Linköping studies in science and technology. Dissertations.
No. 1231

Ef cient Estimation and Detection Methods for Airborne Applications

Per-Johan Nordlund

Department of Electrical Engineering
Linköping University, SE–581 83 Linköping, Sweden

Linköping 2008

Linköping studies in science and technology. Dissertations.
No. 1231

Ef cient Estimation and Detection Methods for Airborne Applications

Per-Johan Nordlund

perno@isy.liu.se
www.control.isy.liu.se
Division of Automatic Control
Department of Electrical Engineering
Linköping University
SE–581 83 Linköping
Sweden

ISBN 978-91-7393-720-7 ISSN 0345-7524

Copyright © 2008 Per-Johan Nordlund

Printed by LiU-Tryck, Linköping, Sweden 2008

To Caroline and Elliot

Abstract

The overall purpose with this thesis is to investigate and provide computationally efficient methods for estimation and detection. The focus is on airborne applications, and we seek estimation and detection methods which are accurate and reliable yet effective with respect to computational load. In particular, the methods shall be optimized for terrain-aided navigation and collision avoidance respectively. The estimation part focuses on particle filtering and the in general much more efficient marginalized particle filter. The detection part focuses on finding efficient methods for evaluating the probability of extreme values. This is achieved by considering the, in general, much easier task to compute the probability of level-crossings.

The concept of aircraft navigation using terrain height information is attractive because of the independence of external information sources. Typically terrain-aided navigation consists of an inertial navigation unit supported by position estimates from a terrain-aided positioning (TAP) system. TAP integrated with an inertial navigation system is challenging due to its highly nonlinear nature. Today, the particle filter is an accepted method for estimation of more or less nonlinear systems. At least when the requirements on computational load are not rigorous. In many on-line processing applications the requirements are such that they prevent the use of the particle filter. We need more efficient estimation methods to overcome this issue, and the marginalized particle filter constitutes a possible solution. The basic principle for the marginalized particle filter is to utilize linear and discrete substructures within the overall nonlinear system. These substructures are used for efficient estimation by applying optimal filters such as the Kalman filter. The computationally demanding particle filter can then be concentrated on a smaller part of the estimation problem.

The concept of an aircraft collision avoidance system is to assist or ultimately replace the pilot in order to to minimize the resulting collision risk. Detection is needed in aircraft collision avoidance because of the stochastic nature of the sensor readings, here we use information from video cameras. Conflict is declared if the minimum distance between two aircraft is less than a level. The level is given by the radius of a safety sphere surrounding the aircraft. We use the fact that the probability of conflict, for the process studied here, is identical to the probability for a down-crossing of the surface of the sphere. In general, it is easier to compute the probability of down-crossings compared to extremes. The Monte Carlo method provides a way forward to compute the probability of conflict. However, to provide a computationally tractable solution we approximate the crossing of the safety sphere with the crossing of a circular disc. The approximate method yields a result which is as accurate as the Monte Carlo method but the computational load is decreased significantly.

Populärvetenskaplig sammanfattning

Det främsta syftet med den här avhandlingen är att undersöka och tillhandahålla beräkningseffektiva metoder för estimering och detektering. Fokus är på luftburna applikationer, och vi söker estimerings- och detekteringsmetoder som är noggranna och tillförlitliga samtidigt som dem är effektiva med avseende på beräkningsbelastningen. Framförallt så ska metoderna vara optimerade för terrängbaserad navigering och kollisionundvikning. Estimeringsdelen fokuserar på partikelfiltret och det i allmänhet mycket mer effektiva marginaliserade partikelfiltret. Detekteringsdelen fokuserar på att hitta effektiva metoder för att utvärdera sannolikheten för extremvärden. Det uppnår vi genom att betrakta den i allmänhet mycket lättare uppgiften att beräkna sannolikheten för att passera en viss nivå.

Konceptet för att navigera ett flygplan med hjälp av information från terränghöjden är attraktiv på så sätt att systemet blir oberoende av yttre informationskällor. Typiskt består terrängbaserad navigering av ett tröghetsnavigeringssystem stöttat med positionsskattningar från ett terrängbaserat positioneringssystem. Att integrera ett terrängbaserat positioneringssystem med ett tröghetsnavigeringssystem är utmanande på grund av dess kraftigt olinjära karaktär. Idag är partikelfiltret att betrakta som en accepterad estimeringsmetod för mer eller mindre olinjära system. Åtminstone när kraven på beräkningsbelastning inte är alltför stränga. För många realtidstillämpningar är kraven emellertid sådana att dem förhindrar användandet av partikelfiltret. Vi behöver därför mer beräkningseffektiva metoder för att överbrygga problemet, och det marginaliserade partikelfiltret utgör en möjlig lösning. Principen för det marginaliserade partikelfiltret är att utnyttja linjär och diskret struktur inom det totalt sett olinjära systemet. Dessa strukturer går att utnyttja för effektiv estimering genom att skatta dessa delar optimalt med hjälp av tex kalmanfiltret. Det beräkningskrävande partikelfiltret koncentrerar man sedan till en mindre del av skattningsproblemet.

Konceptet för kollisionsundvikning för flygplan är att stötta och i förlängningen ersätta piloten för att minimera kollisionsrisken. Detektering är nödvändig inom kollisionsundvikning på grund av den stokastiska naturen hos sensormätningar. Här använder vi oss av videokamera. Konflikt uppstår om det minsta avståndet mellan flygplanen passerar under en viss nivå. Nivån ges av radien till en säkerhetssfär som omger flygplanen. Vi använder oss av det faktum att sannolikheten för konflikt, för den process vi studerar här, är identisk med sannolikheten för skärning av sfärens yta. Generellt så är det lättare att beräkna sannolikheten för skärning jämfört med extremvärden. Monte Carlo metoden tillhandahåller en väg framåt för att beräkna sannolikheten för konflikt. För att tillhandahålla en beräkningseffektiv lösning approximerar vi emellertid skärningen av säkerhetssfären med skärningen av en cirkulär skiva. Den approximativa metoden ger ett resultat som är lika noggrannt som Monte Carlo metodens men beräkningsbelastningen sjunker signifikant.

Acknowledgments

First of all I would like to thank my supervisor Prof Fredrik Gustafsson for his support throughout my research. Without his never ending patience reading and commenting my articles this thesis would certainly not look the same. I would like to thank Lic Jan Palmqvist for giving me the chance to combine work at Saab with part time PhD studies during $1999-2002$, and Dr Predrag Pucar for giving me the opportunity to resume my research in 2006. Dr Sören Molander deserves attention for all our discussions on collision avoidance. I am also very grateful to Prof Lennart Ljung for giving me the opportunity to join the control & communication group. Moreover, secretary Ulla Salaneck deserves extra gratitude and attention for handling administrative work with ease.

The combination of working at Saab and doing research at the university have been most stimulating. The combination means that encountered technical problems can get the attention they deserve. Working in a project with a tight time schedule means that there is seldom time to solve problems on a theoretical level. This is not always necessary, but often it is. If these problems are solved only partially they have a tendency of emerging again in the future.

Several people have read different versions of this thesis, Prof Fredrik Gustafsson, Dr David Törnqvist, Dr Sören Molander, Dr Umut Orguner, Bengt-Göran Sundqvist, Zoran Sjanic, Roger Larsson and Fredrik Lindsten. Their valuable comments and suggestions have improved the quality of this thesis significantly. Ass Prof Thomas Schön should be credited for our article on the marginalized particle filter. Dr Gustaf Hendeby deserves appreciation for solving my LaTeX-issues, and Fredrik Lindsten for letting me use his illustrative picture on the sense and avoid concept.

My work has been financed partially by the competence center Information Systems for Industrial Control and Supervision ISIS, the National Aerospace Research Funding NFFP and partially by Saab Aerosystems. All are gratefully acknowledged.

I would also like to take the chance and thank my parents, sister, relatives, friends and colleagues for believing in me and my ambitions. Finally, I would like to dedicate this thesis to Caroline and Elliot for putting up with me during this autumn.

Linköping, December 2008
Per-Johan Nordlund

Contents

I Background **1**

1 Introduction **3**
1.1 Estimation for Aircraft Navigation 3
1.2 Detection for Collision Avoidance 6
1.2.1 Manned Air Traffic 6
1.2.2 Unmanned Aerial Vehicles 7
1.3 Outline 8
1.3.1 Outline of Part I 8
1.3.2 Outline of Part II 8
1.4 Other Publications 11

2 Ef cient State Estimation **13**
2.1 Terrain-Aided Navigation 13
2.2 Recursive Estimation 14
2.2.1 Linear Models 16
2.2.2 Multiple Models 17
2.2.3 Nonlinear Models 19
2.3 Particle Filter 20
2.3.1 Importance Sampling 20
2.3.2 Recursive Importance Sampling 22
2.3.3 Resampling 23
2.3.4 The Algorithm 24
2.3.5 Asymptotic Properties 25
2.4 Marginalized Particle Filter 28
2.4.1 General Description 28

2.4.2 Filter for Terrain-Aided Navigation 30
2.4.3 Asymptotic Properties 31

3 Ef cient Detection 35
3.1 Sense and Avoid 35
3.1.1 Sensor 35
3.1.2 Tracking 36
3.1.3 Near Mid-Air Collision Avoidance 38
3.2 Probability of Near Mid-Air Collision 40
3.3 Detection using hypothesis testing 41
3.4 Probability of level-crossings 44
3.4.1 One dimension and constant velocity 45
3.4.2 One dimension and piecewise constant velocity 47
3.4.3 Two dimensions 48
3.4.4 Three dimensions 50

4 Concluding Remarks 55
4.1 Summary 55
4.2 Future Research 56

A Derivation of $p_\tau(t)$ 57

Bibliography 59

II Publications 67

A Marginalized Particle Filters for Mixed Linear/Nonlinear State-space Models 69
1 Introduction 71
2 Marginalization 73
2.1 The Standard Particle Filter 73
2.2 Diagonal Model 75
2.3 Triangular Model 77
2.4 The General Case 78
3 Important Special Cases and Extensions 81
3.1 Generalized Noise Assumptions 81
3.2 An Important Model Class 82
4 An Illustrating Example 83
5 Integrated Aircraft Navigation 84
5.1 The Dynamic Model 85
5.2 Result 85
6 Conclusions 86
A Appendix 88
A.1 Proof for Theorem 1 88
References 90

B Marginalized Particle Filter for Accurate and Reliable Aircraft Navigation 95
1 Introduction . . . 97
2 Outline . . . 100
3 INS Error Dynamics . . . 100
4 Terrain-Aided Positioning . . . 104
5 The Marginalized Particle Filter . . . 105
6 Blended INS/TAP using MPF . . . 110
6.1 The applied algorithm . . . 110
6.2 Convergence Analysis of Algorithm 1 . . . 113
7 Simulation Results . . . 115
8 Conclusions . . . 118
A Appendix . . . 118
A.1 INS Motion Equations . . . 118
A.2 Discrete-Time Propagation Matrices . . . 121
References . . . 122

C Probabilistic Con ict Detection for Piecewise Straight Paths 129
1 Introduction . . . 131
2 Problem formulation . . . 133
3 Crossing in case of constant velocity . . . 134
3.1 Time-to-go . . . 134
3.2 Conflict probability for a time interval . . . 136
3.3 Monte-Carlo Approximation . . . 138
3.4 Numerical Approximation . . . 139
4 Conflict in case of piecewise constant velocity . . . 141
4.1 Crossing of the line segment . . . 142
4.2 Implementation . . . 144
5 Simulation results . . . 145
6 Conclusions . . . 146
A Appendix . . . 147
A.1 Proof of Lemma 1 . . . 147
A.2 Proof of Theorem 1 . . . 147
A.3 Derivation of (43) . . . 149
References . . . 149

D Probabilistic Near Mid-Air Collision Avoidance 151
1 Introduction . . . 153
2 Problem formulation . . . 155
3 Crossing of the safety zone . . . 156
3.1 Constant velocity . . . 156
3.2 Piecewise constant velocity . . . 158
4 Monte-Carlo Approximation . . . 159
5 Approximate the safety zone with a disc . . . 160
5.1 Constant velocity . . . 160
5.2 Piecewise constant velocity . . . 163
6 Implementation of the disc approximation . . . 165

6.1 Computing $\mathrm{P}\big(s^{(j,m)}(t) < R\big)$ 165
6.2 Computing $\mathrm{P}\big(\mathrm{NMAC}_{(0,T)}\big)$ 168
7 Simulation results 169
8 Conclusions 170
A Appendix 171
A.1 Proof of Theorem 1 171
References 175

Notation

Abbreviations and Acronyms

Abbreviation	Meaning
ATC	Air Traffic Control
CLT	Central Limit Theorem
CPA	Closest Point of Approach
EKF	Extended Kalman Filter
ELOS	Equivalent Level of Safety
GNSS	Global Navigation Satellite System
GPB	Generalized Pseudo Bayesian
GPS	Global Positioning System
ILS	Instrument Landing System
IMM	Interacting Multiple Model
INS	Inertial Navigation System
KF	Kalman Filter
MCMC	Markov Chain Monte Carlo
MPF	Marginalized Particle Filter
MS	Modified Spherical
MSC	Modified Spherical Coordinates
NMAC	Near Mid-Air Collision
PF	Particle Filter
RA	Radar Altimeter
RBPF	Rao-Blackwellized Particle Filter
RMSE	Root Mean Square Error
RNP	Required Navigation Performance
RP	Range Parameterized
RPV	Remotely Piloted Vehicle
RVR	Runway Visual Range

Abbreviation	Meaning
TAN	Terrain-Aided Navigation
TAP	Terrain-Aided Positioning
TCAS	Traffic Alert and Collision Avoidance System
TTG	Time-To-Go
UAV	Unmanned Aerial Vehicle
VFR	Visual Flight Rules

Symbols and Mathematical Notation

Notation	Meaning
$\alpha_{t\ t-k}^{(k)}$	GPB weight for model k at discrete time t based on $\mathbb{Y}_k$.
e_t	Measurement noise vector at discrete time t.
F_t	Discrete time state transition matrix.
$f(\cdot)$	State propagation mapping.
G_t	Discrete time noise to state propagation matrix.
$g(x)$	An arbitrary integrable function of the random variable x.
$\hat{g}_t$	Expected value of $g(x_t)$, $\hat{g}_t = \mathbb{E}_{p(\cdot)}[g(x_t)]$.
$\hat{g}_t^N$	Estimated value on $\hat{g}_t$ using N i.i.d. samples of x_t.
$\hat{g}_{\mathrm{R},t}^N$	Estimated value on $\hat{g}_t$ using N i.i.d. samples of a subset of x_t.
$h(\cdot)$	State to noise-free measurement mapping.
H_t	State to noise-free measurement transition matrix.
K_t	Kalman filter gain matrix.
λ_t	Discrete time Markov chain at time t.
Λ_t	Stacked vector of λ_t, $\Lambda_t = \ \lambda_0, \ldots, \lambda_t\ $.
N	Number of samples, realizations or particles.
N_{eff}, $\hat{N}_{\mathrm{eff}}$	Exact and estimated number of efficient samples.
n_x	Dimension of the vector x or x_t.
$P_{t\ t-k}$, P_x	Covariance of $\hat{x}_{t\ t-k}$ and x respectively.
Q_t	Process noise covariance matrices.
R	Radius of safety circle or sphere.
R_t	Measurement noise covariance matrix.
s, $s(t)$	Relative position.
S_t	Kalman filter residual covariance matrix.
Σ_t, $\Sigma_{\mathrm{R},t}$	Sampling covariances.
τ	Time-to-go.
u_t	Process noise vectors at discrete time t.
v, $v(t)$	Relative velocity.
v	Relative velocity perpendicular to line-of-sight.
$w(\mathbb{X}_t)$	Importance weight.
$w_t^{(i)}$	Importance weight evaluated at $\mathbb{X}_t^{(i)}$.
$\bar{w}_t^{(i)}$	Normalized importance weight, $\ _i\, \bar{w}_t^{(i)} = 1$.
$\dot{x}$	Time derivative of x.

Notation	Meaning
x_t	State vector at discrete time t.
$x(t)$	State vector at continuous time t.
$\mathbb{X}_t$	Stacked vector of states, $\mathbb{X}_t = \quad x_0, \ldots, x_t \quad .$
$x_t^{(i)}$, $\mathbb{X}_t^{(i)}$	A sample/realization of x_t and $\mathbb{X}_t$ respectively.
$\hat{x}_{t\ t-k}$	Least mean square estimate of x_t given the measurements $\mathbb{Y}_k$.
y_t	Measurement vector at discrete time t.
$\mathbb{Y}_t$	Stacked vector of measurements, i.e. $\mathbb{Y}_t = \quad y_0, \ldots, y_t \quad .$
$\mathcal{N}(m, P)$	Gaussian probability density with mean m and covariance P.
χ_l^2	Central χ^2 distribution with l degrees of freedom.
$\chi_l^2(\lambda)$	Non-central χ^2 distribution with l degrees of freedom and non-centrality parameter λ.
$q(x)$	Importance function used for creating samples of a random variable x.
$p(x)$, $p_x(\cdot)$	Probability density function for the random variable x.
$p(x\ y)$, $p_{x\ y}(\cdot)$	Conditional probability density function for x given y.
$\mathrm{P}(\cdot)$, $\mathrm{Pr}(\cdot)$	Probability function.
$\mathbb{E}_{p(x)}[g(x)]$	Expectation of the function $g(x)$ with respect to the probability density $p(x)$.
$\underline{\ }^{d}$	Convergence in distribution.
$\underline{\ }^{a.s.}$	Almost surely convergence.
$\det P$	The determinant of matrix P.
P^T	The transpose of matrix P.
$\delta_{x^{(i)}}(x)$	Dirac delta function with mass located in $x^{(i)}$.
	Proportional relation.
$\sim$	Distributed according to relation.

Part I

Background

1

Introduction

Autonomy is a concept that has received a lot of attention during recent years. To define autonomy is not easy, since it includes a wide spectrum of concepts which differ from case to case. Usually an autonomous vehicle means that there is some degree of self-guidance inherent in the vehicle. This provides the capability to move from one location to another in a more or less predetermined environment. However, the challenge for autonomous vehicles is how to deal with unknown and dynamic environments. This put requirements on the system to be capable of creating situational awareness, and that the vehicle is capable of reacting on unforeseen situations. The main driver for overcoming the technical issues is that it should be possible to significantly cut costs by using autonomous vehicles for monotonous, time-consuming and dangerous missions. Two areas are known as enabling technologies for autonomous vehicles; navigation and collision avoidance. The purpose of navigation is to estimate the own vehicle's kinematic state, e.g. position and velocity. The principle of collision avoidance is to estimate other vehicle's kinematic states and then to find a trajectory which constitutes low risk of collision with respect to the tracked vehicles.

The purpose of this thesis is to introduce the reader to the concepts of estimation and detection in general, and applied to navigation and collision avoidance for airborne systems in particular. Both estimation and detection are about obtaining accurate values of (functions of) quantities based on noisy readings of a surrounding environment. Although the material in this thesis is focused on airborne navigation and collision avoidance the theory should be applicable to many other areas.

1.1 Estimation for Aircraft Navigation

Today the Air Navigation Service (ANS) for civil air traffic still relies on ground based navigation aids, e.g. Non-Directional Beacons (NDB), VHF Omni-directional radio Range (VOR) and Distance Measuring Equipment (DME). Many of existing ground based navi-

gation aids are currently being phased out [108, 15] for cost reasons. The ability to replace them is made possible through the concept of Required Navigation Performance (RNP). RNP authorizes access through requirements on the navigation system performance regardless of hardware. The requirements are among other that the system shall have a certain accuracy and integrity. Integrity is a statistical measure of a navigation system's capability to stay within a containment region, or actually to provide a warning if the aircraft position can not be guaranteed to be within the region [87]. Loss of integrity can either be caused by undetected faults or that the fault-free accuracy is too low [92]. The systems that are currently acting as the primary source of information, and are expected to do so in the foreseeable future, are satellite based system often referred to as Global Navigation Satellite Systems (GNSS). The most known is probably the Global Positioning System (GPS), but there are others e.g. GLONASS and Galileo. A GNSS operates using satellites with known positions, and the satellites transmit signals making it possible for receivers to compute their own positions through triangulation [58].

For phases of flight where severe requirements are put on the navigation system accuracy, e.g. during precision approach and landing, dedicated systems are most often used. The currently most widespread system is the Instrument Landing System (ILS), which guides the pilot to touchdown in limited visibility conditions [58]. ILS landings are usually categorized as Cat I, II, IIIa, IIIb and IIIc, corresponding to successively lower decision heights and shorter runway visual range (RVR) in the order given. Decision height and RVR represent the point from where the pilot must be able to see the runway. Efforts are being made to find complements to ILS, and again GNSS are considered as the primary substitute.

There are however concerns with GNSS related to their reliability or integrity. GNSS is sensitive to disturbances, e.g. athmospheric phenomena and undetected hardware failures. To overcome the integrity issue GNSS must be augmented and monitored. This can be done using different techniques, usually referred to as either ground, aircraft or satellite based augementation systems. Here we will focus on Aircraft Based Augmentation Systems (ABAS), where the principle is to use other onboard navigation sensors. A typical sensor for augmentation with GNSS is the Inertial Navigation System (INS). The INS operates through a set of rather complicated differential equations describing the vehicle's motion. Input to the equations are measured angular rate and acceleration, making INS independent of external sources.

Accurate and reliable navigation systems are becoming even more important due to the introduction of Unmanned Aerial Vehicles (UAVs). Requirements on navigation accuracy and reliability are amplified since the pilot in manned aircraft also function as a monitor of system performance. Requirements are also put on sensor cost and weight, at least for small tactical UAVs. No single, stand-alone sensor is capable of meeting all requirements. The remedy is to adopt the concept of integrated navigation. Integrated navigation means that the output from two or more navigation sensors are blended to obtain better accuracy and robustness than what the individual sensors can achieve. In many cases, the Inertial Navigation System (INS) is typically seen as the primary source of navigation data. One of the reasons is that INS besides position and velocity also provides the vehicle's orientation, which is important for autonomous control. However, its navigation accuracy degrades with time as sensor errors are mathematically integrated through the navigation equations. The standard system for stabilizing the INS drift is currently the

Global Positioning System (GPS) [18, 21, 11]. The GPS signal is however weak making it sensitive to intentional or unintentional disturbances. A possible and interesting supplement to GPS is Terrain-Aided Positioning (TAP), since it can serve as a monitor and back-up system to GPS. The principle of TAP is to use terrain height information to estimate position, see Figure 1.1.

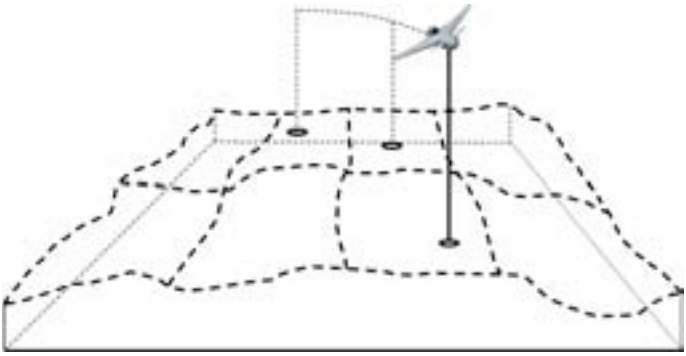

Figure 1.1: *Terrain-aided positioning.*

Here we will study INS integrated with TAP. The challenge with TAP is how to deal with its highly nonlinear, non-analytical nature. The terrain height measurements can either be processed sequentially one by one, which is the method under consideration here, or collected into batches and then processed using profile matching [39]. The particle filter constitutes a generic tool for recursive state estimation of arbitrary systems at the expense of a high computational load. In Paper A the marginalized particle filter is described. It is, when applicable, superior to the standard particle filter. It uses linear or nearly linear substructures which, conditionally upon the nonlinear part, is estimated using linear filters. The remaining nonlinear part is estimated by the particle filter. It is shown that the computational load can be reduced significantly.

There are similar results to be found in the literature on the marginalized particle filter as those presented in this thesis, although obtained independently and of slightly different forms. The most equivalent result can be found in [24], where the aim is also to partition the state vector and apply the particle filter on the truly nonlinear part only. They refer to the resulting filter as the mixture Kalman filter. In [27, 4] similar results are also obtained, although the formulations are rather different. An application is given in [5], where they apply the partitioning technique on amplitude and phase modulated signals. Moreover, similar techniques applied on jump Markov linear systems are given in [28, 32]. For early versions of the result presented here see [79, 78], which are refined in [97].

For integrated INS/TAP we can extend the idea of marginalization to handle a third discrete substructure. In Paper B we provide the details of the filter applied to INS/TAP and show that excellent performance is achieved for a tractable amount of computational load.

1.2 Detection for Collision Avoidance

The purpose of collision avoidance systems is to minimize the risk of collision between vehicles, see Figure 1.2. In an encounter between two manned vehicles, where a collision is imminent, the pilot of each vehicle will normally initiate actions to avoid the collision. There are also procedures to follow and in a near-collision scenario the right-of-way rule applies. In a scenario where one of the vehicles is an UAV the situation is different. If no

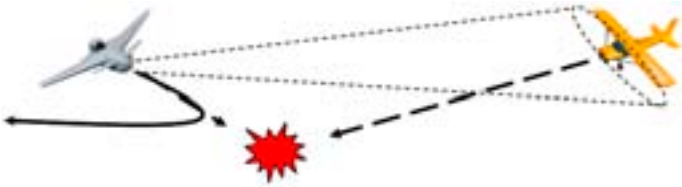

Figure 1.2: *Collision scenario.*

measures are taken there is only one pilot to detect a hazardous situation and react accordingly. To put all responsibility on the pilot in the manned vehicle is deemed unacceptable.

1.2.1 Manned Air Traffic

For manned aircraft today there are different layers of safety, as illustrated by Figure 1.3. The safety measures described below are applicable either partially or as a whole depending on airspace and aircraft. The outer layer consists of procedures. All flights have to comply with rules and procedures as decided by authorities. For example in some cases one needs to file a flight plan, describing among other things your destination, and receive an Air Traffic Control (ATC) clearance. The next layer is separation, a service provided by ATC, where an air traffic controller keeps track of aircraft by surveillance radar and/or transponder. The controller detects aircraft on collision course and informs the pilots how to maneuver in order to avoid the potential conflict. The third layer consists of a cooperative collision aviodance through the use of Traffic Alert and Collision Avoidance System (TCAS) [67]. TCAS automatically detects and evaluates cooperative traffic by interrogating transponders on speed, height and bearing and may advise the pilot to climb or descend to avoid a collision. The inner layer consists of see and avoid, a function which relies on the pilot to see an incoming aircraft and maneuver if necessary. A close encounter or a Near Mid-Air Collision (NMAC) is declared if the minimum distance between two aircraft is less than 150 m [35, 10].

The airspace is partitioned into classes depending on, among other things, altitude [33, 35, 104]. The classification is rather complex and can differ from country to country. One of the most important airspaces for tactical UAVs is class E. Class E includes the major part of the lower airspace up to roughly 5000 m. Within this airspace for aircraft flying by Visual Flight Rules (VFR) [35, 104] radio communication with ATC is not required and separation assistance from ATC is not likely. This implies additional requirements on the capability of see and avoid.

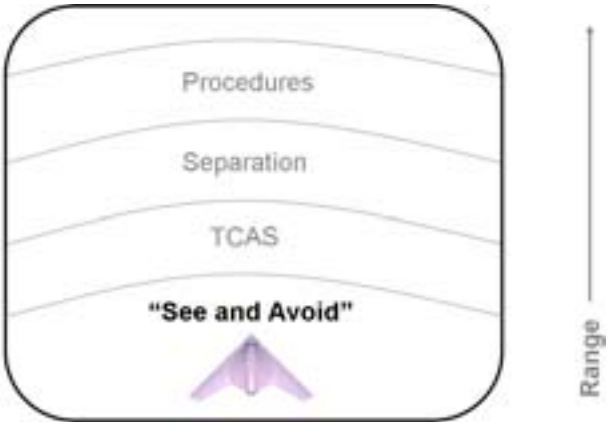

Figure 1.3: *Layers of safety in controlled airspace.*

1.2.2 Unmanned Aerial Vehicles

For the introduction of UAVs into civil airspace the authority requires an Equivalent Level Of Safety (ELOS). This means that the overall safety accomplished for manned aircraft as described above is also required for an unmanned aircraft. This implies that it is not sufficient, for example, to replace the human see and avoid capability with an automatic sense and avoid system. The whole chain of safety measures, from procedures down to see and avoid, needs to be considered to be able to claim ELOS. For example, a seemingly simple task of implementing TCAS on Remotely Piloted Vehicles (RPVs) is in reality not easy [66, 110]. The concerns of implementing TCAS on RPVs are of such a magnitude that the recommendation is *not* to equip with TCAS, at least until the opposite is demonstrated [66]. The concerns are primarily additional delays due to communication links for manual response and criticality of TCAS for autonomous response [110].

There are conceptually two different ways to quantify ELOS. Either we study the human capability and try to mimic it, or we analyze historical statistics on NMACs and empirically determine what a system must achieve. An example on the first method is given in [76], where they present established requirements on parameters such as detection ranges for the US military RPVs Global Hawk and Predator. Examples on the latter method are given by [96] based on 10-year statistics for mid-air collisions in the United States airspace, and [61] based on surveillance radar readings over United States airspace and extracting those encounters which involve VFR aircraft. In [7] a study on TCAS performance is reported, based on an assumed NMAC rate for European airspace. The result of the study indicates a TCAS conflict resolution capability of the original NMACs around 90%, although a great deal new, induced NMACs were also created.

Here we will focus on the inner layer, i.e. to replace the pilot with an automatic sense and avoid in order to avoid NMACs. Although the equivalent level of safety put requirements on a sense and avoid system for unmanned aircraft in any controlled airspace, it is within class E and against aircraft flying VFR we have the most obvious need for sense and avoid. There are many proposals for collision avoidance or conflict resolution, see the survey given by [69]. We are interested in the probability of NMAC for a predicted

trajectory. The most similar result to the ones presented in this thesis is found in the part on short range conflict detection in [91]. The main difference is that in [91] the initial condition is assumed known.

The process for certification of TCAS involved detailed safety studies, numerous encounter simulations and exhaustive field tests. The certification of a sense and avoid system must surely pass a similar process [70]. All actions taken to shorten the process are of course valuable and means that the cost to develop a system is decreased. The development and verification would benefit from a system design which minimize the amount of encounter simulations. For example having a system which per design has the capability of providing correct collision risk measure for a given encounter would be highly beneficial. As will be shown this is one of the main motivators why it is of interest to compute probability of NMAC for a predicted trajectory.

Based on conditions defining a near mid-air collision (NMAC) anytime in the future we can compute the probability of NMAC using the Monte Carlo method. Monte Carlo means that we draw samples from a probability density function defining the initial relative state between the vehicles. The samples are then simulated along the predicted flight trajectory and for each sample the outcome, NMAC or not NMAC, is recorded. The mean of the outcome of the samples constitutes an empirical estimate of the probability of NMAC. The Monte Carlo method is however computer intensive mainly due to the small probabilities that the application is required to detect. Typically the Monte Carlo method is not possible to use for real-time processing. Instead, in Paper C we propose a method for computing collision risk as accurate as the Monte Carlo method but with a significantly reduced computational load. The result is extended to three dimensions in Paper D.

1.3 Outline

1.3.1 Outline of Part I

Part I consists of two main subparts; estimation in Section 2 and detection in Section 3. The estimation part starts in Section 2.1 by providing a more detailed description of terrain-aided navigation. Recursive state estimation is presented in Section 2.2, and for linear, multiple and nonlinear models in particular. Section 2.3 deals with the particle filter including asymptotic properties. Section 2.4 is devoted to the marginalized particle filter. The detection part starts with giving a more detailed description of the sense and avoid system in Section 3.1. Definition of probability of NMAC is given in Section 3.2 and detection principles using hypotheses is provided in Section 3.3. In Section 3.4 we introduce the reader to the concept of using level-crossings as the mean to compute probability of conflict. We start with one dimension, then two dimensions, and finally three dimensions. Part I is concluded in Section 4, including a discussion on possible future research.

1.3.2 Outline of Part II

Part II consists of a collection of papers which constitute the main contributions of the thesis.

Paper A: Marginalized Particle Filters for Mixed Linear/Nonlinear State-space Models

T.B. Schön, F. Gustafsson, and P-J. Nordlund. Marginalized particle filters for mixed linear/nonlinear state-space models. *IEEE Transactions on Signal Processing*, 53(7):2279–2289, July 2005.

Summary: The particle filter offers a general numerical tool to approximate the filtering density function for the state in nonlinear and non-Gaussian filtering problems. While the particle filter is fairly easy to implement and tune, its main drawback is that it is quite computer intensive, with the computational complexity increasing quickly with the state dimension. One remedy to this problem is to marginalize out the states appearing linearly in the dynamics. The result is that one Kalman filter is associated with each particle. The main contribution in this paper is to derive the details for the marginalized particle filter for a general nonlinear state-space model. Several important special cases occurring in typical signal processing applications are also discussed. The marginalized particle filter is applied to a simplified integrated aircraft navigation system assuming a known altitude. It is demonstrated that the complete high-dimensional system can be based on a particle filter using marginalization for all but two states. Excellent performance on real flight data is reported.

Background and contribution: The article is based on the theory given in [78], which provides the derivation of the marginalized particle filter. The main parts of the article is credited to T.B. Schön who has also refined the results.

Paper B: Marginalized Particle Filter for Accurate and Reliable Aircraft Navigation

P-J. Nordlund and F. Gustafsson. Marginalized particle filter for accurate and reliable terrain-aided navigation. *Accepted for publication in IEEE Transactions on Aerospace and Electronic Systems*, 2008. http://www.control.isy.liu.se/research/reports/2008/2870.pdf.

Summary: This paper details an approach to the integration of INS (Inertial Navigation System) and TAP (Terrain-Aided Positioning). The solution is characterized by a joint design of INS and TAP, meaning that the highly nonlinear TAP is not designed separately but jointly with the INS using one and the same filter. The applied filter extends the theory of the MPF (Marginalized Particle Filter) given by [98]. The key idea with MPF is to estimate the nonlinear part using the particle filter and the part which is linear, conditionally upon the nonlinear part, is estimated using the Kalman filter. The extension lies in the possibility to deal with a third multi-modal part, where the discrete mode variable is also estimated jointly with the linear and nonlinear parts. Conditionally upon the mode and the nonlinear part, the resulting subsystem is linear and estimated using the Kalman filter. Given the nonlinear motion equations which the INS uses to compute navigation data, the INS equations must be linearized for the MPF to work. A set of linearized equations is derived and the linearization errors are shown to be insignificant with respect to the final

result. Simulations are performed and the result indicates near-optimal accuracy when compared to the Cramer-Rao lower bound.

Background and contribution: Here the marginalized particle filter is extended to cover systems which, besides linear and nonlinear substructures, also contains a discrete unknown mode. The linearization errors when applying Taylor expansion on the INS navigation equations are shown to be insignificant. The resulting filter is applied on a complete integrated INS/TAP system.

Paper C: Probabilistic Con ict Detection for Piecewise Straight Paths

P-J. Nordlund and F. Gustafsson. Probabilistic conflict detection for piecewise straight paths. *Submitted to Automatica*, 2008.
http://www.control.isy.liu.se/research/reports/2008/2871.pdf.

Summary: We consider probabilistic methods for detecting conflicts as a function of predicted trajectory. A conflict is an event representing collision or imminent collision between vehicles or objects. The computations use state estimate and covariance from a target tracking filter based on sensor readings. Existing work is primarily concerned with risk estimation at a certain time instant, while the focus here is to compute the integrated risk over the critical time horizon. This novel formulation leads to evaluating the probability for level-crossing. The analytic expression involves a multi-dimensional integral which is hardly tractable in practice. Further, a huge number of Monte Carlo simulations would be needed to get sufficient reliability for the small risks that the applications often require. Instead, we propose a sound numerical approximation that leads to evaluating a one-dimensional integral which is suitable for real-time implementations.

Background and contribution: Here we derive an efficient method for computing probability of conflict in two dimensions using theory for level-crossings. The method is derived for not only linear motion but also a relative motion which follows a piecewise straight path.

Paper D: Probabilistic Near Mid-Air Collision Avoidance

P-J. Nordlund and F. Gustafsson. Probabilistic near mid-air collision avoidance. *Submitted to IEEE Transactions on Aerospace and Electronic Systems*, 2008.
http://www.control.isy.liu.se/research/reports/2008/2872.pdf.

Summary: We propose a probabilistic method to compute the near mid-air collision risk as a function of predicted flight trajectory. The computations use state estimate and covariance from a target tracking filter based on angle-only sensors such as digital video cameras. The majority of existing work is focused on risk estimation at a certain time instant. Here we derive an expression for the integrated risk over the critical time horizon. This is possible using probability for level-crossing, and the expression applies to a three-dimensional piecewise straight flight trajectory. The Monte Carlo technique provides a

method to compute the probability, but a huge number of simulations is needed to get sufficient reliability for the small risks that the applications require. Instead we propose a method which through sound geometric and numerical approximations yield a solution suitable for real-time implementations. The algorithm is applied to realistic angle-only tracking data, and shows promising results when compared to the Monte Carlo solution.

Background and contribution: Here we derive the conditions for a near mid-air collision to occur. Through a sound geometrical approximation and the extension of Paper C to piecewise linear motion in three dimensions we derive an efficient method for computing probability of near mid-air collision. The resulting method is applied to realistic angle-only tracking data.

1.4 Other Publications

Publications not included but of related interest are:

T. Hektor, H. Karlsson, and P-J. Nordlund. A marginalized particle filter approach to an integrated INS/TAP system. In *Proceedings of the IEEE/ION Position, Location and Navigation Symposium*, pages 766–770, May 2008.

P-J. Nordlund and F. Gustafsson. The probability of near midair collisions using level-crossings. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 2008.

F. Gustafsson, T.B. Schön, R. Karlsson, and P-J. Nordlund. State-of-the-art for the marginalized particle filter. In *Proceedings of the IEEE Nonlinear Statistical Signal Processing Workshop*, pages 172–174, Sept 2006.

P-J. Nordlund, F. Gunnarsson, and F. Gustafsson. Particle filters for positioning in wireless networks. In *Procedings of EUSIPCO*, Toulouse, France, September 2002.

F. Gustafsson, F. Gunnarsson, N. Bergman, U. Forssell, R. Karlsson, and P-J. Nordlund. Particle filters for positioning, navigation and tracking. *IEEE Transactions on signal processing*, 50(2):425–437, Feb 2002.

P-J. Nordlund and F. Gustafsson. Recursive estimation of three-dimensional aircraft position using terrain-aided positioning. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Proceesing*, volume 2, pages 1121–1124, 2002.

F. Gustafsson, F. Gunnarsson, N. Bergman, U. Forssell, R. Karlsson, and P-J. Nordlund. A framework for particle filtering in positioning, navigation and tracking problems. In *Proceedings of the 11th IEEE Signal Processing Workshop on Statistical Signal Processing*, pages 34–37, Aug 2001.

P-J. Nordlund and F. Gustafsson. Sequential Monte Carlo filtering techniques applied to integrated navigation systems. In *Proceedings of the 2001 American Control Conference*, volume 6, pages 4375–4380, 2001.

2

Ef cient State Estimation

2.1 Terrain-Aided Navigation

The purpose of Terrain-Aided Navigation (TAN), as for any other navigation system, is to provide an accurate and reliable estimate of the kinematic state of the own platform, where the state typically consists of position and its derivatives. The navigation solution from TAN is based on sensors which do not rely on external information sources. This makes TAN resistant to disturbances and jamming. The principle for TAN is to integrate information from a Inertial Navigation System (INS) and a Terrain-Aided Positioning (TAP) system, see Figure 2.1. The TAP system is basically a Radar Altimeter (RA) measuring distance to ground and a database with stored terrain height. The RA measurement subtracted from INS altitude provides a terrain height measurement. The database gives terrain height for sampled horizontal positions, ranging from a couple of meters to several hundred depending on the database. In the case studied here terrain height is given at every 50 meter. Terrain height measurements are matched with stored terrain height. The points where stored height matches measured height yield aircraft position candidates. Gradually the measurements form a terrain height profile with fewer and fewer matches in the database. Thereby the position candidates become fewer and fewer until there is only one left, see Figure 2.2 for an illustration. The idea of using terrain as a navigation aid is not new, see [19] for a interesting historical survey on terrain navigation systems.

The task of navigating using terrain height can be cast as recursive state estimation problem. Let the navigation quantities, such as position, velocity and orientation, be comprised in the state vector x_t. The expression for terrain height provided by the terrain information system is

$$h(x_t^{\mathrm{pos}}), \tag{2.1}$$

where $h(x_t^{\mathrm{pos}})$ is the terrain height given by the database as a function of horizontal position. Note that no analytical expression exists for $h(\cdot)$. Also, usually several different

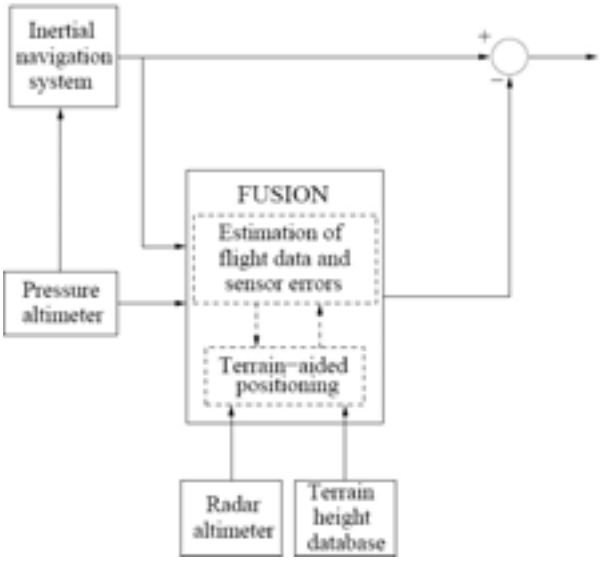

Figure 2.1: *Concept of integrated INS/TAP.*

positions yield the same height making the problem multimodal. This makes the particle filter ideal for this application. The inertial navigation system computes the state x_t from measured acceleration and angular rate u_t. The computations are based on a set of rather complicated nonlinear differential equations, in discrete time given by $f(x_t, u_t)$. However, in paper B we show that the nonlinearities are weak. The expression for time update of INS data then become

$$x_{t+1} = f(x_t, u_t) \approx F_t x_t + G_t u_t. \tag{2.2}$$

By splitting the state vector

$$x_t = \begin{bmatrix} x_t^{\text{pos}} \\ x_t^{\text{nav}} \end{bmatrix}, \tag{2.3}$$

the prerequisites for applying the marginalized particle filter are at place.

2.2 Recursive Estimation

We consider discrete-time state space descriptions with additive noise

$$\begin{aligned} x_{t+1} &= f(x_t) + G_t(x_t)u_t, \\ y_t &= h(x_t) + e_t, \end{aligned} \tag{2.4}$$

where $x_t \quad \mathbb{R}^{n_x}$ represents the unknown state vector and $y_t \quad \mathbb{R}^{n_y}$ is the observation. The subscript t denotes a discrete-time index assuming a sampling time T sec. The process noise u_t and measurement noise e_t are stochastic processes with known probability

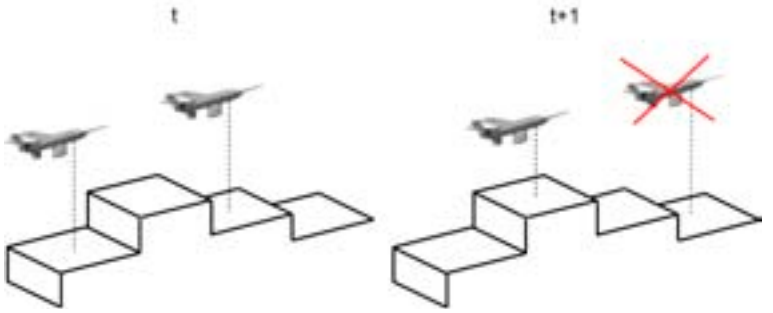

Figure 2.2: *Principle of terrain-aided positioning, with two position candidates at time t but only one at time t+1.*

densities $p(u_t)$ and $p(e_t)$. The noise sequences are both assumed white and independent of each other, i.e.

$$\begin{aligned} p(u_{t+k}, u_t) &= p(u_{t+k})p(u_t), \qquad k = 0, \\ p(e_{t+l}, e_t) &= p(e_{t+l})p(e_t), \qquad l = 0, \\ p(u_t, e_t) &= p(u_t)p(e_t). \end{aligned} \tag{2.5}$$

We denote by $\mathbb{X}_t = \quad x_0, \ldots, x_t \quad$ and $\mathbb{Y}_t = \quad y_0, \ldots, y_t \quad$ the stacked vector of all the states and measurements up to time t. By the assumptions on the noise according to (2.5) and the definition of conditional density [44]

$$p(x \; y) = \frac{p(x, y)}{p(y)} \tag{2.6}$$

one can show that x_t is a Markov process

$$p(\mathbb{X}_t) = \prod_{k=0}^{t} p(x_k \; x_{k-1}), \tag{2.7}$$

where $p(x_0 \; x_{-1}) = p(x_0)$, and that the conditional density of the measurements given the states are independent

$$p(\mathbb{Y}_t \; \mathbb{X}_t) = \prod_{k=0}^{t} p(y_k \; x_k) = \prod_{k=0}^{t} p_{e_t}(y_k - x_k). \tag{2.8}$$

The goal is to compute or estimate the *posterior* probability density function $p(x_t \; \mathbb{Y}_t)$. When a new measurement is available we want our estimate to be updated using the new information. To avoid having to re-calculate everything we need recursive expressions for how the new information should be incorporated. Using Bayes' formula [52] and the assumptions (2.7)–(2.8) we can derive the following recursions for the posterior probability

$$p(x_t \; \mathbb{Y}_t) = \frac{p(y_t \; x_t)p(x_t \; \mathbb{Y}_{t-1})}{p(y_t \; \mathbb{Y}_{t-1})}, \tag{2.9a}$$

$$p(y_t \; \mathbb{Y}_{t-1}) = \int\limits_{\mathbb{R}^{n_x}} p(y_t \; x_t)p(x_t \; \mathbb{Y}_{t-1})dx_t, \tag{2.9b}$$

referred to as the *measurement update*. The *time update* is provided by the law of total probability

$$p(x_{t+1} \; \mathbb{Y}_t) = \int\limits_{\mathbb{R}^{n_x}} p(x_{t+1} \; x_t)p(x_t \; \mathbb{Y}_t)dx_t. \tag{2.10}$$

Based on the posterior probability we can compute the point estimate $\hat{x}_{t \; t}$ which minimizes the expectation of the estimation error squared, i.e.

$$\begin{aligned} \hat{x}_{t \; t} &= \arg\min_{\hat{x}_t} \mathbb{E}_{p(x_t \; \mathbb{Y}_t)}\big[(x_t - \hat{x}_t)(x_t - \hat{x}_t)^T\big] \\ &= \arg\min_{\hat{x}_t} \int\limits_{\mathbb{R}^{n_x}} (x_t - \hat{x}_t)(x_t - \hat{x}_t)^T p(x_t \; \mathbb{Y}_t)dx_t. \end{aligned} \tag{2.11}$$

The estimate $\hat{x}_{t \; t}$ is given by Theorem 2.1.

Theorem 2.1 (Least-mean-square estimation)
Given two random variables (scalar- or vector-valued) x_t and $\mathbb{Y}_t$, the optimal least-mean-square estimate of x_t given $\mathbb{Y}_t$ is

$$\hat{x}_{t \; t} = \mathbb{E}_{p(x_t \; \mathbb{Y}_t)}[x_t]. \tag{2.12}$$

Proof: See [55]. □

2.2.1 Linear Models

For linear estimation we have the Kalman filter, thoroughly described in e.g. [55]. Consider the linear state-space model

$$\begin{aligned} x_{t+1} &= F_t x_t + G_t u_t, \\ y_t &= H_t x_t + e_t, \end{aligned} \tag{2.13}$$

where

$$\begin{aligned} &\mathbb{E}\begin{bmatrix} u_t \\ e_t \\ x_0 \end{bmatrix} \begin{bmatrix} u_{t+k}^T & e_{t+k}^T & x_0^T & 1 \end{bmatrix} = \begin{bmatrix} Q_t\delta_k & 0 & 0 & 0 \\ 0 & R_t\delta_k & 0 & 0 \\ 0 & 0 & P_0 & \hat{x}_0 \end{bmatrix}, \\ &\delta_k = \begin{cases} 1 & \text{if } k = 0 \\ 0 & \text{otherwise.} \end{cases} \end{aligned} \tag{2.14}$$

With Gaussian distributed u_t, e_t and x_0, i.e.

$$u_t \sim \mathcal{N}(0, Q_t), \quad e_t \sim \mathcal{N}(0, R_t), \quad x_0 \sim \mathcal{N}(\hat{x}_0, P_0), \tag{2.15}$$

the optimal filter is the Kalman filter, see Theorem 2.2.

Theorem 2.2 (The Kalman lter)
Consider the linear state-space model according to (2.13)*. Moreover, assume that the noise processes have the properties according to* (2.14)–(2.15) *Then* x_t *and* x_{t+1}*, conditionally upon* $\mathbb{Y}_t$*, are Gaussian distributed for any time* $t \geq 0$*, i.e.*

$$\begin{aligned} p(x_t \ \mathbb{Y}_t) &= \mathcal{N}(\hat{x}_{t\ t}, P_{t\ t}), \\ p(x_{t+1} \ \mathbb{Y}_t) &= \mathcal{N}(\hat{x}_{t+1\ t}, P_{t+1\ t}), \end{aligned} \tag{2.16}$$

where the mean and covariance propagate through the measurement update

$$\begin{aligned} \hat{x}_{t\ t} &= \hat{x}_{t\ t-1} + P_{t\ t-1} H_t^T S_t^{-1} (y_t - H_t \hat{x}_{t\ t-1}), \\ P_{t\ t} &= P_{t\ t-1} - P_{t\ t-1} H_t^T S_t^{-1} H_t P_{t\ t-1}, \\ S_t &= R_t + H_t P_{t\ t-1} H_t^T, \end{aligned} \tag{2.17}$$

and the time update

$$\begin{aligned} \hat{x}_{t+1\ t} &= F_t \hat{x}_{t\ t}, \\ P_{t+1\ t} &= F_t P_{t\ t} F_t^T + G_t Q_t G_t^T, \end{aligned} \tag{2.18}$$

with initial values $\hat{x}_{0\ -1} = \hat{x}_0$ *and* $P_{0\ -1} = P_0$*.*

Proof: See [40, 3, 55]. □

Note that for non-Gaussian distributed u_t, e_t and x_0 the best *linear* least-mean-square solution is still provided by the Kalman filter [55].

2.2.2 Multiple Models

In the multiple model approach we extend the linear model to cover the case with multiple linear models. The multitude is handled by a random variable λ_t which can take a finite number of discrete values $1, 2, \ldots, M$. The model under consideration is

$$\begin{aligned} x_{t+1} &= F_t(\lambda_{t+1}) x_t + G_t(\lambda_{t+1}) u_t(\lambda_{t+1}), \\ y_t &= H_t(\lambda_t) x_t + e_t(\lambda_t), \end{aligned} \tag{2.19}$$

and we assume λ_t follows a discrete Markov process

$$\Pr(\Lambda_t) = \prod_{k=0}^{t} \Pr(\lambda_k \ \lambda_{k-1}), \tag{2.20}$$

where $\Pr(\lambda_0 \ \lambda_{-1}) = \Pr(\lambda_0)$. The mode transition probabilities $\Pr(\lambda_t \ \lambda_{t-1})$ are given by the matrix

$$\begin{matrix} \Pr(1\ 1) & \ldots & \Pr(1\ M) \\ \vdots & & \vdots \\ \Pr(M\ 1) & \ldots & \Pr(M\ M) \end{matrix} . \tag{2.21}$$

From (2.19) we see that if we know the mode sequence, $\Lambda_t = \quad \lambda_0, \ldots, \lambda_t$, we are back to the original linear model according to (2.13), with the solution given by the Kalman filter. In practice, we usually do not have access to the mode sequence and we are confined to use the probability of each possible sequence $\Pr(\Lambda_t \; \mathbb{Y}_t)$. Using the law of total probability we can write the filtering posterior probability density as

$$p(x_t \; \mathbb{Y}_t) = \quad p(x_t \; \mathbb{Y}_t, \Lambda_t)\Pr(\Lambda_t \; \mathbb{Y}_t). \tag{2.22}$$

The number of possible mode sequences grows exponentially with time, there are M^{t+1} different possibilities to account for at time t, starting from time $t = 0$.

One way to limit the number of mode sequences is to only account for the most recent time instances, say the last L. This means that we only need to account for M^L different hypotheses, and it enables us to keep the computational load constant over time. At each update of the filters we first split each mode sequence into M new ones. Each mode sequence is now $L + 1$ time instances long. We then apply a Kalman filter on each extended mode sequence. Next we merge M sequences which are now equal when considering only the L most recent time instances. This is the principle behind the *Generalized Pseudo-Bayesian* (GPB) filter, see Algorithm 2.1 for $L = 1$. The measurement update of the weight $\alpha_{t\;t}^{(\lambda_t)} = \Pr(\lambda_t \; \mathbb{Y}_t)$ is given by

$$\begin{aligned} &\Pr(\lambda_t \; \mathbb{Y}_t) \quad \Pr(\lambda_t \; \mathbb{Y}_{t-1}) \quad p(y_t \; \lambda_t, x_t)p(x_t \; \lambda_t, \mathbb{Y}_{t-1})dx_t \\ &= e^{-\frac{1}{2}(y_t - H_t(\lambda_t)\hat{x}_{t|t-1}^{(\lambda_t)})^T S_t^{-1}(\lambda_t)(y_t - H_t(\lambda_t)\hat{x}_{t|t-1}^{(\lambda_t)})}\Pr(\lambda_t \; \mathbb{Y}_{t-1}). \end{aligned} \tag{2.23}$$

Algorithm 2.1 (The GPB1 filter).

1. For $j = 1, \ldots, M$, initialize the weights $\bar{\alpha}_0^{(\lambda_0)} = \Pr(\lambda_0)$ and the estimates $\hat{x}_0^{(\lambda_0)}$, $P_0^{(\lambda_0)}$.

2. Estimate measurement update: Compute $\hat{x}_{t\;t}^{(\lambda_t)}$, $P_{t\;t}^{(\lambda_t)}$ using (2.17) conditionally upon λ_t.

3. Weight measurement update: Compute new weights using

$$\begin{aligned} \alpha_{t\;t}^{(\lambda_t)} &= e^{-\frac{1}{2}(y_t - H_t(\lambda_t)\hat{x}_{t|t-1}^{(\lambda_t)})^T S_t^{-1}(\lambda_t)(y_t - H_t(\lambda_t)\hat{x}_{t|t-1}^{(\lambda_t)})}\bar{\alpha}_{t\;t-1}^{(\lambda_t)}, \\ \bar{\alpha}_{t\;t}^{(\lambda_t)} &= \frac{\alpha_{t\;t}^{(\lambda_t)}}{\sum_{\lambda_t=1}^{M} \alpha_{t\;t}^{(\lambda_t)}} \end{aligned}$$

4. Merge:

$$\begin{aligned} \hat{x}_{t\;t} &= \sum_{\lambda_t=1}^{M} \bar{\alpha}_{t\;t}^{(\lambda_t)} \hat{x}_{t\;t}^{(\lambda_t)}, \\ P_{t\;t} &= \sum_{\lambda_t=1}^{M} \bar{\alpha}_{t\;t}^{(\lambda_t)} (P_{t\;t}^{(\lambda_t)} + (\hat{x}_{t\;t}^{(\lambda_t)} - \hat{x}_{t\;t})(\hat{x}_{t\;t}^{(\lambda_t)} - \hat{x}_{t\;t})^T). \end{aligned}$$

5. Estimate time update: Compute $\hat{x}_{t+1\ t}^{(\lambda_{t+1})}$, $P_{t+1\ t}^{(\lambda_{t+1})}$ using (2.18) conditionally upon λ_{t+1}.

6. Weight time update: Compute new weights using

$$\bar{\alpha}_{t+1\ t}^{(\lambda_{t+1})} = \sum_{\lambda_t=1}^{M} \Pr(\lambda_{t+1}\ \lambda_t)\bar{\alpha}_{t\ t}^{(\lambda_t)}$$

7. Iterate from step 2.

The conceptually similar *Interacting Multiple Model* (IMM) filter uses a slighly more sophisticated way of computing the weights [16, 12, 56]. See [106] for a nice survey and comparison of different multiple model approaches.

2.2.3 Nonlinear Models

When we are facing a nonlinear estimation problem given by (2.4) we can still use the Kalman filter recursions given by (2.17)–(2.17). The difference is that we use a set of linearized equations given by the first order terms in the Taylor expansion around the current estimate

$$H_t = \left.\frac{\partial h(x_t)}{\partial x_t}\right|_{x_t=\hat{x}_{t|t-1}}, \quad F_t = \left.\frac{\partial f(x_t)}{\partial x_t}\right|_{x_t=\hat{x}_{t|t}}. \tag{2.24}$$

The result is referred to as the extended Kalman filter (EKF) [55, 3, 93]. More sophisticated versions of the EKF exist which account for the second order terms in the Taylor expansion [49].

The Gaussian sum filter [102, 2, 3] can be seen as an extension to the EKF. The idea is to split the estimation using several EKFs working in parallel, where each EKF is concentrated on a subspace of the overall posterior probability density. The overall posterior probability is approximately given by a mixture of Gaussians

$$p(x_t\ Y_t) \approx \sum_{i=1}^{N} \bar{w}_t^{(i)} \mathcal{N}(\hat{x}_{t\ t}^{(i)}, P_{t\ t}^{(i)}), \qquad \sum_{i=1}^{N} \bar{w}_t^{(i)} = 1, \tag{2.25}$$

where the mean and covariance of each Gaussian is provided by a dedicated EKF. Typically the splitting is based on the initial values $\hat{x}_0$ and P_0 such that the covariance for each EKF $P_0^{(i)}$ is small enough. Small enough here means that the linearized equations should provide an accurate description around $\hat{x}_{t\ t}^{(i)}$ as long as we do not deviate more than given by $P_0^{(i)}$. For an example on a Gaussian sum filter application see Example 2.1.

Example 2.1

The Range Parameterized EKF (RP-EKF) is a method for tracking other vehicles based on angle-only measurements. Tracking with angle-only measurements normally results in highly uncertain range estimates. The principle for RP-EKF is to split the applicable range into subintervals, each having the same so-called coefficient of variation, C_r. The

C_r is defined as the ratio between standard deviation and mean of each range segment. The performance of angle-only tracking is highly dependent on C_r, the lower the better [88]. A large uncertainty in range compared to the mean yields a high C_r. Suppose the range is uniformly distributed over the interval $(r_{\min}, r_{\max})$, which yields

$$C_r = \frac{\sigma_r}{\hat{r}} = \frac{\frac{r_{\max}-r_{\min}}{\sqrt{12}}}{\frac{r_{\max}+r_{\min}}{2}} \tag{2.26}$$

We can achieve a smaller C_r by splitting the range interval into a number N_f of subintervals, where subinterval i is defined by $(r_{\min}\rho^{i-1}, r_{\min}\rho^{i})$ and $\rho = \left(r_{\max}/r_{\min}\right)^{1/N_f}$. Now it is straightforward to show that $C_r^{(i)}$ is the same for each segment and is given by [9]

$$C_r^{(i)} = \frac{\sigma_r^{(i)}}{\hat{r}^{(i)}} = \frac{2(\rho-1)}{\sqrt{12}(\rho+1)}. \tag{2.27}$$

The principle of the point-mass filter [64, 65, 13] is to discretize the support of the posterior density into a finite set of points. The expression for the posterior probability becomes

$$p(x_t \; Y_t) \approx \sum_{i=1}^{N} \bar{w}_t^{(i)} \delta_{x_t^{(i)}}(x_t), \tag{2.28}$$

where $\delta(\cdot)$ is the Dirac delta function.

2.3 Particle Filter

Particle filters, or sequential Monte Carlo methods [31, 38], can be seen as simulation-based methods, i.e. we simulate a large number of possible state trajectories using e.g. the time propagation equations from (2.4). Based on the measurements we associate a weight to each trajectory, using the measurement equation from (2.4). The realizations together with the associated weigths provide an empirical approximation of the joint posterior distribution. Probably two of the first references on particle filtering are [45] and [46]. In these papers importance sampling, described in Section 2.3.1 and 2.3.2, was introduced on sequential estimation problems. The crucial step of resampling, described in Section 2.3.3, was however first introduced in [38]. The two ingredients, importance sampling and resampling, form the basis for the particle filter algorithm given in Section 2.3.4.

2.3.1 Importance Sampling

Suppose that we have drawn N *independent identically distributed* (i.i.d.) samples of x_t from the probability density $q(x_t \; y_t)$. In the literature, the function $q(x_t \; y_t)$ is usually

referred to as the *importance function*, see e.g. [30]. The first requirement on this distribution is that it should be possible to draw samples from it. See e.g. [53] for details on how to create samples from a distribution. We can use the set of samples to form an empirical approximation of $q(x_t \; y_t)$, i.e.

$$q(x_t \; y_t) \approx \frac{1}{N} \sum_{i=1}^{N} \delta_{x_t^{(i)}}(x_t), \tag{2.29}$$

where $\delta(\cdot)$ is the Dirac delta function. What (2.29) really means is that the probability mass for a small volume is approximated by the relative frequency of samples that are located within the volume. A more correct notation would therefore be

$$q(dx_t \; y_t) \approx \frac{1}{N} \sum_{i=1}^{N} \delta_{x_t^{(i)}}(dx_t), \tag{2.30}$$

but this is not pursued further. What we really would like is to be able to draw realizations from $p(x_t \; y_t)$, referred to as *perfect sampling*, but unfortunately this is not possible in general. The best we can hope for is that $q(x_t \; y_t)$ is as close to $p(x_t \; y_t)$ as possible. Let us introduce the importance weight defined as

$$w(x_t) = \frac{p(x_t \; y_t)}{q(x_t \; y_t)}, \tag{2.31}$$

or in the Bayesian framework, where the normalization constant $p(y_t)$ in general is analytically intractable,

$$w(x_t) = \frac{p(y_t \; x_t)p(x_t)}{q(x_t \; y_t)}. \tag{2.32}$$

See Example 2.2 for the most basic choice of importance function.

Example 2.2

A natural choice of importance function is to use the prior probability density for the state vector

$$q(x_t \; y_t) = p(x_t). \tag{2.33}$$

The weights are then computed according to

$$w(x_t) = \frac{p(y_t \; x_t)p(x_t)}{p(x_t)} = p(y_t \; x_t). \tag{2.34}$$

The importance weight, $w(x_t)$, can be looked upon as a measure of the skewness of $q(x_t \; y_t)$ relative to $p(x_t \; x_t)$. From (2.31) or (2.32) we see that for the importance weight to be defined the support of $p(x_t \; y_t)$ must be included in the support of $q(x_t \; y_t)$. Based

on the realizations from (2.29) and the importance weights computed using (2.32) we can express an empirical approximation of the posterior density $p(x_t \; y_t)$ according to[1]

$$\begin{aligned} p(x_t \; y_t) &= \frac{w(x_t)q(x_t \; y_t)}{w(x_t)q(x_t \; y_t)dx_t} \\ &\approx \frac{\sum_{i=1}^{N} w_t^{(i)} \delta_{x_t^{(i)}}(x_t)}{\sum_{i=1}^{N} w_t^{(i)}} = \sum_{i=1}^{N} \bar{w}_t^{(i)} \delta_{x_t^{(i)}}(x_t), \end{aligned} \tag{2.35}$$

where we have defined

$$w_t^{(i)} = \frac{p(y_t \; x_t^{(i)})p(x_t^{(i)})}{q(x_t^{(i)} \; y_t)}, \quad \bar{w}_t^{(i)} = \frac{w_t^{(i)}}{\sum_{j=1}^{N} w_t^{(j)}}. \tag{2.36}$$

Using (2.35) we can now perform any kind of operation that previously was intractable, e.g. computing any kind of estimate. For any integrable function $g(x_t)$ we can estimate its expected value according to

$$\begin{aligned} \hat{g}_t = \mathbb{E}_{p(x_t \; y_t)}[g(x_t)] &= \int g(x_t)p(x_t \; y_t)dx_t \\ &\approx \int g(x_t) \sum_{i=1}^{N} \bar{w}_t^{(i)} \delta_{x_t^{(i)}}(x_t)dx_t = \sum_{i=1}^{N} \bar{w}_t^{(i)} g(x_t^{(i)}) = \hat{g}_t^N. \end{aligned} \tag{2.37}$$

Using (2.37) we see that the least-mean-square estimate $\hat{x}_{t \; t}$ and its covariance are estimated using $g(x_t) = x_t$ and $g(x_t) = (x_t - \hat{x}_{t \; t})(x_t - \hat{x}_{t \; t})^T$ respectively, i.e.

$$\hat{x}_{t \; t} \approx \sum_{i=1}^{N} \bar{w}_t^{(i)} x_t^{(i)}, \tag{2.38a}$$

$$P_{t \; t} \approx \sum_{i=1}^{N} \bar{w}_t^{(i)} (x_t^{(i)} - \hat{x}_{t \; t})(x_t^{(i)} - \hat{x}_{t \; t})^T. \tag{2.38b}$$

2.3.2 Recursive Importance Sampling

The question now is how to recursively obtain a set of realizations

$$\mathbb{X}_t^{(i)} \; {}_{i=1}^{N} = \; x_0^{(i)}, \ldots, x_t^{(i)} \; {}_{i=1}^{N} \tag{2.39}$$

which together with the weights $\bar{w}_t^{(i)} \; {}_{i=1}^{N}$ are distributed approximately according to $p(\mathbb{X}_t \; \mathbb{Y}_t)$. The new set of realizations and weights should be based upon the realizations $\mathbb{X}_{t-1}^{(i)} \; {}_{i=1}^{N}$ and importance weights $\bar{w}_{t-1}^{(i)} \; {}_{i=1}^{N}$.

Suppose the realizations $\mathbb{X}_{t-1}^{(i)} \; {}_{i=1}^{N}$ have been drawn from some arbitrary but known density $q(\mathbb{X}_{t-1} \; \mathbb{Y}_{t-1})$, i.e.

$$q(\mathbb{X}_{t-1} \; \mathbb{Y}_{t-1}) \approx \frac{1}{N} \sum_{i=1}^{N} \delta_{\mathbb{X}_{t-1}^{(i)}}(\mathbb{X}_{t-1}) = \frac{1}{N} \sum_{i=1}^{N} \delta_{x_0^{(i)}}(x_0) \cdots \delta_{x_{t-1}^{(i)}}(x_{t-1}), \tag{2.40}$$

[1] Note that we will in the sequel not explicitly state the space on which the integration is taken place, here $(\mathbb{R}^{n_x})^{(t+1)}$, assuming it is clear from the context.

which together with the importance weights are such that

$$p(\mathbb{X}_{t-1}\ \mathbb{Y}_{t-1}) \approx \sum_{i=1}^{N} \bar{w}_{t-1}^{(i)} \delta_{\mathbb{X}_{t-1}^{(i)}}(\mathbb{X}_{t-1}). \tag{2.41}$$

We now wish to extend the existing realizations and modify the existing weights, such that we obtain a set of realizations which, together with the modified weights, is approximately distributed according to $p(\mathbb{X}_t\ \mathbb{Y}_t)$. The importance function can be rewritten recursively, using Bayes' rule, according to

$$q(\mathbb{X}_t\ \mathbb{Y}_t) = q(x_t\ \mathbb{X}_{t-1}, \mathbb{Y}_t) q(\mathbb{X}_{t-1}\ \mathbb{Y}_t). \tag{2.42}$$

In order to be a practical method the new measurement y_t should not affect the existing realizations $\mathbb{X}_{t-1}^{(i)}\ {}_{i=1}^{N}$. Otherwise, we would be forced to create new realizations for the state history at each time t, which would imply a computational load which grows as $\mathcal{O}(t)$. Therefore, we require that

$$q(\mathbb{X}_{t-1}\ \mathbb{Y}_t) = q(\mathbb{X}_{t-1}\ \mathbb{Y}_{t-1}). \tag{2.43}$$

Inserting (2.40) into (2.42) we obtain an expression for how to produce new samples for x_t

$$q(x_t\ \mathbb{Y}_t) = \quad q(\mathbb{X}_t\ \mathbb{Y}_t) d\mathbb{X}_{t-1} \approx \sum_{i=1}^{N} q(x_t\ \mathbb{X}_{t-1}^{(i)}, \mathbb{Y}_t). \tag{2.44}$$

To create a new set of samples $x_t^{(i)}\ {}_{i=1}^{N}$ based on (2.44) we draw N times from the importance function according to

$$x_t^{(i)} \sim q(x_t\ \mathbb{X}_{t-1}^{(i)}, \mathbb{Y}_t), \quad i = 1, \ldots, N. \tag{2.45}$$

The last step is to obtain a properly weighted set of realizations with respect to $p(\mathbb{X}_t\ \mathbb{Y}_t)$. What we need is an expression for how the new weight $w(\mathbb{X}_t)$ is computed recursively in terms of $w(\mathbb{X}_{t-1})$. We have

$$w(\mathbb{X}_t) = \frac{p(\mathbb{Y}_t\ \mathbb{X}_t) p(\mathbb{X}_t)}{q(\mathbb{X}_t\ \mathbb{Y}_t)} = \frac{p(y_t\ x_t) p(x_t\ x_{t-1})}{q(x_t\ \mathbb{X}_{t-1}, \mathbb{Y}_t)} w(\mathbb{X}_{t-1}). \tag{2.46}$$

2.3.3 Resampling

If the weights $w_t^{(i)}\ {}_{i=1}^{N}$ are very skewed, i.e. most of the weights are close to zero and only a few of them are significant, most of the samples do not contribute to the approximation. To create a set of samples with equal weights we can perform a so-called resampling [101]. This means that we multiply those samples with a large importance weight, and samples with small weights are discarded. Call the number of offsprings from each sample $N_t^{(i)}$. To meet the requirement that the unweighted set of samples is still approximately distributed according to $p(\mathbb{X}_t\ \mathbb{Y}_t)$ we need

$$\mathbb{E}[N_t^{(i)}] = N \bar{w}_t^{(i)}. \tag{2.47}$$

Note however that one could be deceived to think that resampling achieves the goal of having a set of samples perfectly matched to $p(\mathbb{X}_t \; \mathbb{Y}_t)$. But the resampling introduces another problem, we obtain a set of *dependent* samples. However, to make the sequential importance sampling technique operational the resampling procedure is necessary.

The standard resampling method is the *multinomial resampling*, originally introduced in [38]. The principle is to pick a realization from $\mathbb{X}_t^{(i)} \; {}_{i=1}^{N}$, with replacement, where the probability to pick $\mathbb{X}_t^{(i)}$ is equal to $\bar{w}_t^{(i)}$. This procedure can be achieved in $\mathcal{O}(N)$ operations, see e.g. [20] for the details.

A second method known to introduce less additional variance than multinomial resampling, is *residual resampling* [74]. The principle behind residual resampling is to first multiply/discard particles deterministically according to $N\bar{w}_t^{(i)}$, where the notation x corresponds to the integer part of x. The second step is to perform multinomial resampling on the rest, i.e. on

$$N_t^{\text{rest}} = N - {}_{i=1}^{N} \; N\bar{w}_t^{(i)} \; . \tag{2.48}$$

A third strategy is to multiply and discard samples entirely deterministically, a *deterministic resampling* [60]. This can be achieved by, for each $i = 1, \ldots, N$, finding the largest integer $N_t^{(i)}$ such that

$$\frac{N_t^{(i)} - \Delta}{N} < \bar{w}_t^{(i)}, \quad 0 \leq \Delta < 1. \tag{2.49}$$

2.3.4 The Algorithm

For a summary of the particle filter see Algorithm 2.2.

Algorithm 2.2 (The particle filter).

1. For $i = 1, \ldots, N$, sample $x_0^{(i)} \sim p(x_0)$ and set $w_{-1}^{(i)} = \frac{1}{N}$
2. For each $i = 1, \ldots, N$, update $w_t^{(i)} = \frac{p(y_t \; x_t^{(i)}) p(x_t^{(i)} \; x_{t-1}^{(i)})}{q(x_t^{(i)} \; \mathbb{X}_{t-1}^{(i)}, \mathbb{Y}_t)} \bar{w}_{t-1}^{(i)}$ and normalize $\bar{w}_t^{(i)} = \frac{w_t^{(i)}}{\sum_j w_t^{(j)}}$.
3. If resampling (e.g. if $\hat{N}_{\text{eff}} < N_{\text{th}}$), apply one of the resampling procedures described in Section 2.3.3 on $\bar{w}_t^{(i)} \; {}_{i=1}^{N}$.
4. For $i = 1, \ldots, N$, sample $x_{t+1}^{(i)} \sim q(x_{t+1} \; \mathbb{X}_t^{(i)}, \mathbb{Y}_{t+1})$.
5. Iterate from step 2.

A simple and intuitive choice for importance function, which is also the choice made in [38], is to use

$$q(x_t \; \mathbb{X}_{t-1}, \mathbb{Y}_t) = p(x_t \; x_{t-1}). \tag{2.50}$$

The importance weights are then updated with

$$w(\mathbb{X}_t) = p(y_t \; x_t) w(\mathbb{X}_{t-1}). \tag{2.51}$$

The problem with choosing the weights according to (2.51) is that we could end up with very skewed weights, particularly if the measurement y_t is very informative (informative in the sense that the support of $p(y_t \; x_t)$ is concentrated compared to $p(x_t \; \mathbb{Y}_{t-1})$).

From (2.46) we see that if we choose

$$q(x_t \; \mathbb{X}_{t-1}, \mathbb{Y}_t) = p(x_t \; x_{t-1}, y_t) = \frac{p(y_t \; x_t) p(x_t \; x_{t-1})}{p(y_t \; x_{t-1})}, \tag{2.52}$$

the weights according to (2.46) are updated with

$$w(\mathbb{X}_t) = p(y_t \; x_{t-1}) w(\mathbb{X}_{t-1}), \tag{2.53}$$

which is independent of x_t. This means that the optimal choice of $q(x_t \; \mathbb{X}_{t-1}, \mathbb{Y}_t)$ with respect to making the weights as uniform as possible with respect to x_t, is given by (2.52) [30]. The problem with the choice in (2.53) is that it only works for certain systems. More specifically we must be able to compute

$$p(y_t \; x_{t-1}) = \quad p(y_t \; x_t) p(x_t \; x_{t-1}) dx_t, \tag{2.54}$$

which is not possible in the general case. Examples on when it is analytically tractable are when the function $h(\cdot)$ in (2.4) is linear with respect to x_t and the process and measurement noises are Gaussian distributed (or distributed as Gaussian mixtures). Another special case where (2.54) is analytically tractable is when x_t only takes on a finite number of different values, changing the integral to a summation.

A third alternative is to use only those realizations $\mathbb{X}_{t-1}^{(i)}$ which provide a high likelihood for y_t. This is possible, before we draw new samples for x_t, by evaluating e.g. $p(y_t \; f(x_{t-1}^{(i)}))$, where $f(x_{t-1}^{(i)})$ is the prediction of $x_{t-1}^{(i)}$ given by the state-space model (2.4). A resampling at this stage, where the probability to pick $\mathbb{X}_{t-1}^{(i)}$ is $p(y_t \; f(x_{t-1}^{(i)}))$, would yield the desired result. This is the principle behind the auxiliary particle filter, see [89], and we have

$$w(\mathbb{X}_t) = \frac{p(y_t \; x_t)}{p(y_t \; f(x_{t-1}))} w(\mathbb{X}_{t-1}). \tag{2.55}$$

2.3.5 Asymptotic Properties

One can show that, under mild assumptions, a central limit theorem holds for the importance sampling method, see Theorem 2.3.

Theorem 2.3 (Central limit theorem for importance sampling)
Assume i.i.d. samples $\mathbb{X}_t^{(i)} \; {}_{i=1}^{N}$, *the support of* $q(\mathbb{X}_t \; \mathbb{Y}_t)$ *includes that of* $p(\mathbb{X}_t \; \mathbb{Y}_t)$, *and the expectations*

$$\mathbb{E}_{p(\mathbb{X}_t \; \mathbb{Y}_t)}[g(x_t)], \quad \mathbb{E}_{p(\mathbb{X}_t \; \mathbb{Y}_t)}[w(\mathbb{X}_t)], \quad \mathbb{E}_{p(\mathbb{X}_t \; \mathbb{Y}_t)}[g(x_t) g(x_t)^T w(\mathbb{X}_t)],$$

exists and are finite. Then we have convergence in distribution for $\hat{g}_t^N$ according to

$$\begin{aligned} \overline{N}(\hat{g}_t^N - \hat{g}_t) &\xrightarrow{d} \mathcal{N}(0, \Sigma_t), \\ \Sigma_t &= \frac{\mathbb{E}_{q(\mathbb{X}_t \ \mathbb{Y}_t)}\left[w(\mathbb{X}_t)^2 (g(x_t) - \hat{g}_t)(g(x_t) - \hat{g}_t)^T\right]}{\left(\mathbb{E}_{q(\mathbb{X}_t \ \mathbb{Y}_t)}\left[w(\mathbb{X}_t)\right]\right)^2}, \end{aligned} \tag{2.56}$$

as N . Moreover, a consistent estimate of the covariance Σ_t is given by

$$\mathrm{cov}(\hat{g}_t^N) = \sum_{i=1}^{N} \left(\bar{w}_t^{(i)}\right)^2 (g(x_t^{(i)}) - \hat{g}_t^N)(g(x_t^{(i)}) - \hat{g}_t^N)^T, \tag{2.57a}$$

$$N\mathrm{cov}(\hat{g}_t^N) \xrightarrow{a.s.} \Sigma_t. \tag{2.57b}$$

Proof: See [36]. □

We see from (2.57b) that the sample covariance of $\hat{g}_t^N$ tends to zero as $\mathcal{O}(N^{-1})$.

Remark 2.1. Note that in the perfect sampling case, i.e. $q(\mathbb{X}_t \ \mathbb{Y}_t) = p(\mathbb{X}_t \ \mathbb{Y}_t)$, and with $g(x_t) = x_t$ we have

$$\Sigma_t = P_{t \ t}, \tag{2.58}$$

where $P_{t \ t}$ is the covariance of the estimate $\hat{x}_{t \ t}$.

As a measure of sampling efficiency we can use the ratio between the sample variance when using a set of samples drawn from the posterior directly (perfect sampling) and the sample variance obtained through the use of importance sampling. The sampling efficiency in its turn gives an expression for the effective sample size, $N_{\text{eff}} \leq N$. In [63] it is shown that the sampling efficiency, assuming a scalar-valued function $g(x_t)$, is approximately given by

$$\hat{N}_{\text{eff}} = \frac{\sum_i (g(x_t^{(i)}) - \hat{g}_t^N)^2 \bar{w}_t^{(i)}}{\sum_i (g(x_t^{(i)}) - \hat{g}_t^N)^2 (\bar{w}_t^{(i)})^2}. \tag{2.59}$$

When we apply resampling we will lose the independence among the samples, and thereby destroying the basic property for the convergence results to hold. A reasonable question is whether it is necessary to perform any kind of resampling at all. The problem here is that the variance of the importance weights can only increase with time. This property is shown in [63] and it implies that as time progresses the weights will become more and more nonuniform. A standard method for deciding when to resample is to use the expression for approximate sampling efficiency according to (2.59). As soon as $\hat{N}_{\text{eff}} < N_{\text{th}}$, where N_{th} is some predefined threshold, a resampling is carried out. Note however that after the first resampling this expression must be used with care since it assumes independent samples.

For the resampling to have its intended effect, i.e. to increase the effective sample size, the sampling step must be such that it scatters the samples. Otherwise we will experience what is referred to as the *depeletion* problem. To overcome the depletion problem we can add artificial noise to the system, u_t^{add}. The idea here is to use more process noise in our

model than what the true system really exhibits. Note that the additional noise should only be added after a resampling. A standard choice for the distribution for u_t^{add} is

$$u_t^{\text{add}} \sim \mathcal{N}(0, \kappa P_{t\ t}), \tag{2.60}$$

where $P_{t\ t}$ is estimated covariance of $\hat{x}_{t\ t}$ computed for example according to (2.38b) and the discount factor κ is a small number typically in the order of $0.01 - 0.001$. The idea of introducing additional noise is not such a bad idea, considering the fact that we discretize the state vector and in turn the densities involved. A way of dealing with discretization errors is to add uncertainty in form of additional noise. The κ should therefore be such that it accounts for the discretization error, i.e. κ should be a function of N. The drawback with introducing additional noise is that we will inevitably lose information, meaning that the estimated covariance of $\hat{x}_{t+1\ t}$ is larger than the actual covariance. One way of dealing with this is to introduce correlation between x_t and u_t^{add} such that the covariances for $\hat{x}_{t+1\ t}$ and $\hat{x}_{t\ t}$ are the same, see [73] for more information on this subject. More sophisticated methods for handling the depletion problem is to use kernel theory on the samples [77], or to apply Markov Chain Monte Carlo (MCMC) methods [37].

Some convergence results for the particle filter exist. In [37] a central limit theorem is presented for the case of sequential importance sampling with resampling and MCMC moves. A more interesting result is given in [25] on mean-square convergence, see Theorem 2.4.

Theorem 2.4 (Convergence for bounded functions)
Assume $\sup_{x_t\ \mathbb{R}^{n_x}}\ p_{e_t}(y_t - h(x_t)) \ <\ $. *Then there exists* c_t *independent of* N *such that for any bounded function* $g(x_t)$

$$\mathbb{E}\big[(\hat{g}_t^N - \hat{g}_t)^2\big] \leq c_t \frac{\sup_{x_t\ \mathbb{R}^{n_x}}\ g(x_t)}{N}, \tag{2.61}$$

where $\hat{g}_t^N$ *and* $\hat{g}_t$ *are given by* (2.37).

Proof: See [25]. □

There are two problems with Theorem 2.4. First of all, it only includes *bounded* functions $g(x_t)$, i.e. the standard least-mean-square estimate using $g(x_t) = x_t$ is not covered. Secondly, c_t can very well grow with time, implying that an ever increasing number of particles has to be used. To ensure a uniform convergence ($c_t = c$), requirements must be put on the dynamic model to exponentially forget any error. In practice this means that for example the more noise the system exhibits, the better the particle filter should be working. A very interesting result on convergence for the particle filter is given by [50]. This result includes *unbounded* functions $g(x_t)$, e.g. the least-mean-square estimate $\hat{x}_{t\ t}$, and is given by Theorem 2.5.

Theorem 2.5 (Convergence for unbounded functions)
For mild assumptions on the unbounded function $g(x_t)$ *and involved densities, see [50] for details, there exists* c_t *independent of* N *such that*

$$\mathbb{E}\big[(\hat{g}_t^N - \hat{g}_t)^4\big] \leq c_t(\mathbb{Y}_t) \frac{\max_{s=0,\ldots,t}\ 1, \mathbb{E}_{p(x_s\ \mathbb{Y}_s)}[g^4(x_s)]}{N^2}, \tag{2.62}$$

where $\hat{g}_t^N$ *and* $\hat{g}_t$ *are given by* (2.37).

Proof: See [50]. □

A problem with Theorem 2.5 is that it is not uniform with respect to the measurements $\mathbb{Y}_s$. Moreover, the applicability of c_t is not obvious as it typically grows rather quickly with time t.

2.4 Marginalized Particle Filter

Theorems 2.4 and 2.5 show that asymptotically the rate of convergence as a function of N is $\mathcal{O}(N^{-1/2})$. Note that this does not mean that the particle filter is independent of the state dimension. On the contrary, in [26] they demonstrate that computational complexity with respect to dimension is between polynomial and exponential, depending on the problem and chosen particle filter algorithm. This is also confirmed in practice, where it is easily shown by comparing problems of different dimensions that there is a dependence. One possibility that can achieve a significant reduction of the sampling variance is to take advantage of potential structure within the state-space model. If there is any structure that can be solved using analytical, closed-form estimation techniques then we should of course do so. For the remaining parts to be estimated we can resort to sampling techniques. The remaining parts are however of lower dimension, meaning that we should not have to use as many samples as before for the same accuracy.

2.4.1 General Description

Consider a state-space model which can be written on the form

$$x_{t+1}^{\mathrm{n}} = f_t^{\mathrm{n}}(x_t^{\mathrm{n}}) + F_t^{\mathrm{n}} x_t^{\mathrm{l}} + G_t^{\mathrm{n}} u_t^{\mathrm{n}}, \tag{2.63a}$$

$$x_{t+1}^{\mathrm{l}} = f_t^{\mathrm{l}}(x_t^{\mathrm{n}}) + F_t^{\mathrm{l}} x_t^{\mathrm{l}} + G_t^{\mathrm{l}} u_t^{\mathrm{l}}, \tag{2.63b}$$

$$y_t = h_t(x_t^{\mathrm{n}}) + H_t x_t^{\mathrm{l}} + e_t, \tag{2.63c}$$

where $x_t = \begin{bmatrix}(x_t^{\mathrm{n}})^T & (x_t^{\mathrm{l}})^T\end{bmatrix}^T$. The superscripts n and l denote what part of the state vector is nonlinear and linear respectively. We assume the noises and initial distribution for x_t^{l} have the properties given by (2.14)– (2.15). The essence of the marginalized particle filter is the factorization of the pdf

$$p(x_t^{\mathrm{l}}, \mathbb{X}_t^{\mathrm{n}}\ \mathbb{Y}_t) = p(x_t^{\mathrm{l}}\ \mathbb{X}_t^{\mathrm{n}}, \mathbb{Y}_t) p(\mathbb{X}_t^{\mathrm{n}}\ \mathbb{Y}_t). \tag{2.64}$$

By using the Rao-Blackwellization technique we use the fact that $p(x_t^{\mathrm{l}}\ \mathbb{X}_t^{\mathrm{n}}, \mathbb{Y}_t)$ exists as a closed-form expression, and that $\mathbb{E}_{p(x_t^{\mathrm{l}}\ \mathbb{X}_t^{\mathrm{n}}, \mathbb{Y}_t)}[g(x_t^{\mathrm{l}})]$ exists and is possible to evaluate. Recall the expression for the expected value of an integrable function $g(x_t)$ from (2.37) repeated here for convenience

$$\hat{g}_t = \mathbb{E}_{p(x_t\ \mathbb{Y}_t)}[g(x_t)] \approx \sum_{i=1}^{N} w_t^{(i)} g(x_t^{(i)}) = \hat{g}_t^N. \tag{2.65}$$

The expected value of $g(x_t) = g(x_t^{\mathrm{l}}, x_t^{\mathrm{n}})$ can be written according to

$$\begin{aligned}
\hat{g}_t &= \mathbb{E}_{p(x_t \ \mathbb{Y}_t)}[g(x_t)] = \mathbb{E}_{p(\mathbb{X}_t^{\mathrm{n}}, x_t^{\mathrm{l}} \ \mathbb{Y}_t)}[g(x_t^{\mathrm{n}}, x_t^{\mathrm{l}})] \\
&= \quad \mathbb{E}_{p(x_t^{\mathrm{l}} \ \mathbb{X}_t^{\mathrm{n}}, \mathbb{Y}_t)}[g(x_t^{\mathrm{n}}, x_t^{\mathrm{l}})] p(\mathbb{X}_t^{\mathrm{n}} \ \mathbb{Y}_t) d\mathbb{X}_t^{\mathrm{n}} \\
&= \frac{\mathbb{E}_{q(\mathbb{X}_t^{\mathrm{n}} \ \mathbb{Y}_t)}\left[\mathbb{E}_{p(x_t^{\mathrm{l}} \ \mathbb{X}_t^{\mathrm{n}}, \mathbb{Y}_t)}[g(x_t^{\mathrm{n}}, x_t^{\mathrm{l}})] w(\mathbb{X}_t^{\mathrm{n}})\right]}{\mathbb{E}_{q(\mathbb{X}_t^{\mathrm{n}} \ \mathbb{Y}_t)}[w(\mathbb{X}_t^{\mathrm{n}})]} \\
&\approx \sum_{i=1}^{N} \bar{w}_t^{(i)} \mathbb{E}_{p(x_t^{\mathrm{l}} \ \mathbb{X}_t^{\mathrm{n},(i)}, \mathbb{Y}_t)}[g(x_t^{\mathrm{n},(i)}, x_t^{\mathrm{l}})] = \hat{g}_{\mathrm{R},t}^N.
\end{aligned} \tag{2.66}$$

From (2.66) we see that the mean and covariance of $g(x_t) = x_t^{\mathrm{n}}$ are the same as before, i.e. computed according to (2.38a) and (2.38b). However, the mean and covariance of x_t^{l} are from (2.66) given by

$$\begin{aligned}
\hat{x}_{t \ t}^{\mathrm{l}} &\approx \sum_{i=1}^{N} \bar{w}_t^{(i)} \mathbb{E}_{p(x_t^{\mathrm{l}} \ \mathbb{X}_t^{\mathrm{n},(i)}, \mathbb{Y}_t)}[x_t^{\mathrm{l}}] \\
&= \sum_{i=1}^{N} \bar{w}_t^{(i)} \hat{x}_{t \ t}^{\mathrm{l},(i)}, \\
P_{t \ t}^{\mathrm{l}} &\approx \sum_{i=1}^{N} \bar{w}_t^{(i)} \mathbb{E}_{p(x_t^{\mathrm{l}} \ \mathbb{X}_t^{\mathrm{n},(i)}, \mathbb{Y}_t)}[(x_t^{\mathrm{l}} - \hat{x}_{t \ t}^{\mathrm{l}})(x_t^{\mathrm{l}} - \hat{x}_{t \ t}^{\mathrm{l}})^T] \\
&= \sum_{i=1}^{N} \bar{w}_t^{(i)} \left(P_{t \ t}^{\mathrm{l},(i)} + (\hat{x}_{t \ t}^{\mathrm{l},(i)} - \hat{x}_{t \ t}^{\mathrm{l}})(\hat{x}_{t \ t}^{\mathrm{l},(i)} - \hat{x}_{t \ t}^{\mathrm{l}})^T\right).
\end{aligned} \tag{2.67}$$

In the expressions (2.67) we have used that the estimate and covariance of x_t^{l}, *conditionally upon a given realization* $\mathbb{X}_t^{\mathrm{n},(i)}$, denoted by $\hat{x}_{t \ t}^{\mathrm{l},(i)}$ and $P_{t \ t}^{\mathrm{l},(i)}$ respectively, where

$$P_{t \ t}^{\mathrm{l},(i)} = \mathbb{E}_{p(x_t^{\mathrm{l}} \ \mathbb{X}_t^{\mathrm{n},(i)}, \mathbb{Y}_t)}[(x_t^{\mathrm{l}} - \hat{x}_{t \ t}^{\mathrm{l},(i)})(x_t^{\mathrm{l}} - \hat{x}_{t \ t}^{\mathrm{l},(i)})^T], \tag{2.68}$$

are analytically tractable. Algorithm 2.3 provides the conceptual steps in the marginalized particle filter. For details the reader is referred to Paper A.

Algorithm 2.3 (Marginalized particle filter).

1. Initialization: Initialize the particles and set initial values for the linear state variables, to be used in the Kalman filter.
2. Particle filter measurement update: evaluate the importance weights and normalize.
3. Resample with replacement (if necessary).
4. Kalman filter measurement update
5. Particle filter time update: Predict new particles.
6. Kalman filter time update.
7. Iterate from step 2.

2.4.2 Filter for Terrain-Aided Navigation

The terrain-navigation application can be posed as a recursive state estimation problem according to

$$x_{t+1}^{\mathrm{n}} = f_t^{\mathrm{n}}(x_t^{\mathrm{n}}) \qquad + F_t^{\mathrm{n}} x_t^{\mathrm{l}} + G_t^{\mathrm{n}} u_t^{\mathrm{n}}, \tag{2.69a}$$

$$x_{t+1}^{\mathrm{d}} = f_t^{\mathrm{d}}(x_t^{\mathrm{n}}) + F_t^{\mathrm{d}} x_t^{\mathrm{d}} \qquad + G_t^{\mathrm{d}} u_t^{\mathrm{d}}, \tag{2.69b}$$

$$x_{t+1}^{\mathrm{l}} = f_t^{\mathrm{l}}(x_t^{\mathrm{n}}) \qquad + F_t^{\mathrm{l}} x_t^{\mathrm{l}} + G_t^{\mathrm{l}} u_t^{\mathrm{l}}, \tag{2.69c}$$

$$y_t = h_t(x_t^{\mathrm{n}}) + H_t x_t^{\mathrm{d}} \qquad + e_t(\lambda_t). \tag{2.69d}$$

Here x_t^{n}, x_t^{d} and x_t^{l} correspond to horizontal position, altitude and remaining navigation quantities respectively. Note the mode dependent measure noise $e_t(\lambda_t)$. The purpose of the mode variable λ_t is to model the bimodal character of the radar measurements. For example, flying over dense forrest yields radar measurements which can either correspond to distance to ground or the tree tops. We could apply Algorithm 2.3 to estimate $[x_t^{\mathrm{l}}\ x_t^{\mathrm{d}}]^T$ and x_t^{n} respectively. However, using the GPB1 filter according to Algorithm 2.1, we would need two multi-dimensional Kalman filters running in parallel. The remedy is to use the structural independence in (2.69) between x_t^{d} and x_t^{l}, which makes it possible to further split the estimation. The posterior probability density can be factorized according to

$$p(x_t^{\mathrm{l}}, x_t^{\mathrm{d}}, \mathbb{X}_t^{\mathrm{n}}\ \mathbb{Y}_t) = p(x_t^{\mathrm{l}}\ \mathbb{X}_t^{\mathrm{n}}, \mathbb{Y}_t) p(x_t^{\mathrm{d}}\ \mathbb{X}_t^{\mathrm{n}}, \mathbb{Y}_t) p(\mathbb{X}_t^{\mathrm{n}}\ \mathbb{Y}_t). \tag{2.70}$$

This means that, conditionally upon $\mathbb{X}_t^{\mathrm{n}}$, we can estimate x_t^{l} and x_t^{d} independently of each other. We can estimate the mode dependent x_t^{d} with two one-dimensional Kalman filters and apply a single Kalman filter for estimation of x_t^{l}. Algorithm 2.4 provides the conceptual steps in the Terrain-Aided Navigation Filter. For details the reader is referred to Paper B.

Algorithm 2.4 (Terrain-Aided Navigation Filter).

1. Initialization: Initialize the particles and set initial values for the linear state variables, to be used in the Kalman filter.
2. Particle filter measurement update: evaluate the importance weights and normalize.
3. Resample with replacement (if necessary).
4. GPB filter measurement update and merge
5. Kalman filter measurement update
6. Particle filter time update: Predict new particles.
7. Kalman filter time update.
8. GPB filter time update.
9. Iterate from step 2.

The filter from paper B has been tested on authentic flight data as described in [47, 48].

2.4.3 Asymptotic Properties

Similar to Theorem 2.3 one can derive a central limit theorem for the case with the Rao-Blackwellization technique applied. The result is given by Theorem 2.6

Theorem 2.6 (CLT for Rao-Blackwellization)
Assume i.i.d. samples $\mathbb{X}_t^{(i)}\ {}_{i=1}^{N}$*, the support of* $q(\mathbb{X}_t\ \mathbb{Y}_t)$ *includes that of* $p(\mathbb{X}_t\ \mathbb{Y}_t)$ *and that the involved expectations exist and are finite. Then we have convergence in distribution for* $\hat{g}_t^N$ *according to*

$$\begin{aligned}
&\overline{N}(\hat{g}_{\mathrm{R},t}^N - \hat{g}_t) \xrightarrow{d} \mathcal{N}(0, \Sigma_{\mathrm{R},t}),\\
&\Sigma_{\mathrm{R},t} = \frac{\mathbb{E}_{q(\mathbb{X}_t^{\mathrm{n}}\ \mathbb{Y}_t)}\ \mathbb{E}_{p(x_t^{\mathrm{l}}\ \mathbb{X}_t^{\mathrm{n}},\mathbb{Y}_t)}[(g(x_t)-\hat{g}_t)(g(x_t)-\hat{g}_t)^T]w(\mathbb{X}_t^{\mathrm{n}})^2}{\left(\mathbb{E}_{q(\mathbb{X}_t^{\mathrm{n}}\ \mathbb{Y}_t)}\left[w(\mathbb{X}_t^{\mathrm{n}})\right]\right)^2},
\end{aligned} \tag{2.71}$$

as N *. A consistent estimate of* $\Sigma_{\mathrm{R},t}$ *and the sample covariance* $\mathrm{cov}(\hat{g}_{\mathrm{R},t}^N)$ *are given by*

$$\begin{aligned}
&\mathrm{cov}(\hat{g}_{\mathrm{R},t}^N)\\
&= {}_{i=1}^{N} \left(\mathbb{E}_{p(x_t^{\mathrm{l}}\ \mathbb{X}_t^{\mathrm{n},(i)},\mathbb{Y}_t)}[g^{(i)}(x_t)-\hat{g}_t]\right)\left(\mathbb{E}_{p(x_t^{\mathrm{l}}\ \mathbb{X}_t^{\mathrm{n},(i)},\mathbb{Y}_t)}[g^{(i)}(x_t)-\hat{g}_t]\right)^T(\bar{w}_t^{(i)})^2,\\
&N\mathrm{cov}(\hat{g}_{\mathrm{R},t}^N) \xrightarrow{a.s.} \Sigma_{\mathrm{R},t}
\end{aligned} \tag{2.72}$$

where $g^{(i)}(x_t) = g(x_t^{\mathrm{n},(i)}, x_t^{\mathrm{l}})$.

Proof: See [29]. □

One can also show that, in accordance with our intuition, the variance obtained when applying Rao-Blackwellization is always less than or equal to the variance when not applying it, see Corollary 2.1.

Corollary 2.1 (Rao-Blackwellization variance reduction)
Assume that the assumptions given in Theorem 2.3 are valid and that the expectations involved exist and are finite. Then

$$\Sigma_t - \Sigma_{\mathrm{R},t} = \frac{\mathbb{E}_{q(\mathbb{X}_t^{\mathrm{n}}\ \mathbb{Y}_t)}\ \mathrm{cov}_{q(\mathbb{X}_t^{\mathrm{l}}\ \mathbb{X}_t^{\mathrm{n}},\mathbb{Y}_t)}\big((g(x_t)-\hat{g}_t)w(\mathbb{X}_t)\big)}{\left(\mathbb{E}_{q(\mathbb{X}_t\ \mathbb{Y}_t)}\left[w(\mathbb{X}_t)\right]\right)^2} \geq 0, \tag{2.73}$$

where Σ_t *and* $\Sigma_{\mathrm{R},t}$ *are given by* (2.56) *and* (2.71) *respectively.*

Proof: See [28]. □

From Lemma 2.1 we conclude that, when the average conditional variance of $g(x_t)$ is high we can gain a lot using Rao-Blackwellization.

To be more specific about what we can gain using Rao-Blackwellization, consider a system according to (2.63) where $H_t \equiv 0$. Compare with terrain-aided navigation (2.69) where $H_t = 0$ with respect to x_t^{l}. In this case we have

$$p(\mathbb{X}_t\ \mathbb{Y}_t) = p(\mathbb{X}_t^{\mathrm{l}}\ \mathbb{X}_t^{\mathrm{n}}, \mathbb{Y}_t)p(\mathbb{X}_t^{\mathrm{n}}\ \mathbb{Y}_t) = p(\mathbb{X}_t^{\mathrm{l}}\ \mathbb{X}_t^{\mathrm{n}})p(\mathbb{X}_t^{\mathrm{n}}\ \mathbb{Y}_t), \tag{2.74}$$

where the second equality stems from the fact that if $\mathbb{X}_t^{\mathrm{n}}$ is given then $\mathbb{Y}_t$ provides no further information. Based on (2.74) and the assumption that we use $q(\mathbb{X}_t \; \mathbb{Y}_t) = p(\mathbb{X}_t)$ as importance function we can simplify the expression for the importance weight according to

$$\begin{aligned} w(\mathbb{X}_t) &= p(\mathbb{Y}_t)\frac{p(\mathbb{X}_t \; \mathbb{Y}_t)}{p(\mathbb{X}_t)} = p(\mathbb{Y}_t)\frac{p(\mathbb{X}_t^{\mathrm{l}} \; \mathbb{X}_t^{\mathrm{n}})p(\mathbb{X}_t^{\mathrm{n}} \; \mathbb{Y}_t)}{p(\mathbb{X}_t^{\mathrm{l}} \; \mathbb{X}_t^{\mathrm{n}})p(\mathbb{X}_t^{\mathrm{n}})} \\ &= p(\mathbb{Y}_t)\frac{p(\mathbb{X}_t^{\mathrm{n}} \; \mathbb{Y}_t)}{p(\mathbb{X}_t^{\mathrm{n}})} = w(\mathbb{X}_t^{\mathrm{n}}). \end{aligned} \tag{2.75}$$

This means that the importance weight only depends on $\mathbb{X}_t^{\mathrm{n}}$ whether we apply Rao-Blackwellization or not. We can now rewrite (2.73) according to

$$\begin{aligned} \Sigma_t - \Sigma_{\mathrm{R},t} &= \frac{\mathbb{E}_{p(\mathbb{X}_t^{\mathrm{n}})}\left[w(\mathbb{X}_t^{\mathrm{n}})^2 \mathrm{cov}_{p(x_t^{\mathrm{l}} \; \mathbb{X}_t^{\mathrm{n}})}\big(g(x_t) - \hat{g}_t\big)\right]}{\left(\mathbb{E}_{p(\mathbb{X}_t^{\mathrm{n}})}\left[w(\mathbb{X}_t^{\mathrm{n}})\right]\right)^2} \\ &\approx N \sum_{i=1}^{N} (\bar{w}_t^{(i)})^2 \mathrm{cov}_{p(x_t^{\mathrm{l}} \; \mathbb{X}_t^{\mathrm{n},(i)})}\big(g(x_t^{\mathrm{n},(i)}, x_t^{\mathrm{l}}) - \hat{g}_t\big). \end{aligned} \tag{2.76}$$

From (2.76) we see that for $g(x_t) = x_t^{\mathrm{n}}$ we gain nothing using Rao-Blackwellization in this case, because $\mathrm{cov}_{p(x_t^{\mathrm{l}} \; \mathbb{X}_t^{\mathrm{n}})}(x_t^{\mathrm{n}} - \hat{x}_{t\;t}^{n}) = 0$. On the other hand, for $g(x_t) = x_t^{\mathrm{l}}$ we have that

$$\mathrm{cov}_{p(x_t^{\mathrm{l}} \; \mathbb{X}_t^{\mathrm{n},(i)})}(x_t^{\mathrm{l}} - \hat{x}_{t\;t}^{\mathrm{l}}) = P_t^{\mathrm{l},(i)}, \tag{2.77}$$

which inserted into (2.76) yields

$$\Sigma_t - \Sigma_{\mathrm{R},t} \approx N \sum_{i=1}^{N} (\bar{w}_t^{(i)})^2 P_t^{\mathrm{l},(i)}. \tag{2.78}$$

Before we can draw any conclusions we have to consider the resulting variance from (2.72) for $g(x_t) = x_t^{\mathrm{l}}$, which is

$$\Sigma_{\mathrm{R},t} \approx N \sum_{i=1}^{N} (\bar{w}_t^{(i)})^2 (\hat{x}_{t\;t}^{\mathrm{l},(i)} - \hat{x}_{t\;t}^{\mathrm{l}})(\hat{x}_{t\;t}^{\mathrm{l},(i)} - \hat{x}_{t\;t}^{\mathrm{l}})^T. \tag{2.79}$$

By comparing (2.78) with (2.79) we see that if the covariances $P_t^{\mathrm{l},(i)}$ provided by the Kalman filters are large compared to the corresponding spread of the mean terms $(\hat{x}_{t\;t}^{\mathrm{l},(i)} - \hat{x}_{t\;t}^{\mathrm{l}})(\hat{x}_{t\;t}^{\mathrm{l},(i)} - \hat{x}_{t\;t}^{\mathrm{l}})^T$ we should be able to gain a lot by applying Rao-Blackwellization.

In practice we are confined to use a finite number of samples in combination with resampling. In this case, including when the Kalman filter covariances $P_t^{\mathrm{l},(i)}$ are small or even zero, there is a potentially even higher gain in using Rao-Blackwellization. For an illustration see Example 2.3.

Example 2.3: Rao-Blackwellization

Consider the system

$$\begin{aligned} x_{t+1} = \begin{bmatrix} x_t^{\mathrm{n}} \\ x_t^{\mathrm{l}} \end{bmatrix} &= \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} x_t + \begin{bmatrix} 1 \\ 1 \end{bmatrix} u_t. \\ y_t &= h(x_t^{\mathrm{n}}) + e_t, \end{aligned} \tag{2.80}$$

In this case

$$x_{t+1}^{\mathrm{l}} = x_t^{\mathrm{l}} + u_t = x_t^{\mathrm{l}} + x_{t+1}^{\mathrm{n}} - x_t^{\mathrm{n}} - x_t^{\mathrm{l}} = x_{t+1}^{\mathrm{n}} - x_t^{\mathrm{n}}, \tag{2.81}$$

where we have used $u_t = x_{t+1}^{\mathrm{n}} - x_t^{\mathrm{n}} - x_t^{\mathrm{l}}$. In words, x_{t+1}^{l} is known given x_{t+1}^{n} and x_t^{n} meaning that $P_{t+1\ t}^{\mathrm{l},(i)} = 0$. From (2.78) and (2.79), we can conclude that here we gain nothing from Rao-Blackwellization. However, this is not necessarily true for a finite N in combination with resampling. A study with 10000 simulations and

$$h(x_t^{\mathrm{n}}) = x_t^{\mathrm{n}}, \quad Q = 2^2, \quad R = 5^2, \quad P_0 = \begin{bmatrix} 50^2 & 0 \\ 0 & 10^2 \end{bmatrix}, \tag{2.82}$$

yields the result as shown in Figure 2.3. Here we use a linear measurement equation to be able compute and compare with the optimal estimate given by the Kalman filter. The Root Mean Square Error (RMSE) is computed according to

$$\mathrm{RMSE} = \left(\frac{1}{10000} \sum_{m=1}^{10000} (\hat{x}_{3\ 2}^{(m)} - x_{3\ 2}^{\mathrm{true},(m)})(\hat{x}_{3\ 2}^{(m)} - x_{3\ 2}^{\mathrm{true},(m)})^T \right)^{1/2}. \tag{2.83}$$

We see that here it is most effective to use both resampling and Rao-Blackwellization. The reason is that resampling by itself, despite introducing dependence, improves estimation performance. Moreover, a resampling in combination with Rao-Blackwellization increases the performance even more. The reason is that if we do not use Rao-Blackwellization we initialize by sampling both x_0^{n} and x_0^{l}. A resampling will result in many samples where $x_0^{\mathrm{n},(i)}$ are the same but also many $x_0^{\mathrm{l},(i)}$. The time update of $x_0^{\mathrm{n},(i)}$ is given by

$$x_{1\ 0}^{\mathrm{n},(i)} \sim \mathcal{N}\big(x_0^{\mathrm{n},(i)} + x_0^{\mathrm{l},(i)}, Q\big). \tag{2.84}$$

Here many of the samples will still be approximately the same, particularly if Q is small. For the case with Rao-Blackwellization we time update the samples according to

$$x_{1\ 0}^{\mathrm{n},(i)} \sim \mathcal{N}\big(x_0^{\mathrm{n},(i)}, Q + P_0^{\mathrm{l}}\big). \tag{2.85}$$

We see that the diversity among the samples increases because of the additional covariance term P_0^{l} in (2.85).

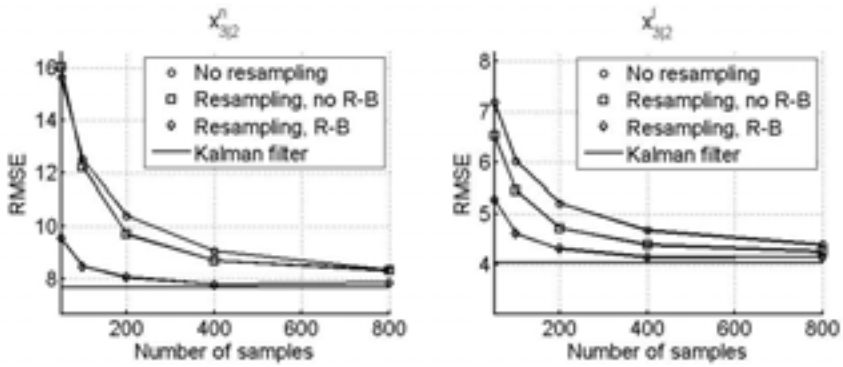

Figure 2.3: *A comparison between* $\hat{x}^{\mathrm{n}}_{3\ 2}$ *and* $\hat{x}^{\mathrm{l}}_{3\ 2}$ *for the system* (2.80) *with parameters given by* (2.82) *computed by using no resampling, resampling, resampling and Rao-Blackwellization (R-B), and the Kalman filter. The RMSE is based on* 10000 *simulations. Resampling is applied after the first measurement.*

3

Ef cient Detection

3.1 Sense and Avoid

The principle for the sense and avoid system under consideration is depicted in Figure 3.1. The intruder must first be detected by a sensor, in this case a video camera. After detection the sensor feeds a tracking filter with readings of azimuth and elevation angles relative to the intruder. Based on the angle measurements the tracking filter estimates target position and velocity. Finally we compute the probability of near mid-air collision using estimated relative position and velocity. If the probability is large an avoidance maneuver is initiated.

3.1.1 Sensor

There are many sensors applicable for collision avoidance. We will focus on electro-optical sensors, and more specifically video cameras. The advantage with video cameras is that they are much less expensive and readily available as Commercial Off-The-Shelf (COTS) compared to for example radars. Moreover, the bearing accuracy is very high (similar to or better than the human eye). The resolution of a video camera is today typically $0.1 - 1$ mrad. As an example the human eye normally has a spatial resolution of about 0.3 mrad. Video cameras with silicon Charged Coupled Devices (CCDs) have sensitivities well into the near infrared area ($\approx 1 \cdot 10^{-6}$ m), which exceeds the capability of the human eye. The most prominent problems with cameras are that range is not measured and that they are limited to high visibility atmospheric conditions and daytime light conditions. Note however that the visibility conditions apply to the human see and avoid capability as well. Several sensors can be mounted on a rack to cover required field of regard or view, ± 110 deg horisontally and ± 15 deg vertically, see Figure 3.2.

Background movement is measured by analyzing the optical flow between successive frames. An object can then be detected which moves in a manner that deviates from the background [75]. For example, a vehicle on collision course is approximately still in

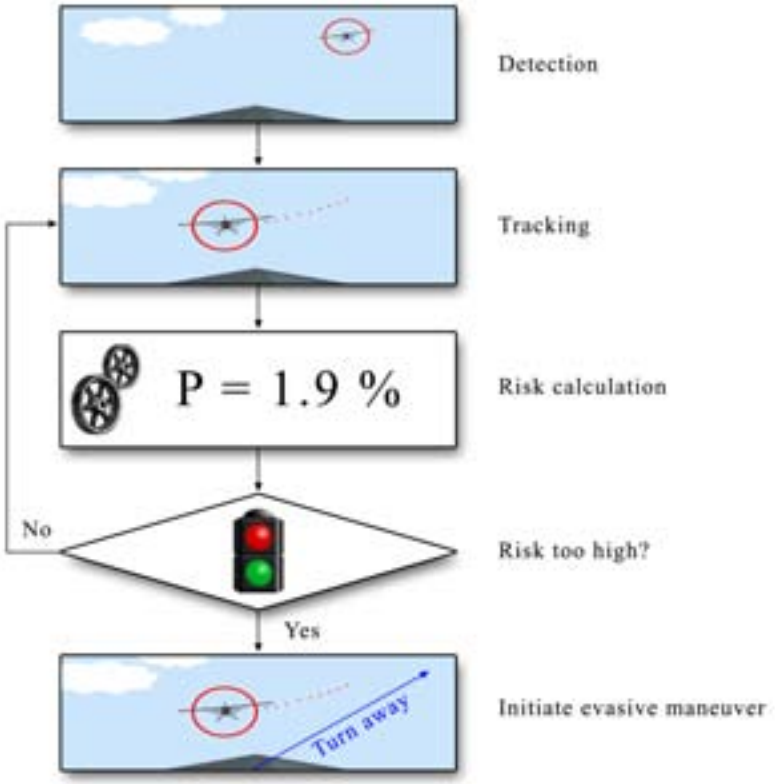

Figure 3.1: *Concept for sense and avoid.*

successive frames and thereby provides the means to distinguish it from the background. Typical detection distance for a video camera with a resolution of 0.3 mrad, assuming good visibility conditions and a target cross-section of $3\ \mathrm{m}^2$, is around 4000 m with 90% probability of detection. This is confirmed by the demonstration described in [75]. The demonstration showed a probability of detection, here defined as the number of correct detections divided by number of frames processed, using off-line processing close to 100% at a distance of 4000 m. At the same time probability of false alarm, here defined as number of false detections divided by number of frames and pixels per frames, was approximately 0.01%. For a comparison the human eye is capable of detecting an intruder at a distance of approximately 3000 m (F-16 with probability 90%) [75]. Intutively, on-line processing with platform vibrations will worsen the result [107].

3.1.2 Tracking

The purpose of target tracking is to estimate the kinematic state of a target or intruder based on noisy measurements. This is a recursive state estimation problem, and the theory in Section 2 is applicable. The state is usually confined to relative position s and velocity

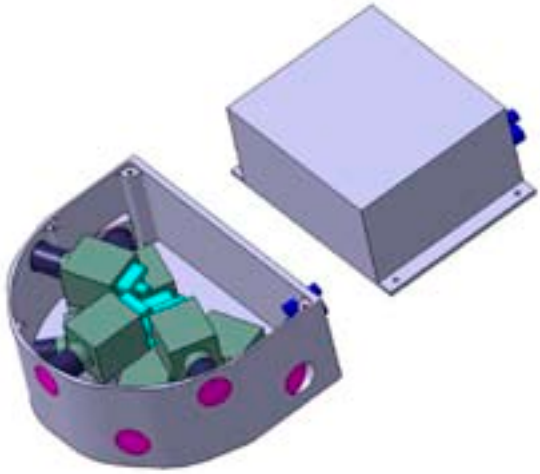

Figure 3.2: *Example on camera mounting.*

v, in Cartesian coordinates given by

$$x^{\text{car}} = \begin{bmatrix} s_X & s_Y & s_Z & v_X & v_Y & v_Z \end{bmatrix}^T. \tag{3.1}$$

Relative position is obtained by subtracting intruder position and own position, i.e.

$$s = s^{\text{int}} - s^{\text{own}}, \tag{3.2}$$

and equivalently for relative velocity. In angle-only tracking the measurements are azimuth and elevation, denoted by η and ϵ respectively. These angles are defined by

$$\eta = \arctan\left(\frac{s_Y}{s_X}\right), \tag{3.3}$$

$$\epsilon = \arctan\left(\frac{s_Z}{\sqrt{s_X^2 + s_Y^2}}\right). \tag{3.4}$$

For good tracking performance it is important to model the motion of the intruder as accurately as possible. The simplest model is based on the assumption that the target has approximately constant velocity, i.e.

$$\begin{bmatrix} \dot{v}_X^{\text{int}} & \dot{v}_Y^{\text{int}} & \dot{v}_Z^{\text{int}} \end{bmatrix}^T \approx \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^T. \tag{3.5}$$

If it is inappropriate to assume constant velocity there are other possible models which deal with motion in one way or another [94]. The own vehicle is subject to acceleration particularly during an avoidance maneuver

$$\begin{bmatrix} \dot{v}_X^{\text{own}} & \dot{v}_Y^{\text{own}} & \dot{v}_Z^{\text{own}} \end{bmatrix}^T = \begin{bmatrix} a_X & a_Y & a_Z \end{bmatrix}^T, \tag{3.6}$$

where the acceleration is measured accurately typically by an Inertial Navigation System (INS).

It is a well known fact that distance or range is unobservable for angle-only measurements and a constant relative velocity. To gain full observability the observer platform must out-maneuver the target, i.e. there must be higher non-zero derivative of the observer motion compared to the target [14, 99]. Very inaccurate range estimates can yield large linearization errors. This can affect the other, observable states such that they diverge. In other words, the unobservable range can affect tracking filter stability.

There are two widely used methods for angle-only tracking, Range Parameterized EKF (RP-EKF) and Modified Spherical EKF (MS-EKF). Both are known to provide stable estimates of the target state vector. The principle behind MS-EKF is to decouple the observable states from the non-observable range. This is accomplished by representing the state vector in modified spherical coordinates according to [34]

$$\begin{aligned} x^{\mathrm{msc}} &= \begin{bmatrix} \frac{1}{r} & \eta & \epsilon & \frac{\dot{r}}{r} & \dot{\eta}\cos\epsilon & \dot{\epsilon} \end{bmatrix}^T, \\ r &= \sqrt{s_X^2 + s_Y^2 + s_Z^2}. \end{aligned} \tag{3.7}$$

From the motion equations for x^{msc} it is clear that the first state representing the inverse of range is decoupled from the other states during periods with constant velocity [34]. The principle of RP-EKF is to apply a Gaussian sum filter as demonstrated by Example 2.1. For examples on typical tracking filter performance when using an angle-only sensor see e.g. [56, 34, 72]. Note that it is highly important to accurately estimate not only the mean but the covariance matrix, denoted by P, as well. This is due to the fact that when we compute probability of Near Mid-Air Collision (NMAC) based on tracking data we need the probability density function (pdf) for the state vector. In case of a Gaussian distributed state vector the mean and covariance give a complete description of the pdf. A justified question is whether the output from the tracking filter is normally distributed or not. The results using MS-EKF given by [72] indicate that the output from the tracking filter is indeed normally distributed. Due to the low observability along line-of-sight, angle-only tracking typically results in large covariances for the variables along line-of-sight (range and closing speed).

The natural extension of the standard EKF to deal with a maneuvering target, where the maneuvering is well modeled by a discrete parameter, is to apply the IMM filter [16]. Note however it is not clear that the IMM outperforms EKF merely based on whether the target maneuvers or not [59]. Another important issue to deal with is data or measurement association in case of multiple-target tracking [14].

3.1.3 Near Mid-Air Collision Avoidance

There are numerous methods which provide collision avoidance, see for example [69] for a survey of different methods for conflict detection and resolution. Note that many of the methods investigated in [69] are primarily for separation assurance and not NMAC avoidance, although the basic principle is the same. The main difference is with respect to time, separation conflict is usually detected and resolved at a much earlier stage compared to a last resort collision avoidance maneuver.

We adopt a probabilistic framework, since the uncertainties are of the same magnitude or larger compared to the safety zone. The uncertainties come from the tracking filter estimating the intruder s position and velocity. Given an estimated state vector and

covariance from the tracking filter we must avoid NMACs with a predefined probability. At the same time we must not initiate an avoidance maneuver if not really necessary, i.e. the probability for nuisance maneuvers shall not exceed a predefined level. A nuisance maneuver causes an unnecessary interruption to the mission of the vehicle which is undesirable.

We are interested in the probability of NMAC for a predicted trajectory. As long as there is at least one trajectory which complies with an acceptable maximum risk nothing happens, i.e. the flight continues along its predetermined trajectory. If the probability for a particular trajectory exceeds the acceptable level, we need to find another trajectory which has a probability less than the acceptable level, compare with Figure 3.3. If the trajectory

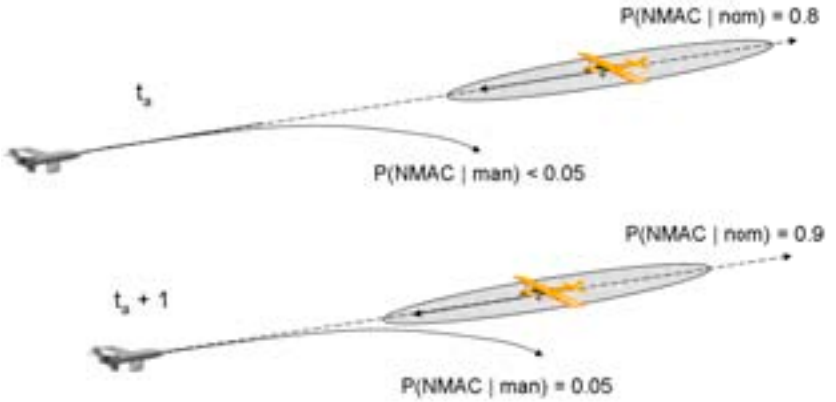

Figure 3.3: *An illustration of an encounter with the ellipse representing the uncertainty of intruder position. Here two trajectories, corresponding to a maneuver (man) and no maneuver (nom), are checked. At time t_a the maneuver trajectory results in an acceptable $\mathrm{P(NMAC\ man)}$, here less than 0.05, and the flight continues along nom. At time $t_a + 1$ the maneuver trajectory yields $\mathrm{P(NMAC\ man)} = 0.05$. If no other maneuver yields a lower probability man is initiated.*

corresponding to a last resort avoidance maneuver exceeds the level of acceptable risk at the subsequent time instant the evasive maneuver is executed immediately. The main reason for executing an avoidance maneuver as late as possible is to minimize the amount of nuisance maneuvers [103]. The tracking filter accuracy increases over time since the estimate is based on more measurements. The more certain estimate of intruder position and velocity we have access to the better we can judge whether an avoidance maneuver is really necessary or not.

For an accurate estimate of probability of NMAC we must be able to accurately predict the relative trajectory. The predicted trajectory must also be feasible, i.e. the trajectory must be such that it corresponds to a maneuver which the vehicle can actually perform. This means that the acceleration in (3.6) is limited, where the actual limit dependence on vehicle platform and power.

3.2 Probability of Near Mid-Air Collision

Denote the predicted relative position with $s(t)$, where $t = 0$ represents the present time. For a predicted trajectory the probability of NMAC is

$$\mathrm{P}(\mathrm{NMAC}_{t>0}) = \mathrm{P}\big(\min_{t>0} s(t) < \mathrm{R}\big), \tag{3.8}$$

where $s(t)$ corresponds to the distance between the vehicles and R is the safety zone radius. Here we assume the safety zone is given by a sphere. This is not necessary, but it makes the notation simpler. The probability (3.8) is a prediction from current time into the future. This means that (3.8) is actually a function of current time, denoted with t_{a} (a as in absolute time). Usually we omit the dependence on t_{a} for notational convenience.

Closest point of approach (CPA) is the point where the distance between two vehicles is at its minimum. For an illustration see Example 3.1.

Example 3.1

For a straight trajectory CPA occurs when the vectors representing relative position and relative velocity respectively are perpedendicular. Time to CPA, t_{cpa}, is for a constant velocity defined by the equation

$$v^T s(t_{\mathrm{cpa}}) = v^T(s(0) + vt_{\mathrm{cpa}}) = 0. \tag{3.9}$$

Using the closest point of approach we can write the probability of NMAC as

$$\mathrm{P}(\mathrm{NMAC}_{t>0}) = \mathrm{P}\big(\ \mathrm{CPA} < \mathrm{R}\big). \tag{3.10}$$

There are other suggestions on risk measures, where the most common found in literature is the instantaneous probability of NMAC (or conflict with respect to separation) given by [90, 91, 22, 23]

$$\mathrm{P}(\mathrm{NMAC}_t) = \mathrm{P}\big(\ s(t) < \mathrm{R}\big), \tag{3.11}$$

which is then typically maximized over t. A collision detection system based on bearing measurements explicitly is described in [6], without providing any details about parameter settings. In [62] they investigate the performance of a collision avoidance system based on line-of-sight rate, and come to the conclusion that range information is highly beneficial.

The definition of probability of NMAC according to (3.8) and (3.10) yields the probability for a predicted trajectory. We can compute probability of NMAC for a predicted trajectory by sampling N times from estimated intruder state vector and for each sample i compute $\mathrm{CPA}^{(i)}$. Then $\mathrm{P}(\mathrm{NMAC}_{t>0})$ is approximately given by

$$\mathrm{P}(\mathrm{NMAC}_{t>0}) \approx \frac{1}{N}\sum_{i=1}^{N} I\big(\ \mathrm{CPA}^{(i)} < \mathrm{R}\big), \tag{3.12}$$

where $I(\cdot)$ is the indicator function. For an illustration see Figure 3.4.

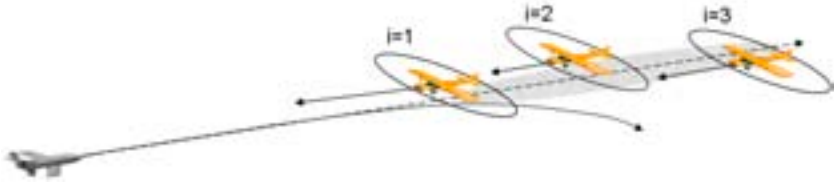

Figure 3.4: *Sample N times (in the picture $N = 3$) from estimated intruder state vector and compute $\mathrm{CPA}^{(i)}\ {}_{i=1}^{N}$. Then $\mathrm{P}(\mathrm{NMAC})$ is approximately given by the frequency of $\mathrm{CPA}^{(i)} < \mathrm{R}$.*

3.3 Detection using hypothesis testing

Near mid-air collision avoidance can be cast as a hypothesis testing [57, 105, 51], where the hypotheses are

$$\begin{aligned} \mathcal{H}_0 &: \text{no NMAC encounter} \quad \overline{\mathrm{NMAC}}_{t>0} \quad \mathrm{CPA} > \mathrm{R}, \\ \mathcal{H}_1 &: \text{NMAC encounter} \quad \mathrm{NMAC}_{t>0} \quad \mathrm{CPA} < \mathrm{R}. \end{aligned} \tag{3.13}$$

These hypotheses are either outcomes or assumed known depending on the context. The actions of the NMAC avoidance system is either an avoidance maneuver initiated at any absolute time t_a (= avoidance) or no avoidance maneuver at all (= no avoidance). Since we consider relative position and velocity to be stochastic we do not know, in advance, what the outcome of the encounter will be exactly. We need to resort to the probability of the outcome. The characterization of the performance of the NMAC avoidance system is possible through [109]

$$\text{avoidance unsuccessful} \quad \mathcal{H}_1 \,\big|\, \text{avoidance}, \tag{3.14a}$$

$$\text{avoidance necessary} \quad \mathcal{H}_1 \,\big|\, \text{no avoidance}. \tag{3.14b}$$

The events in (3.14) means that the encounter results in a NMAC given an avoidance maneuver or no avoidance maneuver respectively. It is desirable to have a system which achieves a low probability of unsuccessful avoidance when the necessity of avoidance is high. On the other hand, when there is no need for avoidance, i.e. probability of avoidance necessary is low, then the system should not initate any avoidance maneuver.

Remark 3.1. Compare with standard notation in detection theory, i.e. missed detection and false alarm, which here corresponds to

$$\text{missed detection} \quad \text{no avoidance} \,\big|\, \mathcal{H}_1, \tag{3.15a}$$

$$\text{false alarm} \quad \text{avoidance} \,\big|\, \mathcal{H}_0. \tag{3.15b}$$

The second event (3.15b) is also referred to as a nuisance maneuver. The main problem with (3.15) is that it does not provide any information regarding the success of any applied avoidance maneuver.

The principle of computing the probability of NMAC for a predicted trajectory fits nicely in a setting where there is a requirement on maximum acceptable probability of

NMAC. Assume the requirement is that the collision avoidance system must have a certain capability of avoiding NMACs, i.e.

$$\mathrm{P}(\text{avoidance unsuccessful}) = \mathrm{P}_{\mathrm{req}}. \tag{3.16}$$

Assume that we, at a certain absolute time t_a, have a computed probability of NMAC that is $\mathrm{P}_{\mathrm{req}}$. The computed value is based on the relative state probability density function (pdf) and a predicted relative trajectory. Consider now that we simulate a large number of encounter simulations at the same time t_a using (3.12). The initial conditions are given by the same relative state pdf and the predicted trajectory is the same as well. The outcome, i.e. the frequency of NMACs, will tend to $\mathrm{P}_{\mathrm{req}}$ as the number of simulations tend to infinity. This is quite intuitive, and is also confirmed in [72]. To simplify further analysis and be able to focus on the risk assessment computations we need to make three assumptions

- The video camera has detected the intruder at a minimum distance,
- The tracking filter estimate of relative state is sufficiently accurate,
- The vehicle has a minimum maneuverability.

The extent of these assumptions (minimum distance, sufficiently accurate and minimum maneuverability) must be such that $\mathrm{P}_{\mathrm{req}}$ is larger than computed $\mathrm{P}(\mathcal{H}_1 \mid \text{avoidance})$ at the point in time t_a when the first risk assessment computation is performed. Then for successive computations of $\mathrm{P}(\mathcal{H}_1 \mid \text{avoidance})$, if it at any point reaches $\mathrm{P}_{\mathrm{req}}$ we initiate the avoidance maneuver, compare with Figure 3.3. The outcome of encounter simulations, the frequency of unsuccessful avoidance, will then tend to

$$\mathrm{P}(\text{avoidance unsuccessful}) = \mathrm{P}_{\mathrm{req}}. \tag{3.17}$$

The fact that we in practice compute $\mathrm{P}(\mathcal{H}_1 \mid \text{avoidance})$ at discrete points in time t_{a} means that the result will never be identical to $\mathrm{P}_{\mathrm{req}}$. Note that (3.17) does not make any distinction between whether the encounter is NMAC or non-NMAC for the original, no avoidance trajectory. The case with a non-NMAC encounter resulting in avoidance unsuccessful is often referred to as an *induced* NMAC. We also know that $\mathrm{P}_{\mathrm{req}}$ will not be exceeded if no avoidance maneuver is executed, i.e. the outcome of encounter simulations will tend to

$$\mathrm{P}(\text{avoidance necessary}) < \mathrm{P}_{\mathrm{req}}, \tag{3.18}$$

This result is explained by the fact that the probability of avoidance unsuccessful and avoidance necessary will approach each other and eventually coincide at or near closest point of approach, see Example 3.2. Any avoidance maneuver becomes less effective the closer the relative position is to CPA. Since an avoidance maneuver is never initiated and the probabilities coincide the relation follows.

Compare the result above with instantaneous probability, or any other method for that matter, where we need to find thresholds P_{th} which yield as a result the required probability because $\mathrm{P}_{\mathrm{th}} = \mathrm{P}_{\mathrm{req}}$. The encounter simulations would now have to include

the verification of the thresholds. Moreover, the thresholds are in general neither easy to find or constant with respect to e.g. tracking performance [72].

Using the probability of NMAC computed for a predicted trajectory the encounter simulations can focus on verifying sensor and tracking filter performance. For example we need camera probability of detection and tracking filter convergence, both as functions of distance. The result must be such that there is enough distance left for at least one avoidance maneuver, with a collision risk below the requirement, to be available. Moreover, camera probability of false alarm and tracking filter performance must be such that maximum level of false alarm rate is not exceeded. For example, the more uncertain the tracking filter is the more nuisance avoidance maneuvers will be executed because the probability of NMAC will be larger for non-NMAC encounters, i.e.

$$\mathrm{P}(\text{avoidance} \mid \mathcal{H}_0) = \mathrm{P}(\text{nuisance}) \tag{3.19}$$

will be larger for a worse tracking performance. As noted earlier this is the argument for waiting as long as possible before initiating an avoidance maneuver. The longer we can wait the more time will the sensor and tracking filter have to detect and converge respectively. To obtain the probability of nuisance maneuvers we typically have to resort to simulations. For an illustration of probability of avoidance unsuccessful and avoidance necessary see Example 3.2.

Example 3.2

Consider two near head-on collision scenarios in two dimensions, see Figure 3.5, with

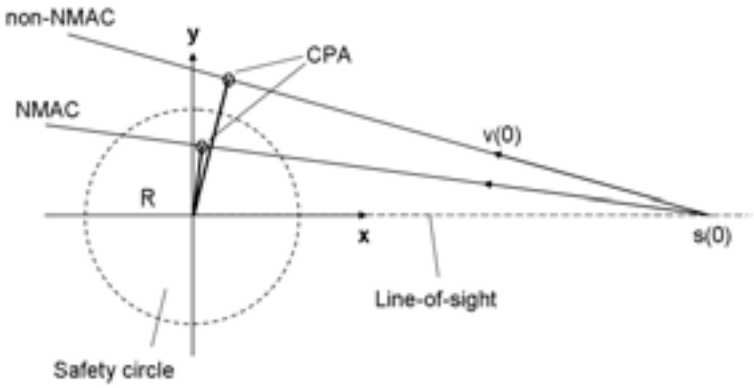

Figure 3.5: *A NMAC and a non-NMAC encounter in a relative coordinate system with the $x-$axis pointing along line-of-sight. The ratio of mean and standard deviation is set to five for distance and closing speed. Standard deviation for distance and speed orthogonal to line-of-sight are both two. Initial distance $s(0)$ is 1500 meters and speed $v(0)$ is 100 m/s.*

$\mathrm{CPA} = 100$ and 200 meters respectively. The two scenarios correspond to a NMAC and non-NMAC encounter respectively. Probability of NMAC with and without avoidance maneuver is shown in Figure 3.6 as a function of absolute time t_{a}. The avoidance maneuver is simplified such that an instantaneous turn of 60 degrees is performed at time

$t_a + 3$. Note that the two curves eventually coincide which is quite intuitive considering the effect of an avoidance maneuver is gradually reduced and eventually it has no effect at all at or near closest point of approach. Here we assume $\mathrm{P_{req}} = 0.05$, which means that $\mathrm{P}(\mathcal{H}_1 \mid \text{avoidance})$ is 0.05. In the left plot we conclude that the $\mathrm{P}(\mathcal{H}_1 \mid \text{no avoidance})$ is approximately $0.98-0.99$, which is obtained from the gap between zero and the no avoidance trajectory curve at the time the avoidance trajectory curve crosses the $\mathrm{P_{req}}-$level. In the right plot the avoidance trajectory curve never exceeds 0.05 and therefore no avoidance maneuver will be executed. This is desirable because probability of avoidance necessary, represented by the no avoidance trajectory curve, tends to a small value below the $\mathrm{P_{req}}-$level. If a smaller $\mathrm{P_{req}}$ is used then probability of avoidance unsuccessful will be smaller. However, we would also obtain a larger probability of avoidance unnecessary if the lower $\mathrm{P_{req}}-$level cuts the avoidance trajectory curve in the non-NMAC scenario. For more information and several examples on the performance of collision avoidance systems see [68, 109].

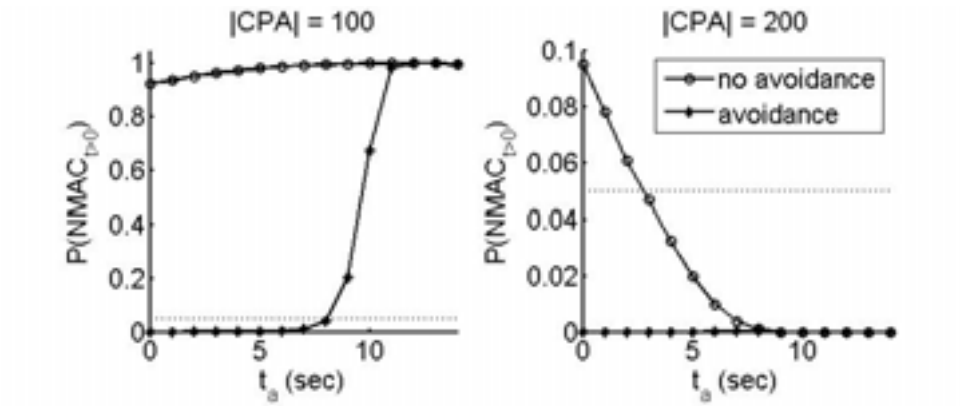

Figure 3.6: *Probability of NMAC as a function of absolute time, no maneuver and avoidance maneuver (instantaneous after 3 sec) respectively. Left plot shows the NMAC encounter and the right the non-NMAC encounter from Figure 3.5.*

3.4 Probability of level-crossings

The problem with the Monte-Carlo method (3.12) is that it requires a too large number of samples, see Paper C. We need to find another method for computing probability of NMAC which is computationally tractable for real-time processing. A way forward is to look upon NMAC as a level-crossing, or down-crossing, of the surface of the safety sphere, see Figure 3.7. Here we use a coordinate system which is rotated such that

$$\hat{s}_y(0) = \hat{s}_z(0) = 0, \tag{3.20}$$

i.e. the $x-$axis is aligned with line-of-sight.

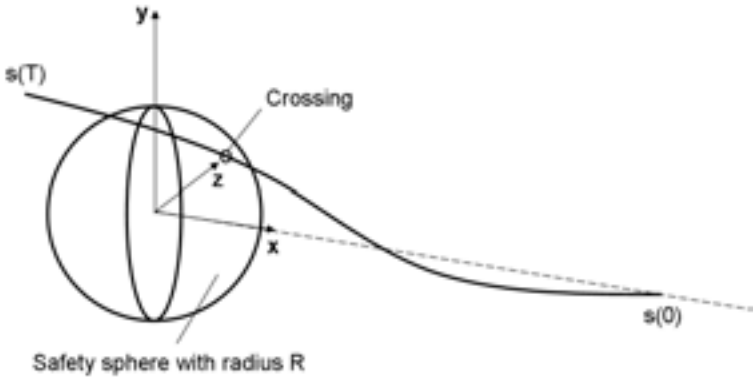

Figure 3.7: *A NMAC occurs if the relative position crosses the surface of the sphere.*

We assume the predicted relative trajectory is deterministic conditionally upon initial intruder state. That is, given the initial state the future trajectory of the intruder is known. Typically we assume the intruder follows a straight path. Here we consider short periods of time, say prediction times up to a minute. For disturbances along the path which are not excessive the assumption on a known future relative trajectory given the initial state should not be too far from the truth. For the own vehicle we have the possibility to use more or less complex motion models for prediction due to known dynamics. It is probably useless though to use too complex models since the true trajectory will deviate more or less from the predicted. The result on short-range conflict detection given in [91] can be used to determine the effect of disturbances and modelling errors. To be able to compute probability of NMAC for a predicted trajectory we will assume that the relative position follows a piecewise straight path. In practice this is obviously not correct, but it is possible to make the error arbitrarily small by making the straight segments short enough.

For computational tractability we need to limit the predicition time t to a finite T, i.e. we consider the event $\text{NMAC}_{(0,T)}$ instead of $\text{NMAC}_{t>0}$. This is no restriction as long as T is larger than the time it takes to perform an avoidance maneuver and some additional time for the effect of the maneuver to become evident in the probability.

3.4.1 One dimension and constant velocity

In one dimension the process describing relative position $s_x(t)$ is confined to lie on a straight line. Assume we wish to compute the probability

$$\mathrm{P}\big(\min_{0<t<T} s_x(t) < 0\big). \tag{3.21}$$

Recall that t represents prediction time, and that (3.21) is also a function of absolute time t_a. This probability can be split using two mutually exclusive events, $s_x(0) < 0$ and $s_x(0) > 0$. At the same time we note that if $s_x(0) < 0$ we automatically have that the

minimum of the process is less than zero, i.e.

$$\begin{aligned}&\mathrm{P}\big(\min_{0<t<T} s_x(t) < 0\big)\\&= \mathrm{P}\big(\min_{0<t<T} s_x(t) < 0 \qquad s_x(0) > 0\big) + \mathrm{P}\big(s_x(0) < 0\big).\end{aligned} \tag{3.22}$$

The first probability in the sum in (3.22) can be seen as a level-crossings problem. The position $s_x(t)$ must cross the level zero since it starts with $s_x(0) > 0$, i.e. a down-crossing. Let us start by considering a constant velocity

$$\dot{s}_x(t) = v_x(t), \quad 0 < t < T, \tag{3.23a}$$

$$v_x(t) = v_x(0), \tag{3.23b}$$

which means that we can reformulate the down-crossings probability according to

$$\mathrm{P}\big(\min_{0<t<T} s_x(0) + v_x(0)t < 0 \qquad s_x(0) > 0\big). \tag{3.24}$$

If a down-crossing is to occur the relative velocity must be negative, otherwise $s_x(t)$ will only increase. Moreover, the minimum of $s_x(0) + v_x(0)t$ with a negative velocity and $0 < t < T$ is attained at $t = T$. We can therefore write the probability as

$$\mathrm{P}\big(s_x(0) + v_x(0)T < 0 \qquad s_x(0) > 0 \qquad v_x(0) < 0\big). \tag{3.25}$$

This event is depicted in Figure 3.8. The ellipse represents the joint probability density function (pdf) for $s_x(0)$ and $v_x(0)$ and is denoted by $p_{s_x(0),v_x(0)}(s, v)$. We see from

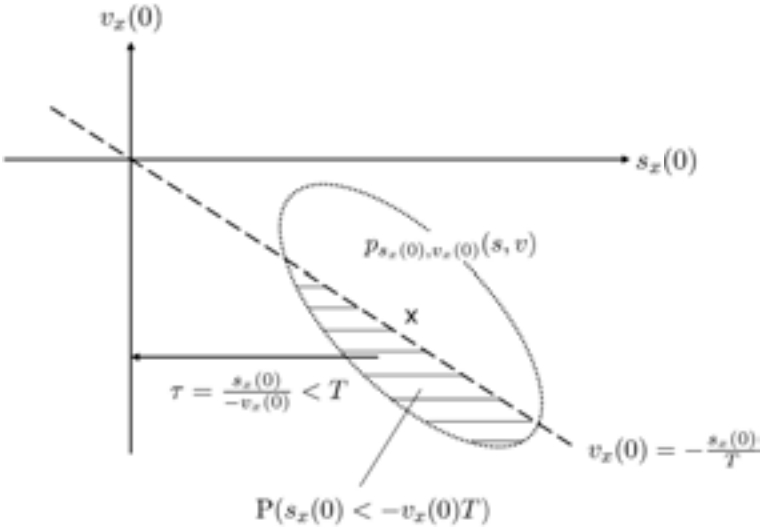

Figure 3.8: *The probability density for $s_x(0)$ and $v_x(0)$ integrated over the area $s_x(0) > 0$ and $v_x(0) < -s_x(0)/T$ yields the probability of $\tau < T$.*

Figure 3.8 that the sought probability is given by the integral of $p_{s_x(0),v_x(0)}(s, v)$ over the

area limited by $s_x(0) > 0$ and $v_x(0) < -s_x(0)/T$. To simplify the expression for the probability we can define a time τ according to

$$\tau = \begin{cases} \frac{s_x}{-v_x} & \text{if } s_x > 0 \quad v_x < 0, \\ & \text{otherwise,} \end{cases} \tag{3.26}$$

which yields the expression

$$\mathrm{P}\big(s_x(0) + v_x(0)T < 0 \quad s_x(0) > 0 \quad v_x(0) < 0\big) = \mathrm{P}(\tau < T). \tag{3.27}$$

The time τ can be interpreted as a time-to-go (ttg), i.e. the time left before $s_x(\tau) = 0$. Based on the distribution for τ it is straightforward to obtain the expression for the probability density function

$$p_\tau(t) = \int_{-\infty}^{0} -v p_{s_x,v_x}(-vt, v)dv, \quad t < \infty. \tag{3.28}$$

See Example 3.3 for the case with Gaussian distributed variables.

Example 3.3: Gaussian distributed $s_x(0)$ and $v_x(0)$

If $s_x(0)$ and $v_x(0)$ are jointly Gaussian distributed,

$$\begin{bmatrix} s_x \\ v_x \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \hat{s}_x \\ \hat{v}_x \end{bmatrix}, \underbrace{\begin{bmatrix} \sigma_{sx}^2 & \rho_x \sigma_{sx} \sigma_{vx} \\ \rho_x \sigma_{sx} \sigma_{vx} & \sigma_{vx}^2 \end{bmatrix}}_{P_{sx}} \right), \tag{3.29}$$

we have

$$\begin{aligned} &\mathrm{P}(\tau < T) \\ &= \frac{1}{2\pi\sigma_{sx}\sigma_{vx}(1-\rho_x)^{1/2}} \int_{-\infty}^{0} \int_{0}^{-vT} e^{-\frac{1}{2}\begin{bmatrix} s-\hat{s}_x \\ v-\hat{v}_x \end{bmatrix}^T P_{sx}^{-1} \begin{bmatrix} s-\hat{s}_x \\ v-\hat{v}_x \end{bmatrix}} ds dv \end{aligned} \tag{3.30}$$

The expression for $p_\tau(t)$ is derived in Appendix A.

Note that it is possible to obtain the expression for $\mathrm{P}(\tau < T)$ through the use of Rice's formula [71]. Rice's formula gives a method for computing the intensity of level-crossings for a stochastic process.

3.4.2 One dimension and piecewise constant velocity

The result for a straight path is possible to extend to a piecewise straight path given by

$$\dot{s}(t) = v^{(j)}, \quad T_j < t < T_{j+1}, \tag{3.31a}$$

$$v^{(j)} = v(0) + \sum_{l=1}^{j} \Delta v^{(l)}, \tag{3.31b}$$

where all $\Delta v^{(j)}$ are known. An important observation is that if it is unlikely that $v_x^{(j)}$ changes sign, i.e.

$$\mathrm{P}\big(\mathrm{sign}(v_x^{(j)}) = \mathrm{sign}(v_x^{(0)})\big) = 1, \tag{3.32}$$

or very close to one, then $\mathrm{NMAC}_{(T_j,T_{j+1})}$ for different j:s are mutually exclusive. With this assumption we can write

$$\begin{aligned}\mathrm{P}(\mathrm{NMAC}_{(0,T)}) &= \mathrm{P}(\ \ {}_{j=0}^{J-1}\mathrm{NMAC}_{(T_j,T_{j+1})}) \\ &= \sum_{j=0}^{J-1} \mathrm{P}(\mathrm{NMAC}_{(T_j,T_{j+1})}).\end{aligned} \tag{3.33}$$

Now we can focus the computation on each segment with constant velocity $v^{(j)}$. By considering what the initial conditions at $t = 0$ would be if the position and velocity are $s(T_j)$ and $v^{(j)}$ at $t = T_j$ we can reuse the results from the case with constant velocity. Denote the new relative position at $t = 0$ by $s^{(j)}$, and we have

$$s^{(j)} = s(T_j) - T_j v^{(j)}. \tag{3.34}$$

An advantage using $s^{(j)}$ instead of $s(T_j)$ comes from the fact that the covariance of $s^{(j)}$ is equal to the covariance of $s(0)$ for all $j = 0, \ldots, J-1$. This is clear from the expression

$$s^{(j)} = s(0) - \sum_{l=1}^{j} T_l \Delta v^{(l)}, \tag{3.35}$$

which is derived in Paper C. The probability of down-crossing is now given by

$$\mathrm{P}(\mathrm{NMAC}_{(0,T)}) = \sum_{j=0}^{J-1} \mathrm{P}(T_j < \tau^{(j)} < T_{j+1}), \tag{3.36}$$

where

$$\tau^{(j)} = \begin{cases} \frac{s_x^{(j)}}{-v_x^{(j)}} & \text{if } s_x^{(j)} > 0 \quad v_x^{(j)} < 0, \\ & \text{otherwise,} \end{cases} \tag{3.37}$$

3.4.3 Two dimensions

In two dimensions the probability of NMAC is

$$\mathrm{P}(\mathrm{NMAC}_{(0,T)}) = \mathrm{P}\big(\min_{0<t<T} \sqrt{s_x^2(t) + s_y^2(t)} < \mathrm{R}\big), \tag{3.38}$$

that is the relative position shall not cross the circle with radius R. A reasonable approximation is to consider the crossing of a line perpendicular to line-of-sight, see Figure 3.9. For notational convenience we place the line at $x = 0$, keeping in mind that we have the possibility to change the outcome by changing the location of the line. The

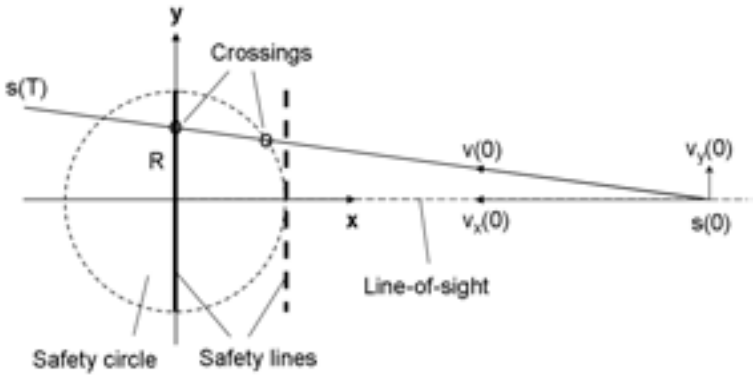

Figure 3.9: *The crossing of the circle is approximated by the crossing of a line.*

only requirement is that the line must be orthogonal to line-of-sight ($x-$axis). The reason is that we can reuse the results from Section 3.4.1. The stochastic time τ provides a measure on time-to-go, i.e. the time left before the relative position crosses the $y-$axis. If the velocity $v_y(0)$ is too small there will be a crossing, too small in the sense that $v_y(0)\tau + s_y(0) < \mathrm{R}$. This means that we can express the probability of NMAC within T seconds according to

$$\mathrm{P}(\widehat{\mathrm{NMAC}}_{(0,T)}) = \mathrm{P}(\ v_y(0)\tau + s_y(0) < \mathrm{R} \quad \tau < T). \tag{3.39}$$

The event $\widehat{\mathrm{NMAC}}_{(0,T)}$ denotes the crossing of the safety line instead of the safety circle, see Figure 3.9. In the case where τ is independent of $s_y(0)$ and $v_y(0)$ it is possible to express the sought probability according to

$$\mathrm{P}(\widehat{\mathrm{NMAC}}_{(0,T)}) = \int_0^T \mathrm{P}(\ v_y(0)t + s_y(0) < \mathrm{R}) p_\tau(t) dt. \tag{3.40}$$

For an illustration see Example 3.4.

Example 3.4: Gaussian distributed state vector in two dimensions

If $s_y(0)$ and $v_y(0)$ are Gaussian distributed

$$\begin{bmatrix} s_y(0) \\ v_y(0) \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \hat{s}_y(0) \\ \hat{v}_y(0) \end{bmatrix}, P_y \right), \tag{3.41}$$

then $v_y(0)t + s_y(0)$ is also Gaussian distributed

$$s_y(0) + v_y(0)t \sim \mathcal{N}\left(\hat{s}_y(0) + \hat{v}_y(0)t, \begin{bmatrix} 1 & t \end{bmatrix} P_y \begin{bmatrix} 1 \\ t \end{bmatrix} \right). \tag{3.42}$$

Moreover, if $s_x(0)$ and $v_x(0)$ are Gaussian distributed as in Example 3.3, then the expression for $p_\tau(t)$ is given by Appendix A. If $s_x(0)$ and $v_x(0)$ are uncorrelated with

$s_y(0)$ and $v_y(0)$ the expression for probability of NMAC according to (3.40) is applicable. Probability of NMAC can then computed by applying Simpson's rule [1] on (3.40).

The extension to piecewise straight paths is equivalent to the one-dimensional case. By using $s^{(j)}$ (3.35) and $v^{(j)}$ (3.31b) we have

$$\mathrm{P}(\widehat{\mathrm{NMAC}}_{(0,T)}) = \prod_{j=0}^{J-1} \mathrm{P}\big(v_y^{(j)}\tau^{(j)} + s_y^{(j)} < \mathrm{R} \quad T_j < \tau^{(j)} < T_{j+1}\big). \tag{3.43}$$

As for a straight path we can simplify the expression in (3.43) for the independent case according to

$$\mathrm{P}(\widehat{\mathrm{NMAC}}_{(0,T)}) = \prod_{j=0}^{J-1} \int_{T_j}^{T_{j+1}} \mathrm{P}\big(v_y^{(j)}t + s_y^{(j)} < \mathrm{R}\big) p_\tau(t)dt. \tag{3.44}$$

For the computation of (3.39) and (3.43) in the general, dependent case see Paper C.

3.4.4 Three dimensions

In three dimensions the probability of NMAC is

$$\mathrm{P}(\mathrm{NMAC}_{(0,T)}) = \mathrm{P}\big(\min_{0<t<T} \sqrt{s_x^2(t) + s_y^2(t) + s_z^2(t)} < \mathrm{R}\big), \tag{3.45}$$

that is the relative position shall not cross a sphere with radius R. The approximation is to consider the crossing of a circular disc perpendicular to line-of-sight, see Figure 3.10. Similar to the two-dimensional case we can express the approximate probability of

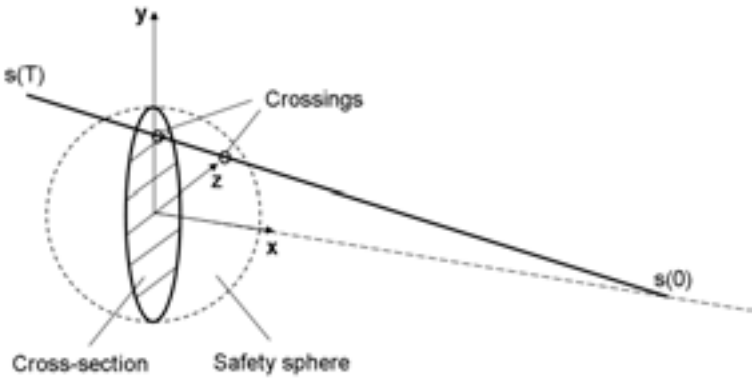

Figure 3.10: *The crossing of the sphere is approximated by the crossing of a disc.*

NMAC within T seconds according to

$$\mathrm{P}(\widehat{\mathrm{NMAC}}_{(0,T)}) = \mathrm{P}(\tau \quad \underbrace{\overline{v_y^2(0) + v_z^2(0)}}_{v_\perp} < \mathrm{R} \quad \tau < T). \tag{3.46}$$

For notational brevity we assume here that $s_y(0) \equiv 0$ and $s_z(0) \equiv 0$. The variance of $s_y(0)$ and $s_y(0)$ are given by [8]

$$\begin{aligned} \sigma_{sy} &= \mathrm{var}\big(s_x(\eta - \hat{\eta})\big) = \hat{s}_x^2 \sigma_\eta^2 + \sigma_{sx}^2 \sigma_\eta^2, \\ \sigma_{sz} &= \mathrm{var}\big(s_x(\epsilon - \hat{\epsilon})\big) = \hat{s}_x^2 \sigma_\epsilon^2 + \sigma_{sx}^2 \sigma_\epsilon^2. \end{aligned} \tag{3.47}$$

For angle-only tracking with an angle measurement error less than 0.5 mrad and an initial distance less than 4000 meters we can neglect $s_y(0)$ and $s_z(0)$ without introducing significant errors. If the simplification is not appropriate it is however straightforward to include $s_y(0)$ and $s_z(0)$, compare with two dimensions in Section 3.4.3 and Paper C. In the case where τ is independent of $v_y(0)$ and $v_z(0)$ the probability is given by

$$\mathrm{P}(\widehat{\mathrm{NMAC}}_{(0,T)}) = \int_0^T \mathrm{P}(v \quad t < \mathrm{R}) p_\tau(t) dt. \tag{3.48}$$

The main difference between two and three dimensions is, for a given $\tau = t$, how to compute the probability of $v \quad$ to be within a circle with radius R/t. If $v_y(0) = v_y$ and $v_z(0) = v_z$ are Gaussian distributed

$$v_y \sim \mathcal{N}(\hat{v}_y, \sigma_y^2), \quad v_z \sim \mathcal{N}(\hat{v}_z, \sigma_z^2) \tag{3.49}$$

we can use the fact that the sum of the squares is possible to express as a weighted sum of two non-central χ^2- variables each with one degree of freedom, denoted by $\chi_1^2(\lambda)$. Here λ is the non-centrality parameter. Consider the simplest case where v_y and v_z are uncorrelated. This is no restriction since the correlated case is handled equivalently [95, 100]. Then the velocities squared and normalized with their respective variances are

$$\begin{aligned} \frac{v_y^2}{\sigma_y^2} &\sim \chi_1^2(\lambda_y), \quad \lambda_y = \frac{\hat{v}_y^2}{\sigma_y^2}, \\ \frac{v_z^2}{\sigma_z^2} &\sim \chi_1^2(\lambda_z), \quad \lambda_z = \frac{\hat{v}_z^2}{\sigma_z^2}. \end{aligned} \tag{3.50}$$

One method to compute the probability of the weighted sum is via the characteristic function, for details see Paper D. Another method is to express the weighted sum as an infinite series of central χ^2- variables [95, 100], denoted by χ_l^2, i.e.

$$\mathrm{P}(v^2 < \frac{\mathrm{R}^2}{t^2}) = \sum_{l=0} c_l \mathrm{P}\big(\xi_l < \frac{\mathrm{R}^2}{\kappa t^2}\big), \quad \xi_l \sim \chi_{2+2l}^2. \tag{3.51}$$

In practice we need to truncate the sum after a finite number of terms L, and the positive scale parameter κ can be used for better convergence. We know from [95] that

$\kappa = \min(\sigma_y^2, \sigma_z^2)$ provides a fast convergence. Moreover, it guarantees a mixture representation, i.e.

$$c_l = 1, \text{ and } c_l \geq 0, \quad l. \tag{3.52}$$

A mixture representation together with the fact that the central χ^2- distribution is a decreasing function with respect to its degree of freedom, $\mathrm{P}\big(\xi_{l+1} < x\big) \leq \mathrm{P}\big(\xi_l < x\big)$, yield an upper bound on the truncation error according to

$$\begin{aligned} &\hat{\mathrm{P}}_L(v^2 < \frac{\mathrm{R}^2}{t^2}) - \mathrm{P}(v^2 < \frac{\mathrm{R}^2}{t^2}) = \\ &c_l \mathrm{P}\big(\xi_l < \frac{\mathrm{R}^2}{\kappa t^2}\big) \leq \\ &\mathrm{P}\big(\xi_L < \frac{\mathrm{R}^2}{\kappa t^2}\big)\big(1 - c_l\big). \end{aligned} \tag{3.53}$$

See Algorithm 3.1 for details on the realization.

Algorithm 3.1 ($\mathrm{P}(v^2 < \frac{\mathrm{R}^2}{t^2})$ as an infinite series).

1. Initialization: Assume $\sigma_z \leq \sigma_y$

$$\begin{aligned} &\kappa = \sigma_z^2 \\ &c_0 = \frac{\kappa}{\sigma_y \sigma_z} e^{-\frac{\lambda_y + \lambda_z}{2}}, \\ &\mathrm{P}\big(\xi_0 < \frac{\mathrm{R}^2}{\kappa t^2}\big) = 1 - e^{-\frac{\mathrm{R}^2}{2\kappa t^2}}, \\ &\hat{\mathrm{P}}_0(v^2 < \frac{\mathrm{R}^2}{t^2}) = c_0 \mathrm{P}\big(\xi_0 < \frac{\mathrm{R}^2}{\kappa t^2}\big). \end{aligned}$$

2. Repeat until $\mathrm{P}\big(\xi_L < \frac{\mathrm{R}^2}{\kappa t^2}\big)\big(1 - \sum_{l=0}^{L} c_l\big) < \varepsilon$

$$\begin{aligned} &c_l = \frac{1}{l} \sum_{r=1}^{l} g_r c_{l-r}, \\ &g_r = \frac{r\lambda_y}{2}(1 - \frac{\kappa}{\sigma_y^2})^{r-1} + \frac{r\lambda_z}{2}(1 - \frac{\kappa}{\sigma_z^2})^{r-1} + \\ &\frac{1 - r\lambda_y}{2}(1 - \frac{\kappa}{\sigma_y^2})^r + \frac{1 - r\lambda_z}{2}(1 - \frac{\kappa}{\sigma_z^2})^r, \\ &\mathrm{P}\big(\xi_l < \frac{\mathrm{R}^2}{\kappa t^2}\big) = \mathrm{P}\big(\xi_{l-1} < \frac{\mathrm{R}^2}{\kappa t^2}\big) - \frac{\mathrm{R}^{2l}}{(2\kappa t^2)^l l!} e^{-\frac{\mathrm{R}^2}{2\kappa t^2}}, \\ &\hat{\mathrm{P}}_l(v^2 < \frac{\mathrm{R}^2}{t^2}) = \hat{\mathrm{P}}_{l-1}(v^2 < \frac{\mathrm{R}^2}{t^2}) + c_l \mathrm{P}\big(\xi_l < \frac{\mathrm{R}^2}{\kappa t^2}\big). \end{aligned}$$

Remark 3.2. Simulations indicate that

$$\kappa = \frac{2\sigma_y^2\sigma_z^2}{\sigma_y^2 + \sigma_z^2} \tag{3.54}$$

yields a faster convergence than $\kappa = \min(\sigma_y^2, \sigma_z^2)$. To be able to quarantee a mixture representation the conditions in (3.52) must be met. We know from [95] that for $_{l=0}\, c_l = 1$ a sufficient requirement is that $\kappa < 2\min(\sigma_y^2, \sigma_z^2)$. Moreover, a sufficient condition for $c_l \geq 0, \quad l$ is that $g_r \geq 0, \quad r$. It is possible to show that this is the case, assuming $\sigma_z \leq \sigma_y$, if

$$\lambda_y \geq \lambda_z \frac{\sigma_y^2}{\sigma_z^2} \qquad \hat{v}_y \;\geq\; \hat{v}_z \;\frac{\sigma_y^2}{\sigma_z^2}. \tag{3.55}$$

Remark 3.3. For large $\lambda_y + \lambda_z$ the number of terms L needs to be large for the truncation error to be small. This can also be deduced from Algorithm 3.1 where $c_0 \approx 0$ for large $\lambda_y + \lambda_z$.

Example 3.5 illustrates how to compute the probability in (3.48) for the Gaussian case.

Example 3.5: Gaussian distributed state vector in three dimensions

If $v_y(0)$ and $v_z(0)$ are Gaussian distributed

$$\begin{bmatrix} v_y(0) \\ v_z(0) \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \hat{v}_y(0) \\ \hat{v}_z(0) \end{bmatrix}, P_{yz} \right), \tag{3.56}$$

we apply a change of variables according to

$$\begin{bmatrix} v_{y'}(0) \\ v_{z'}(0) \end{bmatrix} = U \begin{bmatrix} v_y(0) \\ v_z(0) \end{bmatrix}, \quad P_{yz} = U \begin{bmatrix} \sigma_{y'}^2 & 0 \\ 0 & \sigma_{z'}^2 \end{bmatrix} U^T, \tag{3.57}$$

where U is a unitary matrix. We can then compute $\mathrm{P}(v \;\; t < \mathrm{R})$ using Algorithm 3.1 with mean and variance for $v_{y'}(0)$ and $v_{z'}(0)$ inserted. If $s_x(0)$ and $v_x(0)$ are Gaussian distributed as well the expression for $p_\tau(t)$ is given by Appendix A. The probability $\mathrm{P}(\widehat{\mathrm{NMAC}}_{(0,T)})$ can then computed by applying Simpson's rule [1] on (3.40), assuming τ is independent of v . Note that when applying Simpson's formula we need to compute $\mathrm{P}(v \;\; t_k < \mathrm{R})$ for a number K of discretized time instants t_k. This means that we need to keep both K and L small for a low total computational load.

The extension to piecewise straight paths is equivalent to the one-dimensional case. By using $s^{(j)}$ (3.35) and $v^{(j)}$ (3.31b) we have

$$\begin{aligned} &\mathrm{P}(\widehat{\mathrm{NMAC}}_{(0,T)}) \\ &= \sum_{j=0}^{J-1} \mathrm{P}\Big(\sqrt{(v_y^{(j)}\tau^{(j)} + s_y^{(j)})^2 + (v_z^{(j)}\tau^{(j)} + s_z^{(j)})^2} < \mathrm{R} \quad T_j < \tau^{(j)} < T_{j+1} \Big). \end{aligned} \tag{3.58}$$

The computation of (3.46) and (3.58) are given by Paper D.

4

Concluding Remarks

In this section we provide a summary of the results of this thesis and discuss possible future research.

4.1 Summary

In this thesis we have provided efficient methods for estimation and detection in general, and for terrain-aided navigation and collision avoidance in particular.

The marginalized particle filter, presented in Paper A, utilizes linear substructure for efficient estimation. The part of the state vector which is confined to a linear model, conditionally upon the nonlinear part, is estimated using the Kalman filter. The remaining nonlinear part with lower dimension is estimated by the particle filter. The filter is applied to a simplified terrain-aided navigation in which we assume altitude is known and excellent performance is achieved. The idea of marginalization is extended in Paper B, where we derive a filter which is optimized for the structure inherent in terrain-aided navigation. Here we split the linear substructure in two parts, one unimodal part estimated by the Kalman filter and one multimodal part estimated by the GPB1 filter. The multimodal part corresponds to altitude, which has a separable posterior pdf conditionally upon the nonlinear states. Application of the filter on both simulated and authentic data show excellent performance.

In Paper C we derive the basic theory for efficient conflict detection in two dimensions. The efficiency is obtained by using the concept of level-crossings to describe conflicts. Conflict is declared if the relative position at any point in time crosses a line segment. This makes the computation of probability of conflict computationally tractable as opposed to a sampling based method. The formulation based on level-crossings makes it possible to apply marginalization. For example in the two-dimensional case with independence between variables along and cross line-of-sight we can compute three out of four integrals analytically. The fourth integral is computed numerically using Simpson's formula. The

theory is then extended for near mid-air collision avoidance in three dimensions in Paper D. Here we approximate the crossing of a safety sphere with the crossing of a disc in order to make the idea of level-crossings applicable. Application of the method on simulated data yields promising results.

4.2 Future Research

The marginalized particle filter is, when applicable, a very effective method to decrease the computational load compared to the standard particle filter. There exist theorems which provide measures on what is gained in terms of sampling variance when the marginalized particle filter is applied instead of the particle filter. However, these theorems assume independent samples. This means that resampling is not covered. Intuition and simulations indicate that the marginalized particle filter is even more effective when resampling is introduced. More studies should be conducted to clarify the theory behind this effect and also if any new insights could be used to further increase the effectiveness.

Here we have derived computationally tractable methods for computing the probability of conflict for a predicted trajectory. The method is applicable to three dimensions, uncertain initial conditions and piecewise straight paths. The main restriction concerns potential disturbances and uncertainties along the predicted relative flight trajectory. Typical disturbances are unknown intentions of the intruder and that the vehicles do not follow the assumed path exactly due to wind and simplified aircraft dynamics. The obvious extension is to include the effect of disturbances by combining the results here and the ones given in [91] for short term conflict detection.

The assumption on a normally distributed estimate of the state vector from the tracking filter is usually quite good [72]. This is true at least for the application considered here with short initial distances, around 4000 meters, and high measurement update rate, around 10 Hz. Probably there are cases where the assumption on normality fails. The derived methods for computing probability of conflict are generic in the sense that they do not require the input to be Gaussian distributed. They do however need the probability density function (pdf) for the relative state vector and for the non-Gaussian case this can be difficult to capture. An interesting question is if it is possible to obtain an upper bound on the probability of conflict based on estimated moments, typically mean and covariance, of the state vector instead of the pdf. A potential solution could be convex optimization techniques applied to Chebyshev bounds [17].

The expressions for probability of conflict are rather complicated, especially in the case with dependence between the variables along and cross line-of-sight. It is also difficult to obtain useful bounds on the approximation error, particularly for the geometric approximation but in the general case also for the numerical approximation. The Gram-Charlier expansion, and the similar Edgeworth expansion, constitute methods for expanding a distribution in terms of the normal distribution and its derivatives [54]. It would be interesting to investigate the applicability of these expansions on the probability of conflict and to see if expressions are attainable with bounds on the approximation errors.

A

Derivation of $p_\tau(t)$

Derivation of the expression for $p_\tau(t)$ in the case with normally distributed s_x and v_x.

$$
\begin{aligned}
p_\tau(t) &= \int_{-\infty}^{0} -v p_{s_x,v_x}(-vt, v)\,dv \\
&= \frac{1}{2\pi(\det P_x)^{1/2}} \int_{-\infty}^{0} -v e^{-\frac{1}{2}\begin{bmatrix} -vt-\hat{s}_x \\ v-\hat{v}_x \end{bmatrix}^T P_x^{-1} \begin{bmatrix} -vt-\hat{s}_x \\ v-\hat{v}_x \end{bmatrix}} dv.
\end{aligned}
\tag{A.1}
$$

Diagonalize P_x

$$
\begin{aligned}
P_x &= \begin{bmatrix} 1 & \kappa \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \sigma_{sx}^2(1-\rho_x^2) & 0 \\ 0 & \sigma_{vx}^2 \end{bmatrix} \begin{bmatrix} 1 & \kappa \\ 0 & 1 \end{bmatrix}^T, \\
P_x^{-1} &= \begin{bmatrix} 1 & -\kappa \\ 0 & 1 \end{bmatrix}^T \begin{bmatrix} \frac{1}{\sigma_{sx}^2(1-\rho_x^2)} & 0 \\ 0 & \frac{1}{\sigma_{vx}^2} \end{bmatrix} \begin{bmatrix} 1 & -\kappa \\ 0 & 1 \end{bmatrix},
\end{aligned}
\tag{A.2}
$$

where $\kappa = \rho_x \sigma_{sx}/\sigma_{vx}$. Insert (A.2) into the exponent in (A.1) and write as a polynomial of v

$$
\begin{aligned}
&\begin{bmatrix} -vt-\hat{s}_x-\kappa(v-\hat{v}_x) \\ v-\hat{v}_x \end{bmatrix}^T \begin{bmatrix} \frac{1}{\sigma_{sx}^2(1-\rho_x^2)} & 0 \\ 0 & \frac{1}{\sigma_{vx}^2} \end{bmatrix} \begin{bmatrix} -vt-\hat{s}_x-\kappa(v-\hat{v}_x) \\ v-\hat{v}_x \end{bmatrix} \\
&= \frac{(vt+\hat{s}_x+\kappa(v-\hat{v}_x))^2}{\sigma_{sx}^2(1-\rho_x^2)} + \frac{(v-\hat{v}_x)^2}{\sigma_{vx}^2} \\
&= \left(\frac{(t+\kappa)^2}{\sigma_{sx}^2(1-\rho_x^2)} + \frac{1}{\sigma_{vx}^2}\right)v^2 - 2\left(\frac{(t+\kappa)(\kappa\hat{v}_x-\hat{s}_x)}{\sigma_{sx}^2(1-\rho_x^2)} + \frac{\hat{v}_x}{\sigma_{vx}^2}\right)v \\
&+ \left(\frac{(\kappa\hat{v}_x-\hat{s}_x)^2}{\sigma_{sx}^2(1-\rho_x^2)} + \frac{\hat{v}_x^2}{\sigma_{vx}^2}\right) = g_2^2 v^2 - 2g_1 v + g_0.
\end{aligned}
\tag{A.3}
$$

Complete the square for v

$$g_2^2 v^2 - 2g_1 v + g_0 = (g_2 v - \frac{g_1}{g_2})^2 - \frac{g_1^2}{g_2^2} + g_0, \tag{A.4}$$

and insert into (A.1)

$$\begin{aligned} p_\tau(t) &= \frac{1}{2\pi(\det P_x)^{1/2}} \int_{-\infty}^{0} -v e^{-\frac{1}{2}\left((g_2 v - \frac{g_1}{g_2})^2 - \frac{g_1^2}{g_2^2} + g_0'\right)} dv \\ &= g_0 e^{\frac{g_1^2}{2g_2^2}} \int_{-\infty}^{0} -v e^{-\frac{(g_2 v - \frac{g_1}{g_2})^2}{2}} dv. \end{aligned} \tag{A.5}$$

Change variable to $x = g_2 v - g_1/g_2$

$$\begin{aligned} p_\tau(t) &= \frac{g_0}{g_2^2} e^{\frac{g_1^2}{2g_2^2}} \int_{-\infty}^{-\frac{g_1}{g_2}} -\left(x + \frac{g_1}{g_2}\right) e^{-\frac{x^2}{2}} dv \\ &= \frac{g_0}{g_2^2} e^{\frac{g_1^2}{2g_2^2}} \left(e^{-\frac{g_1^2}{2g_2^2}} - \sqrt{2\pi} \frac{g_1}{g_2} \Phi\left(-\frac{g_1}{g_2}\right) \right). \end{aligned} \tag{A.6}$$

Bibliography

[1] M. Abramowitz and I.A. Stegun, editors. *Handbook of Mathematical Functions With Formulas, Graphs and Mathematical Tables*. US Department of Commerce, 10 edition, 1964.

[2] D. L. Alspach and H. W. Sorenson. Nonlinear Bayesian estimation using Gaussian sum approximations. *IEEE Transactions on Automatic Control*, 17(4):439–448, Aug 1972.

[3] B. D. O. Anderson and J. B. Moore. *Optimal Filtering*. Prentice Hall, Englewood Cliffs, N.J. USA, 1979.

[4] C. Andrieu and A. Doucet. Particle filtering for partially observed Gaussian state space models. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, 64(4), 2002.

[5] C. Andrieu and A. Doucet. Optimal estimation of amplitude and phase modulated signals. Technical Report CUED/F-INFENG/TR395, Department of Engineering, University of Cambridge CB2 1PZ Cambridge, 2000.

[6] P. Angelov, C. Bocaniala, C.D.and Xideas, C. Patchett, D. Ansell, M. Everett, and G. Leng. A passive approach to autonomous collision detection and avoidance in uninhabited aerial systems. In *IEEE Proceedings of the Tenth International Conference on Computer Modeling and Simulation*, 2008.

[7] T Arino, K. Carpenter, S. Chabert, H. Hutchinson, T. Miquel, B. Raynaud, K. Rigotti, and E. Vallauri. ACAS analysis programme ACASA project work package 1 final report on studies on the safety of ACAS in Europe. Technical Report Version 1.3, EEC Brétigny, March 2002.

[8] L.A. Aroian. The probability function of the product of two normally distributed variables. *The Annals of Mathematical Statistics*, 18(2), 1947.

[9] S. Arulampalam and B. Ristic. Comparison of the particle filter with range-parameterised and modified polar ekfs for angle-only tracking. In *Proceedings of SPIE, Signal and Data Processing of Small Target*, 2000.

[10] *F2411-07 Standard Specification for Design and Performance of an Airborne Sense-and-Avoid System*. ASTM, 2007.

[11] B. Azimi-Sadjadi and P. S. Krishnaprasad. Approximate nonlinear filtering and its application in navigation. *Automatica*, 41(6), June 2005.

[12] Y. Bar-Shalom and X-R. Li. *Estimation and Tracking: Principles, Techniques and Software*. Artech House, Boston, London, 1993.

[13] N. Bergman. *Recursive Bayesian Estimation, Navigation and Tracking Applications*. PhD thesis 579, Department of Electrical Engineering, Linköping University, Linköping, Sweden, 1999.

[14] S. Blackman and R. Popoli. *Design and Analysis of Modern Tracking Systems*. Artech House, 1999.

[15] J.H. Blakey. Navigating towards the future: transitioning from terrestrial radio navigation to satellite navigation and airborne surveillance. In *Proceedings of the 24th Digital Avionics Systems Conference*, volume 2, Nov 2005.

[16] H. A. P. Blom and Y. Bar-Shalom. The interacting multiple model algorithm for systems with Markovian switching coefficients. *IEEE Transactions on Automatic Control*, 33(8):780–783, Aug 1988.

[17] S.P. Boyd and L. Vandenberghe. *Convex Optimization*, volume 1. Cambridge University Press, 2004.

[18] R. G. Brown and P. Y. C. Hwang. *Introduction to random signals and applied Kalman filtering*. John Wiley & Sons, 3rd edition, 1992.

[19] J.L. Campbell. *Application of airborne laser scanner - aerial navigation*. PhD thesis, Dept. of Electrical Engineering, Ohio University, 2006.

[20] J. Carpenter, P. Clifford, and P. Fearnhead. Improved particle filter for nonlinear problems. In *IEE Proceedings in Radar, Sonar and Navigation*, volume 146, 1999.

[21] H. Carvalho, P. del Moral, A. Monin, and G. Salut. Optimal nonlinear filtering in GPS/INS integration. *IEEE Transactions on Aerospace and Electronic Systems*, 33 (3):835–849, July 1997.

[22] K. Chan. Analytical expressions for computing spacecraft collision probabilities. In *Proceedings of the 11th Annual AAS/AIAA Space Flight/Mechanics Meeting*, 2001.

[23] K. Chan. Improved analytical expressions for computing spacecraft collision probabilities. *Advances in the Astronautical Sciences*, 114, Part II: Spaceflight Mechanics, 2003.

[24] R. Chen and J. S. Liu. Mixture Kalman filters. *Journal of the Royal Statistical Society*, 62(3):493–508, 2000.

[25] D. Crisan and A. Doucet. A survey of convergence results on particle filtering methods for practitioners. *IEEE Transactions on Signal Processing*, 50(3), March 2002.

[26] F. Daum and J. Huang. Curse of dimensionality and particle filters. In *Proceedings of the IEEE Aerospace Conference*, 2003.

[27] A. Doucet and C. Andrieu. Particle filtering for partially observed Gaussian state space models. Technical Report CUED/F-INFENG/TR393, Department of Engineering, University of Cambridge CB2 1PZ Cambridge, Sep 2000.

[28] A. Doucet, N. J. Gordon, and V. Krishnamurthy. Particle filters for state estimation of jump Markov linear systems. Technical Report CUED/F-INFENG/TR359, Department of Engineering, University of Cambridge CB2 1PZ Cambridge, 1999.

[29] A. Doucet, N. de Freitas, K. Murphy, and S. Russell. Rao-Blackwellised particle filtering for dynamic Bayesian networks. *Uncertainty in Artificial Intelligence*, 2000.

[30] A. Doucet, S. J. Godsill, and C. Andrieu. On sequential simulation-based methods for Bayesian filtering. *Statistics and Computing*, 10(3):197–208, 2000.

[31] A. Doucet, N. de Freitas, and N. Gordon, editors. *Sequential Monte Carlo methods in practice*. Springer-Verlag, 2001.

[32] A. Doucet, N. J. Gordon, and V. Krishnamurthy. Particle filters for state estimation of jump Markov linear systems. *IEEE Transactions on Signal Processing*, 49(3), March 2002.

[33] A.C. Drumm, J.W. Andrews, T.D. Hall, V.M. Heinz, J.K. Kuchar, S.D. Thompson, and J.D. Welch. Remotely piloted vehicles in civil airspace: Requirements and analysis methods for the traffic alert and collision avoidance system (tcas) and see-and-avoid systems. In *Proceedings of the 23rd Digital Avionics Systems Conference*, volume 2, 2004.

[34] T. Erlandsson. Angle-only target tracking. Master's thesis LiTH-ISY-EX-07/3904, Department of Electrical Engineering, Linköping University, Linköping, Sweden, 2007.

[35] *Aeronautical Information Manual - Official Guide to Basic Flight Information and ATC Procedures*. FAA, 2008.

[36] J. Geweke. Bayesian inference in econometric models using Monte Carlo integration. *Econometrica*, 57(6):1317–1339, November 1989.

[37] W. R. Gilks and C. Berzuini. Following a moving target - Monte Carlo inference for dynamic Bayesian models. *Journal of the Royal Statistical Society*, 63(1):127–146, 2001.

[38] N. J. Gordon, D. J. Salmond, and A. F. M. Smith. Novel approach to nonlinear/non-Gaussian Bayesian state estimation. In *IEE Proceedings on Radar and Signal Processing*, volume 140, pages 107–113, 1993.

[39] P.D. Groves, R.J. Handley, and A.R. Runnalls. Optimising the integration of terrain referenced navigation with INS and GPS. *The Journal of Navigation*, 59(1), 2006.

[40] F. Gustafsson. *Adaptive Filtering and Change Detection*. John Wiley & Sons Inc, 1st edition, 2000.

[41] F. Gustafsson, F. Gunnarsson, N. Bergman, U. Forssell, R. Karlsson, and P-J. Nordlund. A framework for particle filtering in positioning, navigation and tracking problems. In *Proceedings of the 11th IEEE Signal Processing Workshop on Statistical Signal Processing*, pages 34–37, Aug 2001.

[42] F. Gustafsson, F. Gunnarsson, N. Bergman, U. Forssell, R. Karlsson, and P-J. Nordlund. Particle filters for positioning, navigation and tracking. *IEEE Transactions on signal processing*, 50(2):425–437, Feb 2002.

[43] F. Gustafsson, T.B. Schön, R. Karlsson, and P-J. Nordlund. State-of-the-art for the marginalized particle filter. In *Proceedings of the IEEE Nonlinear Statistical Signal Processing Workshop*, pages 172–174, Sept 2006.

[44] A. Gut. *An Intermediate Course in Probability*. Springer-Verlag, 1995.

[45] J. E. Handschin. Monte Carlo techniques for prediction and filtering of non-linear stochastic processes. *Automatica*, 6:555–563, 1970.

[46] J. E. Handschin and D. Q. Mayne. Monte Carlo techniques to estimate the conditional expectation in multi-stage non-linear filtering. *International Journal of Control*, 9(5):547–559, 1969.

[47] T. Hektor. Marginalized particle filter for aircraft navigation in 3-D. Master's thesis LiTH-ISY-EX-07/4024, Department of Electrical Engineering, Linköping University, Linköping, Sweden, 2007.

[48] T. Hektor, H. Karlsson, and P-J. Nordlund. A marginalized particle filter approach to an integrated INS/TAP system. In *Proceedings of the IEEE/ION Position, Location and Navigation Symposium*, pages 766–770, May 2008.

[49] G. Hendeby. *Performance and Implementaion Aspects of Nonlinear Filtering*. PhD thesis 1161, Department of Electrical Engineering, Linköping University, Linköping, Sweden, 2008.

[50] X-L. Hu, T.B. Schön, and L. Ljung. A basic convergence result for particle filtering. *IEEE Transactions on Signal Processing*, 56(4), 2008.

[51] J. Jansson and F. Gustafsson. A framework and automotive application of collision avoidance decision making. *Automatica*, 2008.

[52] A. H. Jazwinski. *Stochastic processes and filtering theory*. Academic Press, 1970.

[53] N.L. Johnson, S. Kotz, and N. Balakrishnan. *Continuous Univariate Distributions*, volume 1. John Wiley & Sons, Inc., second edition, 1994.

[54] N.L. Johnson, S. Kotz, and N. Balakrishnan. *Continuous Multivariate Distributions*, volume 1: Models and Applications. John Wiley & Sons, Inc., second edition, 2000.

[55] T. Kailath, A. Sayed, and B. Hassibi. *Linear estimation*. Prentice Hall, 2001.

[56] R. Karlsson. *Particle Filtering for Positioning and Tracking Applications*. PhD thesis 924, Department of Electrical Engineering, Linköping University, Linköping, Sweden, 2005.

[57] S.M. Kay. *Fundamentals of Statistical Signal Processing: Detection Theory*, volume 2. Prentice Hall Inc, 1998.

[58] M. Kayton and W. R. Fried. *Avionics navigation systems*. John Wiley & Sons, 2nd edition, 1997.

[59] T. Kirubarajan and Y. Bar-Shalom. Kalman filter versus imm estimator: When do we need the latter? *IEEE Transactions on Aerospace and Electronic Systems*, 39 (4), 2003.

[60] G. Kitagawa. Monte Carlo filter and smoother for non-Gaussian nonlinear state space models. *Journal of Computational and Graphical Statistics*, 5(1):1–25, 1996.

[61] M. J. Kochenderfer, L.P. Espindle, J. D. Griffith, and J. K. Kuchar. Encounter modeling for sense and avoid development. In *Proceedings of the Integrated Communications, Navigation and Surveillance Conference*, 2008.

[62] M. J. Kochenderfer, J. D. Griffith, and J. K. Kuchar. Hazard alerting using line-of-sight rate. In *Proceedings of the AIAA Guidance, Navigation and Control Conference and Exhibit*, 2008.

[63] A. Kong, J. S. Liu, and W. H. Wong. Sequential imputations and Bayesian missing data problems. *Journal of the American Statistical Association*, 89(524):278–288, March 1994.

[64] S. C. Kramer and H. W. Sorenson. Bayesian parameter estimation. *IEEE Transactions on Automatic Control*, 33(2):217–222, Feb 1988.

[65] S. C. Kramer and H. W. Sorenson. Recursive Bayesian estimation using piece-wise constant approximations. *Automatica*, 24(6):789–801, 1988.

[66] J. Kuchar, J. Andrews, A. Drumm, T. Hall, V. Heinz, and S. Thompson. A safety analysis process for the traffic alert and collision avoidance system (tcas) and see-and-avoid systems on remotely piloted vehicles. In *Proceedings of the AIAA 3rd "Unmanned Unlimited" Technical Conference, Workshop and Exhibit*, September 2004.

[67] J. K. Kuchar and A. C. Drumm. The traffic alert and collision avoidance system. *Lincoln Laboratory Journal*, 16(2), 2007.

[68] J.K. Kuchar. *A unified methodology for the evaluation of hazard alerting systems*. PhD thesis, Massachusetts Inst. of Technology, Dept. of Aeronautics and Astronautics, Cambridge, MA, 1995.

[69] J.K. Kuchar and L.C. Yang. A review of conflict detection and resolution methods. *IEEE Transactions on Intelligent Transportation Systems*, 1(4):179–189, 2000.

[70] A. R. Lacher, D. R. Maroney, and A. D. Zeitlin. Unmanned aircraft collision avoidance - technology assessment and evaluation methods. Technical Report 07-0095, The MITRE Corporation, 2007.

[71] G. Lindgren. Lectures on stationary stochastic processes. http://www.maths.lth.se/matstat/staff/georg/Publications/lecture.pdf, 2004.

[72] F. Lindsten. Angle-only based collision risk assessment for unmanned aerial vehicles. Master's thesis LiTH-ISY-EX-08/4192, Department of Electrical Engineering, Linköping University, Linköping, Sweden, 2008.

[73] J. Liu and M. West. Combined parameter and state estimation in simulation-based filtering. In *Sequential Monte Carlo methods in Practice*. Addison-Wesley, 2001.

[74] J. S. Liu and R. Chen. Sequential Monte Carlo methods for dynamic systems. *Journal of the American Statisticial Association*, 93(443), Sep 1998.

[75] J. McCalmont, J. Utt, and M. Deschenes. Detect and avoid technology demonstration. In *Proceedings of the 1st Unmanned Aerospace Vehicles, Systems, Technologies, and Operations Conference and Workshop*, 2002.

[76] J. McCalmont, J. Utt, and M. Deschenes. Sense and avoid, phase I (man-in-the-loop) advanced technology demonstration. In *Proceedings of Infotech@Aerospace*, 2005.

[77] C. Musso, N. Oudjane, and F. Le Gland. Improving regularised particle filters. In *Sequential Monte Carlo methods in Practice*. Addison-Wesley, 2001.

[78] P-J. Nordlund. *Sequential Monte Carlo Filters and Integrated Navigation*. Licentiate thesis 945, Department of Electrical Engineering, Linköping University, Linköping, Sweden, 2002.

[79] P-J. Nordlund. Recursive state estimation of nonlinear systems with applications to integrated navigation. Technical Report LITH-ISY-R-2321, Department of Electrical Engineering, Linköping University, SE-58183 Linköping, Sweden, Nov 2000.

[80] P-J. Nordlund and F. Gustafsson. The probability of near midair collisions using level-crossings. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 2008.

[81] P-J. Nordlund and F. Gustafsson. Marginalized particle filter for accurate and reliable terrain-aided navigation. *Accepted for publication in IEEE Transactions on Aerospace and Electronic Systems*, 2008. http://www.control.isy.liu.se/research/reports/2008/2870.pdf.

[82] P-J. Nordlund and F. Gustafsson. Probabilistic conflict detection for piecewise straight paths. *Submitted to Automatica*, 2008. http://www.control.isy.liu.se/research/reports/2008/2871.pdf.

[83] P-J. Nordlund and F. Gustafsson. Probabilistic near mid-air collision avoidance. *Submitted to IEEE Transactions on Aerospace and Electronic Systems*, 2008. http://www.control.isy.liu.se/research/reports/2008/2872.pdf.

[84] P-J. Nordlund and F. Gustafsson. Sequential Monte Carlo filtering techniques applied to integrated navigation systems. In *Proceedings of the 2001 American Control Conference*, volume 6, pages 4375–4380, 2001.

[85] P-J. Nordlund and F. Gustafsson. Recursive estimation of three-dimensional aircraft position using terrain-aided positioning. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Proceesing*, volume 2, pages 1121–1124, 2002.

[86] P-J. Nordlund, F. Gunnarsson, and F. Gustafsson. Particle filters for positioning in wireless networks. In *Procedings of EUSIPCO*, Toulouse, France, September 2002.

[87] P. B. Ober. Designing integrity into position estimation. *The Journal of Navigation*, 53(2), 2000.

[88] N. Peach. Bearings-only tracking using a set of range-parameterised extended Kalman filters. In *IEE Proceedings of Control Theory and Applications*, volume 142, 1995.

[89] M. K. Pitt and N. Shephard. Filtering via simulation: auxiliary particle filters. *Journal of the American Statistical Association*, 94:590–599, 1999.

[90] M. Prandini, J. Lygeros, A. Nilim, and S. Sastry. Randomized algorithms for probabilistic aircraft conflict detection. In *Proceedings of the 38th Conference on Decision and Control*, 1999.

[91] M. Prandini, J Hu, J Lygeros, and S. Sastry. A probabilistic approach to aircraft conflict detection. *IEEE Transactions on Intelligent Transportation Systems*, 1(4): 199–220, 2000.

[92] P. Pucar and J. Palmqvist. Saab NINS/NILS - an autonomous landing system for Gripen. In *Proceedings of the Position Location and Navigation Symposium, IEEE 2000*, 2000.

[93] K. Reif, S. Günther, E. Yas, and R. Unbehauen. Stochastic stability of the discrete-time extended Kalman filter. *IEEE Transactions on Automatic Control*, 44(4): 714–728, April 1999.

[94] X. Rong Li and V.P. Jilkov. Survey of maneuvering target tracking, part I, dynamic models. *IEEE Transactions on Aerospace and Electronic Systems*, 39(4):1333–1364, Oct 2003.

[95] H. Ruben. Probability content of regions under spherical normal distributions, iv: the distribution of homogeneous and non-homogeneous quadratic functions of normal variables. *The Annals of Mathematical Statistics*, 33(2), 1962.

[96] R. J. Schaefer. A standards-based approach to sense-and-avoid technology. In *Proceedings of the 3rd AIAA Unmanned Unlimited Technical Conference, Workshop and Exhibition*, 2004.

[97] T. Schön. *Estimation of Nonlinear Dynamic Systems: Theory and Applications*. PhD thesis 998, Department of Electrical Engineering, Linköping University, Linköping, Sweden, 2006.

[98] T.B. Schön, F. Gustafsson, and P-J. Nordlund. Marginalized particle filters for mixed linear/nonlinear state-space models. *IEEE Transactions on Signal Processing*, 53(7):2279–2289, July 2005.

[99] O. Shakernia, W-Z. Chen, and V. M. Raska. Passive ranging for UAV sense and avoid applications. In *Proceedings of Infotech@Aerospace*, 2005.

[100] J. Sheil and I. O'Muircheartaigh. Algorithm as 106: The distribution of non-negative quadratic forms in normal variables. *Applied Statistics*, 26(1), 1977.

[101] A. F. M. Smith and A. E. Gelfand. Bayesian statistics without tears: A sampling-resampling perspective. *The American Statistician*, 46(2), 1992.

[102] H. W. Sorenson and D. L. Alspach. Recursive Bayesian estimation using Gaussian sums. *Automatica*, 7:465–479, 1971.

[103] B-G. Sundqvist. Auto-ACAS - robust nuisance-free collision avoidance. In *Proceedings of the 44th Conference on Decision and Control and 2005 European Control Conference*, December 2005.

[104] *Aeronautical Information Publication (AIP) Sweden*. Swedish Civil Aviation Authority, 2007. http://www.lfv.se/sv/LFV/Flygtrafiktjansten/FPC/IAIP/.

[105] D. Törnqvist. *Estimation and Detection with Applications to Navigation*. PhD thesis 1216, Dept. of Electrical Engineering, Ohio University, 2008.

[106] J. K. Tugnait. Detection and estimation for abruptly changing systems. *Automatica*, 18(5):607–615, 1982.

[107] J. Utt, J. McCalmont, and M. Deschenes. Development of a sense and avoid system. In *Proceedings of Infotech@Aerospace*, 2005.

[108] C. Whelan. Evolution of air navigation services during this decade. *The Journal of Navigation*, 54(3), 2003.

[109] L.C. Yang and J.K. Kuchar. Prototype conflict alerting system for free flight. *Journal of Guidance, Control and Dynamics*, 20(4):768–773, 1997.

[110] A. D. Zeitlin and M. P. McLaughlin. Safety of cooperative collision avoidance for unmanned aircraft. *IEEE Aerospace and Electronic Systems Magazine*, 22(4), 2007.