chosen for these aircraft. Recall that we usually avoid calculating $P_c(t) > P_{c_{max}}$, since a number of analyses in Chapter 4 depend on $P_c(t)$ being small, including the vehicle applicability test and the FOSM calculation error corrections. The increased risk is therefore intended to expose inclement influences on the calculation of $P_c^{FOSM}(t)$ and the vehicle application test, if any.

5.2.2 Results

Fly-by Simulation

The fly-by simulation results in Figure 5-5 again attest to the accuracy of $P_c(t_h)$ and $P_c^{FOSM}(t_h)$ when compared to $P_c^{true}(t_h)$. $P_c(t_h)$ is approximately 8 percent larger than $P_c^{true}(t_h)$ and $P_c^{FOSM}(t_h)$ is about ten percent larger.

The difference between $P_c(t_h)$ and $P_c^{FOSM}(t_h)$ is only about 2 percent, much smaller than the predicted 18 percent approximate maximum from a vehicle applicability test. We again expected the improved calculation error performance, since the maximal error (as usual) was derived from a worst-case head-on approach.

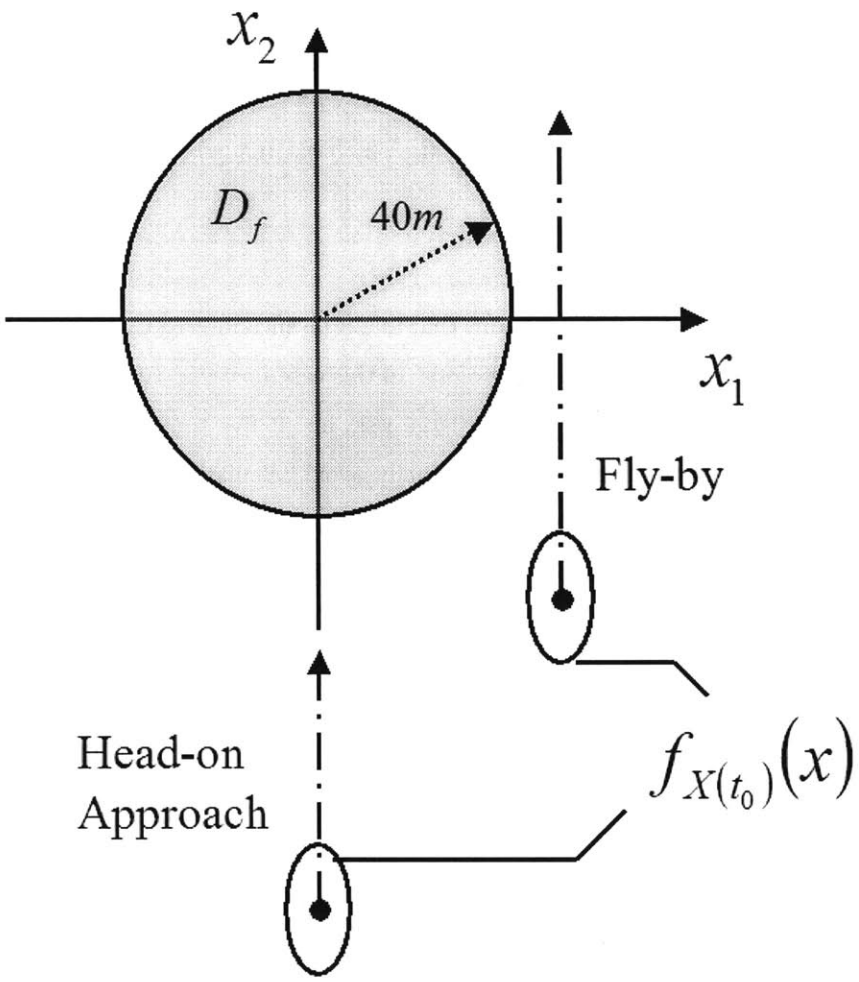

Figure 5-4: The Two Transport vs. Transport Application Example Approach Scenarios

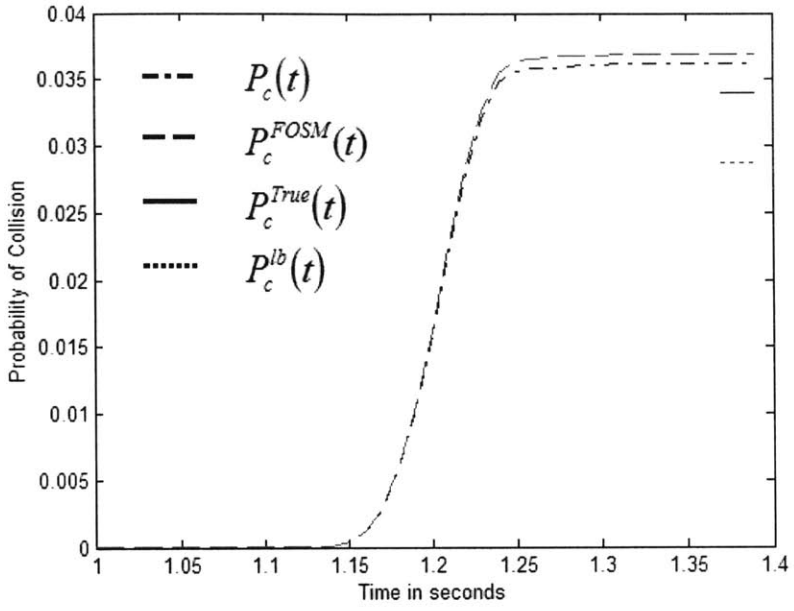

Figure 5-5: Risk Accumulation for Transport Fly-by

$P_c^{lb}(t_h)$ is approximately 18 percent smaller than $P_c^{true}(t_h)$. Approximating $P_c^{true}(t_h)$ with $P_c^{lb}(t_h)$ would grow progressively worse with increased durations of exposure to D_f. Further simulations have illustrated this decreased accuracy when the vehicles' relative velocities are reduced.

Typical risk accumulation is evident as a function of exposure time and proximity to D_f, similar to the results from other simulations, including the one resulting in Figure 5-2.

Head-on Simulation

The head-on conflict scenario for two large commercial transports was simulated and the collision probability results, for part of one finite time horizon, are shown in Figure 5-6. This scenario was specifically tailored to again induce the worst case performance of the algorithm for $P_c(t_h)$.

The typical exponential increase in risk probability is evident as the two vehicles in-

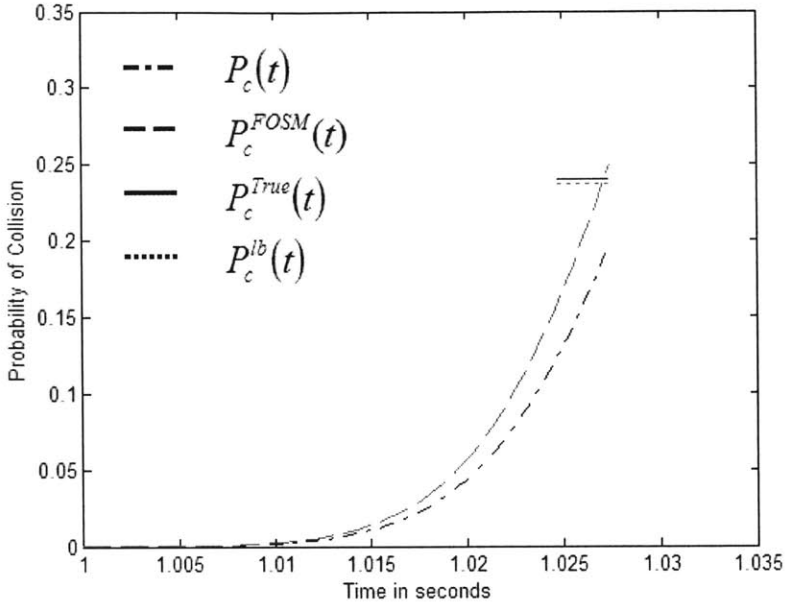

Figure 5-6: Risk Accumulation for Transport Head-on Approach

evitably head towards a collision. Relative vehicle velocities are approximately 200m/s for this simulation, resulting in a rapid rise in collision risk over time. As a consequence, $P_c^{lb}(t_h)$ and $P_c^{true}(t_h)$ are separated by about a percentage point. $P_c^{FOSM}(t_h)$ is approximately 4 percent larger than $P_c^{true}(t_h)$. Although the lower bound is therefore more accurate during this scenario, $P_c^{FOSM}(t_h)$ is very accurate.

$P_c(t_h)$ is 28 percent smaller than $P_c^{FOSM}(t_h)$, only 4 percent larger than the 24 percent value calculated with the approximate vehicle applicability test for this worst case scenario. When $P_c^{FOSM}(t) = 0.05$, $P_c(t_h)$ is only 22 percent smaller. As $P_c^{FOSM}(t_h)$ increases, the vehicle applicability test does understandably become slightly less accurate, since the test hinges on the assumption that $P_c^{FOSM}(t_h) < P_{c_{max}}$ is small. At the same time, and for the same reason, when $P_c^{FOSM}(t) = 0.05$, it is only 2 percent larger than $P_c^{true}(t)$, implying that $P_c^{FOSM}(t)$ is itself more accurate at lower collision risk. In all though, this simulation again validates the vehicle applicability test developed in Chapter 4 and $P_c^{FOSM}(t)$ again provides an accurate estimate of collision risk, with higher performance when $P_c^{FOSM}(t_h)$

is small.

5.3 Chapter Summary

The risk calculation algorithm in Figure 4-2 is implemented in a Matlab and Simulink environment. Two application examples are considered: A small UAV, the PCUAV NGM-II, encountering a tree trunk sized hazard; and two large commercial transports in hazardous proximity to each other. Two conflict scenarios are tested, illustrated and discussed for each application. It is shown that both the risk calculation and the error approximations perform expectedly well, thereby validating the developed approach.

Chapter 6

Conclusions and Future Work

This thesis further develops the field of probabilistic collision avoidance systems, specifically for autonomous aircraft, and builds primarily upon the works of Yang and Kuchar[70][69][42][41], but also on approaches by Paielli and Erzberger[49][50] and Prandini et al[54]. The problem is treated in the form of a multi-dimensional, non-cooperative game between two vehicles, or equivalently, one vehicle and an obstacle.

6.1 Primary Contributions

The following is a list of the primary contributions of this research:

1. Order reductive collision metrics are applied in order to provide a tractable algorithmic solution to the normally intractable first passage time problem of multi-dimensional collision risk calculation within a finite time horizon.

2. A tractable real-time algorithm is developed for the calculation of collision risk as a function of time along any deterministic flight trajectory.

3. Calculation and approximation errors resulting from application of the algorithm are characterized and quantified.

4. Simulations are employed to verify the accuracy of the collision risk algorithm and major calculation error components and to probe the applicability of the research to various application examples and conflict scenarios.

5. The concept of a system operating characteristic (SOC), developed by Kuchar[41], in aid of calculating alerting thresholds, is expanded to account for the safety benefits of delaying collision avoidance action. A condensed SOC (CSOC) results.

The remainder of this chapter describes the applicability of the research to various collision avoidance problems, further summarizes the conclusions and contributions made during the course of the research and lists future research opportunities stemming from this work.

6.2 Applicability and Comparisons

Section 2.2.3 describes how most conflict alerting systems may be divided into three categories, namely: Single path, worst case and probabilistic methods of approach. Single path and worst case approaches are typically attempts at simplifying the generally intractable probabilistic approach, in order to provide a tractable solution. Such approaches are therefore usually applicable to specialized collision avoidance problems and many are of an ad hoc nature. The method developed in this thesis, follows the probabilistic approach. This approach typically gains generality of application and accuracy, at the cost of calculation complexity. We show how calculation complexity may be reduced, while still retaining most of the advantages of the probabilistic approach.

The work expands upon the probabilistic approaches followed by Kuchar, Yang, Paielli and Erzberger and Prandini et al. It is closest in comparison to that of Paielli and Erzberger's approach, since their work is also only applicable to the two-vehicle problem and follows a similar stochastic development. Kuchar and Yang's work is applicable to multiple domains of failure, but their Monte Carlo approach sets it apart, and their solution to collision risk can be more approximate, in order to facilitate real-time tractability.

Our work primarily differs from that of Paiell and Erzberger in the following ways:

1. We calculate an upper bound to collision risk, which closely approximates the true collision risk, while Paielli and Erzberger's results may return upper or lower bound estimates, depending on the amount of time spent in proximity to a hazard. A feasible

upper bound provides added robustness to collision avoidance, while a lower bound may compromise on safety.

2. We developed an approach which is of more general application to collision avoidance problems. This fact stems from the fact that they assume constant vehicle velocities and constant state distributions over the conflict interval, while also not recognizing that additional conflict risk arises from exposure to a hazard over time. These assumptions may become problematic when considering, for example, a UAV loitering in the vicinity of a hazard, accumulating risk over time. In general, our approach is therefore also more accurate.

3. Our approach, even though it is tractable in real-time, is of much higher computational complexity than that of Paielli and Erzberger. Increased complexity might prohibit use on small, uncomplicated and inexpensive vehicles with limited processing power.

The work developed in this dissertation is aimed at application to autonomous vehicles. In cases where human intent may be modelled stochastically, it may also be applied to human piloted vehicles, with an associated increase in false alarm rates and a reduction in correct detection rates. The reduction in performance would result from a typical increase in uncertainty of intent, not algorithmic error.

Furthermore, the work is applicable to two-vehicle fixed and rotary wing aircraft problems, during glancing and loitering encounters with hazards, making it of very general use. The work is however not applicable to scenarios where the precise and complex shapes of hazards need to be maintained during risk calculation, such as the PCUAV autonomous mid-air docking system[51].

6.3 Summary and Conclusions

The following is a more detailed list of the approach and contributions of this research:

1. It is illustrated that automatic collision avoidance systems for autonomous vehicles need to be developed in order to: Allow UAVs to operate in FAA controlled airspace; increase the effectiveness of commercial aircraft collision avoidance when under au-

topilot control; and ensure safety when piloted and autonomous vehicles are required to interact.

2. Differences between piloted and autonomous vehicles that influence conflict avoidance design are discussed. The effectiveness of alerting systems for autonomous vehicles is shown to be less sensitive to the occurrence of false alarms. Uncertainties usually associated with future pilot intent can be reduced dramatically by deterministic autopilots, thereby opening the door to more precise probabilistic conflict detection.

3. The autonomous conflict detection and collision avoidance problem is cast into a state space problem, similar to that developed by Kuchar[41]. The autonomous vehicle problem is then extended into a closed loop representation where it is shown that mean and covariance propagation may be used to estimate future state uncertainty through finite horizon model predictive simulation. This development is a direct consequence of the reduction of uncertainty of pilot intent because of autonomous control. Pailli and Erzberger[50] also illustrate a similar initial development, but for vehicles with constant velocity.

4. The calculation of collision risk within a finite time horizon is shown to be equivalent to problems encountered in a number of separate fields of research. Two such fields are: Reliability theory, especially as applied to civil engineering problems where catastrophic failures are characterized as a function of time; and the first passage time problem described by Kolmogorov equations, or as special cases thereof, such as the Fokker-Planck equation often encountered in physical problems and stochastic control.

5. The calculation of collision risk within a finite and discretized time horizon is cast as a reliability theory problem and developed from first principles. The solution is developed as a set of discrete propagation equations illustrated in Figure 3-3 and it is shown that the calculations are numerically intractable when real-time solutions are required.

6. The notion of the system operating characteristic (SOC)[42][41] employed as a trade-off between $P(SA)$ and $P(UA)$ when making alerting decisions is extended. The SOC at any point in time is based on the outcome of making an avoidance decision at that

time. It is shown that the effects of time-delayed avoidance action may be incorporated into the SOC representation, resulting in a best-case or time condensed SOC (CSOC), for a specific time horizon and a given set avoidance trajectories. This allows us to make use of signal detection theory in order to optimize alerting thresholds, while at the same time investigating the pay-off of delayed avoidance action. As a direct consequence, the work of Kuchar may be employed when optimizing alerting thresholds[41].

7. An intuitive collision metric is introduced that is defined as a positive definite quadratic measure of state space. In three-dimensional position space the metric manifests itself as a spheroid domain of failure or hazard space surrounding vehicles and hazards. The quadratic form is shown to act as a order reductive metric, compressing n-dimensional state space into a one dimensional metric.

8. It is shown how the quadratic collision metric lends itself to solving the collision risk algorithm illustrated in Figure 3-3 through simple and proven approximations, resulting in the tractable algorithm described in Figure 4-2. The intractable $\mathcal{O}(N^{2n})$ numerical complexity required by the solution in Figure 3-3 at each time step in a finite horizon window of solution is thereby replaced by significantly reduced $\mathcal{O}(N^2)$ complexity.

9. We also illustrate how research lead by Jaschke[35] and Hughett[32] in the field of econometrics may be employed to efficiently solve for the quadratic PDFs required by the risk solution algorithm in Figure 4-2. The calculations are shown to be solvable through Fourier inversion of known probabilistic characteristic functions, and we show how to employ Inverse Fast Fourier Transforms (IFFT) in order to find the PDFs according to predefined bounds of representation accuracy (error). The bounds are linked to the characteristic function, which is known exactly before inversion takes place. In this way, we may guarantee and control calculation error bounds even before attempting IFFTs. The efficient IFFT solutions of guaranteed accuracy allow further reduction of computational complexity at each time step in the finite horizon to $\mathcal{O}(Nlog_2(N))$ for the algorithm in Figure 4-2.

10. A number of properties of the time derivative of the collision metric are derived. The

risk solution algorithm in Figure 4-2, for example, requires that the PDF of the time derivative of the quadratic collision metric be calculable. It is therefore shown that this derivative is itself a quadratic form for most dynamic models of second order or more. It is also shown that we may always convert the representation of the rate of change into a quadratic form of second order or more. Jaschke and Hughett's methods may thus also be used to solve for the required PDF of $\dot{\gamma}$.

11. The calculation errors induced by the assumptions made to convert the intractable algorithm in Figure 3-3 to the tractable version displayed in Figure 4-2 are identified. Each type of error is either quantified exactly, upper bounded or approximated through FOSM methods, depending on the nature and severity of the error. These calculations may be performed during real-time simulation, in order to provide error estimates on each estimation of collision risk.

12. A vehicle applicability test was devised which can be run off-line, before implementing the bulk of the solution strategy. The test gauges the worst-case error of the solution strategy based on the bulk of the calculation error.

13. Practical aspects such as simulation step size, the time resolution of investigating avoidance trajectories and the horizon time length are investigated. The major contributing factors influencing these aspects are identified and a number of metrics are suggested in order to constrain and optimize these quantities.

14. The risk calculation algorithm in Figure 4-2 is implemented in a Matlab and Simulink environment. Two application examples are considered: A small UAV, the PCUAV NGM-II, encountering a tree trunk sized hazard; and two large commercial transports in hazardous proximity to each other. Two conflict scenarios are tested, illustrated and discussed for each application. It is shown that both the risk calculation and the error approximations perform expectedly well, thereby validating the developed approach. At the same time, results are compared to that of a lower bound estimate of collision risk suggested by Prandini et al and a host of other publications.

6.4 Future Work

Various avenues of future work result from the research detailed in this thesis, many of which fall outside of the classical realm of probabilistic collision avoidance. A select number of avenues are briefly discussed in this section.

6.4.1 General State Avoidance

The problem of collision avoidance is generally treated as one of avoiding a time variant positional state space, i.e., $D_f(t) \subset R^m$ where $m \leq 3$. That is, the problem is a subset of the larger problem of general state avoidance, where $m \leq n$.

General state avoidance may be employed in a wide variety of situations, including: Avoiding hazardous domains of position and velocity during autonomous landing; Avoiding hazardous domains position, velocity and acceleration for agile maneuvering autonomous vehicles; and Avoiding hazardous domains of orientation outside of, for example, GPS antenna beam width and tilt sensor range.

The research detailed in this document may be adapted for use as a general state avoidance system, or at least for determining the risk of transgression into a general hazardous state domain. In order to do this, either the $D_f(t)$ of $D_f^c(t)$ needs to be quadratically bounded or approximated as such. Most of the derivations in Chapter 4 may be extended to general quadratic metrics of state, not only operating on position space. The metric must however always be positive definite, and the FOSM error correction of $P_c(t)$ would need to be extended to be able to deal with quadratic forms that are not complete squares.

6.4.2 Control and Trajectory Scheduling for Minimal Risk

The probabilistic methods discussed here are aimed at the evaluation of conflict risk along a pre-defined set of flight trajectories. The size and variety of this set and the time-resolution at which avoidance trajectories are investigated can be traded off against effectiveness of collision avoidance $P(SA)$ and computational complexity. At least three schools of thought exist that allow us to revise the notion of employing pre-defined sets of trajectories:

Firstly, the set of trajectories may be very large, but a selection algorithm may be developed to rank or prune the set, in real-time, given the current state estimation and the nominal trajectory. In this way, only a selected subset of the original set, and of tractable size, may be entertained during each finite horizon.

Secondly, the concept of pre-defined trajectories may be replaced entirely by alternative forms of collision avoidance control. The model-predictive nature and step-wise propagation embedded in the approach detailed in this thesis, allow for the application of optimization methods employing, for example, scaled control input primitives along the course of a finite horizon in order to optimize the collision risk outcome. In this way, collision avoidance is a matter of tweaking nominal trajectories to bend around hazards, without radical changes in trajectory. Similar research is being undertaken at the Charles Stark Draper Laboratory.

Thirdly, pre-defined sets of avoidance trajectories may be replaced by a single nominal path, with control around this path being altered by non-linear control laws. Such control laws may be dormant when collision risk is minimal, but may become effective in close proximity to hazards. These laws may either be gain scheduled according to collision risk, or may be more directly reliant on geometrical properties of conflict scenarios such as laws of inverse proportional navigation. Methods of adjoint analysis may, for example, be applied in order to efficiently calculate miss distances and trajectory changes.

6.4.3 n-Multiple and Compound Vehicles

The work may be extended from non-cooperative two vehicle or one-vehicle-one-obstacle applications. Direct application of the research would imply that n-multiple vehicle problems be divided into $(n-1)$ separate conflict analyses running on each one of n different vehicles. This might result in extreme computational burden. Complicated scenarios may also be envisioned around airports or within fleets of cooperative UAVs, where non-cooperation might lead to acutely sub-optimal avoidance capability. Various methods of cooperative data analysis should be researched if this work is to be adapted to such scenarios.

Finally, each vehicle might also be divided into compound sets of intersecting, or non-intersecting domains of failure. This is linked to the computational complexity we see when

dealing with n-multiple vehicles, and intersecting domains would pose yet another probabilistic challenge.

Dividing the multi-vehicle collision risk problem into 2-vehicle pairs, however also begs recognition of the fact that the risk of collision between two vehicles is conditional on the risk between all other pairs. This issue would have to be resolved before the research can be applied to n-multiple and compound vehicles.

Bibliography

[1] Joongki Ahn, Suchang Lee, and Jinsoo Kim. A robust approach to pre-concept design of UCAVs, considering survivability. *9th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, September 2002. Agency for Defense Development.

[2] M. R. Akella and K. T. Alfriend. Probability of collision between space objects. *Journal of Guidance, Control, and Dynamics*, 23(5):769–772, 2000.

[3] K. T. Alfriend. Probability of collision error analysis. *Journal of Space Debris*, 1:21–35, 1999.

[4] J. W. Andrews. A relative motion analysis of horizontal collision. *SAFE Journal*, 8(2), 1978.

[5] Alfredo Hua-Sing Ang and Wilson H. Tang. *Probability Concepts in Engineering Planning and Design, Basic Principles*, volume I - Basic Principles. John Wiley and Sons, July 1975.

[6] O. Barndorff-Nielsen and C. Kluppelberg. Tail exactness of multivariate saddlepoint approximations. *Scand. J. Statist*, 26:253–264, 1999.

[7] Berry et al. *2001-2002 Military Almanac.* Center for Defense Information (CDI), 2002.

[8] Boff et al. *Handbook of Perception and Human Performance: Sensory Processes and Perception, Cognitive Processes and Performance.* John Wiley and Sons, December 1986.

[9] H. Bohman. A method to calculate the distribution function when the characteristic function is known. *BIT*, 10:237–242, 1970.

[10] Andrei N. Borodin and Paavo Salminen. *Handbook of Brownian Motion: Facts and Formulae.* Birkhauser Verlag, 2nd edition, 2002.

[11] F. Ken Chan. Analytical expressions for computing spacecraft collision probabilities. *AAS/AIAA Spaceflight Mechanics Meeting*, February 2001.

[12] F. Ken Chan. Collision probability analysis for earth-orbiting satellites. *Core Technologies for Space Systems Conference*, 2001.

[13] Lt Col USAF Richard M. Clark. *Uninhabited Combat Aerial Vehicles - Airpower by the People, For the People, But Not with the People.* Air University Press, August 2000. CADRE Paper No. 8.

[14] W. G. Cochran. *Proc. Camb. Philos. Soc.*, 30:178–191, 1934.

[15] S. H. Crandall. First-crossing probabilities of the linear oscillator. *Journal of Sound and Vibration*, 12(3):285–299, 1970.

[16] S. H. Crandall et al. Some first-passage problems in random vibration. *Journal of Applied Mechanics, ASME Transactions*, 33:532–538, 1966.

[17] R. B. Davies. Numerical inversion of characteristic functions. *Biometrika*, 60:415–417, 1973.

[18] Efficient Eigenvalue Decomposition. X-Plore Statistics web site. URL: http://www.xplore-stat.de/tutorials/approxvarnode3.html.

[19] Gael Desilles. Differential Kolmogorov equations for transiting processes. Master's thesis, Massachusetts Institute of Technology, June 1998.

[20] A. C. Drumm. Lincoln Laboratory evaluation of TCAS II logic version 6.04a, Lincoln Laboratory, MIT, Volume I, 1996. Technical report.

[21] M. S. Eby. A self-organizational approach for resolving air traffic conflicts. *Lincoln Laboratory Journal*, 7(2), 1994.

[22] Ross L. Finney and George B. Thomas, Jr. *Calculus.* Addison-Wesley Publishing Company, 1990.

[23] R. A. Fisher. *Metron*, 5:90–104, 1925.

[24] R. L. Ford. The protected volume of airspace generated by an airborne collision avoidance system. *Journal of Navigation*, 39(2):139–158, 1986.

[25] R. L. Ford. The conflict resolution process for tcas ii and some simulation results. *Journal of Navigation*, 40(3):283–303, 1987.

[26] R. L. Ford and D. L. Powell. A new threat detection criterion for airborne collision avoidance systems. *Journal of Navigation*, 43(3):391–403, 1990.

[27] Emilio Frazzoli. *Robust Hybrid Control for Autonomous Vehicle Motion Planning.* PhD thesis, Massachusetts Institute of Technology, June 2001.

[28] Crispin W. Gardiner. *Handbook of Stochastic Methods for Physics, Chemistry, and the Natural Sciences.* Springer-Verlag, 2nd edition, 1985. Springer Series in Synergetics Vol. 13.

[29] Arthur Gelb et al. *Applied Optimal Estimation.* The M.I.T. Press, 1974.

[30] C. M. Haissig et al. Designing and airborne alerting system for closely spaced parallel approaches. *AIAA 99-3986*, pages 280–287, 1999.

[31] Robert V. Hogg. On the independence of certain Wishart variables. *Annals of Mathematical Statistics*, 34(3):935–939, September 1963.

[32] Paul Hughett. Error bounds for numerical inversion of a probability characteristic function. *SIAM J. Numer. Anal.*, 35(4):1368–1392, August 1998.

[33] M. Innocenti. Air traffic management using probability function fields. *AIAA-99-4149*, pages 1088–1097, 1999.

[34] Stefan R. Jaschke. General properties of delta-gamma-normal models. Web URL. http://www.xplore-stat.de/tutorials/approxvarnode3.html.

[35] Stefan R. Jaschke. The cornish-fisher-expansion in the context of delta-gamma-normal approximations. Weiestraβ-Institut fur Angewandte Analysis und Stochastic, Berlin, Germany, December 2001.

[36] D. G. Jones and M. R. Endsley. Sources of situation awareness errors in aviation. *Aviation, Space and Environmental Medicine*, 67(6):507–512, 1996.

[37] Thomas Jones and Jorge Lozano. PCUAV web site. Web URL. http://web.mit.edu/aeroastro/pcuav/.

[38] Myron Kayton. One hundred years of aircraft electronics. *Journal of Guidance, Control, and Dynamics*, 26(2):193–213, March 2003.

[39] Samuel Kotz et al. *Encyclopedia of Statistical Sciences*, volume 7: Plackett Family of Distributions to Regression, Wrong. John Wiley and Sons, 1986. pages 375-378.

[40] J. Krozel et al. Free flight conflict detection and resolution maneuvers for air traffic management. *AIAA Guidance, Navigation, and Control Conference*, 1996.

[41] James K. Kuchar. *A Unified Methodology for the Evaluation of Hazard Alerting Systems.* PhD thesis, Massachusetts Institute of Technology, 1995.

[42] James K. Kuchar. Methodology for alerting-system performance evaluation. *AIAA Journal of Guidance, Navigation, and Dynamics*, 19(2), March-April 1996.

[43] James K. Kuchar and Lee C. Yang. Survey of conflict detection and resolution modelling methods. *AIAA Guidance, Navigation, and Control Conference*, August 1997.

[44] R. A. LeClair. Probability of collision in the geosynchronous orbit. *Space Control Conference, MIT Lincoln Laboratory*, April 2000.

[45] Y. S. Lee and T. K. Lin. Higher order cornish-fisher expansion. *Journal of Applied Statistics*, 41:233–240, 1992.

[46] J. R. Lucas and J. J. Gallimore. Evalulation of human performance in a simulated ucav control station. *Proceedings of the Summer Computer Simulation Conference (SCSC)*, July 2000.

[47] James Masey et al. *Shepard's Unmanned Vehicles Handbook 2002.* The Shephard Press, 2002.

[48] C. A. Miller et al. Initiatives to improve tcas-atc compatibility. *Journal of ATC*, July-September 1994.

[49] Russel A. Paielli and Heinz Erzberger. Conflict probability estimation for free flight. *Journal of Guidance, Control, and Dynamics*, 20(3):588–596, May-June 1997.

[50] Russel A. Paielli and Heinz Erzberger. Conflict probability estimation generalized to non-level flight. *Air Traffic Control Quaterly*, 7(3), October 1999.

[51] Sanghyuk Park. Integration of parent-child unmanned air vehicle focusing on control system development. Master's thesis, Massachusetts Institute of Technology, February 2001.

[52] Peyton Z. Peebles, Jr. *Probability, Random Variables, and Random Signal Principles.* McGraw-Hill, 3rd edition, 1993.

[53] Charles L. Phillips and H. Troy Nagle. *Digital Control System Analysis and Design.* Prentice Hall International, Inc., 3rd edition, 1995.

[54] M. Prandini et al. A probabilistic framework for aircraft conflict detection. *AIAA Guidance, Navigation, and Control Conference*, August 1999.

[55] John G. Proakis and Dimitris G. Manolakis. *Digital Signal Processing: Principles, Algorithms, and Applications", publisher = Prentice Hall, year = 1996, edition = 3rd,.*

[56] RTCA (Radio Technical Committee on Aeronautics). Minimum performance specifications for tcas airborne equipment. Doc. no. rtca/do-185, RTCA, September 1983.

[57] Jan Roskam. *Airplane Flight Dynamics and Automatic Flight Controls Part II: Rigid and Elastic Airplane Dynamics and Automatic Flight Controls.* Roskam Aviation and Engineering Corporation, 1979.

[58] C.P. Sanders. Real-time collision avoidance for autonomous vehicles. Master's thesis, Massachusetts Institute of Technology, January 1998.

[59] M. Shewchun and E. Feron. Linear matrix inequalities for analysis of free flight conflict problems. *IEEE Conference on Decision and Control*, pages 2417–2422, 1997.

[60] M. Shinozuka. On the one-sided two-barrier problem. *Journal of applied Probability*, 2(1):79, 1965.

[61] M. Shinozuka and J. T. P. Yao. On the two-sided time-dependent barrier problem. *Journal of Sound and Vibration*, 6(1):98–104, 1967.

[62] R. Teo and C.J. Tomlin. Computing danger zones for provably safe closely spaced parallel approaches. To appear, AIAA Journal of Guidance, Control, and Dynamics, 2002.

[63] Francois Urbain. Vehicle design, flight control avionics, and flight tests for the parent and child unmanned air vehicle. Master's thesis, Massachusetts Institute of Technology, August 2001.

[64] E. H. Vanmarcke. On the distribution of the first-passage time for normal stationary random processes. *Trans. Am. Soc. Mech. Eng, Journal of Applied Mechanics*, (75-APMW-12):1–6, 1975.

[65] Daniele Veneziano. Crossings and extremes of random functions. C.N.R. / University of Florence technical seminar report, Universita' di Firenze; Ingegneria Strutturale, June 1979. Ref: UFIST/05/1979.

[66] A. Warren. Medium term conflict detection for free routing: Operational concepts and requirements analysis. *16th AIAA/IEEE Digital Avionics Systems Conference Proceedings*, 9(3):27–34, 1997.

[67] T. Williamson et al. Development and operation of the traffic alert and collision avoidance system. *Proceedings of the IEEE*, 77(11), 1989.

[68] Lee F. Winder and James K. Kuchar. Evaluation of collision avoidance maneuvers for parallel approach. *Journal of Guidance, Control, and Dynamics*, 22(6):801–807, 1999. AIAA.

[69] Lee C. Yang. *Aircraft Conflict Analysis and Real-time Conflict Probing Using Probabilistic Trajectory Modeling.* PhD thesis, Massachusetts Institute of Technology, 2000.

[70] Lee C. Yang and James K. Kuchar. Prototype conflict alerting logic for free flight. *Journal of Guidance, Control, and Dynamics*, 20(4):768–773, July-August 1997. AIAA.

[71] K. Zeghal. Towards the logic of an airborne collision avoidance system which ensures cooperation with multiple cooperative intruders. *ICAS Proceedings*, (3):2208–2218, 1994.

[72] Dennis G. Zill and Michael R. Cullen. *Advanced Engineering Mathematics.* PWS-Kent, 1st edition, 1992.

Appendix A

Matlab Simulation Code

A.1 Sample Code for Calculation of Probability of Collision Within One Finite Horizon

```
% Collision Risk Calculation Algorithm Main File
% Thomas Jones
% Units:
%   All times in seconds
%   All translations in m
%   All velocities in m/s
%   INCLUDING FOSM ERROR CORRECTION
% This example calculates collision risk for a short, specific
% PCUAV NGM-II fly-by encounter, longer intervals
% require discretization step size scheduling not shown here

close all;
clear all;

%figure;

% SETUP
```

```
% Starttime in seconds
    t0=0;
% Total simulation time in seconds
    totalsimtime=0.11;
% Lateral control time constant in rad/s
    vehicletimeconstant=1.5;
% Collision zone radius in meters
    R0=1;
    Gam0=R0^2;
% Deffinition of collision zone
    Gam=[1 0 0 0;0 0 0 0;0 0 1 0;0 0 0 0];
% Simplified collision zone rate for error calculation
    GamD=[0 1 0 0;1 0 0 0;0 0 0 1;0 0 1 0];
% Characteristic function frequency range +-
    wmax=800;
% Number of discretizsation points
    N=2^17; % Keep 2^k to use FFT i.s.o. DFT
% Average forward velocity in m/s
    Vavg=20;
% Dimensions/System order
    n=4;
% Number of Reference inputs
    nu=2;
% Vehicle reference inputs (Required flight path)
    Uref1=2*ones(1,22);
    Uref2=linspace(2.2,4.4,22);
    Uref=[Uref1;Uref2];

% Vehicle initial states
    X0=[Uref(1,1) 0 -1.8 Vavg];

% Find simulation parameters based on above data (rule of thumb)
```

```
    horizontime=0.11;
    simtimestep=0.005;
    manuvtimestep=horizontime/5;

% Get Vehicle Model
    pcuavmodelextract1;

for t=[t0:simtimestep:totalsimtime-horizontime] % Main simulation loop
    % Next Horizon interval time vector
        th=[t:simtimestep:t+horizontime-simtimestep];
        Lt=length(th);
    % Create Px and Mx time history over this horizon
        Px=Pss;  % Make this more complicated later, if required

        SYS = SS(A,Bu,C,D); % From pcuavmodelextract1 function call
        U=Uref(:,t/simtimestep+1:Lt+t/simtimestep);

        Mx=LSIM(SYS,U,th-t,X0);

    % Propagate and accum. Pc and Pcerr over horizon
        % Initialise prob of collision vector for this horizon
            PcH=zeros(1,Lt);
        % Initialize covariance vector for this horizon
            Rho=zeros(1,Lt);    %Cor. Coef
            Merr=zeros(1,Lt);   %Mean error
            Verr=zeros(1,Lt);   %Variance error
            deltaPerr=zeros(1,Lt); % Change in Prob Col error
        k=0;
    for th=[t:simtimestep:t+horizontime-simtimestep]    % Begin Horizon Loop
        k=k+1;
        % Find density function of Gamma
            [GamDist,GamVec,GamCharac]=finddist(Mx(k,:)',...
```

```
      Px,Gam,wmax,N,0,0,zeros(1,n),1);
  % Get GamDot into correct form for finddist
      % No need when remnants used and added to tfm in finddist
  % Find density function of GammaDot
      [deltaGamDotDist,deltaGamDotVec,deltaGamDotCharac]=finddist(Mx(k,:)',...
       Px,2*simtimestep*A.'*Gam,wmax,N,0,0,2*simtimestep*U(:,k).'*Bu.'*Gam,0);

  % Calculation error estimates
  % findrho finds quadratic form statistics
  [Rho(k),stdA,stdB,meanA,meanB]=findrho(Gam,simtimestep*GamD,Mx(k,:)',Px);
  varA=(stdA)^2;
  varB=(stdB)^2;
  mslope=(-varB+varA+sqrt(varB^2-2*varA*varB+varA^2+...
      4*Rho(k)^2*varA*varB))/2/Rho(k)/stdA/stdB;
  if(k>1)
      dP=PcH(k)-PcH(k-1);
      ma=Gam0; % A close approx, worst case is =0
      %ma=0;    % Worst-case (these options exist to simplify)
      deltam=1/mslope*dP/(1-dP)*(meanA-ma);
      deltav=0; % Negligible effect ignored
  else
      deltam=0;
      deltav=0;
  end;

  Merr(k)=deltam;
  Verr(k)=deltav-2*Rho(k)*stdA*stdB;

% Accumulate Probability of Collision

if k>1    % Start Risk Accumulation
```

```
% NB: May REDUCE Calculations if Gam<0 is not carried around!!
% Cut density function of Gamma into Df and Ds parts f=g1+g2
   leq=( sqrt(varA+Verr(k)) )/stdA*Gam0-Merr(k)
   I=find(GamVecPrev>leq);
   GamDistPrev=abs(GamDistPrev);
   GamDistPrev=GamDistPrev/sum(GamDistPrev)*2*wmax;
   g1=[GamDistPrev(1:I(1)-1) zeros(1,I(1))];
   g2=[zeros(1,I(1)-1) GamDistPrev(I(1):N)];
   PosMinPc=sum(g1)/(2*wmax);

% Find Characteristic function of new Gamma

    PropGamDistCut=g2/(1-PosMinPc);
    PropGamVecNext=GamVec;
    PropGamDistCut=[PropGamDistCut(N/2+1:N) PropGamDistCut(1:N/2)];
    PropGamCharacCut=fft(PropGamDistCut);
% Plots for debugging
    %figure;
    %plot(GamVec,deltaGamDotDist);

   % Convolve
    PropGamCharacNext=PropGamCharacCut.*deltaGamDotCharacPrev;
    PropGamDistNext=ifft(PropGamCharacNext);
   % Flip, normalize and remove numerical IFFT imaginary errors
    PropGamDistNext=[PropGamDistNext(N/2+1:N) PropGamDistNext(1:N/2)];
    PropGamDistNext=abs(PropGamDistNext);
    PropGamDistNext=PropGamDistNext/sum(PropGamDistNext)*2*wmax;

 % Plots for debugging
    %figure;
    %plot(PropGamVec,PropGamDistNext);
```

```
      g1=[PropGamDistNext(1:I(1)-1) zeros(1,I(1))];
      g2=[zeros(1,I(1)-1) PropGamDistNext(I(1):N)];
      deltaPc=sum(g1)/(2*wmax);
      PcH(k)=PcH(k-1)+deltaPc*(1-PcH(k-1));

   % Pause for stepping and debugging
       %pause;


else
     leq=( sqrt(varA+Verr(k)) )/stdA*Gam0-Merr(k)
     I=find(GamVec>leq);
      g1=[GamDist(1:I(1)-1) zeros(1,I(1))];
      g2=[zeros(1,I(1)-1) GamDist(I(1):N)];
     deltaPc=sum(g1)/(2*wmax);
     PcH(k)=deltaPc;
     MinPc=deltaPc; % find risk for Prandini's lower bound
     PosMinPc=0;  %Just initialize for subsequent if loop
     PropGamDistNext=GamDist;
end;              % End Risk Accumulation

% Update Variables for next loop
GamVecPrev=GamVec;
GamDistPrev=GamDist;
deltaGamDotCharacPrev=deltaGamDotCharac;

% Find the Prandini's lower bound so far
if PosMinPc>MinPc
     MinPc=PosMinPc;
end;
```

```
      if mod(k,1)==0
           %figure;
           disp(k);
           %plot(GamVec,PropGamDistNext,'b');
           %hold on;
           %plot(GamVec,GamDist,'r');
           %hold off;
           %pause;
      end;

    end;    % End Horizon loop

    if mod(t,0.5)==0
        disp(t);
    end;

end;    % End Main simulation loop

figure;
title{'Accumulated Prob. of Col. vs. sample'};
plot(PcH);
hold on;
plot(PcH-deltaPerr,'r');
plot(MinPc*ones(1,Lt),'k');

grid on;
zoom on;

legend('Upper Bound Pc','FOSM corrected Pc','Lower Bound Pc');

%title('Pc');
%figure;
```

```
%plot(Rho);
%title('Rho');
%figure;
%plot(Merr);
%title('Merr');
%figure;
%plot(Verr);
%title('Verr');
```

A.2 Sample Code for Calculation of Statistics of Two Dependent Quadratic Forms, "findrho.m":

```
% Thomas Jones
% Feb 2003
% Finding Covariance between symmetric
% quadratic forms: y=x'Ax and z=x'Bx

function [rho,stdA,stdB,meanA,meanB]=findrho(A,B,mx,sigx)

% Means
meanA=trace(A*sigx)+mx.'*A*mx;
meanB=trace(B*sigx)+mx.'*B*mx;

% Standard Deviations
stdA=sqrt(2*trace(A*sigx*A*sigx)+4*mx.'*A*sigx*A*mx);
stdB=sqrt(2*trace(B*sigx*B*sigx)+4*mx.'*B*sigx*B*mx);

% Correlation Coefficient
rho=(2*trace(A*sigx*B*sigx)+4*mx.'*A*sigx*B*mx)/stdA/stdB;
```


5000
23