Multiple model dynamic estimation approach in target tracking and localization

Yang Rong

2012

Yang R. (2012). Multiple model dynamic estimation approach in target tracking and localization. Doctoral thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/48671>

<https://doi.org/10.32657/10356/48671>

**MULTIPLE MODEL DYNAMIC
ESTIMATION APPROACH IN TARGET
TRACKING AND LOCALIZATION**

YANG RONG

School of Electrical & Electronic Engineering

A thesis submitted to the Nanyang Technological University
in partial fulfillment of the requirement for the degree of
Doctor of Philosophy

2012

Acknowledgements

First and foremost, I would like to thank my supervisor Professor Ng Boon Poh for his constant encouragement and guidance during my PhD study. I am grateful to him for offering me an opportunity for PhD research in the target tracking area which is relevant to my work. I benefited from his considerate working style, which allows me to pursue my PhD studies and work concurrently.

I would also like to thank Dr. Ng Gee Wah, who brought me into the target tracking research area and encouraged me to further my PhD studies in this area. He is always approachable and willing to offer valuable advice.

Also I wish to extend my appreciation to Professor Lew Henry, who was my supervisor in my first year studies. Dr. Lew provided me many assistance during my PhD application phase, and recommended me to Dr. Ng Boon Poh when he left Nanyang Technological University.

I also thank my friends Dr. Foo Pek Hui for her careful proof reading for this thesis.

I appreciate the research work environment of my sponsor company, DSO National Laboratories, and the financial support over the years.

Thank to my husband for his understanding, encouragement and support.

Contents

Summary	vii
List of Figures	x
List of Tables	xiv
List of Abbreviations	xv
Chapter 1 Introduction	1
1.1 Motivation	1
1.2 Objectives	4
1.3 Major Contributions of the Thesis	7
1.4 Organization of the Thesis	9
Chapter 2 Technical Fundamentals on Multiple Model Target	
Tracking	12
2.1 Target Kinematic Models	13
2.1.1 Constant Velocity Model	14
2.1.2 Constant Acceleration Model	15
2.1.3 Constant Turn Rate Model	16
2.2 Dynamic Estimation Algorithms	18
2.2.1 Kalman Filter	19
2.2.2 Extended Kalman Filter	20
2.2.3 Unscented Kalman Filter	22

2.2.4	Particle Filter	25
2.3	Multiple Model Approaches	27
2.3.1	Static MM Estimator	27
2.3.2	Dynamic MM Estimator	29
2.4	Cramer-Rao Lower Bound	34
2.5	Summary and Remarks	36
 Chapter 3 An Improved Model for Tracking an Accelerated Target		38
3.1	Acceleration Motion and the Error Generated by the Existing Acceleration Model	39
3.2	Nonlinear Constant Heading Acceleration Model and Estimation Algorithms	46
3.3	Two-stage Least Squares Method for Track Initiation	49
3.4	Simulation and Analysis	54
3.5	Conclusion	61
 Chapter 4 A Pruning Based Multiple Model Filter with Sojourn-Time-Dependent Semi-Markov Switching		62
4.1	Sojourn-Time-Dependent Semi-Markov Processes	65
4.2	A Pruning Based Multiple Model Filter with STD M Switching .	70
4.3	Apply the STD M-PMM Filter on Tracking a Maneuvering Target	79
4.4	Apply the STD M-PMM Filter on Tracking in the Presence of Outliers	88
4.4.1	Related Work on the Outlier Problem	89
4.4.2	STD M-PMM Filter on the Outlier Problem and Simulation Results	91
4.5	Conclusion	96

Chapter 5	MM Application on Target Localization and Antenna Auto-Calibration (I): Spiral Antenna System	98
5.1	Introduction	99
5.2	Spiral Antenna System and Two-stage Estimation Method . . .	104
5.2.1	Spiral Antenna System	104
5.2.2	Two-stage Localization Algorithm	107
5.3	Problem Formulation and Analysis	109
5.3.1	Antenna Bias and Biased Pattern	109
5.3.2	Problem Formulation	110
5.3.3	Cramer-Rao Lower Bound	112
5.3.4	Observability Analysis	113
5.4	Estimation Algorithms	117
5.4.1	Augmented UKF	117
5.4.2	Multiple Model UKF	118
5.5	Simulation and Analysis	120
5.5.1	Scenario	120
5.5.2	Multiple Model Construction	121
5.5.3	Results and Analysis	122
5.6	Conclusion	129
Chapter 6	MM Application on Target Localization and Antenna Auto-Calibration (II): Two-element Array Antenna System	130
6.1	Introduction on Linear Monopole Phased Array System	131
6.2	Problem Formulation and Analysis	133
6.2.1	Perfect Antenna Pattern Case	135
6.2.2	Distorted Antenna Pattern Case	137
6.2.3	Posterior Cramer-Rao Lower Bound	138
6.2.4	Observability Analysis	140

6.3	Dynamic Estimation Algorithms	143
6.3.1	UKF and Constrained UKF	143
6.3.2	Multiple Model UKF	145
6.4	Experimental Results on a Simulated Antenna System	146
6.5	Experimental Results on a Real Antenna System	151
6.6	Conclusion	154
 Chapter 7 MM Application on Indoor Contaminant Source Es-		
	timation	155
7.1	Introduction	156
7.2	Transport Models and Estimation Algorithms in Contaminant Source Estimation Problem	158
7.2.1	Transport Models	159
7.2.2	Estimation Algorithms	162
7.3	Problem Formulation and Estimation Algorithms	166
7.3.1	Single Model Formulation and MCMC Algorithm	166
7.3.2	Multiple Model Formulation and MM-UKF	168
7.4	Simulation Results and Analysis	172
7.4.1	Simulation Scenario	172
7.4.2	Performance of the MM-UKF on Different Sensor Deploy- ments	174
7.4.3	Comparison on the MM-UKF and the MCMC method	179
7.5	Conclusion	182
 Chapter 8 Conclusion and Future Work		183
8.1	Conclusion	183
8.2	Future Work	185
 Author's Publications		187

Bibliography

189

Summary

Target tracking and localization is an interesting research area due to its wide applications in military and civilian systems. Tracking and localization problem is often formulated as a dynamic estimation problem, which is described by a transition model and a measurement model in estimation theory. Since the models in a tracking and localization problem are usually uncertain, a hybrid system with multiple models has been proposed to present such a system. Thus, multiple model estimation approach for the hybrid system has been an intense research topic for several decades.

The objective of this thesis is to conduct further research in Multiple Model (MM) estimation approach. More specifically, we conduct research on model and algorithm enhancement and new application exploration. In model and algorithm enhancement, we focus on model design for accelerated target tracking and Sojourn-Time-Dependent semi-Markov (STDm) switching for dynamic MM estimation. As for application exploration, the MM approach is introduced to solve some difficult problems, which include Radio Frequency (RF) localization using distorted antenna beam patterns and indoor contaminant source localization with limited number of sensors.

Models are of fundamental importance in the MM approach. An inaccurate model in the model set can greatly decrease the performance of an MM algorithm. We suggest a nonlinear constant heading model for tracking a target

in an acceleration motion. The proposed nonlinear model takes the correlation among the Cartesian coordinates into consideration, which is ignored by the existing acceleration models. Simulation tests show that the proposed model outperforms the existing acceleration model.

Markov switching with fixed transition probabilities is normally used in MM approaches. However, it is observed that the model switching probability is related to the model sojourn time. Although there is an STDMM switching MM algorithm represented in the literature, it is too complicated to be implemented in real-time applications. In this thesis, we develop a real-time implementable semi-Markov switching MM filter. The proposed filter is called STDMM-PMM filter, which is a pruning-based MM filter with STDMM switching. The filter is applied to some target tracking problems and better performance is achieved, compared to two existing MM estimators.

Passive localization is a challenging problem when the antenna beam patterns are distorted in the detection system. We propose a Multiple Model Unscented Kalman Filter (MM-UKF) to locate an RF emitter with antenna auto-calibration capability. The new approach formulates the localization problem as a discrete dynamic estimation problem, which consists of an augmented state with target position and antenna bias. Thus, the target position and antenna bias can be estimated simultaneously. Since the bias is unknown to the observer, an MM-UKF is suggested to construct the models with different initial states to handle the antenna distortion uncertainty. We apply the new approach to a spiral antenna system and a two-element array antenna system, and analyze problem observability for these two antenna systems. Simulation and real tests show that the MM-UKF outperforms the conventional approaches.

Indoor contaminant source estimation is an attractive research area to prevent attacks from covert chemical warfare agents. In this application, a building contaminant condition is monitored by a limited number of chemical sensors.

Once contaminant intensity exceeds a normal level, the contaminant source location, the source emission rate and the release time need to be identified rapidly. We introduce an MM-UKF to this application. The models in the system represent the multiple zones in the building, and a UKF is deployed in each model to estimate the source emission rate and the release time, conditioned on the given zone of the model. The simulation results show that the MM-UKF can perform real-time estimation well with limited number of sensors. A comparison is made between the proposed MM-UKF and an existing Markov Chain Monte Carlo (MCMC) estimation method. The MM-UKF obviously outperforms the MCMC method in terms of estimation accuracy and efficiency.

List of Figures

1.1	A generic structure of the multiple model approach in target tracking	2
2.1	Static MM approach	28
2.2	Model sequence hypothesis tree in the dynamic MM approach .	29
2.3	IMM filter with three models	30
3.1	In the constant acceleration motion, the direction of the acceleration vector $\mathbf{a}$ must be consistent with the velocity vector $\mathbf{s}$. The wrong acceleration vector $\mathbf{a}'$ results in the vehicle heading being changed.	41
3.2	Target heading errors on different noise levels and different initial conditions	54
3.3	A test scenario to simulate a surface target taking a constant deceleration maneuvering from a cruising speed 15m/s to a slow speed 2.5m/s in 50s	55
3.4	RMSE on position vs. time when $\sigma_p=7.07\text{m}$, $n=8$ (category I) .	58
3.5	RMSE on position vs. time when $\sigma_p=14.14\text{m}$, $n=6$ (category II)	59
3.6	RMSE on position vs. time when $\sigma_p=14.14\text{m}$, $n=5$ (category III)	59
4.1	A dynamic multiple model system with model sojourn time τ .	65
4.2	Probability mass functions in an STDM system with two models	70
4.3	Transition probability ϕ for each hypothesis in a two-model STDM system	70

4.4	The estimation steps of the STD-M-PMM filter	71
4.5	Hypotheses formation at time k	72
4.6	Hypothesis probability computation.	75
4.7	A generic maneuvering scenario in air traffic control for testing the MM algorithms	80
4.8	Model index vs. scan of the IMM, the M ³ H and the STD-M-PMM filters ($\lambda_T = 0.05$).	84
4.9	The target position estimation accuracy comparisons on the IMM, the M ³ H and the STD-M-PMM filters ($\lambda_T = 0.05$).	84
4.10	The target speed estimation accuracy comparisons on the IMM, the M ³ H and the STD-M-PMM filters ($\lambda_T = 0.05$).	85
4.11	Model index vs. scan of the IMM, the M ³ H and the STD-M-PMM filters ($\lambda_T = 0.1$).	86
4.12	The target position estimation accuracy comparisons on the IMM, the M ³ H and the STD-M-PMM filters ($\lambda_T = 0.1$).	87
4.13	The target speed estimation accuracy comparisons on the IMM, the M ³ H and the STD-M-PMM filters ($\lambda_T = 0.1$).	87
4.14	Target position RMSEs for the IMM, the M ³ H and the STD-M- PMM filters when outlier occurrence rate is 0.05.	94
4.15	Target position RMSEs for the IMM, the M ³ H and the STD-M- PMM filters when outlier occurrence rate is 0.20.	94
5.1	The existing two-stage method and the new approach	102
5.2	Antenna beam patterns in a four-antenna system	105
5.3	Relative antenna amplitude vs. bearing	105
5.4	Relative antenna amplitude in dB vs. bearing	106

5.5	Geometric relationship of the emitter and the platform moving in a constant velocity model. (a) The three adjacent points locate on both sides of $\overline{OH}$. (b) The three adjacent points locate on one side of $\overline{OH}$	115
5.6	The test scenario for computer simulation	120
5.7	Time average of target position RMSE vs. antenna peak bias b_G	124
5.8	Target position RMSE at last cycle (time index 120) vs. antenna peak bias b_G	124
5.9	Target position RMSE vs. time, when the antenna peak bias $b_G = 4dB$	125
5.10	Target position RMSE vs. time, when the antenna peak bias $b_G = 3dB$	126
5.11	Target position RMSE vs. time, when the antenna peak bias $b_G = 0dB$	126
5.12	Model probabilities when $b_G = 4dB$	127
5.13	Model probabilities when $b_G = -1dB$	128
6.1	Two fixed beams in a two-element array antenna system	132
6.2	$\log_{10} det(\mathbf{O}) $ in the surveillance sector when the antenna system moves in path A	142
6.3	$\log_{10} det(\mathbf{O}) $ in the surveillance sector when the antenna system moves in path B	142
6.4	$\log_{10} det(\mathbf{O}) $ in the surveillance sector when the antenna system moves in path C	142
6.5	UKF with Constraints	144
6.6	Test scenario of the simulated antenna system	147
6.7	Time averages of the target position RMSE vs. system distortion parameter d_r	148

6.8	The RMSE of the estimated target position vs. time when the actual distortion level is 0.8	149
6.9	The RMSE of the estimated distortion parameter d_r vs. time when the actual distortion level is 0.8	149
6.10	The model probabilities vs. time in the MM-UKF when the actual distortion level is 0.8	150
6.11	Hardware components of the indoor antenna system	152
6.12	The test scenario of the real antenna system	153
6.13	The RMSE on the estimated target position vs. time in the real antenna system	153
7.1	Contaminant source estimation in a building	157
7.2	Lookup table structure. M is the number of sensors zones, and N is the number of zones in a building. Each zone has at most one sensor, i.e. $M \leq N$	161
7.3	The structure of the MM-UKF with N models	170
7.4	Floor plan for a 3-story building	173
7.5	Likelihoods at different time cycles when the source is located at zone 3	177
7.6	Likelihoods at different time cycles when the source is located at zone 13	178
8.1	Hypothesis tree with association and motion model combination, where CV is the constant velocity model, CA is the constant acceleration model, CT is the constant turn model, and m1 is a measurement in the association gate.	186

List of Tables

2.1	SIR filter	26
2.2	Processing steps in an IMM filter	31
3.1	Algorithm performance on target position accuracy.	57
4.1	The STD-M-PMM filter	79
4.2	Performance comparisons on the IMM, the M ³ H and the STD-M-PMM filters ($\lambda_T = 0.05$)	83
4.3	Performance comparisons on the IMM, the M ³ H and the STD-M-PMM filters ($\lambda_T = 0.1$)	86
4.4	Performance comparisons on the IMM, the M ³ H and the STD-M-PMM filters	95
5.1	Time average of target position RMSE on different antenna peak bias b_G	123
5.2	Target position RMSE at last cycle (time Index 120) on different antenna peak bias b_G	123
7.1	MCMC Bayesian algorithm at time k	164
7.2	MM-UKF at time k	171
7.3	MM-UKF performance on the 6-sensor deployment	176
7.4	MM-UKF performance on the 8-sensor deployment	180
7.5	Performance comparison on the MM-UKF and the single model MCMC method	181

List of Abbreviations

ABBREVIATIONS	FULL EXPRESSIONS
CRLB	Cramer-Rao Lower Bounds
CA	Constant Acceleration
CT	Constant Turn
CV	Constant Velocity
DF	Direction Finding
DOA	Direction of Arrival
EKF	Extended Kalman Filter
IMM	Interacting Multiple Model
KF	Kalman Filter
M ³ H	Multiple Model Multiple Hypothesis
MCMC	Markov Chain Monte Carlo
ML	Maximum Likelihood
MM	Multiple Model
MM-UKF	Multiple Model Unscented Kalman Filter
PF	Particle Filter
RF	Radio Frequency
RMSE	Root-Mean-Square Error
RWR	Radar Warning Receiving

STDm	Sojourn-Time-Dependent semi-Markov
UKF	Unscented Kalman Filter

Chapter 1

Introduction

1.1 Motivation

Target tracking and localization is an essential component in both military and civilian systems which perform surveillance, traffic control, guidance, contaminant source identification, etc. It estimates the target kinematic information from a sequence of noisy measurements which are obtained by various sensors, such as radar, sonar, seismic sensor, satellite, etc. Target tracking can be formulated as a discrete dynamic estimation problem, which is governed by two fundamental models, namely, the transition model and the measurement model. The transition model describes the state transition pattern from the previous time cycle to the current time cycle, and the measurement model converts the state vector to the measurement vector.

A challenging problem in target tracking is that models in the system are often uncertain. For example, a vehicle motion model can be a constant velocity model, an acceleration model, or a turn model, etc. The exact model at each

detection time interval is unknown to the observer. Thus, the state estimation can be treated as a hybrid-state system, and is often solved by a Multiple Model (MM) approach. The generic structure of an MM approach in a hybrid system is shown in Figure 1.1. A **Model Set** consists of a set of predefined models,

Figure 1.1: A generic structure of the multiple model approach in target tracking

which includes all possible models in the system. The MM approach is a recursive process with time. There are usually four steps at each time cycle, the **Hypotheses Formation** step generates the hypotheses based on the models in the model set. A hypothesis is a model sequence with time. The **States Estimation** step updates the individual state conditioned by the model indicated in each hypothesis. To prevent the number of hypotheses from increasing exponentially, the **Hypotheses Merging and Pruning** step maintains the number of hypotheses at an acceptable level. The **Final State Computing** step calculates the final state from a weighted sum of the individual states.

The MM approach has been one of the main research areas in target tracking over the past 40 years. Fruitful achievements in both theoretical and practical studies have been presented in the literature [1] [2] [3] [4] [5] [6] [7] [8] [9] [10] [11] [12] [13] [14]. They include Generalized Pseudo-Bayesian algorithms of order n (GPBn) [9] [10] [11] [12], Interacting Multiple Model (IMM) filter [1] [6], Multiple Model Multiple Hypothesis (M³H) filter [13] [15] [16], Viterbi filter [14],

etc. Among them, the IMM filter, a highly merging-based MM filter, is the most widely used and recommended filter in various applications.

An important component in dynamic MM algorithms is the model switching mechanism. Markov processes with a set of fixed transition probabilities is normally applied in MM filters. However, it is observed that the model switching probability does vary and is related to the system sojourn time in the current model. Therefore a Sojourn-Time-Dependent semi-Markov (STDM) switching system, which has adaptive transition probabilities to sojourn times, is more suitable for a dynamic MM approach. Some STDM-based MM algorithms have been proposed before [7] [17] [18] [19] [20] [21]. Most of them were developed to cope with systems with varying update intervals. Their sojourn time pdfs are limited to exponential distributions [17] [18] [19] [20] [21]. An STDM-based IMM filter with arbitrary sojourn time pdf has been suggested [7]. Since the IMM filter is a highly merging-based MM algorithm, sojourn time on each hypothesis is not explicitly defined. This results in high computational cost in estimating STDM transition probabilities which is unfeasible in real-time applications. The above mentioned existing research work motivates us to put effort on developing a real-time implementable STDM-based MM estimator with arbitrary sojourn time pdf.

Model design is another research focus in MM estimation. Commonly used models in target tracking include Constant Velocity (CV) model, Constant Acceleration (CA) model and Constant Turn (CT) model and their variations [1] [2] [5]. The CV and the CA models are linear models with states which consist of target position, speed or acceleration in x, y and possibly z coordinates. The state elements in different coordinates are assumed uncou-

pled. However, this assumption is not always true, especially for CA motion, when the state elements in different coordinates are correlated by the common target heading. Thus, we attempt to design a more accuracy CA model in this research.

There are some difficult estimation problems without effective or efficient solutions. They include passive localization when antenna beams have unknown distortion, contaminant source estimation in an indoor environment. In this PhD research, we propose suitable MM algorithms to solve these difficult problems.

In summary, this PhD research attempts to enhance the MM estimation approach through better model design and better model switching mechanism. We also attempt to extend the MM approach to new applications to solve difficult problems effectively and efficiently.

1.2 Objectives

The aim of the thesis is to conduct further research on MM algorithms for target tracking and localization. The research focuses on algorithm enhancement and new application exploration. The detailed objectives are:

- To develop enhanced MM algorithms which can improve real-time tracking performance in the following two areas:
 - The first area is to design more accurate models for a dynamic system. Models are of fundamental importance in the MM approach. An inaccurate model in the model set can greatly decrease the per-

formance of an MM algorithm. However, some of the common target motion models which are widely used in vehicle tracking, are designed based on coarse assumptions and approximations. These approximations result in estimation error. We focus on improving the model design in this research.

- The second area is to improve the model switching mechanism. Markov switching with fixed transition probabilities is normally used in the MM approach. However, it is observed that a model switching probability is related to the sojourn time of the model. For example, a vehicle accelerates to a desired speed, and then maintains this speed subsequently. The probability of model switching from the acceleration model to the constant velocity model is increasing with its sojourn time in the acceleration model. Obviously, a fixed model switching/transition probability does not describe the situation properly. Thus, the Sojourn-Time-Dependent semi-Markov (STDM) switching, whose transition probability is a function of sojourn time, is more suitable. Although there are some STDM switching MM algorithms have been presented in the literature, they are too complicated to be implemented in real-time applications. In this research, we aim to develop a real-time implementable semi-Markov switching MM algorithm, and apply the new algorithm to some tracking applications.
- To apply the MM approach to some new applications for improving estimation accuracy and efficiency. The MM approach has been widely used in various estimation applications when model uncertainty exists. It can

be further extended to some new areas where problems are not well solved by existing methods. We introduce the MM approach in the following two areas.

- The first area is passive localization through antenna amplitude comparison Direction Finding (DF) technique. The amplitude comparison DF technique is widely used in various applications due to its simplicity and low cost. However, the accuracy of this DF technique is not high. This is because actual antenna beam patterns are usually biased or distorted from the ideal design, and this unknown distortion can degrade the DF performance significantly. To improve the localization accuracy, the MM approach can be introduced to handle the uncertainty of antenna distortion. We will conduct the research in this area by applying the MM approach to improve the localization accuracy.
- The second area is indoor contaminant source estimation. In this application, a building contaminant condition is monitored by a limited number of chemical sensors. Once contaminant intensity exceeds a normal level, the contaminant source, the source emission rate and the release time need to be identified rapidly. The existing algorithms, such as the Bayesian Markov Chain Monte Carlo (MCMC) estimator, incur high computational cost with slow convergence. The research objective in this area is to improve estimation accuracy and efficiency through the MM approach.

1.3 Major Contributions of the Thesis

According to the objectives defined in Section 1.2, this PhD research makes the following contributions:

- Propose a nonlinear constant heading model for tracking a target in an acceleration motion. The proposed nonlinear model takes the correlation among the Cartesian coordinates into consideration, so that it describes the acceleration motion more accurately than the traditional linear model. The Extended Kalman Filter (EKF) and the Unscented Kalman Filter (UKF) are developed for the proposed nonlinear model. The results show that the nonlinear model outperforms the linear model, and the UKF is better than the EKF in terms of track accuracy and sensitivity to the initial condition.
- Propose a Pruning-based Multiple Model filter with STDMM switching (STDMM-PMM filter) to perform dynamic estimation in an MM hybrid system. The proposed STDMM-PMM filter can be applied on any sojourn time probability mass function (pmf), whereas the commonly used MM filters with fixed model transition probabilities can only work on exponential sojourn time distributions. The recursive hypothesis likelihoods are derived based on STDMM interval-transition probabilities and measurement residuals, and a hypothesis structure updating method is suggested such that the STDMM-PMM filter can be implemented in real time. The STDMM-PMM filter has been tested on two tracking problems, namely, maneuvering target tracking and target tracking in the presence of outliers. The simulation results show that the STDMM-PMM filter outperforms the

IMM filter and the M³H filter in the maneuvering target tracking problem. In the outlier tracking problem, the STDMM-PMM filter outperforms the IMM filter, and provides similar results to the M³H filter.

- Propose an antenna auto-calibration approach to estimate a Radio Frequency (RF) emitter location and antenna distortion simultaneously. The conventional approach for RF localization consists of two separate stages, namely, to estimate bearing through a DF algorithm, and to find the emitter location from the estimated bearings. The drawback of such a structure is that it is difficult to identify antenna beam pattern distortion through the DF algorithm alone, and the bearing error caused by antenna distortion can significantly decrease the localization accuracy, especially when a target is located in the longer range. To overcome this drawback, we merge the two stages together to formulate the localization problem as a discrete dynamic estimation problem. Thus, the target location and the antenna distortion can be estimated simultaneously. An MM-UKF is proposed to solve this problem. We apply the MM-UKF to two types of antenna systems, namely, the spiral antenna system and the two-element array antenna system. Simulation and real tests show that the MM-UKF outperforms the conventional approach.
- Propose a multiple model approach for indoor contaminant source estimation problem. Since this is a nonlinear estimation problem, the Markov Chain Monte Carlo (MCMC) approach has been suggested to estimate the state containing the source location, emission rate and release time. This existing approach encounters high computational cost and slow convergence due to invalid random moves. We suggest an MM approach which

partitions the state into two parts. The first part consists of the source location, and the second part gives the source emission rate and the release time. Since the source location belongs to one of the zones in a building, each zone can be considered as a model in the model set. A multiple model approach is therefore suggested with multiple nonlinear filters conditioned for each zone. Simulation results show that the proposed MM algorithm has significant improvements on estimation accuracy and efficiency.

1.4 Organization of the Thesis

Chapter 2 reviews the technical fundamentals on multiple model target tracking. Firstly, the three main kinematic models, which are widely used in MM maneuvering target tracking, are introduced. They are the constant velocity model, the acceleration model and the turn model. Secondly, the dynamic state estimation algorithms, such as Kalman Filter (KF), Extended KF (EKF), Unscented KF (UKF) and Particle Filter (PF), are described. Thirdly, the research achievements on MM target tracking over the past 40 years are presented. Two classes of MM approaches are introduced, namely, the static MM approach and the dynamic MM approach. The static MM approach is suitable for the applications without model switching throughout the whole process. The dynamic MM approach is widely used in the applications with model switching done dynamically. The representative MM algorithms in these two classes are also reviewed. Fourthly, the Posterior Cramer-Rao Lower Bound (PCRLB), which sets a lower bound on the second order error (error covariance) of any unbiased estimator, is introduced for dynamic estimators. The PCRLB can be used as a benchmark to analyze an estimator's performance. Finally, a summary is given

at the end of the chapter.

Chapter 3 suggests a nonlinear acceleration model which describes the constant acceleration motion of a vehicle. Existing acceleration models ignore the correlation among the Cartesian coordinates, and assume that the state elements on different coordinates are totally independent. This assumption leads to a model error in estimation. The suggested nonlinear model takes target heading into consideration, and describes the acceleration motion precisely. The proposed nonlinear model is implemented by the EKF and the UKF. Simulation tests are conducted to demonstrate the improvement of the proposed model.

Chapter 4 focuses on improving the model switching mechanism of MM approaches. A set of fixed model transition probabilities is normally used in MM approaches. It is observed that model transition probabilities do vary and are related to the model sojourn times. We develop a pruning-based MM filter with STDMM switching. The proposed filter can cope with any sojourn time pmf, and its computational cost does not increase much compared to the existing normal Markov switching MM filters (the IMM filter and the M³H filter). Simulation results show that the proposed filter outperforms the other two MM filters in terms of estimation accuracy.

Chapter 5 and **Chapter 6** apply the MM algorithm to localize an RF emitter when antenna systems are distorted. The conventional two-stage method is reviewed. A new approach is proposed to merge the two stages, namely, the DF and the localization, into one discrete dynamic estimation process, such that the target localization and the antenna auto-calibration can be performed simultaneously. A multiple model UKF is then used to estimate the state for this new approach. Chapter 5 describes the new approach for a spiral antenna

system, and Chapter 6 applies the new approach to a two-element array antenna system. Observability analysis is given for the new approach in both antenna systems. Simulation and real tests are conducted to demonstrate localization accuracy improvement through the MM approach.

Chapter 7 applies the MM approach to estimate the contaminant source information, such as source location, source emission rate and release time, in an indoor environment. The problem is formulated as a hybrid dynamic estimation problem, and is solved by an MM-UKF. The models in the hybrid system represent the multiple zones in a building, and a UKF is deployed in each model to estimate the source emission rate and the release time, conditioned on the given zone of the model. This MM-UKF algorithm is tested in a simulated 3-story building with different sensor deployments. The results show that the MM-UKF can perform real-time estimation well. A comparison is also made between the proposed MM-UKF and an MCMC estimation method suggested before. The MM-UKF obviously outperforms the MCMC method in terms of estimation accuracy and efficiency.

Chapter 8 concludes the PhD research and addresses further research work on MM approaches. The future work includes further developments on the STD-M-PMM filter and the contaminant source estimation algorithm with more complicated emission types.

Chapter 2

Technical Fundamentals on Multiple Model Target Tracking

The Multiple Model (MM) approach has been the focus of interest in target tracking for more than 40 years. It solves the model uncertainty problem in dynamic estimation through a hybrid system consisting of multiple models. This chapter describes the techniques that the MM approaches are involved. They include target motion models which are commonly used for target tracking applications, dynamic estimation algorithms and multiple model approaches. The Posterior Cramer-Rao Lower Bound (PCRLB) is also introduced as a performance analysis tool.

In this chapter, we review three main target motion models, namely, the Constant Velocity (CV) model, the Constant Acceleration (CA) model and the Constant Turn rate (CT) model in Section 2.1. The dynamic estimation algorithms, which include the Kalman Filter (KF), the Extended Kalman Filter (EKF), the Unscented Kalman Filter (UKF) and the Particle Filter (PF) are

introduced in Section 2.2. The two classes of MM approaches, namely, the static MM approach and the dynamic MM approach are then presented in Section 2.3. The PCRLB for dynamic estimators is given in Section 2.4, and summary and remarks are given in Section 2.5.

2.1 Target Kinematic Models

A discrete dynamic system can be described by two fundamental equations, namely, the target state transition equation and the measurement equation. The state transition equation describes the evolution of the target state with time, and the measurement equation relates the state to the measurement. These two equations are represented as

$$\mathbf{x}_k = f_{k-1}(\mathbf{x}_{k-1}, \mathbf{w}_{k-1}), \quad (2.1)$$

$$\mathbf{z}_k = h_k(\mathbf{x}_k, \mathbf{r}_k), \quad (2.2)$$

where k is the time index; $\mathbf{x}_k$ is the target state vector, which normally consists of the target kinematic information; $\mathbf{z}_k$ is the measurement vector; $\mathbf{w}_{k-1}$ is the process noise to compensate the transition model error, and $\mathbf{r}_k$ is the measurement noise; $f_{k-1}(\cdot)$ and $h_k(\cdot)$ are the state transition function and the measurement function respectively. The discrete mathematical model of a target motion is defined by the transition function $f_{k-1}(\cdot)$.

Target motions can be classified into two categories, namely, maneuver and nonmaneuver. A nonmaneuvering motion is a straight motion with a constant velocity. A vehicle moves in the nonmaneuvering motion in most of the time. Once it needs to change its speed or direction, a maneuvering motion, such as

acceleration motion or turn motion, is carried out for a short period. In this section, we review three most popular motion models, namely, the CV model, the CA model and the CT model [1] [2] [5] [22] [23]. For simplicity, all the models are illustrated in 2D space, namely, in x and y Cartesian coordinates.

2.1.1 Constant Velocity Model

The CV model is a linear model describing a target moving at a near constant velocity. The typical CV transition equation is

$$\mathbf{x}_k = \mathbf{F}\mathbf{x}_{k-1} + \mathbf{G}\mathbf{w}_{k-1}, \quad (2.3)$$

where the state $\mathbf{x}$ is $[x \ \dot{x} \ y \ \dot{y}]^T$, the process noise $\mathbf{w}$, which is defined as $[\ddot{x} \ \ddot{y}]^T$, is the mean acceleration covering the uncertainty during the constant velocity motion; $\mathbf{F}$ and $\mathbf{G}$ are the transition matrix and the gain matrix respectively, they are

$$\mathbf{F} = \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{G} = \begin{bmatrix} 0.5T^2 & 0 \\ T & 0 \\ 0 & 0.5T^2 \\ 0 & T \end{bmatrix}, \quad (2.4)$$

.

It is obvious that the target kinematic information in coordinates x and y are decoupled in the CV model. Thus, the 2D CV model can be implemented by two reduced-order filters to estimate $[x \ \dot{x}]^T$ and $[y \ \dot{y}]^T$ separately. The

matrices $\mathbf{F}$ and $\mathbf{G}$ become

$$\mathbf{F} = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix}, \quad \mathbf{G} = \begin{bmatrix} 0.5T^2 \\ T \end{bmatrix}. \quad (2.5)$$

The reduced-order filter can save computational cost.

2.1.2 Constant Acceleration Model

CA transition equation is

$$\mathbf{x}_k = \mathbf{F}\mathbf{x}_{k-1} + \mathbf{w}_{k-1}. \quad (2.6)$$

The state $\mathbf{x}$ is defined as $[x \ \dot{x} \ \ddot{x} \ y \ \dot{y} \ \ddot{y}]^T$. The matrix $\mathbf{F}$ is

$$\mathbf{F} = \begin{bmatrix} \mathbf{F}' & 0 \\ 0 & \mathbf{F}' \end{bmatrix}, \quad (2.7)$$

where

$$\mathbf{F}' = \begin{bmatrix} 1 & T & 0.5T^2 \\ 0 & 1 & T \\ 0 & 0 & 1 \end{bmatrix}, \quad (2.8)$$

T is the time interval.

The CA models have different versions based on the process noise settings. The most popular CA model is referred to as the Wiener process acceleration model. The process noise in this model is the maximum acceleration increment expected over the time interval T . Another CA model is called the white jerk

model, which sets the process noise white Gaussian on the expected target jerk (i.e. acceleration derivative). The jerk model has the same $\mathbf{x}$ and $\mathbf{F}$ as those of the Wiener process acceleration model. The covariance of the process noise is derived from the Singer model when the target maneuver time is much greater than the time interval T . The Singer model is a maneuvering model dealing with both nonmaneuvering and maneuvering motions. The details can be found in the literature [24].

As in the CV model, the kinematic information in x and y coordinates are decoupled in the CA model. However, it is not generally true for target acceleration motion. A natural acceleration motion is that of a constant force acting on a target in the target heading direction. It results in that the target moves in a straight line with constantly changing speed. Thus, the kinematic information in x and y is correlated by a constraint of the common heading. Based on this constraint, an acceleration model with a constant heading is suggested in Chapter 3.

2.1.3 Constant Turn Rate Model

There are two CT models that have been proposed and widely used. They are the fixed turn rate model and the adaptive turn rate model. The transition equation for both models is

$$\mathbf{x}_k = \mathbf{F}\mathbf{x}_{k-1} + \mathbf{w}_{k-1}. \quad (2.9)$$

The state vector $\mathbf{x}$ and the transition matrix $\mathbf{F}$ in the fixed turn rate model are

$$\mathbf{x} = [x \ \dot{x} \ y \ \dot{y}]^T, \quad (2.10)$$

$$\mathbf{F} = \begin{bmatrix} 1 & \frac{\sin \omega T}{\omega} & 0 & -\frac{1 - \cos \omega T}{\omega} \\ 0 & \cos \omega T & 0 & -\sin \omega T \\ 0 & \frac{1 - \cos \omega T}{\omega} & 1 & \frac{\sin \omega T}{\omega} \\ 0 & \sin \omega T & 0 & \cos \omega T \end{bmatrix}, \quad (2.11)$$

where ω is the fixed turn rate, which is assumed to be known from a priori knowledge. The fixed turn rate model is a linear model, and can be implemented by a KF easily. However, the turn rate ω has to be assigned a certain value before filtering. Some simple mechanisms [25] [26] are suggested to estimate the turn rate. More advanced approach is to use the multiple models with all possible fixed turn rates to estimate ω by a Gaussian mixture [27].

The adaptive turn rate model includes the turn rate ω in the state, and the state $\mathbf{x}$ and the transition matrix $\mathbf{F}$ are

$$\mathbf{x} = [x \ \dot{x} \ y \ \dot{y} \ \omega]^T, \quad (2.12)$$

$$\mathbf{F} = \begin{bmatrix} 1 & \frac{\sin \omega T}{\omega} & 0 & -\frac{1 - \cos \omega T}{\omega} & 0 \\ 0 & \cos \omega T & 0 & -\sin \omega T & 0 \\ 0 & \frac{1 - \cos \omega T}{\omega} & 1 & \frac{\sin \omega T}{\omega} & 0 \\ 0 & \sin \omega T & 0 & \cos \omega T & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}. \quad (2.13)$$

The adaptive turn rate model is a nonlinear model. Although the turn rate can be estimated online, the state size is larger than that of the fixed turn rate

model, and it requires a nonlinear algorithm to perform state estimation. A nonlinear estimator can only provide a suboptimal solution.

2.2 Dynamic Estimation Algorithms

A dynamic estimation algorithm performs recursive state estimation from a sequence of noisy measurements in a linear or a nonlinear system. A linear filter, such as the Kalman filter, can provide the optimal solution if the noises in the linear system are white Gaussian. However, the transition model and the measurement model are usually nonlinear functions in most of the tracking applications. This gives rise to the development of nonlinear tracking algorithms. The main challenge of a nonlinear estimation is that the probability density function (pdf) of the posterior state becomes non-Gaussian after nonlinear transformation. This results in difficulty in state statistics estimation. Many suboptimal nonlinear tracking algorithms have been proposed in the literature to approximate the posterior pdf. The most popular one is the EKF [28] [29]. It linearizes a nonlinear function through its Jacobian matrix, so that the state covariance can be transited linearly. If the model nonlinearity is not high, the tracking result of an EKF is acceptable. A more accurate nonlinear tracking algorithm is the UKF. It is a sampling method suggested by Julier et al. [30] [31]. A few sigma points are used to represent the pdf of a state, and these sigma points are propagated through nonlinear functions. The mean and the covariance of the posterior state are then obtained through the transited sigma points. Since derivation of Jacobian matrices is no longer required, it is easier to implement the UKF than the EKF, and the UKF has also shown better estimation accuracy than the EKF in nonlinear estimation applications [32]. Another attractive

nonlinear tracking algorithm is the PF introduced by Gordon et al. [33]. It is a sequential Monte Carlo method. Instead of choosing a few sigma points as done in the UKF, the state pdf is represented by a large number of points in the PF. Not as the EKF and the UKF, which approximate the posterior pdf to a Gaussian, the PF can operate on any pdf. However, the computational cost of the PF is much higher than those of the EKF and the UKF.

We present the KF, EKF, UKF and PF in the following subsections.

2.2.1 Kalman Filter

The KF operates on a linear system

$$\mathbf{x}_k = \mathbf{F}_{k-1}\mathbf{x}_{k-1} + \mathbf{w}_{k-1}, \quad (2.14)$$

$$\mathbf{z}_k = \mathbf{H}_k\mathbf{x}_k + \mathbf{r}_k, \quad (2.15)$$

where $\mathbf{F}_{k-1}$ and $\mathbf{H}_k$ are the linear functions for the transition model and the measurement model respectively, $\mathbf{w}_{k-1}$ and $\mathbf{r}_k$ are the white Gaussian process noise and measurement noise with covariance $\mathbf{Q}_{k-1}$ and $\mathbf{R}_k$ respectively. The noise vectors $\{\mathbf{w}_1, \dots, \mathbf{w}_k, \mathbf{r}_1, \dots, \mathbf{r}_k\}$ are all assumed to be mutually independent.

Given the pdf of the state $\mathbf{x}_{k-1}$, which is Gaussian with mean $\hat{\mathbf{x}}_{k-1}$ and covariance $\mathbf{P}_{k-1}$, the KF performs one time step estimation by the following steps.

Prediction steps:

$$\hat{\mathbf{x}}_{k|k-1} = \mathbf{F}_{k-1} \hat{\mathbf{x}}_{k-1} \quad (2.16)$$

$$\mathbf{P}_{k|k-1} = \mathbf{F}_{k-1} \mathbf{P}_{k-1} \mathbf{F}_{k-1}^T + \mathbf{Q}_{k-1} \quad (2.17)$$

Update steps:

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k \mathbf{v}_k \quad (2.18)$$

$$\mathbf{P}_k = \mathbf{P}_{k|k-1} - \mathbf{K}_k \mathbf{S}_k \mathbf{K}_k^T \quad (2.19)$$

or

$$\mathbf{P}_k = [\mathbf{I} - \mathbf{K}_k \mathbf{H}_k] \mathbf{P}_{k|k-1} \quad (2.20)$$

where

$$\mathbf{v}_k = \mathbf{z}_k - \mathbf{H}_k \hat{\mathbf{x}}_{k|k-1}, \quad (2.21)$$

$$\mathbf{S}_k = \mathbf{H}_k \mathbf{P}_{k|k-1} \mathbf{H}_k^T + \mathbf{R}_k, \quad (2.22)$$

$$\mathbf{K}_k = \mathbf{P}_{k|k-1} \mathbf{H}_k^T \mathbf{S}_k^{-1}, \quad (2.23)$$

and $\mathbf{z}_k$ is the measurement vector at time index k . The KF is an optimal filter of linear Gaussian systems.

2.2.2 Extended Kalman Filter

The EKF is an analytic suboptimal filter for a nonlinear system. It linearizes the nonlinear function locally through the Taylor series expansion. The simplest case is to use the first term of the Taylor series, namely, the Jacobian matrix,

to approximate a nonlinear transformation. Therefore, the EKF requires the nonlinear functions in a system to be continuous such that their Jacobians can be derived. A generic dynamic nonlinear system is described by

$$\mathbf{x}_k = f_{k-1}(\mathbf{x}_{k-1}) + \mathbf{w}_{k-1}, \quad (2.24)$$

$$\mathbf{z}_k = h_k(\mathbf{x}_k) + \mathbf{r}_k, \quad (2.25)$$

where $f_{k-1}(\cdot)$ is the transition function, and $h_k(\cdot)$ is the measurement function, $\mathbf{w}_{k-1}$ and $\mathbf{r}_k$ are the white Gaussian process noise and measurement noise with covariance $\mathbf{Q}_{k-1}$ and $\mathbf{R}_k$ respectively. The Jacobian matrixes of $f_{k-1}(\cdot)$ and $h_k(\cdot)$ can be derived as $\hat{\mathbf{F}}_{k-1}$ and $\hat{\mathbf{H}}_k$ respectively. Given $\hat{\mathbf{x}}_{k-1}$ and $\mathbf{P}_{k-1}$, the recursive estimation of the EKF in one time step done as follows:

Prediction steps:

$$\hat{\mathbf{x}}_{k|k-1} = f_{k-1}(\hat{\mathbf{x}}_{k-1}) \quad (2.26)$$

$$\mathbf{P}_{k|k-1} = \hat{\mathbf{F}}_{k-1} \mathbf{P}_{k-1} \hat{\mathbf{F}}_{k-1}^T + \mathbf{Q}_{k-1} \quad (2.27)$$

Update steps:

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k \mathbf{v}_k \quad (2.28)$$

$$\mathbf{P}_k = \mathbf{P}_{k|k-1} - \mathbf{K}_k \mathbf{S}_k \mathbf{K}_k^T \quad (2.29)$$

or

$$\mathbf{P}_k = [\mathbf{I} - \mathbf{K}_k \mathbf{H}_k] \mathbf{P}_{k|k-1} \quad (2.30)$$

where

$$\mathbf{v}_k = \mathbf{z}_k - h_k(\hat{\mathbf{x}}_{k|k-1}), \quad (2.31)$$

$$\mathbf{S}_k = \hat{\mathbf{H}}_k \mathbf{P}_{k|k-1} \hat{\mathbf{H}}_k^T + \mathbf{R}_k, \quad (2.32)$$

$$\mathbf{K}_k = \mathbf{P}_{k|k-1} \hat{\mathbf{H}}_k^T \mathbf{S}_k^{-1}, \quad (2.33)$$

and $\mathbf{z}_k$ is the measurement vector at time index k .

The estimation result of an EKF is acceptable when system nonlinearity is not high. A high nonlinearity model cannot be approximated by a local linear Jacobian function actually. Thus, some higher order EKFs have been suggested to reduce linearization errors [34]. However, due to the high complexity, these higher order EKFs are not commonly applied in real applications.

2.2.3 Unscented Kalman Filter

The UKF algorithm operates on the basis of the unscented transformation, which uses a set of weighted sigma points to represent the state pdf, and performs nonlinear transformations on each sigma point. The number of sigma points is $2n_{\mathbf{x}} + 1$, where $n_{\mathbf{x}}$ is the number of elements in the state vector. The details of the UKF can be found in the literature [30] [31]. We summarize the UKF estimation steps here.

The first step of the UKF is to represent the state pdf by $2n_{\mathbf{x}} + 1$ sigma points. Given an initial state $\mathbf{x}$ and its covariance $\mathbf{P}$, the i th sigma point χ^i

and its weight W^i are

$$\chi^i = \begin{cases} \mathbf{x}, & \text{for } i = 1 \\ \mathbf{x} + (\sqrt{(n_{\mathbf{x}} + \kappa)\mathbf{P}})_{i-1}, & \text{for } i \in \{2, \dots, n_{\mathbf{x}} + 1\} \\ \mathbf{x} - (\sqrt{(n_{\mathbf{x}} + \kappa)\mathbf{P}})_{i-1-n_{\mathbf{x}}}, & \text{for } i \in \{n_{\mathbf{x}} + 2, \dots, 2n_{\mathbf{x}} + 1\} \end{cases} \quad (2.34)$$

and

$$W^i = \begin{cases} \frac{\kappa}{n_{\mathbf{x}} + \kappa}, & \text{for } i = 1 \\ \frac{1}{2(n_{\mathbf{x}} + \kappa)}, & \text{for } i \in \{2, \dots, 2n_{\mathbf{x}} + 1\} \end{cases} \quad (2.35)$$

where $(\sqrt{(n_{\mathbf{x}} + \kappa)\mathbf{P}})_j$ indicates the j th column of the matrix $(\sqrt{(n_{\mathbf{x}} + \kappa)\mathbf{P}})$, and κ is a scalar to determine the spread of sigma points.

The prediction and update steps of the state approximated by the sigma points are then carried out at each time cycle. They are described as follows.

Prediction steps:

$$\chi_{k|k-1}^i = f_{k-1}(\chi_{k-1}^i) \quad (2.36)$$

$$\hat{\mathbf{x}}_{k|k-1} = \sum_{i=1}^N W^i \chi_{k|k-1}^i \quad (2.37)$$

$$\tilde{\chi}_{k|k-1}^i = \chi_{k|k-1}^i - \hat{\mathbf{x}}_{k|k-1} \quad (2.38)$$

$$\mathbf{P}_{k|k-1} = \mathbf{Q}_k + \sum_{i=1}^N W^i \tilde{\chi}_{k|k-1}^i (\tilde{\chi}_{k|k-1}^i)^T \quad (2.39)$$

Update steps:

$$Z_{k|k-1}^i = h(\chi_{k|k-1}^i) \quad (2.40)$$

$$\mathbf{z}_{k|k-1} = \sum_{i=1}^N W^i Z_{k|k-1}^i \quad (2.41)$$

$$\tilde{Z}_{k|k-1}^i = Z_{k|k-1}^i - \mathbf{z}_{k|k-1} \quad (2.42)$$

$$\mathbf{v}_k = \mathbf{z}_k - \mathbf{z}_{k|k-1} \quad (2.43)$$

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k \mathbf{v}_k \quad (2.44)$$

$$\mathbf{P}_k = \mathbf{P}_{k|k-1} - \mathbf{K}_k \mathbf{S}_k \mathbf{K}_k^T \quad (2.45)$$

where

$$N = 2n_{\mathbf{x}} + 1, \quad (2.46)$$

$$\mathbf{K}_k = \mathbf{P}_{xz} \mathbf{S}_k^{-1}, \quad (2.47)$$

$$\mathbf{S}_k = \mathbf{R}_k + \mathbf{P}_{zz}, \quad (2.48)$$

$$\mathbf{P}_{xz} = \sum_{i=1}^N W^i \tilde{\chi}_{k|k-1}^i (\tilde{Z}_k^i)^T, \quad (2.49)$$

$$\mathbf{P}_{zz} = \sum_{i=1}^N W^i \tilde{Z}_{k|k-1}^i (\tilde{Z}_{k|k-1}^i)^T, \quad (2.50)$$

and $\mathbf{z}_k$ is the measurement vector at time index k .

The UKF described is up to the second order of the Taylor series expansion. A higher order UKF can be implemented through extending the sigma points to more higher order sample points.

2.2.4 Particle Filter

PF appearing in the 90's shows attractive performance in nonlinear and non-Gaussian estimation [33] [35] [36] [37] [38] [39] [40]. The particle filter is a Sequential Monte Carlo (SMC) method, which uses a set of weighted Monte Carlo samplers (or particles) to simulate a non-Gaussian state and its nonlinear varieties. The original SMC algorithm for a nonlinear dynamic system refers to as Sequential Importance Sampling (SIS) filter [33] [37] [38]. The SIS filter incurs a degeneracy problem, which is the variance of sample weights keeps increasing with time [37]. This gave rise to Sampling Importance Resampling (SIR) algorithm to correct skewed weights by re-distributing the samples [33]. The SIR filter attaches a resampling step after the SIS filter, and the steps are shown in Table 2.1.

The particle filter has been further developed recently, and has some varieties, which include the MCMC-based PF [39] [40] and the Rao-Blackwellization PF [37] [41]. The MCMC-based PF was proposed to overcome the impoverishment problem of the SIR. The impoverishment problem is that the diversity of the particles reduces after resampling. It results in all the particles being squeezed to one particular point, especially when process noise is small. The MCMC-based PF uses an MCMC move to replace the resampling step in the SIR. Each particle randomly moves to a new state. The move is accepted if it satisfies certain conditions, otherwise, the state returns to the old position. The MCMC has shown better performance than the SIR when particle impoverishment is severe. The Rao-Blackwellization PF involves the estimation problem with high-dimensional state. A large number of particles is required to achieve acceptable accuracy when the state size is large. The Rao-Blackwellization PF

Table 2.1: SIR filter

1. Perform proposal
for $i = 1 : n$
$\mathbf{x}_k^i \sim q(\mathbf{x}_k^i \mathbf{x}_{k-1}^i, \mathbf{z}_k);$
end;
2. Compute weights (or particle probabilities)
for $i = 1 : n$
$\bar{\mu}_k^i = \mu_{k-1}^i \frac{p(\mathbf{z}_k \mathbf{x}_k^i) p(\mathbf{x}_k^i \mathbf{x}_{k-1}^i)}{q(\mathbf{x}_k^i \mathbf{x}_{k-1}^i, \mathbf{z}_k)},$
$\propto \mu_{k-1}^i p(\mathbf{z}_k \mathbf{x}_k^i);$
end;
normalize: $\mu_k^i = \frac{\bar{\mu}_k^i}{\sum_{i=1}^n \bar{\mu}_k^i};$
3. Resampling
$\hat{N}_{eff} = \frac{1}{\sum_{i=1}^n (\mu_k^i)^2},$
if $\hat{N}_{eff} < N_{thd}$
Removes low weight samples, multiplies high weight
samples, and $w_k^i = 1/N,$
end.

where i is the i th particle, $q(\cdot)$ is the proposal density, and N_{thd} is the predefined threshold.

partitions the state into two parts. Part one consists of the state elements which can be estimated by analytical methods such as Kalman-based estimator. The remaining state elements are categorized into part two, which normally have nonlinear non-Gaussian characteristics, and are solved by the particle approach. In this way the computational effort is devoted to the nonlinear non-Gaussian part, and the conditional Kalman-based estimation provides fast convergence for the other elements of the state.

2.3 Multiple Model Approaches

MM approaches can be categorized into two classes, namely, the static MM approach and the dynamic MM approach. The static MM approach is developed based on the assumption that a model from the model set controls a system throughout the whole process. This MM approach is suitable for target model changing with low frequency. The dynamic MM approach takes dynamic model switching into consideration, and the model switching is governed by a Markov process. The research in the dynamic MM is fruitful. Many successful filters, such as Generalized Pseudo-Bayesian algorithms of order n (GPB $_n$), IMM, M³H, Viterbi, etc, are developed and are widely used in various tracking applications.

2.3.1 Static MM Estimator

The static MM suggested by Magill [42] is to estimate a stochastic process with unknown parameters. The parameters are assumed to be from a finite set, which is obtained from a priori knowledge. Figure 2.1 shows a generic static MM structure with a set of three models. The three models operate independently, and there is no switching among them. $\mathbf{z}$ is the measurement vector. M^i represents the i th model, $\mathbf{x}^i$ is the state estimated by M^i . w^i is the i th weighting coefficient (or likelihood). $\mathbf{x}$ is the final estimated state.

The weighting coefficient w_k^i at time k can be updated by the measurement residual $\mathbf{v}_k^i$ and its covariance $\mathbf{S}_k^i$. The measurement residual $\mathbf{v}_k^i$ is defined as the difference between the actual measurement and the predicted measurement. Both $\mathbf{v}_k^i$ and $\mathbf{S}_k^i$ can be obtained from the filter deployed in M^i , and w_k^i is given

Figure 2.1: Static MM approach

by

$$w_k^i = \frac{1}{c} N(\mathbf{v}_k^i; 0, \mathbf{S}_k^i) w_{k-1}^i, \quad (2.51)$$

where c is a normalizing factor. $N(\cdot)$ represents the Gaussian probability density function with mean 0 and covariance $\mathbf{S}_k^i$.

The final output of the static MM with m models is a Gaussian mixture with mean $\mathbf{x}_k$ and covariance $\mathbf{P}_k$ [29] [43], where

$$\mathbf{x}_k = \sum_{i=1}^m w_k^i \mathbf{x}_k^i, \quad (2.52)$$

$$\mathbf{P}_k = \sum_{i=1}^m w_k^i [\mathbf{P}_k^i + (\mathbf{x}_k^i - \mathbf{x}_k)(\mathbf{x}_k^i - \mathbf{x}_k)^T]. \quad (2.53)$$

The static MM estimator has been shown to have superior performance to the single model approach [44] [45]. Since the static MM approach does not take model switching into consideration, it cannot rapidly respond to model change. The static MM approach is only applicable to applications without model switching, or applications without model switching frequently. For example, ballistic target tracking [46] [47] [48].

2.3.2 Dynamic MM Estimator

A maneuvering target can have a model switch at any time. The models involved in switching and the exact switching time are unknown to the observer. Thus hypotheses that consist of model sequence with time are constructed for all possibilities. Figure 2.2 shows the hypotheses of three models within four time cycles. It is obvious that the number of hypotheses m^k is increased exponentially, where m is the number of models, and k is the number of time cycles.

Figure 2.2: Model sequence hypothesis tree in the dynamic MM approach

Since the number of hypotheses is unbounded when time increases, the optimal solution through investigating all the hypotheses is infeasible. Thus, approximation approaches with bounded hypotheses have to be developed. Hypotheses merging and pruning are two basic methods for cutting down the

computational cost.

2.3.2.1 Merging Based Dynamic MM algorithms

The earlier merging based dynamic MM algorithm is the Generalized Pseudo-Bayesian algorithm of order n (GPB n), which limits the number of time cycle depth to n [9] [10] [11] [12]. The number of hypotheses is reduced to m in GPB1 and m^2 in GPB2 through hypotheses merging, where m is the number of models. A more advanced dynamic MM algorithm is the IMM filter [1] [6], which can achieve a comparative performance of the GPB2 with much lower computational cost.

Figure 2.3 illustrates the hypotheses merging and filtering procedures of the IMM filter with three models. The model switching is governed by a Markov process with model probabilities and model transition probabilities.

Figure 2.3: IMM filter with three models

The processing steps of the IMM filter are summarized in Table 2.2 [3] [4]. Given the estimated state $\hat{\mathbf{x}}_{k-1}^i$, the state error covariance $\mathbf{P}_{k-1}^i$ and the model

Table 2.2: Processing steps in an IMM filter

1. Interaction/Mixing	
Predicted model prob.:	$\mu_{k k-1}^i = \sum_{j=1}^m p_{ji} \mu_{k-1}^j$
Mixing weight:	$\mu_{k-1}^{j i} = \frac{p_{ji} \mu_{k-1}^j}{\mu_{k k-1}^i}$
Mixing state:	$\bar{\mathbf{x}}_{k-1}^i = \sum_{j=1}^m \hat{\mathbf{x}}_{k-1}^j \mu_{k-1}^{j i}$
Mixing covariance:	$\bar{\mathbf{P}}_{k-1}^i = \sum_{j=1}^m \mu_{k-1}^{j i} [\mathbf{P}_{k-1}^j$ $+ (\hat{\mathbf{x}}_{k-1}^j - \bar{\mathbf{x}}_{k-1}^i)$ $\times (\hat{\mathbf{x}}_{k-1}^j - \bar{\mathbf{x}}_{k-1}^i)^T]$
2. Model-matched filtering (KF)	
Predicted state:	$\hat{\mathbf{x}}_{k k-1}^i = \mathbf{F}_{k-1}^i \bar{\mathbf{x}}_{k-1}^i$
Predicted covariance:	$\mathbf{P}_{k k-1}^i = \mathbf{F}_{k-1}^i \bar{\mathbf{P}}_{k-1}^i (\mathbf{F}_{k-1}^i)^T + \mathbf{G}_{k-1}^i \mathbf{Q}_{k-1}^i (\mathbf{G}_{k-1}^i)^T$
Predicted measurement:	$\hat{\mathbf{z}}_k^i = \mathbf{H}_k^i \hat{\mathbf{x}}_{k k-1}^i$
Measurement residual:	$\mathbf{v}_k^i = \mathbf{z}_k - \hat{\mathbf{z}}_k^i$
Residual covariance:	$\mathbf{S}_k^i = \mathbf{H}_k^i \mathbf{P}_{k k-1}^i (\mathbf{H}_k^i)^T + \mathbf{R}_k^i$
Kalman gain:	$\mathbf{K}_k^i = \mathbf{P}_{k k-1}^i (\mathbf{H}_k^i)^T (\mathbf{S}_k^i)^{-1}$
Updated state:	$\hat{\mathbf{x}}_k^i = \hat{\mathbf{x}}_{k k-1}^i + \mathbf{K}_k^i \mathbf{v}_k^i$
Updated covariance:	$\mathbf{P}_k^i = \mathbf{P}_{k k-1}^i - \mathbf{K}_k^i \mathbf{S}_k^i (\mathbf{K}_k^i)^T$
3. Model probability updating	
Updated model prob.:	$\mu_k^i = \frac{\mu_{k k-1}^i \mathcal{N}(\mathbf{v}_k^i; 0, \mathbf{S}_k^i)}{\sum_{j=1}^m \mu_{k k-1}^j \mathcal{N}(\mathbf{v}_k^j; 0, \mathbf{S}_k^j)}$
4. State and covariance combination	
Final state:	$\hat{\mathbf{x}}_k = \sum_{j=1}^m \hat{\mathbf{x}}_k^j \mu_k^j$
Final covariance:	$\mathbf{P}_k = \sum_{j=1}^m \mu_k^j [\mathbf{P}_k^j + (\hat{\mathbf{x}}_k^j - \hat{\mathbf{x}}_k)(\hat{\mathbf{x}}_k^j - \hat{\mathbf{x}}_k)^T]$

probability μ_{k-1}^i at time $k-1$, where $i = 1, \dots, m$, the IMM performs the state estimation in four steps. In the mixing step, the m hypotheses at time $k-1$

expand to m^2 hypotheses based on the Markov process, which is governed by the model probability μ_{k-1}^i and the model transition probability p_{ij} . The m^2 hypotheses are then merged to m hypotheses based on a Gaussian mixture. In the model-matched filtering step, the mixing state for each model is updated in parallel. The updating algorithm can be a KF if the model is a linear model, or an EKF, UKF or PF if the model is a nonlinear model. In Table 2.2, a KF is used for illustration. The model probability is updated based on the measurement residual in the model probability updating step. The individual model probability together with the estimated state and covariance are brought to the next IMM cycle. The final state and its covariance for the current cycle is computed based on a Gaussian mixture in the state and covariance combination step.

The IMM filter is the most successful merging based dynamic MM algorithm, and is widely used in various tracking applications [8] [26] [49] [50] [51] [52]. Many variants of the IMM have been developed [53]. The STDm-based IMM filter [7] extends the Markov switching to the semi-Markov switching. The Interacting Multiple Bias Model (IMBM) algorithm [54], the Adaptive IMM (AIMM) algorithm [55], and the Interacting Acceleration Compensation (IAC) algorithm [56] construct the model set with two-stage bias filter models. The Fuzzy IMM (FIMM) filter utilizes fuzzy logic to determine the model probabilities and model set [57] [58] [59]. The Variable-Structure IMM filter (VS-IMM) is proposed with a dynamic model set [60] [61] [62] [63] [64] [65].

2.3.2.2 Pruning Based Dynamic MM algorithms

The simplest pruning based algorithm maintains a fixed number of hypotheses which have the highest probabilities. This algorithm is referred to as the B-best algorithm [4] or the Multiple Model Pruning (MMP) algorithm [27]. The B-best algorithm is further extended to an adaptive number of hypotheses based on the situation. This algorithm is the so-called Multiple Model Multiple Hypothesis (M³H) filter [13] [15] [16]. Similar to the IMM filter, the probabilities of the hypotheses, μ_k^i , are computed from measurement residuals in the B-best and the M³H algorithms, and the final state is a Gaussian mixture of the states from the remaining hypotheses.

Another pruning based algorithm is the Viterbi algorithm [14], which is also known as the Dynamic Programming Approach (DPA). The Viterbi algorithm intends to find the most possible model sequence recursively. A number of m nodes, which represent the m models, are fixed at each time cycle. A cumulated score $\ln(\mu_k^i)$, which is defined as the logarithm of the probability model sequence, is computed recursively. The model sequence with the highest score is selected as the best DPA path. $\ln(\mu_k^i)$ is computed as [4]:

$$\begin{aligned}
 \ln(\mu_k^i) &= \ln(P\{m_k^i | \mathbf{z}_k\}) \\
 &= \ln\left(\frac{1}{c} P\{\mathbf{z}_k | m_k^i, \mathbf{z}_{k-1}\} P\{m_k^i | m_{k-1}^i, \mathbf{z}_{k-1}\} P\{m_{k-1}^i | \mathbf{z}_{k-1}\}\right) \\
 &= \ln(\mu_{k-1}^i) + \ln(P\{\mathbf{z}_k | m_k^i, \mathbf{z}_{k-1}\}) \\
 &\quad + \ln(P\{m_k^i | m_{k-1}^i, \mathbf{z}_{k-1}\}) - \ln(c).
 \end{aligned} \tag{2.54}$$

where c is a normalizing factor.

A comparison was made between the Viterbi and the IMM [66]. Twelve models were deployed, and each was assigned a specific acceleration value ranging from 0 to 4g ($= 40 \text{ m/s}^2$). The results showed that the tracking accuracy depended on the parameter η , which was proportional to the maximum difference between the acceleration levels of any two models. Both the algorithms produced comparable accuracy when η was large. The Viterbi performed better than the IMM when η was small. However, the IMM had a faster response to target model switching, and produced less tracking error in the switching period.

2.4 Cramer-Rao Lower Bound

This section describes the CRLB for dynamic estimators. The CRLB sets a lower bound on the second order error (error covariance) of any unbiased estimator. Since it provides a benchmark on an estimator's performance, the CRLB becomes a generic tool to analyse an estimator's performance.

Let $\hat{\mathbf{X}}_k = \{ \hat{\mathbf{x}}_0, \hat{\mathbf{x}}_1, \dots, \hat{\mathbf{x}}_k \}$ be the estimated state sequence from the measurement sequence $\mathbf{Z}_k = \{ \mathbf{z}_0, \mathbf{z}_1, \dots, \mathbf{z}_k \}$, and the true state sequence is $\mathbf{X}_k = \{ \mathbf{x}_0, \mathbf{x}_1, \dots, \mathbf{x}_k \}$. The CRLB, which is also referred to as the Cramer-Rao inequality, is

$$\mathbb{E} \left[(\hat{\mathbf{X}}_k - \mathbf{X}_k)(\hat{\mathbf{X}}_k - \mathbf{X}_k)^T \right] \geq \mathbf{I}_k^{-1}, \quad (2.55)$$

where $\mathbf{I}_k$ is the state information matrix, and it is defined as

$$\mathbf{I}_k = \mathbb{E} \left\{ [\nabla_{\mathbf{X}_k} \log p(\mathbf{X}_k, \mathbf{Z}_k)] [\nabla_{\mathbf{X}_k} \log p(\mathbf{X}_k, \mathbf{Z}_k)]^T \right\}, \quad (2.56)$$

and

$$\nabla_{\mathbf{x}_k} = \left[\frac{\partial}{\partial \mathbf{x}_1} \quad \frac{\partial}{\partial \mathbf{x}_2} \quad \cdots \quad \frac{\partial}{\partial \mathbf{x}_k} \right]^T. \quad (2.57)$$

It can be seen that the size of $\mathbf{I}_k$ increases with time. To avoid manipulating large matrices, Tichavsky [67] developed a recursive method to compute the Posterior CRLB (PCRLB) for discrete dynamic system. Let $\mathbf{J}_k$ be the information matrix for estimating state vector $\mathbf{x}_k$. The recursion is

$$\mathbf{J}_k = \mathbf{D}_{k-1}^{22} - \mathbf{D}_{k-1}^{21}(\mathbf{J}_{k-1} + \mathbf{D}_{k-1}^{11})^{-1}\mathbf{D}_{k-1}^{12}, \quad (2.58)$$

where

$$\mathbf{D}_{k-1}^{11} = \mathbb{E} \left\{ [\nabla_{\mathbf{x}_{k-1}} \log p(\mathbf{x}_k | \mathbf{x}_{k-1})][\nabla_{\mathbf{x}_{k-1}} \log p(\mathbf{x}_k | \mathbf{x}_{k-1})]^T \right\}, \quad (2.59)$$

$$\mathbf{D}_{k-1}^{12} = \mathbb{E} \left\{ [\nabla_{\mathbf{x}_{k-1}} \log p(\mathbf{x}_k | \mathbf{x}_{k-1})][\nabla_{\mathbf{x}_k} \log p(\mathbf{x}_k | \mathbf{x}_{k-1})]^T \right\}, \quad (2.60)$$

$$\mathbf{D}_{k-1}^{21} = \mathbf{D}_{k-1}^{12}, \quad (2.61)$$

$$\mathbf{D}_{k-1}^{22} = \mathbb{E} \left\{ [\nabla_{\mathbf{x}_k} \log p(\mathbf{x}_k | \mathbf{x}_{k-1})][\nabla_{\mathbf{x}_k} \log p(\mathbf{x}_k | \mathbf{x}_{k-1})]^T \right\} \quad (2.62)$$

$$+ \mathbb{E} \left\{ [\nabla_{\mathbf{x}_k} \log p(\mathbf{z}_k | \mathbf{x}_k)][\nabla_{\mathbf{x}_k} \log p(\mathbf{z}_k | \mathbf{x}_k)]^T \right\}. \quad (2.63)$$

Consider a generic nonlinear system defined by

$$\mathbf{x}_k = f_{k-1}(\mathbf{x}_{k-1}) + \mathbf{w}_{k-1}, \quad (2.64)$$

$$\mathbf{z}_k = h_k(\mathbf{x}_k) + \mathbf{r}_k, \quad (2.65)$$

where $\mathbf{w}_{k-1}$ is the white Gaussian process noise with covariance $\mathbf{Q}_{k-1}$, and $\mathbf{r}_k$ is the white Gaussian measurement noise with covariance $\mathbf{R}_k$. Let $\mathbf{F}_{k-1}$ be the Jacobian matrix of $f_{k-1}(\mathbf{x}_{k-1})$, and $\mathbf{H}_k$ be the Jacobian matrix of $h_k(\mathbf{x}_k)$.

Equation (2.58) can be expressed by

$$\mathbf{J}_k = (\mathbf{F}_{k-1} \mathbf{J}_{k-1}^{-1} \mathbf{F}_{k-1}^T + \mathbf{Q}_{k-1})^{-1} + \mathbf{H}_k^T \mathbf{R}_k^{-1} \mathbf{H}_k. \quad (2.66)$$

The process noise can be considered as $\mathbf{0}$ if we assume the transition model is purely deterministic. The recursive information matrix $\mathbf{J}_k$ can be further simplified as

$$\mathbf{J}_k = (\mathbf{F}_{k-1} \mathbf{J}_{k-1}^{-1} \mathbf{F}_{k-1}^T)^{-1} + \mathbf{H}_k^T \mathbf{R}_k^{-1} \mathbf{H}_k. \quad (2.67)$$

It can be easily derived that the initial value of the recursive information matrix $\mathbf{J}_0$ is $\mathbf{P}_0^{-1}$, which is the initial state error covariance.

The recursive information matrix (2.67) is similar to the computation of predicted state error covariance in the EKF. The only difference is the Jacobian matrixes $\mathbf{F}_{k-1}$ and $\mathbf{H}_k$ are calculated based on the true state in (2.67), whereas the predicted state is used in the EKF.

The CRLB only provides a lower bound for the second order of a posteriori probability of an estimated state. In general, the posterior probability of a nonlinear estimator is not a Gaussian distribution. Thus, CRLB only evaluates the performance of a nonlinear estimator partially. The higher-order moments are not addressed. However, since we lack better generic solutions in the literature, the CRLB is still widely used in performance analysis of nonlinear estimators.

2.5 Summary and Remarks

This chapter gives a brief description on the technical fundamentals of the MM target tracking approach. It includes the main kinematic models, the dynamic

estimation algorithms, the MM estimation algorithms and the CRLB performance analysis tool.

The MM approach provides an attractive solution for target tracking. It outperforms the single model approach. Through the review, we realize that MM models and algorithms can be further enhanced. As mentioned in Section 2.1.2, the existing CA models de-couple the information in different coordinates, such that the correlation among coordinates is totally ignored. This modeling method leads to estimation error. In this PhD research, we will design a more accurate model to describe the CA motion. Another area we intend to enhance is model switching mechanism. The Markov process with fixed transition probabilities is normally applied in model switching. It only addresses the model sojourn time with an exponential pdf. For an arbitrary sojourn time pdf, a sojourn time dependent Markov (STDm) switching is more suitable. Although the STDm-based IMM filter has been introduced before, it cannot be applied to real applications due to its unbounded computational cost. One of the focuses in this PhD research is to develop a real-time implementable STDm-based MM algorithm.

Chapter 3

An Improved Model for Tracking an Accelerated Target

This chapter proposes a nonlinear model to track a maneuvering target in acceleration motion. The acceleration motion is traditionally modeled by a linear function with the state which consists of target position, speed and acceleration in x , y and possibly z coordinates. The state elements in different coordinates are assumed uncoupled. However, this assumption is not generally true, as the state elements in different coordinates are correlated by the common target heading. Thus, a nonlinear constant heading model is suggested in this chapter. To implement this nonlinear model, a two-stage least squares method is developed for track initiation, and an extended Kalman filter (EKF) and an unscented Kalman filter (UKF) are proposed to estimate the state in track maintenance. Performance of the nonlinear model is demonstrated through simulation data, and results show that the proposed nonlinear model outperforms the traditional linear model.

The organization of the chapter is as follows. Section 3.1 introduces the acceleration motion, and indicates the error created by the existing CA model. Section 3.2 proposes a nonlinear constant heading model for tracking an accelerated target. The EKF and the UKF are suggested for the nonlinear dynamic estimation. The Jacobian matrix of the nonlinear model is also derived for implementing the EKF. Section 3.3 develops a two-stage least squares algorithm to initiate the nonlinear track. Section 3.4 presents simulation results on the proposed nonlinear model and the existing linear model. CRLB of the problem is also presented for performance benchmarking. Section 3.5 gives the conclusion.

3.1 Acceleration Motion and the Error Generated by the Existing Acceleration Model

The correctly defined models, which describe various target motion patterns, are the fundamental for the MM tracking system. The main motion patterns for a vehicle include constant velocity motion, constant acceleration motion and turn motion. In this chapter, we focus on the mathematical model of constant acceleration motion.

The acceleration motion pattern has been modeled in many forms over the past four decades [2] [5] [29]. A popular acceleration maneuvering model was designed by Singer [24]. The target acceleration is modeled as a first-order Markov process with the state $[x \ \dot{x} \ \ddot{x}]^T$, which represents the target position, speed and acceleration in the x coordinate. The Singer's model needs a priori knowledge on target maneuver time τ and the model of acceleration probability density to determine its transition equation. This model is a mixture of

acceleration model and constant velocity model rather than a dedicated acceleration model. The acceleration can also be simply modeled as white Gaussian noise in the constant velocity (CV) model [29]. The state vector is $[x \ \dot{x}]^T$, and the acceleration $\ddot{x}$ is totally merged into the process noise in the CV model. It is equivalent to the Singer's model when τ is sufficiently small. More precisely, the acceleration can be modeled by the constant acceleration (CA) model, which includes $\ddot{x}$ in the state vector as $[x \ \dot{x} \ \ddot{x}]^T$ [29]. The most popular CA model is referred to as the Wiener process acceleration model. The process noise in this model is the maximum acceleration increment expected over the scan interval T . Another version of CA model is called the white jerk model. The white jerk model sets the process noise as white Gaussian on the expected target jerk (i.e. acceleration derivative). It can also be viewed as a special case of Singer model when τ is much greater than the scan interval.

All the models mentioned above are linear models, and the states are only represented in one coordinate. For multiple coordinates, the states are simply augmented, for example, the state becomes $[x \ \dot{x} \ \ddot{x} \ y \ \dot{y} \ \ddot{y}]^T$ in the 2D CA model. Although the augmentation makes the models being implemented easily, it is too coarse that the correlation among coordinates is totally ignored and leads to model error.

Some models in target tracking applications couple information in the x-y coordinates through some common parameters [1] [5] [68] [69]. The coordinated turn models compute state transition through target turn rate when a target turns with a constant rate and a constant speed [1] [5]. In ground target tracking applications, a constant speed target moving alone a road is very common. A preferred target heading can be obtained from the road map in such a motion.

Thus, heading information is often used in model design. The direction process noise model couples the x-y coordinates by process noise, which is computed from the preferred direction/heading [1] [68]. Kastella and Kreucher suggested a more precise nonlinear model by including heading in its state vector [69]. Its state is $[x \ y \ h \ v]^T$, where h and v are target heading and speed respectively, and they couple the x-y coordinates effectively. Although these models are not designed for acceleration motion, their coordinates coupling methods can be utilized to propose new CA models.

An acceleration motion for most vehicles is of that a constant force acting on the vehicle body at the heading direction resulting that the vehicle moves in a straight line with constantly changing speed. Figure 3.1 illustrates this constant

Figure 3.1: In the constant acceleration motion, the direction of the acceleration vector $\mathbf{a}$ must be consistent with the velocity vector $\mathbf{s}$. The wrong acceleration vector $\mathbf{a}'$ results in the vehicle heading being changed.

acceleration motion. The vehicle velocity vector $\mathbf{s}$ and the acceleration vector $\mathbf{a}$ must be consistent (in the same or opposite direction), such that the vehicle can perform an acceleration or deceleration maneuvering without heading change. If the vector $\mathbf{a}'$ (see Figure 3.1), which acts on the vehicle, is not in line with the vector $\mathbf{s}$, the vehicle will change its direction. This is not the constant acceleration motion we discussed. Thus, a constraint on the CA motion is

$$\frac{\mathbf{a}}{|\mathbf{a}|} = \pm \frac{\mathbf{s}}{|\mathbf{s}|}. \quad (3.1)$$

In 2D space, (3.1) can be written as

$$\frac{\dot{x}}{\dot{y}} = \frac{\ddot{x}}{\ddot{y}}, \quad (3.2)$$

where $\dot{x}, \dot{y}, \ddot{x}, \ddot{y}$ are the target speeds and accelerations in x and y coordinates respectively.

However, the commonly used existing acceleration models are all based on the assumption that the coupling among the coordinates can be neglected for simplicity, and the state elements between coordinates are independent. This assumption does not guarantee that an accelerating target moves in a straight line. Thus a model error is introduced and leads to a decrease in track accuracy.

Now we use the uncoupled CA model in 2D space to illustrate its model error. The state vector $\mathbf{x}$ of such a model is defined as

$$\mathbf{x} = [x \ \dot{x} \ \ddot{x} \ y \ \dot{y} \ \ddot{y}]^T, \quad (3.3)$$

where x and y are the target positions, $\dot{x}$ and $\dot{y}$ are the target speeds, $\ddot{x}$ and $\ddot{y}$ are the target accelerations, in x and y coordinates respectively. The transition and measurement equations are

$$\mathbf{x}_k = \mathbf{F}_1 \mathbf{x}_{k-1} + \mathbf{w}_1, \quad (3.4)$$

$$\mathbf{z}_k = \mathbf{H}_1 \mathbf{x}_k + \mathbf{r}, \quad (3.5)$$

where

$$\mathbf{F}_1 = \begin{bmatrix} 1 & T & 0.5T^2 & 0 & 0 & 0 \\ 0 & 1 & T & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & T & 0.5T^2 \\ 0 & 0 & 0 & 0 & 1 & T \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad (3.6)$$

$$\mathbf{H}_1 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}, \quad (3.7)$$

T is the time interval; k is the time index; $\mathbf{w}_1$ is the process noise to compensate the model error, and it is assumed to be the white Gaussian with covariance $\mathbf{Q}_1$; $\mathbf{r}$ is the white Gaussian measurement noise with covariance $\mathbf{R}$.

The state mean $\hat{\mathbf{x}}_k$ and its error covariance $\mathbf{P}_k$ are estimated by the Kalman filter, and they are

$$\hat{\mathbf{x}}_k = \mathbf{F}_1 \hat{\mathbf{x}}_{k-1} + \mathbf{K}_k \mathbf{v}_k, \quad (3.8)$$

$$\mathbf{P}_k = [\mathbf{I} - \mathbf{K}_k \mathbf{H}_1] \mathbf{P}_{k|k-1}, \quad (3.9)$$

where

$$\mathbf{v}_k = \mathbf{z}_k - \mathbf{H}_1 \mathbf{F}_1 \hat{\mathbf{x}}_{k-1}, \quad (3.10)$$

$$\mathbf{P}_{k|k-1} = \mathbf{F}_1 \mathbf{P}_{k-1} \mathbf{F}_1^T + \mathbf{Q}_1, \quad (3.11)$$

$$\mathbf{K}_k = \mathbf{P}_{k|k-1} \mathbf{H}_1^T \mathbf{S}_k^{-1}, \quad (3.12)$$

$$\mathbf{S}_k = \mathbf{H}_1 \mathbf{P}_{k|k-1} \mathbf{H}_1^T + \mathbf{R}. \quad (3.13)$$

The estimated state in (3.8) is the optimal solution without constraint (3.2). Now we take the constraint into consideration. The constraint can be rewritten as

$$\mathbf{D}\mathbf{x}_k = \mathbf{d}, \quad (3.14)$$

where $\mathbf{d}$ is $[0 \ 0]^T$, and

$$\mathbf{D} = \begin{bmatrix} 0 & -1 & 0 & 0 & \tan(h) & 0 \\ 0 & 0 & -1 & 0 & 0 & \tan(h) \end{bmatrix}, \quad (3.15)$$

h is the target heading, and it is defined as a clockwise angle from the true north. The optimal solution of the state with the constraint is

$$\tilde{\mathbf{x}}_k = \hat{\mathbf{x}}_k - \mathbf{x}_e, \quad (3.16)$$

where $\mathbf{x}_e$ can be considered as the model error of the existing uncoupled linear model, and it can be obtained by the Lagrangian method in a minimisation problem with a constraint [70]. The problem is formulated as

$$\tilde{\mathbf{x}}_k = \arg \min_{\mathbf{x}_k} (\mathbf{x}_k - \hat{\mathbf{x}}_k)^T \mathbf{P}_k^{-1} (\mathbf{x}_k - \hat{\mathbf{x}}_k), \quad (3.17)$$

subject to the constraint

$$\mathbf{D}\mathbf{x}_k = \mathbf{d}. \quad (3.18)$$

The Lagrangian function of the problem is

$$L = (\mathbf{x}_k - \hat{\mathbf{x}}_k)^T \mathbf{P}_k^{-1} (\mathbf{x}_k - \hat{\mathbf{x}}_k) + 2\lambda^T (\mathbf{D}\mathbf{x}_k - \mathbf{d}), \quad (3.19)$$

where λ is the Lagrangian multiplier. The first-order conditions for a minimum L are

$$\frac{\partial L}{\partial \mathbf{x}_k} = \mathbf{0}, \quad (3.20)$$

$$\frac{\partial L}{\partial \lambda} = \mathbf{0}, \quad (3.21)$$

which lead to,

$$\mathbf{P}_k^{-1}(\tilde{\mathbf{x}}_k - \hat{\mathbf{x}}_k) + \mathbf{D}^T \lambda = \mathbf{0}, \quad (3.22)$$

$$\mathbf{D} \tilde{\mathbf{x}}_k = \mathbf{d}. \quad (3.23)$$

The results of (3.22) and (3.23) are

$$\lambda = (\mathbf{D} \mathbf{P}_k^{-1} \mathbf{D})^{-1} (\mathbf{D} \hat{\mathbf{x}}_k - \mathbf{d}), \quad (3.24)$$

$$\tilde{\mathbf{x}}_k = \hat{\mathbf{x}}_k - \mathbf{P}_k \mathbf{D}^T (\mathbf{D} \mathbf{P}_k \mathbf{D}^T)^{-1} (\mathbf{D} \hat{\mathbf{x}}_k - \mathbf{d}). \quad (3.25)$$

Thus, the model error $\mathbf{x}_e$ is

$$\mathbf{x}_e = \mathbf{P}_k \mathbf{D}^T (\mathbf{D} \mathbf{P}_k \mathbf{D}^T)^{-1} (\mathbf{D} \hat{\mathbf{x}}_k - \mathbf{d}). \quad (3.26)$$

The state obtained from (3.25) is the same as the result of the so-called constrained Kalman filter which provide a solution to compute the constrained state $\tilde{\mathbf{x}}_k$ from the unconstrained state $\hat{\mathbf{x}}_k$. The assumption is that the constraint is known from a priori knowledge. However, in our problem, the target heading is unknown, and the constraint matrix D is therefore uncertain. Thus, it is difficult to apply the constraint KF to the problem.

3.2 Nonlinear Constant Heading Acceleration Model and Estimation Algorithms

Our approach to solve the estimation problem with constraint is to reduce the state size. Obviously, the six elements in the 2D uncoupled linear model can be reduced to five elements after constraint (3.2) being applied. Thus we suggest a constant heading acceleration model in this section. The state vector $\mathbf{x}$ and the measurement vector $\mathbf{z}$ are defined as

$$\mathbf{x} = [x \ y \ s \ a \ h]^T, \quad (3.27)$$

$$\mathbf{z} = [x_m \ y_m]^T, \quad (3.28)$$

where x and y are the target positions in x and y coordinates respectively; s , a and h represent the target speed, acceleration and heading, respectively; x_m and y_m are the measured target positions in x and y coordinates respectively. The transition equation is nonlinear, and is described by

$$\mathbf{x}_k = \mathbf{f}(\mathbf{x}_{k-1}) + \mathbf{w}_2, \quad (3.29)$$

where $\mathbf{w}_2$ is the white Gaussian process noise with covariance $\mathbf{Q}_2$, and

$$\mathbf{f} = [f_1 \ f_2 \ f_3 \ f_4 \ f_5]^T, \quad (3.30)$$

where

$$f_1 = x_{k-1} + [s_{k-1}T + 0.5a_{k-1}T^2] \sin h_{k-1}, \quad (3.31)$$

$$f_2 = y_{k-1} + [s_{k-1}T + 0.5a_{k-1}T^2] \cos h_{k-1}, \quad (3.32)$$

$$f_3 = s_{k-1} + a_{k-1}T, \quad (3.33)$$

$$f_4 = a_{k-1}, \quad (3.34)$$

$$f_5 = h_{k-1}. \quad (3.35)$$

The measurement equation is linear, and is defined as

$$\mathbf{z}_k = \mathbf{H}_2 \mathbf{x}_k + \mathbf{r}, \quad (3.36)$$

where

$$\mathbf{H}_2 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix}, \quad (3.37)$$

and $\mathbf{r}$ is the white Gaussian measurement noise with covariance $\mathbf{R}$, which is assumed to be obtained from a priori knowledge.

It is clear that the proposed nonlinear model describes the acceleration motion more accurately than the uncoupled linear model. However, there is no optimal algorithm for nonlinear tracking. The sub-optimal algorithms, such as EKF, UKF and PF, provide approximate solutions. In this research, we select the EKF and the UKF to perform nonlinear estimation. The PF is not considered as it is computational intensive.

In the proposed model, the transition function is a nonlinear function. Jacobian matrix of $f(\cdot)$ is required in the EKF. The Jacobian matrix $\mathbf{F}_{2,k}$ is defined

as

$$\mathbf{F}_{2,k} = \frac{\partial f(\mathbf{x}_k)}{\partial \mathbf{x}_k}. \quad (3.38)$$

Substituting (3.31)-(3.35) into (3.38), $\mathbf{F}_{2,k}$ is obtained as

$$\mathbf{F}_{2,k} = \begin{bmatrix} 1 & 0 & F_{13} & F_{14} & F_{15} \\ 0 & 1 & F_{23} & F_{24} & F_{25} \\ 0 & 0 & 1 & T & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad (3.39)$$

where

$$F_{13} = T \sin h_k, \quad (3.40)$$

$$F_{14} = 0.5T^2 \sin h_k, \quad (3.41)$$

$$F_{15} = [s_k T + 0.5a_k T^2] \cos h_k, \quad (3.42)$$

$$F_{23} = T \cos h_k, \quad (3.43)$$

$$F_{24} = 0.5T^2 \cos h_k, \quad (3.44)$$

$$F_{25} = -[s_k T + 0.5a_k T^2] \sin h_k. \quad (3.45)$$

With $\mathbf{F}_{2,k}$, the EKF can be easily implemented by the steps described in Chapter 2.

Implementing the UKF is straightforward. Given an initial state $\mathbf{x}_0$ and its error covariance $\mathbf{P}_0$, $2n_{\mathbf{x}} + 1 = 11$ sigma points are generated, where $n_{\mathbf{x}}$ is the state size, which is 5 in the proposed model. The state vector $\mathbf{x}$ and its error covariance $\mathbf{P}$ are then updated based on the steps described in Chapter 2.

Both EKF and UKF require an initial state such that consequential estimation on the state can be performed recursively. The accuracy of the initial state is important in nonlinear estimation. Track initiation for both EKF and UKF will be discussed in the next section.

3.3 Two-stage Least Squares Method for Track Initiation

Track initiation is an important step in tracking system, especially for nonlinear tracking. A wrong initial value can cause state divergence or mislead the recursive process to converge to a local optimum. A typical method is to view the problem as a nonlinear optimization problem. Normally a batch process is applied in the measurements of the first few time cycles. The nonlinear optimization problem can be solved by the Newton's algorithm iteratively [3] [71]. Another approach is the Markov Chain Monte Carlo (MCMC) method [72], which suggests a sampling method based on the likelihood linked to the observation equation. Both methods compute the initial state through iteration processes until some predetermined conditions are met.

In order to initiate an accurate nonlinear state rapidly, we develop a two-stage least squares method without iteration process. The target initial state and its error covariance can be efficiently estimated through this method.

The measurements in the first n times are

$$\mathbf{Z}_n = [\mathbf{z}_1 \quad \mathbf{z}_2 \quad \dots \quad \mathbf{z}_n], \quad (3.46)$$

where $\mathbf{z}_i$ is defined in (3.28), $i \in \{1, \dots, n\}$. The parameters, which are going to be estimated, are separated into two sets, $\mathbf{S}_1$ and $\mathbf{S}_2$. $\mathbf{S}_1$ consists of information in the first coordinate, for example, the x coordinate. $\mathbf{S}_2$ contains the information in another coordinate and the constraint between the two coordinates which is related to the target heading. $\mathbf{S}_1$ and $\mathbf{S}_2$ are defined as

$$\mathbf{S}_1 = [x_0 \quad \dot{x}_0 \quad \ddot{x}]^T, \quad (3.47)$$

$$\mathbf{S}_2 = [y_0 \quad r]^T, \quad (3.48)$$

where x_0 , $\dot{x}_0$ and $\ddot{x}$ are the target initial position, initial speed and acceleration in x coordinate, respectively; y_0 is the target initial position in y coordinate; r is the cotangent of the target heading, which is defined as

$$r = \cot h = \frac{\dot{y}_0}{\dot{x}_0} = \frac{\ddot{y}}{\ddot{x}}. \quad (3.49)$$

These two sets are estimated in two stages, which are described as follows.

Stage one estimates $\mathbf{S}_1$. The problem is formulated as

$$\mathbf{Z}_n(1, :)^T = \mathbf{A}\mathbf{S}_1 + \mathbf{e}_x, \quad (3.50)$$

where

$$\mathbf{Z}_n(1, :)^T = [x_{m,1} \ \cdots \ x_{m,n}]^T, \quad (3.51)$$

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & 0 \\ 1 & T & 0.5T^2 \\ 1 & 2T & 0.5(2T)^2 \\ \vdots & \vdots & \vdots \\ 1 & (n-1)T & 0.5[(n-1)T]^2 \end{bmatrix}, \quad (3.52)$$

$$\mathbf{e}_x = \begin{bmatrix} e_{x,1} & \cdots & e_{x,n} \end{bmatrix}^T, \quad (3.53)$$

where $e_{x,i}$ is the measurement error in x coordinate, $i \in \{1, \dots, n\}$, and its variance is σ^2 .

Using the least squares method, $\mathbf{S}_1$ is estimated by

$$\hat{\mathbf{S}}_1 = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{Z}_n(1, :)^T. \quad (3.54)$$

$\mathbf{S}_2$ is estimated in the second stage. By using the result obtained in the first stage, the measurement equation in the second coordinate can be written as

$$\mathbf{Z}_n(2, :)^T = \mathbf{B} \mathbf{S}_2 + \mathbf{e}_y, \quad (3.55)$$

where

$$\mathbf{Z}_n(2, :)^T = [y_{m,1} \ \cdots \ y_{m,n}]^T, \quad (3.56)$$

$$\mathbf{B} = \begin{bmatrix} 1 & 0 \\ 1 & T\dot{x}_0 + 0.5T^2\ddot{x} \\ 1 & 2T\dot{x}_0 + 0.5(2T)^2\ddot{x} \\ \vdots & \vdots \\ 1 & (n-1)T\dot{x}_0 + 0.5[(n-1)T]^2\ddot{x} \end{bmatrix}, \quad (3.57)$$

$$\mathbf{e}_y = \begin{bmatrix} e_{y,1} & \cdots & e_{y,n} \end{bmatrix}^T, \quad (3.58)$$

where $e_{y,i}$ is the measurement error in y coordinate, $i \in \{1, \dots, n\}$, and its variance is σ^2 .

$\mathbf{S}_2$ is estimated as

$$\hat{\mathbf{S}}_2 = (\mathbf{B}^T \mathbf{B})^{-1} \mathbf{B}^T \mathbf{Z}(2, :)^T. \quad (3.59)$$

The initial states defined in (3.27) for the first n times can then be obtained from $\hat{\mathbf{S}}_1$ and $\hat{\mathbf{S}}_2$. The target positions are then given by

$$x(1:n) = A\hat{\mathbf{S}}_1, \quad (3.60)$$

$$y(1:n) = B\hat{\mathbf{S}}_2. \quad (3.61)$$

The target speed, acceleration and heading are computed by, respectively,

$$s(1:n) = \sqrt{1+r^2}(\dot{x} + \mathbf{U}\ddot{x}), \quad (3.62)$$

$$a(1:n) = \mathbf{L}\sqrt{1+r^2}\ddot{x}, \quad (3.63)$$

$$h(1:n) = \mathbf{L}\tan^{-1}(1/r), \quad (3.64)$$

where $\mathbf{L}$ and $\mathbf{U}$ are

$$\mathbf{U} = \begin{bmatrix} 0 & 1 & \cdots & n-1 \end{bmatrix}_{1 \times n}, \quad (3.65)$$

$$\mathbf{L} = \begin{bmatrix} 1 & 1 & \cdots & 1 \end{bmatrix}_{1 \times n}. \quad (3.66)$$

It is difficult to estimate the state error covariance $\mathbf{P}$ analytically. This is because the error has been introduced to matrix $\mathbf{B}$ by $\dot{x}_0$ and $\ddot{x}$ in the second stage estimation. Hence, the error of $\mathbf{S}_2$ cannot be calculated using the least squares method. Thus, except the position x , the error variance for other state elements have to be obtained through approximation and Monte Carlo test. In this thesis, we compute and approximate the error variances for the first four state elements, namely, x , y , s and a as

$$\sigma_x^2 = \Sigma(1, 1) \quad (3.67)$$

$$\sigma_y^2 \approx \Sigma(1, 1) \quad (3.68)$$

$$\sigma_s^2 \approx \alpha \Sigma(2, 2) \quad (3.69)$$

$$\sigma_a^2 \approx \alpha \Sigma(3, 3) \quad (3.70)$$

where α is a scalar, and

$$\Sigma = (\mathbf{A}^T \mathbf{A})^{-1} \sigma^2. \quad (3.71)$$

The standard deviation σ_h on the heading error has to be obtained through a Monte Carlo experiment. We conducted an experiment on different measurement errors σ and different number of initial cycles n . Figure 3.2 shows the result obtained from 500 Monte Carlo runs. It is seen that σ_h is reduced with the number of initial cycles, and it increases with the measurement error

variance σ .

Figure 3.2: Target heading errors on different noise levels and different initial conditions

The state error covariance $\mathbf{P}_n$ at time n is approximated by a diagonal matrix as

$$\mathbf{P}_n = \text{diag}[\sigma_x^2 \quad \sigma_y^2 \quad \sigma_s^2 \quad \sigma_a^2 \quad \sigma_h^2]. \quad (3.72)$$

Now the initial states for the first n time cycles and the state error covariance at time n are completely obtained. Since there is no iteration process, the computational cost is much less compared to Newton's algorithm and the MCMC method.

3.4 Simulation and Analysis

This section demonstrates the performance of the proposed nonlinear acceleration model through simulation experiments. To illustrate the performance gained from the proposed nonlinear model, comparison on tracking accuracy is

made between the nonlinear model and the uncoupled linear model. The nonlinear model is implemented by the EKF and the UKF with the two-stage least squares track initiation. The Wiener CA model is selected as a representative of the uncoupled models, and the KF is used to estimate the state in this model. In the experiments, α in (3.69) and (3.70) is set to 2.

The test scenario is presented in Figure 3.3. It simulates a surface target taking a constant deceleration maneuvering from a cruising speed 15m/s to a slow speed 2.5m/s in 50 seconds. The target heading is 45 degrees. The target positions in x and y coordinates are detected every second. The measurement error is assumed to be white Gaussian in both x and y coordinates.

Figure 3.3: A test scenario to simulate a surface target taking a constant deceleration maneuvering from a cruising speed 15m/s to a slow speed 2.5m/s in 50s

The experiments are conducted on different measurement errors and different initial conditions. The Root-Mean-Squared Error (RMSE) on target position is

recorded for comparison. The position error at time k is computed by

$$p_{err,k} = \sqrt{x_{err,k}^2 + y_{err,k}^2} \quad (3.73)$$

where x_{err} and y_{err} are the differences between the estimated position and the ground truth in x and y coordinates respectively. The CRLB on position is also computed as a benchmark for performance evaluation. The CRLB can be obtained from the filtering information matrix recursively [36] [67] [73]. It is

$$\text{CRLB}_k = \sqrt{\mathbf{J}_k^{-1}(1, 1) + \mathbf{J}_k^{-1}(2, 2)} \quad (3.74)$$

where $\mathbf{J}_k$ is the information matrix, and it is given by

$$\mathbf{J}_k = [\mathbf{F}_{2,k-1}^{-1}]^T \mathbf{J}_{k-1} \mathbf{F}_{2,k-1}^{-1} + \mathbf{H}_2^T \mathbf{R}^{-1} \mathbf{H}_2 \quad (3.75)$$

where $\mathbf{F}_2$ and $\mathbf{H}_2$ are defined in (3.38) and (3.37).

The number of initial cycles n is set to 5, 6, 7 and 8 in different tests. The standard deviations of measurement noise σ in x and y coordinates are set to 5m and 10m, which can convert to the standard deviation of position noise σ_p as 7.07m and 14.14m respectively. $\mathbf{Q}_1$ is set as a diagonal matrix with diagonal elements $[1, 1, 0.1, 0.1, 0.01, 0.01]$, and $\mathbf{Q}_2$ is also a diagonal matrix with diagonal elements $[1, 1, 0.1, 0.01, (0.5\pi/180)^2]$. The test is conducted on 500 Monte Carlo runs. The results are summarized in Table 3.1.

In Table 3.1, the RMSE column represents the time averaging on the ensemble RMSE of p_{err} . The Noise Redu. column represents the noise reduction

percentage, which is computed by

$$\text{NoiseRedu.} = \frac{\sigma_p - \text{RMSE}}{\sigma_p} \times 100(\%). \quad (3.76)$$

Table 3.1: Algorithm performance on target position accuracy.

n	Algo. and CRLB	$\sigma_p=7.07\text{m}$		$\sigma_p=14.14\text{m}$	
		RMSE (m)	Noise Redu. (%)	RMSE (m)	Noise Redu. (%)
5	KF	4.16	41.2	8.17	42.2
	EKF	3.78	46.5	9.46	33.1
	UKF	3.75	47.0	7.93	43.9
	CRLB	3.49	50.6	6.99	50.6
6	KF	4.10	42.0	8.13	42.5
	EKF	3.75	47.0	7.62	46.1
	UKF	3.74	47.1	7.48	47.1
	CRLB	3.41	51.8	6.82	51.8
7	KF	4.01	43.3	8.02	43.3
	EKF	3.67	48.1	7.44	47.4
	UKF	3.66	48.2	7.39	47.7
	CRLB	3.34	52.8	6.67	52.8
8	KF	3.95	44.1	7.94	43.8
	EKF	3.65	48.4	7.39	47.7
	UKF	3.64	48.5	7.37	47.9
	CRLB	3.28	53.6	6.55	53.7

dark gray: Category I has small error on initial state;

middle gray: Category II has middle error on initial state;

light gray: Category III has large error on initial state.

It is seen from Table 3.1 that all RMSEs are 33% – 54% lower than the measurement errors. It indicates that all the algorithms are performing well.

The UKF and the EKF in the proposed nonlinear model generate lower RMSEs than the KF in the uncoupled linear model, except an EKF case with $n = 5$ and $\sigma_p = 14.14m$. Thus, the proposed nonlinear model outperforms the existing uncoupled linear model. It is known that the UKF has better performance than the EKF in nonlinear tracking. This point is further confirmed by the results in Table 3.1. The UKF shows less RMSEs than the EKF in all the cases.

Figure 3.4: RMSE on position vs. time when $\sigma_p=7.07m$, $n=8$ (category I)

To further analyze the performance difference between the UKF and the EKF, we categorize the results in Table 3.1 into three categories, namely, I, II and III. In category I, the UKF and the EKF have the equivalent performance, the difference of RMSEs between the EKF and the UKF is less than 1m. It can be seen that 6 cases, namely, all 4 cases from $\sigma_p = 7.07m$ and 2 cases from $\sigma_p = 14.14m$ with $n > 6$, belong to this category. Figure 3.4 shows the plot for position RMSE vs. time of a representative ($\sigma_p=7.07m$, $n = 8$) in category I.

Figure 3.5: RMSE on position vs. time when $\sigma_p=14.14\text{m}$, $n=6$ (category II)

Figure 3.6: RMSE on position vs. time when $\sigma_p=14.14\text{m}$, $n=5$ (category III)

It is obtained from 500 Monte Carlo runs. The results of both UKF and EKF converge to the CRLB, and their performances are almost the same in every time cycle. The result of the KF operating on the existing linear model has a gap to the CRLB. The cases in this category are either with smaller measurement error or longer initial cycles. These yield accurate initial state obtained from the two-stage least squares method. From Figure 3.2 we can observe that the heading error standard deviation σ_h of this category is lower than 8 degrees. It is an excellent initial condition as it results in excellent results on both EKF and UKF.

In category II, the performance difference of the EKF and the UKF can be seen, and both UKF and EKF are still better than the KF. There is only one case ($\sigma_p = 14.14\text{m}$ and $n = 6$) in our simulation. Figure 3.5 shows the plot for position RMSE vs. time for this case. It can be seen that the RMSE difference between the EKF and the UKF is obvious. σ_h of this category is around 10 degrees. The initial condition is not as good as that in Category I.

In category III, the EKF shows worse performance than the KF, and the UKF is still better than the KF. There is also only one case ($\sigma_p = 14.14\text{m}$ and $n = 5$) in our simulation. σ_h of this category reaches 15 degrees. Figure 3.6 shows the plot for position RMSE vs. time for this case. It can be seen that the EKF is diverged. This is because that the Jacobian approximation is inaccurate when the state error covariance is large. Although the UKF operates on the same initial condition, an unscented transform is more accurate than a Jacobian approximation, and hence it results in better performance in the UKF. The results confirm that the EKF is more sensitive to the initial condition than the UKF.

From these observations, we can conclude that the proposed nonlinear model performs better than the existing uncoupled linear model. Even when initial state has large error (σ_h reaches 15 degree), the nonlinear UKF still outperforms the linear KF. The UKF is less sensitive to the accuracy of the initial state than the EKF. Better nonlinear tracking performance can be obtained by improving the initial state accuracy, and that can be achieved by increasing the number of initial cycles.

3.5 Conclusion

This chapter proposes a nonlinear constant heading model for tracking a target which moves in a constant acceleration motion. The proposed nonlinear model describes the acceleration motion more accurately than the existing uncoupled linear mode, and the error created by the uncoupled linear model is proven. A two-stage least squares method is developed for nonlinear track initiation, and the EKF and the UKF are suggested for state updating. Simulation experiments on the proposed nonlinear model and the existing uncoupled linear model are conducted. The results show that the nonlinear model outperforms the linear model, and the UKF is better than the EKF in terms of track accuracy and sensitivity to the initial condition.

Chapter 4

A Pruning Based Multiple Model Filter with Sojourn-Time-Dependent Semi-Markov Switching

Model switching mechanism is an important component in dynamic Multiple Model (MM) approach. A Markov switching system with a fixed set of model transition/switching probabilities is usually applied on various dynamic MM algorithms [6] [74]. The model transition probabilities are assumed to be obtained from a priori knowledge. However, a priori knowledge is not always available in real cases. Researchers therefore proposed some methods to estimate transition probabilities online [75] [76]. Tugnait proposed a maximum likelihood approach to solve this problem [75]. However, the maximum likelihood method incurs high computational cost and is not affordable for real-time applications.

Orguner and Demirekler utilized Expectation Maximization (EM) procedure in an IMM filter with computational complexity N^3 , where N is the number of models in the system [76].

The above mentioned methods were developed based on Markov switching, which has a fixed set of model transition/switching probabilities. It is observed that a model transition probability does vary and is related to model sojourn time. Therefore a Sojourn-Time-Dependent semi-Markov (STDM) switching system, which has adaptive transition probabilities to sojourn times, is more suitable for the dynamic MM approach.

An STDM process is a probabilistic system [19] [77], which is more generic than a Markov process. Its transitions among models are based on two parameters, namely, the transition probability matrix of an embedded Markov process, and the holding/sojourn time in each process. The sojourn time could be a random variable governed by any probability density function (pdf). A Markov process is a special case of STDM processes when the sojourn times are exponentially distributed. The exponential sojourn time distributions lead to a fixed model transition probability matrix if the time interval of a discrete dynamic hybrid system is constant [2] [19]. Many MM algorithms have been proposed incorporating with these exponential sojourn time distributions [17] [18] [19] [20] [21]. They can operate on discrete systems with constant or variable time intervals. Strictly speaking, these MM algorithms belong to the class of the MM algorithms with normal Markov switching.

STDM-based MM algorithms with arbitrary sojourn time pdf have been proposed before [7] [78] [79]. An STDM-based IMM filter has been suggested with conditional STDM model transition probabilities. Since it belongs to the class of

merging-based MM approaches, the computation of the STDMM transition probabilities involves high computational cost, which increases with time. Thus, it is difficult to implement the STDMM-IMM filter in real applications. A fixed lag smoother for abruptly changing system has also been developed [78]. The model switching of the smoother can be a semi-Markov process with predefined sojourn time statistics. Its transition probability is approximated by the latest STDMM process, and the contribution of history processes is ignored. Recently, a Switching-Time-Conditioned IMM (STC-IMM) filter has been developed to cope with a special type of sojourn time pdf, which forces the models to remain fixed for some time [79]. The STC-IMM is less general, and has no advantage on applications with very short sojourn times. For example, the outlier sojourn time is only 1 time cycle in target tracking with outlier.

This chapter proposes a real-time implementable Pruning-based MM (PMM) filter with STDMM switching. A recursive form of hypothesis likelihoods is developed. It is a function of measurement residuals and STDMM interval-transition probabilities. The chapter is organized as follows. Section 4.1 describes the STDMM processes in detail. Section 4.2 proposes the STDMM-PMM filter with four steps. They are: 1) the hypotheses formation; 2) the hypothesis states estimation; 3) the hypothesis likelihoods computation and hypotheses pruning; 4) the final state updating. Section 4.3 and Section 4.4 apply the STDMM-PMM filter to two tracking problems, namely, the maneuvering target tracking and the target tracking with outlier presence. Comparisons are made among the STDMM-PMM, the Interacting Multiple Model (IMM) filter and the Multiple Model Multiple Hypothesis (M^3H) filter through simulation data. Section 4.5 gives the conclusion of the chapter.

4.1 Sojourn-Time-Dependent Semi-Markov Processes

This section describes STDMM processes incorporating with a dynamic multiple model system. A dynamic multiple model system can be viewed as a series of processes. Figure 4.1 shows such a system, which jumps among a finite number of models after a random sojourn/holding time τ .

Figure 4.1: A dynamic multiple model system with model sojourn time τ

To describe an STDMM system, two components need to be defined. They are the model transition probability from the current model to the next model and the model sojourn time τ . These two components are independent.

The model transition probabilities are represented by the embedded Markov transition probability matrix $\mathbf{P}$, which is used to determine the next model to shift from the current model after a holding time. It is pre-determined, and is independent of the sojourn time τ . Assuming the number of models in an MM hybrid system is M , the matrix $\mathbf{P}$ consists of $M \times M$ elements, and each element p_{ij} is defined as

$$p_{ij} \triangleq \text{prob}\{\text{next model is } j \mid \text{current model is } i\}, \quad (4.1)$$

and the following conditions must be satisfied:

$$\sum_{j=1}^M p_{ij} = 1, \quad (4.2)$$

where $i \in \{1, \dots, M\}$ and $p_{ij} \geq 0$.

The second component of an STDM system is the sojourn time τ in each process. τ is treated as a random variable governed by a probability density function $w_{ij}(\tau)$, where τ is the sojourn time of the process in model i before transiting to the next model j . In a generic STDM system, the number of $w_{ij}(\tau)$ is $M \times M$, where $i \in \{1, \dots, M\}$, $j \in \{1, \dots, M\}$.

Let $\{\mathcal{M}(t_0) = i, \mathcal{M}(t_0^-) \neq i\}$ represent the event of a system entering model i at time t_0 , and $\{\mathcal{M}(t_0 + t) = j\}$ be the event of the system in model j at time $t_0 + t$, where $\mathcal{M}$ is the variable of the system model. The interval-transition probability $\phi_{ij}(t)$ between t_0 and $t_0 + t$ is defined as

$$\phi_{ij}(t) = P\{\mathcal{M}(t_0 + t) = j \mid \mathcal{M}(t_0) = i, \mathcal{M}(t_0^-) \neq i\}, \quad (4.3)$$

and it can be derived as [77] [19]

$$\phi_{ij}(t) = \left[\sum_{m=1}^M p_{im} \int_t^\infty w_{im}(\tau) d\tau \right] \delta_{ij} + \sum_{m=1}^N p_{im} \int_0^t \phi_{mj}(t - \tau) w_{mj}(\tau) d\tau, \quad (4.4)$$

where

$$\delta_{ij} = \begin{cases} 1, & \text{for } i = j, \\ 0, & \text{for } i \neq j. \end{cases}$$

The first term on the right-hand side of equation (4.4) is the probability of the system in model i from time t_0 to $t_0 + \tau$ with $\tau > t$. It includes the hypotheses

that the system is in model i from t_0 to $t_0 + t$, and switches to other models at all possible times after $t_0 + t$. The second term on the right-hand side of the equation combines two independent events. Firstly, the system makes a direct transition from model i to model m with sojourn time τ , where $\tau < t$. Secondly, the system transits from model m to model j through any arbitrary path during time $t - \tau$, and maintains in model j until time $t_0 + t$. Obviously, the interval-transition probability $\phi_{ij}(t)$ takes all hypotheses from model i to model j into consideration. It is suitable for a merging based MM approach.

In this chapter, we focus on a pruning based MM approach. The interval-transition probability on each hypothesis needs to be computed. The interval-transition probability $\phi_{ij}^n(t)$ for a hypothesis H^n is defined as

$$\phi_{ij}^n(t) = P\{\mathcal{M}(t + t_0) = j \mid \mathcal{M}(t_0) = i, \mathcal{M}(t_0^-) \neq i, H^n\}, \quad (4.5)$$

where n is the hypothesis index.

Before we derive $\phi_{ij}^n(t)$, the definition of a transition needs to be clarified. A transition in a semi-Markov system means a system makes a movement from a model to another model. This is called a real transition, as the model is changed in two consecutive processes. Some STDM systems allow a "virtual" transition. The system can transit to the same model, and the two consecutive processes share the same model. In this research, we only allow real transitions. The two consecutive processes in the same model are treated as one combined process with a longer sojourn time. Thus, the diagonal elements of $\mathbf{P}$ are all 0.

Now let us define the hypothesis structure. A hypothesis consists of a sequence of processes. Each process has two properties, namely, its model and its sojourn time. Thus, a hypothesis H^n can be described by two vectors, namely,

the model sequence $\mathbf{M}^n$ and the sojourn time sequence $\mathbf{\Gamma}^n$,

$$\mathbf{M}^n = [\mathcal{M}_1^n, \mathcal{M}_2^n, \dots, \mathcal{M}_L^n], \quad (4.6)$$

$$\mathbf{\Gamma}^n = [\tau_1^n, \tau_2^n, \dots, \tau_L^n], \quad (4.7)$$

where L is the number of processes in hypothesis H^n . For example, the hypothesis described in Figure 4.1 can be represented by

$$\mathbf{M}^n = [1, 2, 1, 3], \quad (4.8)$$

$$\mathbf{\Gamma}^n = [5, 3, 6, 2]. \quad (4.9)$$

Since most sensor measurements are updated regularly, a target tracking problem is normally defined as a discrete dynamic system. Thus, τ can be treated as a discrete random variable, and is governed by its probability mass function (pmf) $w_{ij}(\tau)$. To reduce the complexity of an STD system, we assume that the pmf of τ does not relate to the next model to be transited, and it can be completely determined by the current model. Thus, the number of pmfs is reduced to M by setting

$$w_i(\tau) = w_{ij}(\tau) \quad (4.10)$$

where $i \in \{1, \dots, M\}$ and $j \in \{1, \dots, M\}$.

Since model indices i and j , and interval time t have been defined in the hypothesis, $\phi_{ij}^n(t)$ can be denoted as ϕ^n for simplicity. The probability ϕ^n is then computed by the probabilities of a sequence of independent processes specified

in the hypothesis H^n as

$$\begin{aligned}
 \phi^n &= \text{prob}\{H^n\} \\
 &= \prod_{l=1}^L p_l \\
 &= \begin{cases} \sum_{\tau=\tau_1^n}^{\infty} w_{\mathcal{M}_1^n}(\tau_1^n), & \text{for } L = 1 \\
 w_{\mathcal{M}_1^n}(\tau_1^n) \left(\prod_{l=2}^{L-1} p_{\mathcal{M}_{l-1}^n, \mathcal{M}_l^n} w_{\mathcal{M}_l^n}(\tau_l^n) \right) \sum_{\tau=\tau_L^n}^{\infty} p_{\mathcal{M}_{L-1}^n, \mathcal{M}_L^n} w_{\mathcal{M}_L^n}(\tau_L^n), & \text{for } L > 1 \end{cases}
 \end{aligned} \tag{4.11}$$

where L is the number of processes, and p_l is the probability of the l th process in hypothesis H^n .

We use an example to illustrate the computation of the interval-transition probabilities in an STDM system with two models. The embedded Markov transition probability matrix $\mathbf{P}$ is

$$\mathbf{P} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}. \tag{4.12}$$

The two sojourn time pmfs, $w_1(\tau)$ and $w_2(\tau)$, are shown in Figure 4.2.

The STDM interval-transition probabilities for eight hypotheses shown in Figure 4.3 are computed by (4.11). For examples, there is one process with $M^1 = 1$ and $\Gamma^1 = 4$ in hypothesis H^1 . The interval-transition of H^1 is $\phi^1 = \sum_{\tau=4}^{\infty} w_1(\tau)$. In hypothesis H^5 , there are 3 processes with $M^5 = [1, 2, 1]$ and $\Gamma^5 = [1, 1, 2]$. The interval-transition of H^5 is then $\phi^5 = w_1(1)p_{12}w_2(1)\sum_{\tau=2}^{\infty} w_1(\tau) =$

Figure 4.2: Probability mass functions in an STDM system with two models

Figure 4.3: Transition probability ϕ for each hypothesis in a two-model STDM system

$$w_1(1)w_2(1) \sum_{\tau=2}^{\infty} w_1(\tau).$$

4.2 A Pruning Based Multiple Model Filter with STDM Switching

This section suggests a pruning based MM filter with STDM switching. This filter is named as STDM-PMM filter. It is developed on the basis of the M³H

filter, which is a pruning-based MM filter with normal Markov switching [13]. Similar to a generic pruning based MM filter, the proposed STDM-PMM filter consists of four steps shown in Figure 4.4. Step 1 performs hypotheses formation.

Figure 4.4: The estimation steps of the STDM-PMM filter

It creates the new hypotheses from the existing hypotheses. Step 2 estimates the state for each hypothesis. A KF, EKF or other estimation algorithm is usually applied for linear or nonlinear estimation conditioned on the model defined in each hypothesis. Step 3 computes the likelihood for each hypothesis, and prunes the unlikely hypotheses. Step 4 computes the final state from the remaining hypotheses. These four steps are generic for all pruning-based MM algorithms, and detailed steps of the STDM-PMM filter are described in the following paragraphs.

Step 1. Hypotheses formation: Hypotheses constituted by STDM processes are established recursively at each time cycle. This step forms the hypotheses and updates the hypothesis structures.

The hypotheses formation at time k is illustrated in Figure 4.5. The number of hypotheses remaining at time $k - 1$ is N_{k-1} . The latest models in these hypotheses are $\mathcal{M}_{k-1}^1, \dots, \mathcal{M}_{k-1}^{N_{k-1}}$. The number of models in the system is M . All M models are attached to the existing hypotheses. The number of newly formed hypotheses is then $\bar{N}_k = N_{k-1} \times M$, and the newly formed hypotheses are $H_k^1, H_k^2, \dots, H_k^{\bar{N}_k}$ with model sequences $\mathbf{M}_k^1, \mathbf{M}_k^2, \dots, \mathbf{M}_k^{\bar{N}_k}$ and sojourn time sequences $\mathbf{\Gamma}_k^1, \mathbf{\Gamma}_k^2, \dots, \mathbf{\Gamma}_k^{\bar{N}_k}$. Let $\mathcal{M}_{k-1}^n$ and $\mathcal{M}_k^n$ be the models of hypothesis H_k^n

Figure 4.5: Hypotheses formation at time k

at time $k-1$ and k respectively, the model sequence $\mathbf{M}^n$ and the sojourn time sequence $\mathbf{\Gamma}^n$ are updated by

Case 1 ($\mathcal{M}_{k-1}^n \neq \mathcal{M}_k^n$):

$$\mathbf{M}_k^n = [\mathbf{M}_{k-1}^n, \mathcal{M}_k^n], \quad (4.13)$$

$$\mathbf{\Gamma}_k^n = [\mathbf{\Gamma}_{k-1}^n, 1]. \quad (4.14)$$

Case 2 ($\mathcal{M}_{k-1}^n = \mathcal{M}_k^n$):

$$\mathbf{M}_k^n = \mathbf{M}_{k-1}^n, \quad (4.15)$$

$$\mathbf{\Gamma}_k^n(end) = \mathbf{\Gamma}_k^n(end) + 1. \quad (4.16)$$

With $\mathbf{M}_k^n$ and $\mathbf{\Gamma}_k^n$, the probability ϕ_k^n for the hypothesis H_k^n can be computed using (4.11).

Step 2. Hypothesis states estimation: Once hypotheses are formed,

the hypothesis states are estimated in parallel based on the models indicated in the hypotheses. A dynamic estimation is governed by a transition equation and a measurement equation. For hypothesis H^n , they are described in (4.17) and (4.18) respectively.

$$\mathbf{x}_k^n = f^m(\mathbf{x}_{k-1}^n) + \mathbf{q}_{k-1}^m, \quad (4.17)$$

$$\mathbf{z}_k = h^m(\mathbf{x}_k^n) + \mathbf{r}_k, \quad (4.18)$$

where n is the hypothesis index, and $n \in \{1, \dots, \bar{N}_k\}$; m represents the model index defined in hypothesis H^n , and $m \in \{1, \dots, M\}$; $\mathbf{x}_{k-1}^n$ is the state vector of hypothesis H^n at time $k-1$; $f^m(\cdot)$ is the state transition function of model m ; $\mathbf{q}_{k-1}^m$ is the process noise of model m ; $\mathbf{z}_k$ is the measurement vector; $h^m(\cdot)$ is the measurement function of model m ; $\mathbf{r}_k$ is the measurement noise. A KF is applied when both $f^m(\cdot)$ and $h^m(\cdot)$ are linear functions, whereas an Extended Kalman Filter (EKF) or other nonlinear filter can be used when either $f^m(\cdot)$ or $h^m(\cdot)$ is a nonlinear function.

The hypothesis states and their error covariances estimated by a linear or nonlinear filter are $\mathbf{x}_k^1, \dots, \mathbf{x}_k^{\bar{N}_k}$ and $\mathbf{P}_k^1, \dots, \mathbf{P}_k^{\bar{N}_k}$ respectively. The measurement residual $\mathbf{v}_k$ and its covariance $\mathbf{S}_k$ are also computed by the filter for each hypothesis. The measurement residual $\mathbf{v}_k$ is defined as the difference between the predicted measurement and the real measurement at time k . $\mathbf{v}_k^1, \dots, \mathbf{v}_k^{\bar{N}_k}$ and $\mathbf{S}_k^1, \dots, \mathbf{S}_k^{\bar{N}_k}$ will be used for hypothesis likelihoods computation in Step 3.

Step 3. Hypothesis likelihoods computation and hypotheses pruning: A recursive likelihood computation for each hypothesis is derived, and hypotheses are then pruned based on the hypothesis likelihoods.

a) Compute hypothesis likelihoods

The likelihood of the hypothesis H_k^i , given $\mathbf{Z}_k$, is denoted as $P\{H_k^i|\mathbf{Z}_k\}$, where $i \in \{1, \dots, \bar{N}_k\}$, and $\mathbf{Z}_k$ is the measurement sequence $\{z_\kappa\}_{\kappa=1}^k$. $P\{H_k^i|\mathbf{Z}_k\}$ can be derived from Bayes' rule as

$$\begin{aligned} P\{H_k^i|\mathbf{Z}_k\} &= P\{H_k^i|z_k, \mathbf{Z}_{k-1}\} \\ &= \frac{P\{z_k|H_k^i, \mathbf{Z}_{k-1}\}P\{H_k^i|\mathbf{Z}_{k-1}\}}{\sum_{j=1}^{\bar{N}_k} P\{z_k|H_k^j, \mathbf{Z}_{k-1}\}P\{H_k^j|\mathbf{Z}_{k-1}\}}. \end{aligned} \quad (4.19)$$

$P\{z_k|H_k^i, \mathbf{Z}_{k-1}\}$ is the conditional probability of the observation z_k , given the hypothesis H_k^i and the measurement sequence $\mathbf{Z}_{k-1}$. It can be computed by

$$P\{z_k|H_k^i, \mathbf{Z}_{k-1}\} = \mathcal{N}(\mathbf{v}_k^i; 0, \mathbf{S}_k^i), \quad (4.20)$$

where $\mathcal{N}(\cdot)$ represents the Gaussian probability density function with mean 0 and covariance $\mathbf{S}_k^i$, where $\mathbf{v}_k^i$ and $\mathbf{S}_k^i$ are computed in Step 2.

To derive $P\{H_k^i|\mathbf{Z}_{k-1}\}$, we represent the hypothesis H_k^i by two joint events $\{H_{k-1}^j, s_k^i\}$, where H_{k-1}^j is the parent hypothesis of H_k^i at time $k-1$, and s_k^i is the event of adding the latest model of H_k^i to H_{k-1}^j at time k . $P\{H_k^i|\mathbf{Z}_{k-1}\}$ is then computed by

$$\begin{aligned} P\{H_k^i|\mathbf{Z}_{k-1}\} &= \frac{P\{H_k^i, \mathbf{Z}_{k-1}\}}{P\{\mathbf{Z}_{k-1}\}} \\ &= \frac{P\{s_k^i, H_{k-1}^j, \mathbf{Z}_{k-1}\}}{P\{\mathbf{Z}_{k-1}\}} \\ &= \frac{P\{s_k^i, H_{k-1}^j, \mathbf{Z}_{k-1}\}P\{H_{k-1}^j, \mathbf{Z}_{k-1}\}}{P\{H_{k-1}^j, \mathbf{Z}_{k-1}\}P\{\mathbf{Z}_{k-1}\}} \\ &= P\{s_k^i|H_{k-1}^j, \mathbf{Z}_{k-1}\}P\{H_{k-1}^j|\mathbf{Z}_{k-1}\}. \end{aligned} \quad (4.21)$$

Since s_k^i and $\mathbf{Z}_{k-1}$ are independent, $P\{s_k^i|H_{k-1}^j, \mathbf{Z}_{k-1}\}$ can be simplified as

$$P\{s_k^i|H_{k-1}^j, \mathbf{Z}_{k-1}\} = P\{s_k^i|H_{k-1}^j\}. \quad (4.22)$$

The computation of $P\{s_k^i|H_{k-1}^j\}$ is illustrated in Figure 4.6. M hypotheses are

Figure 4.6: Hypothesis probability computation.

generated from the parent hypothesis H_{k-1}^j , and H_k^i is one of M hypotheses. From (4.11), we know that the probabilities of H_{k-1}^j and H_k^i are ϕ_{k-1}^j and ϕ_k^i respectively. Thus, $P\{s_k^i|H_{k-1}^j\}$ is given by

$$\begin{aligned} P\{s_k^i|H_{k-1}^j\} &= \frac{P\{s_k^i, H_{k-1}^j\}}{P\{H_{k-1}^j\}} \\ &= \frac{P\{H_k^i\}}{P\{H_{k-1}^j\}} \\ &= \frac{\phi_k^i}{\phi_{k-1}^j}. \end{aligned} \quad (4.23)$$

Putting (4.20), (4.21), (4.22), (4.23) into (4.19), we obtain the recursive form

of $P\{H_k^i|\mathbf{Z}_k\}$ as

$$P\{H_k^i|\mathbf{Z}_k\} = \frac{\frac{\phi_k^i}{\phi_{k-1}^j} \mathcal{N}(\mathbf{v}_k^i; 0, \mathbf{S}_k^i) P\{H_{k-1}^j|\mathbf{Z}_{k-1}\}}{\sum_{l=1}^{\bar{N}_k} \frac{\phi_k^l}{\phi_{k-1}^{l1}} \mathcal{N}(\mathbf{v}_k^l; 0, \mathbf{S}_k^l) P\{H_{k-1}^l|\mathbf{Z}_{k-1}\}}. \quad (4.24)$$

To simplify the expression, let $\bar{u}_k^i = P\{H_k^i|\mathbf{Z}_k\}$ and $u_{k-1}^j = P\{H_{k-1}^j|\mathbf{Z}_{k-1}\}$, we obtain

$$\bar{u}_k^i = \frac{\frac{\phi_k^i}{\phi_{k-1}^j} \mathcal{N}(\mathbf{v}_k^i; 0, \mathbf{S}_k^i) u_{k-1}^j}{\sum_{l=1}^{\bar{N}_k} \frac{\phi_k^l}{\phi_{k-1}^{l1}} \mathcal{N}(\mathbf{v}_k^l; 0, \mathbf{S}_k^l) u_{k-1}^{l1}}, \quad (4.25)$$

where $l1$ is the parent hypothesis index of the hypothesis H_k^l , and $l1 \in \{1, \dots, N_{k-1}\}$.

b) Prune hypotheses and update hypothesis structures

The hypotheses pruning strategy is the same as that of the M³H filter. Hypothesis i is pruned if

$$\frac{\mu_k^i}{\hat{\mu}_k^i} < \lambda_T, \quad (4.26)$$

where

$$\hat{i} = \arg \max_i \mu_k^i, \quad (4.27)$$

λ_T is a predefined pruning threshold, $\lambda_T \in (0, 1]$. A higher value of λ_T represents that more hypotheses are pruned. For example, only one hypothesis, namely the one with the maximum likelihood $\mu_k^{\hat{i}}$, is left, if λ_T is set to 1.

To prevent the computational cost on the transition probability ϕ from increasing with time, the model sequence $\mathbf{M}_k$ and the sojourn time sequence $\mathbf{\Gamma}_k$ should be updated. The earlier processes can be removed from the sequences if

they are shared by all the remaining hypotheses. For example, if the remaining hypotheses in Figure 4.3 are

$$\begin{aligned}
H_k^1 : \quad \mathbf{M}_k^1 &= 1 & \Gamma_k^1 &= 4; \\
H_k^2 : \quad \mathbf{M}_k^2 &= [1, 2] & \Gamma_k^2 &= [3, 1]; \\
H_k^3 : \quad \mathbf{M}_k^3 &= [1, 2, 1] & \Gamma_k^3 &= [2, 1, 1]; \\
H_k^4 : \quad \mathbf{M}_k^4 &= [1, 2] & \Gamma_k^4 &= [2, 2],
\end{aligned} \tag{4.28}$$

the first processes of the four hypotheses are model 1 with different sojourn times. They are not the same process as their sojourn times are different. The hypothesis structures are not updated at time k . If the most likely hypothesis in Figure 4.3 happens to be on another branch, the four remaining hypotheses will be located at the lower part, and they will be

$$\begin{aligned}
H_k^5 : \quad \mathbf{M}_k^5 &= [1, 2, 1] & \Gamma_k^5 &= [1, 1, 2]; \\
H_k^6 : \quad \mathbf{M}_k^6 &= [1, 2, 1, 2] & \Gamma_k^6 &= [1, 1, 1, 1]; \\
H_k^7 : \quad \mathbf{M}_k^7 &= [1, 2, 1] & \Gamma_k^7 &= [1, 2, 1]; \\
H_k^8 : \quad \mathbf{M}_k^8 &= [1, 2] & \Gamma_k^8 &= [1, 3].
\end{aligned} \tag{4.29}$$

It can be seen that the first processes of the four hypotheses are all model 1 with sojourn time 1. They are the same, and can be removed from the sequences.

The hypothesis sequences are then updated as

$$\begin{aligned}
H_k^5 : \quad \mathbf{M}_k^5 &= [2, 1] & \Gamma_k^5 &= [1, 2]; \\
H_k^6 : \quad \mathbf{M}_k^6 &= [2, 1, 2] & \Gamma_k^6 &= [1, 1, 1]; \\
H_k^7 : \quad \mathbf{M}_k^7 &= [2, 1] & \Gamma_k^7 &= [2, 1]; \\
H_k^8 : \quad \mathbf{M}_k^8 &= 2 & \Gamma_k^8 &= 3.
\end{aligned} \tag{4.30}$$

The hypothesis sequences updating method mentioned above can maintain the hypothesis structure effectively. Only those processes related to computation of the semi-Markov transition probabilities are kept in hypothesis sequences. In this way, hypothesis sequences are not increasing with time.

Step 4. Final state updating: The final state decision is quite straight forward. It can be either a weighted sum of the remaining states as

$$\mathbf{x}_k = \sum_{i=1}^{n^k} \mu_k^i \mathbf{x}_k^i, \quad (4.31)$$

$$\mathbf{P}_k = \sum_{i=1}^{N_k} \mu_k^i [\mathbf{P}_k^i + (\mathbf{x}_k^i - \mathbf{x}_k)(\mathbf{x}_k^i - \mathbf{x}_k)^T], \quad (4.32)$$

or the state of the hypothesis with the highest likelihood as

$$\mathbf{x}_k = \mathbf{x}_k^{\hat{i}}, \quad (4.33)$$

$$\mathbf{P}_k = \mathbf{P}_k^{\hat{i}}, \quad (4.34)$$

where

$$\hat{i} = \arg \max_i \mu_k^i, \quad (4.35)$$

The state history can also be updated if the state is selected by the highest hypothesis.

Now we summarize the estimation steps of the proposed STDM-PMM filter. Given the hypothesis model sequences $\mathbf{M}_{k-1}^i$, the hypothesis sojourn time sequences $\mathbf{T}_{k-1}^i$, the hypothesis likelihood μ_{k-1}^i , the hypothesis state $\mathbf{x}_{k-1}^i$ and the state error covariance $\mathbf{P}_{k-1}^i$, where $i = 1, \dots, N_{k-1}$, the estimation steps are shown in Table 4.1.

Table 4.1: The STDM-PMM filter

1. Hypotheses formation	
· compute the number of newly formed hypotheses	
$\bar{N}_k = N_{k-1} \times M$,	
where N_{k-1} is the number of hypotheses at time $k - 1$,	
M is the number of models.	
· update $\mathbf{M}_k^i$ and $\mathbf{\Gamma}_k^i$ based on (4.13)-(4.16).	
· for $i = 1 : \bar{N}_k$	
$j \sim$ the index of the parent hypothesis of the current hypothesis H_k^i ;	
$\mathbf{x}_{k-1}^i = \mathbf{x}_{k-1}^j$;	
$\mathbf{P}_{k-1}^i = \mathbf{P}_{k-1}^j$;	
end.	
2. Hypothesis states estimation	
for $i = 1 : \bar{N}_k$	
$[\mathbf{x}_k^i, \mathbf{P}_k^i, \mathbf{v}_k^i, \mathbf{S}_k^i] = KF(\mathbf{x}_{k-1}^i, \mathbf{P}_{k-1}^i)$ or,	
$[\mathbf{x}_k^i, \mathbf{P}_k^i, \mathbf{v}_k^i, \mathbf{S}_k^i] = EKF(\mathbf{x}_{k-1}^i, \mathbf{P}_{k-1}^i)$,	
end.	
3. Hypothesis likelihoods computation and hypotheses pruning	
· hypothesis likelihoods computation	
for $i = 1 : \bar{N}_k$	
$j \sim$ the index of the parent hypothesis of the current hypothesis H_k^i ;	
$l1 \sim$ the index of the parent hypothesis of the current hypothesis H_k^l ;	
$\bar{u}_k^i = \frac{\frac{\phi_k^i}{\phi_{k-1}^j} \mathcal{N}(\mathbf{v}_k^i; 0, \mathbf{S}_k^i) u_{k-1}^j}{\sum_{l=1}^{\bar{N}_k} \frac{\phi_k^l}{\phi_{k-1}^{l1}} \mathcal{N}(\mathbf{v}_k^l; 0, \mathbf{S}_k^l) u_{k-1}^{l1}};$	
end.	
· hypotheses pruning	
- prune the unlikely hypotheses based on (4.26);	
- $N_k \sim$ the number of the remaining hypotheses;	
- $\mu_k^i \sim$ re-normalized hypothesis likelihoods, where $i = 1, \dots, N_k$;	
· hypothesis structures updating	
- update $\mathbf{M}_k^i$ and $\mathbf{\Gamma}_k^i$ by removing the common processes in the earlier stage.	
4. Final state updating	
update $\mathbf{x}_k, \mathbf{P}_k$ from $\mathbf{x}_k^i, \mathbf{P}_k^i$, and μ_k^i using (4.31)-(4.34), where $i = 1, \dots, N_k$.	

4.3 Apply the STDM-PMM Filter on Tracking a Maneuvering Target

In this section, the performance of the proposed STDM-PMM filter is demonstrated through simulation experiments on tracking a maneuvering target. To

analyze the performance improvement of the STDMM-PMM filter, comparisons are made among the IMM, the M³H and the STDMM-PMM filters. The tracking accuracy in position and speed is shown in the comparison results. The model index versus time is also studied.

The test scenario is a generic Air Traffic Control (ATC) problem shown in Figure 4.7. The radar scan interval T is 5 seconds. An aircraft with an assumed speed of 120m/s travels at a constant velocity for 15 scans, and then it executes a 90° turn in 12 scans with a turn rate of 1.5°/s. The aircraft returns to the constant velocity model and maintains for 15 scans after the turn. The radar provides position-only measurements (after the polar to Cartesian conversion) with Root-Mean-Square Error (RMSE) of 100m in each of the two Cartesian coordinates. The RMSE on position is then 141.1m.

Figure 4.7: A generic maneuvering scenario in air traffic control for testing the MM algorithms

Two models ($M = 2$), namely, a linear constant velocity model as model 1 and a nonlinear constant turn model as model 2, are deployed in the three

filters. A KF is used to estimate the state in model 1, whereas an EKF is used for model 2. The transition equation and the measurement equation for both models are defined as follows:

$$\mathbf{x}_k^n = \mathbf{F}^m \mathbf{x}_{k-1}^n + \mathbf{G}^m \mathbf{w}_{k-1}^m, \quad (4.36)$$

$$\mathbf{z}_k = \mathbf{H}^m \mathbf{x}_k^n + \mathbf{r}_k, \quad (4.37)$$

where m is the model index, $m \in \{1, 2\}$; n is the hypothesis index; $\mathbf{w}_{k-1}^m$ is the white Gaussian process noise to cater for small acceleration with covariance $\mathbf{Q}_{k-1}^m$, and $\mathbf{Q}_{k-1}^m$ is a 2×2 diagonal matrix; the diagonal elements in $\mathbf{Q}_{k-1}^m$ are set to 1 for both models; $\mathbf{r}_k$ is the white Gaussian measurement noise with covariance $\mathbf{R}_k$, and it is the 2×2 diagonal matrix with diagonal elements of 10000. The state vector for model 1 is

$$\mathbf{x}_k = [x_k \quad \dot{x}_k \quad y_k \quad \dot{y}_k]^T,$$

and it is

$$\mathbf{x}_k = [x_k \quad \dot{x}_k \quad y_k \quad \dot{y}_k \quad \omega_k]^T,$$

for model 2, where x_k and y_k are the target positions at time k in x and y coordinates respectively; $\dot{x}_k$ and $\dot{y}_k$ are the target speeds at time k in x and y coordinates respectively; ω_k is the target turn rate at time k . The measurement vectors for both models are the same, and they are

$$\mathbf{z}_k = [x_{m,k} \quad y_{m,k}]^T,$$

where $x_{m,k}$ and $y_{m,k}$ are the measured target positions at time k in x and

y coordinates respectively. The transition matrices for the two models are, respectively,

$$\mathbf{F}^1 = \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{F}^2 = \begin{bmatrix} 1 & \frac{\sin \omega T}{\omega} & 0 & -\frac{1 - \cos \omega T}{\omega} & 0 \\ 0 & \cos \omega T & 0 & -\sin \omega T & 0 \\ 0 & \frac{1 - \cos \omega T}{\omega} & 1 & \frac{\sin \omega T}{\omega} & 0 \\ 0 & \sin \omega T & 0 & \cos \omega T & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix},$$

where T is the time interval. The gain matrices for the two models are, respectively,

$$\mathbf{G}^1 = \begin{bmatrix} 0.5T^2 & 0 \\ T & 0 \\ 0 & 0.5T^2 \\ 0 & T \end{bmatrix}, \quad \mathbf{G}^2 = \begin{bmatrix} 0.5T^2 & 0 \\ T & 0 \\ 0 & 0.5T^2 \\ 0 & T \\ 0 & 0 \end{bmatrix}.$$

The measurement matrices are

$$\mathbf{H}^1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}, \quad \mathbf{H}^2 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix}.$$

Normal Markov switching is used in the IMM filter and the M³H filter. The fixed transition probability matrix for both filters is

$$\mathbf{P} = \begin{bmatrix} 0.95 & 0.05 \\ 0.10 & 0.90 \end{bmatrix}. \quad (4.38)$$

In the STDMM-PMM filter, the embedded Markov transition probability matrix

$\mathbf{P}$ is

$$\mathbf{P} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \quad (4.39)$$

and the probability mass functions of sojourn time τ in the STDMM switching system are similar to those of Example 3 of [78], and they are:

$$w_1(\tau) = \begin{cases} \varepsilon, & \text{for } \tau < 10, \\ \alpha_1 e^{-|\tau-15|/10}, & \text{for } \tau \geq 10, \end{cases} \quad (4.40)$$

$$w_2(\tau) = \begin{cases} \varepsilon, & \text{for } \tau < 10, \\ \alpha_2 e^{-|\tau-12|/2}, & \text{for } \tau \geq 10, \end{cases} \quad (4.41)$$

where ε is a small value 0.0001; α_1 and α_2 are set to particular values, such that both $\sum_{\tau=1}^{\infty} w_1$ and $\sum_{\tau=1}^{\infty} w_2$ are 1. The STDMM interval-transition probability ϕ is computed by (4.11) based on $w_1(\tau)$, $w_2(\tau)$ and $\mathbf{P}$. The final state is the state in the most likely hypothesis.

It is known that the performance of a pruning-based MM approach depends on the number of remaining hypotheses, and the number of remaining hypotheses is decided by the pruning threshold λ_T . To analyze the algorithms performance, we conduct simulation tests on two different λ_T , namely, $\lambda_T = 0.05$ and $\lambda_T = 0.1$.

Table 4.2: Performance comparisons on the IMM, the M³H and the STDMM-PMM filters ($\lambda_T = 0.05$)

Algorithm	RMSE on Position	RMSE on Speed	Computational Time/Run
IMM	102.9m	15.9m/s	20.5ms
M ³ H	97.9m	12.3m/s	76.5ms
STDMM-PMM	96.2m	11.5m/s	79.5ms

Figure 4.8: Model index vs. scan of the IMM, the M^3H and the STDM-PMM filters ($\lambda_T = 0.05$).

Figure 4.9: The target position estimation accuracy comparisons on the IMM, the M^3H and the STDM-PMM filters ($\lambda_T = 0.05$).

The three filters are applied on the test scenario for 500 Monte Carlo runs. For the tests with $\lambda_T = 0.05$, it is observed that the average number of remaining hypotheses per scan is 4.38. The errors on the target position and speed are

Figure 4.10: The target speed estimation accuracy comparisons on the IMM, the M³H and the STD-M-PMM filters ($\lambda_T = 0.05$).

computed based on

$$p_{err,k} = \sqrt{x_{err,k}^2 + y_{err,k}^2}, \quad (4.42)$$

$$sp_{err,k} = \sqrt{\dot{x}_{err,k}^2 + \dot{y}_{err,k}^2}, \quad (4.43)$$

where $p_{err,k}$ and $sp_{err,k}$ are the target position and speed errors at time k respectively, $x_{err,k}$, $y_{err,k}$, $\dot{x}_{err,k}$ and $\dot{y}_{err,k}$ are the estimated position and speed errors in x and y coordinates at time k , respectively. The test results are displayed in Figure 4.8, Figure 4.9, Figure 4.10 and Table 4.2. Figure 4.8 shows the model index vs. scan, and the RMSEs on position and speed vs. scan are shown in Figure 4.9 and Figure 4.10 respectively. Table 4.2 displays the time averaging of RMSEs on all scans and the computational time spent per run. It can be seen that the STD-M-PMM filter outperforms the other two filters which use normal Markov switching. In Figure 4.8, we observe that the IMM filter adapts to the

correct model slower than the other two filters, and these delays create peaks in RMSEs on the position and speed estimation. The STDM-PMM filter responds to the model switching almost immediately, thus its RMSE does not increase as much as that of the IMM filter during model switching periods. The performance of the M^3H filter is in between of the IMM filter and the STDM-PMM filter for this pruning threshold ($\lambda_T = 0.05$).

Figure 4.11: Model index vs. scan of the IMM, the M^3H and the STDM-PMM filters ($\lambda_T = 0.1$).

Table 4.3: Performance comparisons on the IMM, the M^3H and the STDM-PMM filters ($\lambda_T = 0.1$)

Algorithm	RMSE on Position	RMSE on Speed	Computational Time/Run
IMM	103.1m	15.9m/s	20.5ms
M^3H	107.4m	15.1m/s	21.4ms
STDM-PMM	98.0m	12.2m/s	22.5ms

For the tests with $\lambda_T = 0.1$, the average number of remaining hypotheses per scan is reduced to 1.92. The test results are displayed in Figure 4.11,

Figure 4.12: The target position estimation accuracy comparisons on the IMM, the M^3H and the STD-M-PMM filters ($\lambda_T = 0.1$).

Figure 4.13: The target speed estimation accuracy comparisons on the IMM, the M^3H and the STD-M-PMM filters ($\lambda_T = 0.1$).

Figure 4.12, Figure 4.13 and Table 4.3. It can be seen that the performance of the M^3H filter obviously drops when the number of hypotheses is reduced, whereas the performance of the STD-M-PMM filter does not decrease much. It

is still the best among the three algorithms.

The computational times of one run (42 scans) for the three filters are recorded and shown in Table 4.2 and Table 4.3. The experiment is conducted in Matlab operating on a single-core computer. The IMM filter is the most efficient algorithm among the three filters. Both pruning-based filters have similar computational cost, and cost of the STDM-PMM filter is slightly higher than that of the M³H filter. The computational cost of the pruning-based filters depends on the pruning threshold λ_T . Their computational costs are 3 to 4 times comparing to the IMM filter when $\lambda_T = 0.05$. However, their computational cost is similar to the IMM filter when λ_T is increased to 0.1.

From the test results on the generic scenario of ATC, we can see that the STDM-PMM filter outperforms the IMM and the M³H filters in terms of tracking accuracy without increasing the computational cost much. It confirms that the sojourn time dependent transition probabilities represent target behavior better than the fixed transition probability.

4.4 Apply the STDM-PMM Filter on Tracking in the Presence of Outliers

Outliers often occur in detection systems. For example, a radar glint noise due to interference between the reflected radar waves is much larger than a normal measurement noise. The glint noise occurs among the normal radar measurement noise occasionally. From the point of view of statistics, the outlier introduces a heavy-tailed non-Gaussian component in the probability density function (pdf) of detection noise. This non-Gaussian pdf can severely degrade

the accuracy of estimation algorithms which are designed based on Gaussian assumption. This section applies the STDM-PMM Filter to solve this challenging outlier problem in target tracking. The basic idea is that the noise can be described by two models, namely, the normal noise model and the outlier model, and the STDM-PMM filter then operates on these two models to perform target tracking.

4.4.1 Related Work on the Outlier Problem

Research on detection with the presence of outlier has been conducted over the past decades. The achievements include modeling noise with outlier [80] [81] [82] [83] [84] [85] and estimating state based on non-Gaussian noise [86] [87] [88] [89]. A brief review is given here.

In the earlier stage, the detection noise with outlier was modeled by some single heavy-tailed distributions. They include Student-t function and generalized hypergeometric function [80] [81] [82]. These statistical models are complicated and difficult to be implemented in estimation algorithms. A simplified approach with mixture distributions was therefore suggested and used commonly. A mixture model with two zero-mean Gaussians has been proposed by Hewer, Martin and Zeh [83]. In this mixture model, the outlier is represented by a large variance Gaussian $\mathcal{N}(\mathbf{r}; 0, \mathbf{R}^2)$ with a lower coefficient λ , and the normal detection error is described by a small variance Gaussian $\mathcal{N}(\mathbf{r}; 0, \mathbf{R}^1)$ with a higher coefficient $(1 - \lambda)$. The mixture function of detection error $\mathbf{r}$ is

$$f(\mathbf{r}) = (1 - \lambda)\mathcal{N}(\mathbf{r}; 0, \mathbf{R}^1) + \lambda\mathcal{N}(\mathbf{r}; 0, \mathbf{R}^2). \quad (4.44)$$

This Gaussian mixture model has been verified with empirical glint radar noise through QQ-plots analysis. The results demonstrated that the glint noise has been well simulated by the mixture model [83] [84]. A similar alternative was also suggested, where a Laplacian function was used to represent the outlier [85].

Many estimation algorithms have been suggested for the above mentioned non-Gaussian noise models [86] [87] [88] [89]. Masreliez proposed a filter by introducing a nonlinear score function to correct heavy-tailed error [86]. Although the Masreliez filter achieves nearly optimal result, it is difficult to be implemented in real applications due to a complicated convolution process in the score function computation. A more advanced approach suggested by Daeipour and Bar-Shalom formulated the problem as a two-model Markov switching problem, and solved the problem by an IMM filter [87] [88]. Two models were defined in the system, namely, a Gaussian normal noise model and a Gaussian (or Laplacian) outlier model. The IMM filter has model adaptation capability. The model probabilities are updated by the measurement residuals in every time cycle, such that the correct model can be assigned to a higher weight. Simulation results showed that the IMM filter outperformed the score function method [87] [88]. Recently, a Particle Filter (PF) [89] has also been suggested using a single fixed mixture model throughout the whole process. Similar to all other PFs, it has the advantage on non-Gaussian nonlinear estimation through simulating the state by weighted particles. Since a fixed mixture model is used for every time cycle, the PF does not have the model adaptation capability of the IMM filter, and involves high computational cost.

4.4.2 STDMM-PMM Filter on the Outlier Problem and Simulation Results

Now we apply the STDMM-PMM filter to a 2-dimensional radar tracking problem. The two models with different measurement noises are constructed in the STDMM switching system. They are the normal noise model as model 1 and the outlier model as model 2. The embedded Markov transition probability matrix $\mathbf{P}$ is

$$\mathbf{P} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}. \quad (4.45)$$

The probability mass functions (pmf) of sojourn time τ are defined as

$$w_1(\tau) = \alpha_1 e^{-|\tau - \frac{1}{\lambda}|}, \quad (4.46)$$

$$w_2(\tau) = \alpha_2 e^{-|\tau - \frac{1}{1-\lambda}|}, \quad (4.47)$$

where λ is the outlier occurrence rate. α_1 and α_2 are set to particular values, such that both $\sum_{\tau=1}^{\infty} w_1$ and $\sum_{\tau=1}^{\infty} w_2$ are 1.

The transition equation and measurement equation for both models are described as follows:

$$\mathbf{x}_k^n = \mathbf{F}\mathbf{x}_{k-1}^n + \mathbf{G}\mathbf{w}_{k-1}, \quad (4.48)$$

$$\mathbf{z}_k = h(\mathbf{x}_k^n) + \mathbf{r}_k^m, \quad (4.49)$$

where k is the radar scan index; n is the hypothesis index; m is the model index; the state $\mathbf{x}_k = [x_k \ \dot{x}_k \ y_k \ \dot{y}_k]^T$ represents the target positions and velocities in x and y coordinates respectively; the measurement vector $\mathbf{z}_k = [rg_k \ bg_k]^T$

gives the range and the bearing detected by the radar; the transition matrix $\mathbf{F}$ and the gain matrix $\mathbf{G}$ are, respectively,

$$\mathbf{F} = \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{G} = \begin{bmatrix} T^2/2 & 0 \\ T & 0 \\ 0 & T^2/2 \\ 0 & T \end{bmatrix}, \quad (4.50)$$

where T is the radar scan interval, and is set to 1s; the measurement function $h(\mathbf{x}_k)$ is

$$h(\mathbf{x}_k) = \begin{bmatrix} \sqrt{x_k^2 + y_k^2} \\ \tan^{-1}(x_k/y_k) \end{bmatrix}; \quad (4.51)$$

$\mathbf{w}_{k-1}$ is the white Gaussian with covariance $\mathbf{Q}_{k-1}$ to compensate the transition model error,

$$\mathbf{Q}_{k-1} = \begin{bmatrix} \sigma_{\ddot{x}}^2 & 0 \\ 0 & \sigma_{\ddot{y}}^2 \end{bmatrix}, \quad (4.52)$$

where $\sigma_{\ddot{x}}$ and $\sigma_{\ddot{y}}$ are the standard deviations of the target accelerations in x and y coordinates respectively. $\mathbf{r}_k^j$ is the measurement error, which can either be the normal Gaussian error with covariance $\mathbf{R}_k^1$ or the glint Gaussian error with covariance $\mathbf{R}_k^2$. They are

$$\mathbf{R}_k^1 = \begin{bmatrix} \sigma_r^2 & 0 \\ 0 & \sigma_{b1}^2 \end{bmatrix}, \quad \mathbf{R}_k^2 = \begin{bmatrix} \sigma_r^2 & 0 \\ 0 & \sigma_{b2}^2 \end{bmatrix}, \quad (4.53)$$

where σ_r is the range error standard deviation, σ_{b1} and σ_{b2} are the bearing error standard deviations for the normal noise model and the glint noise model respectively. Since the glint noise is normally incurred by bearing measurement

in radar tracking only, the standard deviation of range error σ_r is shared by both models.

The EKF is applied to both models for hypothesis states estimation. In the simulation tests, both filters share the following parameters: $\sigma_{\ddot{x}} = \sigma_{\ddot{y}} = 0.1\text{m/sec}^2$, $\sigma_r = 50\text{m}$, $\sigma_{b1} = 0.5^\circ$ and $\sigma_{b2} = 5^\circ$. The hypotheses pruning threshold is set to $\lambda_T = 0.9$ in the STDM-PMM filter. The final state is a Gaussian sum of the states in the remaining hypotheses.

To demonstrate the performance of the STDM-PMM filter, the IMM filter and the M³H filter are also applied to the problem. Normal Markov switching with fixed transition probabilities is used in the IMM filter and the M³H filter. It is known that a switching system with fixed transition probabilities is a special case of the STDM system, where the model sojourn times are governed by exponential pdfs. The fixed transition probability matrix in the IMM filter and the M³H filter can then be derived as [2] [19]

$$\mathbf{P} = \begin{bmatrix} e^{-\lambda T} & 1 - e^{-\lambda T} \\ 1 - e^{-(1-\lambda)T} & e^{-(1-\lambda)T} \end{bmatrix}, \quad (4.54)$$

where T is the radar scan interval, and λ is the outlier occurrence rate.

Two outlier occurrence rates are investigated in the simulation tests. The first one has a glint noise occurrence rate of $\lambda = 0.05$, and its corresponding average outlier waiting time is 20s. The second one has a larger occurrence rate of $\lambda = 0.20$. The corresponding average waiting time is reduced to 5s. Figure 4.14 and Figure 4.15 show the RMSE on the target position vs. time for the two cases. The average RMSEs for the two occurrence rates are summarized in Table 4.4. The results are obtained from 500 Monte Carlo runs.

Figure 4.14: Target position RMSEs for the IMM, the M³H and the STDMM-PMM filters when outlier occurrence rate is 0.05.

Figure 4.15: Target position RMSEs for the IMM, the M³H and the STDMM-PMM filters when outlier occurrence rate is 0.20.

It can be seen that the two pruning based MM filters (the M³H and the STDMM-PMM filters) outperform the IMM filter. It is observed that some peaks appear in the RMSE results in the IMM filter in Figure 4.14. The peak locations

are related to the occurrence of outliers. The RMSE of the IMM filter suddenly increases when an outlier occurs, whereas, both pruning based MM filters do not have such significant jumps. This is because the estimation results of the two models are obviously different, and this results in that the correct model can be easily recognized by the hypothesis likelihoods. In the tests, the number of remaining hypotheses of the pruning based MM filters is 1, and this remaining hypothesis is the correct hypothesis in most of the time. Thus, the pruning based filters have better estimation accuracy. The estimation result of the IMM filter is a weighted sum of the two models, and its accuracy is decreased by the contribution from the wrong model.

We also observed that the results of the STDM-PMM filter and the M³H filter are similar. The only difference between these two filters is their sojourn time pmfs. The sojourn time pmfs are defined by (4.46) and (4.47) for the STDM-PMM filter, whereas the M³H filter uses exponential sojourn time pmfs. The different pmfs do not lead to significant estimation difference in this problem. This is because hypothesis likelihoods are functions of measurement residuals and STMD interval-transition probabilities. If models are easily distinguished by the measurement residuals, the contribution of the STMD interval-transition probabilities will not be significant.

Table 4.4: Performance comparisons on the IMM, the M³H and the STDM-PMM filters

Algorithm	RMSE on Position $\lambda = 0.05$	RMSE on Position $\lambda = 0.20$	Computational Time/Run
IMM	34.6m	49.9m	22.0ms
M ³ H	29.8m	38.6m	31.9ms
STDM-PMM	29.7m	38.4m	32.8ms

The computational cost is also compared among the STDM-PMM, IMM

and M³H filters. The average times for one run (100 scans) are displayed in Table 4.4. The test is carried out with Matlab on a single-core computer. It can be seen, that the IMM filter requires less computational time. Although the M³H filter and the STDMM-PMM filter need longer time to complete the whole process, they are still fast enough for real-time implementation.

4.5 Conclusion

This chapter proposes a pruning-based STDMM-PMM filter. The proposed STDMM-PMM filter can deploy any sojourn time pdf in its STDMM switching, whereas the commonly used MM filters with fixed model transition probabilities can only work on exponential sojourn time distributions. The recursive hypothesis likelihoods are derived based on STDMM interval-transition probabilities and measurement residuals, and a hypothesis structure updating method is suggested such that the STDMM-PMM filter can be implemented in real-time.

The STDMM-PMM filter has been applied to two tracking problems, namely, maneuvering target tracking and target tracking in the presence of outliers. The simulation results show that the STDMM-PMM filter outperforms two normal Markov switching filters (the IMM filter and the M³H filter) in the maneuvering target tracking problem. The STDMM-PMM filter can adapt to the correct models much faster than the other filters. In the outlier tracking problem, the STDMM-PMM filter outperforms the IMM filter, and provides similar results to the M³H filter. In this problem, the normal noise model and the outlier model can easily be distinguished by hypothesis likelihoods, and wrong model can be pruned easily. Thus, the pruning based MM filters (the STDMM-PMM

filter and the M³H filter) provide better accuracy than the IMM filter, which provides combining results of all models. Since hypothesis likelihood is mainly contributed by the measurement residual in this problem, and the contribution of the sojourn time pmf is not significant. This leads to the similar results of the STDMM-PMM filter and the M³H filter.

The computational complexity of the STDMM-PMM filter is the same as that of the M³H filter, that is $O(n)$, where n is the number of hypotheses before pruning in each time cycle. The computational complexity of the IMM filter is $O(m)$, where m is the number of models in the system. The existing STDMM-based IMM filter has computational complexity $O(k^4)$ [7] [90], where k is the number of time intervals of track duration. Obviously, the STDMM-PMM filter has bounded computational complexity and is possible for real-time applications, whereas the computational cost of the existing STDMM-IMM filter keeps increasing with time and causes excessive cost in real implementation.

Since the proposed STDMM-PMM filter is a pruning based MM filter, unlikely hypotheses are pruned based on a predefined threshold λ_T . A good λ_T should keep the correct hypothesis and prune as many incorrect hypotheses as possible. When correct and wrong hypotheses can be well distinguished by their likelihoods, a high λ_T should be selected. Otherwise, a low λ_T should be used for keeping more hypotheses to maintain the robustness of the system. Furthermore, the STDMM-PMM filter requires that sojourn time pmfs are predefined. If these predefined pmfs are not matched to real problems, good results will not be achieved.

Chapter 5

MM Application on Target

Localization and Antenna

Auto-Calibration (I): Spiral

Antenna System

RF emitter localization is a challenging problem when the beam pattern of the antennas in the detection system are distorted. This problem is conventionally solved by two separate stages, namely, to estimate target bearings through a Direction Finding (DF) algorithm, and to obtain the emitter location from the estimated bearings. The drawback of such a structure is that it is difficult to obtain accurate bearings when antenna beam patterns are distorted, and the bearing error caused by antenna distortion can significantly decrease the localization accuracy, especially when a target is located in longer range. Thus, we propose a new approach to merge the two stages, namely, the DF and the

localization, into one discrete dynamic estimation process, such that target localization and antenna auto-calibration can be conducted simultaneously. A multiple model UKF is suggested to estimate the state for this new approach. We apply the new approach to different antenna systems. This chapter focuses on a spiral antenna system, and a two-array antenna system will be addressed in Chapter 6.

This chapter is organized as follows. Section 5.1 introduces the problem of obtaining RF location when antenna distortion exists. Section 5.2 discusses the spiral antenna system and the existing two-stage localization method, which includes the amplitude comparison DF algorithm and bearing-only localization algorithm. Section 5.3 formulates the localization problem with unknown antenna bias as a dynamic estimation problem. The state transition equation and the measurement equation are defined. The observability analysis for the problem is also given. Section 5.4 presents the Augmented UKF (A-UKF) and the Multiple Model UKF (MM-UKF) algorithms for dynamic estimation. Section 5.5 shows the simulation results, which consist of comparisons on the A-UKF, the MM-UKF and the conventional two-stage localization algorithm. The conclusion is drawn in Section 5.6.

5.1 Introduction

Passive localization has a wide range of civilian and military applications, and has many varieties. The system can consist of a single sensor or multiple sensors on moving or stationary platforms. The target can also be either moving or stationary. The problem addressed in this chapter is to locate a stationary

RF emitter from a moving passive receiver, which observes the emitter by an antenna system.

The conventional approach to solve this problem usually consists of two sequential stages. The first stage estimates the target Direction Of Arrival (DOA) or bearing through a Direction Finding (DF) algorithm [91] [92] [93] [94]. The second stage obtains the target location from the estimated bearings [95] [96] [97] [98] [99]. Since a target is normally located in far range in many military applications, the location accuracy is highly related to the bearing accuracy estimated by a DF algorithm. It is known that the performance of a DF algorithm depends on the accuracy of the antenna beam pattern, which usually contains unknown bias. Since the existing DF algorithms estimate bearing from data in one time cycle, it is difficult or almost impossible to cancel antenna bias. Thus, more advanced localization algorithms, which are able to cope with antenna bias, need to be investigated.

The antenna system of interest consists of a few directional spiral antennas deployed in the bearing space. The antenna beam patterns are assumed to be known from a priori knowledge. Thus, the bearing can be obtained from the comparison of the signals received from different antenna elements. Many techniques, such as amplitude comparison, phase comparison, and some hybrids have been developed over the past decades. We focus on the amplitude comparison technique in this chapter [91] [92] [93] [94].

The amplitude comparison method computes the target bearing from a ratio of signal powers received by antennas based on predetermined antenna beam patterns. The ratio is obtained from the two strongest signals among the signals received from all antenna elements. The two antenna elements with the

strongest signals are usually adjacent in the antenna system. Although the amplitude comparison method is conceptually simple, it faces difficulties in real applications. An actual antenna beam pattern may not match to the predetermined antenna beam pattern exactly. This is because that distortion, due to antenna manufacturing, system installation and some online reasons, is often encountered. One method to cancel the distortion is to carry out calibration off-line. However, the calibration process usually incurs high cost, and re-calibration is required when bias shifts with time [100].

To overcome the challenges mentioned above, we suggest a new approach, which formulates the passive localization problem as a dynamic estimation problem, and applies nonlinear tracking algorithms to the problem. The proposed approach merges the two stages, namely, the DF and the target localization, in one step. It estimates target location and antenna bias from amplitudes received by antenna system recursively. The dynamic system consists of an augmented state with target location and antenna bias, and a measurement obtained from the amplitudes received. The relationship between the state and the measurement is linked by a nonlinear measurement equation, which is defined by biased antenna beam patterns. The advantage of the new approach is that the antenna distortion can be auto-calibrated online.

There are two basic estimation approaches, namely, a static batch approach and a dynamic recursive approach. The batch approach estimates the stationary parameters from all the measurements received in the whole process. Although the batch process can provide the optimal solution for the given batch of measurements, it often incurs high computational cost. The recursive approach updates the estimation parameters recursively by the measurement received in

sequence. It is more efficient and can provide real-time estimation output. In this research, we select the recursive estimation approach.

Figure 5.1 illustrates a comparison between the existing two-stage method and the new proposed approach in one time cycle. The inputs of the both approaches are the amplitudes received by the antennas. The outputs of the two-stage method is the emitter location only, whereas, the new approach provides the emitter location and the antenna bias parameters. The off-line antenna calibration is required to obtain acceptable bearing estimation accuracy in the two-stage method. The new approach has no such requirement. The two-stage method is a suboptimal solution, as bearing estimation and location estimation are separated, and only a local optimum can be provided for each separated step. The new approach can provide the optimal solution as the two separated estimation processes are merged into one.

Figure 5.1: The existing two-stage method and the new approach

Since the proposed approach uses nonlinear antenna beam pattern to define its measurement equation, the problem has to be solved by nonlinear estimation

algorithms. One of challenging problems in nonlinear estimation is the track initiation. The accuracy of the initial state is important to a nonlinear estimation problem, as a wrong initial value can mislead algorithms into converging to a local optimum instead of the global optimum, spends longer time in convergence if the problem is observable. To achieve a good initial state, optimization algorithms are usually applied to a batch of measurements collected in the first few time cycles. A typical method is to formulate the optimization problem by the Maximum Likelihood (ML) method, and then to solve the problem using Newton's algorithm iteratively [3] [71]. Another method is called the Markov Chain Monte Carlo (MCMC) method [72] [101], which is a sampling method based on the likelihood linked to the measurement equation. Both methods find the initial state through iterations until some predefined conditions are met. There are two drawbacks of these existing approaches. Firstly, the iteration process introduces high computational cost. Secondly, since a rapid convergence to the global optimum requires an accurate initial state, the initial batch size must be large enough such that the initial state can reach the accuracy requirement. The large batch size can result in an obvious initiation delay.

We propose a static MM approach to deal with the state initiation problem. The static MM approach was suggested by Magill and Lainiotis [42] [102] for adaptive optimal estimation. Here, we construct multiple models with a set of initial model states computed from the representative antenna bias. The initial states can either be accurate or inaccurate. The state and the probability of each model are estimated by the UKF. The final state is then a weighted sum of all the model states. The computational complexity of the MM approach is $O(n)$, where n is the number of models in the system. However, the computational complexities of those iterative algorithms are difficult to be identified as the

number of iterations cannot be predetermined. Furthermore, the MM adapts model probabilities throughout the whole process, where models with accurate initial states are assigned higher weights, and inaccurate models are given lower weights. It is not required to make effort on determining the size of batch to obtain an acceptable initial state.

5.2 Spiral Antenna System and Two-stage Estimation Method

This section introduces the spiral antenna system and the conventional two-stage localization algorithm.

5.2.1 Spiral Antenna System

Spiral antenna systems are widely used in microwave direction finding applications such as surveillance, Radar Warning Receiving (RWR), communication, etc. We use the following example to illustrate the use of a spiral antenna system for direction finding. The system consists of four antenna elements uniformly distributed in bearing space. Their beam patterns are shown in Figure 5.2. The squinted angle θ_s , which is defined as half of the angle between two adjacent antennas' facing directions, is 45° . The facing directions for the four antennas are 45° , 135° , -135° and -45° . Each pair of adjacent antennas monitors a 90° sector between their facing directions. For example, the bearing of the target in Figure 5.2 is estimated by the amplitudes received from antenna 1 and antenna 2, which are represented in solid lines. The four antenna beam patterns in the

space of amplitude vs. bearing are displayed in Figure 5.3. Figure 5.4 shows the same information by converting amplitude to dB, which is usually used in a DF algorithm.

Figure 5.2: Antenna beam patterns in a four-antenna system

Figure 5.3: Relative antenna amplitude vs. bearing

The spiral antenna beam pattern is the Gaussian-shaped pattern. A typical

Figure 5.4: Relative antenna amplitude in dB vs. bearing

Gaussian-shaped pattern is given by

$$a_i(\theta) = A_i \exp \left[-2.77 \left(\frac{\theta - \theta_i}{\theta_{Bi}} \right)^2 \right], \quad i = 1, 2, 3, 4 \quad (5.1)$$

where θ is the bearing angle, A_i , θ_i and θ_{Bi} are the relative amplitude of peak, the facing direction and the half-power beam-width for the i th antenna element, respectively. Converting (5.1) to dB, we obtain

$$\begin{aligned} g_i &= G_i - 10 \times 2.77 \log_{10}(e) \left(\frac{\theta - \theta_i}{\theta_{Bi}} \right)^2 \\ &\approx G_i - 12.04 \left(\frac{\theta - \theta_i}{\theta_{Bi}} \right)^2, \end{aligned} \quad (5.2)$$

where

$$G_i = 10 \log_{10}(A_i). \quad (5.3)$$

G_i becomes 0, when relative peak A_i is set to 1.

5.2.2 Two-stage Localization Algorithm

The two-stage localization algorithm estimates a target location in two separate stages, namely, to estimate target bearing using the DF algorithms, and to find the target location using the bearing-only localization algorithms.

5.2.2.1 Conventional DF Algorithm

The DF algorithm discussed here is based on amplitude comparison. The bearing of a target is computed from the gain difference of two adjacent antennas. Assuming the target is located in the sector from 45° to 135° , the gain difference Δg is

$$\begin{aligned}\Delta g &= g_1 - g_2 \\ &= G_1 - G_2 \\ &\quad - 12.04 \left[\left(\frac{\theta - \theta_1}{\theta_{B1}} \right)^2 - \left(\frac{\theta - \theta_2}{\theta_{B2}} \right)^2 \right].\end{aligned}\tag{5.4}$$

Equation (5.4) can be further simplified to a linear function if two squinted antennas are identical. It means that the two antenna beam patterns have the same peak gain G and the same beam-width θ_B . The linear function is

$$\Delta g = c\theta + d,\tag{5.5}$$

where

$$c = -24.08 \frac{\theta_2 - \theta_1}{\theta_B^2},\tag{5.6}$$

$$d = 12.04 \frac{\theta_2^2 - \theta_1^2}{\theta_B^2}.\tag{5.7}$$

The target bearing can then be easily obtained by

$$\theta = \frac{\Delta g - d}{c}. \quad (5.8)$$

Note that the bearing estimated by (5.8) is based on the assumption that the two antenna beam patterns perfectly match the predefined pattern. The estimated bearing will not be accurate if real antenna beam patterns are biased.

5.2.2.2 Bearing-Only Localization Algorithm

The bearings estimated by the DF algorithm along a platform trajectory are triangulated to obtain the target location. Many algorithms have been proposed in the literature to find the best target location. The Stansfield algorithm [95] and the Maximum Likelihood (ML) algorithm [96] are the two most popular ones among them. Each of them estimates the target location through a batch process on all the obtained bearings. A comparison of these two algorithms has been made by Gavish and Weiss [103]. It concluded that the Stansfield algorithm is better than the ML algorithm in terms of RMSE, although it is a biased estimator. Research on localization using a recursive EKF has also been conducted [99]. The recursive method is more suitable for real-time applications, but the estimation performance depends on the accuracy of the initial value.

In this chapter, we utilize the following two-stage algorithm for performance comparison in simulation tests. The algorithm uses a batch based algorithm to compute an initial target location from the first few bearings, and then to update the subsequent states through a recursive algorithm. Firstly, the target bearings are estimated by (5.8), and secondly the bearing-only localization is

performed by a UKF recursively. The state to be estimated is defined as $\mathbf{x} = [x \ y]^T$, where x and y are the target locations respectively. The measurement is the bearing θ . The transition equation and the measurement equation are, respectively,

$$\mathbf{x}_k = \mathbf{I}_2 \mathbf{x}_{k-1} + \mathbf{w}_{k-1}, \quad (5.9)$$

$$\theta_k = \arctan\left(\frac{x_k - x_{p,k}}{y_k - y_{p,k}}\right) + r_k, \quad (5.10)$$

where k is the time index, $\mathbf{I}_2$ is the 2×2 identity matrix, x_p and y_p are the moving platform locations respectively, r_k is the measurement noise, and $\mathbf{w}_{k-1}$ is the process noise. $\mathbf{w}_{k-1}$ can be set to $\mathbf{0}$ when the emitter is stationary. The state is initiated by the Stansfield algorithm.

5.3 Problem Formulation and Analysis

This section formulates the RF emitter geolocation problem as a dynamic estimation problem, which consists of a state transition equation and a measurement equation. The observability analysis and CRLB of the problem are also given. Since the antenna bias is included in the dynamic models, we discuss the bias before addressing the problem formulation.

5.3.1 Antenna Bias and Biased Pattern

The bias we considered in the Gaussian-shaped antenna beam pattern is the channel unbalance, which is the difference of peak gains in dB between the two

operating antennas. It is

$$b_G = G_1 - G_2, \quad (5.11)$$

where G_1 and G_2 are defined in (5.3). Putting (5.11) in (5.4), we obtain

$$\begin{aligned} \Delta g &= b_G - 12.04 \left(\frac{\theta - \theta_1}{\theta_B} \right)^2 + 12.04 \left(\frac{\theta - \theta_2}{\theta_B} \right)^2 \\ &= b_G + c\theta + d, \end{aligned} \quad (5.12)$$

where c and d are defined in (5.6) and (5.7) respectively.

It is clear that target bearing θ cannot be obtained from Equation (5.12) alone, when bias is unknown to the observer. The conventional DF algorithm, which computes the bearing by ignoring the bias, can only provide inaccurate bearing information. Thus, a more advanced approach needs to be developed.

5.3.2 Problem Formulation

A dynamic system has the following components: the state, the measurement, the state transition equation and the measurement equation. In our problem, the augmented state consists of target location and bias. It is

$$\mathbf{x} = [x \ y \ b_G]^T, \quad (5.13)$$

where x and y are the target positions in x and y coordinates respectively, b_G is the antenna bias defined in (5.11). The transition equation can be written in a linear form,

$$\mathbf{x}_k = \mathbf{F}\mathbf{x}_{k-1} + \mathbf{w}_{k-1}, \quad (5.14)$$

where k is the time index, $\mathbf{F}$ is the 3×3 identity matrix to perform a stationary transition, and $\mathbf{w}_{k-1}$ is the process noise, which can be set to $\mathbf{0}$ for a stationary model. In our problem, the target is stationary. If we assume that the variation of the antenna beam patterns is not obvious during the observation period, the stationary model is true in general.

The platform state vector $\mathbf{x}_p$ is defined as

$$\mathbf{x}_p = [x_p \ y_p \ 0]^T, \quad (5.15)$$

where x_p and y_p are the platform positions in x and y coordinates respectively. The relative state vector is defined as

$$\begin{aligned} \mathbf{x}_r &= \mathbf{x} - \mathbf{x}_p \\ &= [x - x_p \ y - y_p \ b_G]^T. \end{aligned} \quad (5.16)$$

The measurement of the dynamic system, z , is the gain difference of the two antennas. The measurement equation is

$$z_k = h(\mathbf{x}_{r,k}) + r_k, \quad (5.17)$$

where r_k is the white Gaussian noise with variance R . Putting (5.16) into (5.12), we obtain

$$h(\mathbf{x}_{r,k}) = \mathbf{x}_{r,k}(3) + c\theta_k + d, \quad (5.18)$$

where $\mathbf{x}_{r,k}(i)$ stands for the i th element in $\mathbf{x}_{r,k}$. The bearing θ_k is given by

$$\theta_k = \arctan\left(\frac{\mathbf{x}_{r,k}(1)}{\mathbf{x}_{r,k}(2)}\right). \quad (5.19)$$

It is seen that the transition equation is a linear function, but the measurement equation is a nonlinear equation. Thus, a nonlinear tracking algorithm is required to implement the dynamic system estimation.

5.3.3 Cramer-Rao Lower Bound

This section derives the PCRLB for the problem. For a generic nonlinear discrete system

$$\mathbf{x}_k = f(\mathbf{x}_{k-1}) + \mathbf{w}_{k-1}, \quad (5.20)$$

$$\mathbf{z}_k = h(\mathbf{x}_k) + \mathbf{r}_k, \quad (5.21)$$

the PCRLB recursion is

$$\mathbf{J}_k = [\mathbf{F}_{k-1} \mathbf{J}_{k-1}^{-1} \mathbf{F}_{k-1}^T + \mathbf{Q}_{k-1}]^{-1} + \mathbf{H}_k^T \mathbf{R}_k^{-1} \mathbf{H}_k, \quad (5.22)$$

where $\mathbf{J}_k$ is the information matrix for the estimated state vector $\mathbf{x}_k$, $\mathbf{F}_k$ is the Jacobian matrix of $f(\mathbf{x}_{k-1})$, $\mathbf{H}_k$ is the Jacobian matrix of $h(\mathbf{x}_k)$, $\mathbf{Q}_{k-1}$ and $\mathbf{R}_k$ are the process noise covariance and measurement noise covariance respectively. $\mathbf{Q}_{k-1}$ can be considered as $\mathbf{0}$. $\mathbf{F}_{k-1}$ in our problem is an identity matrix, and $\mathbf{H}_k$ is derived as

$$\mathbf{H}_k = \begin{bmatrix} \frac{c(y_k - y_{p,k})}{r_k^2} & -\frac{c(x_k - x_{p,k})}{r_k^2} & 1 \end{bmatrix}, \quad (5.23)$$

where

$$r_k^2 = (x_k - x_{p,k})^2 + (y_k - y_{p,k})^2.$$

The recursive information matrix $\mathbf{J}_k$ is then simplified as

$$\mathbf{J}_k = \mathbf{J}_{k-1} + \mathbf{H}_k^T \mathbf{R}_k^{-1} \mathbf{H}_k. \quad (5.24)$$

The PCRLB of the emitter position pos and the bias parameter b_G at time k can be written as

$$PCRLB_{pos,k} = \sqrt{\mathbf{J}_k^{-1}(1,1) + \mathbf{J}_k^{-1}(2,2)}, \quad (5.25)$$

$$PCRLB_{b_G,k} = \sqrt{\mathbf{J}_k^{-1}(3,3)}. \quad (5.26)$$

It can be easily derived that the initial value of the recursive information matrix $\mathbf{J}_1$ is the inverse of the initial state error covariance matrix $\mathbf{P}_1$. With $\mathbf{J}_1$, $\mathbf{R}_k$ and $\mathbf{H}_k$, the PCRLB of pos and b_G can be computed by (5.25) and (5.26) recursively.

5.3.4 Observability Analysis

In dynamic system theory, a nonlinear system is locally observable, if the linearization of the system is observable. The nonlinear system in our problem can be linearized as

$$\mathbf{x}_k = \mathbf{F}_{k-1} \mathbf{x}_{k-1}, \quad (5.27)$$

$$z_k = \mathbf{H}_k \mathbf{x}_k, \quad (5.28)$$

where $\mathbf{F}_{k-1}$ is defined as an identity matrix, and $\mathbf{H}_k$ is the Jacobian matrix of $h(\mathbf{x}_k)$, which has been derived in (5.23). The observability of a linear system can be analyzed by Theorem 5.3.1.

Theorem 5.3.1. *A linear discrete system is observable, if its observability matrix $\mathbf{O}$ is linearly independent, namely, $\text{rank}(\mathbf{O}) = n$, that is*

$$\text{rank}(\mathbf{O}) = \text{rank} \left(\begin{bmatrix} H \\ HF \\ \vdots \\ HF^{n-1} \end{bmatrix} \right) = n, \quad (5.29)$$

where $F \in \mathbb{R}^{n \times n}$ and $H \in \mathbb{R}^{1 \times n}$ [104].

Given any three adjacent time cycles k , $k+1$ and $k+2$, the observability matrix of the linear system defined in (5.27) and (5.28) can be derived as

$$\begin{aligned} \mathbf{O} &= \begin{bmatrix} \frac{c(y - y_{p,k})}{r_k^2} & -\frac{c(x - x_{p,k})}{r_k^2} & 1 \\ \frac{c(y - y_{p,k+1})}{r_{k+1}^2} & -\frac{c(x - x_{p,k+1})}{r_{k+1}^2} & 1 \\ \frac{c(y - y_{p,k+2})}{r_{k+2}^2} & -\frac{c(x - x_{p,k+2})}{r_{k+2}^2} & 1 \end{bmatrix} \\ &= \begin{bmatrix} cy_a/r_a^2 & -cx_a/r_a^2 & 1 \\ cy_b/r_b^2 & -cx_b/r_b^2 & 1 \\ cy_c/r_c^2 & -cx_c/r_c^2 & 1 \end{bmatrix}, \end{aligned} \quad (5.30)$$

where

$$\begin{aligned} x_a &= x - x_{p,k}, & y_a &= y - y_{p,k}, & r_a &= r_k, \\ x_b &= x - x_{p,k+1}, & y_b &= y - y_{p,k+1}, & r_b &= r_{k+1}, \\ x_c &= x - x_{p,k+2}, & y_c &= y - y_{p,k+2}, & r_c &= r_{k+2}. \end{aligned}$$

The determinant of $\mathbf{O}$ is

$$\begin{aligned} \det(\mathbf{O}) = & \frac{c^2}{r_a^2 r_b^2} (x_a y_b - x_b y_a) + \frac{c^2}{r_a^2 r_c^2} (x_c y_a - x_a y_c) \\ & + \frac{c^2}{r_b^2 r_c^2} (x_b y_c - x_c y_b). \end{aligned} \quad (5.31)$$

We assume that the platform is moving in a constant velocity model, or the platform moving pattern can be approximated to a constant velocity model in the short period from time k to $k+2$. Then x_b , y_b , x_c and y_c can be written as

$$\begin{aligned} x_b &= x_a + T\dot{x}, & y_b &= y_a + T\dot{y}, \\ x_c &= x_a + 2T\dot{x}, & y_c &= y_a + 2T\dot{y}, \end{aligned} \quad (5.32)$$

where T is the time interval, $\dot{x}$ and $\dot{y}$ are the platform velocities in x and y coordinates respectively. Substituting (5.32) to (5.31), we obtain

$$\det(\mathbf{O}) = \frac{c^2 T}{r_a^2 r_b^2 r_c^2} (x_a \dot{y} - y_a \dot{x}) (r_a^2 + r_c^2 - 2r_b^2). \quad (5.33)$$

Figure 5.5: Geometric relationship of the emitter and the platform moving in a constant velocity model. (a) The three adjacent points locate on both sides of $\overline{OH}$. (b) The three adjacent points locate on one side of $\overline{OH}$.

Now we prove that $r_d = (r_a^2 + r_c^2 - 2r_b^2) \neq 0$. Figure 5.5 illustrates the

geometric relationship of the emitter and the platform moving in a constant velocity model for two possible cases. Case (a) shows the three adjacent points located on both sides of $\overline{OH}$, whereas, case (b) shows the three adjacent points located on one side of $\overline{OH}$. From the law of cosines, we obtain that

$$r_a^2 - r_b^2 = 2r_a s \cos \theta_1 - s^2, \quad (5.34)$$

$$r_c^2 - r_b^2 = 2r_c s \cos \theta_2 - s^2. \quad (5.35)$$

Adding (5.34) and (5.35), r_d is derived as

$$r_d = 2s(r_a \cos \theta_1 + r_c \cos \theta_2 - s) \quad (5.36)$$

$$= \begin{cases} 2s(\overline{AH} + \overline{CH} - s), & \text{case(a)} \\ 2s(\overline{AH} - \overline{CH} - s), & \text{case(b)} \end{cases} \quad (5.37)$$

$$= 2s(2s - s) \quad (5.38)$$

$$= 2s^2. \quad (5.39)$$

Substituting (5.39) into (5.33), we obtain

$$\det(\mathbf{O}) = \frac{2c^2 s^2 T}{r_a^2 r_b^2 r_c^2} (x_a \dot{y} - y_a \dot{x}). \quad (5.40)$$

Since $\frac{2c^2 s^2 T}{r_a^2 r_b^2 r_c^2} > 0$, the unique solution for $\det(\mathbf{O}) = 0$ is that $(x_a \dot{y} - y_a \dot{x}) = 0$, namely,

$$\frac{x_a}{y_a} = \frac{\dot{x}}{\dot{y}}. \quad (5.41)$$

This means that the platform bearing θ_B and heading angle θ_H must satisfy $\theta_B - \theta_H = k\pi$, $k \in \{0, 1, 2, \dots\}$. Under this condition, the platform trajectory must be moving towards or away from the RF emitter directly. In other words,

the system is observable unless the angle of arrival is constant over time.

5.4 Estimation Algorithms

This section proposes two estimation algorithms, namely, the A-UKF and the MM-UKF, for the dynamic system formulated in Section 5.3.

5.4.1 Augmented UKF

The A-UKF estimates the augmented state defined in (5.13) using a single model UKF. As we discussed in Chapter 2, the UKF uses a set of sigma points to represent the state pdf. These sigma points are transited through the nonlinear transformation functions. The mean and the covariance of the posterior state are computed through the transited sigma points. The number of sigma points is $2n_{\mathbf{x}} + 1$, where $n_{\mathbf{x}}$ is the number of elements in the state vector. In our nonlinear model, $n_{\mathbf{x}}$ is 3, and the number of sigma points is 7. We do not repeat the estimation steps of the UKF here. They have been described in Section 2.2.4.

The initial state needs to be specified in the UKF. It consists of the antenna bias b_G and the emitter position. Since the bias b_G is unknown to the observer, it is set to mean value zero for initial state calculation. The initial target location (x, y) is approximately obtained by the conventional DF and the Stansfield algorithms. The DF algorithm computes two bearings from the measurements at time 1 and time l , and the Stansfield algorithm estimates the target location from these two bearings, where l is an integer and $l > 1$. The value of l is selected such that the difference between the two estimated bearings, θ_1 and θ_l ,

is greater than a certain angle. This is because the small angle between θ_1 and θ_l indicates these two bearing lines are almost parallel, and this will result in large position error of the triangulation point. Once the initial state is decided, the recursive UKF performs state estimation using the data received from time 2 to time $l - 1$, and subsequently updates the state after time l .

The initial state mentioned above is computed based on the assumption that the antenna beam pattern is unbiased. Obviously, an initial error is introduced when the antenna beam pattern is biased. Although an inaccurate initial point can converge to the optimal eventually in this observable problem, a larger initial error will lead a longer convergence time. To shorten this convergence time, a more robust algorithm should be developed.

5.4.2 Multiple Model UKF

The unknown antenna bias results in uncertainty on the initial state. A multiple model approach constructed by multiple initial states is suggested to solve this uncertainty problem. The number of representative biases, n , is selected to compute a set of initial states $\mathbf{x}_1^1, \mathbf{x}_1^2, \dots, \mathbf{x}_1^n$. The multiple models are constructed by this set of initial states with the same transition function and measurement function. The hybrid system is described as

$$Model^1 \quad \begin{cases} \mathbf{x}_k^1 = \mathbf{I}_3 \mathbf{x}_{k-1}^1, \\ \mathbf{x}_{r,k}^1 = \mathbf{x}_k^1 - \mathbf{x}_{p,k}, \\ z_k = h(\mathbf{x}_{r,k}^1) + r_k, \end{cases} \quad (5.42)$$

$$\begin{aligned}
Model^2 \quad & \begin{cases} \mathbf{x}_k^2 = \mathbf{I}_3 \mathbf{x}_{k-1}^2, \\ \mathbf{x}_{r,k}^2 = \mathbf{x}_k^2 - \mathbf{x}_{p,k}, \\ z_k = h(\mathbf{x}_{r,k}^2) + r_k, \end{cases} \\
& \vdots
\end{aligned} \tag{5.43}$$

$$\begin{aligned}
Model^n \quad & \begin{cases} \mathbf{x}_k^n = \mathbf{I}_3 \mathbf{x}_{k-1}^n, \\ \mathbf{x}_{r,k}^n = \mathbf{x}_k^n - \mathbf{x}_{p,k}, \\ z_k = h(\mathbf{x}_{r,k}^n) + r_k, \end{cases}
\end{aligned} \tag{5.44}$$

where I_3 is the 3×3 identity matrix; $\mathbf{x}_{p,k}$ is the platform state defined in (5.15); $h(\cdot)$ is the measurement function defined in (5.17).

The static MM estimator described in Section 2.3.1 is applied to the hybrid system. Each cycle of the static MM estimator consists of three steps, namely, filtering, model likelihood evaluation, and combination. In the filtering step, n UKF filters run in parallel to estimate the states with different initial values. In the second step, the likelihood of each model is computed. Finally, the hybrid state, which is the final output of the cycle, is combined from the individual model states.

The models with the relatively correct initial states will be assigned higher weights through the MM approach automatically, whereas, those models with wrong initial states will have lower weights. This adaptation makes a fast convergence rate comparing to the single model. It also prevents estimation from divergence or converging into a local optimum.

5.5 Simulation and Analysis

This section demonstrates the performance of the proposed algorithms through simulation experiments. The A-UKF, the MM-UKF and the conventional two-stage localization algorithm are applied to a test scenario for comparison. In this section, we firstly describe the scenario, and then construct the multiple models by representative biases. Simulation results and analysis are given in the last part.

5.5.1 Scenario

The test scenario is shown in Figure 5.6. A stationary RF emitter is located at $(0\text{m}, 0\text{m})$, and a platform located at $(-25000\text{m}, -25000\text{m})$ moves from south to north with a speed of 300m/s . The total duration is 120s .

Figure 5.6: The test scenario for computer simulation

A RF receiver deployed on the platform has four antenna elements, which are

uniformly distributed in 360° bearing space. The design values of four antenna facing directions are $\theta_1 = 45^\circ$, $\theta_2 = 135^\circ$, $\theta_3 = -135^\circ$, and $\theta_4 = -45^\circ$. The squinted angle θ_s is 45° . The beam-width θ_B and the relative peak gain G for the four antennas are 75° and 0dB respectively. The measurement noise on relative antenna gain is assumed as white Gaussian with standard deviation 0.5dB. The receiver processing time interval T is 1s. Since the bearing of the emitter with respect to the platform is within a range from 45° to 135° , the measurements of antenna 1 and antenna 2 are used to estimate the target location.

5.5.2 Multiple Model Construction

The multiple models are constructed by a set of representative antenna biases. If we assume that the peak gain G is varied from -2dB to 2dB, the range of bias b_G is

$$b_G \in [-4dB, 4dB]. \quad (5.45)$$

Five models, M^1 , M^2 , ..., M^5 , which have five initiation states $\mathbf{x}_1^1$, $\mathbf{x}_1^2$, ..., $\mathbf{x}_1^5$, are defined based on five b_G values. The five b_G values are -4dB, -2dB, 0dB, 2dB and 4dB. The bearings at time 1 and time l for model j ($j \in \{1, \dots, 5\}$) can be estimated by

$$\theta_1^j = \frac{\Delta g_1 - d - b_G^j}{c}, \quad (5.46)$$

$$\theta_l^j = \frac{\Delta g_l - d - b_G^j}{c}, \quad (5.47)$$

where c and d are defined in (5.6) and (5.7). Given bearings θ_1^j and θ_l^j , and the corresponding platform locations $[x_{p,1}, y_{p,1}]$ and $[x_{p,l}, y_{p,l}]$, the emitter initial location $[x_1^j, y_1^j]$ can be computed using the Stansfield algorithm. The initial

state for model j is

$$\mathbf{x}_1^j = [x_1^j \ y_1^j \ b_G^j]^T. \quad (5.48)$$

The five multiple models with the five initial states are then completely constructed.

5.5.3 Results and Analysis

Simulation tests are conducted on the scenario described in Figure 5.6. Three algorithms, namely, the conventional two-stage localization algorithm, the A-UKF and the MM-UKF are tested on 9 bias levels. In these 9 bias levels, b_G value is set from -4dB to 4dB with 1dB interval. l is set to 15. The initial state error covariance $\mathbf{P}$ is set as a 3×3 diagonal matrix with diagonal elements $[7000^2 \ 7000^2 \ 1]$. The measurement error variance R is 1. The UKF parameter κ is set to 1. Each test is carried out with 100 Monte Carlo runs. The target position RMSE on Monte Carlo runs is recorded. The position RMSE is computed based on (3.73). The averages of the position RMSE on the whole duration (120 cycles) and the final position RMSE at the last cycle (time index 120) are shown in Table 5.1 and Table 5.2 respectively. The same results are presented in graphic form in Figure 5.7 and Figure 5.8.

It is obvious that the MM-UKF generates the smallest overall position RMSE among the three algorithms, whereas the conventional two-stage algorithm produces the largest RMSE. The performance of the A-UKF is in between the other two algorithms. It is also observed that the MM-UKF is more stable to different bias levels, whereas the performance of the two-stage method and the A-UKF vary with bias. The RMSEs of these two algorithms increase with bias. Since

Table 5.1: Time average of target position RMSE on different antenna peak bias b_G

$b_G(dB)$	Position RMSE average (m)		
	2-stage	A-UKF	MM-UKF
-4	6185.6	4558.2	2649.3
-3	4851.7	3621.4	2108.6
-2	3232.3	2467.6	1916.3
-1	1888.0	1554.8	1662.5
0	698.3	928.0	1481.9
1	1679.3	1343.7	1611.9
2	2985.9	2157.6	1902.9
3	4340.4	3176.9	2215.7
4	5783.6	4200.0	2781.5

Table 5.2: Target position RMSE at last cycle (time Index 120) on different antenna peak bias b_G

$b_G(dB)$	Position RMSE at time index 120 (m)		
	2-stage	A-UKF	MM-UKF
-4	5188.6	1033.7	456.1
-3	3891.9	780.9	470.0
-2	2600.9	659.7	484.7
-1	1312.4	471.8	458.8
0	183.6	376.8	423.4
1	1308.6	470.7	443.5
2	2580.6	677.2	447.0
3	3862.1	1092.1	414.6
4	5118.6	1563.8	482.7

the two-stage algorithm does not take bias into consideration, this causes the position error to increase with bias. As for the A-UKF, although it can adapt to the bias, its performance depends on the accuracy of the initial state. Since the A-UKF is initiated based on zero bias, the initial error is increased when

Figure 5.7: Time average of target position RMSE vs. antenna peak bias b_G

Figure 5.8: Target position RMSE at last cycle (time index 120) vs. antenna peak bias b_G

the bias is large, and this results in an increase in the estimation error. The MM-UKF overcomes the nonlinear initiation problem through a hybrid system with multiple initiation states. Therefore, better estimation is achieved.

We also notice that the target position RMSE of the MM-UKF is slightly larger than the other two algorithms when the bias is zero or near zero. It is understood that a single accurate model is better than multiple models which consist of inaccurate models. The model in the two-stage method is accurate, and the A-UKF has an accurate initial state, when the bias is zero. The MM-UKF consists of five models, where only M^3 is initiated by zero bias, and the other models are inaccurate. It is not doubted that the MM-UKF cannot outperform these two single model algorithms when the bias is zero or near zero. However, the unbiased antenna beam patterns defined in (5.2) are obtained based on the theoretical exponential function, which cannot be exactly synthesized. It is impossible to evade the antenna bias since they do exist in real systems. Thus, the MM-UKF is still a superior algorithm.

Figure 5.9: Target position RMSE vs. time, when the antenna peak bias $b_G = 4dB$

Target position RMSE vs. time is also investigated in the simulation tests. Figure 5.9, Figure 5.10 and Figure 5.11 show the results obtained when the

Figure 5.10: Target position RMSE vs. time, when the antenna peak bias $b_G = 3dB$

Figure 5.11: Target position RMSE vs. time, when the antenna peak bias $b_G = 0dB$

antenna peak bias b_G is set to 4dB, 3dB and 0dB respectively. We can see that the MM-UKF obviously outperforms the A-UKF, and the A-UKF is much better than the two-stage algorithm when the bias is not close to zero, which

are the cases demonstrated in Figure 5.9 and Figure 5.10. However, the MM-UKF yields worse performance when the bias is near zero, which is presented in Figure 5.11. The reason has been given in the preceding paragraph. Here we notice that although the MM-UKF has slower convergence than the other two single model algorithms, the final position RMSE is still comparable to that of the two single models.

Figure 5.12: Model probabilities when $b_G = 4\text{dB}$

The model probability behavior in the MM-UKF is also studied in the simulation tests. Five model probabilities vs. time are shown in Figure 5.12 and Figure 5.13 for two particular biases. In Figure 5.12, the antenna peak bias b_G is set to 4dB. As expected, the model M^5 , whose state is initiated based on $b_G = 4\text{dB}$, shows the highest probability. Its closest model M^4 has the second highest probability. The probabilities of the other models decrease to zero or a small value. Figure 5.13 shows the result obtained with the antenna peak bias $b_G = -1\text{dB}$, and there is no exact model matching to this bias level. The closer models are M^3 and M^2 , whose initial states are computed from $b_G = 0\text{dB}$

Figure 5.13: Model probabilities when $b_G = -1dB$

and $b_G = -2dB$ respectively. We note that M^3 and M^2 show higher probabilities than the other models. These results demonstrate that model probabilities can automatically adapt to antenna bias. Therefore, better estimation can be generated by the MM-UKF.

We also observe that correct model probabilities are gradually achieved with time, and they might be wrong at the beginning stage. This is because a certain number of measurements are required to reach a satisfactory level of estimation. This is especially true for our problem, which estimates a state with three elements through a scalar measurement. Thus, using a single model with a batch initiation process, such as the ML or the MCMC method, on a fixed number of measurements may take longer time for track initiation. This is because the batch-based track initiation requires a number of measurements, such that an acceptable initial state can be obtained. The proposed MM-UKF hybrid estimation method has no limitation on the batch size. The inaccurate initial states are assigned lower probabilities, whereas the more accurate initial states

are given higher probabilities automatically. The probabilities are updated in every time cycle recursively.

5.6 Conclusion

This chapter formulates the passive localization problem as a nonlinear dynamic estimation problem, such that the bias in the antenna system and the target location can be estimated simultaneously. The observability of the problem is analyzed. Two algorithms, namely, the A-UKF and the MM-UKF, are proposed to solve the nonlinear problem. The A-UKF estimates the augmented state, which consists of the target position and the antenna bias, using the antenna gain difference between two adjacent antennas in the system. It is known the performance of a UKF depends on the accuracy of the initial state, but it is difficult to obtain an initial state with acceptable accuracy. The MM-UKF is therefore suggested to deal with the uncertainty in the initial state. The multiple models are constructed by different initial states, which can be either accurate or inaccurate, and the final state is computed by a weighted sum of the individual model states. Simulation experiments are conducted on different bias levels. The results show that the A-UKF and the MM-UKF are obviously better than the conventional two-stage localization method in terms of target position accuracy. The MM-UKF shows faster convergence rate than the A-UKF, and the RMSE on position is more stable for the MM-UKF than for the other two algorithms. Thus, a nonlinear track initiation problem can be solved by a multiple model approach.

Chapter 6

MM Application on Target Localization and Antenna Auto-Calibration (II): Two-element Array Antenna System

As we addressed in Chapter 5, locating an RF emitter from a distorted antenna system is a challenging problem. The MM-UKF algorithm has been proposed to solve this problem in a distorted spiral antenna system. In this chapter, we continue the research on the problem using a two-element array antenna system.

The chapter is organized as follows. Section 6.1 introduces the two-element array antenna system. Section 6.2 formulates the localization problem as a dynamic estimation problem. The state transition equation and the measure-

ment equation are defined for two cases, namely, a perfect case without pattern distortion, and a distorted case where the distance between the two antennas deviates from the design value. The posterior CRLB of the distorted case is derived for performance benchmarking, and observability analysis for the problem is conducted as well. Section 6.3 describes the UKF and the MM-UKF algorithms for solving the dynamic estimation problem. Section 6.4 shows the experimental results on a simulated antenna system and Section 6.5 tests the algorithms on a real two-element array antenna system. Finally, the conclusion is given in Section 6.6.

6.1 Introduction on Linear Monopole Phased Array System

A linear monopole phased array system consists of multiple monopole antenna elements which are located linearly. The beam pattern of such a system can be controlled by the phase and amplitude of excitation on each monopole antenna element. Thus, the system can be designed to rotate/steer its beam over a surveillance sector through applying a set of predefined excitations. The rotating can be performed electronically.

A generic passive direction finding method of such a system is through beam rotating. When the beam points to an emitter, the strongest amplitude response to the emitter is generated. The emitter DOA is then estimated by the peak of detected amplitude. This DF method requires forming of narrow beam to achieve good DOA accuracy. As the beamwidth is related to the number of monopole elements used in the system, a sharper beam can be generated by

increasing the number of antenna elements. This increases the system complexity.

In this chapter, we suggest an alternative DF method by using amplitude comparison method. Instead of rotating antenna beams in all directions of surveillance sector, the beams are formed at two specific look directions only. Figure 6.1 illustrates the two antenna beam patterns. The look directions of the two beams point to the fixed bearings, namely, -60° and -120° . The surveillance region is in between the two look directions. The DOA of the target is

Figure 6.1: Two fixed beams in a two-element array antenna system

estimated through comparing the amplitudes received from two beam patterns. Different from the rotating method, this DF method prefers wide beams. This is because the surveillance sector must be covered by the overlapping of two beams, so that amplitude comparison can be performed. Thus, fewer number of monopole elements is required. In this research, we suggest an antenna system with two monopole elements. The advantages of the system are its simplicity, and less computational cost in beamforming and steering process. However, the

DOA accuracy highly depends on the accuracy of beam patterns. Beam distortion, which is often observed in real systems, can decrease DOA estimation significantly. For example, the antenna locations may drift in an unmanned environment. Thus auto-calibration is important for a distorted system. In this chapter, we focus on the passive localization problem with auto-calibration capability for such a system, and it is formulated as a dynamic estimation problem in the next section.

6.2 Problem Formulation and Analysis

The problem we address in this chapter is to locate a stationary RF emitter using a two-element array antenna system deployed on a moving platform. We make two assumptions for the problem: 1) The stationary target is located in the detection sector throughout the whole process, for example, in the bearing range from -60° to -120° in Figure 6.1. This assumption is set for simplicity. The detection bearing coverage can be easily extended to 360° by using multiple linear arrays. 2) The altitude difference z_r between the platform and the emitter is known. It is a reasonable assumption for ground and surface emitters, as the platform altitude is known, and the emitter altitude is fixed and known. We make this assumption because the beam pattern is a function of z_r .

The antenna amplitude response to the emitter signal $s(t)$ for the i th beam ($i \in \{1, 2\}$) is

$$a_i(t) = s(t) \sum_{k=0}^1 (\omega_i^{(k)})^* e^{j \frac{2\pi k d}{\lambda} \sin(b-B_i) \sin(e-E_i)}, \quad (6.1)$$

where k represents the k th antenna element in the system, B_i and E_i define the look direction of the i th beam in bearing and elevation coordinates respec-

tively, b and e are the bearing and the elevation of the emitter respectively, d is the spacing between two elements, λ is the wavelength, $(\omega_i^{(k)})^*$ is the complex conjugate of the weightage for the k th array element. Note that bearing is defined as a clockwise angle from the true north, and elevation is defined as an anti-clockwise angle from the horizon. The signal power is then derived as

$$P_i = 2S \left(1 + \cos \left(\frac{2\pi d}{\lambda} \sin \Delta b_i \sin \Delta e_i \right) \right), \quad (6.2)$$

where S is the power of the emitter signal, and

$$\Delta b_i = b - B_i, \quad (6.3)$$

$$\Delta e_i = e - E_i. \quad (6.4)$$

The i th antenna beam pattern is represented by P_i . Converting (6.2) to dB, we obtain

$$\begin{aligned} g_i &= 10 \log_{10}(P_i) \\ &= C + 10 \log_{10} \left(1 + \cos (\pi d_r \sin \Delta b_i \sin \Delta e_i) \right), \end{aligned} \quad (6.5)$$

where

$$C = 10 \log_{10}(2S), \quad (6.6)$$

$$d_r = \frac{2d}{\lambda}. \quad (6.7)$$

We define d_r as the pattern distortion parameter, which is contributed by the uncertainty of the spacing d between the two elements and the emitter

wavelength λ .

We investigate the antenna system for two cases, namely, the perfect antenna system and the distorted antenna system. In the perfect system, the actual antenna patterns perfectly match to the predefined patterns. The parameter d_r is completely known from a priori knowledge. In the distorted system, d_r is uncertain. This uncertainty results in the patterns being distorted from the predefined shapes. It is obvious that the distorted system is more realistic, as the distance between the antennas and the target frequency are uncertain in reality. The following paragraphs will address the dynamic models for the perfect case and the distorted case.

6.2.1 Perfect Antenna Pattern Case

The problem is formulated as a discrete dynamic estimation problem. A dynamic estimation problem consists of two fundamental models, namely, a transition model and a measurement model. The transition model describes the state transition pattern from the previous time cycle to the current time cycle, and the measurement model converts the state vector to the measurement vector.

The state in the perfect antenna system is defined as

$$\mathbf{x}_k = [x_k \ y_k]^T, \quad (6.8)$$

where x_k and y_k are the stationary target positions in x and y coordinates at time k respectively. The transition equation can be written as a linear form

$$\mathbf{x}_k = \mathbf{F}\mathbf{x}_{k-1} + \mathbf{w}_{k-1}, \quad (6.9)$$

where $\mathbf{F}$ is the 2×2 identity matrix to perform a stationary transition. $\mathbf{w}_k$ is the process noise, and it is set to $\mathbf{0}$ for a stationary target. The platform state vector $\mathbf{x}_{p,k}$ is defined as

$$\mathbf{x}_{p,k} = [x_{p,k} \ y_{p,k}]^T, \quad (6.10)$$

where $x_{p,k}$ and $y_{p,k}$ are the platform positions in x and y coordinates at time k respectively. The relative state vector $\mathbf{x}_{r,k}$ is defined as

$$\begin{aligned} \mathbf{x}_{r,k} &= \mathbf{x}_k - \mathbf{x}_{p,k} \\ &= [x_{r,k} \ y_{r,k}]^T. \end{aligned} \quad (6.11)$$

The measurement of the dynamic system, z_k , is a scalar. It is defined as the difference between the power responses of two beams. The measurement equation can be derived as

$$\begin{aligned} z_k &= h(\mathbf{x}_{r,k}) + r_k \\ &= g_{1,k} - g_{2,k} + r_k \\ &= 10 \log_{10} (1 + \cos(\pi d_r \sin \Delta b_{1,k} \sin \Delta e_{1,k})) \\ &\quad - 10 \log_{10} (1 + \cos(\pi d_r \sin \Delta b_{2,k} \sin \Delta e_{2,k})) \\ &\quad + r_k, \end{aligned} \quad (6.12)$$

where $h(\mathbf{x}_{r,k})$ converts the relative state to the measurement; r_k is the white

Gaussian measurement noise with variance R_k , and

$$\begin{aligned}\Delta b_{i,k} &= \tan^{-1} \left(\frac{x_{r,k}}{y_{r,k}} \right) - B_i, \\ \Delta e_{i,k} &= \tan^{-1} \left(\frac{z_{r,k}}{\sqrt{x_{r,k}^2 + y_{r,k}^2}} \right) - E_i,\end{aligned}\tag{6.13}$$

with $i \in \{1, 2\}$. It is seen that the emitter power S has been totally removed from the measurement equation.

6.2.2 Distorted Antenna Pattern Case

To track the distortion caused by the spacing between the two antenna elements and the emitter frequency, we augment the emitter positions with d_r in the state vector as

$$\mathbf{x}_k = [x_k \quad y_k \quad d_{r,k}]^T.\tag{6.14}$$

The relative state vector is

$$\mathbf{x}_{r,k} = [x_{r,k} \quad y_{r,k} \quad d_{r,k}]^T.\tag{6.15}$$

The transition equation and the measurement equation are

$$\mathbf{x}_k = \mathbf{F}\mathbf{x}_{k-1} + \mathbf{w}_{k-1},\tag{6.16}$$

$$z_k = h(\mathbf{x}_{r,k}) + r_k,\tag{6.17}$$

where $\mathbf{F}$ is the 3×3 identity matrix, $\mathbf{w}_{k-1}$ is set to $\mathbf{0}$, and the measurement

function is the same as that for the perfect system defined in (6.12).

6.2.3 Posterior Cramer-Rao Lower Bound

This section derives the CRLB of the distorted case for performance banchmarking. The CRLB sets a lower bound on the second order error (error covariance) of any unbiased estimator. The PCRLB developed by Tichavsky is a recursive form of the CRLB [67]. It avoids manipulating large matrices and is suitable for discrete dynamic systems. For a generic nonlinear discrete system

$$\mathbf{x}_k = f(\mathbf{x}_{k-1}) + \mathbf{w}_{k-1}, \quad (6.18)$$

$$\mathbf{z}_k = h(\mathbf{x}_k) + \mathbf{r}_k, \quad (6.19)$$

the PCRLB recursion is

$$\mathbf{J}_k = [\mathbf{F}_{k-1} \mathbf{J}_{k-1}^{-1} \mathbf{F}_{k-1}^T + \mathbf{Q}_{k-1}]^{-1} + \mathbf{H}_k^T \mathbf{R}_k^{-1} \mathbf{H}_k, \quad (6.20)$$

where $\mathbf{J}_k$ is the information matrix for the estimated state vector $\mathbf{x}_k$, $\mathbf{F}_{k-1}$ is the Jacobian matrix of $f(\mathbf{x}_{k-1})$, and $\mathbf{H}_k$ is the Jacobian matrix of $h(\mathbf{x}_k)$. The process noise covariance $\mathbf{Q}_{k-1}$ can be considered as $\mathbf{0}$. The $\mathbf{F}_{k-1}$ in our problem is an identity matrix. The recursive information matrix $\mathbf{J}_k$ is then simplified as

$$\mathbf{J}_k = \mathbf{J}_{k-1} + \mathbf{H}_k^T \mathbf{R}_k^{-1} \mathbf{H}_k. \quad (6.21)$$

It can be easily derived that the initial value of the recursive information matrix $\mathbf{J}_1$ is the inverse of the initial state error covariance matrix $\mathbf{P}_1$. The

PCRLB for the position pos and d_r can be written as

$$PCRLB_{pos,k} = \sqrt{\mathbf{J}_{11,k}^{-1} + \mathbf{J}_{22,k}^{-1}}, \quad (6.22)$$

$$PCRLB_{d_r,k} = \sqrt{\mathbf{J}_{33,k}^{-1}}. \quad (6.23)$$

Now we derive the Jacobian matrix $\mathbf{H}_k$ from (6.12). To simplify the expressions, we omit the time index k in the following derivation.

$$\mathbf{H} = \begin{bmatrix} \frac{\partial h}{\partial x_r} & \frac{\partial h}{\partial y_r} & \frac{\partial h}{\partial d_r} \end{bmatrix}, \quad (6.24)$$

where

$$\frac{\partial h}{\partial x_r} = \frac{\partial g_1}{\partial x_r} - \frac{\partial g_2}{\partial x_r}, \quad (6.25)$$

$$\frac{\partial h}{\partial y_r} = \frac{\partial g_1}{\partial y_r} - \frac{\partial g_2}{\partial y_r}, \quad (6.26)$$

$$\frac{\partial h}{\partial d_r} = \frac{\partial g_1}{\partial d_r} - \frac{\partial g_2}{\partial d_r}, \quad (6.27)$$

$$\frac{\partial g_i}{\partial x_r} = -\frac{10 \sin \theta_i}{\ln 10(1 + \cos \theta_i)} \frac{\partial \theta_i}{\partial x_r}, \quad (6.28)$$

$$\frac{\partial g_i}{\partial y_r} = -\frac{10 \sin \theta_i}{\ln 10(1 + \cos \theta_i)} \frac{\partial \theta_i}{\partial y_r}, \quad (6.29)$$

$$\frac{\partial g_i}{\partial d_r} = -\frac{10\pi \sin \theta_i \sin \Delta b_i \cos \Delta e_i}{\ln 10(1 + \cos \theta_i)}, \quad (6.30)$$

$$\theta_i = \pi d_r \sin \Delta b_i \sin \Delta e_i, \quad (6.31)$$

$$\begin{aligned} \frac{\partial \theta_i}{\partial x_r} &= \pi d_r \cos \Delta b_i \sin \Delta e_i \frac{\partial \Delta b_i}{\partial x_r} \\ &\quad + \pi d_r \sin \Delta b_i \cos \Delta e_i \frac{\partial \Delta e_i}{\partial x_r}, \end{aligned} \quad (6.32)$$

$$\begin{aligned} \frac{\partial \theta_i}{\partial y_r} &= \pi d_r \cos \Delta b_i \sin \Delta e_i \frac{\partial \Delta b_i}{\partial y_r} \\ &\quad + \pi d_r \sin \Delta b_i \cos \Delta e_i \frac{\partial \Delta e_i}{\partial y_r}, \end{aligned} \quad (6.33)$$

and

$$\frac{\partial \Delta b_i}{\partial x_r} = \frac{y_r}{x_r^2 + y_r^2}, \quad (6.34)$$

$$\frac{\partial \Delta e_i}{\partial x_r} = \frac{x_r z_r}{(x_r^2 + y_r^2 + z_r^2) \sqrt{x_r^2 + y_r^2}}, \quad (6.35)$$

$$\frac{\partial \Delta b_i}{\partial y_r} = \frac{-x_r}{x_r^2 + y_r^2}, \quad (6.36)$$

$$\frac{\partial \Delta e_i}{\partial y_r} = \frac{y_r z_r}{(x_r^2 + y_r^2 + z_r^2) \sqrt{x_r^2 + y_r^2}}, \quad (6.37)$$

where $i \in \{1, 2\}$. With $\mathbf{H}_k$, the PCRLBs of the emitter position pos and the pattern distortion parameter d_r can be computed recursively by (6.22) and (6.23) respectively.

6.2.4 Observability Analysis

This section analyzes the problem observability of the distorted case. As we discussed in Chapter 5, a nonlinear system is locally observable, if its observability matrix $\mathbf{O}$ is linearly independent, namely, $rank(\mathbf{O}) = n$ or $det(\mathbf{O}) \neq 0$, where n is the state size. The observability matrix $\mathbf{O}$ is defined as

$$\mathbf{O} = [\mathbf{H}_k \quad \mathbf{H}_{k+1}\mathbf{F}_k \quad \mathbf{H}_{k+2}\mathbf{F}_{k+1}\mathbf{F}_k]^T, \quad (6.38)$$

in the distorted pattern estimation problem. Since $\mathbf{F}_k$ is an identity matrix, the observability matrix becomes

$$\mathbf{O} = [\mathbf{H}_k \quad \mathbf{H}_{k+1} \quad \mathbf{H}_{k+2}]^T. \quad (6.39)$$

It is only contributed by the Jacobian matrix $\mathbf{H}$. However, it is difficult to analyze $\det(\mathbf{O})$ theoretically, as $\mathbf{H}$ derived in (6.24) is too complicated. Alternatively, we may use a numerical method to illustrate the problem observability. The problem is stated as: Given a two-element array antenna system, and its moving path of three consecutive times, namely, from time index k to $k+2$, compute $\det(\mathbf{O})$ for all the possible state $\mathbf{x}$ in a predefined surveillance region.

The surveillance region in the observability study is limited to a sector with a bearing range from 60° to 120° and a distance range from $0m$ to $100m$. The two look directions of the beam patterns are 60° and 120° from true north. We compute $\det(\mathbf{O})$ for the following three antenna paths:

- Path A shown in Figure 6.2: Three consecutive locations of the antenna system are $(-30m, -10m)_k$, $(-20m, -5m)_{k+1}$ and $(-10m, 0m)_{k+2}$;
- Path B shown in Figure 6.3: Three consecutive locations of the antenna system are $(-30m, 0m)_k$, $(-20m, 0m)_{k+1}$ and $(-10m, 0m)_{k+2}$;
- Path C shown in Figure 6.4: Three consecutive locations of the antenna system are $(-10m, -10m)_k$, $(-10m, 0m)_{k+1}$ and $(-10m, 10m)_{k+2}$.

The results on different d_r (0.8, 0.9, 1.0) of the three paths are displayed in Figure 6.2, Figure 6.3 and Figure 6.4. Since $\det(\mathbf{O})$ in this problem covers a large range of values, we use logarithmic scale $\log_{10}|\det(\mathbf{O})|$ to present the

Figure 6.2: $\log_{10}|det(\mathbf{O})|$ in the surveillance sector when the antenna system moves in path A

Figure 6.3: $\log_{10}|det(\mathbf{O})|$ in the surveillance sector when the antenna system moves in path B

Figure 6.4: $\log_{10}|det(\mathbf{O})|$ in the surveillance sector when the antenna system moves in path C

results. In the figures, lighter color represents higher observability. The black color indicates the problem is not observable when the emitter is located at those points.

It is observed that some black lines are appeared in Figure 6.2 and Figure 6.3. These black lines are all along the antenna moving paths. The black lines

represent that $\det(\mathbf{O})$ is 0. It means the problem is not observable if the emitter is located on these lines. In Figure 6.4, the extension of path C is out of the surveillance region, and no black line is observed. Thus, we can conclude that the problem is observable if the emitter location is not along the antenna moving path. The moving path is important to observability. Path C, which is neither moving to nor away from the surveillance region, shows the highest observability among the three paths. This information is useful for path planning.

We also observed that the observability is related to the antenna distorted parameter d_r and the distance between the emitter and the antenna system. Observability increases with d_r , as the figures with $d_r = 1.0$ are lighter than those figures with smaller d_r . The observability decreases with distance, as it is quite obvious from all the observability figures that further regions have darker color than nearer regions.

6.3 Dynamic Estimation Algorithms

The UKF and the MM-UKF are proposed to estimate the states for the problems formulated in Section 6.2. The UKF performs the state estimation for both the perfect antenna and the distorted antenna problems. The MM-UKF provides an enhanced state estimation on the distorted antenna problem.

6.3.1 UKF and Constrained UKF

The UKF algorithm is applied to estimate the states of the perfect antenna system and the distorted antenna system formulated in Section 6.2. The sigma points used to represent the state pdf need to be assigned first in the UKF. The

number of sigma points is $2n_{\mathbf{x}} + 1$, where $n_{\mathbf{x}}$ is the number of elements in the state vector. For the perfect antenna case, the size of the state is 2. The number of sigma points is 5. For the distorted antenna case, the state size increases to 3. The number of sigma points is 7.

Figure 6.5: UKF with Constraints

The generic UKF described in Section 2.2.4. can be applied to the perfect antenna case. For the distorted case, we apply a constrained UKF to perform estimation. This is because an antenna system will not work properly if the distance between antenna elements exceeds half of the signal wavelength. To keep a valid antenna system, we make an assumption that the maximum value of the distorted parameter d_r is 1. Thus, a constraint, $d_r \leq 1$, is applied to state estimation of the distorted antenna case. A constrained UKF [105] can be illustrated in Figure 6.5. It is similar to the UKF except an additional projection step. In this additional step, the unconstrained sigma points are validated. If they are outside the feasible region, they will be projected onto the boundary of the feasible region.

Initial values of the state need to be specified in the UKF algorithms. A common challenge of a nonlinear estimator is that a wrong initial value may

cause subsequent estimation divergence or convergence at a local optimum. To obtain the global optimum, a more robust algorithm should be developed. The MM-UKF is therefore proposed to solve the initiation problem in Section 6.3.2.

6.3.2 Multiple Model UKF

The multiple model estimation approach is widely used in dynamic state estimation when model or parameter uncertainty exists. Initial uncertainty of the distorted antenna parameter d_r can be formulated as an MM hybrid system. A generic MM hybrid system can be described as

$$\mathbf{x}_k^j = f^j(\mathbf{x}_{k-1}^j) + \mathbf{w}_k, \quad (6.40)$$

$$z_k = h^j(\mathbf{x}_k^j) + r_k, \quad (6.41)$$

where j is the system model index.

In the distorted antenna system, we assume the range of d_r is known from a priori knowledge. A number of initial states are assigned with different initial d_r . d_r values are uniformly distributed on its feasible range. The multiple models are constructed by this set of initial states with the same transition and measurement functions defined in (6.16) and (6.17) respectively. The static MM algorithm described in Section 2.3.1 is applied to the problem with three steps, namely, filtering, model likelihood evaluation, and combination. In the filtering step, multiple constrained UKF filters run in parallel to estimate the states with different initial values. In the second step, the likelihood of each model is computed based on measurement residual. Finally, the hybrid state, which is the final output of the time cycle, is combined from all the individual

model states.

The MM approach has the advantages of preventing estimation from divergence and performing fast convergence, as the algorithm can adapt to the correct models through the likelihoods, which are computed from measurement residuals.

6.4 Experimental Results on a Simulated Antenna System

This section demonstrates the performance of the proposed algorithms through experiments on a simulated antenna system. The constrained UKF and the MM-UKF are applied on the distorted antenna system. The UKF is applied to the assumed perfect antenna system for comparison purpose.

The test scenario is shown in Figure 6.6. A stationary RF emitter is located at (0m, 0m, 0m), and a platform moves from south to north with speed 200m/s from the starting point (4000m, -2000m, 500m). The altitude of the platform does not change. The total duration is 20s.

A two-element array antenna system is deployed on the moving platform. Two fixed beams with look directions -60° and -120° are formed from the antenna array. The measurement noise is assumed as white Gaussian with standard deviation 0.5dB. The receiver processing time interval T is 1sec.

Three algorithms, namely, UKF-P, UKF-D and MM-UKF, are applied on the scenario. The UKF-P stands for the UKF applied on the assumed perfect antenna system. d_r is set to a fixed value of 1 in the UKF-P. The initial state

Figure 6.6: Test scenario of the simulated antenna system

$\mathbf{x}(1)$ is set to $[1000 \ 1000]^T$, and the initial state error covariance $\mathbf{P}(1)$ is set as the 2×2 diagonal matrix with diagonal elements $[1000^2 \ 1000^2]$. The UKF-D is the constrained UKF applied on the distorted antenna system with the state vector augmented with d_r . The initial state $\mathbf{x}(1)$ is set to $[1000 \ 1000 \ 1]^T$, and the initial state error covariance $\mathbf{P}(1)$ is set as the 3×3 diagonal matrix with diagonal elements $[1000^2 \ 1000^2 \ 0.05^2]$. Three models, namely, M^1 , M^2 and M^3 , are used in the MM-UKF. The initial states of the three models are $[1000 \ 1000 \ 1]^T$, $[1000 \ 1000 \ 0.9]^T$ and $[1000 \ 1000 \ 0.8]^T$ respectively. The initial state error covariances for the three models are all set as a 3×3 diagonal matrix with diagonal elements $[1000^2 \ 1000^2 \ 0.05^2]$. The measurement error variance R is 1.

The tests are carried out on different distortion levels, which are specified by the parameter d_r . d_r values on each test are 0.80, 0.85, 0.90, 0.95 and 1.00 respectively. 100 Monte Carlo runs are carried out on each test. Figure 6.7 shows the position accuracy of the three algorithms. The time averages of the

position RMSE vs. d_r are displayed.

Figure 6.7: Time averages of the target position RMSE vs. system distortion parameter d_r

It can be seen that the MM-UKF generates the smallest position RMSE among the three algorithms. The RMSEs generated by the UKF-P and the UKF-D increase tremendously when d_r is less than 0.95, whereas the RMSE of the MM-UKF is much more stable. The performance of the single-model estimators, namely UKF-P and UKF-D, is decreasing with the bias. The UKF-P assumes that d_r is always 1. It does not take the distortion into consideration. Thus, the biggest RMSE is created among the three algorithms when distortion exists. As for the UKF-D algorithm, although it has an adaptive d_r , the adaptation is limited by the accuracy of the initial state. The results show that the UKF-D only works well when its initial d_r is close to the actual d_r . Once the initial error exceeds a certain level, its RMSE increases significantly. The MM-UKF overcomes the nonlinear initiation problem through a hybrid system with multiple initiation states. The hybrid system can adapt to the correct model

automatically. A better and stable estimation is therefore achieved through the robustness on the initial setting.

Figure 6.8: The RMSE of the estimated target position vs. time when the actual distortion level is 0.8

Figure 6.9: The RMSE of the estimated distortion parameter d_r vs. time when the actual distortion level is 0.8

The RMSE vs. time is also investigated in the simulation tests. Figure 6.8

Figure 6.10: The model probabilities vs. time in the MM-UKF when the actual distortion level is 0.8

shows the RMSE of the estimated position vs. time, and Figure 6.9 shows the RMSE of the estimated distortion parameter d_r vs. time. The results are obtained from 100 Monte Carlo runs when the actual distortion parameter d_r is set to 0.8. Since the UKF-P does not estimate d_r , only the results of the UKF-D and the MM-UKF are presented in Figure 6.9. Note that the initial RMSEs of d_r for these two algorithms are different (see Figure 6.9). The initial error for the UKF-D is 0.2, as it is the difference between the initial value 1 and the actual value 0.8. The initial error for the MM-UKF is 0.1. It is the difference between the average of the initial values in the three models and the actual value. The initial value of the CRLB is set as the same as that in UKF-D. From the results we can see that the MM-UKF obviously outperforms the UKF-D, and RMSEs generated by the MM-UKF converge to the CRLBs.

The model probability behavior in the MM-UKF algorithm is also studied in the simulation test. The model probabilities of M^1 , M^2 and M^3 vs. time

are shown in Figure 6.10. The actual distortion parameter is set to 0.8 in the test. As expected, the model M^3 , whose d_r is initiated as 0.8, shows the highest probability, whereas, the probabilities of the other models decrease to zero. The results demonstrate that model probabilities can automatically adapt to the correct model to achieve a robust result.

6.5 Experimental Results on a Real Antenna System

An indoor test system is modified from the smart antenna system developed for WiFi localization [106]. All the hardware components in the test system are displayed in Figure 6.11. They consist of a RF emitter and a two-element array antenna connected to a notebook PC through a beamformer. The PC can control the beamformer by indicating the look direction of the scanning beam. In each time step, two beams with predefined look directions are formed, and the difference between the gains on the two beams is recorded by the data collection system. The gain differences are the measurements in the nonlinear system.

The test scenario is shown in Figure 6.12. The RF emitter is located on a table at (0cm, 0cm, 165cm). The array is mounted on the ceiling with height 215cm, and the array elements point to the ground. Two beams generated by the beamformer have fixed look directions in bearing -120° and 180° . The antenna array moves from (0cm, 294cm, 215cm) to (470cm, 294cm, 215cm) in 16 time steps. Including the starting point, 17 measurements are collected in total.

Figure 6.11: Hardware components of the indoor antenna system

The three algorithms, UKF-P, UKF-D and MM-UKF, are applied on the collected data. The initial states of the UKF-P and the UKF-D are set to $[200 \ 200]^T$ and $[200 \ 200 \ 0.8]^T$ respectively, and the initial states for the three models of the MM-UKF are set to $[200 \ 200 \ 0.8]^T$, $[200 \ 200 \ 0.9]^T$ and $[200 \ 200 \ 1]^T$ respectively. For all the algorithms, the position error variances on both x and y coordinates are set to 200^2 , and the error variance on d_r is set to 0.05^2 .

The RMSE on the estimated target position vs. time is shown in Figure 6.13. The results are similar to that of the simulated antenna system shown in Fig-

Figure 6.12: The test scenario of the real antenna system

ure 6.8. The RMSE on the estimated target position in the MM-UKF converges to a small value, whereas the UKD-P and UKF-D produce large RMSE error. This result further confirms that the MM-UKF is a robust algorithm when initial condition is uncertain.

Figure 6.13: The RMSE on the estimated target position vs. time in the real antenna system

6.6 Conclusion

This chapter formulates the passive localization problem using a two-element array antenna system as a nonlinear dynamic estimation problem, which augments the target location and the antenna pattern distortion in the state. This formulation has antenna auto-calibration capability, in which the target position and the pattern distortion can be estimated simultaneously. The problem observability is analyzed numerically. Two algorithms, namely, the constrained UKF and the MM-UKF, are proposed to solve the nonlinear problem. The constrained UKF estimates the augmented state using the gain difference between two fixed beams. The results show the improvement achieved by this UKF is limited due to the inaccurate initial state. To obtain a robust initiation, the MM-UKF is therefore suggested to deal with initiation uncertainty. The multiple models are constructed by different initial states, and the final state is automatically computed by a weighted sum of all the individual model states. The experiments are conducted on a simulated and a real antenna systems. The results show that the MM-UKF is obviously better than the UKF. The RMSE generated by the MM-UKF is more stable and converges to the CRLB. The model probabilities in the MM-UKF are also investigated. The results show that the highest probability is assigned to the correct model automatically.

In this chapter, we only address the pattern distortion caused by the uncertainty of the spacing between antenna elements and the emitter frequency. The proposed approach can be extended to estimate other types of distortions, such as, gain/phase and mutual coupling errors, by including them into the state.

Chapter 7

MM Application on Indoor Contaminant Source Estimation

This chapter proposes a real-time approach to estimate the contaminant source information, such as source location, source emission rate and release time, in an indoor environment. The problem is formulated as a hybrid dynamic estimation problem, and is solved by a multiple model approach. The models in the hybrid system represent the multiple zones in the building, and an Unscented Kalman Filter (UKF) is deployed in each model to estimate the source emission rate and the release time, conditioned on the given zone of the model. This Multiple Model UKF (MM-UKF) algorithm is tested in a simulated 3-story building. The results show that the MM-UKF can perform real-time estimation well. A comparison is also made between the proposed MM-UKF and the Markov Chain Monte Carlo (MCMC) estimation method suggested before [107] [108] [109]. The MM-UKF obviously outperforms the MCMC method in terms of estimation accuracy and efficiency.

The chapter is organized as follows. Section 7.1 introduces the contaminant source estimation problem. Section 7.2 describes the related work, which includes the forward transport model and the MCMC estimation approach. The multiple zone airflow model is utilized in the forward transport model in this research. The MCMC is a well-known approach for contaminant source estimation, and it is used for comparison with our proposed MM-UKF. Section 7.3 formulates the indoor contaminant source estimation problem as dynamic estimation problems, and proposes algorithms to solve the problems. Firstly, a single model system is formulated, and the MCMC algorithm is applied to estimate the state consisting of the source location, emission rate and release time. Secondly, a multiple model formulation is described, and the MM-UKF is suggested to perform hybrid estimation. Section 7.4 demonstrates the performance of the MM-UKF through tests on a simulated 3-story building. A comparison study involving the MM-UKF and the MCMC method is also conducted. Section 7.5 gives a conclusion on this work.

7.1 Introduction

The contaminant source estimation problem is getting increasing importance due to more and more occurrences of sick building syndrome and attacks from covert chemical warfare agents. To monitor a building contaminant condition, a number of sensors are connected through a network, and the sensor measurements are sent to a fusion center to estimate contaminant source information. A real-time estimation algorithm is required such that timely actions can be taken to mitigate the adverse effects. Figure 7.1 illustrates the structure of a contaminant source estimation setup. The building to be monitored is divided

into zones. A zone can be a room, a corridor, etc. Sensors deployed in the pre-defined zones measure the contaminant intensity every detection time cycle. A real-time algorithm at fusion center estimates the contaminant source information, such as the source location, the emission rate and the release time, based on the sensor measurements received.

Figure 7.1: Contaminant source estimation in a building

There are three common contaminant emission types, namely, instantaneous emission, continuous emission and exponential emission. The instantaneous emission releases chemical one time only. Its emission rate is a scalar. The continuous emission disperses chemical with a constant rate. Its emission rate can be represented by a scalar as well. The emission rate of the exponential emission is a function of time, and it decreases exponentially. In this research, we focus on the instantaneous emission type.

7.2 Transport Models and Estimation Algorithms in Contaminant Source Estimation Problem

Many studies in contaminant source estimation have been presented in the literature [110] [111]. The existing methods can be categorized into two classes, namely, the forward approach [112] [113] [114] [115] [116] [117] and the backward approach [118] [119] [120] [121]. The forward approach assumes that the forward model, $\mathbf{z} = h(\mathbf{x})$, is known, where $\mathbf{z}$ is the measurement vector which consists of all the contaminant concentrations detected by the sensors, $\mathbf{x}$ is the state vector which contains the contaminant source information such as source location, emission rate and so on. The forward approach predicts the sensor measurements from different contaminate sources, and decides on the most possible source by matching the predicted measurements and the real measurements. The backward approach assumes that the backward model, $\mathbf{x} = h^{-1}(\mathbf{z})$, is known. The source information can then be obtained from the sensor measurements directly. Since the backward model is not always available, we focus on the forward approach in this research.

Two main components of the forward approach are the forward transport model and the estimation algorithm. Brief reviews on popular transport models and estimation algorithms are given in Sections 7.2.1 and 7.2.2 respectively.

7.2.1 Transport Models

Two commonly used transport methods for indoor contaminant are the multi-zone airflow model [122] [123] and the Computational Fluid Dynamics (CFD) model [124] [125]. The multi-zone airflow model generally represents rooms of a building as zones with homogeneous properties and can provide a quick prediction of airflow and contaminant migration. The CFD model can provide more spatial and temporal details on airflow and contaminant migration. However, it would not be practical to use a CFD model in real-world scenario simulation, due to the high computational complexity and cost required. The multi-zone airflow model provides a better balance between accuracy and speed.

In the multi-zone airflow model, the building can be first idealized as a complicated interlacing grid system of air-mass flow paths, where nodes represent the zones of the building, and the connections between nodes simulate airflow paths. These airflow paths include flow resistances caused by open or closed doors and windows as well as air leakage through cracks in walls and non-airtight openings. Henceforth, the resulting nodal network resembles an electrical circuit with the airflow paths acting as resistances to flow generated by applied pressure differences. Typically, the airflow rate $F_{j,i}$ from zone j to zone i , is a nonlinear function of the pressure drop along the flow path. Pressure differences can be generated either naturally by the action of wind on the building surface and buoyancy due to differences in density, or mechanically by the action of fans.

The nodal formulation of the network leads to a set of algebraic mass balance equations that must be satisfied simultaneously, while the nonlinearity of the airflow paths necessitates a nonlinear equation solver such as the Newton-

Raphson method

$$\xi_k = \xi_{k-1} - \mathbf{J}_{k-1}^{-1} \mathbf{v}_{k-1}, \quad (7.1)$$

where k is the iteration index, $\mathbf{J}$ represents the Jacobian matrix of residual derivatives, ξ represents the pressure vector and $\mathbf{v}$ represents the flow residual vector.

The multi-zone transport model subsequently captures the transport of contaminants by advection via calculated inter-zone airflows and well-defined mechanical system flows via the mass conservation equation [126]

$$\frac{dm_i}{dt} = -R_i C_i - \sum_j F_{i,j} C_i + \sum_j F_{i,j} (1 - \eta_{j,i}) C_j + G_i, \quad (7.2)$$

where t represents time, i and j are the zone indices, m_i is the contaminant mass in zone i (kg), R_i is the removal rate in zone i (kg air/s), C_i is the contaminant concentration in zone i (kg/kg air), $F_{i,j}$ is the airflow rate from zone i to zone j (kg air/s), $\eta_{j,i}$ is the filtration efficiency between zone i and zone j , and G_i is the contaminant generation rate in zone i (kg/s).

In this research, we consider instantaneous emission as the emission type. The contaminant generation rate G_i of the instantaneous emission is defined as

$$G_i = \begin{cases} \rho, & \text{for } t = \tau, \\ 0, & \text{for } t > \tau, \end{cases} \quad (7.3)$$

where τ is the release time, and ρ is the emission rate.

Although the multi-zone airflow model provides much less computational cost than the CFD model, it may still be too complex for practical implementation. Here, we approximate the multi-zone airflow transport model by a lookup

table [107] [108]. The lookup table is built up based on the multi-zone airflow transport model off-line. The structure of the lookup table is shown in Figure 7.2. It stores concentration value versus time for all possible dispersions from a source zone to a sensor zone. Thus, the size of the table would be $M \times N \times K$, where M is the number of sensors deployed among the zones (each zone has at most one sensor), N is the number of zones in a building, and K is the number of time units in the whole dispersion process.

Figure 7.2: Lookup table structure. M is the number of sensors zones, and N is the number of zones in a building. Each zone has at most one sensor, i.e. $M \leq N$.

The emission rates in the lookup table are set to a standard unit value. If we assume that the resulting concentrations at sensor zones are approximately proportional to the source strength, all possible measurement values can even-

tually be obtained from the lookup table. This assumption is true for the case of instantaneous emission.

7.2.2 Estimation Algorithms

The forward estimation algorithms can be categorized into two classes, namely, the optimization approach and the Bayesian approach. The optimization approach [112] [113] [114] [115] [116] [117] [127] is commonly used in groundwater pollution and gas release source identification and estimation. This approach obtains the source information through optimizing an objective function. The objective function is normally the least squares function on the measurement residual

$$\hat{\mathbf{x}} = \arg \min_{\mathbf{x}} \left\{ [h(\mathbf{x}) - \mathbf{z}]^T [h(\mathbf{x}) - \mathbf{z}] \right\}, \quad (7.4)$$

where $\mathbf{z}$ is the measurement vector which consists of all the contaminant concentrations detected by the sensors; $\mathbf{x}$ is the state vector which contains the contaminant source information, such as source location, emission rate and so on, to be estimated; $h(\cdot)$ is the transport model which can be represented by the lookup table. Linear programming, Newton-Raphson algorithm, regularization method have been suggested to solve this optimization problem. These algorithms often incur high computational cost due to iterative computation processes. Thus the optimization approach is only applicable to the problems with simple $h(\mathbf{x})$. Furthermore, a priori knowledge on pollution sources is required in this approach. This is because a proper initial condition from a priori knowledge is essential for the algorithms to converge to the global optimum rather than local optima.

On the other hand, the Bayesian approach is more general compared to the optimization approach. Bayesian inference is often combined with an MCMC sampling procedure to solve the contaminant source estimation problem [107] [108] [109]. The MCMC algorithm [39] [128] allows parameters in complex models to be estimated. It can be applied on both static and dynamic estimation system. Since we formulate the problem as a dynamic estimation problem (refer to Section 7.3), we describe the dynamic MCMC algorithm here.

Firstly, m Markov chains are initiated. Each Markov chain consists of a state. The value of state is randomly assigned such that all the Markov chains are uniformly distributed in the possible state space. An iteration process is then applied on the Markov chains in each time cycle. Given m Markov chains with state $\mathbf{x}_{k-1}^j$, where $j = 1, \dots, m$ is the Markov chain index, and k is the time index, the MCMC iteration process for time k is described in Table 7.1.

In Table 7.1, $U(0, 1)$ is the uniform distribution over the interval $(0, 1)$; t_N is the maximum length of a Markov chain, which is predefined to prevent the number of iterations from increasing to an absurd number; the acceptance probability of the Metropolis-Hastings algorithm $\alpha(\mathbf{x0}^j, \mathbf{x1}^j)$ is [128]

$$\alpha(\mathbf{x0}^j, \mathbf{x1}^j) = \min \left(1, \frac{\pi(\mathbf{x1}^j)q(\mathbf{x0}^j|\mathbf{x1}^j)}{\pi(\mathbf{x0}^j)q(\mathbf{x1}^j|\mathbf{x0}^j)} \right), \quad (7.5)$$

where $\pi(\cdot)$ is the distribution of the Markov chain, $q(\cdot)$ is the proposal distribution. If we assume the proposal is symmetric, namely $q(\mathbf{x0}^j|\mathbf{x1}^j) = q(\mathbf{x1}^j|\mathbf{x0}^j)$, then $\alpha(\mathbf{x0}^j, \mathbf{x1}^j)$ becomes

$$\alpha(\mathbf{x0}^j, \mathbf{x1}^j) = \min \left(1, \frac{\pi(\mathbf{x1}^j)}{\pi(\mathbf{x0}^j)} \right). \quad (7.6)$$

<p>Table 7.1: MCMC Bayesian algorithm at time k</p> <hr/> <p>for $j = 1$ to m</p> <ul style="list-style-type: none"> • $\mathbf{x0}_0^j = f_{k-1}(\mathbf{x}_{k-1}^j);$ <p>end for</p> <p>$t = 0;$</p> <p>$G = 9999;$</p> <p>while $[(G - 1 \geq \varepsilon) \cap (t \leq t_N)]$</p> <ul style="list-style-type: none"> • $t = t + 1;$ • validate random move by the Metropolis-Hastings algorithm <p>for $j = 1$ to m</p> <ul style="list-style-type: none"> – draw a sample $\mathbf{x1}_t^j \sim q(\cdot \mathbf{x0}_{t-1}^j);$ – draw a sample $u \sim U(0, 1);$ – compute acceptance probability $\alpha(\mathbf{x0}_{t-1}^j, \mathbf{x1}_t^j);$ – if $\alpha \geq u$, then $\mathbf{x0}_t^j = \mathbf{x1}_t^j,$ else $\mathbf{x0}_t^j = \mathbf{x0}_{t-1}^j;$ <p>end for</p> <ul style="list-style-type: none"> • compute convergence parameter $G;$ <p>end while</p> <p>for $j = 1$ to m</p> <ul style="list-style-type: none"> • $\mathbf{x}_k^j = \mathbf{x0}_t^j;$ <p>end</p> <p>$\hat{\mathbf{x}}_k = mean(\mathbf{x}_k^1, \dots, \mathbf{x}_k^m)$</p> <hr/>

Equation (7.6) can be further developed in a recursive form after taking sensor

measurements into consideration. It is [39]

$$\alpha(\mathbf{x0}^j, \mathbf{x1}^j) = \min \left(1, \frac{p(\mathbf{z}_k | \mathbf{x1}^j) p(\mathbf{x1}^j | \mathbf{x0}^j)}{p(\mathbf{z}_k | \mathbf{x0}^j) p(\mathbf{x0}^j | \mathbf{x0}^j)} \right). \quad (7.7)$$

To determine the statistical convergence to the posterior distribution of the contaminant source information, a convergence parameter G is computed from the between-chain variance B and the within-chain variance W [107] [108]. It is

$$G = \frac{n-1}{n} + \frac{B}{nW}, \quad (7.8)$$

where n is the length of Markov chains, and

$$B = \frac{n}{m-1} \sum_{j=1}^m (\bar{\mathbf{x}}^j - \bar{\mathbf{x}})^2, \quad (7.9)$$

$$W = \frac{1}{m} \sum_{j=1}^m \left(\frac{1}{n-1} \sum_{i=1}^n (\mathbf{x0}_i^j - \bar{\mathbf{x}}^j)^2 \right), \quad (7.10)$$

and

$$\bar{\mathbf{x}}^j = \frac{1}{n} \sum_{i=1}^n \mathbf{x0}_i^j, \quad (7.11)$$

$$\bar{\mathbf{x}} = \frac{1}{m} \sum_{j=1}^m \bar{\mathbf{x}}^j. \quad (7.12)$$

The iterative process is terminated when either statistical convergence is attained ($|G - 1| < \varepsilon$, where ε is a small predefined value), or the maximum length t_N of the Markov chains is reached.

Although the MCMC algorithm can estimate parameters in complex non-linear systems, it incurs high computational cost. This is because the MCMC

algorithm performs state estimation based on random moves iteratively, and only those valid moves contribute to the estimation. A large number of invalid moves consume a lot of computational resources. This causes the MCMC method to encounter difficulty in real-time applications. Thus, more advanced algorithms need to be developed.

7.3 Problem Formulation and Estimation Algorithms

The problem considered in this research is that a source emits contaminant material instantaneously in a building. A set of sensors are deployed in certain zones of the building to monitor the contaminant material intensities. The sensor measurements are sent to a fusion center to estimate the contaminant source location, the source emission rate and the release time every time interval.

This section formulates the contaminant source estimation problem as two types of dynamic estimation problems, namely, a single model dynamic estimation system and a multiple model dynamic estimation system. The following subsections describe these two systems and their corresponding estimation algorithms.

7.3.1 Single Model Formulation and MCMC Algorithm

As we described in Chapter 2, the dynamic estimation system can be described by two fundamental equations, namely, the state transition equation and the measurement equation. The state transition equation describes the evolution of

the target state with time, and the measurement equation converts the state to the measurement. These two equations are represented as

$$\mathbf{x}_k = f_{k-1}(\mathbf{x}_{k-1}) + \mathbf{w}_{k-1}, \quad (7.13)$$

$$\mathbf{z}_k = h_k(\mathbf{x}_k) + \mathbf{r}_k, \quad (7.14)$$

where $\mathbf{x}_k$ is the target state vector which normally consists of the parameters to be estimated, $\mathbf{z}_k$ is the measurement vector, $\mathbf{w}_{k-1}$ is the process noise to compensate the transition model error, and $\mathbf{r}_k$ is the measurement noise; $f_{k-1}(\cdot)$ and $h_k(\cdot)$ are the state transition function and the measurement function respectively.

In our problem, we estimate a single contaminant source with instantaneous emission. The state vector $\mathbf{x}_k$ and the measurement vector $\mathbf{z}_k$ are defined as

$$\mathbf{x}_k = [\rho \quad \tau \quad \lambda]^T, \quad (7.15)$$

$$\mathbf{z}_k = [z_1 \quad z_2 \quad \cdots \quad z_M]^T, \quad (7.16)$$

where ρ is the instantaneous emission rate of the source, τ is the release time, λ is the source zone index which represents the source location, z_i is the contaminant concentration detected by sensor i , for $i = 1, \dots, M$, with M being the number of sensors. In this research, the state $\mathbf{x}$ does not change with time, hence the transition equation can be simplified as

$$\mathbf{x}_k = \mathbf{I}_3 \mathbf{x}_{k-1}, \quad (7.17)$$

where $\mathbf{I}_3$ is the 3×3 identity matrix, and the process noise $\mathbf{w}_{k-1}$ becomes $\mathbf{0}$. The

measurement function $h_k(\cdot)$ is the transport model, which has been described in Section 7.2.1.

Since the transport model stored in the lookup table is a complex nonlinear model, the MCMC algorithm described in Section 7.2.2 can be applied to estimate the state. It is known that the MCMC algorithm incurs high computational cost and has slow convergence rate. We suggest a multiple model approach in Section 7.3.2.

7.3.2 Multiple Model Formulation and MM-UKF

To overcome the problems of high computational cost and slow convergence in the existing algorithms, we formulate the problem to a multiple model hybrid system. The approach is motivated by the Rao-Blackwellized particle filter, which is a sequential Monte Carlo sampling Bayesian estimator [37] [41]. The basic idea is to partition the state vector into two parts. The first part consists of the state elements which can be estimated by an analytical method such as a Kalman-based estimator. The second part consists of the remaining state elements, which normally have nonlinear non-Gaussian characteristics. The second part can then be solved by the Bayesian Monte Carlo approach. In this way, the computational effort is mainly devoted to the nonlinear non-Gaussian part, and a Kalman-based estimator provides fast convergence for the first part.

In this research, we categorize the estimation parameters into two parts: the emission rate and the release time, which are estimated by a Kalman-based filter, in the first part, and the discrete source location in the second part. Since the number of zones in an indoor environment is finite, random Bayesian Monte Carlo estimation of source location is unnecessary. Instead of obtaining the

location through iterative random moves, conducting a full search on all the zones is more efficient.

In estimation theory, the problem can be viewed as a hybrid multiple model approach. The hybrid system is constructed by all possible models when model uncertainty exists. In our problem, the contaminant source is located in one of the zones in a building, and the exact zone is uncertain to the observer. Thus, the problem can be formulated in a hybrid system with N models, where N is the number of zones in the building. Each model represents a particular zone, and performs estimation of the emission rate and the release time conditioned on the given zone. The final result is then obtained from the output of each model. It could be either a weighted sum of the individuals or the most possible one. In this research, we select the most possible model for the final output.

We propose an MM-UKF to estimate the contaminant source information. The MM-UKF consists of multiple UKFs being run in parallel. Each UKF performs estimation of the emission rate and the release time conditioned on a distinct zone within the building. The likelihood for each UKF can be computed based on the measurement residual. The source location is then determined by the zone of the UKF with the highest likelihood, and the source emission rate and the release time are the estimation result of the corresponding UKF. Figure 7.3 illustrates the structure of the multiple model approach formulated for our problem. N UKFs provide N individual estimates, and the LK function computes the likelihood (or weight) for each model. The final output of the hybrid system $\hat{\mathbf{x}}_k$ is the individual state with the highest weight.

Figure 7.3: The structure of the MM-UKF with N models

The models or hypotheses in the MM-UKF are constructed by

$$\begin{aligned}
 Model^1 & \left\{ \begin{array}{l} \lambda^1 = 1, \\ \mathbf{x}_k^1 = \mathbf{I}_2 \mathbf{x}_{k-1}^1, \\ \mathbf{z}_k = \mathbf{h}_k(\mathbf{x}_k^1, 1) + \mathbf{r}_k, \end{array} \right. \\
 Model^2 & \left\{ \begin{array}{l} \lambda^2 = 2, \\ \mathbf{x}_k^2 = \mathbf{I}_2 \mathbf{x}_{k-1}^2, \\ \mathbf{z}_k = \mathbf{h}_k(\mathbf{x}_k^2, 2) + \mathbf{r}_k, \end{array} \right. \\
 \vdots & \\
 Model^N & \left\{ \begin{array}{l} \lambda^N = N, \\ \mathbf{x}_k^N = \mathbf{I}_2 \mathbf{x}_{k-1}^N, \\ \mathbf{z}_k = \mathbf{h}_k(\mathbf{x}_k^N, N) + \mathbf{r}_k, \end{array} \right.
 \end{aligned} \tag{7.18}$$

where N is the number of models in the hybrid system, and it is equal to the number of zones in the building; λ is the source zone index; $\mathbf{x}_k^i$ is the state of model i , and it is defined as $[\rho^i \ \tau^i]^T$, with ρ^i and τ^i being the source emission rate and the release time of model i respectively; $\mathbf{z}_k$ is the sensor measurements defined in (7.16); $h(\cdot)$ is the transport model defined by the lookup table.

Given N states, their error covariances and weights $\mathbf{x}_{k-1}^i, \mathbf{P}_{k-1}^i, w_{k-1}^i$, where $i = 1, \dots, N$, the MM-UKF for time k is described in Table 7.2.

Table 7.2: MM-UKF at time k
for $i = 1$ to N

- $[\mathbf{x}_k^i, \mathbf{P}_k^i, \mathbf{v}_k^i, \mathbf{S}_k^i] = UKF(\mathbf{x}_{k-1}^i, \mathbf{P}_{k-1}^i, \mathbf{z}_k^i);$
- LK: $w_k^i = LK(\mathbf{v}_k^i, \mathbf{S}_k^i, w_{k-1}^i);$

end for

$$\hat{i} = \arg \max_{i=1:N} w_k^i;$$

$$\hat{\lambda}_k = \hat{i};$$

$$\hat{\mathbf{x}}_k = \mathbf{x}_k^{\hat{i}};$$

$$\mathbf{P}_k = \mathbf{P}_k^{\hat{i}}.$$

In the table, the outputs of the UKF for model i are the state $\mathbf{x}_k^i$, state error covariance $\mathbf{P}_k^i$, measurement residual $\mathbf{v}_k^i$, and residual covariance $\mathbf{S}_k^i$. The weight for model i is updated by

$$\begin{aligned} w_k^i &= LK(\mathbf{v}_k^i, \mathbf{S}_k^i, w_{k-1}^i) \\ &= \frac{1}{c} \mathcal{N}(\mathbf{v}_k^i; 0, \mathbf{S}_k^i) w_{k-1}^i \end{aligned} \quad (7.19)$$

where c is a normalizing factor.

The algorithm we introduced above is a static MM algorithm. The two optimal assumptions, namely, the correct model is in the model set, and the model does not vary with time, are valid in our problem. This is because that the source location is in one of the possible zones, and it is fixed.

7.4 Simulation Results and Analysis

A computer simulation study is conducted to assess the feasibility of the MM-UKF proposed in this research. The tests are conducted for a simulated 3-story building, which is described in Section 7.4.1. Two types of tests are carried out, namely, the MM-UKF performance on different sensor deployments and a comparison between the MM-UKF and the single model MCMC method. They are described in Section 7.4.2 and Section 7.4.3 respectively.

7.4.1 Simulation Scenario

The simulation test is conducted for a 3-story building. The floor plan of the building is displayed in Figure 7.4. It is partitioned into $N = 32$ zones. The links between any two zones indicate there is airflow between them. There are six zones in the building, namely, zones 4, 5, 14, 15, 27 and 28, excluded from the simulation study. These zones are negatively pressurized areas, where exhaust fans are installed. Chemicals/contaminants are dispelled out of the building instead of being dispersed into the other zones within the building in these zones. Some zones in the building belong to low dispersion zones. These zones are located at the end of the airflow paths in general. It is difficult for them to disperse chemicals/contaminants to their neighbours. The low dispersion zones in our simulation include zones 22, 24 and 25.

Limited number of sensors are deployed among the zones to monitor the building condition. The simplest sensor deployment is based on the following rules. Firstly, we place sensors at the junctions, which have the most connections to other zones. They are zones 1, 19 and 32. In the actual building, these three

Figure 7.4: Floor plan for a 3-story building

zones are common corridors. Secondly, we place sensors at the low dispersion zones, which are zones 22, 24 and 25. This is because they only disperse a small amount of chemicals to other zones, and the observability is very low if the estimation depends on the sensor measurements in other zones. After considering the rules mentioned above, six sensors are placed at zone 1, 19, 32,

22, 24 and 25, which are presented in grey in Figure 7.4. The measurements of these sensors are sent to the fusion center for contaminant source estimation every time cycle with interval $T = 10\text{s}$.

7.4.2 Performance of the MM-UKF on Different Sensor Deployments

To evaluate the performance of the proposed MM-UKF, a series of tests are conducted for the entire environment with different sensor deployments. We first test the MM-UKF on the simplest six-sensor deployment shown in Figure 7.4. Secondly, another test is conducted with an eight-sensor deployment. This is because some zones have similar dispersion patterns, and it is difficult to differentiate those zones promptly with six-sensor measurements.

In the tests, the emission rate ρ is set to 1kg/s , and the release time $\tau = 5T$. This means that the contaminant source starts to emit at the 5th time cycle with the instantaneous emission rate 1kg/s . The test is carried out on each of the 26 possible zones (exclude six negatively pressurized zones), namely, the source location $\lambda \in \{1, 2, 3, 6, 7, 8, 9, 10, 11, 12, 13, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 29, 30, 31, 32\}$. In each test, a simulator generates the sensor measurements every time cycle using lookup table based on ρ and τ for a particular source zone. Since the measurement accuracy of a chemical sensor is decreasing with the concentration, we assume the measurement error is proportional to the measurement value. Thus, the white Gaussian noise with standard deviation of 10% of measurement value is added on the simulated sensor measurement. The measurement error covariance $\mathbf{R}$ is the $M \times M$ diagonal matrix (M is the number of sensors). It adapts to the measurement values. For example, if the measurement is

$\mathbf{z}_k = [z_1 \ z_2 \ z_3 \ z_4 \ z_5 \ z_6]^T$, the measurement error covariance $\mathbf{R}_k$ is then approximated by

$$\mathbf{R}_k = \begin{bmatrix} 0.01z_1^2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.01z_2^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.01z_3^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.01z_4^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.01z_5^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.01z_6^2 \end{bmatrix}. \quad (7.20)$$

The initial state error covariance $\mathbf{P}_0$ for each UKF is the 2×2 diagonal matrix with diagonal elements $[0.2^2 \ 1^2]$. The model probabilities are initiated uniformly. All of them are set to $1/26$.

The test focuses on the duration taken in finding a contaminant source after emission. The number of time cycles is recorded from the release time to the time that the source location is estimated correctly. Here the correct estimation means that the MM-UKF estimates the source location correctly in three consecutive times. The errors on the emission rate ρ and release time τ are also recorded based on the third correct estimation of the MM-UKF. Table 7.3 displays the results obtained from 50 Monte Carlo runs for the six-sensor deployment.

From the table, we observe that the average on the duration of finding the sources is 6T. The mean errors on the emission rate and the release time are 0.064kg/s and 0.24T respectively. The contaminant source can be located promptly for most of the zones, and some of them achieve the fastest possible speed, namely 3T. Since we set a 3T buffer for declaring a correct estimation,

Table 7.3: MM-UKF performance on the 6-sensor deployment

zone	duration (T)	error on ρ (kg/s)	error on τ (T)
1*	6	0.029	0.2
19*	5	0.043	0.2
22*	3	0.069	0.4
24*	3	0.046	0.4
25*	3	0.058	0.3
32*	5	0.030	0.2
2	7	0.142	0.2
3	14	0.030	0.1
6	6	0.076	0.2
7	5	0.067	0.3
8	11	0.028	0.1
9	4	0.067	0.2
10	5	0.081	0.2
11	9	0.049	0.2
12	5	0.069	0.2
13	18	0.028	0.2
16	7	0.244	0.3
17	4	0.062	0.3
18	5	0.068	0.2
20	6	0.058	0.3
21	4	0.061	0.3
23	4	0.043	0.2
26	4	0.066	0.3
29	5	0.057	0.2
30	4	0.056	0.3
31	4	0.045	0.3
Avg	6.0	0.064	0.24

* indicates zone with sensor

the MM-UKF actually finds the correct location at the first time cycle for those fastest speed zones.

There are some zones, namely, zones 3, 8 and 13, where the sources need longer time to be located. This is because these zones have similar dispersion patterns to other zones, and cannot be distinguished by the measurements obtained from the six sensors in a short period. Figure 7.5 shows the likelihoods

Figure 7.5: Likelihoods at different time cycles when the source is located at zone 3

of all zones at different time cycles when the contaminant source is located at zone 3. It can be seen that zones 8 and 9 are competing with zone 3, especially

Figure 7.6: Likelihoods at different time cycles when the source is located at zone 13

for zone 8, which has been assigned a higher likelihood than the correct zone 3 at earlier stage. Figure 7.6 shows the likelihoods when the contaminant source

is located at zone 13. Clearly, zone 12 is competing with the correct zone 13.

To improve the performance, we add two more sensors at zones 8 and 13. The test is re-conducted, and the results are shown in Table 7.4. It can be seen that the performance improved with this 8-sensor deployment. The average durations of finding the source is improved to 4.6T. The durations of finding the source for all zones are fewer than 10T. The errors on emission rate and release time do not increase much with these shorter durations.

7.4.3 Comparison on the MM-UKF and the MCMC method

A simulation test is carried out to compare the performances of the proposed MM-UKF and the MCMC method. The test is based on the 8-sensor deployment. It is known that better estimation accuracy can be obtained with a longer duration (more time cycles). To make a fair comparison, the errors on the estimation rate ρ and the release time τ are all computed based on the estimation results at time cycle 40. The test results are shown in Table 7.5.

It can be seen that the MM-UKF apparently outperforms the MCMC method. The average durations of finding the source is around 4T faster than the MCMC method. The MM-UKF also provides much better accuracy on the estimations of ρ and τ .

The efficiency is also compared between these two algorithms. The MM-UKF spends 0.5s to complete the process of 40 time cycles, whereas the MCMC method uses 0.8s for the same time cycles. Thus, we can conclude that the MM-UKF provides a better estimation in terms of accuracy and efficiency.

The computational costs of both the MM-UKF and MCMC algorithm in-

Table 7.4: MM-UKF performance on the 8-sensor deployment

zone	duration (T)	error on ρ (kg/s)	error on τ (T)
1*	5	0.040	0.2
8*	3	0.047	0.2
13*	3	0.044	0.3
19*	4	0.039	0.3
22*	3	0.062	0.4
24*	3	0.061	0.3
25*	3	0.066	0.4
32*	5	0.025	0.2
2	4	0.073	0.3
3	8	0.030	0.2
6	5	0.052	0.1
7	5	0.033	0.2
9	4	0.083	0.3
10	5	0.047	0.1
11	9	0.020	0.2
12	4	0.046	0.3
16	7	0.478	0.5
17	4	0.048	0.3
18	5	0.048	0.2
20	6	0.030	0.3
21	4	0.073	0.3
23	4	0.035	0.3
26	4	0.048	0.3
29	5	0.049	0.2
30	4	0.041	0.3
31	4	0.046	0.2
Avg	4.6	0.064	0.27

* indicates zone with sensor

crease with the number of zones in a building. The computational complexity of the MM-UKF is $O(N)$, where N is the number of zones in a building. For the MCMC algorithm, its computational complexity is $O(k)$, where k is the number of iterations. It is understood that k increases with the number of zones, although the specific value of k cannot be determined without experiments. To implement the proposed method on a large building problem, for example, a

Table 7.5: Performance comparison on the MM-UKF and the single model MCMC method

zone	GB-UKF			MCMC		
	duration (T)	error on ρ (kg/s)	error on τ (T)	duration (T)	error on ρ (kg/s)	error on τ (T)
1*	5	0.040	0.2	3	0.082	0.5
8*	3	0.047	0.2	3	0.070	0.5
13*	3	0.044	0.3	3	0.079	0.3
19*	4	0.039	0.3	3	0.063	0.5
22*	3	0.062	0.4	3	0.082	0.4
24*	3	0.061	0.3	3	0.092	0.5
25*	3	0.066	0.4	3	0.067	0.5
32*	5	0.025	0.2	3	0.098	0.5
2	4	0.073	0.3	5	0.077	0.5
3	8	0.030	0.2	7	0.063	0.5
6	5	0.052	0.1	9	0.105	0.4
7	5	0.033	0.2	14	0.143	0.5
9	4	0.083	0.3	15	0.303	0.4
10	5	0.047	0.1	7	0.079	0.4
11	9	0.020	0.2	5	0.240	0.5
12	4	0.046	0.3	18	0.212	0.5
16	7	0.478	0.5	14	0.073	1.1
17	4	0.048	0.3	4	0.054	0.4
18	5	0.048	0.2	10	0.069	0.4
20	6	0.030	0.3	17	0.362	0.4
21	4	0.073	0.3	4	0.111	0.6
23	4	0.035	0.3	13	0.320	0.5
26	4	0.048	0.3	18	0.342	0.6
29	5	0.049	0.2	6	0.294	0.5
30	4	0.041	0.3	4	0.061	0.4
31	4	0.046	0.2	24	0.349	0.5
Avg	4.6	0.064	0.27	8.4	0.150	0.50

* indicates zone with sensor

hotel with thousands of rooms, the algorithm can be enhanced to a variable-structure MM approach [60], which only applies estimation on those possible zones.

7.5 Conclusion

In this research, an MM-UKF is introduced for the task of real-time contaminant source estimation in a given indoor environment. A simulation study has been carried out to evaluate the practicality of the proposed approach. The indoor area of interest is a multi-zone environment within a 3-story building. The simulation test results corroborate the feasibility of the MM-UKF. Generally, the method can achieve accurate estimation of the required contaminant source characteristics (location, emission rate, and release time) in real-time.

To provide justification for applying the MM-UKF to the contaminant source estimation problem, a simulation test has been conducted to compare the performances of the MM-UKF and the MCMC method. The numerical results show that the MM-UKF outperforms the MCMC method in terms of estimation accuracy and computational efficiency.

For further research, the proposed MM-UKF can be applied to problems which are more complex, such as estimation with different emission types in missed detection and false alarm environments.

Chapter 8

Conclusion and Future Work

8.1 Conclusion

In this research, we have proposed a nonlinear model to track an accelerated target. The proposed model takes the correlation among the Cartesian coordinates into consideration, and can provide better estimation accuracy than the existing acceleration model.

We have also developed a pruning-based MM filter with STDN switching (STDN-PMM filter). This filter can deploy any sojourn time pdf in its STDN switching. We have applied the STDN-PMM filter to two tracking problems, namely, maneuvering target tracking and target tracking in the presence of outliers. The simulation results demonstrate that the STDN-PMM filter yields better accuracy than those commonly used MM filters.

We have extended the MM approach to some new applications, namely, locating an RF emitter when a detection system with distorted antennas and estimating contaminant source in an indoor environment.

In RF emitter localization, we have proposed a new approach to formulate the problem as a dynamic estimation problem with augmented state vector consisting of emitter location and antenna distortion parameters, such that antenna auto-calibration and target localization can be performed simultaneously. An MM-UKF has been suggested to deal with uncertainty of the distortion. The multiple models are constructed by multiple initial states uniformly distributed in a known distortion range. Through a static multiple model approach, the higher probabilities are assigned to the correct models automatically to achieve better estimation accuracy. We have applied this new approach to two antenna systems, namely, the spiral antenna system and the two-element array antenna system. The observability analysis for the two systems has been conducted. In the simulation and real tests, the proposed new approach with MM-UKF obviously outperforms the existing methods.

The MM approach has also been introduced to the contaminant source estimation problem by formulating uncertainty of source location to a hybrid system with multiple models. Each model represents a zone in the building. A UKF is deployed in each model to estimate the source emission rate and the release time, conditioned on the given zone of the model. This MM-UKF algorithm is tested in a simulated 3-story building with a contaminant source emitting chemical instantaneously. Results demonstrate the MM-UKF's performance on different sensor deployments. A comparison is also made between the proposed MM-UKF and the existing MCMC estimation with single model. The MM-UKF shows better estimation accuracy with less computational cost.

The approaches proposed in this research have been tested through simulated or real data. Most of the measurement noises in the simulated data are modeled

as white Gaussian noises. In real tracking and localization applications, heavy-tailed distributions are often observed in measurement noises. Since a Gaussian is the easiest pdf to cope with analytical estimation algorithms, approximating a heavy-tailed pdf to a Gaussian is still preferred. However, other methods have to be used when a real pdf has severely deformation from Gaussian. Particle Filter (PF) is an effective method to solve the problem. It models a pdf through a large number of particles and estimates a state numerically, and can therefore cope with non-Gaussian pdfs. Since the number of particle is large, a PF involves very high computational cost [36]. This may result in real-time implementation problem for some applications. An alternative method is using a Gaussian mixture to approximate a non-Gaussian pdf. We have presented this method to model measurement noise with outlier presence in Chapter 4.4.2. An MM approach with multiple noise models is then applied to perform estimation without high computational cost.

8.2 Future Work

The sojourn-time-dependant semi-Markov switching MM filter described in Chapter 4 can be extended to “track to measurement association” problem. Track to measurement association is a challenging problem in low detection and highly cluttered environment, especially when close targets exist. The Multiple Hypothesis Tracking (MHT) algorithm is often applied to perform association between tracks and measurements. STDN switching can be introduced to the computation of association switching probability between “detection” and “missed detection” in each hypothesis. The hypotheses in association can combine with the hypotheses in target transition model. A hypothesis tree with such a com-

bination is shown in Figure 8.1. We would like to investigate a joint sojourn time probability of model switching and association switching to contribute a more accurate hypothesis probability.

Figure 8.1: Hypothesis tree with association and motion model combination, where CV is the constant velocity model, CA is the constant acceleration model, CT is the constant turn model, and m1 is a measurement in the association gate.

The contaminant source estimation approach described in Chapter 7 works on the instantaneous emission type. Further research could be extended to include estimation on other emission types, such as continuous emission, exponential emission, etc. These emission types are much more complicated, as sensor detection is not proportional to the emission rate. This means that different emission rates will have different lookup tables. One approach to cope with these complicated emission types is using multiple measurement models. We would like to investigate this multiple measurement model approach. Furthermore, the real sensors used in the sensor network are with large quantization error. An estimation algorithm should be able to address such errors. We would also like to conduct further research in this area.

Author's Publications

Journal

1. Rong Yang, Pek Hui Foo, Boon Poh Ng, Gee Wah Ng, "RF emitter geolocation using amplitude comparison with auto-calibrated relative antenna gains", *IEEE Transactions on Aerospace and Electronic Systems*, vol. 47, no. 3, pp 2098-2110, July 2011.
2. Rong Yang, Boon Poh Ng, Gee Wah Ng, "WTDM-based M³H filter for target tracking in the presence of outliers", *IEEE Signal Processing Letters*, vol. 17, no. 7, pp 623-626, July 2010.
3. Rong Yang, Boon Poh Ng, Gee Wah Ng, "Multiple model multiple hypothesis with sojourn-time-dependent semi-Markov switching", *IEEE Signal Processing Letters*, vol. 16, no. 6, pp 529-532, June 2009.
4. Rong Yang, Boon Poh Ng, Gee Wah Ng, "RF emitter localization with antenna auto-calibration capability", *IET Radar, Sonar and Navigation*, submitted, 2011.
5. Rong Yang, Hai Leong Chieu, Boon Poh Ng, Gee Wah Ng, "A pruning based multiple model filter with sojourn-time-dependent semi-Markov switching", *IEEE Transactions on Aerospace and Electronic Systems*, submitted, 2011.
6. Felix Govaers, Rong Yang, Hoe Chee Lai, Wolfgang Koch, Loo Nin Teow, Gee Wah Ng, "Track-before-detect in distributed sensor applications", *EURASIP Journal on Advances in Signal Processing*, 2011:20, Jul. 2011.
7. Pek Hui Foo, Gee Wah Ng, Khin Hua Ng, Rong Yang, "Application of intent inference for air defense and conformance monitoring", *Journal of Advances in Information Fusion*, vol 4, no 1, pp 3-26, June 2009.

Conference

1. Rong Yang, Boon Poh Ng, Gee Wah Ng, "Tracking an accelerated target with a nonlinear constant heading model", *Proceedings of the 13th International Conference on Information Fusion*, Edinburgh, UK, July 2010.
2. Rong Yang, Gee Wah Ng, Ning Ma, Chin Swee Chia, "A bearing and heading tracker from a stationary sonar sensor", *Proceedings of IEEE Oceans Conference*, Singapore, May, 2006.
3. Rong Yang, Gee Wah Ng, "Deghosting in multi-passive acoustic sensors", *Proceedings of SPIE Conference on Multisource Information Fusion: Architectures, Algorithms, and Applications 2004*, USA, April, 2004.
4. Gee Wah Ng, Rong Yang (corresponding author), "Comparison of decentralized tracking algorithms", *Proceedings of the Sixth International Conference on Information Fusion*, Australia, vol. 1, pp. 107-113, July, 2003.
5. Rong Yang, Pek Hui Foo, Peng Yen Tan, Boon Poh Ng, Gee Wah Ng, "Real-time indoor contaminant source estimation using a multiple model unscented Kalman filter", *Proceedings of the 15th International Conference on Information Fusion*, submitted 2012.
6. Hoe Chee Lai, Rong Yang, Gee Wah Ng, "Enhanced self-organizing map for passive sonar tracking to improve situation awareness", *Proceedings of IEEE International Conference on Information Fusion*, USA, July, 2007.
7. Ning Ma, Chin Swee Chia, Rong Yang, Gee Wah Ng, "Target localization by two fixed linear arrays", *Proceedings of IEEE Oceans Conference*, Singapore, May, 2006.
8. Gee Wah Ng, Khin Hua Ng, Rong Yang, Pek Hui Foo, "Intent inference for attack aircraft through fusion", *Proceedings of SPIE Conference on Multisource Information Fusion: Architectures, Algorithms, and Applications 2006*, USA, April, 2006.

Bibliography

- [1] Y. Bar-Shalom and X. R. Li, *Multitarget-Multisensor Tracking: Principles and Techniques*, Storrs, CT: YBS Publishing, 1995.
- [2] S. S. Blackman and R. Popoli, *Design and Analysis of Modern Tracking Systems*, Norwood MA: Artech House, 1999.
- [3] T. Kirubarajan, Y. Bar-Shalom, and D. Lerro, "Bearings-only tracking of maneuvering targets using a batch-recursive estimator," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 37, no. 3, pp. 770-780, Jul. 2001.
- [4] X. R. Li and V. P. Jilkov, "Survey of maneuvering target tracking. Part V. Multiple-model methods," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 41, no. 4, pp 1255-1321, Oct. 2005.
- [5] —, "Survey of maneuvering target tracking. Part I: Dynamic models," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 39, no. 4, pp. 1333-1364, Oct. 2003.
- [6] H. Blom and Y. Bar-Shalom, "The interacting multiple model algorithm for systems with Markovian switching coefficients," *IEEE Transactions on Automatic Control*, vol. 33, no. 8, pp 780-783, Aug. 1988.
- [7] L. Campo, P. Mookerjee, and Y. Bar-Shalom, "State estimation for systems with sojourn-time-dependent Markov model switching," *IEEE Transactions on Automatic Control*, vol. 36, no. 2, pp 238-243, Feb. 1991.
- [8] H. Blom, "A sophisticated tracking algorithm for air traffic control surveillance," *Proceedings of the International Radar Conference*, pp. 393-398, May 1984.
- [9] G. A. Ackerson and K. S. Fu, "On state estimation in switching environments," *IEEE Transactions on Automatic Control*, vol. 15, no. 1, pp. 10-17, Jan. 1970.

- [10] A. G. Jaffer and S. C. Gupta, "On estimation of discrete processes under multiplicative and additive noise conditions," *Information Science*, vol. 3, pp. 267-276, 1971.
- [11] C. B. Chang and M. Athans, "State estimation for discrete systems with switching parameters," *IEEE Transactions on Aerospace and Electronic Systems*, vol. AES-14, no. 5, pp. 418-425, May 1978.
- [12] S. Sugimoto and I. Ishizuka, "Identification and estimation algorithms for Markov chain plus AR process," *In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 247-250, 1983.
- [13] Y. Boers and H. Driessen, "A multiple model multiple hypothesis filter for Markovian switching systems," *Automatica*, vol. 41, no. 4, pp. 709-716, 2005.
- [14] G. D. Forney, "The Viterbi algorithm," *Proceedings of the IEEE*, vol. 61, no. 3, pp. 268-278, Mar. 1973.
- [15] H. Driessen and Y. Boers, "A multiple model multiple hypothesis filter for tracking maneuvering targets," *Proceedings of the 2001 SPIE Conference on Signal and Data Processing of Small Targets*, vol. 4473, pp. 279-288, 2001.
- [16] Y. Boers and H. Driesen, "A multiple model multiple hypothesis filter for systems with possibly erroneous measurements," *Proceedings of the 2002 International Conference on Information Fusion*, pp. 700-704, Jul 2002.
- [17] R. L. Moose and P. P. Wang, "An adaptive estimator with learning for a plant containing semi-Markov switching parameters," *IEEE Transactions on Systems, Man and Cybernetics*, vol. SMC-3, pp. 277-281, May 1973.
- [18] R. L. Moose, "An adaptive state estimation solution to the maneuvering target problem," *IEEE Transactions on Automatic Control*, vol. 20, pp. 359-362, Jun. 1975.
- [19] N. H. Gholson, "Tracking maneuvering targets via semi-Markov maneuver modeling," Ph.D Thesis, Virginia Polytechnic Institute and State University, Feb. 1977.
- [20] N. H. Gholson and R. L. Moose, "Maneuvering target tracking using adaptive state estimation," *IEEE Transactions on Aerospace and Electronic Systems*, vol. AES-13, no. 3, pp. 310-318, May 1977.

- [21] R. L. Moose, H. F. VanLandingham, and D. H. McCabe, "Modeling and estimation for tracking maneuvering targets," *IEEE Transactions on Aerospace and Electronic Systems*, vol. AES-15, no. 3, pp. 448-456, May 1979.
- [22] K. V. Ramachandra, *Kalman Filtering Techniques for Radar Tracking*, New York: M. Dekker, 2000.
- [23] G. W. Ng, *Intelligent Systems - Fusion, Tracking and Control*, Baldock, Hertfordshire : Research Studies Press, 2003.
- [24] R. A. Singer, "Estimating optimal tracking filter performance for manned maneuvering targets," *IEEE Transactions on Aerospace and Electronic Systems*, vol. AES-6, no. 4, pp. 473-483, Jul. 1970.
- [25] G. A. Watson and W. D. Blair, "IMM algorithm for tracking targets that maneuver through coordinated turns," *In Proceedings of the 1992 SPIE Conference on Signal and Data Processing for Small Targets*, vol. 1698, pp. 236-247, 1992.
- [26] X. R. Li and Y. Bar-Shalom, "Design of an interacting multiple model algorithm for air traffic control tracking," *IEEE Transactions on Control Systems Technology*, vol. 1, no. 3, pp 186-194, Sep. 1993.
- [27] F. Gustafsson, *Adaptive Filtering and Change Detection*, New York: Wiley, 2001.
- [28] A. Gel, *Applied Optimal Estimation*, Cambridge, MA: MIT Press, 1974.
- [29] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation: Theory, Algorithms and Software*, New York: Wiley, 2001.
- [30] S. J. Julier and J. K. Uhlmann, "A new extension of the Kalman filter to nonlinear systems," *In Proceedings of AeroSense: The 11th International Symposium on Aerospace/Defence Sensing, Simulation and Controls*, Orlando, FL, 1997.
- [31] S. J. Julier, J. K. Uhlmann, and H. F. Durrant-Whyte, "A new method for the nonlinear transformation of means and covariances in filters and estimators," *IEEE Transactions on Automatic Control*, vol. 45, no. 3 pp. 477-482, Mar. 2000.

- [32] S. J. Julier and J. K. Uhlmann, "Unscented filtering and nonlinear estimation," *Proceedings of the IEEE*, vol. 92, no. 3, pp. 401-422, Mar. 2004.
- [33] N. J. Gordon, D. J. Salmond, and A. F. M. Smith, "Novel approach to nonlinear/non-Gaussian Bayesian state estimation," *Proceedings of IEE*, vol. 140, no. 2, pp. 107-113, Apr. 1993.
- [34] D. Simon, *Optimal State Estimation: Kalman, H_∞ , and Nonlinear Approaches*, Hoboken, New Jersey: John Wiley and Sons, 2006.
- [35] M. S. Arulampalam, N. J. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 174-188, Feb. 2002.
- [36] B. Ristic, S. Arulampalam, and N. Gordon, *Beyond the Kalman Filter: Particle Filters for Tracking Applications*, Boston: Artech House, 2004.
- [37] A. Doucet, S. Godsill, and C. Andrieu, "On sequential Monte Carlo sampling methods for Bayesian filtering," *Statistics and Computing*, vol. 10, no. 2, pp. 197-208, 2000.
- [38] A. Doucet, N. de Freitas, and N. Gordon, "An introduction to sequential Monte Carlo methods," in *Sequential Monte Carlo Methods in Practice*, New York: Springer-Verlag, 2001.
- [39] W. R. Gilks and C. Berzuini, "Following a moving target: Monte Carlo inference for dynamic Bayesian models," *Journal of the Royal Statistical Society: Series B*, vol. 63, pp. 127-146, Jan. 2001.
- [40] D. S. Lee and N. K. K. Chia, "A particle algorithm for sequential Bayesian parameter estimation and model selection," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 326-336, Feb. 2002.
- [41] G. Casella and C. P. Robert, "Rao-Blackwellisation of sampling schemes," *Biometrika*, vol. 83, pp. 81-94, 1996.
- [42] D. T. Magill, "Optimal adaptive estimation of sampled stochastic processes," *IEEE Transactions on Automatic Control*, vol. AC-10, pp. 434-439, 1965.
- [43] D. G. Lainiotis and F. L. Sims, "Performance measure for adaptive Kalman estimators," *IEEE Transactions on Automatic Control*, vol. AC-15, pp. 249-250, Apr. 1970.

- [44] Y. Oshman, J. Shinar, and S. A. Weizman, "Using a multiple model adaptive estimator in a random evasion missile/aircraft encounter," *AIAA Journal of Guidance, Control, and Dynamics*, vol. 24, no. 6, pp 176-1186, 2001.
- [45] D. W. Kyger and P. S. Maybeck, "Reducing lag in virtual display using multiple model adaptive estimation," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 34, no. 4, pp. 237-248, Oct. 1998.
- [46] P. F. Easthope, "Using TOTS for more accurate and responsive multi-sensor, end-to-end ballistic missile tracking," *In Proceedings of the 2000 SPIE Conference on Signal and Data Processing of Small Targets*, vol. 4048, Apr. 2000.
- [47] P. F. Easthope and N. W. Heys, "Multiple-model target-oriented tracking system," *In Proceedings of the 1994 SPIE Conference on Signal and Data Processing of Small Targets*, vol. 2235, Apr. 1994.
- [48] T. Shima, Y. Oshman, and J. Shinar, "Efficient multiple model adaptive estimation in ballistic missile interception scenarios," *AIAA Journal of Guidance, Control, and Dynamics*, vol. 25, no. 4, pp. 667-675, 2002.
- [49] P. Vacher and M. Gauthier, "Design of a tracking algorithm for an advanced ATC system," *Multitarget-multisensor Tracking: Applications and Advances*, vol. 2, Norwood, MA:Artech House, 1992.
- [50] W. D. Blair and G. A. Watson, "IMM algorithm for solution to benchmark problem for tracking maneuvering targets," *Proceedings of the SPIE Symposium on Acquisition, Tracking, and Pointing*, vol. 2221, pp. 476-488, Apr. 1994.
- [51] G. A. Watson and W. D. Blair, "Solution to second benchmark problems for tracking maneuvering targets in the presence of false alarms and ECM," *Proceedings of the SPIE Symposium on Signal and Data Processing of Small Targets*, vol. 2561, pp. 263-274, 1995.
- [52] E. Sviestins, "Multiradar tracking for theater missile defense," *Proceedings of the SPIE Symposium on Signal and Data Processing of Small Targets*, vol. 2561, pp. 384-394, Jul. 1995.
- [53] E. Mazar, A. Averbuch, Y. Bar-Shalom, and J. Dayan, "Interacting multiple model methods in target tracking: a survey," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 34, no. 1, pp. 103-123, Jan. 1998.

- [54] W. D. Blair and G. A. Watson, "Interacting multiple bias model algorithm with application to tracking maneuvering targets," *Proceedings of the 31st IEEE Conference on Decision and Control*, vol. 4, pp. 3790-3795, 1992.
- [55] A. Munir and D. P. Atherton, "Adaptive interacting multiple model algorithm for tracking a manoeuvring target," *IEE Proceedings of Radar, Sonar, and Navigation*, vol. 142, no. 1, pp. 11-17, Feb. 1995.
- [56] G. A. Watson and W. D. Blair, "Interacting acceleration compensation algorithm for tracking maneuvering targets," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 31, no. 3, pp. 1152-1159, Jul. 1995.
- [57] S. McGinnity and G. W. Irwin, "Fuzzy logic approach to maneuvering target tracking," *IEE Proceedings of Radar, Sonar, and Navigation*, vol. 145, no. 6, pp. 337-341, Dec. 1998.
- [58] Z. Ding, H. Leung, and K. Chan, "Model-set adaptation using a fuzzy Kalman filter," *Proceedings of the Third International Conference on Information Fusion*, vol. 1, pp. 10-13, Jul 2000.
- [59] D. G. Zuo, C. Z. Han, Z. Lin, H. Y. Zhu, and H. Hong, "Fuzzy multiple model tracking algorithm for manoeuvring target," *Proceedings of the Fifth International Conference on Information Fusion*, vol. 2, pp. 818-823, 2002.
- [60] X. R. Li and Y. Bar-Shalom, "Multiple-model estimation with variable structure," *IEEE Transactions on Automatic Control*, vol. 41, no. 4, pp. 478-493, Apr. 1996.
- [61] X. R. Li, "Multiple-model estimation with variable structure-part II: Model-set adaptation," *IEEE Transactions on Automatic Control*, vol. 45, no. 11, pp. 2047-2060, Nov. 2000.
- [62] X. R. Li, X. R. Zhi, and Y. M. Zhang, "Multiple-model estimation with variable structure-part III: Model-group switching algorithm," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 35, no. 1, pp. 225-241, Jan. 1999.
- [63] X. R. Li, Y. M. Zhang, and X. R. Zhi, "Multiple-model estimation with variable structure-part IV: Design and evaluation of model-group switching algorithm," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 35, no. 1, pp. 242-254, Jan. 1999.

- [64] X. R. Li and Y. M. Zhang, "Multiple-model estimation with variable structure-part V: Likely-model set algorithm," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 36, no. 2, pp. 448-466, Jan. 2000.
- [65] X. R. Li, V. P. Jilkov, and J. Ru, "Multiple-model estimation with variable structure-part VI: Expected-mode augmentation," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 41, no. 3, pp. 853-867, Jul. 2005.
- [66] A. Averbuch, S. Itzikowitz, and T. Kapon, "Radar target tracking-Viterbi versus IMM," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 27, no. 3, pp. 550-563, May 1991.
- [67] P. Tichavsky, "Posterior Cramer-Rao bounds for discrete-time nonlinear filtering," *IEEE Transactions on Signal Processing*, vol. 46, no. 5, pp. 1386-1396, May 1998.
- [68] K. R. P. T. Kirubarajan, Y. Bar-Shalom and I. Kadar, "Ground target tracking with variable structure IMM estimator," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 36, no. 1, pp. 26-46, Jan. 2000.
- [69] K. Kastella and C. Kreucher, "Multiple model nonlinear filtering for low signal ground target applications," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 41, no. 2, pp. 549-564, Apr. 2005.
- [70] S. Dan and C. T. Li, "Kalman filtering with state equality constraints," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 38, no. 1, pp. 128-136, Jan. 2002.
- [71] C. Jauffret and Y. Bar-Shalom, "Track formation with bearing and frequency measurements in clutter," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 26, no. 6, pp. 999-1010, Nov. 1990.
- [72] T. Brehard and J. P. Le Cadre, "Hierarchical particle filter for bearings-only tracking," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 43, no. 4, pp. 1567-1585, Oct. 2007.
- [73] T. H. Kerr, "Status of CR-like lower bounds for nonlinear filtering," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 25, no. 5, pp. 590-601, Sep. 1989.
- [74] A. Logothetis and V. Krishnamurthy, "Expectation maximization algorithms for map estimation of jump Markov linear systems," *IEEE Transactions on Signal Processing*, vol. 47, no. 8, pp. 2139-2156, Aug. 1999.

- [75] J. K. Tugnait, "Adaptive estimation and identification for discrete systems with markov jump parameters," *IEEE Transactions on Automa. Control*, vol. AC-27, pp. 1054-1065, Oct. 1982.
- [76] U. Orguner and M. Demirekler, "Maximum likelihood estimation of transition probabilities of jump Markov linear systems," *IEEE Transactions on Signal Processing*, vol. 56, no. 10, pp. 5093-5107, Oct. 2008.
- [77] R. A. Howard, "System analysis of semi-Markov processes," *IEEE Transactions on Military Electronics*, vol. 8, no. 2, pp. 114-124, Apr. 1964.
- [78] V. J. Mathews and J. K. Tugnait, "Detection and estimation with fixed lag for abruptly changing systems," *IEEE Transactions on Aerospace and Electronic Systems*, vol. AES-19, no. 5, pp. 730-739, Sep. 1983.
- [79] D. Svensson and L. Svensson, "A new multiple model filter with switch time conditions," *IEEE Transactions on Signal Processing*, vol. 58, no. 1, pp. 11-25, Aug. 2009.
- [80] R. H. Delano, "A theory of target glint or angular scintillation in radar tracking," *Proceedings of IRE*, vol. 41, pp. 1778-1784, Dec. 1952.
- [81] B. H. Borden and M. L. Mumford, "A statistical glint/radar cross section target model," *IEEE Transactions on Aerospace and Electronic Systems*, vol. AES-19, no. 1, pp. 781-785, Sep. 1983.
- [82] G. S. Sandhu and A. V. Saylor, "A real-time statistical radar target model," *IEEE Transactions on Aerospace and Electronic Systems*, vol. AES-21, no. 4, pp. 490-507, Jul. 1985.
- [83] G. A. Hewer, R. D. Martin, and J. Zeh, "Robust preprocessing for Kalman filtering of glint noise," *IEEE Transactions on Aerospace and Electronic Systems*, vol. AES-23, no. 1, pp. 120-128, Jan. 1987.
- [84] W. R. Wu, "Maximum likelihood identification of glint noise," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 32, no. 1, pp. 41-51, Jan. 1996.
- [85] W. R. Wu, "Target tracking with glint noise," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 29, no. 1, pp. 174-185, Jan. 1993.
- [86] C. Masreliez, "Approximate non-Gaussian filtering with linear state and observation relations," *IEEE Transactions on Automatic Control*, vol. 20, no. 1, pp. 107-110, Feb. 1975.

- [87] E. Daeipour and Y. Bar-Shalom, "An interacting multiple model approach for target tracking with glint noise," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 31, no. 2, pp. 706-715, Jul. 1995.
- [88] E. Daeipour and Y. Bar-Shalom, "IMM tracking of maneuvering targets in the presence of glint," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 34, no. 3, pp. 996-1003, Jul. 1998.
- [89] H. T. Hu, Z. L. Jing, A. P. Li, S. Q. Hu, and H. W. Tian, "An MCMC-based particle filter for tracking target in glint noise environment," *Proceedings of the 2004 International Conference on Information Fusion*, pp. 922-927, Jun. 2004.
- [90] R. Yang, B. P. Ng, and G. W. Ng, "Multiple model multiple hypothesis filter with sojourn-time-dependent semi-markov switching," *IEEE Signal Processing Letters*, vol. 16, no. 6, pp. 529-532, Jun. 2009.
- [91] L. G. Bullock, G. R. Oeh, and J. J. Sparagna, "An analysis of wide-band microwave monopulse direction-finding techniques," *IEEE Transactions on Aerospace and Electronic Systems*, vol. AES-7, no. 1, pp. 188-203, Jan. 1971.
- [92] B. M. Leiner, "An analysis and comparison of energy direction finding systems," *IEEE Transactions on Aerospace and Electronic Systems*, vol. AES-15, no. 6, pp. 861-873, Nov. 1979.
- [93] C. Brinegar, "Passive direction finding: combining amplitude and phase based methods," *Proceedings of the IEEE Conference on National Aerospace and Electronics Conference*, pp. 78-84, Oct. 2000.
- [94] E. E. Agrama, O. A. Abdel-Alim, and M. A. Ezz-El-Arab, "Second order discriminant function for amplitude comparison monopulse," *Conference Record of 7th IEEE Conference on Instrumentation and Measurement Technology Conference*, pp. 173-176, Feb. 1990.
- [95] R. Stansfield, "Statistical theory of DF fixing," *J. IEE* 94, vol. 15, pp. 762-770, Dec. 1947.
- [96] W. Foy, "Position-location solutions by Taylor-series estimation," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 12, no. 2, pp. 187-194, Mar. 1976.

- [97] C. Ancker, "Airborne direction finding the theory of navigation errors," *IRE Transactions on Aeronautical Navigational Electron*, vol. 5, pp. 199-210, Dec. 1958.
- [98] M. Wax, "Position location from sensors with position uncertainty," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 19, no. 5, pp. 658-662, Sep. 1983.
- [99] K. Spingarn, "Passive position location estimation using the extended Kalman filter," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 23, no. 4, pp. 558-567, Jul. 1987.
- [100] E. Boch and M. Stapleton, "Direction-finding performance of a Ka-band ESM receiver," *IEEE Microwave and Guided Wave Letters*, vol. 4, no. 1, pp. 9-10, Jan. 1994.
- [101] W. R. Gilks, S. Richardson, and D. J. Spiegelhalter, *Markov chain Monte Carlo in practice*, London: Chapman & Hall, 1996.
- [102] D. G. Lainiotis, "Optimal adaptive estimation: structure and parameter adaption," *IEEE Transactions on Automatic Control*, vol. AC-16, no. 2, pp. 160-170, Apr. 1971.
- [103] M. Gavish and A. J. Weiss, "Performance analysis of bearing-only target location algorithms," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 28, no. 3, pp. 817-828, Jul. 1992.
- [104] C.-T. Chen, *Linear System Theory and Design*, New York, NY, USA: Oxford University Press, 1998.
- [105] R. Kandepu, L. Imsland, and B. A. Foss, "Constrained state estimation using the unscented Kalman filter," *Proceedings of the 16th Mediterranean Conference on Control and Automation*, July 2008.
- [106] C. H. Lim, Y. H. Wan, B. P. Ng, and C. M. S. See, "A real-time indoor WiFi localization system utilizing smart antennas," *IEEE Transaction on Consumer Electronics*, vol. 53, no. 2, pp. 618-622, May 2007.
- [107] F. K. Chow, B. Kosovic, and S. Chan, "Source inversion for contaminant plume dispersion in urban environments using building-resolving simulations," *Proceedings of the 86th American Meteorological Society Annual Meeting*, atlanta, Georgia, USA, 2006.

- [108] —, “Source inversion for contaminant plume dispersion in urban environments using building-resolving simulations,” *Journal of Applied Meteorology and Climatology*, vol. 47, no. 6, pp. 1553–1572, 2008.
- [109] M. D. Sohn, P. Reynolds, N. Singh, and A. J. Gadgil, “Rapidly locating and characterizing pollutant releases in buildings,” *Air and Waste Management Association*, vol. 52, no. 12, pp. 1422–1432, 2002.
- [110] X. Liu and Z. Zhai, “Inverse modeling methods for indoor airborne pollutant tracking: literature review and fundamentals,” *Indoor Air*, vol. 17, no. 6, pp. 419–438, 2007.
- [111] K. S. Rao, “Source estimation methods for atmospheric dispersion,” *Atmospheric Environment*, vol. 41, no. 33, pp. 6964–6973, 2007.
- [112] S. M. Gorelick, B. Evans, and I. Remson, “Identifying sources of ground-water pollution: An optimization approach,” *Water Resources Research*, vol. 19, no. 3, pp. 779–790, 1983.
- [113] P. S. Mahar, “Optimal monitoring network and ground-water pollution source identification,” *Water Resources Planning and Management*, vol. 123, no. 4, pp. 199–207, 1997.
- [114] T. H. Skaggs and Z. J. Kabala, “Recovering the release history of a ground-water contaminant,” *Water Resources Research*, vol. 30, no. 1, pp. 71–79, 1994.
- [115] S. Alapati and Z. J. Kabala, “Recovering the release history of a ground-water contaminant using a non-linear least-squares method,” *Hydrol. Process*, vol. 14, pp. 1003–1016, 2000.
- [116] P. S. Mahar and B. Datta, “Identification of pollution sources in transient groundwater systems,” *Water Resources Management*, vol. 14, pp. 209–227, 2000.
- [117] —, “Optimal identification of ground-water pollution sources and parameter estimation,” *Water Resources Planning and Management*, vol. 127, no. 1, pp. 20–29, 2001.
- [118] T. H. Skaggs and Z. J. Kabala, “Recovering the history of a groundwater contaminant plume: method of quasi-reversibility,” *Water Resources Research*, vol. 31, pp. 2669–2673, 1995.

- [119] J. Atmadja and A. Bagtzoglou, “Groundwater pollution source identification using the backward beam equation method,” *Computational Methods for Subsurface Flow and Transport*, balkema Publishers, Rotterdam, Netherlands, pp. 397-404, 2000.
- [120] R. M. Neupauer and J. L. Wilson, “Adjoint method for obtaining backward-in-time location and travel time probabilities of a conservative groundwater contaminant,” *Water Resources Research*, vol. 35, pp. 3389–3398, 1999.
- [121] —, “Adjoint-derived location and travel time probabilities for a multi-dimensional groundwater system,” *Water Resources Research*, vol. 37, pp. 1657–1668, 2000.
- [122] A. Schaelin, V. Dorer, J. van der Maas, and A. Moser, “Improvement of multizone model predictions by detailed flow path values from CFD calculation,” *ASHRAE Transactions*, vol. 99, no. 2, pp. 709–720, 1993.
- [123] A. Musser, “An analysis of combined CFD and multizone IAQ model assembly issues,” *ASHRAE Transactions*, vol. 107, no. 1, pp. 371–382, 2001.
- [124] P. V. Nielsen, “The selection of turbulence models for prediction of room airflow,” *ASHRAE Transactions*, vol. 104, no. 1B, pp. 1119–1127, 1998.
- [125] J. D. Spengler and Q. Y. Chen, “Indoor air quality factors in designing a healthy building,” *Annual Review of Energy and the Environment*, vol. 25, no. 1B, pp. 567–600, 2000.
- [126] G. Emanuel, *Analytical Fluid Dynamics, 2nd Edition*, CRC Press LLC, 2000.
- [127] V. N. Christopoulos and S. Roumeliotis, “Adaptive sensing for instantaneous gas release parameter estimation,” *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, Apr. 2005.
- [128] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, “Equations of state calculations by fast computing machine,” *J. Chem. Phys.*, vol. 21, pp. 1087–1091, 1953.