

OUTLIER ROBUST FILTERS AND THEIR MULTIPLE MODEL EXTENSIONS

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

İLKNUR ŞAHİN BOZGAN

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
ELECTRICAL AND ELECTRONICS ENGINEERING

SEPTEMBER 2019

Approval of the thesis:

**OUTLIER ROBUST FILTERS AND THEIR MULTIPLE MODEL
EXTENSIONS**

submitted by **İLKNUR ŞAHİN BOZGAN** in partial fulfillment of the requirements
for the degree of **Master of Science in Electrical and Electronics Engineering De-
partment, Middle East Technical University** by,

Prof. Dr. Halil Kalıpçılar
Dean, Graduate School of **Natural and Applied Sciences** _____

Prof. Dr. İlkay Ulusoy
Head of Department, **Electrical and Electronics Engineering** _____

Assist. Prof. Dr. Emre Özkan
Supervisor, **Electrical and Electronics Engineering, METU** _____

Examining Committee Members:

Prof. Dr. Umut Orguner
Electrical and Electronics Engineering, METU _____

Assist. Prof. Dr. Emre Özkan
Electrical and Electronics Engineering, METU _____

Prof. Dr. Murat Efe
Electrical and Electronics Engineering, Ankara University _____

Assoc. Prof. Dr. Afşar Saranlı
Electrical and Electronics Engineering, METU _____

Assist. Prof. Dr. Mustafa Mert Ankaralı
Electrical and Electronics Engineering, METU _____

Date:

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Surname: İlknur Şahin Bozgan

Signature :

ABSTRACT

OUTLIER ROBUST FILTERS AND THEIR MULTIPLE MODEL EXTENSIONS

Şahin Bozgan, İlknur

M.S., Department of Electrical and Electronics Engineering

Supervisor: Assist. Prof. Dr. Emre Özkan

September 2019, 149 pages

Kalman filter (KF), which is an algorithm that is utilized to estimate unknown variables based on noisy measurements, has been successfully employed in many applications such as navigation, control, signal processing and target tracking. It is the optimum Bayesian filter in terms of mean square error (MSE) for linear Gaussian state-space models (SSMs). However, in many real world applications, the performance of KF degrades due to the presence of outliers in noises. Motivated by this problem, several algorithms have been proposed to provide robustness towards outliers. In this thesis, existing outlier robust filters are investigated regarding their theoretical derivations, validity of their assumptions, and performances. Furthermore, multi-model extensions of the filters are derived and the merits of the algorithms are illustrated in simulations.

Keywords: Kalman Filter, Student's-t Filter, Variational Bayesian, Interacting Multiple Model, Gaussian Distribution, Student's-t Distribution, Heavy-Tailed Noise, Tar-

get Tracking, Inverse Wishart, Gamma-Gaussian, Outliers, Robustness

ÖZ

AYKIRI DEĞER FİLTRELERİ VE ÇOKLU MODEL UZANTILARI

Şahin Bozgan, İlknur

Yüksek Lisans, Elektrik ve Elektronik Mühendisliği Bölümü

Tez Yöneticisi: Dr. Öğr. Üyesi. Emre Özkan

Eylül 2019 , 149 sayfa

Gürültülü ölçümlere dayanarak bilinmeyen değişkenleri tahmin etmek için kullanılan bir algoritma olan Kalman filtresi (KF), navigasyon, kontrol, sinyal işleme ve hedef takibi gibi birçok uygulamada kullanılmıştır. Bu filtre, doğrusal ve Gauss gürültülü durum-uzay modelleri için ortalama kare hatası cinsinden en iyi Bayes filtresidir. Ancak, birçok gerçek hayat uygulamasında, Kalman filtresinin performansı, gürültülerdeki aykırı değerlerin varlığı nedeniyle bozulur. Bu problemden yola çıkarak, aykırı değerlere karşı dayanıklılık sağlamak için çeşitli algoritmalar önerilmiştir. Bu tezde, mevcut aykırı değer filtrelerinin teorik çıkarımları, varsayımlarının geçerliliği ve performansları incelenmiştir. Buna ek olarak, filtrelerin çoklu model uzantıları türetilmiş ve algoritmaların kabiliyetleri simülasyonlarda gösterilmiştir.

Anahtar Kelimeler: Kalman Filtresi, Student's-t Filtresi, Varyasyonel Bayes, Etkileşimli Çoklu Model, Gauss Dağılımı, Student's-t Dağılımı, Ağır Kuyruklu Gürültü, Hedef Takibi, Inverse Wishart, Gamma-Gauss, Aykırı Değerler, Sağlamlık

To my family,

ACKNOWLEDGMENTS

First of all, I would like to express my sincere gratitude to my supervisor Assist. Prof. Dr. Emre Özkan for his continued guidance, encouragement and support throughout my research. He provided advice, and assistance that greatly enhanced my studies and experience.

I would like to thank my company ASELSAN A.Ş. for its support on my graduate education. I am also grateful to several of my colleagues and all seniors for their contributions on the improvement of my engineering skills.

I also have to thank my parents Rabia Şahin and Necmettin Şahin and my brothers Burak Şahin and Kürşat Şahin for their endless love, belief and help throughout my life. Moreover, I would like to thank my friends Dicle Türköne and Hatice Manisalı for their motivational supports and fellowships.

Most of all, I want to thank my dear husband Caner Bozgan for his love, motivational and technical support and always believing in me.

TABLE OF CONTENTS

ABSTRACT	v
ÖZ	vii
ACKNOWLEDGMENTS	ix
TABLE OF CONTENTS	x
LIST OF TABLES	xii
LIST OF FIGURES	xiv
LIST OF ABBREVIATIONS	xxviii
CHAPTERS	
1 INTRODUCTION	1
2 BACKGROUND	7
2.1 Bayesian Filtering and Smoothing	7
2.2 Kalman Filter and Smoother	9
2.2.1 Simulation Example: Implementing KF and KS for 2-D Target Tracking Problem	17
2.3 Interacting Multiple Model (IMM) Algorithm	19
2.3.1 Simulation Example: Implementing KF and IMM Algorithm for Tracking Problem of a Maneuvering Target in 2-D Space	23
2.4 Variational Bayesian Methods	27
3 MULTIPLE-MODEL EXTENSION OF STUDENT'S T FILTER (IMM-STF)	31

3.1	Student's t Filter	33
3.1.1	Simulation Example: Comparison of STF and KF	35
3.2	Derivation of IMM-STF	43
3.3	Performance Evaluation	49
4	OUTLIER ROBUST FILTERS USING VB APPROACH	61
4.1	Gamma-Gaussian (GG) Approach	61
4.1.1	Derivation of Variational Bayesian Algorithm using Gamma-Gaussian Approach (VB-GG)	64
4.2	Inverse Wishart (IW) Approach	70
4.2.1	Derivation of VB Algorithm using Inverse Wishart Approach (VB-IW)	70
4.3	Performance Evaluation	78
5	MULTIPLE-MODEL EXTENSIONS OF OUTLIER ROBUST FILTERS USING VB APPROACH	105
5.1	Multiple-Model Extension of VB-GG Algorithm (IMM-VB-GG)	106
5.2	Multiple-Model Extension of VB-IW Algorithm (IMM-VB-IW)	112
5.3	Performance Evaluation	120
6	CONCLUSION	143
	REFERENCES	145

LIST OF TABLES

TABLES

Table 3.1 ARMSEs of KF and STF for Case A	39
Table 3.2 ARMSEs of KF and STFs with different degrees of freedom parameters for Case A	40
Table 3.3 ARMSEs of KF and STF for Case B	41
Table 3.4 ARMSEs of KF and STF for Case C	43
Table 3.5 ARMSEs of IMM and IMM-STF for Case A for CV mode	53
Table 3.6 ARMSEs of IMM and IMM-STF for Case A for CT mode	54
Table 3.7 ARMSEs of IMM and IMM-STF for Case B for CV mode	54
Table 3.8 ARMSEs of IMM and IMM-STF for Case B for CT mode	59
Table 3.9 ARMSEs of IMM and IMM-STF for Case C for CV mode	59
Table 3.10 ARMSEs of IMM and IMM-STF for Case C for CT mode	60
Table 4.1 ARMSEs of KF, VB-GG, VB-IW and STF for Case 1	86
Table 4.2 ARMSEs of KF, VB-GG, VB-IW and STF for Case 2	86
Table 4.3 ARMSEs of KF, VB-GG, VB-IW and STF for Case 3	90
Table 4.4 ARMSEs of KF, VB-GG, VB-IW and STF for Case 4	92
Table 4.5 ARMSEs of KF, VB-GG, VB-IW and STF for Case 5	94
Table 4.6 ARMSEs of KF, VB-GG, fixed prior VB-IW and STF for Case 3	98

Table 4.7	ARMSEs of KF, VB-GG, fixed prior VB-IW and STF for Case 4 . . .	98
Table 4.8	ARMSEs of KF, VB-GG, fixed prior VB-IW and STF for Case 5 . . .	98
Table 5.1	ARMSEs of IMM-VB-GG and IMM-VB-IW for Case 1	123
Table 5.2	ARMSEs of IMM-VB-GG and IMM-VB-IW for Case 2	128
Table 5.3	ARMSEs of IMM-VB-GG and IMM-VB-IW for Case 3	128
Table 5.4	ARMSEs of IMM-VB-GG and IMM-VB-IW for Case 4	129
Table 5.5	ARMSEs of IMM-VB-GG and IMM-VB-IW for Case 5	129

LIST OF FIGURES

FIGURES

Figure 2.1	True and estimated trajectories of the target. True trajectory of the target is shown by blue line, the estimated trajectory by KF is shown by orange dashed line and the estimated trajectory by KS is shown by green dash-dot line. The measurements are shown by black stars. . . .	18
Figure 2.2	Position errors of KF and KS. The error of KF is shown by orange line and the error of KS is shown by green line.	18
Figure 2.3	The block diagram of a single step of IMM algorithm for N-models.	22
Figure 2.4	True and estimated trajectories of the target. True trajectory of the target is shown by blue line, the estimated trajectory by IMM algorithm is shown by green dash-dot line and the estimated trajectory by the single KF is shown by orange dashed line. The measurements are shown by black stars.	25
Figure 2.5	Position errors of IMM algorithm and a single KF (with CV model) for all time steps. The error of IMM algorithm is shown by green line and the error of KF is shown by orange line.	26
Figure 2.6	Velocity errors of IMM algorithm and a single KF (with CV model) for all time steps. The error of IMM algorithm is shown by green line and the error of KF is shown by orange line.	27
Figure 2.7	Mode Probabilities of CV and CT models. The mode probability of CV model is shown by blue line and the mode probability of CT model is shown by red line.	28

Figure 3.1	PDFs of Student's-t distributions with different degrees of freedom values and a Gaussian distribution.	32
Figure 3.2	Position errors by STF and KF for Case A. Only a specific time interval is shown. The error of KF is shown by orange line and the error of STF is shown by green line.	37
Figure 3.3	RMSEs of the position for 500 Monte Carlo run for Case A. RMSEs of the position by KF is shown by orange line and RMSEs of the position by STF is shown by green line.	38
Figure 3.4	RMSEs of the velocity for 500 Monte Carlo run for Case A. RMSEs of the velocity by KF is shown by orange line and RMSEs of the velocity by STF is shown by green line.	38
Figure 3.5	RMSEs of the position for KF and STFs with different degrees of freedom parameters for 500 Monte Carlo run for Case A.	39
Figure 3.6	RMSEs of the position for 500 Monte Carlo run for Case B. RMSEs of the position by KF is shown by orange line and RMSEs of the position by STF is shown by green line.	40
Figure 3.7	RMSEs of the velocity for 500 Monte Carlo run for Case B. RMSEs of the velocity by KF is shown by orange line and RMSEs of the velocity by STF is shown by green line.	41
Figure 3.8	RMSEs of the position for 500 Monte Carlo run for Case C. RMSEs of the position by KF is shown by orange line and RMSEs of the position by STF is shown by green line.	42
Figure 3.9	RMSEs of the velocity for 500 Monte Carlo run for Case C. RMSEs of the velocity by KF is shown by orange line and RMSEs of the velocity by STF is shown by green line.	42
Figure 3.10	The block diagram of a single step of IMM-STF for N-models.	44

Figure 3.11 True and estimated trajectories of the target for Case A. True trajectory is shown by blue line, the trajectory estimated by IMM algorithm is shown by orange line and the trajectory estimated by IMM-STF is shown by green line. 51

Figure 3.12 Position errors of IMM and IMM-STF algorithms for time interval $k=0$ and $k=100$ for Case A. In this time interval, the target moves according to CV model. The error of IMM is shown by orange line and the error of IMM-STF is shown by green line. 52

Figure 3.13 Position errors of IMM and IMM-STF algorithms for time interval $k=101$ and $k=150$ for Case A. In this time interval, the target moves according to CT model. The error of IMM is shown by orange line and the error of IMM-STF is shown by green line. 53

Figure 3.14 Position errors of IMM and IMM-STF algorithms for time interval $k=151$ and $k=250$ for Case A. In this time interval, the target moves according to CV model. The error of IMM is shown by orange line and the error of IMM-STF is shown by green line. 54

Figure 3.15 Velocity errors of IMM and IMM-STF algorithms for time interval $k=0$ and $k=100$ for Case A. In this time interval, the target moves according to CV model. The error of IMM is shown by orange line and the error of IMM-STF is shown by green line. 55

Figure 3.16 Velocity errors of IMM and IMM-STF algorithms for time interval $k=101$ and $k=150$ for Case A. In this time interval, the target moves according to CT model. The error of IMM is shown by orange line and the error of IMM-STF is shown by green line. 55

Figure 3.17 Velocity errors of IMM and IMM-STF algorithms for time interval $k=151$ and $k=250$ for Case A. In this time interval, the target moves according to CV model. The error of IMM is shown by orange line and the error of IMM-STF is shown by green line. 56

Figure 3.18	Mode probabilities of CV and CT models for IMM-STF for Case A. The mode probability of CV model is shown by blue line and the mode probability of CT model is shown by red line.	56
Figure 3.19	RMSEs of the position for 250 Monte Carlo run for Case A. RMSEs of the position by IMM is shown by orange line and RMSEs of the position by IMM-STF is shown by green line.	57
Figure 3.20	RMSEs of the velocity for 250 Monte Carlo run for Case A. RMSEs of the velocity by IMM is shown by orange line and RMSEs of the velocity by IMM-STF is shown by green line.	57
Figure 3.21	RMSEs of the position for 250 Monte Carlo run for Case B. RMSEs of the position by IMM is shown by orange line and RMSEs of the position by IMM-STF is shown by green line.	58
Figure 3.22	RMSEs of the velocity for 250 Monte Carlo run for Case B. RMSEs of the velocity by IMM is shown by orange line and RMSEs of the velocity by IMM-STF is shown by green line.	58
Figure 3.23	RMSEs of the position for 250 Monte Carlo run for Case C. RMSEs of the position by IMM is shown by orange line and RMSEs of the position by IMM-STF is shown by green line.	59
Figure 3.24	RMSEs of the velocity for 250 Monte Carlo run for Case C. RMSEs of the velocity by IMM is shown by orange line and RMSEs of the velocity by IMM-STF is shown by green line.	60
Figure 4.1	The nominal covariance, outlier covariances and the effective covariance for Case 1. The nominal covariance R_0 is shown by blue circle, the outlier covariance R_1 is shown by orange circle, the outlier covariance R_2 is shown by green circle and the effective covariance is shown by purple ellipse.	79

Figure 4.2	The nominal covariance, outlier covariances and the effective covariance for Case 2. The nominal covariance R_0 is shown by blue circle, the outlier covariance R_1 is shown by orange circle, the outlier covariance R_2 is shown by green circle and the effective covariance is shown by purple ellipse.	80
Figure 4.3	The nominal covariance, the outlier covariance and the effective covariance for Case 3. The nominal covariance R_0 is shown by blue circle, the outlier covariance $100 \times R_0$ is shown by orange circle and the effective covariance is shown by purple circle.	81
Figure 4.4	The nominal covariance, outlier covariances and the effective covariance for Case 4. The nominal covariance R_0 is shown by blue circle, the outlier covariance R_1 is shown by orange circle, the outlier covariance R_2 is shown by green circle and the effective covariance is shown by purple ellipse.	82
Figure 4.5	The nominal covariance, outlier covariances and the effective covariance for Case 5. The nominal covariance R_0 is shown by blue circle, the outlier covariance R_1 is shown by orange circle, the outlier covariance R_2 is shown by green circle and the effective covariance is shown by purple ellipse.	83
Figure 4.6	RMSEs of the position for 500 Monte Carlo run for Case 1. RMSEs of the position by VB-GG is shown by orange line, RMSEs of the position by VB-IW is shown by green line, RMSEs of the position by STF is shown by purple line and RMSEs of the position by KF is shown by blue line.	84
Figure 4.7	RMSEs of the velocity for 500 Monte Carlo run for Case 1. RMSEs of the velocity by VB-GG is shown by orange line, RMSEs of the velocity by VB-IW is shown by green line, RMSEs of the velocity by STF is shown by purple line and RMSEs of the velocity by KF is shown by blue line.	85

Figure 4.8 Measurement noise covariance tracking of the algorithms VB-GG and VB-IW for Case 1 (for any time interval of a single MC run). The effective measurement noise covariances are shown by purple ellipses, the estimated measurement noise covariances by VB-IW are shown by green ellipses and the effective covariance calculated by VB-GG used in the measurement update of the state in VB iterations are shown by orange circles. The blue circles and black stars show the true positions and the measurements, respectively. 87

Figure 4.9 RMSEs of the position for 500 Monte Carlo run for Case 2. RMSEs of the position by VB-GG is shown by orange line, RMSEs of the position by VB-IW is shown by green line, RMSEs of the position by STF is shown by purple line and RMSEs of the position by KF is shown by blue line. 88

Figure 4.10 RMSEs of the velocity for 500 Monte Carlo run for Case 2. RMSEs of the velocity by VB-GG is shown by orange line, RMSEs of the velocity by VB-IW is shown by green line, RMSEs of the velocity by STF is shown by purple line and RMSEs of the velocity by KF is shown by blue line. 89

Figure 4.11 Measurement noise covariance tracking of the algorithms VB-GG and VB-IW for Case 2 (for any time interval of a single MC run). The effective measurement noise covariances are shown by purple ellipses, the estimated measurement noise covariances by VB-IW are shown by green ellipses and the effective covariance calculated by VB-GG used in the measurement update of the state in VB iterations are shown by orange circles. The blue circles and black stars show the true positions and the measurements, respectively. 90

Figure 4.12 RMSEs of the position for 500 Monte Carlo run for Case 3. RMSEs of the position by VB-GG is shown by orange line, RMSEs of the position by VB-IW is shown by green line, RMSEs of the position by STF is shown by purple line and RMSEs of the position by KF is shown by blue line. 91

Figure 4.13	RMSEs of the velocity for 500 Monte Carlo run for Case 3. RMSEs of the velocity by VB-GG is shown by orange line, RMSEs of the velocity by VB-IW is shown by green line, RMSEs of the velocity by STF is shown by purple line and RMSEs of the velocity by KF is shown by blue line.	91
Figure 4.14	Measurement noise covariance tracking of the algorithms VB-GG and VB-IW for Case 3 (for any time interval of a single MC run). The effective measurement noise covariances are shown by purple ellipses, the estimated measurement noise covariances by VB-IW are shown by green ellipses and the effective covariance calculated by VB-GG used in the measurement update of the state in VB iterations are shown by orange circles. The blue circles and black stars show the true positions and the measurements, respectively.	92
Figure 4.15	RMSEs of the position for 500 Monte Carlo run for Case 4. RMSEs of the position by VB-GG is shown by orange line, RMSEs of the position by VB-IW is shown by green line, RMSEs of the position by STF is shown by purple line and RMSEs of the position by KF is shown by blue line.	93
Figure 4.16	RMSEs of the velocity for 500 Monte Carlo run for Case 4. RMSEs of the velocity by VB-GG is shown by orange line, RMSEs of the velocity by VB-IW is shown by green line, RMSEs of the velocity by STF is shown by purple line and RMSEs of the velocity by KF is shown by blue line.	93
Figure 4.17	Measurement noise covariance tracking of the algorithms VB-GG and VB-IW for Case 4 (for any time interval of a single MC run). The effective measurement noise covariances are shown by purple ellipses, the estimated measurement noise covariances by VB-IW are shown by green ellipses and the effective covariance calculated by VB-GG used in the measurement update of the state in VB iterations are shown by orange circles. The blue circles and black stars show the true positions and the measurements, respectively.	94

Figure 4.18 RMSEs of the position for 500 Monte Carlo run for Case 5. RMSEs of the position by VB-GG is shown by orange line, RMSEs of the position by VB-IW is shown by green line, RMSEs of the position by STF is shown by purple line and RMSEs of the position by KF is shown by blue line. 95

Figure 4.19 RMSEs of the velocity for 500 Monte Carlo run for Case 5. RMSEs of the velocity by VB-GG is shown by orange line, RMSEs of the velocity by VB-IW is shown by green line, RMSEs of the velocity by STF is shown by purple line and RMSEs of the velocity by KF is shown by blue line. 95

Figure 4.20 Measurement noise covariance tracking of the algorithms VB-GG and VB-IW for Case 5 (for any time interval of a single MC run). The effective measurement noise covariances are shown by purple ellipses, the estimated measurement noise covariances by VB-IW are shown by green ellipses and the effective covariance calculated by VB-GG used in the measurement update of the state in VB iterations are shown by orange circles. The blue circles and black stars show the true positions and the measurements, respectively. 96

Figure 4.21 RMSEs of the position for 500 Monte Carlo run for Case 3 by using fixed prior VB-IW algorithm. RMSEs of the position by VB-GG is shown by orange line, RMSEs of the position by fixed prior VB-IW is shown by green line, RMSEs of the position by STF is shown by purple line and RMSEs of the position by KF is shown by blue line. . . . 97

Figure 4.22 RMSEs of the velocity for 500 Monte Carlo run for Case 3 by using fixed prior VB-IW algorithm. RMSEs of the velocity by VB-GG is shown by orange line, RMSEs of the velocity by fixed prior VB-IW is shown by green line, RMSEs of the velocity by STF is shown by purple line and RMSEs of the velocity by KF is shown by blue line. . . . 97

Figure 4.23 Measurement noise covariance tracking of the algorithms VB-GG and fixed prior VB-IW for Case 3 (for any time interval of a single MC run). The effective measurement noise covariances are shown by purple ellipses, the effective covariance calculated by fixed prior VB-IW used in the measurement update of the state in VB iterations are shown by green ellipses and the effective covariance calculated by VB-GG used in the measurement update of the state in VB iterations are shown by orange circles. The blue circles and black stars show the true positions and the measurements, respectively. 99

Figure 4.24 RMSEs of the position for 500 Monte Carlo run for Case 4 by using fixed prior VB-IW algorithm. RMSEs of the position by VB-GG is shown by orange line, RMSEs of the position by fixed prior VB-IW is shown by green line, RMSEs of the position by STF is shown by purple line and RMSEs of the position by KF is shown by blue line. . . 100

Figure 4.25 RMSEs of the velocity for 500 Monte Carlo run for Case 4 by using fixed prior VB-IW algorithm. RMSEs of the velocity by VB-GG is shown by orange line, RMSEs of the velocity by fixed prior VB-IW is shown by green line, RMSEs of the velocity by STF is shown by purple line and RMSEs of the velocity by KF is shown by blue line. . . 100

Figure 4.26 Measurement noise covariance tracking of the algorithms VB-GG and fixed prior VB-IW for Case 4 (for any time interval of a single MC run). The effective measurement noise covariances are shown by purple ellipses, the effective covariance calculated by fixed prior VB-IW used in the measurement update of the state in VB iterations are shown by green ellipses and the effective covariance calculated by VB-GG used in the measurement update of the state in VB iterations are shown by orange circles. The blue circles and black stars show the true positions and the measurements, respectively. 101

Figure 4.27	RMSEs of the position for 500 Monte Carlo run for Case 5 by using fixed prior VB-IW algorithm. RMSEs of the position by VB-GG is shown by orange line, RMSEs of the position fixed prior VB-IW is shown by green line, RMSEs of the position by STF is shown by purple line and RMSEs of the position by KF is shown by blue line.	102
Figure 4.28	RMSEs of the velocity for 500 Monte Carlo run for Case 5 by using fixed prior VB-IW algorithm. RMSEs of the velocity by VB-GG is shown by orange line, RMSEs of the velocity by fixed prior VB-IW is shown by green line, RMSEs of the velocity by STF is shown by purple line and RMSEs of the velocity by KF is shown by blue line. . .	102
Figure 4.29	Measurement noise covariance tracking of the algorithms VB-GG and fixed prior VB-IW for Case 5 (for any time interval of a single MC run). The effective measurement noise covariances are shown by purple ellipses, the effective covariance calculated by fixed prior VB-IW used in the measurement update of the state in VB iterations are shown by green ellipses and the effective covariance calculated by VB-GG used in the measurement update of the state in VB iterations are shown by orange circles. The blue circles and black stars show the true positions and the measurements, respectively.	103
Figure 5.1	The block diagram of a single step of IMM-VB-GG for N-models.	106
Figure 5.2	The block diagram of a single step of IMM-VB-IW for N-models.	113
Figure 5.3	RMSEs of the position for 250 Monte Carlo run for Case 1. RMSEs of the position by IMM-VB-GG is shown by orange line and RMSEs of the position by IMM-VB-IW is shown by green line. . . .	123
Figure 5.4	RMSEs of the velocity for 250 Monte Carlo run for Case 1. RMSEs of the velocity by IMM-VB-GG is shown by orange line and RMSEs of the velocity by IMM-VB-IW is shown by green line. . . .	124

- Figure 5.5 Measurement noise covariance tracking of the algorithms IMM-VB-GG and IMM-VB-IW for Case 1 (for any time interval of a single MC run). The effective measurement noise covariances are shown by purple ellipses, the estimated measurement noise covariances by IMM-VB-IW are shown by green ellipses and the effective covariance calculated by IMM-VB-GG used in the measurement update of the state in VB iterations are shown by orange circles. The blue circles and black stars show the true positions and the measurements, respectively. 125
- Figure 5.6 Mode probabilities of CV and CT models for IMM-VB-GG for Case 1. The mode probability of CV model is shown by blue line and the mode probability of CT model is shown by red line. 126
- Figure 5.7 Mode probabilities of CV and CT models for IMM-VB-IW for Case 1. The mode probability of CV model is shown by blue line and the mode probability of CT model is shown by red line. 127
- Figure 5.8 RMSEs of the position for 250 Monte Carlo run for Case 2. RMSEs of the position by IMM-VB-GG is shown by orange line and RMSEs of the position by IMM-VB-IW is shown by green line. 128
- Figure 5.9 RMSEs of the velocity for 250 Monte Carlo run for Case 2. RMSEs of the velocity by IMM-VB-GG is shown by orange line and RMSEs of the velocity by IMM-VB-IW is shown by green line. 129
- Figure 5.10 Measurement noise covariance tracking of the algorithms IMM-VB-GG and IMM-VB-IW for Case 2 (for any time interval of a single MC run). The effective measurement noise covariances are shown by purple ellipses, the estimated measurement noise covariances by IMM-VB-IW are shown by green ellipses and the effective covariance calculated by IMM-VB-GG used in the measurement update of the state in VB iterations are shown by orange circles.. The blue circles and black stars show the true positions and the measurements, respectively. 130

Figure 5.11 Mode probabilities of CV and CT models for IMM-VB-GG for Case 2. The mode probability of CV model is shown by blue line and the mode probability of CT model is shown by red line. 131

Figure 5.12 Mode probabilities of CV and CT models for IMM-VB-IW for Case 2. The mode probability of CV model is shown by blue line and the mode probability of CT model is shown by red line. 132

Figure 5.13 RMSEs of the position for 250 Monte Carlo run for Case 3. RMSEs of the position by IMM-VB-GG is shown by orange line and RMSEs of the position by IMM-VB-IW is shown by green line. 133

Figure 5.14 RMSEs of the velocity for 250 Monte Carlo run for Case 3. RMSEs of the velocity by IMM-VB-GG is shown by orange line and RMSEs of the velocity by IMM-VB-IW is shown by green line. 134

Figure 5.15 Measurement noise covariance tracking of the algorithms IMM-VB-GG and IMM-VB-IW for Case 3 (for any time interval of a single MC run). The effective measurement noise covariances are shown by purple ellipses, the estimated measurement noise covariances by IMM-VB-IW are shown by green ellipses and the effective covariance calculated by VB-GG used in the measurement update of the state in VB iterations are shown by orange circles. The blue circles and black stars show the true positions and the measurements, respectively. 135

Figure 5.16 Mode probabilities of CV and CT models for IMM-VB-GG for Case 3. The mode probability of CV model is shown by blue line and the mode probability of CT model is shown by red line. 136

Figure 5.17 Mode probabilities of CV and CT models for IMM-VB-IW for Case 3. The mode probability of CV model is shown by blue line and the mode probability of CT model is shown by red line. 136

Figure 5.18 RMSEs of the position for 250 Monte Carlo run for Case 4. RMSEs of the position by IMM-VB-GG is shown by orange line and RMSEs of the position by IMM-VB-IW is shown by green line. 137

Figure 5.19 RMSEs of the velocity for 250 Monte Carlo run for Case 4. RMSEs of the velocity by IMM-VB-GG is shown by orange line and RMSEs of the velocity by IMM-VB-IW is shown by green line. 137

Figure 5.20 Measurement noise covariance tracking of the algorithms IMM-VB-GG and IMM-VB-IW for Case 4 (for any time interval of a single MC run). The effective measurement noise covariances are shown by purple ellipses, the estimated measurement noise covariances by IMM-VB-IW are shown by green ellipses and the effective covariance calculated by VB-GG used in the measurement update of the state in VB iterations are shown by orange circles. The blue circles and black stars show the true positions and the measurements, respectively. 138

Figure 5.21 Mode probabilities of CV and CT models for IMM-VB-GG for Case 4. The mode probability of CV model is shown by blue line and the mode probability of CT model is shown by red line. 139

Figure 5.22 Mode probabilities of CV and CT models for IMM-VB-IW for Case 4. The mode probability of CV model is shown by blue line and the mode probability of CT model is shown by red line. 139

Figure 5.23 RMSEs of the position for 250 Monte Carlo run for Case 5. RMSEs of the position by IMM-VB-GG is shown by orange line and RMSEs of the position by IMM-VB-IW is shown by green line. 140

Figure 5.24 RMSEs of the velocity for 250 Monte Carlo run for Case 5. RMSEs of the velocity by IMM-VB-GG is shown by orange line and RMSEs of the velocity by IMM-VB-IW is shown by green line. 140

Figure 5.25 Measurement noise covariance tracking of the algorithms IMM-VB-GG and IMM-VB-IW for Case 5 (for any time interval of a single MC run). The effective measurement noise covariances are shown by purple ellipses, the estimated measurement noise covariances by IMM-VB-IW are shown by green ellipses and the effective covariance calculated by VB-GG used in the measurement update of the state in VB iterations are shown by orange circles. The blue circles and black stars show the true positions and the measurements, respectively. 141

Figure 5.26 Mode probabilities of CV and CT models for IMM-VB-GG for Case 5. The mode probability of CV model is shown by blue line and the mode probability of CT model is shown by red line. 142

Figure 5.27 Mode probabilities of CV and CT models for IMM-VB-IW for Case 5. The mode probability of CV model is shown by blue line and the mode probability of CT model is shown by red line. 142

LIST OF ABBREVIATIONS

ABBREVIATIONS

<i>ARMSE</i>	Averaged Root Mean Square Error
<i>ASRFN</i>	Averaged Square Root of Frobenius Norm
<i>CT</i>	Coordinated Turn
<i>CV</i>	Constant Velocity
<i>IID</i>	Independent and Identically Distributed
<i>IMM</i>	Interacting Multiple Model
<i>IMM – STF</i>	Multiple-Model Extension of Student's-t Filter based on IMM Approach
<i>IMM – VB – GG</i>	Multiple-Model Extension of Variational Bayesian Algorithm using Gamma-Gaussian Prior based on IMM Approach
<i>IMM – VB – IW</i>	Multiple-Model Extension of Variational Bayesian Algorithm using Inverse-Wishart Prior based on IMM Approach
<i>KF</i>	Kalman Filter
<i>KS</i>	Kalman Smoother
<i>MC</i>	Monte Carlo
<i>MSE</i>	Mean Square Error
<i>PDF</i>	Probability Density Function
<i>PF</i>	Particle Filter
<i>RMSE</i>	Root Mean Square Error
<i>SMC</i>	Sequential Monte Carlo
<i>SSM</i>	State-Space Model
<i>STF</i>	Student's-t Filter
<i>TPM</i>	Transition Probability Matrix

VB	Variational Bayesian
$VB - GG$	Variational Bayesian Algorithm using Gamma-Gaussian Prior
$VB - IW$	Variational Bayesian Algorithm using Inverse Wishart Prior

CHAPTER 1

INTRODUCTION

Target tracking, which is the estimation of the present and prediction of future kinematic state of a moving target based on noisy measurements, is a research area that has been developing for many years. It has wide spectrum of applications from military to civilian such as tracking aircrafts, land vehicles, marine or submarine vehicles, controlling air traffic, etc [13]. The first step for carrying out tracking is to model the system by a state-space model (SSM). SSMs include the dynamic equation of the state and the equation that describes the relation between the state and the measurement. In many cases, some of the state variables cannot be measured directly due to noisy measurements. Thus, Bayesian filtering [9] is used to estimate the states given the measurements.

Kalman filter (KF) [24], which provides the best linear unbiased estimate for linear Gaussian state-space models [45], is the closed form solution to the Bayesian filtering equations. It is the optimum filter in terms of mean square error (MSE) [22]. However, in many real world applications, process and measurement noises have outliers so they cannot be modeled as Gaussian distribution and the performance of KF degrades in such cases. The outliers can stem from unreliable sensors, target maneuver, model mismatch, etc. For such systems, it is more suitable to utilize heavy-tailed noise assumption. A large number of studies have been carried out for the filtering of linear systems with heavy-tailed noises [22].

Earlier efforts for robustification of KF were made in [18, 31–34, 50]. In [32], to robustify the KF for linear SSMs that have non-Gaussian noise, which is identified as heavy-tailed or Gaussian contaminated with outliers, an asymptotically efficient stochastic approximation type estimator is proposed. In addition, the robust filter

proposed in [31] is also based on stochastic approximation. In [33], a robust KF that belongs to the class of M-estimator is derived by introducing Huber cost function to the linear SSM and the double exponential and Cauchy densities are used to define heavy-tailed measurement noise. The outlier robust filter given in [50] is also based on the M-estimation methodology and the Student's-t distribution is referred to as heavy-tailed noise. In addition, in [34], a recursion based on KF recursions is derived by using Student's-t mixture model to obtain outlier robust KF.

The algorithms proposed in [34,43,44,50] use the Student's-t distribution to model the noise distributions. Student's-t distribution is similar to a Gaussian distribution that can exhibit heavy-tails. The parameters of Student's-t distribution are the mean, the scale matrix and the degrees of freedom. The degrees of freedom parameter specifies the heavy-tail property of the distribution. In [44], a Student's-t filter is suggested to estimate the states in linear SSMs with heavy-tailed process and measurement noise. It is demonstrated that the suggested filter provides robustness towards outliers. Furthermore, the suggested filter in [44] is also proposed in [43] by explaining the details that are not provided in [44]. In addition to [44], [43] provides a Student's-t based smoothing algorithm for heavy-tailed process and measurement noise.

In recent years, variational Bayesian (VB) inference has become one of the most used inference techniques to estimate the states of linear SSMs with heavy-tailed, non-Gaussian, inaccurate, time varying or unknown noise distributions. In [46], VB approximations were used for inferring the joint posterior distribution and noise covariances for the first time [20]. The joint distribution of the state and the time varying measurement noise covariance is represented as the product of Gaussian and independent Inverse-Gamma distributions. In [4], an adaptive KF for tracking with unknown sensor characteristics is developed by utilizing VB approximation to infer the state and the measurement noise covariance jointly by choosing Inverse-Wishart prior for the unknown covariance of the Gaussian measurement noise. Another adaptive KF is proposed in [23] for linear SSMs with inaccurate and slowly time varying process and measurement noise covariance matrices. VB approximations are used to estimate the state and noise covariances by choosing Inverse-Wishart priors for the one-step predicted state covariance and measurement noise covariance matrices. As in [23], an adaptive KF, which utilizes VB approximation, is proposed in [5] for linear SSMs

with unknown process and measurement noise covariances. However, unlike [23], the process noise covariance is inferred by choosing Inverse-Wishart prior in [5] instead of the one-step predicted state covariance. Additionally, in [30], an adaptive KF is proposed for linear SSMs with unknown process noise covariance. The state and the measurement noise covariance is inferred using VB approximation by choosing Inverse-Wishart prior for the process noise covariance matrix.

VB algorithms are utilized for not only estimating the unknown, inaccurate or time varying noise covariances but also robustification of KF for the linear SSMs with heavy-tailed noise distributions. In [22], an outlier robust Student's-t based KF is developed using the VB algorithm. One-step predicted PDF and the likelihood PDF are approximated as Student's-t but posterior PDF is approximated as a Gaussian. The Student's-t PDFs can be expressed using Gamma-Gaussian approach. In this approach, Student's-t distribution is represented as an infinite mixture of Gaussians by defining auxiliary variable that is Gamma distributed [42]. As in [22], [51] develops a VB algorithm using Gamma-Gaussian approach for linear systems that have Student's-t distributed measurement noise. Furthermore, in [35], robust filtering and smoothing algorithms, which use VB approximations of posterior distribution, are proposed for linear SSMs with heavy-tailed and skewed measurement noise. The heavy-tailed and skewed measurement noise is modeled as a product of independent univariate skew-t distributions. To apply VB approximations, the likelihood PDF is represented as the integration of the product of normal, truncated normal and Gamma distributions.

In [1], an outlier robust filter, which utilizes VB approximation, is derived for linear SSM with heavy-tailed measurement noise. Unlike in [22, 35, 42, 51], the SSM is defined as a linear Gaussian SSM with an unknown measurement noise covariance matrix in [1]. Then, the Inverse-Wishart distribution is selected as a conjugate prior for the covariance matrix of the measurement noise as in adaptive KFs [4, 5, 23]. It is shown in [1] that the VB approximation using Inverse-Wishart approach has the ability to cope with outliers and heavy-tailed noises as well as unknown, inaccurate or time varying noises.

Particle filters (PF) or Sequential Monte Carlo (SMC) [15] methods are generic and

also robust estimation methods that are based on particle representations of probability distributions. They can be applied to any SSM [6]. However, they are not preferred for the purpose of providing robustness towards outliers since they are not feasible in high-dimensional problems because of the curse of dimensionality.

Apart from the outliers in the noises, model uncertainties, which affect the filtering performance, may occur in many real world applications. For example, a maneuvering target may exhibit maneuvers such as turning, acceleration and deceleration. Single linear SSM is not enough to model such behaviors. Thus, these problems require the use of hybrid systems to track the target accurately with changing motion models [17]. Such hybrid systems consist of multiple SSMs and a discrete mode variable that specifies the active model at each time instant. A finite state Markov chain is used to model the change of the discrete mode. In order to deal with such hybrid systems, several multiple-model algorithms such as autonomous multiple model (AMM), generalized pseudo Bayesian of first order (GPB1), generalized pseudo Bayesian of second order (GPB2) and interacting multiple model (IMM) algorithm [41] are developed. In this thesis, IMM framework [38], which is a method for combining multiple filter models to obtain a better estimation accuracy [17], is used for the multiple-model extensions of the filters.

Earlier studies about IMM framework were carried out in [7, 12, 28]. The IMM algorithm for linear Gaussian systems is developed in [12]. However, the conventional IMM approach under the linear Gaussian SSMs assumption is limited for many real world applications because most systems are non-linear or non-Gaussian. In addition, the noise statistics may be unknown or varying with time. In [29], a recursive multiple model approach for linear SSMs with unknown noise statistics is proposed. It is illustrated that the approach in [29] is effective for time varying noises as well as unknown stationary noises. In addition, an adaptive VB approach for multiple model systems with unknown noise statistics is proposed in [27]. However, the proposed methods in [29] and [27] are not robust towards the outliers in noises so they are not applicable for multiple model systems with heavy-tailed noises.

In [47], an outlier robust VB-based IMM algorithm is proposed to cope with multimodality and outliers in the measurement noise. The state and the noise statistics for

each mode of the hybrid system are recursively estimated by using VB algorithm that utilizes Gamma-Gaussian approach. An algorithm which is similar to that in [47] is proposed in [39] for solving the tracking problem of hypersonic vehicles with heavy-tailed measurement noise. As in [47], Gamma and Gaussian distributions are used as the priors of the state and the latent variable in [39]. It is shown that these VB-based IMM algorithms provide robustness towards the outliers in the measurement in hybrid systems. Another VB-based IMM algorithm is presented in [49] by considering the skewness property of measurement noise distribution towards the possibility of asymmetric and heavy-tailed noise characteristics.

The focus of this study is to investigate the outlier robust filters and to derive their multiple-model extensions. At the presence of outliers, the estimation performance of the KF drastically degrades. In the literature, many algorithms have been proposed to provide robustness towards outliers. We investigate only a sub-class of them. Moreover, we implement the multiple-model extensions of these algorithms for dealing with both the multimodality and outliers in the noises.

First, a Student's-t filter (STF) proposed in [44] is investigated and a multiple-model extension of STF (IMM-STF) is derived based on IMM approach. It is demonstrated that IMM-STF provides lower RMSE than the conventional IMM algorithm for multiple-model systems with heavy-tailed process and measurement noise.

Second, we investigate two VB algorithms which utilize Gamma-Gaussian [22] and Inverse-Wishart [23] priors. We refer to VB algorithm which uses Gamma-Gaussian prior as VB-GG [22] and VB algorithm which uses Inverse-Wishart prior as VB-IW [23]. In the literature, while VB-GG approach is generally utilized for linear SSMs with heavy-tailed noises, VB-IW is generally proposed for inferring the state and the unknown/inaccurate noise covariances. An outlier can also be interpreted as an artifact which results from inaccurate noise covariance in the model. Therefore, VB-IW approach can also be used for the purpose of outlier rejection. In this thesis, we implement VB-GG [22] and VB-IW [23] algorithms for a linear SSM with heavy-tailed measurement noise and the performances of the algorithms are compared in the sense of outlier rejection. We show that VB-IW algorithm provides almost the same robustness as VB-GG algorithm towards the outliers in the measurement noise. It can

also outperform VB-GG algorithm under certain conditions. Therefore, VB-IW can be used for the purpose of outlier rejection in addition to the purpose of inferring the unknown noise covariances.

Finally, multiple-model extensions of VB-GG (IMM-VB-GG) and VB-IW (IMM-VB-IW) algorithms are derived based on IMM approach. In the literature, IMM-VB-GG algorithm is proposed for multiple-model systems with heavy-tailed measurement noise. However, the multiple-model extension of VB-IW algorithm based on IMM approach (IMM-VB-IW) has never been derived. In this thesis, we implement IMM-VB-GG and IMM-VB-IW algorithms for a multiple-model system with heavy-tailed measurement noise and the performances of the algorithms are compared.

The rest of the thesis is organized as follows. In Chapter 2, a brief background information about the basic algorithms is given. Bayesian filtering and smoothing, Kalman filter and smoother, interacting multiple model (IMM) algorithm and variational Bayesian (VB) approximation are explained briefly. In addition, simulation examples of KF, KS and IMM algorithm are given. In Chapter 3, firstly, STF [44] is summarized. Then, the multiple-model extension of the Student's-t filter (IMM-STF) is derived. The IMM-STF is compared with conventional IMM algorithm by simulation. In Chapter 4, the derivations of VB-GG and VB-IW algorithms for linear SSMs with heavy-tailed measurement noises are given. Algorithms are tested in a simulation for different noise statistics and the results are compared. The derivations of multiple-model extensions of VB-GG algorithm (IMM-VB-GG) and VB-IW algorithm (IMM-VB-IW) are given in Chapter 5. The methods are tested on simulations of a moving target that has multiple-model SSM and heavy-tailed measurement noise and the obtained results are compared. Finally, the conclusion of this thesis is given in Chapter 6.

CHAPTER 2

BACKGROUND

In this chapter, some background information, which composes the basis of studies carried out in this thesis, is explained briefly.

2.1 Bayesian Filtering and Smoothing

Bayesian theory was first discovered by Thomas Bayes in [9]. However, it did not get much attention until its modern form was discovered by Laplace. Then, Bayesian inference has been applied successfully in many areas such as statistical decision, detection and estimation, pattern recognition, and machine learning [14]. In addition, it was started to be used for filtering purpose, i.e., Bayesian filtering. It is utilized to infer the posterior density of the states given the measurements for general SSMs which can be described as

$$x_k = f(x_{k-1}, w_{k-1}), \quad (2.1a)$$

$$y_k = g(x_k, v_k), \quad (2.1b)$$

where

- $x_k \in R^n$ is the state of the system at time step k ,
- $y_k \in R^m$ is the measurement at time step k ,
- $f(\cdot)$ is a known linear/nonlinear function of the systems dynamics,
- $g(\cdot)$ is a known linear/nonlinear function of the measurements,

- w_{k-1} is the process noise,
- v_k is the measurement noise.

The predicted distribution $p(x_k|y_{1:k-1})$ and the filtering distribution $p(x_k|y_{1:k})$ are computed recursively by the following Bayesian filtering equations that consist of two steps [45]:

- Time Update:

$$p(x_k|y_{1:k-1}) = \int p(x_k|x_{k-1})p(x_{k-1}|y_{1:k-1})dx_{k-1}. \quad (2.2)$$

- Measurement Update:

$$p(x_k|y_{1:k}) = \frac{p(y_k|x_k)p(x_k|y_{1:k-1})}{\int p(y_k|x_k)p(x_k|y_{1:k-1})dx_k}. \quad (2.3)$$

In filtering algorithms, the measurements obtained up to a time step are used to compute the best possible estimate of the state at that time step. However, for some applications, all measurements are used to estimate states at each time step. This process is named as *smoothing*. It is obvious that the smoothing cannot be used in real time applications.

Bayesian smoothing is used to compute the marginal posterior distribution of the state x_k after obtaining measurements up to a time step T , $p(x_k|y_{1:T})$, where $T > k$. For computing smoothed distributions, first of all, the filtering posterior state distributions are computed by filtering equations. After obtaining the filtering distributions, the smoothed distributions are calculated by the following Bayesian smoothing equations [45]:

$$p(x_{k+1}|y_{1:k}) = \int p(x_{k+1}|x_k)p(x_k|y_{1:k})dx_k, \quad (2.4a)$$

$$p(x_k|y_{1:T}) = p(x_k|y_{1:k}) \int \frac{p(x_{k+1}|x_k)p(x_{k+1}|y_{1:T})}{p(x_{k+1}|y_{1:k})} dx_{k+1}. \quad (2.4b)$$

In equation (2.4), $p(x_k|y_{1:k})$ is the filtering distribution at time step k and $p(x_{k+1}|y_{1:k})$ is the predicted distribution at time step $k + 1$. In order to obtain the smoothing distribution, one needs to compute the filtering distributions up to time step T . Starting from $p(x_T|y_{1:T})$, a backward recursion is carried out to obtain smoothed distribu-

tions. Smoothing is not suitable for online tracking applications, it can be used only for offline tracking.

2.2 Kalman Filter and Smoother

Kalman filter, discovered by Rudolf E. Kalman in [24], is an algorithm that uses a series of noisy measurements observed over time and estimates the unknown state vector. It is the closed form solution of the Bayesian filtering equations for linear Gaussian SSMs. The system dynamic and measurement equations are

$$x_k = Ax_{k-1} + w_{k-1}, \quad (2.5a)$$

$$y_k = Cx_k + v_k, \quad (2.5b)$$

where

- $x_k \in R^n$ is the state,
- $y_k \in R^m$ is the measurement,
- $w_{k-1} \sim \mathcal{N}(0, Q)$ is the process noise,
- $v_k \sim \mathcal{N}(0, R)$ is the measurement noise,
- A is $n \times n$ state transition matrix,
- C is $n \times m$ measurement model matrix.

In addition, the prior distribution is assumed as Gaussian $x_0 \sim \mathcal{N}(m_0, P_0)$. It is assumed that all basic random variables $(x_0, w_0, w_1, \dots, v_0, v_1, \dots)$ are independent and identically distributed (IID) and uncorrelated. The model can be expressed as

$$p(x_k|x_{k-1}) = \mathcal{N}(x_k; Ax_{k-1}, Q), \quad (2.6a)$$

$$p(y_k|x_k) = \mathcal{N}(y_k; Cx_k, R). \quad (2.6b)$$

If we consider that the prior distribution is $p(x_{k-1}|y_{1:k-1}) = \mathcal{N}(x_{k-1}; m_{k-1|k-1}, P_{k-1|k-1})$, the predictive and posterior distributions, which are the result of the filtering equations, will be Gaussian as

$$p(x_k|y_{1:k-1}) = \mathcal{N}(x_k; m_{k|k-1}, P_{k|k-1}), \quad (2.7a)$$

$$p(x_k|y_{1:k}) = \mathcal{N}(x_k; m_{k|k}, P_{k|k}). \quad (2.7b)$$

The parameters $m_{k|k-1}$, $P_{k|k-1}$, $m_{k|k}$ and $P_{k|k}$ are computed by the following KF equations:

- Time Update:

$$m_{k|k-1} = Am_{k-1|k-1}, \quad (2.8a)$$

$$P_{k|k-1} = AP_{k-1|k-1}A^T + Q. \quad (2.8b)$$

- Measurement Update:

$$S_k = CP_{k|k-1}C^T + R, \quad (2.9a)$$

$$K_k = P_{k|k-1}C^T S_k^{-1}, \quad (2.9b)$$

$$m_{k|k} = m_{k|k-1} + K_k(y_k - Cm_{k|k-1}), \quad (2.9c)$$

$$P_{k|k} = P_{k|k-1} - K_k S_k K_k^T. \quad (2.9d)$$

The recursion is started from the prior mean m_0 and covariance P_0 . The derivation of these equations are given as follows.

Derivation of the Time Update Equations:

$$\begin{aligned} m_{k|k-1} &= E[x_k|y_{1:k-1}] = E[Ax_{k-1} + w_{k-1}|y_{1:k-1}] \\ &= E[Ax_{k-1}|y_{1:k-1}] + E[w_{k-1}|y_{1:k-1}] \\ &= Am_{k-1|k-1}. \end{aligned} \quad (2.10)$$

$$\begin{aligned}
P_{k|k-1} &= E[(x_k - m_{k|k-1})(x_k - m_{k|k-1})^T | y_{1:k-1}] \\
&= E[(Ax_{k-1} + w_{k-1} - Am_{k-1|k-1})(Ax_{k-1} + w_{k-1} - Am_{k-1|k-1})^T | y_{1:k-1}] \\
&= E[(A(x_{k-1} - m_{k-1|k-1}) + w_{k-1})(A(x_{k-1} - m_{k-1|k-1}) + w_{k-1})^T | y_{1:k-1}] \\
&= E[(A(x_{k-1} - m_{k-1|k-1}) + w_{k-1})((x_{k-1} - m_{k-1|k-1})^T A^T + w_{k-1}^T) | y_{1:k-1}] \\
&= E[A(x_{k-1} - m_{k-1|k-1})(x_{k-1} - m_{k-1|k-1})^T A^T + A(x_{k-1} - m_{k-1|k-1})w_{k-1}^T \\
&\quad + (x_{k-1} - m_{k-1|k-1})^T A^T w_{k-1} + w_{k-1}^2 | y_{1:k-1}] \\
&= E[A(x_{k-1} - m_{k-1|k-1})(x_{k-1} - m_{k-1|k-1})^T A^T | y_{1:k-1}] + E[w_{k-1}^2 | y_{1:k-1}] \\
&= AP_{k-1|k-1}A^T + Q.
\end{aligned} \tag{2.11}$$

At the end of equations (2.10) and (2.11), the time update equations of KF are obtained.

Derivation of the Measurement Update Equations:

Consider the measurement update equation of filtering equations (2.3). According to this equation, it is known that

$$\mathcal{N}(x_k; m_{k|k}, P_{k|k}) \propto \mathcal{N}(y_k; Cx_k, R)\mathcal{N}(x_k; m_{k|k-1}, P_{k|k-1}). \tag{2.12}$$

Let x be a Gaussian random vector $x \sim \mathcal{N}(x; m, P)$. The probability density function of this random vector is as follows,

$$f_x(x) = |2\pi P|^{-1/2} \exp\left(-\frac{1}{2}(x - m)^T P^{-1}(x - m)\right). \tag{2.13}$$

Therefore, by (2.12) and (2.13), the probability density function of x_k given $y_{1:k}$ is

$$\begin{aligned}
&|2\pi P_{k|k}|^{-1/2} \exp\left(-\frac{1}{2}(x_k - m_{k|k})^T P_{k|k}^{-1}(x_k - m_{k|k})\right) \\
&= |2\pi R|^{-1/2} \exp\left(-\frac{1}{2}(y_k - Cx_k)^T R^{-1}(y_k - Cx_k)\right) \\
&\quad + |2\pi P_{k|k-1}|^{-1/2} \exp\left(-\frac{1}{2}(x_k - m_{k|k-1})^T (P_{k|k-1})^{-1}(x_k - m_{k|k-1})\right).
\end{aligned} \tag{2.14}$$

By taking logarithms of both sides, below equation is obtained:

$$\begin{aligned} (x_k - m_{k|k})^T P_{k|k}^{-1} (x_k - m_{k|k}) &= (y_k - Cx_k)^T R^{-1} (y_k - Cx_k) \\ &\quad + (x_k - m_{k|k-1})^T (P_{k|k-1})^{-1} (x_k - m_{k|k-1}). \end{aligned} \quad (2.15)$$

Then, we obtain

$$\begin{aligned} x_k^T P_{k|k}^{-1} x_k - x_k^T P_{k|k}^{-1} m_{k|k} - m_{k|k}^T P_{k|k}^{-1} x_k + m_{k|k}^T P_{k|k}^{-1} m_{k|k} \\ = y_k^T R^{-1} y_k - y_k^T R^{-1} Cx_k - (Cx_k)^T R^{-1} y_k + x_k^T C^T R^{-1} Cx_k \\ + x_k^T (P_{k|k-1})^{-1} x_k - x_k^T (P_{k|k-1})^{-1} m_{k|k-1} - (m_{k|k-1})^T (P_{k|k-1})^{-1} x_k \\ + (m_{k|k-1})^T (P_{k|k-1})^{-1} m_{k|k-1}, \end{aligned} \quad (2.16)$$

$$\begin{aligned} x_k^T P_{k|k}^{-1} x_k - 2x_k^T P_{k|k}^{-1} m_{k|k} + m_{k|k}^T P_{k|k}^{-1} m_{k|k} \\ = y_k^T R^{-1} y_k - 2x_k^T C^T R^{-1} y_k + x_k^T C^T R^{-1} Cx_k \\ + x_k^T (P_{k|k-1})^{-1} x_k - 2x_k^T (P_{k|k-1})^{-1} m_{k|k-1} + (m_{k|k-1})^T (P_{k|k-1})^{-1} m_{k|k-1}. \end{aligned} \quad (2.17)$$

The components which include the term $x_k^T()$ in both sides of equation (2.17) are chosen and below equation is obtained:

$$x_k^T P_{k|k}^{-1} x_k = x_k^T C^T R^{-1} Cx_k + x_k^T (P_{k|k-1})^{-1} x_k. \quad (2.18)$$

According to equation (2.18), we have

$$P_{k|k}^{-1} = C^T R^{-1} C + (P_{k|k-1})^{-1}, \quad (2.19a)$$

$$P_{k|k} = (C^T R^{-1} C + (P_{k|k-1})^{-1})^{-1}. \quad (2.19b)$$

By using matrix inversion lemma [48];

$$\begin{aligned} P_{k|k} &= P_{k|k-1} - P_{k|k-1} C^T (R + CP_{k|k-1} C^T)^{-1} CP_{k|k-1} \\ &= P_{k|k-1} - K_k S_k K_k^T. \end{aligned} \quad (2.20)$$

For deriving $m_{k|k}$, the components which include the term x_k^T of two sides of equation (2.17) are used

$$2x_k^T P_{k|k}^{-1} m_{k|k} = 2x_k^T C^T R^{-1} y_k + 2x_k^T (P_{k|k-1})^{-1} m_{k|k-1}, \quad (2.21a)$$

$$P_{k|k}^{-1} m_{k|k} = C^T R^{-1} y_k + (P_{k|k-1})^{-1} m_{k|k-1}. \quad (2.21b)$$

By multiplying both sides of equation (2.21) with $P_{k|k}$, the following equation is obtained:

$$m_{k|k} = (C^T R^{-1} C + (P_{k|k-1})^{-1})^{-1} (C^T R^{-1} y_k + (P_{k|k-1})^{-1} m_{k|k-1}). \quad (2.22)$$

By matrix inversion lemma [48];

$$\begin{aligned} (C^T R^{-1} C + (P_{k|k-1})^{-1})^{-1} &= P_{k|k-1} - P_{k|k-1} C^T (R + C P_{k|k-1} C^T)^{-1} C P_{k|k-1} \\ &= P_{k|k-1} - P_{k|k-1} C^T (S_k)^{-1} C P_{k|k-1}. \end{aligned} \quad (2.23)$$

Substituting (2.23) in (2.22) results

$$\begin{aligned} m_{k|k} &= (P_{k|k-1} - P_{k|k-1} C^T (S_k)^{-1} C P_{k|k-1}) (C^T R^{-1} y_k + (P_{k|k-1})^{-1} m_{k|k-1}) \\ &= P_{k|k-1} C^T R^{-1} y_k + m_{k|k-1} - P_{k|k-1} C^T (S_k)^{-1} C P_{k|k-1} C^T R^{-1} y_k \\ &\quad - P_{k|k-1} C^T (S_k)^{-1} C m_{k|k-1} \\ &= P_{k|k-1} C^T (R^{-1} - (S_k)^{-1} C P_{k|k-1} C^T R^{-1}) y_k - P_{k|k-1} C^T (S_k)^{-1} C m_{k|k-1} \\ &\quad + m_{k|k-1} \\ &= P_{k|k-1} C^T (S_k)^{-1} y_k - P_{k|k-1} C^T (S_k)^{-1} C m_{k|k-1} + m_{k|k-1} \\ &= m_{k|k-1} + P_{k|k-1} C^T (S_k)^{-1} (y_k - C m_{k|k-1}) \\ &= m_{k|k-1} + K_k (y_k - C m_{k|k-1}). \end{aligned} \quad (2.24)$$

Hence, KF equations are obtained.

Kalman smoother (Rauch-Tung-Striebel Smoother) is used to calculate the distribution $p(x_k | y_{1:T}) = \mathcal{N}(x_k; m_{k|T}, P_{k|T})$ where $T > k$ for linear Gaussian SSMs (2.5).

First, mean and covariance values of filtering and predicted distributions for all time steps up to time step T are computed by KF equations (2.8) and (2.9). Mean and covariance values of smoothed distributions, which are the state distributions conditional on the whole measurement data, are calculated by the following equations for all time steps:

$$G_k = P_{k|k} A^T (P_{k+1|k})^{-1}, \quad (2.25a)$$

$$m_{k|T} = m_{k|k} + G_k (m_{k+1|T} - m_{k+1|k}), \quad (2.25b)$$

$$P_{k|T} = P_{k|k} + G_k (P_{k+1|T} - P_{k+1|k}) G_k^T, \quad (2.25c)$$

where $m_{k|k}$ and $P_{k|k}$ are the mean and covariance values of filtering distribution at the time step k and $m_{k+1|k}$ and $P_{k+1|k}$ are the mean and covariance values of the predicted distribution at the time step $k + 1$ computed by KF equations (2.8) and (2.9). The recursion is started from the time step T , with $m_{T|T}$ and $P_{T|T}$.

Before starting the derivation of KS equations, the following lemmas should be given [45].

Lemma 1 (*Joint Distribution of Gaussian Variables*) *If random variables $x \in R^n$ and $y \in R^m$ have the Gaussian probability distributions*

$$x \sim \mathcal{N}(m, P), \quad (2.26a)$$

$$y|x \sim \mathcal{N}(Cx, R). \quad (2.26b)$$

Then, the joint distribution of x , y and marginal distribution of y are given as

$$\begin{bmatrix} x \\ y \end{bmatrix} \sim \mathcal{N} \left(\begin{bmatrix} m \\ Cm \end{bmatrix}, \begin{bmatrix} P & PC^T \\ CP & CPC^T + R \end{bmatrix} \right), \quad (2.27a)$$

$$y \sim \mathcal{N}(Cm, CPC^T + R). \quad (2.27b)$$

Lemma 2 (*Conditional Distribution of Gaussian Variables*) *If the random variables*

x and y have the joint Gaussian probability distribution

$$\begin{bmatrix} x \\ y \end{bmatrix} \sim \mathcal{N} \left(\begin{bmatrix} a \\ b \end{bmatrix}, \begin{bmatrix} A & C \\ C^T & B \end{bmatrix} \right), \quad (2.28)$$

the marginal and conditional distributions of x and y are given as follows,

$$x \sim \mathcal{N}(a, A), \quad (2.29a)$$

$$y \sim \mathcal{N}(b, B), \quad (2.29b)$$

$$x|y \sim \mathcal{N}(a + CB^{-1}(y - b), A - CB^{-1}C^T), \quad (2.29c)$$

$$y|x \sim \mathcal{N}(b + C^T A^{-1}(x - a), B - C^T A^{-1}C). \quad (2.29d)$$

Derivation of Kalman Smoother Equations:

By Lemma 1, the joint distribution of x_k and x_{k+1} given $y_{1:k}$ is

$$\begin{aligned} p(x_k, x_{k+1}|y_{1:k}) &= p(x_{k+1}|x_k)p(x_k|y_{1:k}) \\ &= \mathcal{N}(x_{k+1}; Ax_k, Q)\mathcal{N}(x_k; m_{k|k}, P_{k|k}) \\ &= \mathcal{N} \left(\begin{bmatrix} x_k \\ x_{k+1} \end{bmatrix}; m', P' \right), \end{aligned} \quad (2.30)$$

where

$$m' = \begin{bmatrix} m_{k|k} \\ Am_{k|k} \end{bmatrix}, \quad P' = \begin{bmatrix} P_{k|k} & P_{k|k}A^T \\ AP_{k|k} & AP_{k|k}A^T + Q \end{bmatrix}. \quad (2.31)$$

Due to the Markov property, one can write

$$p(x_k|x_{k+1}, y_{1:T}) = p(x_k|x_{k+1}, y_{1:k}). \quad (2.32)$$

Let

$$p(x_k|x_{k+1}, y_{1:T}) = \mathcal{N}(x_k; m'', P''). \quad (2.33)$$

By Lemma (2), m'' and P'' are computed as follows,

$$G_k = P_{k|k}A^T(AP_{k|k}A^T + Q)^{-1}, \quad (2.34a)$$

$$m'' = m_{k|k} + G_k(x_{k+1} - Am_{k|k}), \quad (2.34b)$$

$$P'' = P_{k|k} - G_k(AP_{k|k}A^T + Q)G_k^T. \quad (2.34c)$$

The joint distribution of x_k and x_{k+1} given all the data is

$$\begin{aligned} p(x_{k+1}, x_k | y_{1:T}) &= p(x_k | x_{k+1}, y_{1:T})p(x_{k+1} | y_{1:T}) \\ &= \mathcal{N}(x_k; m'', P'')\mathcal{N}(x_{k+1}; m_{k+1|T}, P_{k+1|T}) \\ &= \mathcal{N}\left(\begin{bmatrix} x_{k+1} \\ x_k \end{bmatrix}; m''', P'''\right), \end{aligned} \quad (2.35)$$

where

$$\begin{aligned} m''' &= \begin{bmatrix} m_{k+1|T} \\ m_{k|k} + G_k(m_{k+1|T} - Am_{k|k}) \end{bmatrix}, \\ P''' &= \begin{bmatrix} P_{k+1|T} & P_{k+1|T}G_k^T \\ G_kP_{k+1|T} & G_kP_{k+1|T}G_k^T + P'' \end{bmatrix}. \end{aligned} \quad (2.36)$$

The marginal distribution of x_k given all measurement data up to time step T can be written as

$$p(x_k | y_{1:T}) = \mathcal{N}(x_k; m_{k|T}, P_{k|T}), \quad (2.37)$$

where the mean and covariance are given by,

$$m_{k|T} = m_{k|k} + G_k(m_{k+1|T} - Am_{k|k}), \quad (2.38a)$$

$$P_{k|T} = P_{k|k} + G_k(P_{k+1|T} - AP_{k|k}A^T - Q)G_k^T. \quad (2.38b)$$

2.2.1 Simulation Example: Implementing KF and KS for 2-D Target Tracking Problem

In this simulation, KF and KS are implemented for tracking a moving target in 2-D space. The target moves according to a constant velocity (CV) model [8] and the position of the target is measured. The motion of the target is modeled by a linear Gaussian SSM (2.5) where $w_{k-1} \sim \mathcal{N}(0, Q)$ and $v_k \sim \mathcal{N}(0, R)$. The state transition matrix and observation matrix are given by

$$A = \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}, \quad (2.39)$$

where $T = 1$. The process and measurement noise parameters are given by

$$Q = \begin{bmatrix} \frac{T^3}{2} & \frac{T^2}{2} & 0 & 0 \\ \frac{3T^2}{2} & T & 0 & 0 \\ 0 & 0 & \frac{T^3}{2} & \frac{T^2}{2} \\ 0 & 0 & \frac{3T^2}{2} & T \end{bmatrix} \times q, \quad q = 1, \quad R = r \times I_2, \quad r = 10. \quad (2.40)$$

The initial state is chosen as $x_0 = [0 \ 10 \ 0 \ 10]^T$ and the simulation is carried out for 20 time steps. Figure 2.1 illustrates the true trajectory of the target, the estimated trajectories of the target by KF and KS and the measurements. The position errors that are calculated for each time step are shown in Figure 2.2. The error for the time step k is calculated by

$$Error = \sqrt{(p_k^x - \hat{p}_k^x)^2 + (p_k^y - \hat{p}_k^y)^2}, \quad (2.41)$$

where (p_k^x, p_k^y) and $(\hat{p}_k^x, \hat{p}_k^y)$ are the true and estimated positions of the target at time step k . As it can be observed from the figures, KS estimates are closer to the true position of the target than KF estimates.

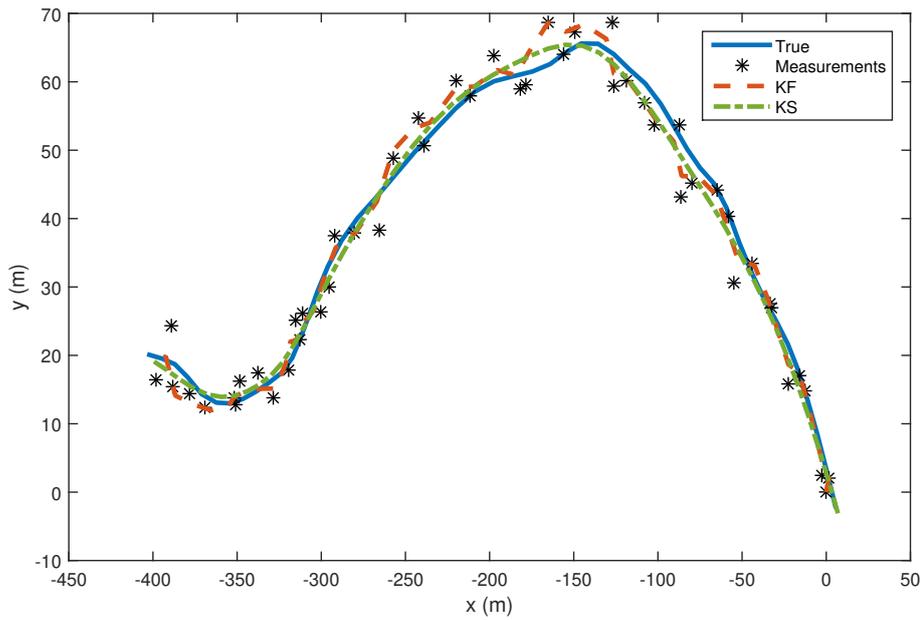


Figure 2.1: True and estimated trajectories of the target. True trajectory of the target is shown by blue line, the estimated trajectory by KF is shown by orange dashed line and the estimated trajectory by KS is shown by green dash-dot line. The measurements are shown by black stars.

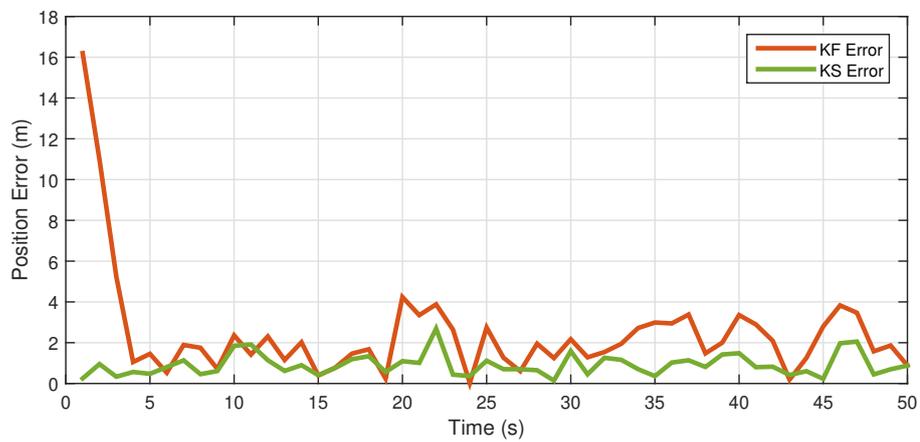


Figure 2.2: Position errors of KF and KS. The error of KF is shown by orange line and the error of KS is shown by green line.

2.3 Interacting Multiple Model (IMM) Algorithm

The interacting multiple model (IMM) algorithm is a method for combining multiple filter models to obtain better state estimate of the target with changing motion behaviors [17]. The filter models that are used in IMM algorithm should be selected to cover several target behaviors.

We consider the hybrid system that has the following linear Gaussian SSM,

$$x_k = A^{r_k} x_{k-1} + w_{k-1}^{r_k}, \quad (2.42a)$$

$$y_k = C^{r_k} x_k + v_k^{r_k}, \quad (2.42b)$$

where

- $x_k \in R^n$ is the state,
- $r_k \in \{1, 2, \dots, N\}$ is the discrete mode variable,
- $y_k \in R^m$ is the measurement,
- $w_{k-1}^{r_k} \sim \mathcal{N}(0, Q^{r_k})$ is the process noise,
- $v_k^{r_k} \sim \mathcal{N}(0, R^{r_k})$ is the measurement noise,
- A^{r_k} is the state transition matrix for the mode r_k ,
- C^{r_k} is the measurement model matrix for the mode r_k .

The discrete mode variable r_k is modeled as a finite Markov chain with transition probability matrix (TPM) $\Pi = [\pi_{ji} \triangleq P(r_k = i | r_{k-1} = j)]$ and the hybrid system consists of N SSMs. The aim is to infer or approximate the posterior distribution $p(x_k | y_{1:k})$. The optimal posterior distribution is a mixture of Gaussians with an exponentially growing number of components. Therefore, we need approximations and the following approximation is made by IMM filter:

$$p(x_k | y_{1:k}) \approx \sum_i^N \mu_k^i \mathcal{N}(x_k; m_{k|k}^i, P_{k|k}^i), \quad (2.43)$$

where

$$\mu_k^i \triangleq P(r_k = i | y_{1:k}), \quad (2.44)$$

are the posterior mode probabilities. The overall posterior mean and covariance are calculated as

$$m_{k|k} = \sum_{i=1}^N \mu_k^i m_{k|k}^i, \quad (2.45a)$$

$$P_{k|k} = \sum_{i=1}^N \mu_k^i [P_{k|k}^i + (m_{k|k}^i - m_{k|k})(m_{k|k}^i - m_{k|k})^T]. \quad (2.45b)$$

The means, covariances and posterior mode probabilities $\{m_{k|k}^i, P_{k|k}^i, \mu_k^i\}_{i=1}^N$ are calculated recursively using $\{m_{k-1|k-1}^j, P_{k-1|k-1}^j, \mu_{k-1}^j\}_{j=1}^N$ statistics from the previous step. We know from the previous step that

$$p(x_{k-1} | y_{1:k-1}, r_{k-1} = j) \triangleq \mathcal{N}(x_{k-1}; m_{k-1|k-1}^j, P_{k-1|k-1}^j). \quad (2.46)$$

Using equation (2.46), the following approximation is obtained:

$$\begin{aligned} p(x_{k-1} | y_{1:k-1}, r_k = i) &\triangleq \sum_{j=1}^N \mu_{k-1|k-1}^{ji} p(x_{k-1} | y_{1:k-1}, r_{k-1} = j), \\ &= \sum_{j=1}^N \mu_{k-1|k-1}^{ji} \mathcal{N}(x_{k-1}; m_{k-1|k-1}^j, P_{k-1|k-1}^j), \\ &\approx \mathcal{N}(x_{k-1}; m_{k-1|k-1}^{0i}, P_{k-1|k-1}^{0i}), \end{aligned} \quad (2.47)$$

where the merged mean $m_{k-1|k-1}^{0i}$ and covariance $P_{k-1|k-1}^{0i}$ are calculated by moment matching as

$$m_{k-1|k-1}^{0i} = \sum_{j=1}^N \mu_{k-1|k-1}^{ji} m_{k-1|k-1}^j, \quad (2.48)$$

$$\begin{aligned}
P_{k-1|k-1}^{0i} &= \sum_{j=1}^N \mu_{k-1|k-1}^{ji} \\
&\times \left[P_{k-1|k-1}^j + (m_{k-1|k-1}^j - m_{k-1|k-1}^{0i})(m_{k-1|k-1}^j - m_{k-1|k-1}^{0i})^T \right].
\end{aligned} \tag{2.49}$$

This merging step is named as "mixing" in the literature. $\mu_{k-1|k-1}^{ji}$ are mixing mode probabilities calculated as

$$\mu_{k-1|k-1}^{ji} = \frac{\pi_{ji} \mu_{k-1}^j}{\sum_{l=1}^N \pi_{li} \mu_{k-1}^l}. \tag{2.50}$$

Then, mean and covariance values of filtered distributions for each mode are calculated by using KF time and measurement update equations (2.8) and (2.9) as in below,

- Time Update:

$$m_{k|k-1}^i = A^i m_{k-1|k-1}^{0i}, \tag{2.51a}$$

$$P_{k|k-1}^i = A^i P_{k-1|k-1}^{0i} (A^i)^T + Q^i. \tag{2.51b}$$

- Measurement Update:

$$S_k^i = C^i P_{k|k-1}^i (C^i)^T + R^i, \tag{2.52a}$$

$$K_k^i = P_{k|k-1}^i (C^i)^T (S_k^i)^{-1}, \tag{2.52b}$$

$$m_{k|k}^i = m_{k|k-1}^i + K_k^i (y_k - C^i m_{k|k-1}^i), \tag{2.52c}$$

$$P_{k|k}^i = P_{k|k-1}^i - K_k^i S_k^i (K_k^i)^T. \tag{2.52d}$$

In addition, updated mode probabilities are computed according to the following equation,

$$\mu_k^i = \frac{\mathcal{N}(y_k; C^i m_{k|k-1}^i, S_k^i) \sum_{j=1}^N \pi_{ji} \mu_{k-1}^j}{\sum_{l=1}^N \mathcal{N}(y_k; C^l m_{k|k-1}^l, S_k^l) \sum_{j=1}^N \pi_{jl} \mu_{k-1}^j}. \tag{2.53}$$

After all these steps, we have means and covariances for all modes and the mode probabilities. At the end of the algorithm, the overall mean and covariance are calcu-

lated by equation (2.42). The block diagram of a single step of IMM algorithm for N-models is shown in Figure 2.3.

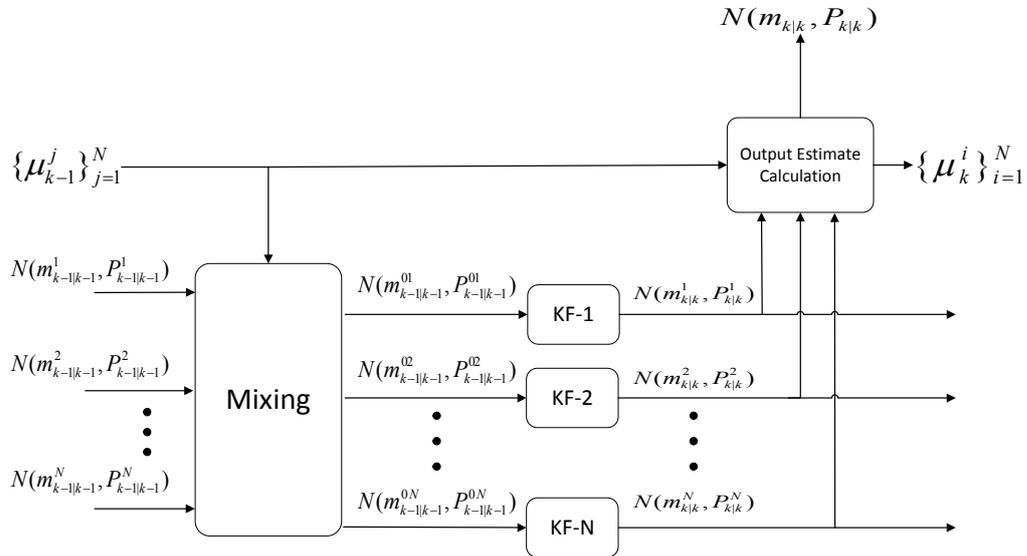


Figure 2.3: The block diagram of a single step of IMM algorithm for N-models.

2.3.1 Simulation Example: Implementing KF and IMM Algorithm for Tracking Problem of a Maneuvering Target in 2-D Space

In this simulation, tracking of a maneuvering target in 2-D space is considered. The target moves according to the constant velocity (CV) model [8] and the coordinated turn (CT) model with known turn rate [40]. The motion of the target is modeled by two linear Gaussian SSMs that are given below.

- While the target is moving according to CV model,

$$\begin{aligned} x_k &= A^{CV} x_{k-1} + w_{k-1}, \\ y_k &= C x_k + v_k. \end{aligned} \quad (2.54)$$

- While the target is moving according to CT model,

$$\begin{aligned} x_k &= A^{CT} x_{k-1} + w_{k-1}, \\ y_k &= C x_k + v_k, \end{aligned} \quad (2.55)$$

where $w_{k-1} \sim \mathcal{N}(0, Q)$ and $v_k \sim \mathcal{N}(0, R)$. The state transition matrices and the observation matrix are

$$A^{CV} = \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix}, A^{CT} = \begin{bmatrix} 1 & \frac{\sin(\omega T)}{\omega} & 0 & -\frac{1 - \cos(\omega T)}{\omega} \\ 0 & \cos(\omega T) & 0 & -\sin(\omega T) \\ 0 & \frac{1 - \cos(\omega T)}{\omega} & 1 & \frac{\sin(\omega T)}{\omega} \\ 0 & \sin(\omega T) & 0 & \cos(\omega T) \end{bmatrix}, \quad (2.56)$$

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}, \quad (2.57)$$

where $T = 1$ s and $\omega = \pi/15$ rad/s. The process and measurement noises are

generated according to the parameters

$$Q = \begin{bmatrix} \frac{T^3}{3} & \frac{T^2}{2} & 0 & 0 \\ \frac{T^2}{2} & T & 0 & 0 \\ 0 & 0 & \frac{T^3}{3} & \frac{T^2}{2} \\ 0 & 0 & \frac{T^2}{2} & T \end{bmatrix} \times q, \quad q = 10, \quad R = r \times I_2, \quad r = 100. \quad (2.58)$$

The transition probability matrix (TPM) is taken as

$$\Pi = \begin{bmatrix} 0.95 & 0.05 \\ 0.05 & 0.95 \end{bmatrix}. \quad (2.59)$$

The motion of the target is simulated according to

- the CV model (2.54) from $k = 1$ to $k = 30$,
- the CT model (2.55) from $k = 31$ to $k = 60$,
- the CV model (2.54) from $k = 61$ to $k = 90$.

In this simulation, the position and the velocity of the target are estimated by both IMM algorithm (with two KF models) and a single KF (with CV model). The performance of these filters are compared in terms of position and velocity errors. True trajectory of the target, measurements and the estimated trajectories by IMM algorithm and a single KF are given in Figure 2.4. In Figure 2.5 and Figure 2.6, the position errors and the velocity errors of IMM algorithm and a single KF are shown, respectively. The errors are computed by equation (2.41). As seen in these figures, the error of KF is greater than the error of IMM algorithm between time steps $k = 31$ and $k = 60$ seconds when the target makes circular motion, i.e., the target moves according to CT model.

Despite the fact that the performance of the single KF degrades when the motion behavior of the target changes, the performance of IMM algorithm does not change by switching the other mode. Figure 2.7 illustrates the mode probabilities of CV and CT models. In Figure 2.7, the mode probability of CT model increases and the

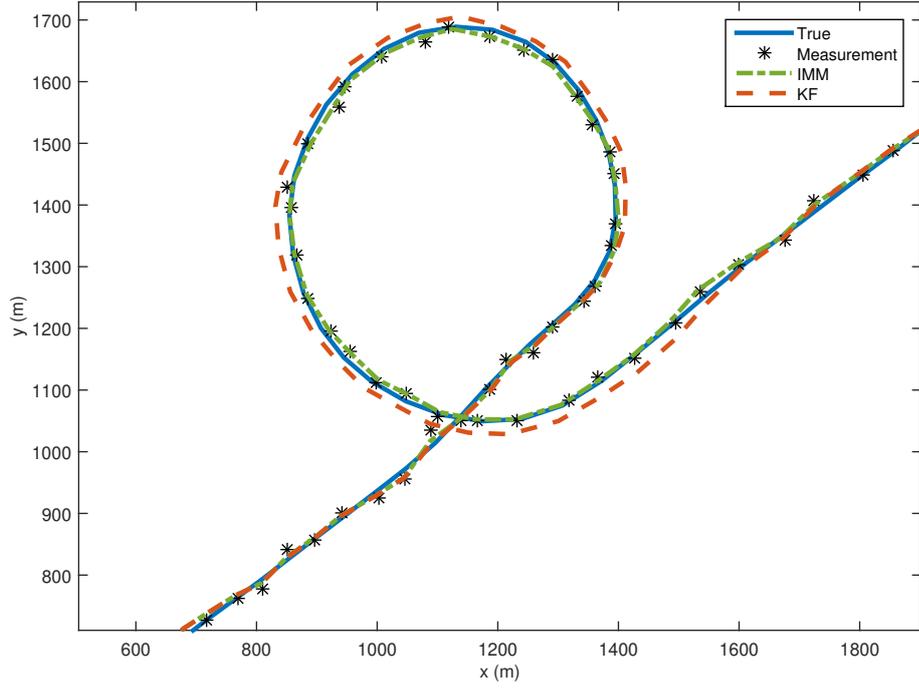


Figure 2.4: True and estimated trajectories of the target. True trajectory of the target is shown by blue line, the estimated trajectory by IMM algorithm is shown by green dash-dot line and the estimated trajectory by the single KF is shown by orange dashed line. The measurements are shown by black stars.

mode probability of CV model decreases between time steps $k = 31$ and $k = 60$ seconds which means that the algorithm switches CT model in that time interval. This switching behavior provides the better estimation accuracy to IMM algorithm.

Also, the averaged position errors are calculated as 18 m and 8 m by the single KF and IMM algorithm, respectively. In addition, the averaged velocity errors are calculated as 16 m/s (by the single KF) and 5.5 m/s (by IMM algorithm). The averaged error is computed by

$$\text{Averaged Error} = \frac{1}{K} \sum_{k=1}^K \sqrt{(p_k^x - \hat{p}_k^x)^2 + (p_k^y - \hat{p}_k^y)^2}, \quad (2.60)$$

where (p_k^x, p_k^y) and $(\hat{p}_k^x, \hat{p}_k^y)$ are respectively true and estimated positions or velocities and K is the total time step. By examining all these results, it is observed that the

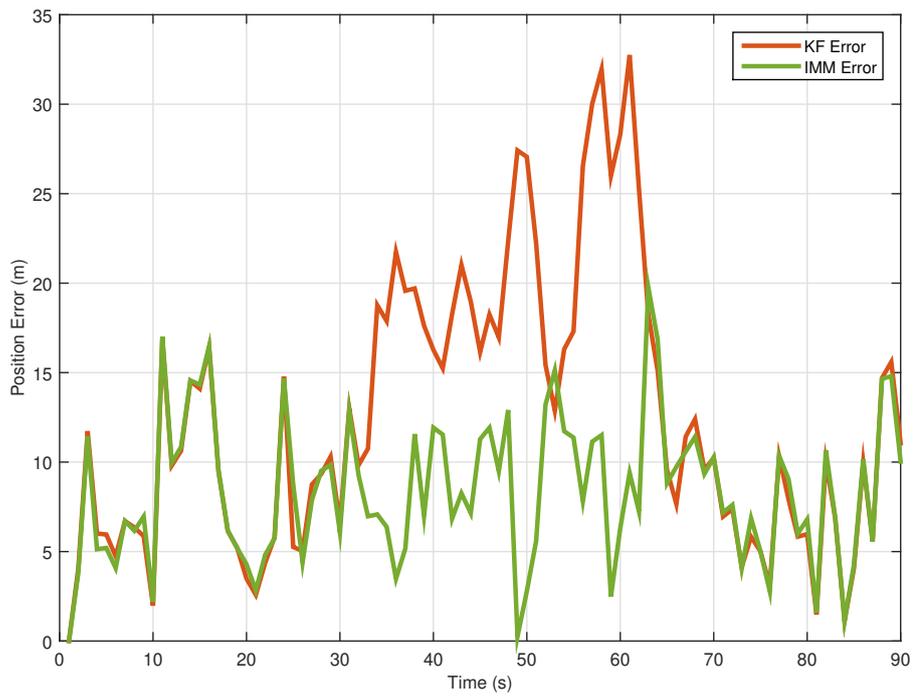


Figure 2.5: Position errors of IMM algorithm and a single KF (with CV model) for all time steps. The error of IMM algorithm is shown by green line and the error of KF is shown by orange line.

filter performance degrades if single filter model is used for tracking a target with changing motion behavior.

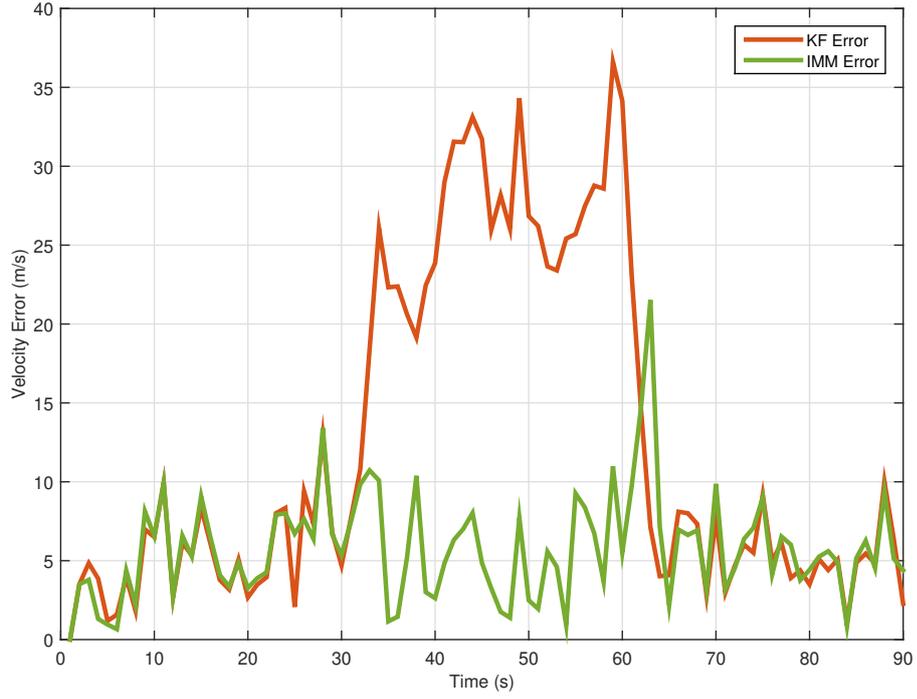


Figure 2.6: Velocity errors of IMM algorithm and a single KF (with CV model) for all time steps. The error of IMM algorithm is shown by green line and the error of KF is shown by orange line.

2.4 Variational Bayesian Methods

Variational Bayesian (VB) methods are the methods that approximate the posterior distribution over a set of variables given the data, i.e., $p(\Theta|Y)$, as $q(\Theta)$;

$$p(\Theta|Y) \approx q(\Theta), \quad (2.61)$$

where $\Theta = \{\theta_1, \theta_2, \dots, \theta_i, \dots, \theta_N\}$. We assume that $q(\Theta)$ is factorized [11] as

$$q(\Theta) = q(\theta_1, \theta_2, \dots, \theta_i, \dots, \theta_N) = \prod_{i=1}^N q(\theta_i). \quad (2.62)$$

Among all PDFs $\{q(\theta_i)\}_{i=1}^N$, an optimal one can be calculated by minimizing the Kullback-Leibler divergence between the approximate posterior PDF $q(\Theta)$ and true

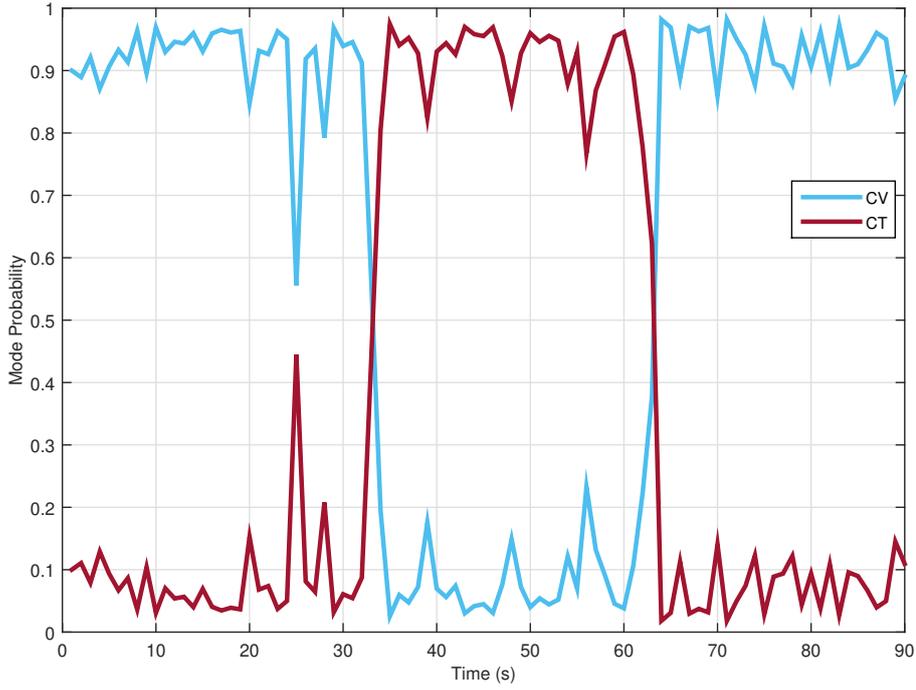


Figure 2.7: Mode Probabilities of CV and CT models. The mode probability of CV model is shown by blue line and the mode probability of CT model is shown by red line.

posterior PDF $p(\Theta|Y)$ [22] as

$$\{q^*(\theta_i)\}_{i=1}^N = \operatorname{argmin} KLD(q(\Theta)||p(\Theta|Y)), \quad (2.63)$$

where

$$KLD(q(\Theta)||p(\Theta|Y)) = \int q(\Theta) \log \frac{q(\Theta)}{p(\Theta|Y)} d\Theta. \quad (2.64)$$

Minimizing the the Kullback-Leibler divergence between $q(\Theta)$ and $p(\Theta|Y)$ corresponds to maximizing the lower bound $\mathcal{L}(q(\Theta))$ [11] that is given as

$$\mathcal{L}(q(\Theta)) = \int q(\Theta) \log \frac{p(\Theta, Y)}{q(\Theta)} d\Theta. \quad (2.65)$$

The optimal solution of equation (2.63) is obtained by the following equation [21],

$$\log q^*(\theta_i) = \mathbf{E}_{\Theta^{-\theta_i}}[\log p(\Theta, Y)] + c_{\theta_i}, \quad (2.66)$$

where $\Theta^{-\theta_i}$ is the set of all elements of Θ except for θ_i , $\mathbf{E}[\cdot]$ is the expectation operation and c_{θ_i} is the constant. For solving equation (2.66), fixed-point iterations should be carried out by updating only one parameter in Θ while keeping other parameters fixed [22].

CHAPTER 3

MULTIPLE-MODEL EXTENSION OF STUDENT'S T FILTER (IMM-STF)

Kalman filter (KF) is one of the most commonly used algorithms for state estimation in linear systems. This is because of its ease of application and optimality in terms of mean square error [3]. However, many real world applications have outliers in process and measurement noises that cannot be modeled by Gaussians. Therefore, the performance of KF degrades for such systems. In this situation, it is more suitable to define heavy-tailed measurement and process noises. Student's-t distribution exhibits such heavy-tail property. A Student's-t distributed random vector $x \in R^d$ is described by a mean vector $m \in R^d$, a scale matrix Σ and the degrees of freedom parameter ν as $St(x; m, \Sigma, \nu)$ where d is the dimension of the random vector. The scale matrix is the scaled version of the covariance matrix. The PDF of t-distributed random vector x is expressed as [26]

$$f_x(x) = \frac{\Gamma\left(\frac{\nu+d}{2}\right)}{\Gamma\left(\frac{\nu}{2}\right)} \frac{1}{(\nu\pi)^{\frac{d}{2}}} \frac{1}{\sqrt{\det(\Sigma)}} \left(1 + \frac{1}{\nu}(x-m)^T \Sigma^{-1}(x-m)\right)^{-\frac{d+\nu}{2}}. \quad (3.1)$$

The covariance P of this t random variable is calculated as

$$P = \frac{\nu}{\nu-2} \Sigma, \quad (3.2)$$

for $\nu > 2$ [26]. The Student's-t distribution converges to Gaussian distribution as ν goes to infinity [44]. Figure 3.1 shows the PDFs of Student's t distributions with different degrees of freedom values and a Gaussian distribution. As seen in the figure, the Student's-t distributions have heavier tails than the Gaussian distribution.

Several methods have been developed to deal with the tracking problem of the systems

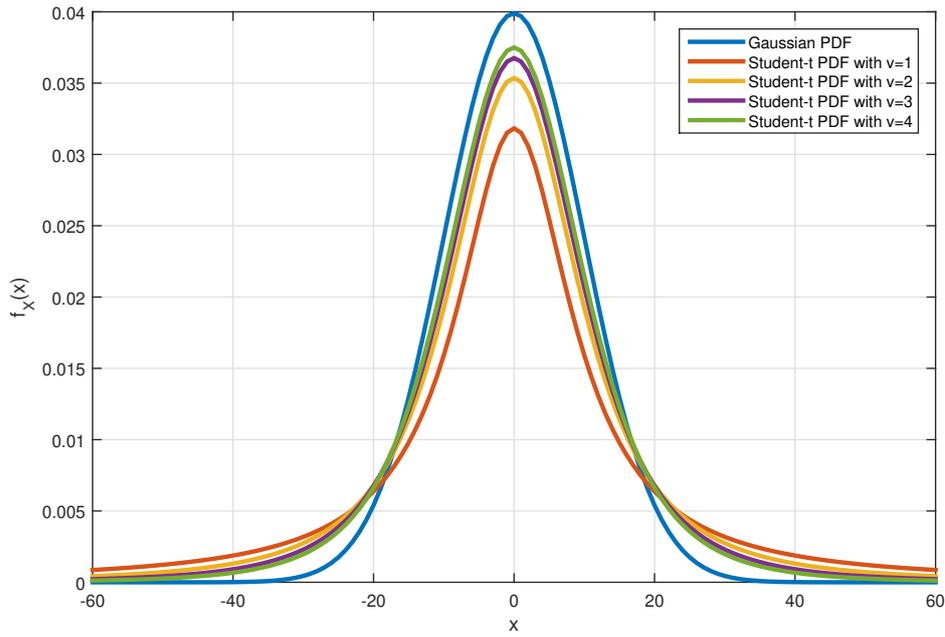


Figure 3.1: PDFs of Student’s-t distributions with different degrees of freedom values and a Gaussian distribution.

that have outliers in process and measurement noises since KF cannot be used in such cases. In [44], a Student’s-t filter (STF) is proposed for linear SSMs that have heavy-tailed process and measurement noises. It is illustrated that the proposed filter provides better estimation accuracy than KF in the presence of outliers in noises [44]. In this chapter, a multiple-model extension of STF [44] based on IMM approach (IMM-STF) is derived to overcome the degrading effects of outliers and to track the changing dynamics of the target. The rest of the chapter is organized as follows: First, the filter derived in [44] is explained briefly. Then, the derivation of IMM-STF is explained in detail. Finally, the performance of IMM-STF and IMM algorithm for a maneuvering target, which have outliers in process and measurement, is tested in simulations and the results are compared.

3.1 Student's t Filter

In this section, the Student's t filter derived in [44] is explained briefly. We consider the linear SSM

$$x_k = Ax_{k-1} + w_{k-1}, \quad (3.3a)$$

$$y_k = Cx_k + v_k, \quad (3.3b)$$

where

- $x_k \in R^n$ is the state at the time step k ,
- $y_k \in R^m$ is the measurement at the time step k ,
- $w_{k-1} \sim St(0, Q, \gamma)$ is the process noise,
- $v_k \sim St(0, R, \delta)$ is the measurement noise,
- The initial state is assumed as $x_0 \sim St(m_0, \Sigma_0, \eta_0)$.

Under these assumptions, the predicted PDF $p(x_k|y_{1:k-1})$ and the filtered PDF $p(x_k|y_{1:k})$ are t-distributed [44]. The algorithm consists of two steps which are time update and measurement update as in KF.

- **Time Update:**

Assume we have prior distribution $p(x_{k-1}|y_{1:k-1}) \sim St(m_{k-1|k-1}, \Sigma_{k-1|k-1}, \eta_{k-1})$ and the state and the process noise are jointly t-distributed, i.e., $\eta_{k-1} = \gamma$. The joint distribution of the state and the process noise is

$$p(x_{k-1}, w_{k-1}|y_{1:k-1}) = St \left(\begin{bmatrix} x_{k-1} \\ w_{k-1} \end{bmatrix}; \begin{bmatrix} m_{k-1|k-1} \\ 0 \end{bmatrix}, \begin{bmatrix} \Sigma_{k-1|k-1} & 0 \\ 0 & Q \end{bmatrix}, \eta_{k-1} \right). \quad (3.4)$$

The mean and the scale matrix of predicted PDF $p(x_k|y_{1:k-1})$ are computed by

$$m_{k|k-1} = Am_{k-1|k-1}, \quad (3.5a)$$

$$\Sigma_{k|k-1} = A\Sigma_{k-1|k-1}A^T + Q. \quad (3.5b)$$

The degrees of freedom parameter η_{k-1} is retained.

- **Measurement Update:**

Assume that the predicted state and the measurement noise are jointly t-distributed, i.e., $\eta_{k-1} = \delta$, and the joint PDF is

$$p(x_k, v_k | y_{1:k-1}) = St \left(\begin{bmatrix} x_k \\ v_k \end{bmatrix}; \begin{bmatrix} m_{k|k-1} \\ 0 \end{bmatrix}, \begin{bmatrix} \Sigma_{k|k-1} & 0 \\ 0 & R \end{bmatrix}, \eta_{k-1} \right). \quad (3.6)$$

Then, the joint density of the state and the measurement noise is obtained by a linear transformation as

$$p(x_k, y_k | y_{1:k-1}) = St \left(\begin{bmatrix} x_k \\ y_k \end{bmatrix}; \begin{bmatrix} m_{k|k-1} \\ Cm_{k|k-1} \end{bmatrix}, \begin{bmatrix} \Sigma_{k|k-1} & \Sigma_{k|k-1}C^T \\ C\Sigma_{k|k-1} & S_k \end{bmatrix}, \eta_{k-1} \right), \quad (3.7)$$

where

$$S_k = C\Sigma_{k|k-1}C^T + R. \quad (3.8)$$

The parameters of filtered PDF are calculated by the equations given below:

$$m_{k|k} = m_{k|k-1} + \Sigma_{k|k-1}C^T(S_k)^{-1}(y_k - Cm_{k|k-1}), \quad (3.9a)$$

$$\Sigma_{k|k} = \frac{\eta_{k-1} + \Delta^2}{\eta_{k-1} + m} (\Sigma_{k|k-1} - \Sigma_{k|k-1}C^T(S_k)^{-1}C\Sigma_{k|k-1}), \quad (3.9b)$$

$$\eta_k = \eta_{k-1} + m, \quad (3.9c)$$

where

$$\Delta^2 = (y_k - Cm_{k|k-1})^T(S_k)^{-1}(y_k - Cm_{k|k-1}). \quad (3.10)$$

Since the degrees of freedom parameter η_k is increased in each measurement update step, the distributions converges to Gaussians and the heavy tail properties are lost

after few steps later [43]. Therefore, the filter converges to KF. In order to avoid this drawback, in [43], it is proposed η_k to be decreased to the previous value η_{k-1} with updating the scale matrix of the filtered PDF by moment matching or the minimization of Kullback-Leibler divergence. Before reducing η_k , according to moment matching, the scale matrix is updated as $\Sigma'_{k|k}$ by

$$\Sigma'_{k|k} = \Sigma_{k|k-1} \left(\frac{\eta_k}{\eta_k - 2} \right) \left(\frac{\eta_{k-1} - 2}{\eta_{k-1}} \right). \quad (3.11)$$

Then, η_k is reduced. For the next time update step, $\Sigma'_{k|k}$ becomes the scale matrix of the prior distribution.

The assumed noise conditions at the time update and the measurement update steps are met with low possibility in real world applications [44]. In [44], a method is suggested for this situation. In the time update step, if $\eta_{k-1} \neq \gamma$, the common degrees of freedom parameter is selected as $\eta'_{k-1} = \min(\eta_{k-1}, \gamma)$ to preserve the heaviest tail. Then, the scale matrices of the prior state and the process noise are updated by the minimization of Kullback-Leibler divergence. Time update equations are carried out by the updated parameters. In the measurement update step, again the common degrees of freedom parameter is selected as $\eta''_{k-1} = \min(\eta'_{k-1}, \delta)$. Then, the scale matrices of the predicted state and the measurement noise are updated by the minimization of Kullback-Leibler divergence. Hence, the measurement update equations are carried out by the updated parameters.

3.1.1 Simulation Example: Comparison of STF and KF

In this simulation, the performances of STF and KF are compared on a target tracking example for three different cases. We assume that the target moves according to a CV model [8] so consider the linear SSM (3.3) with the state-space matrices

$$A = \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}, \quad (3.12)$$

where $T = 0.5$ s. The nominal process and measurement noise parameters are

$$Q = \begin{bmatrix} \frac{T^3}{2} & \frac{T^2}{2} & 0 & 0 \\ \frac{3}{2} & T & 0 & 0 \\ 0 & 0 & \frac{T^3}{2} & \frac{T^2}{2} \\ 0 & 0 & \frac{3}{2} & T \end{bmatrix} \times q, \quad q = 1, \quad R = r \times I_2, \quad r = 100. \quad (3.13)$$

The simulation is run for 1000 time steps and 500 Monte Carlo (MC) run. We consider three different cases for noise generation. These cases are given below.

Case A:

The process and measurement noises are generated according to

$$w_k \sim \begin{cases} \mathcal{N}(0, Q) & \text{w.p. } 0.95 \\ \mathcal{N}(0, 1000Q) & \text{w.p. } 0.05 \end{cases} \quad (3.14)$$

$$v_k \sim \begin{cases} \mathcal{N}(0, R) & \text{w.p. } 0.9 \\ \mathcal{N}(0, 100R) & \text{w.p. } 0.1 \end{cases} \quad (3.15)$$

Case B:

The process noise is generated as $w_k \sim \mathcal{N}(0, Q)$ and the measurement noise is generated according to (3.15), i.e., only measurement noise have outliers.

Case C:

The process noise is generated as $w_k \sim \mathcal{N}(0, Q)$ and the measurement noise is generated as $v_k \sim \mathcal{N}(0, R)$, i.e., both process and measurement noises don't have outliers.

The position and the velocity of the target are estimated by both KF and STF [44]. KF is not aware of the true noise covariances that's why it is implemented by the nominal noise covariance matrices Q and R . STF is also aware of the nominal noise covariances but it assumes that the noises are t-distributed with degrees of freedom parameter 3. Thence, the scale matrices of the noises are adjusted as $0.92 \times Q$ and

$0.92 \times R$ to preserve the 80% probability region. Root mean square error (RMSE) and averaged root mean square error (ARMSE) [22] given in equations (3.16) and (3.17) are used as performance metrics.

$$RMSE = \sqrt{\frac{1}{M} \sum_{mc=1}^M ((x_k^{mc} - \hat{x}_k^{mc})^2 + (y_k^{mc} - \hat{y}_k^{mc})^2)}, \quad (3.16)$$

$$ARMSE = \sqrt{\frac{1}{MK} \sum_{k=1}^K \sum_{mc=1}^M ((x_k^{mc} - \hat{x}_k^{mc})^2 + (y_k^{mc} - \hat{y}_k^{mc})^2)}, \quad (3.17)$$

where (x_k^{mc}, y_k^{mc}) and $(\hat{x}_k^{mc}, \hat{y}_k^{mc})$ are the true and estimated positions or velocities at mc th MC run and K is the simulation time.

Figure 3.2 illustrates the position errors of STF and KF for a specific time interval of single run for Case 1. The errors are calculated by equation (2.41).

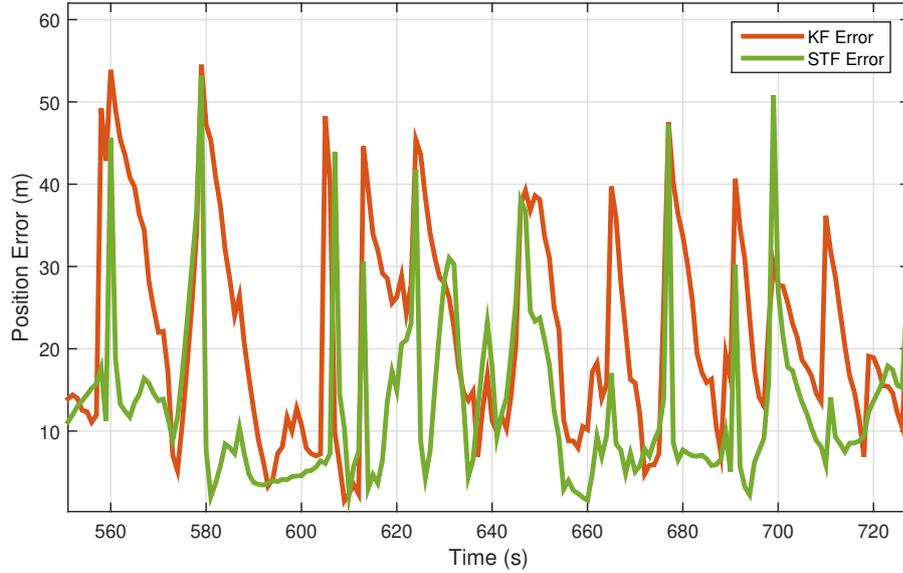


Figure 3.2: Position errors by STF and KF for Case A. Only a specific time interval is shown. The error of KF is shown by orange line and the error of STF is shown by green line.

The spikes, seen in Figure 3.2, occur when the target is exposed to the measurement outliers or it maneuvers. It can be seen in Figure 3.2 that once the target is exposed

to an outlier, KF error decreases in a longer time than STF error. Figures 3.3 and 3.4 show the RMSEs of position and velocity for Case A. The ARMSEs calculated by equation (3.17) for Case A are given in Table 3.1.

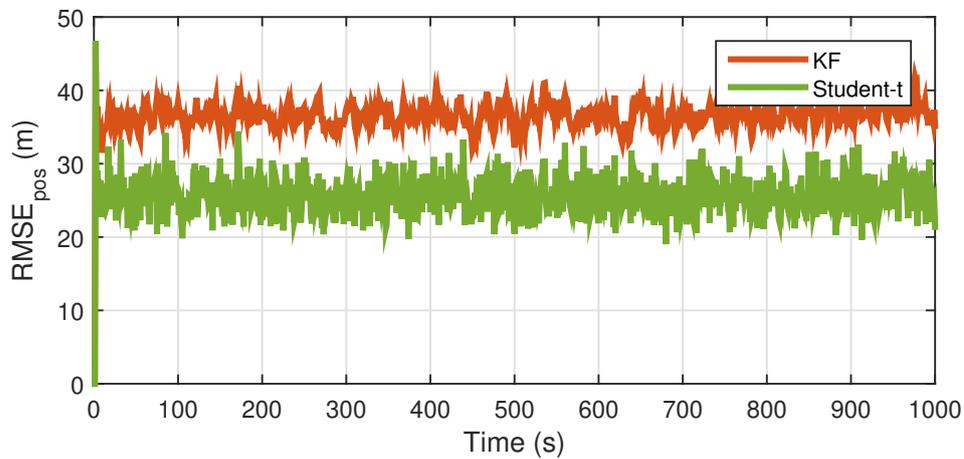


Figure 3.3: RMSEs of the position for 500 Monte Carlo run for Case A. RMSEs of the position by KF is shown by orange line and RMSEs of the position by STF is shown by green line.

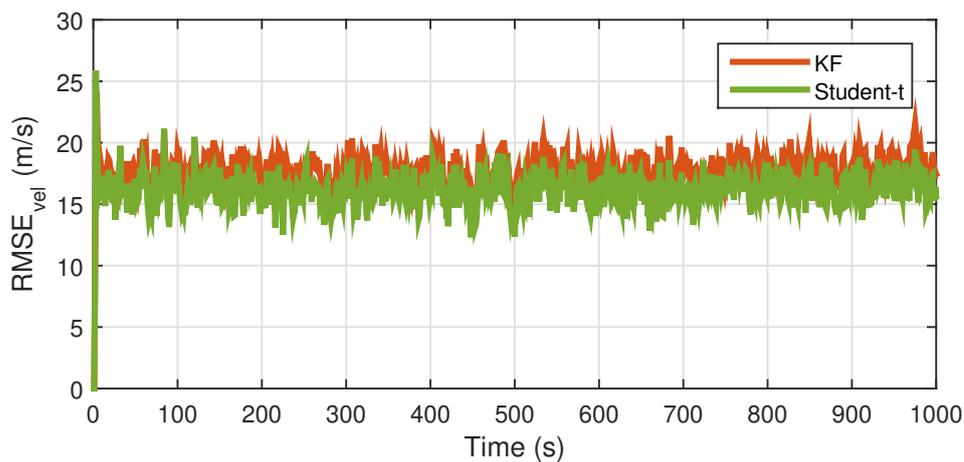


Figure 3.4: RMSEs of the velocity for 500 Monte Carlo run for Case A. RMSEs of the velocity by KF is shown by orange line and RMSEs of the velocity by STF is shown by green line.

Table 3.1: ARMSEs of KF and STF for Case A

	ARMSE of the position (m)	ARMSE of the velocity (m/s)
KF	36.60	17.80
STF	25.80	16.19

Figure 3.5 and shows the RMSEs of position for KF and STFs with different degrees of freedom parameters for 1000 MC run for Case A. The ARMSEs of KF and STFs with different degrees of freedom parameters for Case A are given in Table 3.2. The ARMSE increases as the degrees of freedom parameter increases according to Table 3.2.

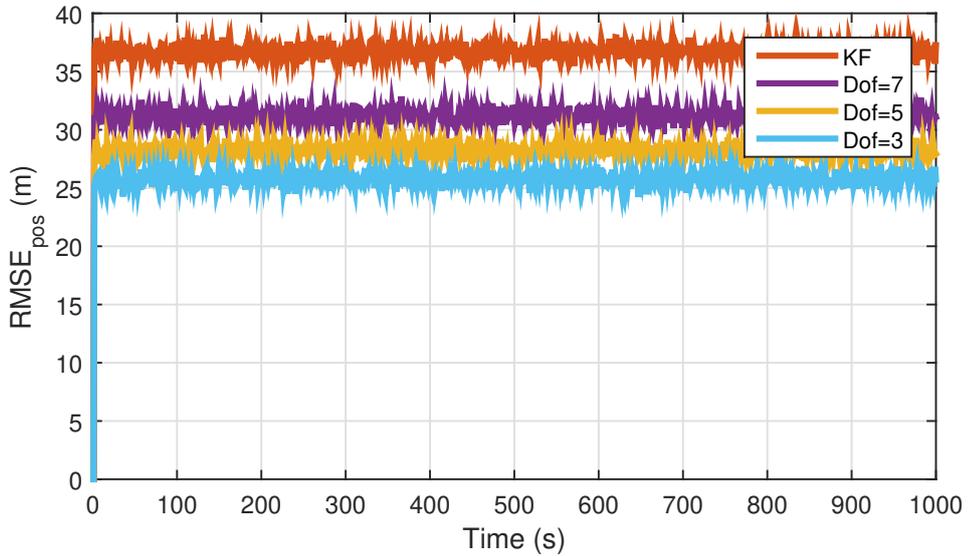


Figure 3.5: RMSEs of the position for KF and STFs with different degrees of freedom parameters for 500 Monte Carlo run for Case A.

Table 3.2: ARMSEs of KF and STFs with different degrees of freedom parameters for Case A

	ARMSE of the position (m)	ARMSE of the velocity (m/s)
KF	36.74	17.82
STF with dof 3	25.83	16.19
STF with dof 5	28.21	16.66
STF with dof 7	32.18	17.08

Figures 3.6 and 3.7 show the RMSEs of position and velocity for Case B. The ARMSEs calculated by equation (3.17) for Case B are given in Table 3.3.

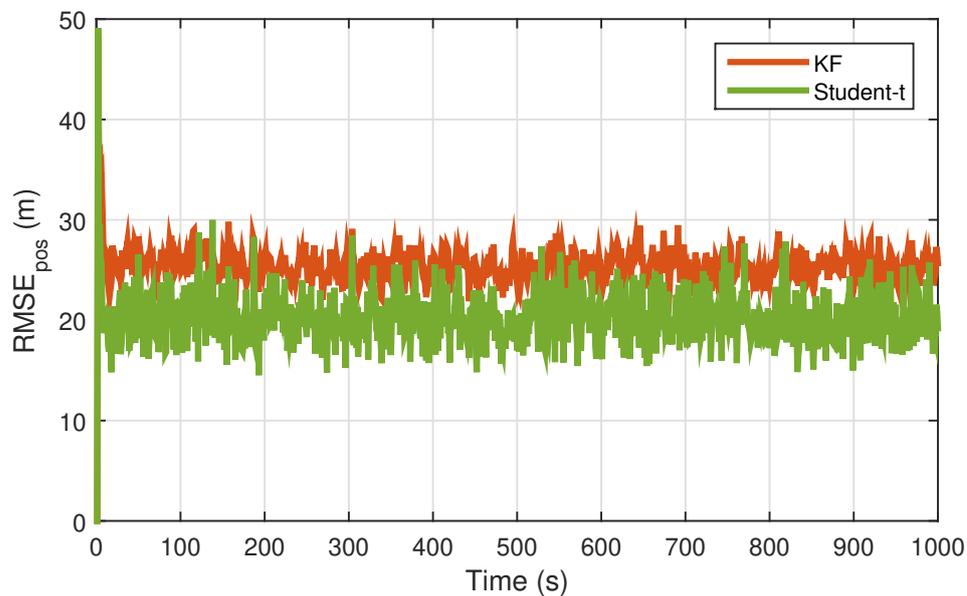


Figure 3.6: RMSEs of the position for 500 Monte Carlo run for Case B. RMSEs of the position by KF is shown by orange line and RMSEs of the position by STF is shown by green line.

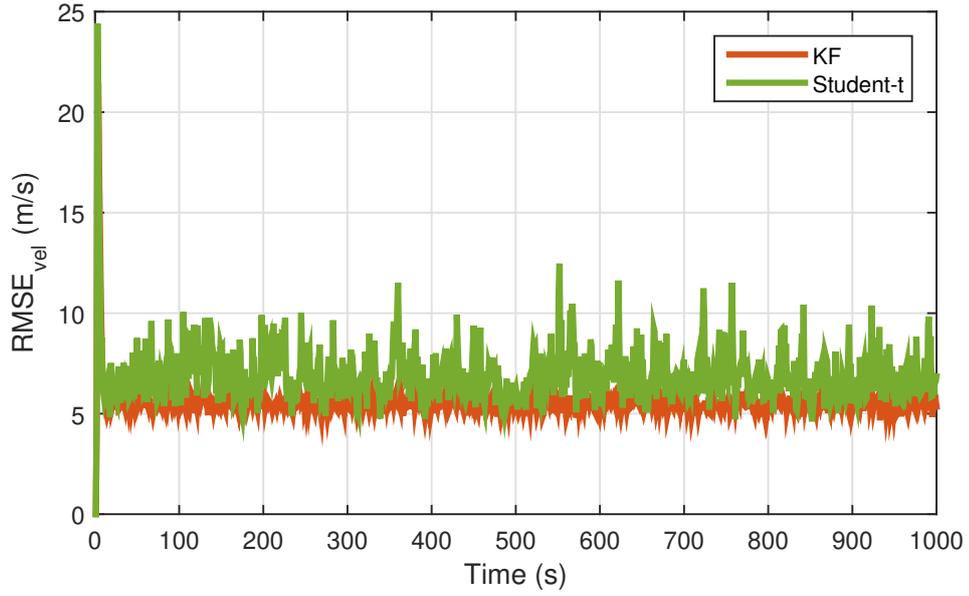


Figure 3.7: RMSEs of the velocity for 500 Monte Carlo run for Case B. RMSEs of the velocity by KF is shown by orange line and RMSEs of the velocity by STF is shown by green line.

Table 3.3: ARMSEs of KF and STF for Case B

	ARMSE of the position (m)	ARMSE of the velocity (m/s)
KF	25.44	5.60
STF	20.37	7.02

Figures 3.8 and 3.9 show the RMSEs of position and velocity for Case C. The ARMSEs calculated by equation (3.17) for Case C are given in Table 3.4.

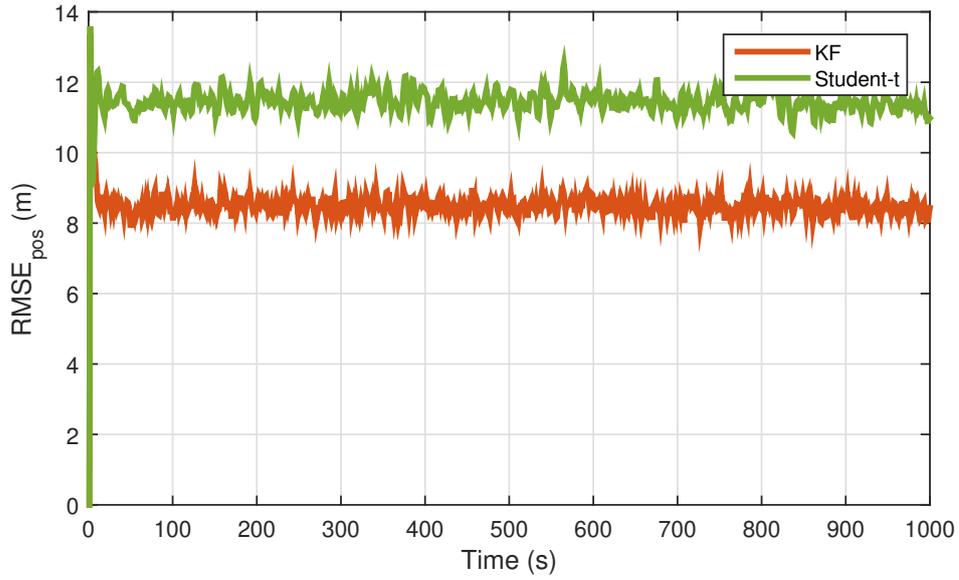


Figure 3.8: RMSEs of the position for 500 Monte Carlo run for Case C. RMSEs of the position by KF is shown by orange line and RMSEs of the position by STF is shown by green line.

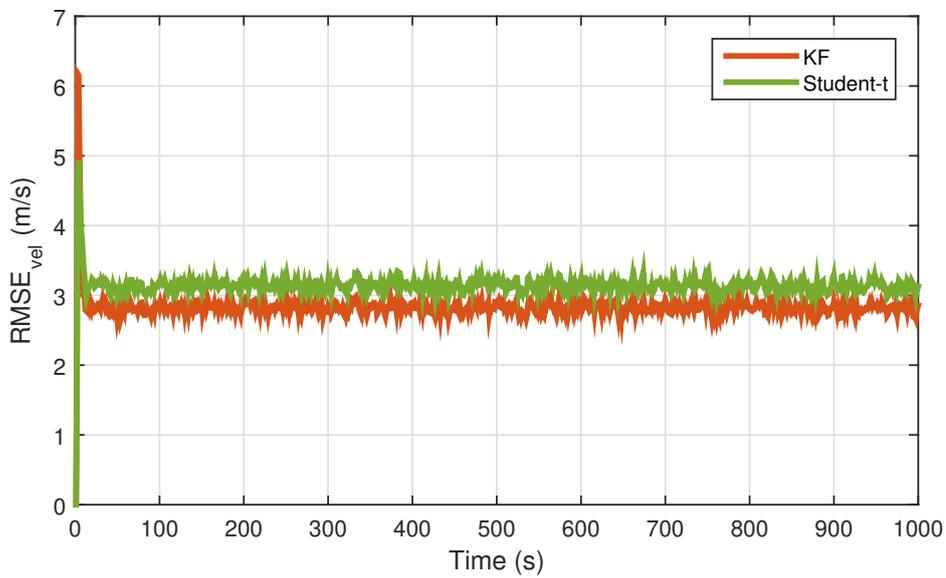


Figure 3.9: RMSEs of the velocity for 500 Monte Carlo run for Case C. RMSEs of the velocity by KF is shown by orange line and RMSEs of the velocity by STF is shown by green line.

Table 3.4: ARMSEs of KF and STF for Case C

	ARMSE of the position (m)	ARMSE of the velocity (m/s)
KF	8.50	2.84
STF	11.43	3.14

These results show that STF proposed in [44] provides better estimation accuracy than KF for the systems that have outliers in process and measurement noise. Also, STF provides lower RMSEs than KF for the system that have outliers in only measurement noise. Therefore, STF provides robustness towards outliers. However, if both process and measurement noises don't have outliers, KF provides better position and velocity estimates than STF since KF is the optimum Bayesian filter in terms of mean square error for linear Gaussian SSMs.

3.2 Derivation of IMM-STF

The conventional IMM algorithm is composed of KF models so that the performance of IMM algorithm degrades in the presence of outliers in process and measurement noise. Therefore, in this section, IMM-STF, which is a multiple-model extension of Student's-t filter based on IMM approach, is derived to overcome the outliers and the changing dynamics of the target. We consider the hybrid system that have the system dynamic and measurement equations (2.42) where

- $w_{k-1}^{r_k}$ is distributed with $St(0, Q^{r_k}, \gamma^{r_k})$,
- $v_k^{r_k}$ is distributed with $St(0, R^{r_k}, \delta^{r_k})$.

The mode state r_k is modeled as a Markov chain that have the transition probability matrix (TPM) $\Pi = [\pi_{ji} \triangleq P(r_k = i | r_{k-1} = j)]$. In addition, we assume that all degrees of freedom values are greater than 2. The block diagram of a single step of IMM-STF is shown in Figure 3.10.

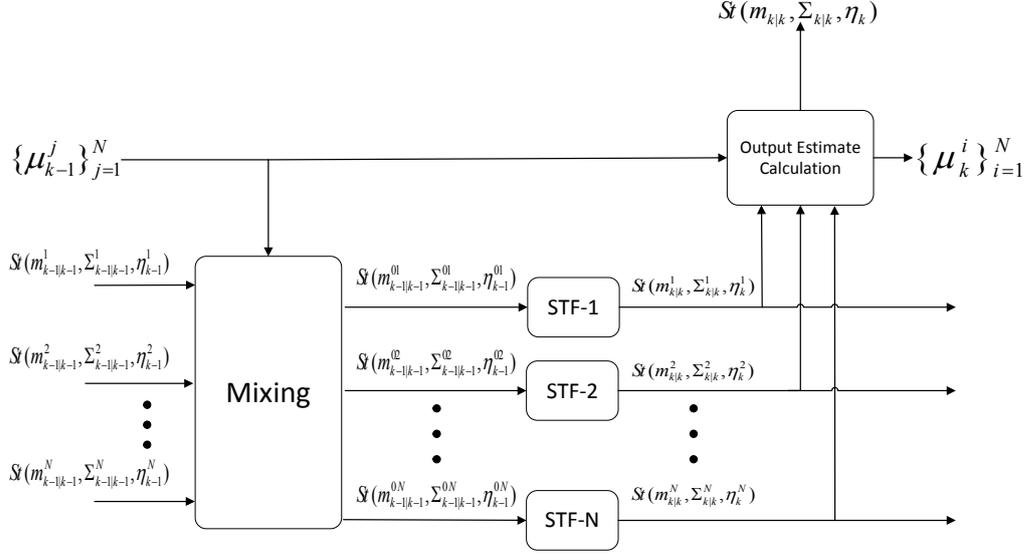


Figure 3.10: The block diagram of a single step of IMM-STF for N-models.

We assume that we have statistics $\{m_{k-1|k-1}^j, \Sigma_{k-1|k-1}^j, \eta_{k-1}^j\}_{j=1}^N$ and mode probabilities $\{\mu_{k-1}^j\}_{j=1}^N$ from the previous step. The recursive equations for statistics of the posterior t densities $\{m_{k|k}^i, \Sigma_{k|k}^i, \eta_k^i\}_{i=1}^N$ and posterior mode probabilities $\{\mu_k^i\}_{i=1}^N$ are derived in this section. The posterior PDF can be expressed as

$$\begin{aligned}
 P(x_k|y_{1:k}) &= \sum_{i=1}^N p(x_k|y_{1:k}, r_k = i) P(r_k = i|y_{1:k}) \\
 &= \sum_{i=1}^N \mu_k^i p(x_k|y_{1:k}, r_k = i)
 \end{aligned} \tag{3.18}$$

$$= \sum_{i=1}^N \mu_k^i \frac{p(y_k|x_k, r_k = i)}{p(y_k|y_{1:k-1}, r_k = i)} p(x_k|y_{1:k-1}, r_k = i). \tag{3.19}$$

Using Bayesian filtering time update equation (2.2);

$$P(x_k|y_{1:k}) = \sum_{i=1}^N \mu_k^i \frac{p(y_k|x_k, r_k = i)}{p(y_k|y_{1:k-1}, r_k = i)} \times \int p(x_k|x_{k-1}, r_k = i) p(x_{k-1}|y_{1:k-1}, r_k = i) dx_{k-1} \quad (3.20)$$

$$= \sum_{i=1}^N \mu_k^i \frac{p(y_k|x_k, r_k = i)}{p(y_k|y_{1:k-1}, r_k = i)} \int p(x_k|x_{k-1}, r_k = i) \times \sum_{j=1}^N p(x_{k-1}|y_{1:k-1}, r_{k-1} = j) P(r_{k-1} = j|r_k = i, y_{1:k-1}) dx_{k-1} \quad (3.21)$$

$$= \sum_{i=1}^N \mu_k^i \frac{p(y_k|x_k, r_k = i)}{p(y_k|y_{1:k-1}, r_k = i)} \int p(x_k|x_{k-1}, r_k = i) \times \sum_{j=1}^N \mu_{k-1|k-1}^{j i} p(x_{k-1}|y_{1:k-1}, r_{k-1} = j) dx_{k-1}. \quad (3.22)$$

From the previous step, we have

$$p(x_{k-1}|y_{1:k-1}, r_{k-1} = j) = St(x_{k-1}; m_{k-1|k-1}^j, \Sigma_{k-1|k-1}^j, \eta_{k-1}^j). \quad (3.23)$$

According to equations (3.20), (3.22), (3.23), we can make the following approximation,

$$p(x_{k-1}|y_{1:k-1}, r_k = i) = \sum_{j=1}^N \mu_{k-1|k-1}^{j i} p(x_{k-1}|y_{1:k-1}, r_{k-1} = j) \quad (3.24)$$

$$= \sum_{j=1}^N \mu_{k-1|k-1}^{j i} St(x_{k-1}; m_{k-1|k-1}^j, \Sigma_{k-1|k-1}^j, \eta_{k-1}^j), \quad (3.25)$$

$$p(x_{k-1}|y_{1:k-1}, r_k = i) \approx St(x_{k-1}; m_{k-1|k-1}^{0i}, \Sigma_{k-1|k-1}^{0i}, \eta_{k-1}^{0i}). \quad (3.26)$$

It is assumed that the degrees of freedom values $\{\eta_{k-1}^j\}_{j=1}^N$ are equal for all j , so that $\eta_{k-1}^j = \eta_{k-1}^{0i} = \eta_{k-1}$. The merged mean $m_{k-1|k-1}^{0i}$ and scale matrix $\Sigma_{k-1|k-1}^{0i}$ are

calculated by moment matching as follows,

$$\begin{aligned}
m_{k-1|k-1}^{0i} &= \sum_{j=1}^N \mu_{k-1|k-1}^{ji} m_{k-1|k-1}^j, \\
\Sigma_{k-1|k-1}^{0i} &= \frac{\eta_{k-1} - 2}{\eta_{k-1}} \sum_{j=1}^N \mu_{k-1|k-1}^{ji} \\
&\times \left[\frac{\eta_{k-1}}{\eta_{k-1} - 2} \Sigma_{k-1|k-1}^j + (m_{k-1|k-1}^j - m_{k-1|k-1}^{0i})(m_{k-1|k-1}^j - m_{k-1|k-1}^{0i})^T \right].
\end{aligned} \tag{3.27}$$

The mixing probabilities are computed as

$$\begin{aligned}
\mu_{k-1|k-1}^{ji} &\triangleq P(r_{k-1} = j | r_k = i, y_{1:k-1}) \\
&\propto P(r_k = i | r_{k-1} = j, y_{1:k-1}) P(r_{k-1} = j | y_{1:k-1}) \\
&\propto P(r_k = i | r_{k-1} = j, y_{1:k-1}) \mu_{k-1}^j \\
&\propto \pi_{ji} \mu_{k-1}^j.
\end{aligned} \tag{3.28}$$

Therefore,

$$\mu_{k-1|k-1}^{ji} = \frac{\pi_{ji} \mu_{k-1}^j}{\sum_{l=1}^N \pi_{li} \mu_{k-1}^l}. \tag{3.29}$$

Equations (3.27) and (3.29) are the mixing step of the algorithm. Now, by using equations (3.22) and (3.26), the following equation is obtained:

$$\begin{aligned}
p(x_k | y_{1:k}) &= \sum_{i=1}^N \mu_k^i \frac{p(y_k | x_k, r_k = i)}{p(y_k | y_{1:k-1}, r_k = i)} \\
&\times \int p(x_k | x_{k-1}, r_k = i) St(x_{k-1}; m_{k-1|k-1}^{0i}, \Sigma_{k-1|k-1}^{0i}, \eta_{k-1}) dx_{k-1}.
\end{aligned} \tag{3.30}$$

Assume that the process noise of the i^{th} filter and the state are jointly t-distributed.

Under this assumption, the following approximation can be done:

$$\begin{aligned} \int p(x_k|x_{k-1}, r_k = i) St(x_{k-1}; m_{k-1|k-1}^{0i}, \Sigma_{k-1|k-1}^{0i}, \eta_{k-1}) dx_{k-1} \\ \approx St(x_k; m_{k|k-1}^i, \Sigma_{k|k-1}^i, \eta_{k-1}), \end{aligned} \quad (3.31)$$

where

$$m_{k|k-1}^i = A^i m_{k-1|k-1}^{0i}, \quad (3.32a)$$

$$\Sigma_{k|k-1}^i = A^i \Sigma_{k-1|k-1}^{0i} (A^i)^T + Q^i, \quad (3.32b)$$

assuming $\eta_{k-1} = \gamma^i$ for all i . η_{k-1} is retained. Now, if equation (3.31) is substituted into equation (3.30), the following expression is obtained:

$$p(x_k|y_{1:k}) = \sum_{i=1}^N \mu_k^i \frac{p(y_k|x_k, r_k = i)}{p(y_k|y_{1:k-1}, r_k = i)} St(x_k; m_{k|k-1}^i, \Sigma_{k|k-1}^i, \eta_{k-1}). \quad (3.33)$$

By performing mode-matched measurement updates the following approximation can be obtained:

$$p(x_k|y_{1:k}) = \sum_{i=1}^N \mu_k^i St(x_k; m_{k|k}^i, \Sigma_{k|k}^i, \eta_k^i). \quad (3.34)$$

The mean and scale matrix for the i^{th} filter are calculated as in STF [44]. Assume $\eta_{k-1} = \delta^i$ for all i ,

$$m_{k|k}^i = m_{k|k-1}^i + \Sigma_{k|k-1}^i (C^i)^T (S_k^i)^{-1} (y_k - \hat{y}_{k|k-1}^i), \quad (3.35a)$$

$$\Sigma_{k|k}^i = \frac{\eta_{k-1} + \Delta_i^2}{\eta_{k-1} + m} (\Sigma_{k|k-1}^i - \Sigma_{k|k-1}^i (C^i)^T (S_k^i)^{-1} C^i \Sigma_{k|k-1}^i), \quad (3.35b)$$

$$\eta_k^i = \eta_k = \eta_{k-1} + m, \quad (3.35c)$$

where

$$S_k^i = C^i \Sigma_{k|k-1}^i (C^i)^T + R^i, \quad (3.36a)$$

$$\hat{y}_{k|k-1}^i = C^i m_{k|k-1}^i, \quad (3.36b)$$

$$\Delta_i^2 = (y_k - \hat{y}_{k|k-1}^i)^T (S_k^i)^{-1} (y_k - \hat{y}_{k|k-1}^i). \quad (3.36c)$$

In order to preserve the heaviest tail property, the degrees of freedom parameter η_k is reduced to η_{k-1} as in STF given in previous section. Before reducing η_k , the filtered scale matrix is updated as

$$\Sigma_{k|k}^i = \Sigma_{k|k}^i \left(\frac{\eta_k}{\eta_k - 2} \right) \left(\frac{\eta_{k-1} - 2}{\eta_{k-1}} \right). \quad (3.37)$$

Now, to obtain the overall mean and scale matrix, mode probabilities $\{\mu_k^i\}_{i=1}^N$ should be computed:

$$\begin{aligned} \mu_k^i &\triangleq P(r_k = i | y_{1:k}) \propto p(y_k | y_{1:k-1}, r_k = i) P(r_k = i | y_{1:k-1}) \\ &\propto St(y_k; \hat{y}_{k|k-1}^i, S_k^i, \eta_{k-1}) \sum_{j=1}^N P(r_k = i | r_{k-1} = j) P(r_{k-1} = j | y_{1:k-1}) \\ &\propto St(y_k; \hat{y}_{k|k-1}^i, S_k^i, \eta_{k-1}) \sum_{j=1}^N \pi_{ji} \mu_{k-1}^j. \end{aligned} \quad (3.38)$$

Therefore,

$$\mu_k^i = \frac{St(y_k; \hat{y}_{k|k-1}^i, S_k^i, \eta_{k-1}) \sum_{j=1}^N \pi_{ji} \mu_{k-1}^j}{\sum_{l=1}^N St(y_k; \hat{y}_{k|k-1}^l, S_k^l, \eta_{k-1}) \sum_{j=1}^N \pi_{jl} \mu_{k-1}^j}. \quad (3.39)$$

Since we have the means, the scale matrices and the updated mode probabilities for all filter models $\{m_{k|k}^i, \Sigma_{k|k}^i, \mu_k^i\}_{i=1}^N$, the overall mean and scale matrix are calculated as

$$m_{k|k} = \sum_{i=1}^N \mu_k^i m_{k|k}^i, \quad (3.40a)$$

$$\Sigma_{k|k} = \frac{\eta_k - 2}{\eta_k} \sum_{i=1}^N \mu_k^i \left[\frac{\eta_k}{\eta_k - 2} \Sigma_{k|k}^i + (m_{k|k}^i - m_{k|k})(m_{k|k}^i - m_{k|k})^T \right]. \quad (3.40b)$$

3.3 Performance Evaluation

In this section, the conventional IMM and IMM-STF algorithms are tested on a simulation of a maneuvering target in 2-D space that have outliers in process and measurement noise. Two motion models, constant velocity (CV) [8] and coordinated turn (CT) with known turn rate [40], are used to model the motion of the target. The state consists of positions and velocities in two dimensions as

$$x_k = \begin{bmatrix} p_k^x \\ v_k^x \\ p_k^y \\ v_k^y \end{bmatrix}, \quad (3.41)$$

where p_k^x, p_k^y, v_k^x and v_k^y are positions and velocities in x and y dimensions at the time k , respectively. The system matrices in equation (2.42) for CV and CT models are

$$A^{CV} = \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix}, A^{CT} = \begin{bmatrix} 1 & \frac{\sin(\omega T)}{\omega} & 0 & -\frac{1 - \cos(\omega T)}{\omega} \\ 0 & \cos(\omega T) & 0 & -\sin(\omega T) \\ 0 & \frac{1 - \cos(\omega T)}{\omega} & 1 & \frac{\sin(\omega T)}{\omega} \\ 0 & \sin(\omega T) & 0 & \cos(\omega T) \end{bmatrix}, \quad (3.42)$$

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}, \quad (3.43)$$

where $T = 1$ s is the sampling time and $\omega = \pi/15$ rad/s is the turn rate of CT model. A^{CV} and A^{CT} are the state transition matrices for CV and CT models, respectively. We consider three different cases for noise generation. These cases are given below.

Case A:

The process and measurement noises are generated according to

$$w_k \sim \begin{cases} \mathcal{N}(0, Q) & \text{w.p. } 0.95 \\ \mathcal{N}(0, 1000Q) & \text{w.p. } 0.05 \end{cases} \quad (3.44)$$

$$v_k \sim \begin{cases} \mathcal{N}(0, R) & \text{w.p. } 0.9 \\ \mathcal{N}(0, 100R) & \text{w.p. } 0.1 \end{cases} \quad (3.45)$$

Case B:

The process noise is generated as $w_k \sim \mathcal{N}(0, Q)$ and the measurement noise is generated according to (3.45), i.e., only measurement noise have outliers.

Case C:

The process noise is generated as $w_k \sim \mathcal{N}(0, Q)$ and the measurement noise is generated as $v_k \sim \mathcal{N}(0, R)$, i.e., both process and measurement noises don't have outliers.

The noise parameters are

$$Q = \begin{bmatrix} \frac{T^3}{2} & \frac{T^2}{2} & 0 & 0 \\ \frac{3}{2}T^2 & T & 0 & 0 \\ 0 & 0 & \frac{T^3}{2} & \frac{T^2}{2} \\ 0 & 0 & \frac{3}{2}T^2 & T \end{bmatrix} \times q, \quad q = 1, \quad R = r \times I_2, \quad r = 100. \quad (3.46)$$

The target is simulated as switching between two modes which are CV mode and CT mode. The motion of the target is simulated according to

- the CV model between $k = 0s$ and $k = 100s$,
- the CT model between $k = 101s$ and $k = 150s$,
- the CV model between $k = 151s$ and $k = 250s$,

and the switching is controlled by TPM

$$\Pi = \begin{bmatrix} 0.95 & 0.05 \\ 0.05 & 0.95 \end{bmatrix}. \quad (3.47)$$

In this simulation, the position and the velocity of the target are estimated with both IMM algorithm and IMM-STF proposed in this chapter. The IMM algorithm is only aware of the nominal noise covariances Q and R . The IMM-STF is also aware of the nominal noise covariances. However, it assumes that the noises are t-distributed with degrees of freedom value 3. Therefore, the scale matrices of the noises are adjusted as $0.92 \times Q$ and $0.92 \times R$ for IMM-STF to preserve the 80% probability region. The simulation is carried out for 250 Monte Carlo (MC) runs.

True and the estimated trajectories obtained by IMM and IMM-STF algorithms for a single run for Case 1 are shown in Figure 3.11. It is seen from Figure 3.11 that the estimated trajectory by IMM-STF is closer to the true trajectory than the estimated trajectory by IMM algorithm, especially at the moments when outliers occur.

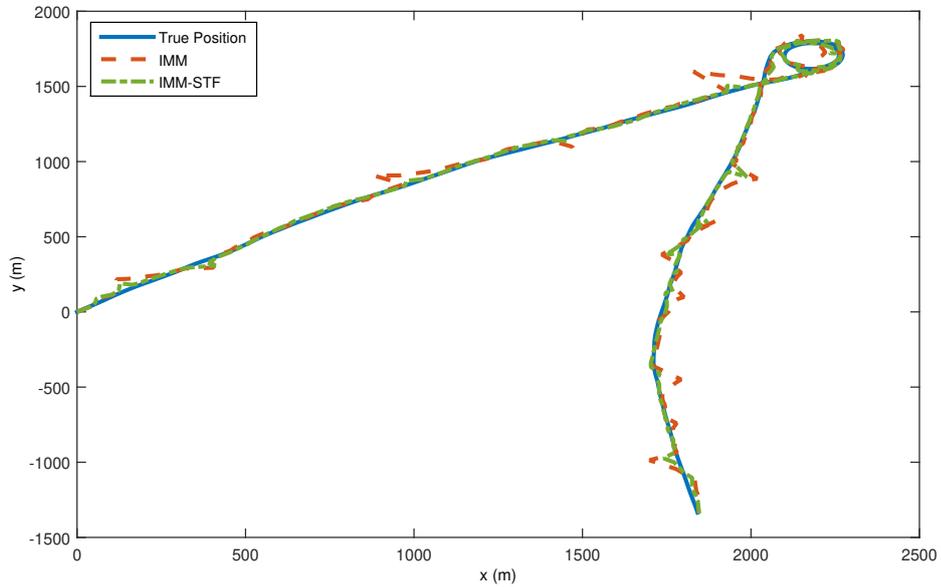


Figure 3.11: True and estimated trajectories of the target for Case A. True trajectory is shown by blue line, the trajectory estimated by IMM algorithm is shown by orange line and the trajectory estimated by IMM-STF is shown by green line.

Figures 3.12, 3.13, 3.14, 3.15, 3.16 and 3.17 show position and velocity errors by IMM and IMM-STF for a single run for Case A. The errors are calculated by equation (2.41). It can be seen from the figures that the error of IMM-STF is generally smaller than the error of IMM when outliers occur. This means that IMM-STF has ability to mask the effect of outliers while IMM does not. In addition, Figure 3.18 illustrates the mode probabilities for IMM-STF for Case A. As seen in Figure 3.18, the algorithm switches to CT model for the time interval between $k = 100$ s and $k = 150$ s and this means that the algorithm detects the motion model of the target.

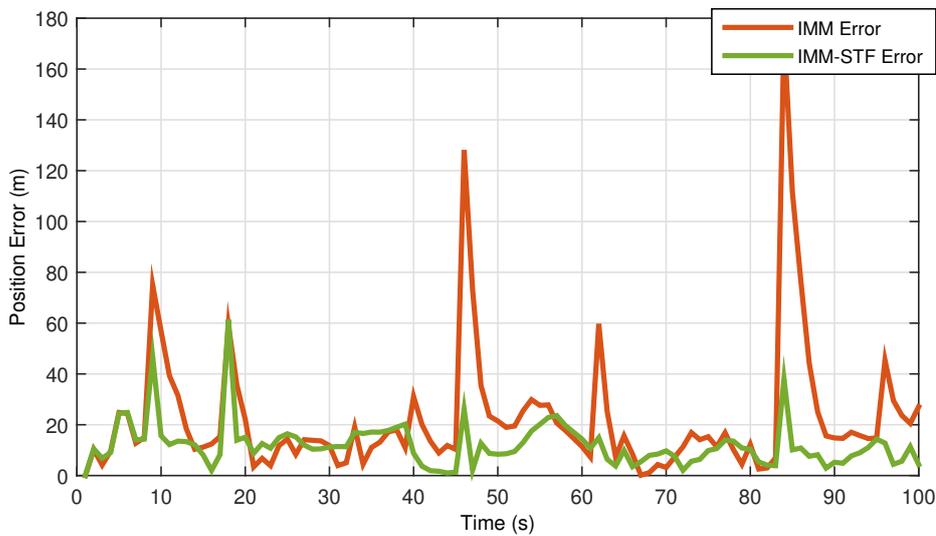


Figure 3.12: Position errors of IMM and IMM-STF algorithms for time interval $k=0$ and $k=100$ for Case A. In this time interval, the target moves according to CV model. The error of IMM is shown by orange line and the error of IMM-STF is shown by green line.

The RMSEs calculated by equation (3.16) and ARMSEs calculated by equation (3.17) of position and velocity by IMM and IMM-STF for Case A are shown in Figures 3.19 and 3.20 and Tables 3.5 and 3.6.

The RMSEs and ARMSEs of position and velocity by IMM and IMM-STF for Case B are shown in Figures 3.21 and 3.22 and Tables 3.7 and 3.8.

The RMSEs and ARMSEs of position and velocity by IMM and IMM-STF for Case C are shown in Figures 3.23 and 3.24 and Tables 3.9 and 3.10.

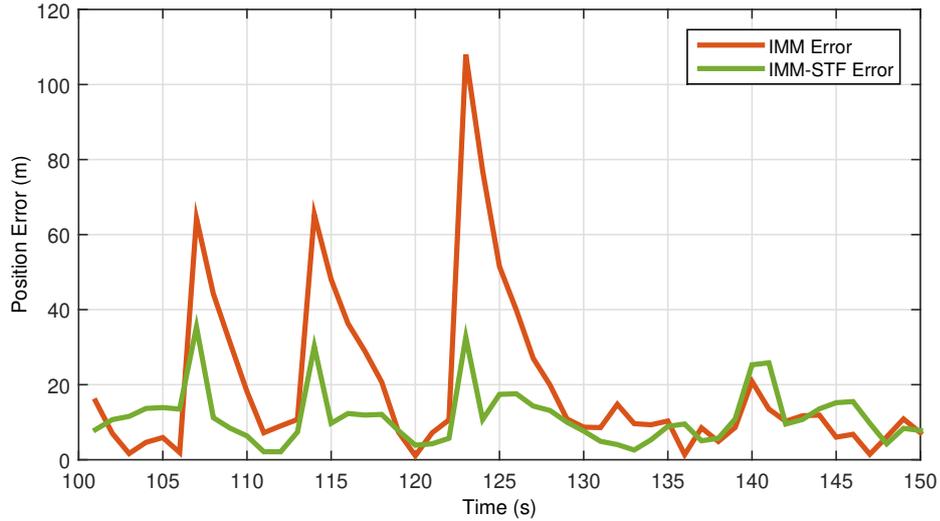


Figure 3.13: Position errors of IMM and IMM-STF algorithms for time interval $k=101$ and $k=150$ for Case A. In this time interval, the target moves according to CT model. The error of IMM is shown by orange line and the error of IMM-STF is shown by green line.

Table 3.5: ARMSEs of IMM and IMM-STF for Case A for CV mode

	ARMSE of the position (m)	ARMSE of the velocity (m/s)
IMM-STF	26.65	17.35
IMM	36.58	25.06

These figures and tables demonstrate that IMM-STF outperforms IMM algorithm in terms of RMSEs of position and velocity for multiple-model systems that have heavy-tailed process and measurement noises. As a result, it can be said that IMM-STF can accomplish both the multimodality and the outliers in the noises. If both process and measurement noise don't have outliers, IMM-STF and IMM algorithm provide nearly the same position estimates.

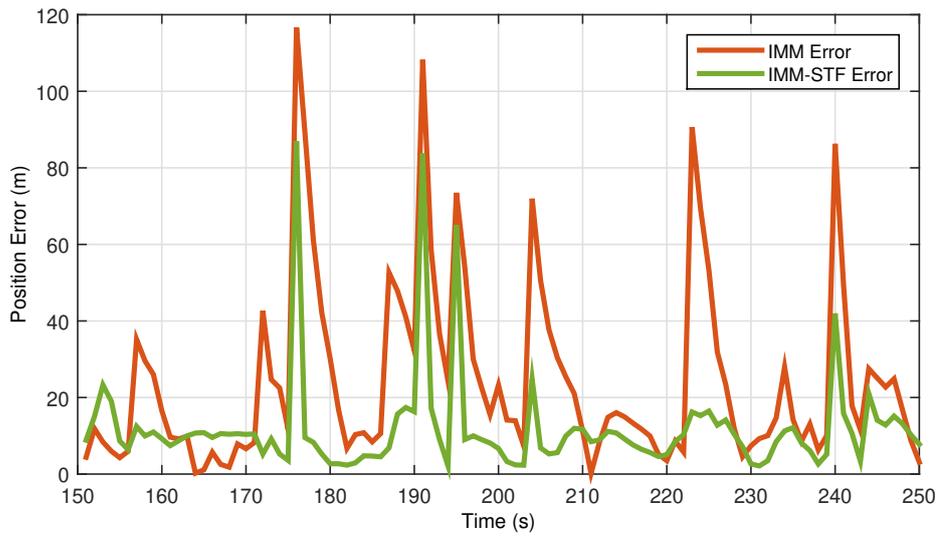


Figure 3.14: Position errors of IMM and IMM-STF algorithms for time interval $k=151$ and $k=250$ for Case A. In this time interval, the target moves according to CV model. The error of IMM is shown by orange line and the error of IMM-STF is shown by green line.

Table 3.6: ARMSEs of IMM and IMM-STF for Case A for CT mode

	ARMSE of the position (m)	ARMSE of the velocity (m/s)
IMM-STF	26.37	17.73
IMM	36.90	26.37

Table 3.7: ARMSEs of IMM and IMM-STF for Case B for CV mode

	ARMSE of the position (m)	ARMSE of the velocity (m/s)
IMM-STF	20.36	7.58
IMM	30.34	11.98

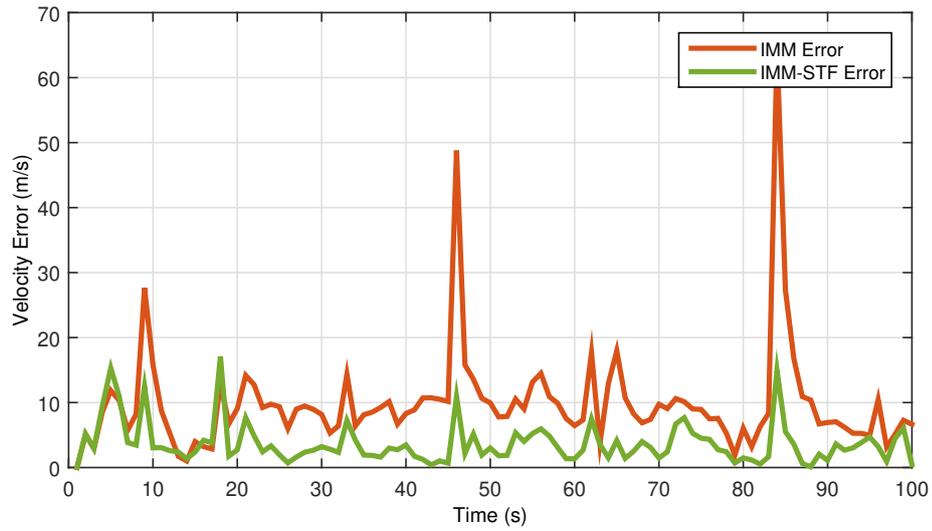


Figure 3.15: Velocity errors of IMM and IMM-STF algorithms for time interval $k=0$ and $k=100$ for Case A. In this time interval, the target moves according to CV model. The error of IMM is shown by orange line and the error of IMM-STF is shown by green line.

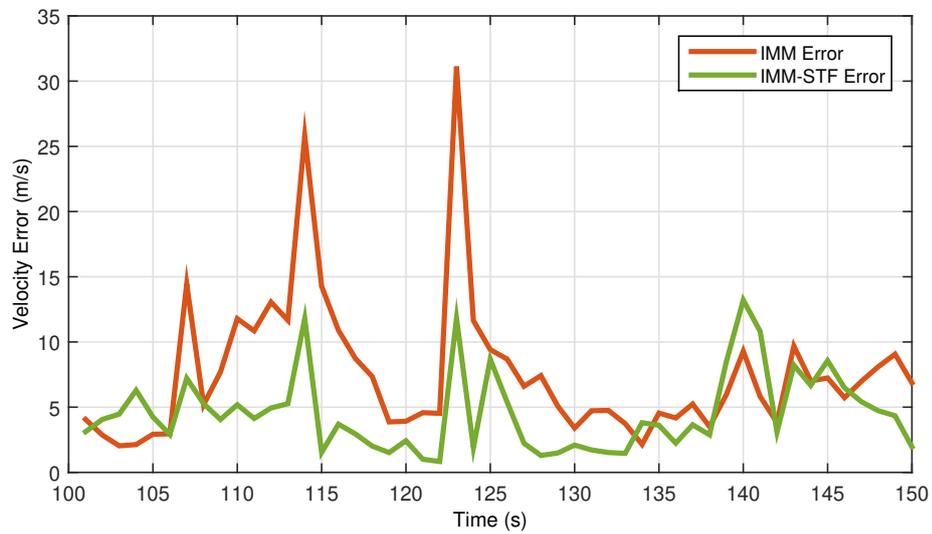


Figure 3.16: Velocity errors of IMM and IMM-STF algorithms for time interval $k=101$ and $k=150$ for Case A. In this time interval, the target moves according to CT model. The error of IMM is shown by orange line and the error of IMM-STF is shown by green line.

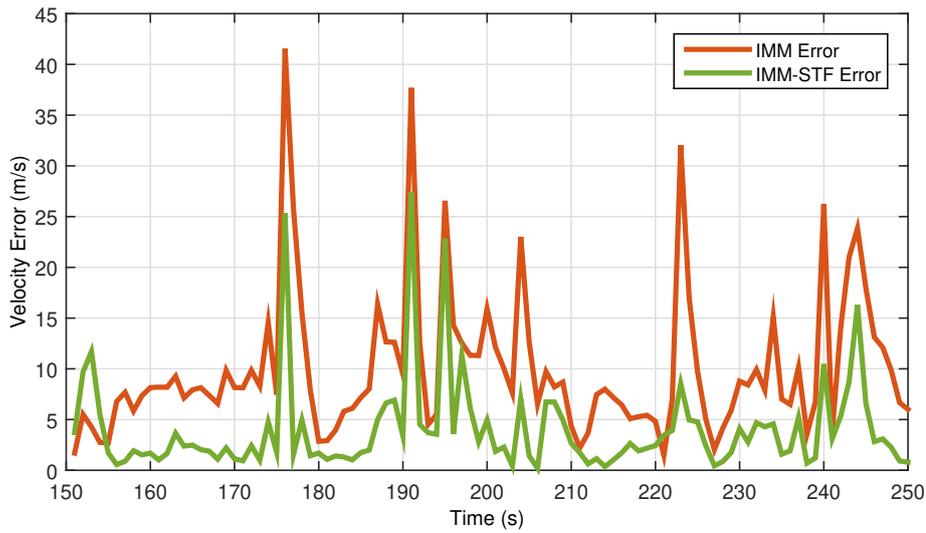


Figure 3.17: Velocity errors of IMM and IMM-STF algorithms for time interval $k=151$ and $k=250$ for Case A. In this time interval, the target moves according to CV model. The error of IMM is shown by orange line and the error of IMM-STF is shown by green line.

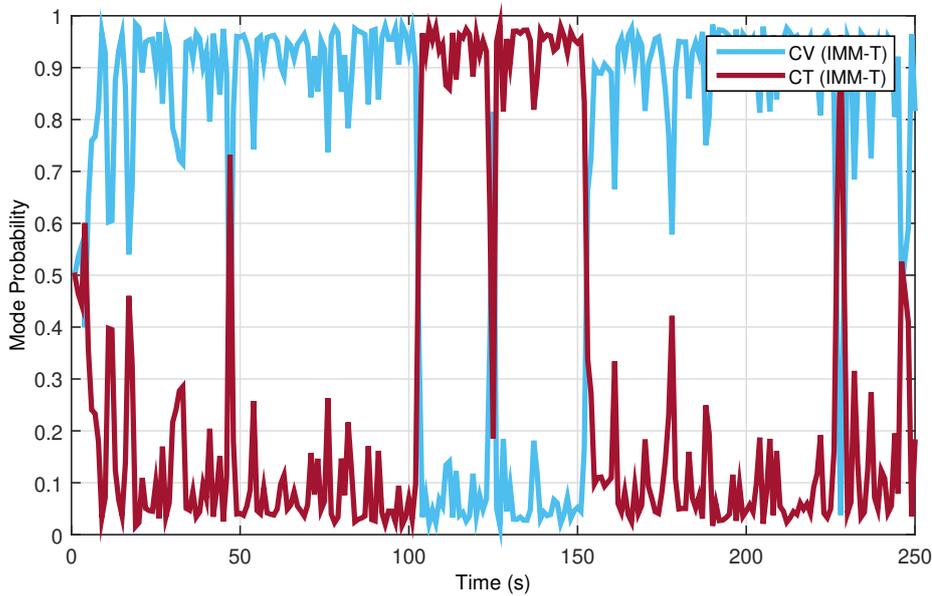


Figure 3.18: Mode probabilities of CV and CT models for IMM-STF for Case A. The mode probability of CV model is shown by blue line and the mode probability of CT model is shown by red line.

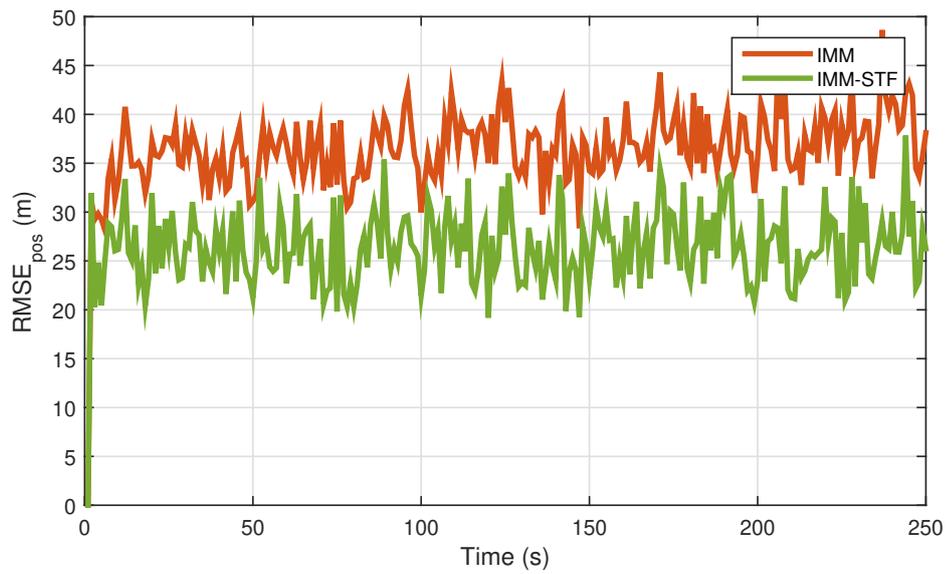


Figure 3.19: RMSEs of the position for 250 Monte Carlo run for Case A. RMSEs of the position by IMM is shown by orange line and RMSEs of the position by IMM-STF is shown by green line.

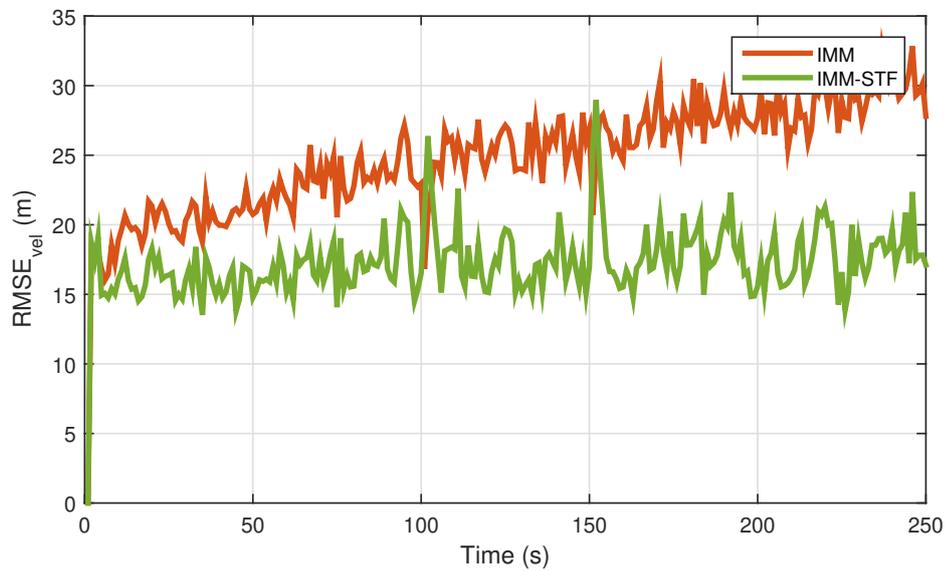


Figure 3.20: RMSEs of the velocity for 250 Monte Carlo run for Case A. RMSEs of the velocity by IMM is shown by orange line and RMSEs of the velocity by IMM-STF is shown by green line.

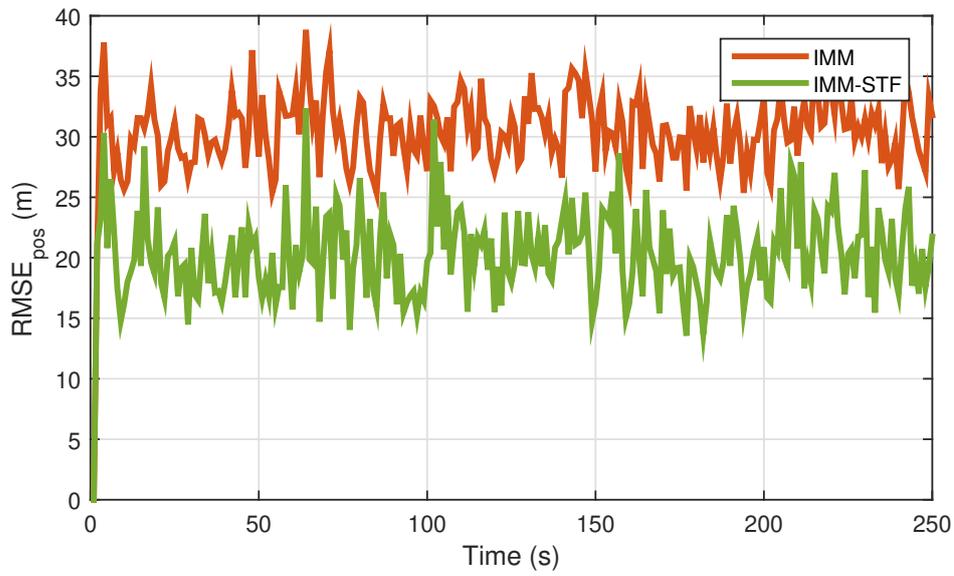


Figure 3.21: RMSEs of the position for 250 Monte Carlo run for Case B. RMSEs of the position by IMM is shown by orange line and RMSEs of the position by IMM-STF is shown by green line.

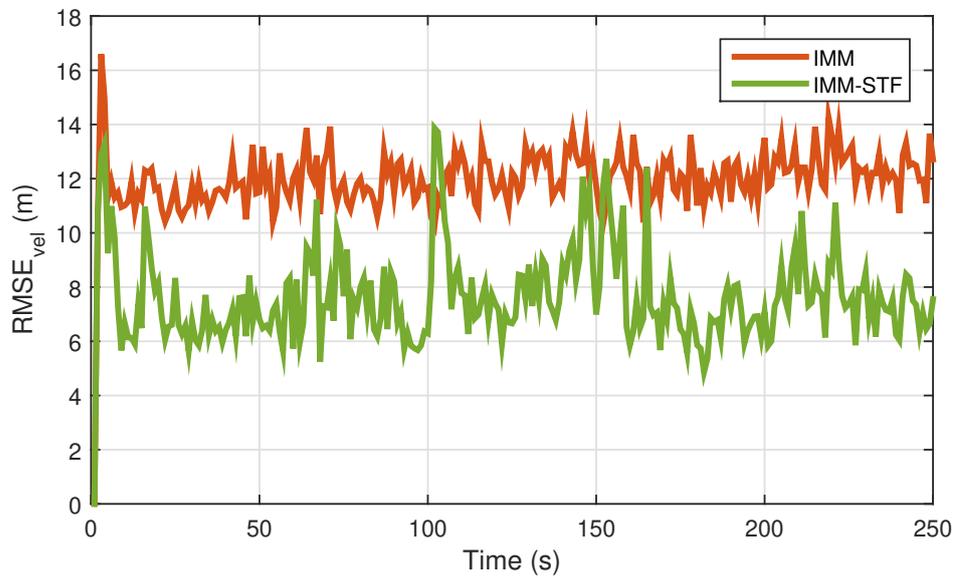


Figure 3.22: RMSEs of the velocity for 250 Monte Carlo run for Case B. RMSEs of the velocity by IMM is shown by orange line and RMSEs of the velocity by IMM-STF is shown by green line.

Table 3.8: ARMSEs of IMM and IMM-STF for Case B for CT mode

	ARMSE of the position (m)	ARMSE of the velocity (m/s)
IMM-STF	21.33	8.61
IMM	31.39	12.19

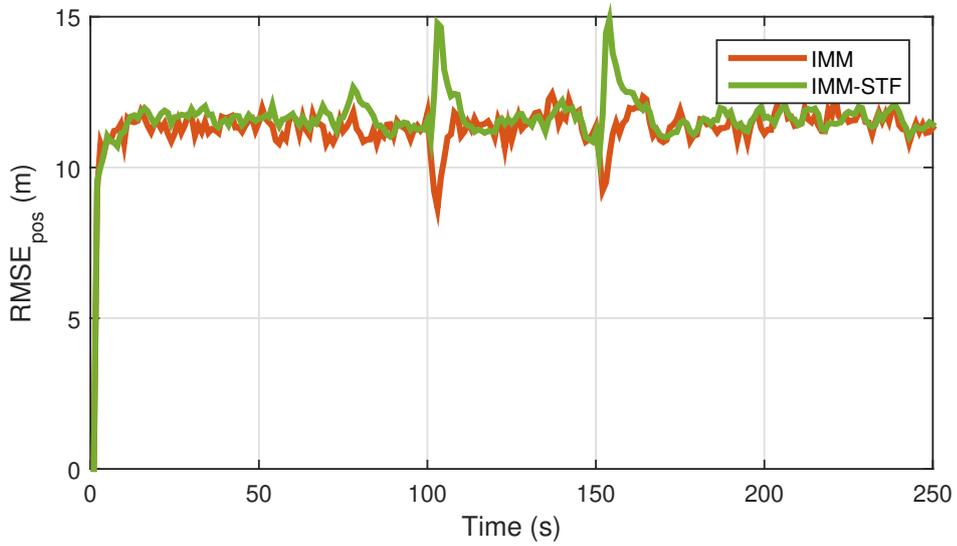


Figure 3.23: RMSEs of the position for 250 Monte Carlo run for Case C. RMSEs of the position by IMM is shown by orange line and RMSEs of the position by IMM-STF is shown by green line.

Table 3.9: ARMSEs of IMM and IMM-STF for Case C for CV mode

	ARMSE of the position (m)	ARMSE of the velocity (m/s)
IMM-STF	11.62	4.12
IMM	11.33	8.29

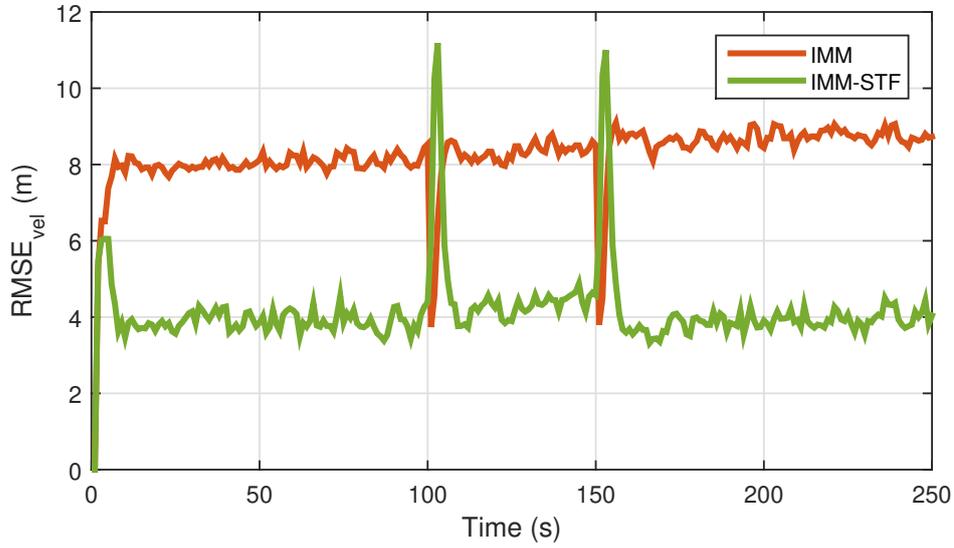


Figure 3.24: RMSEs of the velocity for 250 Monte Carlo run for Case C. RMSEs of the velocity by IMM is shown by orange line and RMSEs of the velocity by IMM-STF is shown by green line.

Table 3.10: ARMSEs of IMM and IMM-STF for Case C for CT mode

	ARMSE of the position (m)	ARMSE of the velocity (m/s)
IMM-STF	11.78	4.92
IMM	11.32	8.17

CHAPTER 4

OUTLIER ROBUST FILTERS USING VB APPROACH

In the previous chapter, STF [44], which is one of the proposed methods to solve the filtering problem of the systems that have heavy-tailed noises, is explained and a multiple-model extension of STF is carried out based on IMM approach. In addition to STF, variational Bayesian (VB) methods have been proposed for the systems that have outliers in noises [1,2,22]. VB methods can also be used to derive noise adaptive filters for unknown, inaccurate or time varying noises [4, 5, 23, 46]. In this chapter, two different VB methods are evaluated in terms of robustness towards outliers. VB approach which uses Gamma-Gaussian prior for heavy-tailed noise is referred as VB-GG approach and VB approach which uses inverse-Wishart prior for the covariance matrix of heavy-tailed noise is referred as VB-IW approach. In contrast to STF [44], these VB algorithms approximate the predicted and filtered PDFs as Gaussians.

4.1 Gamma-Gaussian (GG) Approach

In this section, the derivation of VB-GG is given. In this VB algorithm, Student's-t distributed noise is expressed as the marginal distribution of a random vector when the joint distribution of this random vector and a defined auxiliary variable is a Gamma-Gaussian. This way of derivation of Student's-t distribution is mentioned as "Gamma-Gaussian expression of t distribution" in the rest of the thesis for convenience.

Let the scalar random variable u be Gamma distributed with shape parameter α and

rate parameter β , i.e., $u \sim \text{Gam}(u; \alpha, \beta)$. The PDF of u is

$$\text{Gam}(u; \alpha, \beta) = \begin{cases} \frac{\beta^\alpha}{\Gamma(\alpha)} u^{\alpha-1} \exp(-\beta u), & u > 0 \\ 0, & u \leq 0 \end{cases} \quad (4.1)$$

where $\alpha > 0$, $\beta > 0$ and

$$\Gamma(z) = \int_0^\infty e^{-t} t^{z-1} dt, \quad (4.2)$$

is the Gamma function [42]. Let $X \in R^n$ be a random vector and PDF of X be

$$p_X(x) = \int_0^\infty \mathcal{N}(x; m, P/u) \text{Gam}(u; \alpha, \beta) du. \quad (4.3)$$

Using equations (2.13) and (4.1), the PDF of X becomes

$$p_X(x) = \int_0^\infty \frac{\beta^\alpha}{\Gamma(\alpha)} u^{\alpha-1} \exp(-\beta u) \frac{1}{(2\pi)^{n/2}} \frac{1}{\sqrt{\det(P/u)}} \exp\left(-\frac{u}{2} \Delta^2\right) du, \quad (4.4)$$

where

$$\Delta^2 = (x - m)^T P^{-1} (x - m). \quad (4.5)$$

Equation (4.4) can be rewritten as

$$p_X(x) = \frac{\beta^\alpha}{\Gamma(\alpha)} \frac{1}{(2\pi)^{n/2}} \frac{1}{\sqrt{\det(P)}} \int_0^\infty u^{\frac{n}{2} + \alpha - 1} \exp\left(-u \left(\beta + \frac{\Delta^2}{2}\right)\right) du. \quad (4.6)$$

Now, we define the integral in (4.6) as

$$A = \int_0^\infty u^{\frac{n}{2} + \alpha - 1} \exp\left(-u \left(\beta + \frac{\Delta^2}{2}\right)\right) du. \quad (4.7)$$

In order to calculate A , the Gamma function of $\frac{n}{2} + \alpha$ is defined as

$$\Gamma\left(\frac{n}{2} + \alpha\right) = \int_0^{\infty} e^{-t} t^{\frac{n}{2} + \alpha - 1} dt. \quad (4.8)$$

Let

$$t = u \left(\beta + \frac{\Delta^2}{2} \right) \quad \text{and} \quad dt = \left(\beta + \frac{\Delta^2}{2} \right) du. \quad (4.9)$$

Equation (4.8) can be written as

$$\begin{aligned} \Gamma\left(\frac{n}{2} + \alpha\right) &= \left(\beta + \frac{\Delta^2}{2} \right)^{\frac{n}{2} + \alpha} \int_0^{\infty} \exp\left(-u \left(\beta + \frac{\Delta^2}{2} \right)\right) u^{\frac{n}{2} + \alpha - 1} du \\ &= \left(\beta + \frac{\Delta^2}{2} \right)^{\frac{n}{2} + \alpha} A. \end{aligned} \quad (4.10)$$

From equation (4.10), A is computed as

$$A = \left(\beta + \frac{\Delta^2}{2} \right)^{-\frac{n}{2} - \alpha} \Gamma\left(\frac{n}{2} + \alpha\right). \quad (4.11)$$

Now, we have A so the PDF of X is rewritten as

$$p_X(x) = \frac{\Gamma\left(\frac{n}{2} + \alpha\right)}{\Gamma(\alpha)} \frac{1}{(2\beta\pi)^{n/2}} \frac{1}{\sqrt{\det(P)}} \left(1 + \frac{\Delta^2}{2\beta}\right)^{-\frac{n}{2} - \alpha}. \quad (4.12)$$

If $\alpha = \beta = \frac{\nu}{2}$, equation (4.12) is equal to the PDF of a t-distributed random vector (3.1). Therefore, the PDF of a t-distributed random vector $X \sim St(x; m, P, \nu)$ can be expressed as

$$p_X(x) = \int_0^{\infty} \mathcal{N}(x; m, P/u) \text{Gam}(u; \nu/2, \nu/2) du. \quad (4.13)$$

(4.13) is the Gamma-Gaussian expression of t distribution. This derivation is available in [42].

4.1.1 Derivation of Variational Bayesian Algorithm using Gamma-Gaussian Approach (VB-GG)

An approximate solution for the posterior PDF for a linear SSM with heavy-tailed measurement noise is derived by using VB approach. In this derivation, the heavy-tailed measurement noise is modeled using the Gamma-Gaussian expression. We consider the linear SSM (2.5) where

- w_{k-1} is distributed with $\mathcal{N}(0, Q)$,
- v_k is distributed with $St(0, R, v)$.

One-step predicted PDF $p(x_k|y_{1:k-1})$ and the likelihood PDF $p(y_k|x_k)$ are

$$p(x_k|y_{1:k-1}) = \mathcal{N}(x_k; m_{k|k-1}, P_{k|k-1}), \quad (4.14a)$$

$$p(y_k|x_k) = St(y_k; Cx_k, R, v). \quad (4.14b)$$

The parameters of the one-step predicted PDF are calculated by using KF time update equation (2.8) as

$$m_{k|k-1} = Am_{k-1|k-1}, \quad (4.15a)$$

$$P_{k|k-1} = AP_{k-1|k-1}A^T + Q. \quad (4.15b)$$

The likelihood (4.14b) can be rewritten using (4.13) as

$$p(y_k|x_k) = \int \mathcal{N}(y_k; Cx_k, R/\lambda_k) Gam\left(\lambda_k; \frac{v}{2}, \frac{v}{2}\right) d\lambda_k. \quad (4.16)$$

Therefore, we have

$$p(y_k|x_k, \lambda_k) = \mathcal{N}(y_k; Cx_k, R/\lambda_k), \quad (4.17a)$$

$$p(\lambda_k) = Gam\left(\lambda_k; \frac{v}{2}, \frac{v}{2}\right). \quad (4.17b)$$

In order to estimate the state x_k , the joint posterior PDF $p(x_k, \lambda_k|y_{1:k})$ needs to be computed. However, one cannot obtain an analytical solution for this posterior PDF.

Thus, we need to obtain an approximate solution using VB approach. According to VB approach, the posterior PDF can be approximated as factorized densities

$$p(x_k, \lambda_k | y_{1:k}) \approx q(x_k)q(\lambda_k), \quad (4.18)$$

where $q(\cdot)$ defines the PDF for corresponding variables. Among all such factorized densities, an optimal one can be computed by minimizing the Kullback-Leibler divergence between the approximate posterior PDF and the true posterior PDF [11] as

$$\{q^*(x_k), q^*(\lambda_k)\} = \mathbf{argmin} KLD(q(x_k)q(\lambda_k) || p(x_k, \lambda_k | y_{1:k})), \quad (4.19)$$

where KLD is the Kullback-Leibler divergence given in (2.63). According to equation (2.66),

$$\log q^*(\phi) = \mathbf{E}_{\Theta(-\phi)}[\log p(\Theta, y_{1:k})] + c_\phi, \quad (4.20)$$

where $\Theta \triangleq \{x_k, \lambda_k\}$ and ϕ is an arbitrary element of Θ . $\Theta^{(-\phi)}$ is the set of all elements of Θ except for ϕ , $\mathbf{E}[\cdot]$ is the expectation operation and c_ϕ is a constant. For solving equation (4.20), fixed point iterations should be carried out. The joint PDF $p(\Theta, y_{1:k})$ can be written as

$$p(\Theta, y_{1:k}) = p(y_k | x_k, \lambda_k) p(x_k | y_{1:k-1}) p(\lambda_k). \quad (4.21)$$

Substituting (4.14) and (4.17) in (4.21), we get

$$p(\Theta, y_{1:k}) = \mathcal{N}(y_k; Cx_k, R/\lambda_k) \mathcal{N}(x_k; m_{k|k-1}, P_{k|k-1}) \text{Gam}(\lambda_k; \frac{v}{2}, \frac{v}{2}) \quad (4.22)$$

$$\begin{aligned}
&= \frac{1}{(2\pi)^{\frac{m}{2}}} \frac{1}{\sqrt{\det(R/\lambda_k)}} \exp\left(-\frac{\lambda_k}{2}(y_k - Cx_k)^T R^{-1}(y_k - Cx_k)\right) \\
&\quad \times \frac{1}{(2\pi)^{\frac{n}{2}}} \frac{1}{\sqrt{\det(P_{k|k-1})}} \exp\left(-\frac{1}{2}(x_k - m_{k|k-1})^T P_{k|k-1}^{-1}(x_k - m_{k|k-1})\right) \quad (4.23) \\
&\quad \times \frac{\left(\frac{v}{2}\right)^{\frac{v}{2}}}{\Gamma\left(\frac{v}{2}\right)} \lambda_k^{\left(\frac{v}{2}-1\right)} \exp\left(-\frac{v}{2}\lambda_k\right).
\end{aligned}$$

By taking the logarithm of both sides, we obtain

$$\begin{aligned}
\log p(\Theta, y_{1:k}) &= \left(\frac{m+v}{2} - 1\right) \log \lambda_k - \frac{v}{2} \lambda_k \\
&\quad - \frac{\lambda_k}{2} (y_k - Cx_k)^T R^{-1} (y_k - Cx_k) \quad (4.24) \\
&\quad - \frac{1}{2} (x_k - m_{k|k-1})^T P_{k|k-1}^{-1} (x_k - m_{k|k-1}).
\end{aligned}$$

Let $\phi = \lambda_k$ and by using equations (4.20) and (4.24), the following equation is obtained:

$$\log q(\lambda_k)^{(i+1)} = \left(\frac{m+v}{2} - 1\right) \log \lambda_k - \frac{1}{2} \left[v + \text{tr}(E_k^{(i)} R^{-1}) \right] \lambda_k + c_\lambda, \quad (4.25)$$

where $q(\cdot)^{(i+1)}$ is the approximation of $q(\cdot)$ at $(i+1)$ th iteration, $\text{tr}(\cdot)$ is the trace operation and

$$E_k^{(i)} = \mathbf{E}^{(i)} \left[(y_k - Cx_k)(y_k - Cx_k)^T \right]. \quad (4.26)$$

Using (4.25), $q(\lambda_k)^{(i+1)}$ is updated as Gamma PDF as

$$q(\lambda_k)^{(i+1)} = \text{Gam}(\lambda_k; \gamma_k^{i+1}, \delta_k^{i+1}), \quad (4.27)$$

where

$$\gamma_k^{i+1} = \frac{1}{2}(m+v), \quad (4.28a)$$

$$\delta_k^{i+1} = \frac{1}{2} \left[v + \text{tr}(E_k^{(i)} R^{-1}) \right]. \quad (4.28b)$$

Let $\phi = x_k$ and using equations (4.20) and (4.24), the following equation is obtained:

$$\begin{aligned} \log q(x_k)^{(i+1)} = & -\frac{1}{2} \mathbf{E}[\lambda_k]^{(i+1)} (y_k - Cx_k)^T R^{-1} (y_k - Cx_k) \\ & -\frac{1}{2} (x_k - m_{k|k-1})^T P_{k|k-1}^{-1} (x_k - m_{k|k-1}) + c_x. \end{aligned} \quad (4.29)$$

The modified likelihood PDF $p(y_k|x_k)^{(i+1)}$ is expressed as

$$p(y_k|x_k)^{(i+1)} = \mathcal{N}\left(y_k; Cx_k, \tilde{R}_k^{(i+1)}\right), \quad (4.30)$$

where $\tilde{R}_k^{(i+1)}$ is the effective covariance matrix, which is used in the measurement update of the state at (i+1)th iteration. It is calculated as

$$\tilde{R}_k^{(i+1)} = \frac{R}{\mathbf{E}[\lambda_k]^{i+1}}. \quad (4.31)$$

Using equations (4.29), (4.30) and (4.31), $q(x_k)^{(i+1)}$ is updated as Gaussian PDF as

$$q(x_k)^{(i+1)} = \mathcal{N}\left(x_k; m_{k|k}^{(i+1)}, P_{k|k}^{(i+1)}\right), \quad (4.32)$$

where

$$m_{k|k}^{(i+1)} = m_{k|k-1} + K_k^{(i+1)}(y_k - Cm_{k|k-1}), \quad (4.33a)$$

$$P_{k|k}^{(i+1)} = P_{k|k-1} - K_k^{(i+1)}CP_{k|k-1}, \quad (4.33b)$$

and

$$K_k^{(i+1)} = P_{k|k-1}C^T \left(CP_{k|k-1}C^T + \tilde{R}_k^{(i+1)} \right)^{-1}. \quad (4.34)$$

After N iterations, the approximate posterior PDF of the state becomes

$$q^*(x_k) \approx q(x_k)^{(N)} = \mathcal{N}\left(x_k; m_{k|k}^{(N)}, P_{k|k}^{(N)}\right) = \mathcal{N}(x_k; m_{k|k}, P_{k|k}). \quad (4.35)$$

The expectations $\mathbf{E}[\lambda_k]^{(i+1)}$ and $E_k^{(i)}$ are calculated as

$$\mathbf{E}[\lambda_k]^{(i+1)} = \frac{\gamma_k^{i+1}}{\delta_k^{i+1}}, \quad (4.36a)$$

$$E_k^{(i)} = \left(y_k - C m_{k|k}^{(i)} \right) \left(y_k - C m_{k|k}^{(i)} \right)^T + C P_{k|k}^{(i)} C^T. \quad (4.36b)$$

VB-GG algorithm is summarized in Algorithm 1.

Algorithm 1 VB-GG Algorithm

Inputs: $m_{k-1|k-1}, P_{k-1|k-1}, A, C, y_k, Q, R, m, n, v, N$

Outputs: $m_{k|k}, P_{k|k}$

Time Update:

1. $m_{k|k-1} = Am_{k-1|k-1}$
2. $P_{k|k-1} = AP_{k-1|k-1}A^T + Q$

Measurement Update:

3. Initialization: $m_{k|k}^{(0)} = m_{k|k-1}, P_{k|k}^{(0)} = P_{k|k-1}$

for $i = 0 : N - 1$ **do**

Update $q(\lambda_k)^{(i+1)}$ given $q(x_k)^{(i)}$

4. $E_k^{(i)} = (y_k - Cm_{k|k}^{(i)})(y_k - Cm_{k|k}^{(i)})^T + CP_{k|k}^{(i)}C^T$

5. $\gamma_k^{i+1} = \frac{1}{2}(m + v), \delta_k^{i+1} = \frac{1}{2}\left(v + \text{tr}\left(E_k^{(i)}R^{-1}\right)\right), \mathbf{E}[\lambda_k]^{(i+1)} = \frac{\gamma_k^{i+1}}{\delta_k^{i+1}}$

Update $q(x_k)^{(i+1)}$ given $q(\lambda_k)^{(i+1)}$

6. $\tilde{R}_k^{(i+1)} = R/\mathbf{E}[\lambda_k]^{(i+1)}$

7. $K_k^{(i+1)} = P_{k|k-1}C^T \left(CP_{k|k-1}C^T + \tilde{R}_k^{(i+1)} \right)^{-1}$

8. $m_{k|k}^{(i+1)} = m_{k|k-1} + K_k^{(i+1)}(y_k - Cm_{k|k-1})$

9. $P_{k|k}^{(i+1)} = P_{k|k-1} - K_k^{(i+1)}CP_{k|k-1}$

end for

10. $m_{k|k} = m_{k|k}^{(N)}, P_{k|k} = P_{k|k}^{(N)}$
-

VB method approach in this section is proposed in [22] for linear SSMs with heavy-tailed process and measurement noise. In [22], since the process noise is assumed as Student's-t distributed, one-step predicted PDF of the state is also assumed as Student's-t distributed. Then, inverse-Wishart prior is selected for the covariance matrix of one-step predicted PDF of the state and VB approximations are applied [22].

4.2 Inverse Wishart (IW) Approach

In this section, the derivation of VB-IW is given. In this VB algorithm, it is proposed that the inverse Wishart distribution is used as a conjugate prior distribution for the covariance matrix of heavy-tailed measurement noise. This method enables us to find the effective covariance of the measurement noise so it provides better estimation accuracy.

Let Σ , which is a symmetric positive definite random matrix with the dimension $d \times d$, be inverse Wishart distributed. The PDF of Σ is

$$IW(\Sigma; \alpha, U) = \frac{|U|^{\frac{\alpha}{2}} |\Sigma|^{-\frac{(\alpha+d+1)}{2}} \exp\left(-\frac{1}{2} \text{tr}(U\Sigma^{-1})\right)}{2^{\frac{\alpha d}{2}} \Gamma\left(\frac{\alpha}{2}\right)}, \quad (4.37)$$

where α is the degrees of freedom parameter and U is the inverse scale matrix [36]. In addition,

$$\mathbf{E}[\Sigma^{-1}] = (\alpha - d - 1)U^{-1}. \quad (4.38)$$

4.2.1 Derivation of VB Algorithm using Inverse Wishart Approach (VB-IW)

We consider the linear SSM (2.5) where

- w_{k-1} is distributed with $\mathcal{N}(0, Q)$,
- v_k is distributed with $St(0, R, v)$.

One-step predicted PDF $p(x_k|y_{1:k-1})$ and the likelihood PDF $p(y_k|x_k, \Sigma_k)$ are

$$p(x_k|y_{1:k-1}) = \mathcal{N}(x_k; m_{k|k-1}, P_{k|k-1}), \quad (4.39a)$$

$$p(y_k|x_k, \Sigma_k) = \mathcal{N}(y_k; Cx_k, \Sigma_k), \quad (4.39b)$$

where Σ_k is the effective covariance matrix of the measurement noise at time step k . Since the process noise is assumed as Gaussian distributed, the parameters of one-step

predicted PDF are calculated by KF time update equations (2.8) as

$$m_{k|k-1} = Am_{k-1|k-1}, \quad (4.40a)$$

$$P_{k|k-1} = AP_{k-1|k-1}A^T + Q. \quad (4.40b)$$

Now, the goal is to estimate the state x_k and the effective covariance of the measurement noise Σ_k . The posterior PDF $p(\Sigma_{k-1}|y_{1:k-1})$ of effective measurement noise covariance matrix Σ_{k-1} at time step k is assumed as an inverse Wishart PDF, i.e.,

$$p(\Sigma_{k-1}|y_{1:k-1}) = IW(\Sigma_{k-1}; u_{k-1|k-1}, U_{k-1|k-1}), \quad (4.41)$$

where $u_{k-1|k-1}$ and $U_{k-1|k-1}$ are degrees of freedom parameter and inverse scale matrix, respectively. The prior distribution of effective measurement noise covariance matrix at Σ_k is chosen as inverse Wishart distributed as

$$p(\Sigma_k|y_{1:k-1}) = IW(\Sigma_k; u_{k|k-1}, U_{k|k-1}). \quad (4.42)$$

In [23], the measurement noise covariance matrix is considered as slowly varying with time. This time evaluation is formulated as [23]

$$u_{k|k-1} = \lambda(u_{k-1|k-1} - m - 1) + m + 1, \quad (4.43a)$$

$$U_{k|k-1} = \lambda U_{k-1|k-1}. \quad (4.43b)$$

In this algorithm, the initial effective measurement noise covariance matrix Σ_0 is assumed as inverse Wishart distributed, i.e.,

$$p(\Sigma_0) = IW(\Sigma_0; u_{0|0}, U_{0|0}). \quad (4.44)$$

In many real world applications, the scale matrix of the Student's-t distributed measurement noise is not known. Generally, the nominal covariance of the measurement noise is known. Let R_0 be the nominal covariance of the heavy-tailed measurement noise. In this algorithm, the mean of the initial effective measurement noise covariance matrix Σ_0 is set to the nominal covariance of the measurement noise R_0 . There-

fore,

$$R_0 = \frac{U_{0|0}}{u_{0|0} - m - 1}. \quad (4.45)$$

Let

$$u_{0|0} = \tau + m + 1, \quad (4.46)$$

where $\tau \geq 0$ is a tuning parameter and it is used only at the beginning of the algorithm. Hence, using equations (4.45) and (4.46),

$$U_{0|0} = \tau R_0. \quad (4.47)$$

In order to estimate the state x_k and the effective measurement noise covariance Σ_k , the posterior PDF $p(x_k, \Sigma_k | y_{1:k})$ should be calculated. Therefore, we need to obtain an approximate solution by VB approach. According to VB approach, the posterior PDF is approximated as factorized densities

$$p(x_k, \Sigma_k | y_{1:k}) \approx q(x_k)q(\Sigma_k), \quad (4.48)$$

where $q(\cdot)$ defines the PDF for corresponding variables. Among all such factorized densities, an optimal one can be computed by minimizing the Kullback-Leibler divergence between the approximate posterior PDF and the true posterior PDF as

$$\{q^*(x_k), q^*(\Sigma_k)\} = \mathbf{argmin} KL D(q(x_k)q(\Sigma_k) || p(x_k, \Sigma_k | y_{1:k})). \quad (4.49)$$

According to equation (2.66), we have

$$\log q^*(\phi) = \mathbf{E}_{\Theta(-\phi)} [\log p(\Theta, y_{1:k})] + c_\phi, \quad (4.50)$$

where $\Theta = \{x_k, \Sigma_k\}$. As in VB-GG algorithm that expressed in the previous section,

fixed point iterations should be carried out. The joint PDF can be written as

$$p(\Theta, y_{1:k}) = p(y_k | x_k, \Sigma_k) p(x_k | y_{1:k-1}) p(\Sigma_k | y_{1:k-1}) p(y_{1:k-1}). \quad (4.51)$$

Using equations (4.39) and (4.42), we get

$$\begin{aligned} p(\Theta, y_{1:k}) &= \mathcal{N}(y_k; Cx_k, \Sigma_k) \mathcal{N}(x_k; m_{k|k-1}, P_{k|k-1}) IW(\Sigma_k; u_{k|k-1}, U_{k|k-1}) p(y_{1:k-1}) \\ &= \frac{|\Sigma_k|^{-\frac{1}{2}}}{(2\pi)^{\frac{m}{2}}} \exp\left(-\frac{1}{2}(y_k - Cx_k)^T \Sigma_k^{-1} (y_k - Cx_k)\right) \\ &\quad \times \frac{|P_{k|k-1}|^{-\frac{1}{2}}}{(2\pi)^{\frac{n}{2}}} \exp\left(-\frac{1}{2}(x_k - m_{k|k-1})^T P_{k|k-1}^{-1} (x_k - m_{k|k-1})\right) \\ &\quad \times \frac{|U_{k|k-1}|^{\frac{u_{k|k-1}}{2}} |\Sigma_k|^{-\frac{(u_{k|k-1}+m+1)}{2}}}{2^{\frac{mu_{k|k-1}}{2}} \Gamma\left(\frac{u_{k|k-1}}{2}\right)} \exp\left(-\frac{1}{2} \text{tr}(U_{k|k-1} \Sigma_k^{-1})\right). \end{aligned} \quad (4.52)$$

By taking the logarithm of both sides,

$$\begin{aligned} \log p(\Theta, y_{1:k}) &= -\frac{1}{2}(m + u_{k|k-1} + 2) \log |\Sigma_k| - \frac{1}{2}(y_k - Cx_k)^T \Sigma_k^{-1} (y_k - Cx_k) \\ &\quad - \frac{1}{2} \text{tr}(U_{k|k-1} \Sigma_k^{-1}) - \frac{1}{2}(x_k - m_{k|k-1})^T P_{k|k-1}^{-1} (x_k - m_{k|k-1}) + c_\Theta. \end{aligned} \quad (4.53)$$

Let $\phi = \Sigma_k$ and by using equations (4.50) and (4.53), the following equation is obtained:

$$\log q(\Sigma_k)^{(i+1)} = -\frac{1}{2}(m + u_{k|k-1} + 2) \log |\Sigma_k| - \frac{1}{2} \text{tr}\left((E_k^{(i)} + U_{k|k-1}) \Sigma_k^{-1}\right) + c_\Sigma, \quad (4.54)$$

where

$$\begin{aligned} E_k^{(i)} &= \mathbf{E}^i [(y_k - Cx_k)(y_k - Cx_k)^T] \\ &= (y_k - Cm_{k|k-1}^i)(y_k - Cm_{k|k-1}^i)^T + CP_{k|k-1}^{(i)} C^T. \end{aligned} \quad (4.55)$$

By using (4.53), $q(\Sigma_k)^{(i+1)}$ is updated as inverse Wishart PDF as

$$q(\Sigma_k)^{(i+1)} = IW(\Sigma_k; u_k^{(i+1)}, U_k^{(i+1)}), \quad (4.56)$$

where

$$u_k^{(i+1)} = u_{k|k-1} + 1, \quad (4.57a)$$

$$U_k^{(i+1)} = E_k^{(i)} + U_{k|k-1}. \quad (4.57b)$$

Let $\phi = x_k$ and by using equations (4.50) and (4.53), below equation is obtained:

$$\begin{aligned} \log q(x_k)^{(i+1)} &= -\frac{1}{2}(y_k - Cx_k)^T \mathbf{E}[\Sigma_k^{-1}]^{(i+1)}(y_k - Cx_k) \\ &\quad - \frac{1}{2}(x_k - m_{k|k-1})^T P_{k|k-1}^{-1}(x_k - m_{k|k-1}) + c_x, \end{aligned} \quad (4.58)$$

where

$$\mathbf{E}[\Sigma_k^{-1}]^{(i+1)} = (u_k^{(i+1)} - m - 1)(U_k^{(i+1)})^{-1}. \quad (4.59)$$

The likelihood PDF $p(y_k|x_k)^{(i+1)}$ can be expressed as

$$p(y_k|x_k)^{(i+1)} = \mathcal{N}(y_k; Cx_k, \tilde{\Sigma}_k^{(i+1)}), \quad (4.60)$$

where the estimated measurement noise covariance matrix is

$$\tilde{\Sigma}_k^{(i+1)} = (\mathbf{E}[\Sigma_k^{-1}]^{(i+1)})^{-1}. \quad (4.61)$$

$q(x_k)^{(i+1)}$ is updated as Gaussian PDF using equations (4.58), (4.59), (4.60) and (4.61) as

$$q(x_k)^{(i+1)} = \mathcal{N}(x_k; m_{k|k}^{(i+1)}, P_{k|k}^{(i+1)}), \quad (4.62)$$

where

$$m_{k|k}^{(i+1)} = m_{k|k-1} + K_k^{(i+1)}(y_k - Cm_{k|k-1}), \quad (4.63a)$$

$$P_{k|k}^{(i+1)} = P_{k|k-1} - K_k^{(i+1)}CP_{k|k-1}, \quad (4.63b)$$

and

$$K_k^{(i+1)} = P_{k|k-1}C^T \left(CP_{k|k-1}C^T + \tilde{\Sigma}_k^{(i+1)} \right)^{-1}. \quad (4.64)$$

After N iterations, the posterior PDFs become

$$q^*(x_k) \approx q(x_k)^{(N)} = \mathcal{N}(x_k; m_{k|k}^{(N)}, P_{k|k}^{(N)}) = \mathcal{N}(x_k; m_{k|k}, P_{k|k}), \quad (4.65a)$$

$$q^*(\Sigma_k) \approx q(\Sigma_k)^{(N)} = IW(\Sigma_k; u_k^{(N)}, U_k^{(N)}) = IW(\Sigma_k; u_{k|k}, U_{k|k}). \quad (4.65b)$$

VB-IW algorithm is summarized in Algorithm 2.

Algorithm 2 VB-IW Algorithm

Inputs: $m_{k-1|k-1}$, $P_{k-1|k-1}$, $u_{k-1|k-1}$, $U_{k-1|k-1}$, A , C , y_k , Q , R , m , n , λ , N

Outputs: $m_{k|k}$, $P_{k|k}$, $u_{k|k}$, $U_{k|k}$

Time Update:

1. $m_{k|k-1} = Am_{k-1|k-1}$
2. $P_{k|k-1} = AP_{k-1|k-1}A^T + Q$

Measurement Update:

3. Initialization: $m_{k|k}^{(0)} = m_{k|k-1}$, $P_{k|k}^{(0)} = P_{k|k-1}$, $u_{k|k-1} = \lambda(u_{k-1|k-1} - m - 1) + m + 1$, $U_{k|k-1} = \lambda U_{k-1|k-1}$

for $i = 0 : N - 1$ **do**

Update $q(\Sigma_k)^{(i+1)}$ given $q(x_k)^{(i)}$

4. $E_k^{(i)} = (y_k - Cm_{k|k}^{(i)})(y_k - Cm_{k|k}^{(i)})^T + CP_{k|k}^{(i)}C^T$

5. $u_k^{(i+1)} = u_{k|k-1} + 1$, $U_k^{(i+1)} = E_k^{(i)} + U_{k|k-1}$

Update $q(x_k)^{(i+1)}$ given $q(\Sigma_k)^{(i+1)}$

6. $\mathbf{E}[\Sigma_k^{-1}]^{(i+1)} = (u_k^{(i+1)} - m - 1)(U_k^{(i+1)})^{-1}$

7. $\tilde{\Sigma}_k^{(i+1)} = (\mathbf{E}[\Sigma_k^{-1}]^{(i+1)})^{-1}$

8. $K_k^{(i+1)} = P_{k|k-1}C^T \left(CP_{k|k-1}C^T + \tilde{\Sigma}_k^{(i+1)} \right)^{-1}$

9. $m_{k|k}^{(i+1)} = m_{k|k-1} + K_k^{(i+1)}(y_k - Cm_{k|k-1})$

10. $P_{k|k}^{(i+1)} = P_{k|k-1} - K_k^{(i+1)}CP_{k|k-1}$

end for

11. $m_{k|k} = m_{k|k}^{(N)}$, $P_{k|k} = P_{k|k}^{(N)}$, $u_{k|k} = u_{k|k}^{(N)}$, $U_{k|k} = U_{k|k}^{(N)}$

VB-IW approach expressed in this section is proposed in [23] for linear Gaussian SSMs with unknown process and measurement noise. In this thesis, this approach is proposed for linear SSMs with heavy-tailed measurement noise and it is compared with VB-GG approach in terms of robustness towards outliers.

In contrast to VB-GG, VB-IW approach uses the sufficient statistics of the noise terms as the prior in the next update. If we remove this carry over and fix the noise priors, the algorithms exhibit similar performances. We refer to this algorithm as fixed prior VB-IW. Fixed prior VB-IW algorithm is summarized in Algorithm 3.

Algorithm 3 Fixed Prior VB-IW Algorithm

Inputs: $m_{k-1|k-1}, P_{k-1|k-1}, A, C, y_k, Q, R, m, n, \lambda, N, \tau$

Outputs: $m_{k|k}, P_{k|k}$

Time Update:

1. $m_{k|k-1} = Am_{k-1|k-1}$
2. $P_{k|k-1} = AP_{k-1|k-1}A^T + Q$

Measurement Update:

3. Initialization: $m_{k|k}^{(0)} = m_{k|k-1}, P_{k|k}^{(0)} = P_{k|k-1}, u_k = \tau + m + 1, U_k = \tau R$

for $i = 0 : N - 1$ **do**

Update $q(\Sigma_k)^{(i+1)}$ given $q(x_k)^{(i)}$

4. $E_k^{(i)} = (y_k - Cm_{k|k}^{(i)})(y_k - Cm_{k|k}^{(i)})^T + CP_{k|k}^{(i)}C^T$

5. $u_k^{(i+1)} = u_k + 1, U_k^{(i+1)} = E_k^{(i)} + U_k$

Update $q(x_k)^{(i+1)}$ given $q(\Sigma_k)^{(i+1)}$

6. $\mathbf{E}[\Sigma_k^{-1}]^{(i+1)} = (u_k^{(i+1)} - m - 1)(U_k^{(i+1)})^{-1}$

7. $\tilde{\Sigma}_k^{(i+1)} = (\mathbf{E}[\Sigma_k^{-1}]^{(i+1)})^{-1}$

8. $K_k^{(i+1)} = P_{k|k-1}C^T \left(CP_{k|k-1}C^T + \tilde{\Sigma}_k^{(i+1)} \right)^{-1}$

9. $m_{k|k}^{(i+1)} = m_{k|k-1} + K_k^{(i+1)}(y_k - Cm_{k|k-1})$

10. $P_{k|k}^{(i+1)} = P_{k|k-1} - K_k^{(i+1)}CP_{k|k-1}$

end for

11. $m_{k|k} = m_{k|k}^{(N)}, P_{k|k} = P_{k|k}^{(N)}$
-

4.3 Performance Evaluation

In this section, VB-GG and VB-IW algorithms are tested on a simulation of a moving target in 2-D space that have Gaussian process noise and heavy-tailed measurement noise. These tests are conducted for five different characteristics of measurement noise. The CV model [8] is used to model the motion of the target. The state is the same with (3.41). The system matrices in (2.5) are

$$A = \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix}, C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}, \quad (4.66)$$

where $T = 1s$ is the sampling time. The process noise is distributed with normal distribution $w_k \sim \mathcal{N}(0, Q)$ where

$$Q = \begin{bmatrix} \frac{T^3}{3} & \frac{T^2}{2} & 0 & 0 \\ \frac{T^2}{2} & T & 0 & 0 \\ 0 & 0 & \frac{T^3}{3} & \frac{T^2}{2} \\ 0 & 0 & \frac{T^2}{2} & T \end{bmatrix}. \quad (4.67)$$

We consider five different cases for measurement noise generation. These cases are given below.

Case 1:

The measurement noise is generated according to

$$v_k \sim \begin{cases} \mathcal{N}(0, R_0) & \text{w.p. } p_0 \\ \mathcal{N}(\mu_1, R_1) & \text{w.p. } p_1 \\ \mathcal{N}(\mu_2, R_2) & \text{w.p. } p_2 \end{cases} \quad (4.68)$$

where $p_0 = 0.4$, $p_1 = 0.3$, $p_2 = 0.4$, $R_0 = 100 \times I_2$, $R_1 = R_2 = 50 \times I_2$, $\mu_1 = [70 \ 70]^T$ and $\mu_2 = [-70 \ -70]^T$. The nominal covariance, outlier covariances and

the effective covariance is shown in Figure 4.1.

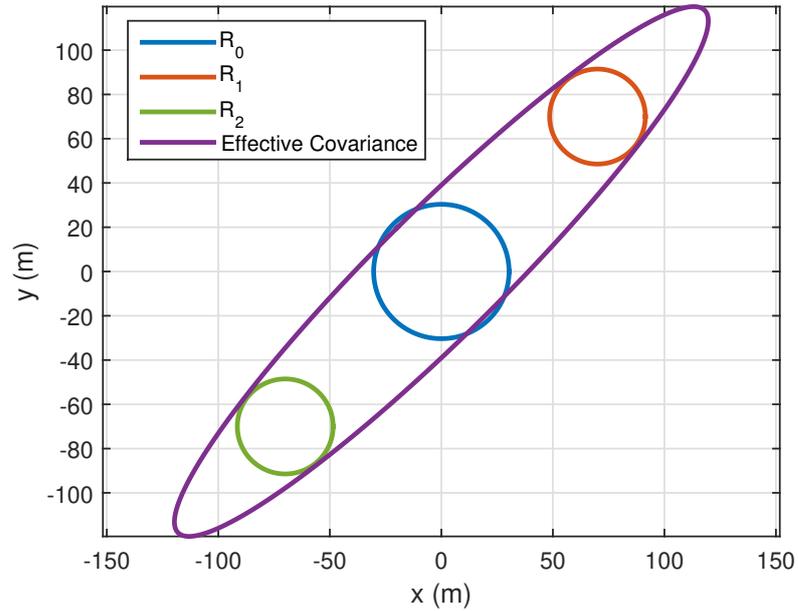


Figure 4.1: The nominal covariance, outlier covariances and the effective covariance for Case 1. The nominal covariance R_0 is shown by blue circle, the outlier covariance R_1 is shown by orange circle, the outlier covariance R_2 is shown by green circle and the effective covariance is shown by purple ellipse.

Case 2:

The measurement noise is generated according to (4.68) where $p_0 = 0.4$, $p_1 = 0.3$, $p_2 = 0.4$, $R_0 = 100 \times I_2$, $R_1 = R_2 = 50 \times I_2$, $\mu_1 = [0 \ 100]^T$ and $\mu_2 = [0 \ -100]^T$. The nominal covariance, outlier covariances and the effective covariance is shown in Figure 4.2.

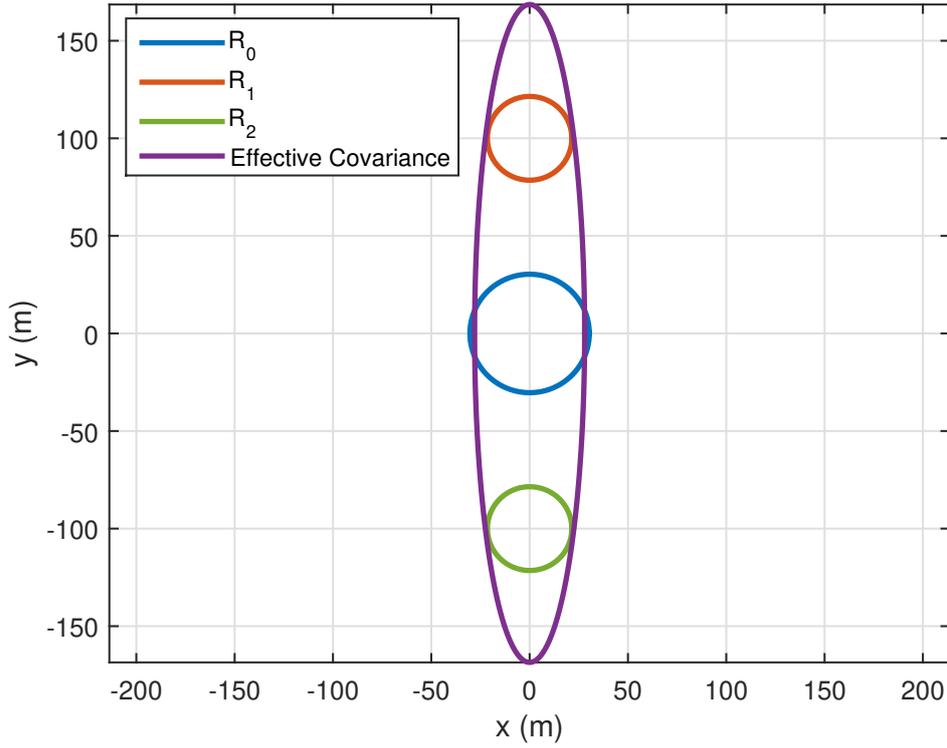


Figure 4.2: The nominal covariance, outlier covariances and the effective covariance for Case 2. The nominal covariance R_0 is shown by blue circle, the outlier covariance R_1 is shown by orange circle, the outlier covariance R_2 is shown by green circle and the effective covariance is shown by purple ellipse.

Case 3:

The measurement noise is generated according to

$$v_k \sim \begin{cases} \mathcal{N}(0, R_0) & \text{w.p. } 0.9 \\ \mathcal{N}(0, 100R_0) & \text{w.p. } 0.1 \end{cases} \quad (4.69)$$

where $R_0 = 100 \times I_2$. The nominal covariance, the outlier covariance and the effective covariance is shown in Figure 4.3.

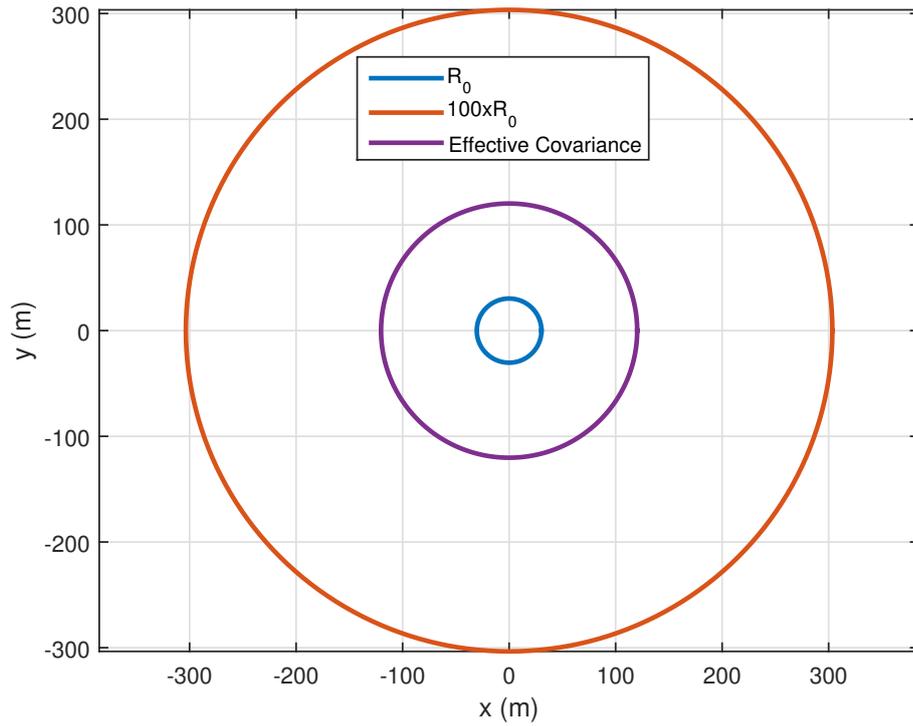


Figure 4.3: The nominal covariance, the outlier covariance and the effective covariance for Case 3. The nominal covariance R_0 is shown by blue circle, the outlier covariance $100 \times R_0$ is shown by orange circle and the effective covariance is shown by purple circle.

Case 4:

The measurement noise is generated according to (4.68) where $p_0 = 0.8$, $p_1 = 0.1$, $p_2 = 0.1$, $R_0 = 100 \times I_2$, $R_1 = R_2 = 50 \times I_2$, $\mu_1 = [1000 \ 1000]^T$ and $\mu_2 = [-1000 \ -1000]^T$. The nominal covariance, outlier covariances and the effective covariance is shown in Figure 4.4.

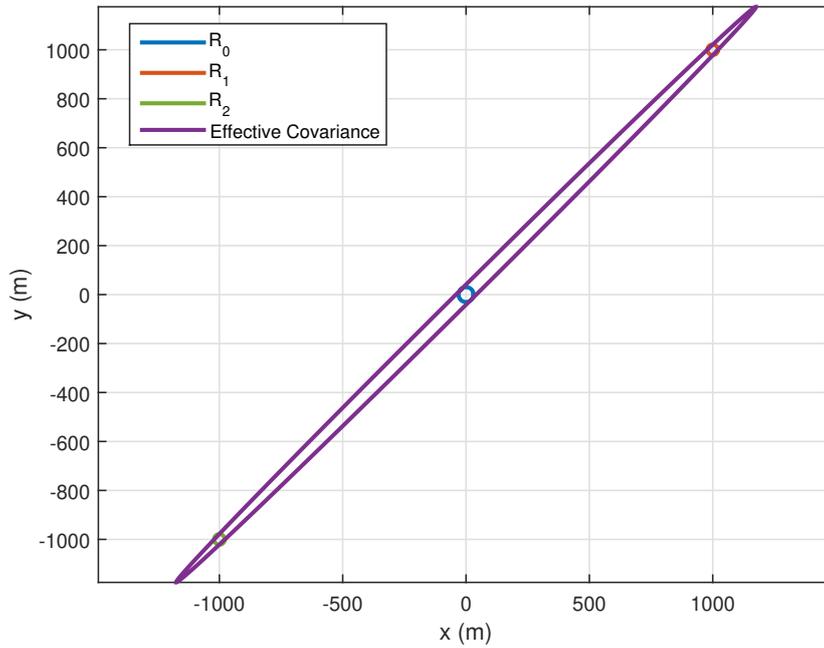


Figure 4.4: The nominal covariance, outlier covariances and the effective covariance for Case 4. The nominal covariance R_0 is shown by blue circle, the outlier covariance R_1 is shown by orange circle, the outlier covariance R_2 is shown by green circle and the effective covariance is shown by purple ellipse.

Case 5:

The measurement noise is generated according to (4.68) where $p_0 = 0.8$, $p_1 = 0.1$, $p_2 = 0.1$, $R_0 = 100 \times I_2$, $R_1 = R_2 = 50 \times I_2$, $\mu_1 = [0 \ 1000]^T$ and $\mu_2 = [0 \ -1000]^T$. The nominal covariance, outlier covariances and the effective covariance is shown in Figure 4.5.

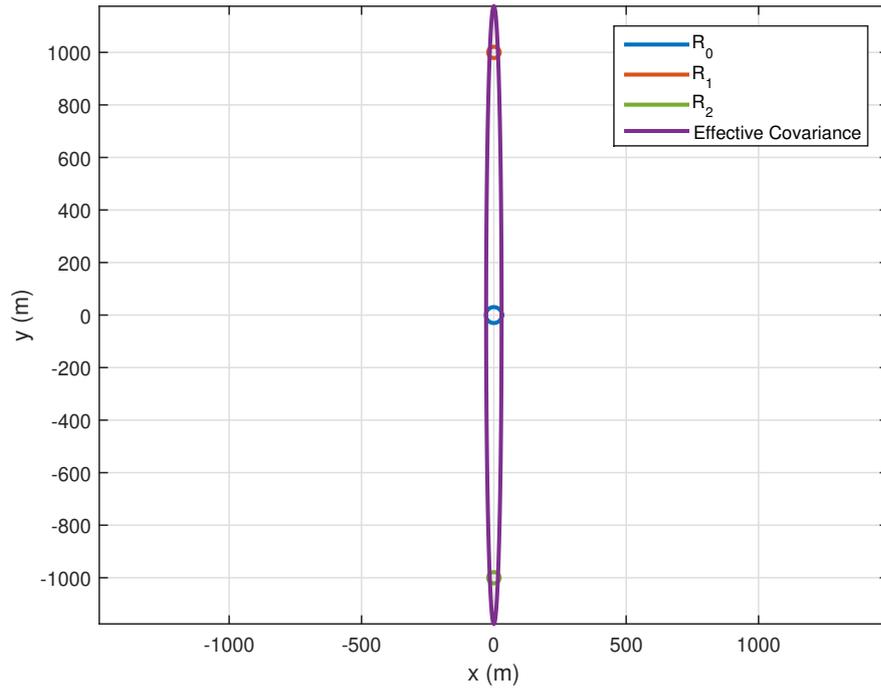


Figure 4.5: The nominal covariance, outlier covariances and the effective covariance for Case 5. The nominal covariance R_0 is shown by blue circle, the outlier covariance R_1 is shown by orange circle, the outlier covariance R_2 is shown by green circle and the effective covariance is shown by purple ellipse.

In this simulation, the position, the velocity and the effective covariance of the measurement noise are estimated with both VB-GG algorithm and VB-IW algorithm. The algorithms are only aware of the nominal measurement noise covariance R_0 for all cases. The simulation is carried out for 500 Monte Carlo (MC) runs, 500 time steps for each MC run and 10 iterations are performed for both VB algorithms. In addition, the degrees of freedom parameter ν in VB-GG algorithm is taken as 3 [22]. The tuning parameter τ and the forgetting factor λ in VB-IW algorithm are taken as 5 and 1 [23], respectively. For evaluating the estimation accuracy of the state, root mean square error (RMSE) and averaged root mean square error (ARMSE) [22] given in equations (3.16) and (3.17) are used as performance metrics. These metrics are computed for both position and velocity.

Figure 4.6 and Figure 4.7 shows respectively the RMSEs of position and velocity for Case 1. The ARMSEs of position and velocity for VB-GG, VB-IW, STF and KF for Case 1 are given in Table 4.1. In Figure 4.8, the effective measurement noise covariance, the estimated measurement noise covariance by VB-IW and the effective covariance calculated by VB-GG used in the measurement update of the state in VB iterations for Case 1 are shown for any time interval of a single MC run.

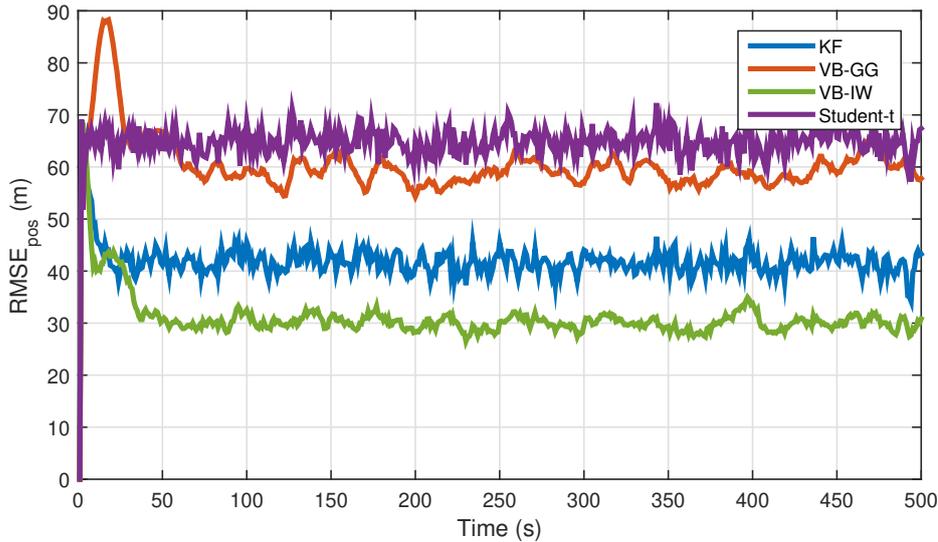


Figure 4.6: RMSEs of the position for 500 Monte Carlo run for Case 1. RMSEs of the position by VB-GG is shown by orange line, RMSEs of the position by VB-IW is shown by green line, RMSEs of the position by STF is shown by purple line and RMSEs of the position by KF is shown by blue line.

Figure 4.9 and Figure 4.10 shows respectively the RMSEs of position and velocity for Case 2. The ARMSEs of position and velocity for VB-GG, VB-IW, STF and KF for Case 2 are given in Table 4.2. In Figure 4.11, the effective measurement noise covariance, the estimated measurement noise covariance by VB-IW and the effective covariance calculated by VB-GG used in the measurement update of the state in VB iterations for Case 2 are shown for any time interval of a single MC run.

Figure 4.12 and Figure 4.13 shows respectively the RMSEs of position and velocity for Case 3. The ARMSEs of position and velocity for VB-GG, VB-IW, STF and KF for Case 3 are given in Table 4.3. In Figure 4.14, the effective measurement noise covariance, the estimated measurement noise covariance by VB-IW and the effective

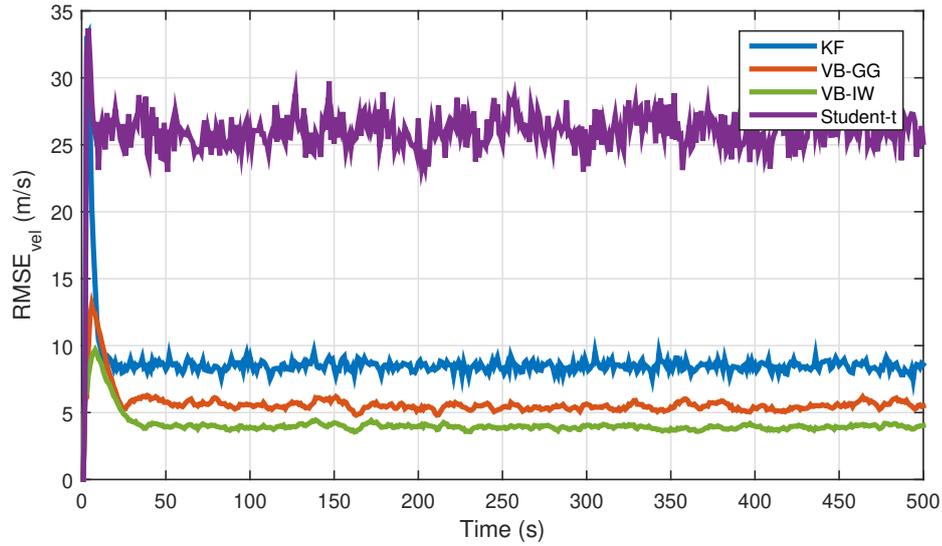


Figure 4.7: RMSEs of the velocity for 500 Monte Carlo run for Case 1. RMSEs of the velocity by VB-GG is shown by orange line, RMSEs of the velocity by VB-IW is shown by green line, RMSEs of the velocity by STF is shown by purple line and RMSEs of the velocity by KF is shown by blue line.

covariance calculated by VB-GG used in the measurement update of the state in VB iterations for Case 3 are shown for any time interval of a single MC run.

Figure 4.15 and Figure 4.16 shows respectively the RMSEs of position and velocity for Case 4. The ARMSEs of position and velocity for VB-GG, VB-IW, STF and KF for Case 4 are given in Table 4.4. In Figure 4.17, the effective measurement noise covariance, the estimated measurement noise covariance by VB-IW and the effective covariance calculated by VB-GG used in the measurement update of the state in VB iterations for Case 4 are shown for any time interval of a single MC run.

Figure 4.18 and Figure 4.19 shows respectively the RMSEs of position and velocity for Case 5. The ARMSEs of position and velocity for VB-GG, VB-IW, STF and KF for Case 5 are given in Table 4.5. In Figure 4.20, the effective measurement noise covariance, the estimated measurement noise covariance by VB-IW and the effective covariance calculated by VB-GG used in the measurement update of the state in VB iterations for Case 5 are shown for any time interval of a single MC run.

Figures 4.6, 4.7, 4.9, 4.10 and Tables 4.1, 4.2 show that VB-IW algorithm provides

Table 4.1: ARMSEs of KF, VB-GG, VB-IW and STF for Case 1

	ARMSE of the position (m)	ARMSE of the velocity (m/s)
KF	42.33	8.90
VB-GG	61.94	5.77
VB-IW	31.86	4.15
STF	64.82	26.07

Table 4.2: ARMSEs of KF, VB-GG, VB-IW and STF for Case 2

	ARMSE of the position (m)	ARMSE of the velocity (m/s)
KF	42.60	9.02
VB-GG	61.23	5.74
VB-IW	31.45	4.17
STF	65.41	26.25

better estimation accuracy than other algorithms for Case 1 and Case 2 since in these cases, outliers near to nominal covariance and they have high probability of occurrence. Therefore, these cases show behavior of unknown covariance instead of outlier behavior. As seen in Figures 4.8 and 4.11, VB-IW algorithm estimates the effective measurement noise covariances accurately. As a result, VB-IW algorithm outperforms VB-GG algorithm for the systems that have unknown measurement noise covariances in terms of RMSE and ARMSE.

Unlike Case 1 and 2, Case 3-5 show outlier behaviour. In these cases, VB-GG algorithm has smaller RMSEs and ARMSEs than VB-IW algorithm as seen in Figures 4.12, 4.13, 4.15, 4.16, 4.18, 4.19 and Tables 4.3, 4.4, 4.5. In order to increase the estimation accuracy of VB-IW algorithm for the systems that have outliers, we use fixed prior VB-IW algorithm and the simulation results for this algorithm are given.

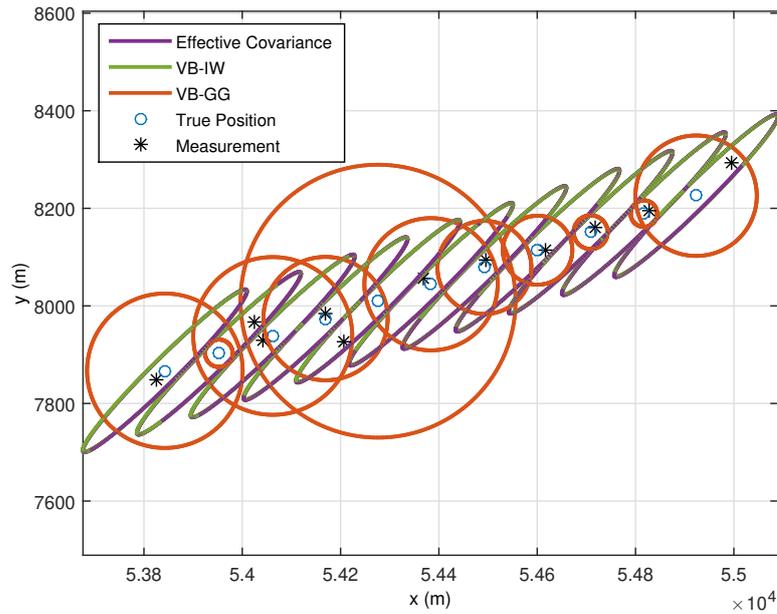


Figure 4.8: Measurement noise covariance tracking of the algorithms VB-GG and VB-IW for Case 1 (for any time interval of a single MC run). The effective measurement noise covariances are shown by purple ellipses, the estimated measurement noise covariances by VB-IW are shown by green ellipses and the effective covariance calculated by VB-GG used in the measurement update of the state in VB iterations are shown by orange circles. The blue circles and black stars show the true positions and the measurements, respectively.

Figure 4.21 and Figure 4.22 shows respectively the RMSEs of position and velocity for Case 3 by using fixed prior VB-IW algorithm. The ARMSEs of position and velocity for VB-GG, fixed prior VB-IW, STF and KF for Case 3 are given in Table 4.6. In Figure 4.23, the effective measurement noise covariance, the effective covariance calculated by fixed prior VB-IW used in the measurement update of the state in VB iterations and the effective covariance calculated by VB-GG used in the measurement update of the state in VB iterations for Case 3 are shown for any time interval of a single MC run.

Figure 4.24 and Figure 4.25 shows respectively the RMSEs of position and velocity for Case 4 by using fixed prior VB-IW algorithm. The ARMSEs of position and velocity for VB-GG, fixed prior VB-IW, STF and KF for Case 4 are given in Table 4.7.

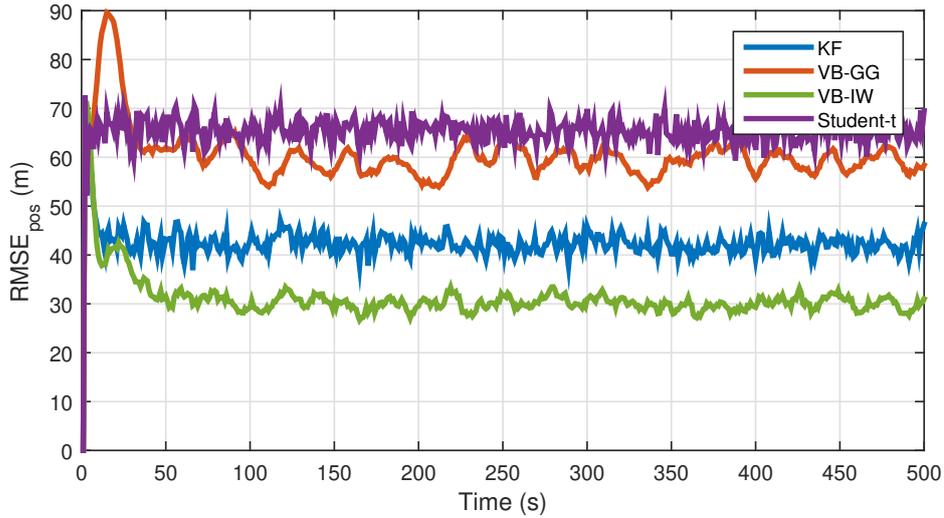


Figure 4.9: RMSEs of the position for 500 Monte Carlo run for Case 2. RMSEs of the position by VB-GG is shown by orange line, RMSEs of the position by VB-IW is shown by green line, RMSEs of the position by STF is shown by purple line and RMSEs of the position by KF is shown by blue line.

In Figure 4.26, the effective measurement noise covariance, the effective covariance calculated by fixed prior VB-IW used in the measurement update of the state in VB iterations and the effective covariance calculated by VB-GG used in the measurement update of the state in VB iterations for Case 4 are shown for any time interval of a single MC run.

Figure 4.27 and Figure 4.28 shows respectively the RMSEs of position and velocity for Case 5 by using fixed prior VB-IW algorithm. The ARMSEs of position and velocity for VB-GG, fixed prior VB-IW, STF and KF for Case 5 are given in Table 4.8. In Figure 4.29, the effective measurement noise covariance, the effective covariance calculated by fixed prior VB-IW used in the measurement update of the state in VB iterations and the effective covariance calculated by VB-GG used in the measurement update of the state in VB iterations for Case 5 are shown for any time interval of a single MC run.

As seen in Figures 4.21, 4.22, 4.24, 4.25, 4.27 and 4.28, the RMSEs of VB-IW algorithm approaches to RMSEs of VB-GG algorithm, i.e., both VB-IW and VB-GG provide almost the same estimation accuracy, by applying proposed approach for the

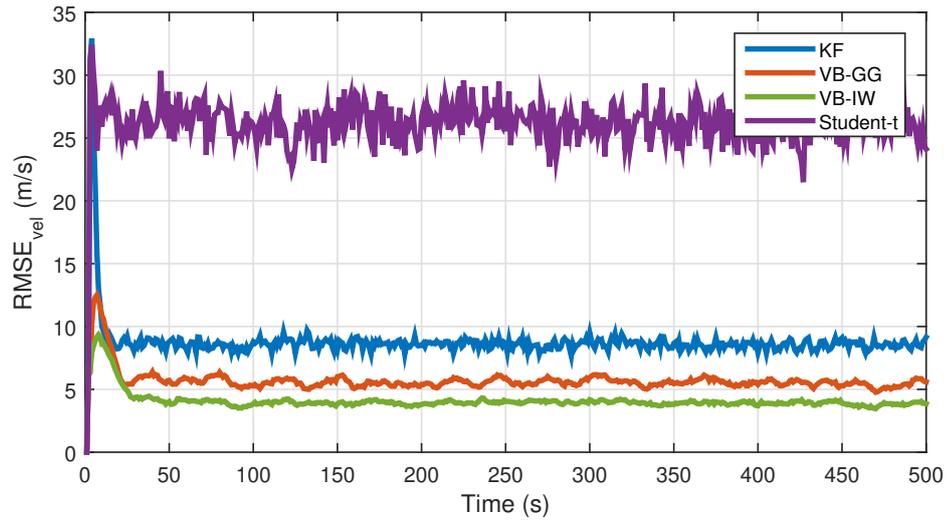


Figure 4.10: RMSEs of the velocity for 500 Monte Carlo run for Case 2. RMSEs of the velocity by VB-GG is shown by orange line, RMSEs of the velocity by VB-IW is shown by green line, RMSEs of the velocity by STF is shown by purple line and RMSEs of the velocity by KF is shown by blue line.

systems that have outliers. Consequently, VB-IW algorithm approaches to VB-GG algorithm when the presence of outliers. It can also outperforms VB-GG algorithm under certain circumstances.

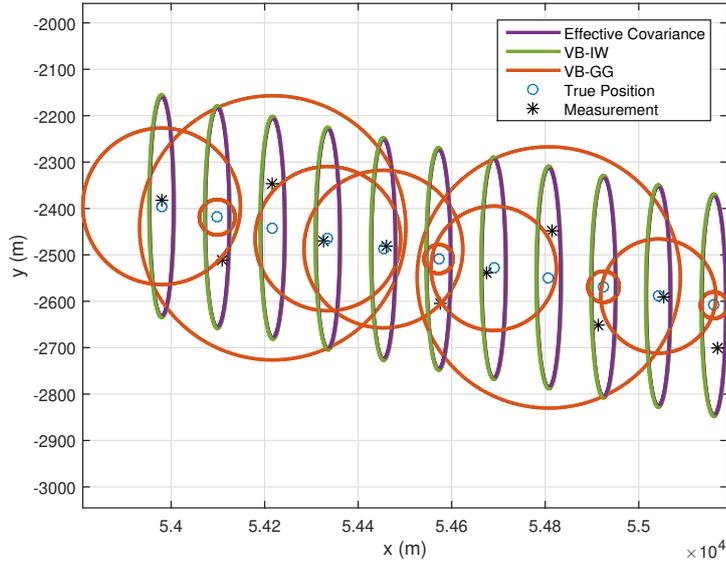


Figure 4.11: Measurement noise covariance tracking of the algorithms VB-GG and VB-IW for Case 2 (for any time interval of a single MC run). The effective measurement noise covariances are shown by purple ellipses, the estimated measurement noise covariances by VB-IW are shown by green ellipses and the effective covariance calculated by VB-GG used in the measurement update of the state in VB iterations are shown by orange circles. The blue circles and black stars show the true positions and the measurements, respectively.

Table 4.3: ARMSEs of KF, VB-GG, VB-IW and STF for Case 3

	ARMSE of the position (m)	ARMSE of the velocity (m/s)
KF	25.42	5.74
VB-GG	9.83	3.04
VB-IW	20.72	3.90
STF	19.49	6.37

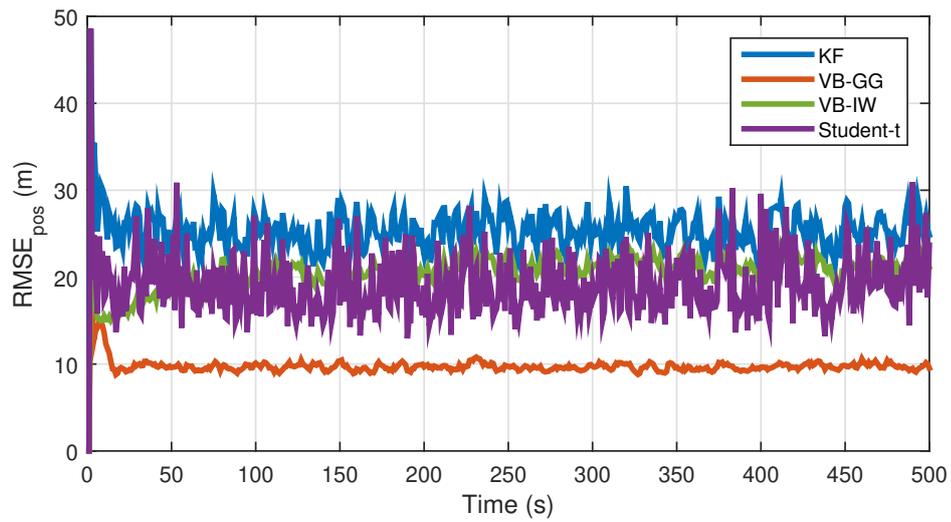


Figure 4.12: RMSEs of the position for 500 Monte Carlo run for Case 3. RMSEs of the position by VB-GG is shown by orange line, RMSEs of the position by VB-IW is shown by green line, RMSEs of the position by STF is shown by purple line and RMSEs of the position by KF is shown by blue line.

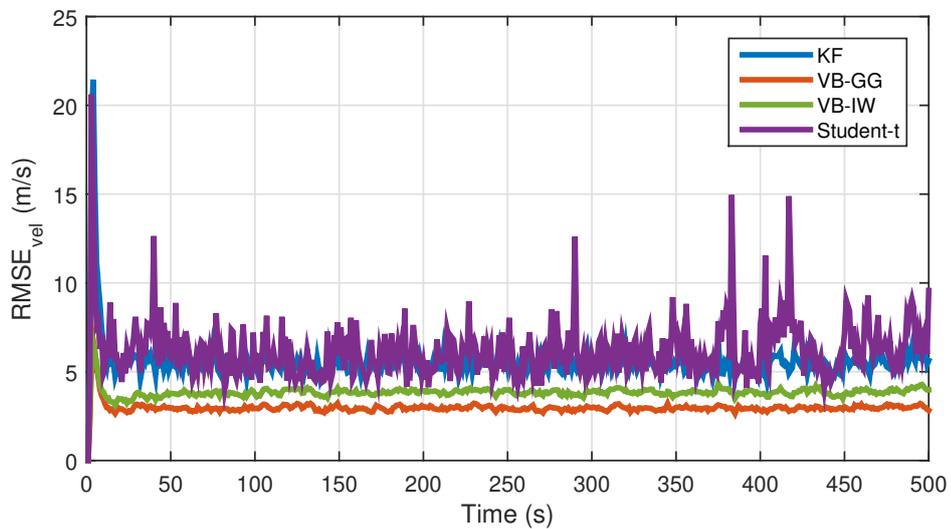


Figure 4.13: RMSEs of the velocity for 500 Monte Carlo run for Case 3. RMSEs of the velocity by VB-GG is shown by orange line, RMSEs of the velocity by VB-IW is shown by green line, RMSEs of the velocity by STF is shown by purple line and RMSEs of the velocity by KF is shown by blue line.

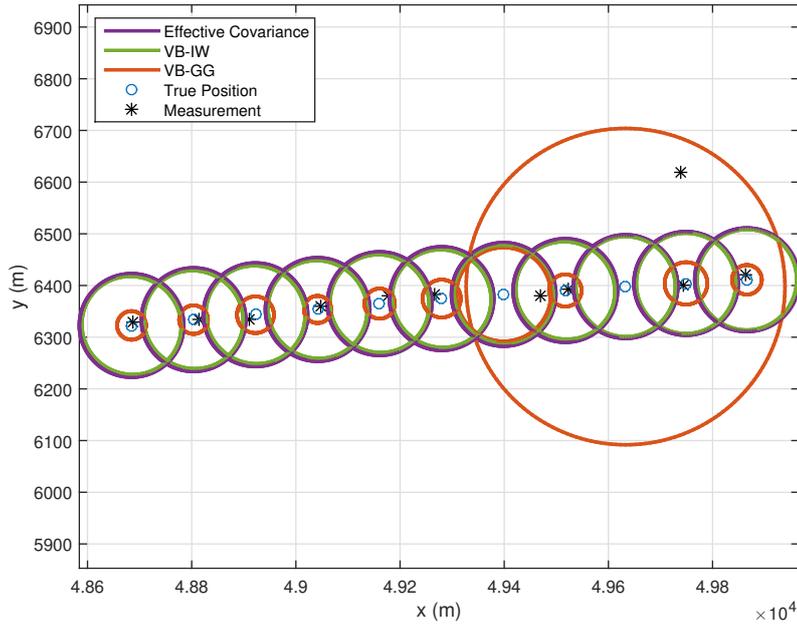


Figure 4.14: Measurement noise covariance tracking of the algorithms VB-GG and VB-IW for Case 3 (for any time interval of a single MC run). The effective measurement noise covariances are shown by purple ellipses, the estimated measurement noise covariances by VB-IW are shown by green ellipses and the effective covariance calculated by VB-GG used in the measurement update of the state in VB iterations are shown by orange circles. The blue circles and black stars show the true positions and the measurements, respectively.

Table 4.4: ARMSEs of KF, VB-GG, VB-IW and STF for Case 4

	ARMSE of the position (m)	ARMSE of the velocity (m/s)
KF	243.42	49.79
VB-GG	9.54	2.97
VB-IW	109.94	5.65
STF	242.59	205.62

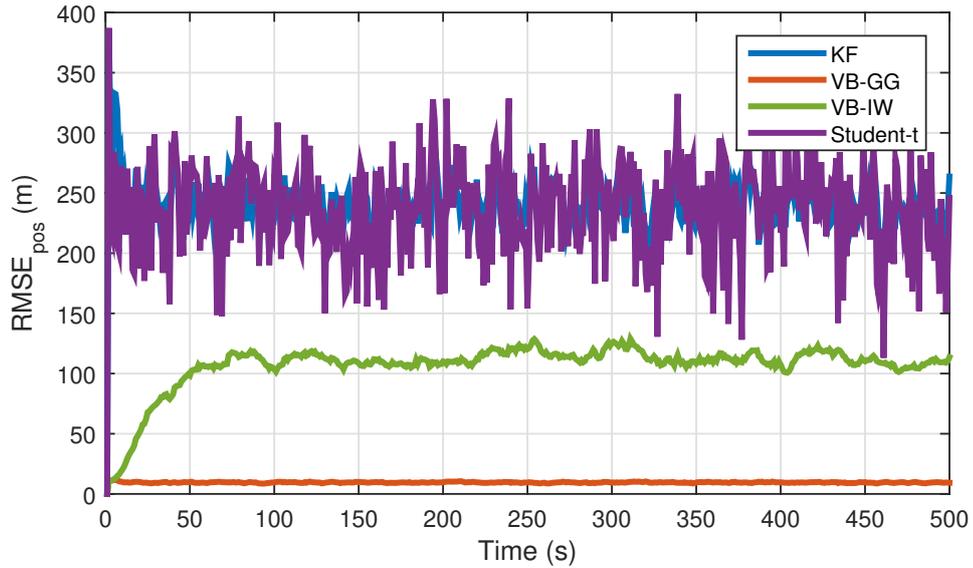


Figure 4.15: RMSEs of the position for 500 Monte Carlo run for Case 4. RMSEs of the position by VB-GG is shown by orange line, RMSEs of the position by VB-IW is shown by green line, RMSEs of the position by STF is shown by purple line and RMSEs of the position by KF is shown by blue line.

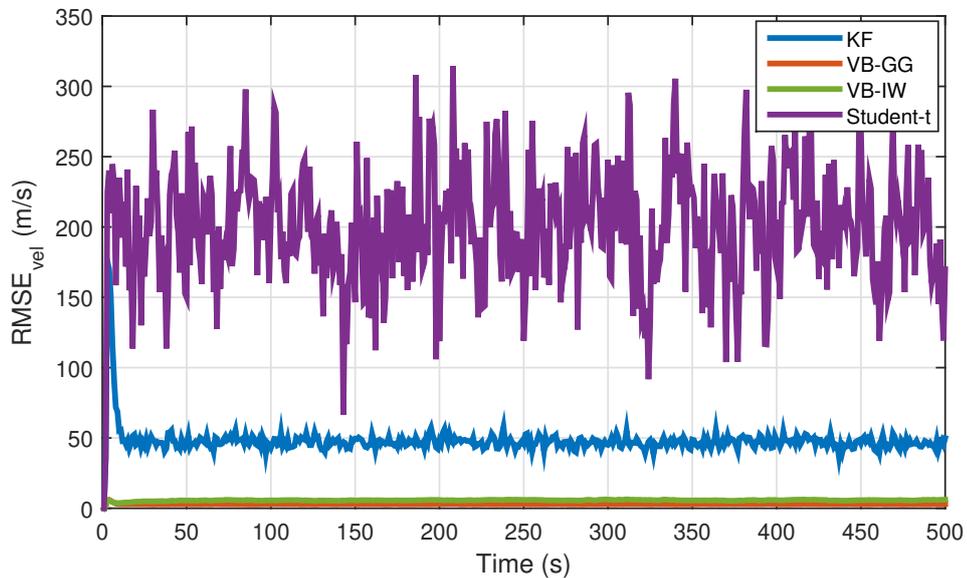


Figure 4.16: RMSEs of the velocity for 500 Monte Carlo run for Case 4. RMSEs of the velocity by VB-GG is shown by orange line, RMSEs of the velocity by VB-IW is shown by green line, RMSEs of the velocity by STF is shown by purple line and RMSEs of the velocity by KF is shown by blue line.

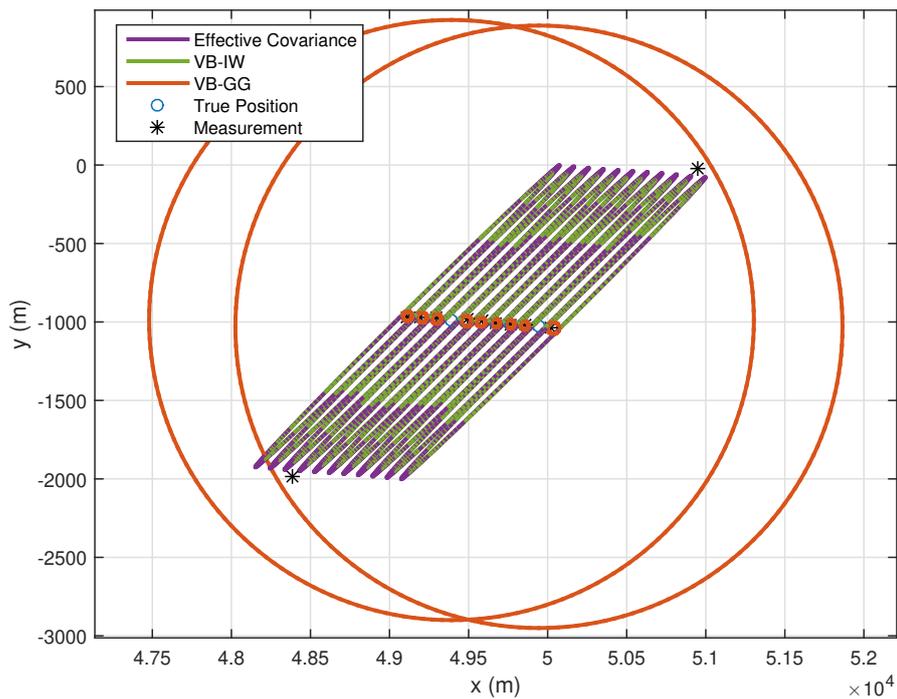


Figure 4.17: Measurement noise covariance tracking of the algorithms VB-GG and VB-IW for Case 4 (for any time interval of a single MC run). The effective measurement noise covariances are shown by purple ellipses, the estimated measurement noise covariances by VB-IW are shown by green ellipses and the effective covariance calculated by VB-GG used in the measurement update of the state in VB iterations are shown by orange circles. The blue circles and black stars show the true positions and the measurements, respectively.

Table 4.5: ARMSEs of KF, VB-GG, VB-IW and STF for Case 5

	ARMSE of the position (m)	ARMSE of the velocity (m/s)
KF	169.91	34.82
VB-GG	9.55	2.97
VB-IW	83.22	5.27
STF	165.73	132.32

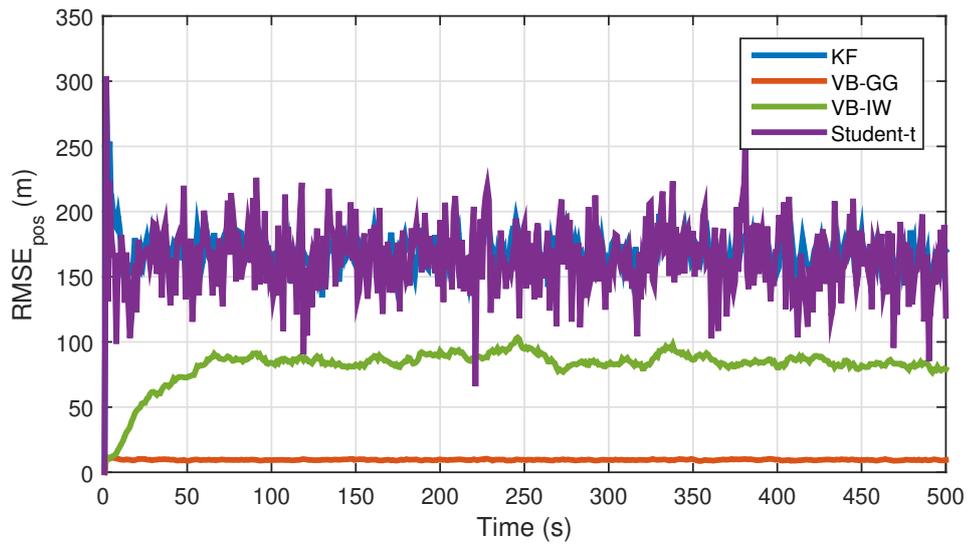


Figure 4.18: RMSEs of the position for 500 Monte Carlo run for Case 5. RMSEs of the position by VB-GG is shown by orange line, RMSEs of the position by VB-IW is shown by green line, RMSEs of the position by STF is shown by purple line and RMSEs of the position by KF is shown by blue line.

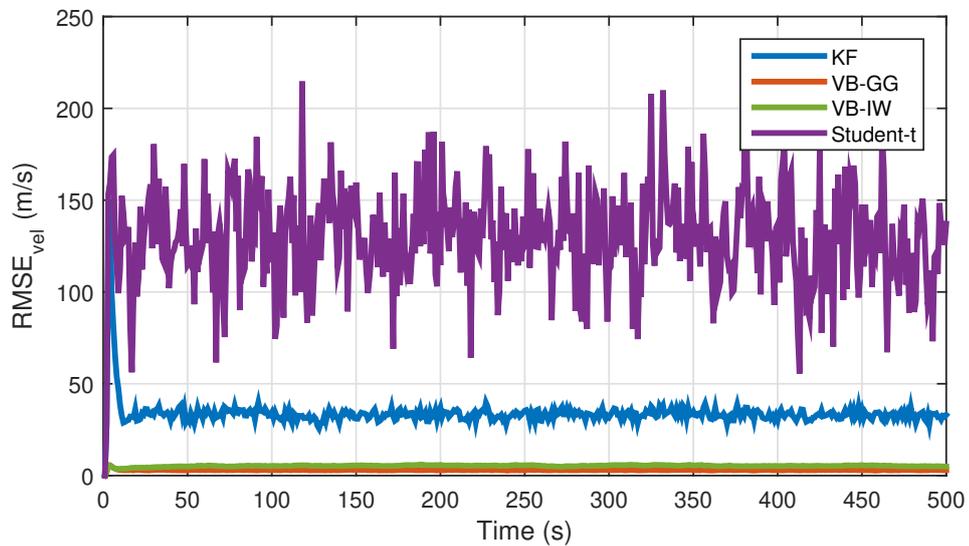


Figure 4.19: RMSEs of the velocity for 500 Monte Carlo run for Case 5. RMSEs of the velocity by VB-GG is shown by orange line, RMSEs of the velocity by VB-IW is shown by green line, RMSEs of the velocity by STF is shown by purple line and RMSEs of the velocity by KF is shown by blue line.

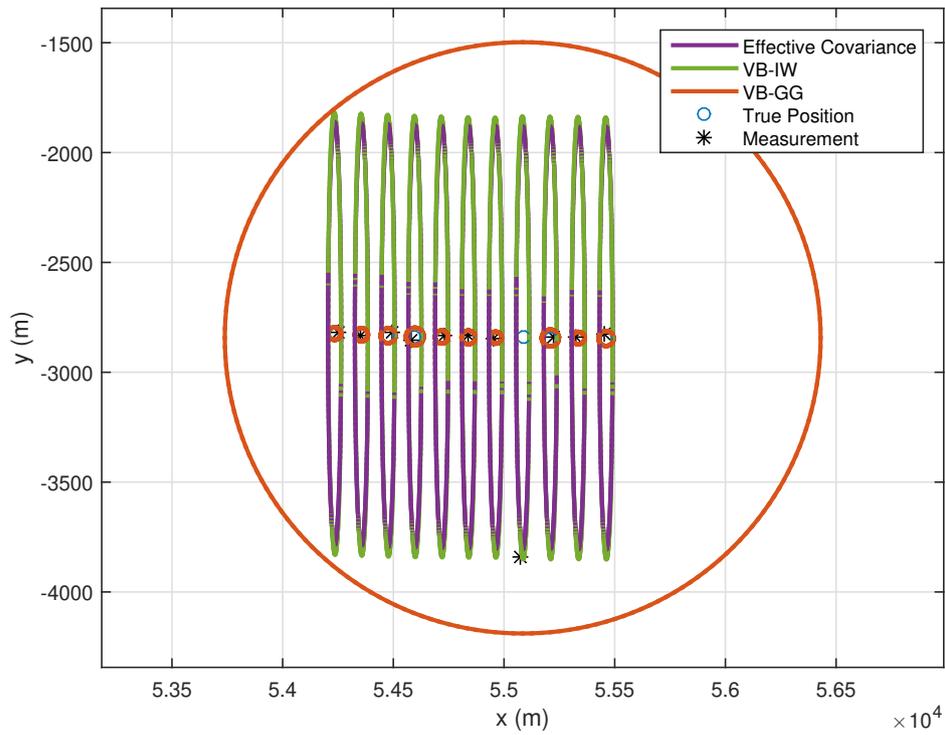


Figure 4.20: Measurement noise covariance tracking of the algorithms VB-GG and VB-IW for Case 5 (for any time interval of a single MC run). The effective measurement noise covariances are shown by purple ellipses, the estimated measurement noise covariances by VB-IW are shown by green ellipses and the effective covariance calculated by VB-GG used in the measurement update of the state in VB iterations are shown by orange circles. The blue circles and black stars show the true positions and the measurements, respectively.

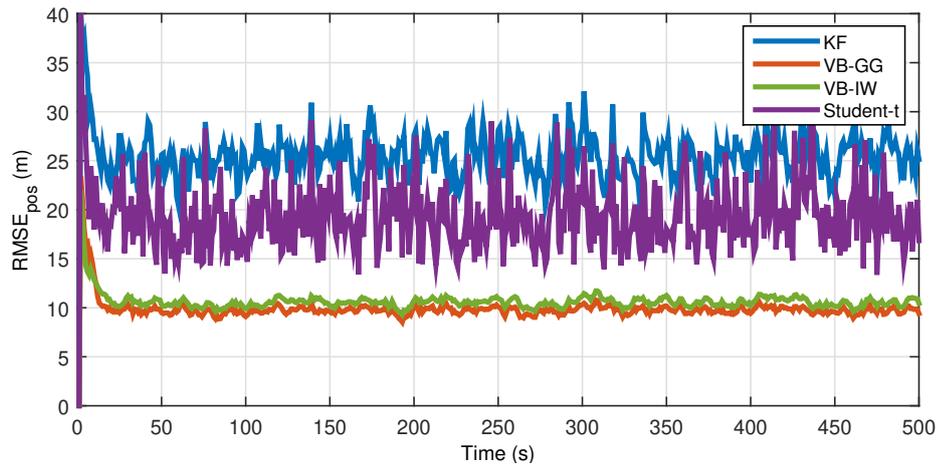


Figure 4.21: RMSEs of the position for 500 Monte Carlo run for Case 3 by using fixed prior VB-IW algorithm. RMSEs of the position by VB-GG is shown by orange line, RMSEs of the position by fixed prior VB-IW is shown by green line, RMSEs of the position by STF is shown by purple line and RMSEs of the position by KF is shown by blue line.

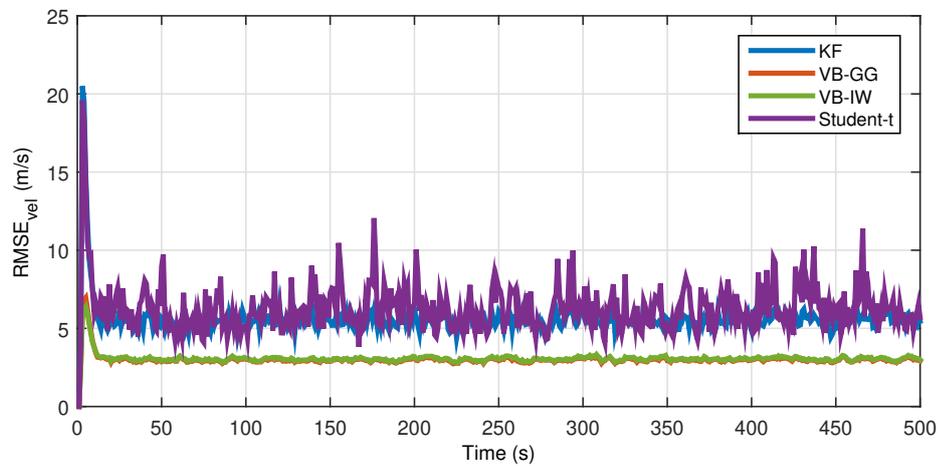


Figure 4.22: RMSEs of the velocity for 500 Monte Carlo run for Case 3 by using fixed prior VB-IW algorithm. RMSEs of the velocity by VB-GG is shown by orange line, RMSEs of the velocity by fixed prior VB-IW is shown by green line, RMSEs of the velocity by STF is shown by purple line and RMSEs of the velocity by KF is shown by blue line.

Table 4.6: ARMSEs of KF, VB-GG, fixed prior VB-IW and STF for Case 3

	ARMSE of the position (m)	ARMSE of the velocity (m/s)
KF	25.63	5.71
VB-GG	9.80	3.00
Fixed Prior VB-IW	10.05	3.06
STF	19.65	6.43

Table 4.7: ARMSEs of KF, VB-GG, fixed prior VB-IW and STF for Case 4

	ARMSE of the position (m)	ARMSE of the velocity (m/s)
KF	242.54	49.56
VB-GG	9.52	2.96
Fixed Prior VB-IW	9.85	3.03
STF	241.20	204.33

Table 4.8: ARMSEs of KF, VB-GG, fixed prior VB-IW and STF for Case 5

	ARMSE of the position (m)	ARMSE of the velocity (m/s)
KF	171.91	35.05
VB-GG	9.55	2.96
Fixed Prior VB-IW	9.94	3.03
STF	164.67	130.50

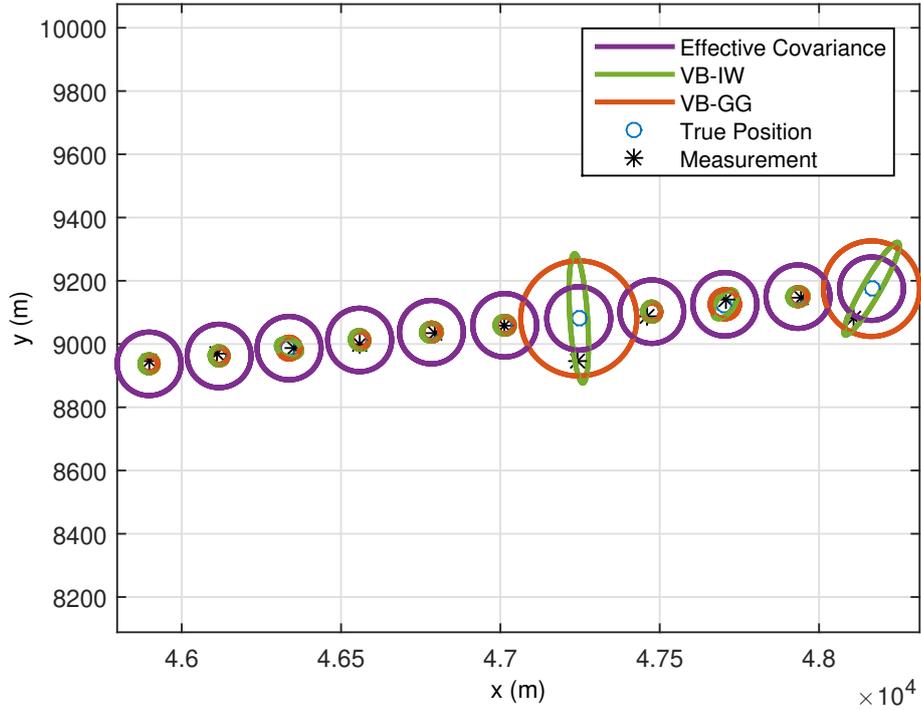


Figure 4.23: Measurement noise covariance tracking of the algorithms VB-GG and fixed prior VB-IW for Case 3 (for any time interval of a single MC run). The effective measurement noise covariances are shown by purple ellipses, the effective covariance calculated by fixed prior VB-IW used in the measurement update of the state in VB iterations are shown by green ellipses and the effective covariance calculated by VB-GG used in the measurement update of the state in VB iterations are shown by orange circles. The blue circles and black stars show the true positions and the measurements, respectively.

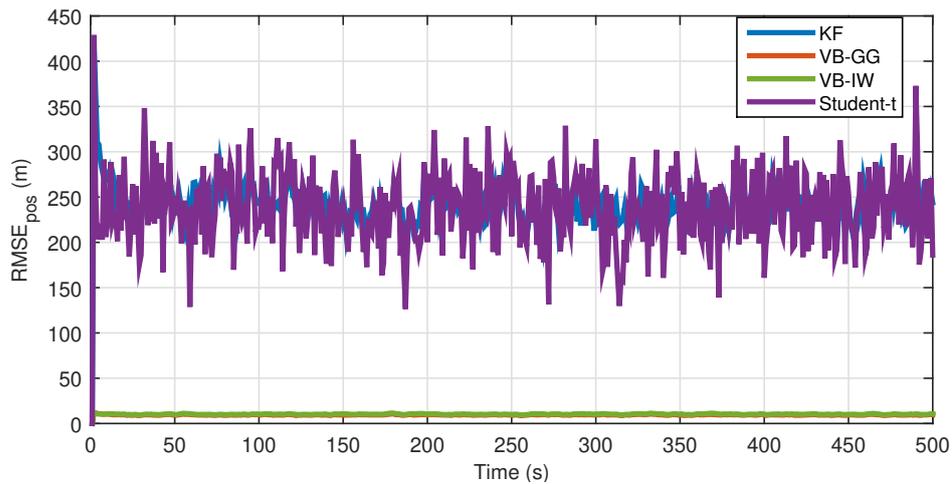


Figure 4.24: RMSEs of the position for 500 Monte Carlo run for Case 4 by using fixed prior VB-IW algorithm. RMSEs of the position by VB-GG is shown by orange line, RMSEs of the position by fixed prior VB-IW is shown by green line, RMSEs of the position by STF is shown by purple line and RMSEs of the position by KF is shown by blue line.

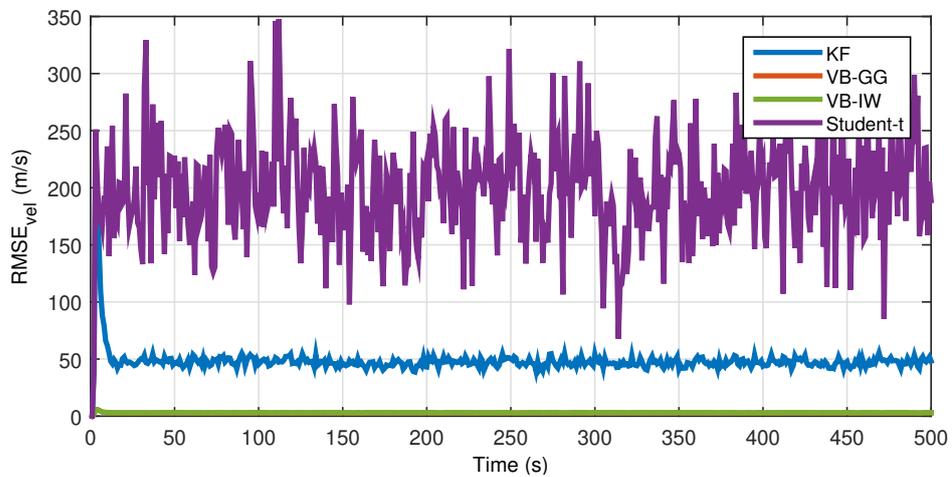


Figure 4.25: RMSEs of the velocity for 500 Monte Carlo run for Case 4 by using fixed prior VB-IW algorithm. RMSEs of the velocity by VB-GG is shown by orange line, RMSEs of the velocity by fixed prior VB-IW is shown by green line, RMSEs of the velocity by STF is shown by purple line and RMSEs of the velocity by KF is shown by blue line.

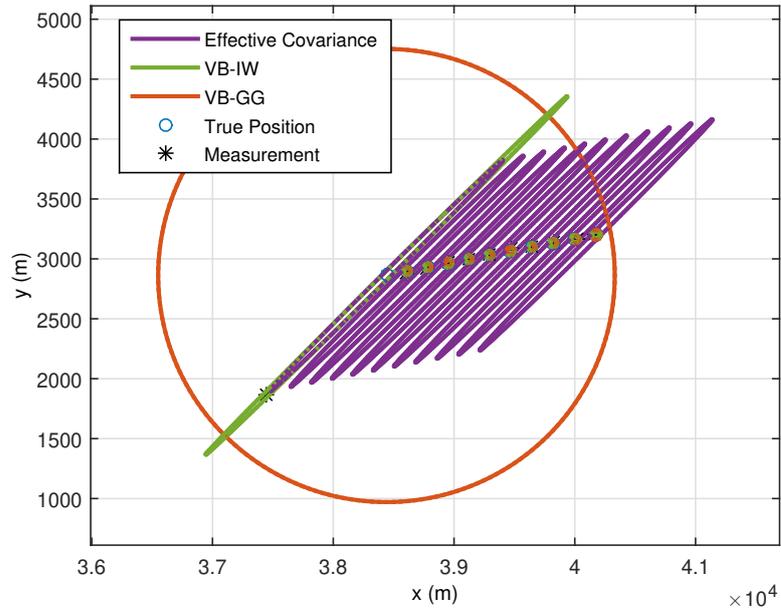


Figure 4.26: Measurement noise covariance tracking of the algorithms VB-GG and fixed prior VB-IW for Case 4 (for any time interval of a single MC run). The effective measurement noise covariances are shown by purple ellipses, the effective covariance calculated by fixed prior VB-IW used in the measurement update of the state in VB iterations are shown by green ellipses and the effective covariance calculated by VB-GG used in the measurement update of the state in VB iterations are shown by orange circles. The blue circles and black stars show the true positions and the measurements, respectively.

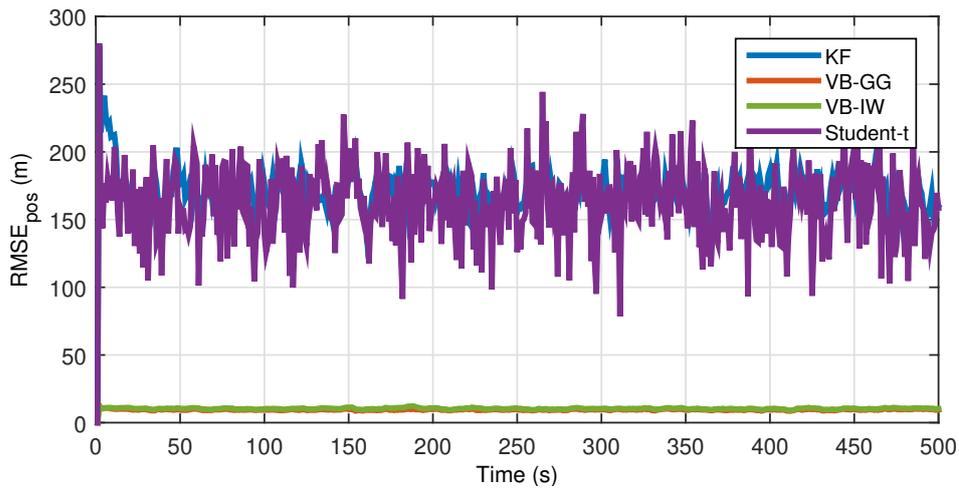


Figure 4.27: RMSEs of the position for 500 Monte Carlo run for Case 5 by using fixed prior VB-IW algorithm. RMSEs of the position by VB-GG is shown by orange line, RMSEs of the position fixed prior VB-IW is shown by green line, RMSEs of the position by STF is shown by purple line and RMSEs of the position by KF is shown by blue line.

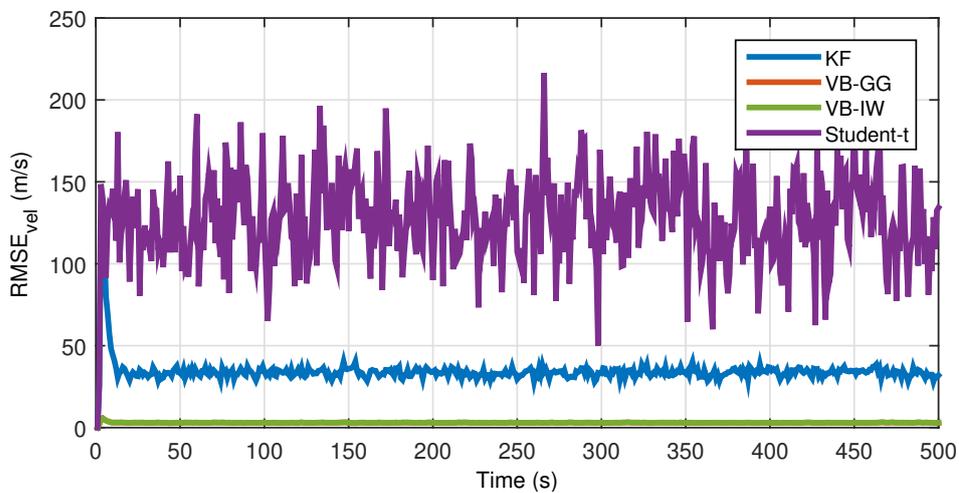


Figure 4.28: RMSEs of the velocity for 500 Monte Carlo run for Case 5 by using fixed prior VB-IW algorithm. RMSEs of the velocity by VB-GG is shown by orange line, RMSEs of the velocity by fixed prior VB-IW is shown by green line, RMSEs of the velocity by STF is shown by purple line and RMSEs of the velocity by KF is shown by blue line.

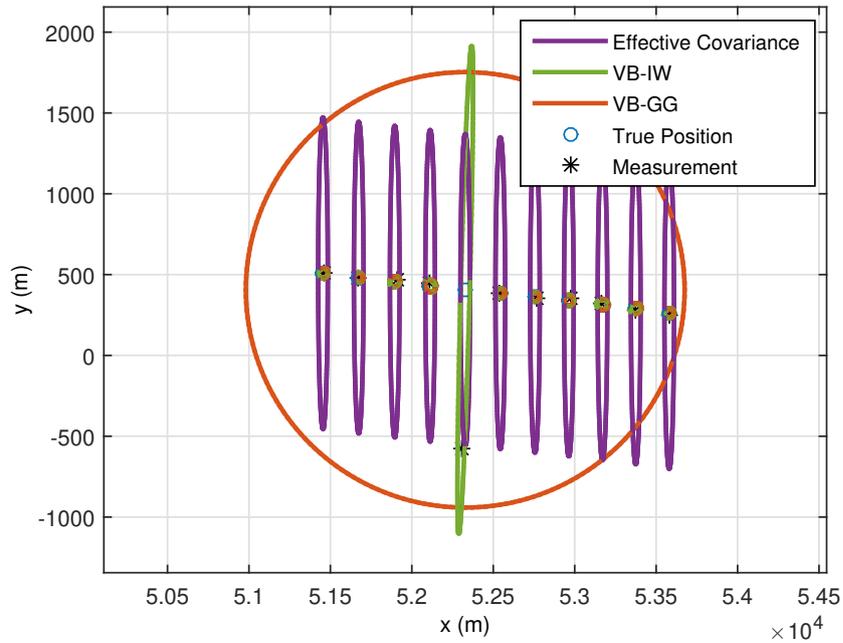


Figure 4.29: Measurement noise covariance tracking of the algorithms VB-GG and fixed prior VB-IW for Case 5 (for any time interval of a single MC run). The effective measurement noise covariances are shown by purple ellipses, the effective covariance calculated by fixed prior VB-IW used in the measurement update of the state in VB iterations are shown by green ellipses and the effective covariance calculated by VB-GG used in the measurement update of the state in VB iterations are shown by orange circles. The blue circles and black stars show the true positions and the measurements, respectively.

CHAPTER 5

MULTIPLE-MODEL EXTENSIONS OF OUTLIER ROBUST FILTERS USING VB APPROACH

As mentioned before, both multimodality and heavy-tailed noises may exist at the same time in many real world applications. In Chapter 3, IMM-STF is derived for the multiple-model systems that have heavy-tailed noises. It is shown that IMM-STF outperforms the conventional IMM algorithm. In addition to STF, VB methods are proposed for the rejection of outliers. In the previous chapter, the derivations of two different VB methods, VB-GG and VB-IW, are given and the performances of these methods are compared. In order to solve the tracking problem of the multiple-model systems that have heavy-tailed measurement noise, a multiple-model extension of VB-GG approach is proposed in [39,47]. It is demonstrated by the simulation results in the previous chapter that VB-IW algorithm provides better estimation accuracy than VB-GG algorithm. Therefore, a multiple-model extension of VB-IW approach is proposed in this chapter for multiple-model systems that have outliers in measurements.

In this chapter, a multiple-model extension of VB-GG algorithm and a multiple-model extension of VB-IW algorithm are derived based on IMM approach. These multiple-model extensions are referred as IMM-VB-GG and IMM-VB-IW. The performances of these two methods are compared for five different measurement noise characteristics.

5.1 Multiple-Model Extension of VB-GG Algorithm (IMM-VB-GG)

In this section, the derivation of IMM-VB-GG is given. We consider the hybrid system (2.42) where

- $w_{k-1}^{r_k}$ is distributed with $\mathcal{N}(0, Q^{r_k})$,
- $v_k^{r_k}$ is distributed with $St(0, R^{r_k}, v^{r_k})$.

The mode state r_k is modeled as a Markov chain that have the TPM $\Pi = [\pi_{ji} \triangleq P(r_k = i | r_{k-1} = j)]$. We assume that we have N modes. The block diagram of a single step of the IMM-VB-GG algorithm is shown in Figure 5.1.

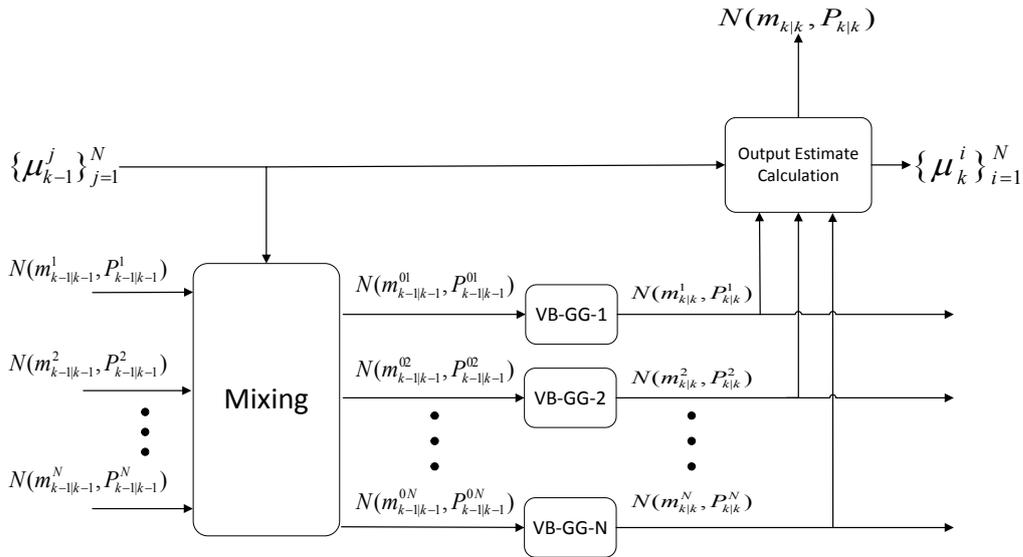


Figure 5.1: The block diagram of a single step of IMM-VB-GG for N-models.

Since in VB-GG algorithm, the filtered PDF $p(x_k | y_{1:k})$ is approximated as Gaussian distributed, the mixing calculations and the calculation of overall mean and covariance are the same as in the conventional IMM algorithm. We assume that we have statistics $\{m_{k-1|k-1}^j, P_{k-1|k-1}^j, \mu_{k-1}^j\}_{j=1}^N$ from the previous step. Then, the statistics

$\{m_{k|k}^i, P_{k|k}^i, \mu_{k|k}^i\}_{i=1}^N$ are obtained at the end of the one step of the algorithm. First of all, the mixing mode probabilities are obtained as

$$\mu_{k-1|k-1}^{ji} = \frac{\pi_{ji} \mu_{k-1}^j}{\sum_{l=1}^N \pi_{li} \mu_{k-1}^l}. \quad (5.1)$$

Then, the prior distributions are mixed using the mixed mode probabilities. The merged means and covariances are computed by equation (2.49). Now, we have the mixed statistics $\{m_{k-1|k-1}^{0i}, P_{k-1|k-1}^{0i}\}_{i=1}^N$ which are the inputs for VB-GG filters. The posterior PDF for each mode is calculated by VB-GG algorithm given in the previous chapter. One-step predicted PDF $p(x_k|y_{1:k-1}, r_k = i)$ and the likelihood PDF $p(y_k|x_k, r_k = i)$ are

$$p(x_k|y_{1:k-1}, r_k = i) = \mathcal{N}(x_k^i; m_{k|k-1}^i, P_{k|k-1}^i), \quad (5.2a)$$

$$p(y_k|x_k, r_k = i) = St(y_k; C^i x_k^i, R^i, v^i). \quad (5.2b)$$

The parameters of the one-step predicted PDF are calculated by using KF time update equation (2.8) since the process noise is Gaussian distributed, i.e.,

$$m_{k|k-1}^i = A^i m_{k-1|k-1}^{0i}, \quad (5.3a)$$

$$P_{k|k-1}^i = A^i P_{k-1|k-1}^{0i} (A^i)^T + Q^i. \quad (5.3b)$$

The likelihood (5.2b) can be rewritten by using the Gamma-Gaussian representation of the Student's-t PDF (4.13) for each mode i

$$p(y_k|x_k, r_k = i) = \int \mathcal{N}(y_k; C^i x_k^i, R^i/\lambda_k^i) Gam(\lambda_k^i; v^i/2, v^i/2) d\lambda_k^i. \quad (5.4)$$

Therefore, we have

$$p(y_k|x_k, \lambda_k, r_k = i) = \mathcal{N}(y_k; C^i x_k^i, R^i/\lambda_k^i), \quad (5.5a)$$

$$p(\lambda_k|r_k = i) = Gam(\lambda_k^i; v^i/2, v^i/2). \quad (5.5b)$$

In order to estimate the state x_k^i , the joint posterior PDF $p(x_k, \lambda_k|y_{1:k}, r_k = i)$ need to

be approximated by VB approach as

$$p(x_k, \lambda_k | y_{1:k}, r_k = i) \approx q(x_k^i)q(\lambda_k^i), \quad (5.6)$$

where $q(\cdot)$ is the approximate posterior PDF. The approximation steps are the same with given in Chapter 4. Thus, $q(\lambda_k^i)^{(s+1)}$ is updated as Gamma PDF as

$$q(\lambda_k^i)^{(s+1)} = \text{Gam} \left(\lambda_k^i; \gamma_k^{i(s+1)}, \delta_k^{i(s+1)} \right), \quad (5.7)$$

where $q(\cdot)^{(s+1)}$ is the approximation of $q(\cdot)$ at $(s+1)$ th iteration and

$$\gamma_k^{i(s+1)} = \frac{1}{2}(m + v^i), \quad (5.8a)$$

$$\delta_k^{i(s+1)} = \frac{1}{2} \left[v^i + \text{tr} \left(E_k^{i(s)} (R^i)^{-1} \right) \right], \quad (5.8b)$$

$$E_k^{i(s)} = \left(y_k - C^i m_{k|k}^{i(s)} \right) \left(y_k - C^i m_{k|k}^{i(s)} \right)^T + C^i P_{k|k}^{i(s)} (C^i)^T. \quad (5.8c)$$

The modified likelihood PDF $p(y_k | x_k, r_k = i)^{(s+1)}$ is defined as

$$p(y_k | x_k, r_k = i)^{(s+1)} = \mathcal{N} \left(y_k; C^i x_k^i, \left[\tilde{R}_k^i \right]^{(s+1)} \right), \quad (5.9)$$

where $\left[\tilde{R}_k^i \right]^{(s+1)}$ is the effective covariance matrix, which is used in the measurement update of the state for mode i at $(s+1)$ th iteration. It is calculated as

$$\left[\tilde{R}_k^i \right]^{(s+1)} = \frac{R^i}{\mathbf{E} [\lambda_k^i]^{(s+1)}}, \quad (5.10)$$

and

$$\mathbf{E} [\lambda_k^i]^{(s+1)} = \frac{\gamma_k^{i(s+1)}}{\delta_k^{i(s+1)}}. \quad (5.11)$$

As given in Chapter 4, $q(x_k^i)^{(s+1)}$ is updated as Gaussian PDF as

$$q(x_k^i)^{(s+1)} = \mathcal{N} \left(x_k^i; m_{k|k}^{i(s+1)}, P_{k|k}^{i(s+1)} \right), \quad (5.12)$$

where

$$m_{k|k}^i{}^{(s+1)} = m_{k|k-1}^i + K_k^i{}^{(s+1)} (y_k - C^i m_{k|k-1}^i), \quad (5.13a)$$

$$P_{k|k}^i{}^{(s+1)} = P_{k|k-1}^i - K_k^i{}^{(s+1)} C^i P_{k|k-1}^i, \quad (5.13b)$$

and

$$S_k^i{}^{(s+1)} = C^i P_{k|k-1}^i (C^i)^T + [\tilde{R}_k^i]^{(s+1)}, \quad (5.14a)$$

$$K_k^i{}^{(s+1)} = P_{k|k-1}^i (C^i)^T (S_k^i{}^{(s+1)})^{-1}, \quad (5.14b)$$

After S iterations, the approximate posterior PDF of the state for each mode i becomes

$$q^*(x_k^i) \approx q(x_k^i)^{(S)} = \mathcal{N}(x_k^i; m_{k|k}^i{}^{(S)}, P_{k|k}^i{}^{(S)}) = \mathcal{N}(x_k^i; m_{k|k}^i, P_{k|k}^i), \quad (5.15)$$

and the innovation covariance is

$$S_k^i = S_k^i{}^{(S)}. \quad (5.16)$$

In order to obtain the overall mean and covariance of the state, mode probabilities should be updated as

$$\begin{aligned} \mu_k^i &\triangleq P(r_k = i | y_{1:k}) \propto p(y_k | y_{1:k-1}, r_k = i) P(r_k = i | y_{1:k-1}) \\ &\propto p(y_k | y_{1:k-1}, r_k = i) \sum_{j=1}^N P(r_k = i | r_{k-1} = j) P(r_{k-1} = j | y_{1:k-1}) \\ &\propto p(y_k | y_{1:k-1}, r_k = i) \sum_{j=1}^N \pi_{ji} \mu_{k-1}^j. \end{aligned} \quad (5.17)$$

$$\mu_k^i = \frac{p(y_k | y_{1:k-1}, r_k = i) \sum_{j=1}^N \pi_{ji} \mu_{k-1}^j}{\sum_{l=1}^N p(y_k | y_{1:k-1}, r_k = l) \sum_{j=1}^N \pi_{jl} \mu_{k-1}^j}. \quad (5.18)$$

The likelihood $p(y_k | y_{1:k-1}, r_k = i)$ in equation (5.18) can be obtained by computing

the following integral:

$$\begin{aligned}
p(y_k|y_{1:k-1}, r_k = i) &= \int \int p(y_k|x_k, \lambda_k, r_k = i)p(x_k|y_{0:k-1}, r_k = i) \\
&\quad \times p(\lambda_k|r_k = i)d\lambda_k dx_k \\
&= \int \int \mathcal{N}(y_k; C^i x_k^i, R^i/\lambda_k^i)\mathcal{N}(x_k^i; m_{k|k-1}^i, P_{k|k-1}^i) \\
&\quad \times \text{Gam}(\lambda_k^i; v^i/2, v^i/2)d\lambda_k^i dx_k^i
\end{aligned} \tag{5.19}$$

Since there is no analytical expression for integral in equation (5.19), it is approximated as in [25]

$$p(y_k|y_{1:k-1}, r_k = i) \approx \mathcal{N}\left(y_k; C^i m_{k|k-1}^i, C^i P_{k|k-1}^i (C^i)^T + [\tilde{R}_k^i]^{(N)}\right) \tag{5.20}$$

By substituting equation (5.20) in (5.18);

$$\mu_k^i = \frac{\mathcal{N}\left(y_k; C^i m_{k|k-1}^i, C^i P_{k|k-1}^i (C^i)^T + [\tilde{R}_k^i]^{(N)}\right) \sum_{j=1}^N \pi_{ji} \mu_{k-1}^j}{\sum_{l=1}^N \mathcal{N}\left(y_k; C^l m_{k|k-1}^l, C^l P_{k|k-1}^l (C^l)^T + [\tilde{R}_k^l]^{(N)}\right) \sum_{j=1}^N \pi_{jl} \mu_{k-1}^j}. \tag{5.21}$$

The overall mean and covariance are computed by equation (2.45). IMM-VB-GG algorithm is summarized in Algorithm 4.

Algorithm 4 IMM-VB-GG Algorithm

Inputs: $\{m_{k-1|k-1}^j, P_{k-1|k-1}^j, \mu_{k-1}^j\}_{j=1}^N, \{A^i, C^i, Q^i, R^i, v^i\}_{i=1}^N, y_k, m, n, S$

for $i=1:N$ **do**

Mixing:

for $j=1:N$ **do**

$$1. \mu_{k-1|k-1}^{ji} = \frac{\pi_{ji} \mu_{k-1}^i}{\sum_{l=1}^N \pi_{li} \mu_{k-1}^l}$$

end for

$$2. m_{k-1|k-1}^{0i} = \sum_{j=1}^N \mu_{k-1|k-1}^{ji} m_{k-1|k-1}^j$$

$$3. P_{k-1|k-1}^{0i} = \sum_{j=1}^N \mu_{k-1|k-1}^{ji} \left[P_{k-1|k-1}^j + (m_{k-1|k-1}^j - m_{k-1|k-1}^{0i})(m_{k-1|k-1}^j - m_{k-1|k-1}^{0i})^T \right]$$

Time Update:

$$4. m_{k|k-1}^i = A^i m_{k-1|k-1}^{0i}$$

$$5. P_{k|k-1}^i = A^i P_{k-1|k-1}^{0i} (A^i)^T + Q^i$$

Measurement Update:

$$6. \text{Initialization: } m_{k|k}^i{}^{(0)} = m_{k|k-1}^i, P_{k|k}^i{}^{(0)} = P_{k|k-1}^i$$

for $s=0:S-1$ **do**

Update $q(\lambda_k^i)^{(s+1)}$ given $q(x_k^i)^{(s)}$

$$7. E_k^{i(s)} = \left(y_k - C^i m_{k|k}^i{}^{(s)} \right) \left(y_k - C^i m_{k|k}^i{}^{(s)} \right)^T + C^i P_{k|k}^i{}^{(s)} (C^i)^T$$

$$8. \gamma_k^{i(s+1)} = \frac{1}{2} (m + v^i), \delta_k^{i(s+1)} = \frac{1}{2} \left(v^i + \text{tr} \left(E_k^{i(s)} (R^i)^{-1} \right) \right), \mathbf{E} [\lambda_k^i]^{(s+1)} = \frac{\gamma_k^{i(s+1)}}{\delta_k^{i(s+1)}}$$

Update $q(x_k^i)^{(s+1)}$ given $q(\lambda_k^i)^{(s+1)}$

$$9. \left[\tilde{R}_k^i \right]^{(s+1)} = \frac{R^i}{\mathbf{E} [\lambda_k^i]^{(s+1)}}$$

$$10. S_k^{i(s+1)} = C^i P_{k|k-1}^i (C^i)^T + \left[\tilde{R}_k^i \right]^{(s+1)}$$

$$11. K_k^{i(s+1)} = P_{k|k-1}^i (C^i)^T \left(S_k^{i(s+1)} \right)^{-1}$$

$$12. m_{k|k}^i{}^{(s+1)} = m_{k|k-1}^i + K_k^{i(s+1)} \left(y_k - C^i m_{k|k-1}^i \right)$$

$$13. P_{k|k}^i{}^{(s+1)} = P_{k|k-1}^i - K_k^{i(s+1)} C^i P_{k|k-1}^i$$

end for

$$14. m_{k|k}^i = m_{k|k}^i{}^{(S)}, P_{k|k}^i = P_{k|k}^i{}^{(S)}, S_k^i = S_k^i{}^{(S)}$$

end for

for $i=1:N$ **do**

$$15. \mu_k^i = \frac{\mathcal{N}\left(y_k; C^i m_{k|k-1}^i, C^i P_{k|k-1}^i (C^i)^T + [\tilde{R}_k^i]^{(N)}\right) \sum_{j=1}^N \pi_{ji} \mu_{k-1}^j}{\sum_{l=1}^N \mathcal{N}\left(y_k; C^l m_{k|k-1}^l, C^l P_{k|k-1}^l (C^l)^T + [\tilde{R}_k^l]^{(N)}\right) \sum_{j=1}^N \pi_{jl} \mu_{k-1}^j}$$

end for

$$16. m_{k|k} = \sum_{i=1}^N \mu_k^i m_{k|k}^i$$

$$17. P_{k|k} = \sum_{i=1}^N \mu_k^i [P_{k|k}^i + (m_{k|k}^i - m_{k|k})(m_{k|k}^i - m_{k|k})^T]$$

Outputs: $\{m_{k|k}^i, P_{k|k}^i, \mu_k^i\}_{i=1}^N, m_{k|k}, P_{k|k}$

5.2 Multiple-Model Extension of VB-IW Algorithm (IMM-VB-IW)

In this section, the derivation of IMM-VB-IW is given. VB-IW algorithm is used to estimate the effective covariance of the measurement noise and the state of each mode of the hybrid system. We consider the hybrid system (2.42) where

- $w_{k-1}^{r_k}$ is distributed with $\mathcal{N}(0, Q^{r_k})$,
- $v_k^{r_k}$ is distributed with $St(0, R^{r_k}, v^{r_k})$.

The mode state r_k is modeled as a Markov chain that have the TPM $\Pi = [\pi_{ji} \triangleq P(r_k = i | r_{k-1} = j)]$. We assume that we have N modes. The block diagram of a single step of the IMM-VB-IW algorithm is shown in Figure 5.2.

As seen in Figure 5.2, we have probability distributions $\{\mathcal{N}(x_{k-1}^j; m_{k-1|k-1}^j, P_{k-1|k-1}^j), IW(\Sigma_{k-1}^j; u_{k-1|k-1}^j, U_{k-1|k-1}^j)\}_{j=1}^N$ from the previous step where Σ_{k-1}^j is the effective covariance of the measurement noise at time step $k-1$ for mode j and the probability distributions $\{\mathcal{N}(x_k^i; m_{k|k}^i, P_{k|k}^i), IW(\Sigma_k^i; u_{k|k}^i, U_{k|k}^i)\}_{i=1}^N$ are obtained at the end of the one step of the algorithm. The prior distributions of the state and the effective covariance of the measurement noise are Gaussian and Inverse-Wishart, respectively. Hence, the joint distribution of the state and the effective covariance of the measurement noise of the j th mode given the measurements can be seen as the product of Gaussian and

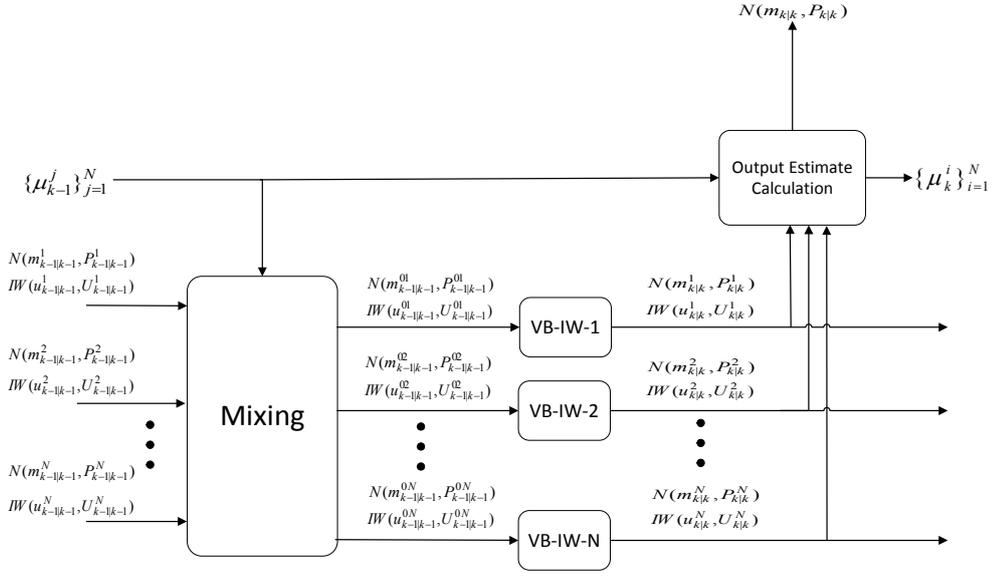


Figure 5.2: The block diagram of a single step of IMM-VB-IW for N-models.

Inverse-Wishart distribution such that

$$p(x_{k-1}, \Sigma_{k-1} | y_{1:k-1}, r_{k-1} = j) = \mathcal{N}(x_{k-1}^j; m_{k-1|k-1}^j, P_{k-1|k-1}^j) \times IW(\Sigma_{k-1}^j; u_{k-1|k-1}^j, U_{k-1|k-1}^j), \quad (5.22)$$

which is named as Normal-Inverse Wishart distribution.

First of all, the mixing mode probabilities are obtained as

$$\mu_{k-1|k-1}^{ji} = \frac{\pi_{ji} \mu_{k-1}^j}{\sum_{l=1}^N \pi_{li} \mu_{k-1}^l}. \quad (5.23)$$

Then, the prior Normal-Inverse Wishart distributions (5.22) are mixed using the mixing mode probabilities. The merged statistics are calculated as

$$m_{k-1|k-1}^{0i} = \sum_{j=1}^N \mu_{k-1|k-1}^{ji} m_{k-1|k-1}^j, \quad (5.24)$$

$$\begin{aligned}
P_{k-1|k-1}^{0i} &= \sum_{j=1}^N \mu_{k-1|k-1}^{ji} \\
&\times \left[P_{k-1|k-1}^j + (m_{k-1|k-1}^j - m_{k-1|k-1}^{0i})(m_{k-1|k-1}^j - m_{k-1|k-1}^{0i})^T \right],
\end{aligned} \tag{5.25}$$

$$\begin{aligned}
U_{k-1|k-1}^{0i} &= (u_{k-1|k-1}^{0i} - m - 1) \\
&\times \left(\sum_{j=1}^N \mu_{k-1|k-1}^{ji} (u_{k-1|k-1}^{0i} - m - 1) \left(U_{k-1|k-1}^j \right)^{-1} \right)^{-1},
\end{aligned} \tag{5.26}$$

and

$$\begin{aligned}
u_{k-1|k-1}^{0i} &= \arg \min_u \left\{ m \log(u - m - 1) - \sum_{l=1}^m \psi_0 \left(\frac{u - m - l}{2} \right) \right. \\
&\quad \left. - \log \left| \sum_{j=1}^N \mu_{k-1|k-1}^{ji} (u - m - 1) \left(U_{k-1|k-1}^j \right)^{-1} \right| \right. \\
&\quad \left. + \sum_{j=1}^N \sum_{l=1}^m \mu_{k-1|k-1}^{ji} \psi_0 \left(\frac{u - m - l}{2} \right) \right. \\
&\quad \left. - \sum_{j=1}^N \mu_{k-1|k-1}^{ji} \log \left| U_{k-1|k-1}^j \right| \right\},
\end{aligned} \tag{5.27}$$

by minimizing the Kullback-Leibler divergence where $\psi_0(\cdot)$ is the digamma function [19], which is the 1st order derivative of the logarithm of the gamma function [10]. Now, we have the mixed statistics $\{m_{k-1|k-1}^{0i}, P_{k-1|k-1}^{0i}, u_{k-1|k-1}^{0i}, U_{k-1|k-1}^{0i}\}_{i=1}^N$ which are the inputs for VB-IW filters. The posterior PDF for each mode is calculated by VB-IW algorithm given in the previous chapter. The one-step predicted PDF $p(x_k|y_{1:k-1}, r_k = i)$ and the likelihood PDF $p(y_k|x_k, \Sigma_k, r_k = i)$ are

$$p(x_k|y_{1:k-1}, r_k = i) = \mathcal{N}(x_k^i; m_{k|k-1}^i, P_{k|k-1}^i), \tag{5.28a}$$

$$p(y_k|x_k, \Sigma_k, r_k = i) = \mathcal{N}(y_k; C^i x_k^i, \Sigma_k^i). \tag{5.28b}$$

Since the process noise is assumed as Gaussian distributed, the parameters of one-step predicted PDF are calculated by KF time update equations (5.3). Now, the goal is to estimate the state x_k^i and the effective covariance of the measurement noise Σ_k^i . The

approximated posterior distributions $q(x_k^i)$ and $q(\Sigma_k^i)$ are obtained for each mode i as given in Chapter 4. Therefore, $q(\Sigma_k^i)^{(s+1)}$ is approximated as Inverse-Wishart PDF as

$$q(\Sigma_k^i)^{(s+1)} = IW\left(\Sigma_k^i; u_k^{i(s+1)}, U_k^{i(s+1)}\right), \quad (5.29)$$

where $q(\cdot)^{(s+1)}$ is the approximation of $q(\cdot)$ at $(s+1)$ th iteration. According to equations (4.57), (4.43) and (4.55), we obtain

$$u_k^{i(s+1)} = u_{k|k-1}^i + 1, \quad (5.30a)$$

$$U_k^{i(s+1)} = E_k^{i(s)} + U_{k|k-1}^i, \quad (5.30b)$$

and

$$u_{k|k-1}^i = \lambda(u_{k-1|k-1}^{0i} - m - 1) + m + 1, \quad (5.31a)$$

$$U_{k|k-1}^i = \lambda U_{k-1|k-1}^{0i}, \quad (5.31b)$$

$$E_k^{i(s)} = (y_k - C^i m_{k|k-1}^i)(y_k - C^i m_{k|k-1}^i)^T + C^i P_{k|k}^{i(s)} (C^i)^T. \quad (5.31c)$$

where λ is the forgetting factor. As given in Chapter 4, $q(x_k^i)^{(s+1)}$ is updated as Gaussian PDF as

$$q(x_k^i)^{(s+1)} = \mathcal{N}\left(x_k^i; m_{k|k}^{i(s+1)}, P_{k|k}^{i(s+1)}\right), \quad (5.32)$$

where

$$m_{k|k}^{i(s+1)} = m_{k|k-1}^i + K_k^{i(s+1)}(y_k - C^i m_{k|k-1}^i), \quad (5.33a)$$

$$P_{k|k}^{i(s+1)} = P_{k|k-1}^i - K_k^{i(s+1)} C^i P_{k|k-1}^i, \quad (5.33b)$$

$$S_k^{i(s+1)} = \left(C^i P_{k|k-1}^i (C^i)^T + [\tilde{\Sigma}_k^i]^{(s+1)}\right), \quad (5.33c)$$

$$K_k^{i(s+1)} = P_{k|k-1}^i (C^i)^T [S_k^{i(s+1)}]^{-1}, \quad (5.33d)$$

and the estimated measurement noise covariance matrix is

$$\left[\tilde{\Sigma}_k^i\right]^{(s+1)} = \left[\left(u_k^{i(s+1)} - m - 1\right) \left(U_k^{i(s+1)}\right)^{-1}\right]^{-1}. \quad (5.34)$$

After S iterations, the approximate posterior PDFs of the state and the effective measurement noise covariance matrix for each mode i become

$$q^*(x_k^i) \approx q(x_k^i)^{(S)} = \mathcal{N}\left(x_k^i; m_{k|k}^i{}^{(S)}, P_{k|k}^i{}^{(S)}\right) = \mathcal{N}\left(x_k^i; m_{k|k}^i, P_{k|k}^i\right), \quad (5.35a)$$

$$q^*(\Sigma_k^i) \approx q(\Sigma_k^i)^{(S)} = IW\left(\Sigma_k^i; u_k^{i(S)}, U_k^{i(S)}\right) = IW\left(\Sigma_k^i; u_{k|k}^i, U_{k|k}^i\right), \quad (5.35b)$$

and the innovation covariance is

$$S_k^i = S_k^{i(S)}. \quad (5.36)$$

In order to obtain the overall mean and covariance of the state, mode probabilities should be updated as

$$\begin{aligned} \mu_k^i &\triangleq P(r_k = i | y_{1:k}) \propto p(y_k | y_{1:k-1}, r_k = i) P(r_k = i | y_{1:k-1}) \\ &\propto p(y_k | y_{1:k-1}, r_k = i) \sum_{j=1}^N P(r_k = i | r_{k-1} = j) P(r_{k-1} = j | y_{1:k-1}) \\ &\propto p(y_k | y_{1:k-1}, r_k = i) \sum_{j=1}^N \pi_{ji} \mu_{k-1}^j. \end{aligned} \quad (5.37)$$

$$\mu_k^i = \frac{p(y_k | y_{1:k-1}, r_k = i) \sum_{j=1}^N \pi_{ji} \mu_{k-1}^j}{\sum_{l=1}^N p(y_k | y_{1:k-1}, r_k = l) \sum_{j=1}^N \pi_{jl} \mu_{k-1}^j}. \quad (5.38)$$

The likelihood $p(y_k | y_{1:k-1}, r_k = i)$ in equation (5.18) can be obtained by computing the following integral:

$$\begin{aligned} p(y_k | y_{1:k-1}, r_k = i) &= \int \int p(y_k | x_k, \Sigma_k^i, r_k = i) p(x_k | y_{0:k-1}, r_k = i) \\ &\quad \times p(\Sigma_k^i | y_{0:k-1}, r_k = i) d\Sigma_k dx_k \end{aligned} \quad (5.39)$$

$$\begin{aligned}
&= \int \int \mathcal{N}(y_k; C^i x_k^i, \Sigma_k^i) \mathcal{N}(x_k^i; m_{k|k-1}^i, P_{k|k-1}^i) \\
&\quad \times IW(\Sigma_k^i; u_{k|k-1}^i, U_{k|k-1}^i) d\Sigma_k^i dx_k^i
\end{aligned} \tag{5.40}$$

Since there is no analytical expression for integral in equation (5.40), it is approximated as in [25]

$$p(y_k | y_{1:k-1}, r_k = i) \approx \mathcal{N} \left(y_k; C^i m_{k|k-1}^i, C^i P_{k|k-1}^i (C^i)^T + \frac{U_{k|k-1}^i}{u_{k|k-1}^i - m - 1} \right) \tag{5.41}$$

In [16] and [37], different approximations are given for this integral. By substituting equation (5.41) in (5.38);

$$\mu_k^i = \frac{\mathcal{N} \left(y_k; C^i m_{k|k-1}^i, C^i P_{k|k-1}^i (C^i)^T + \frac{U_{k|k-1}^i}{u_{k|k-1}^i - m - 1} \right) \sum_{j=1}^N \pi_{ji} \mu_{k-1}^j}{\sum_{l=1}^N \mathcal{N} \left(y_k; C^l m_{k|k-1}^l, C^l P_{k|k-1}^l (C^l)^T + \frac{U_{k|k-1}^l}{u_{k|k-1}^l - m - 1} \right) \sum_{j=1}^N \pi_{jl} \mu_{k-1}^j}. \tag{5.42}$$

The overall mean and covariance values of the state are computed by equation (2.45). IMM-VB-IW algorithm is summarized in Algorithm 5.

Algorithm 5 IMM-VB-IW Algorithm

Inputs: $\{m_{k-1|k-1}^j, P_{k-1|k-1}^j, u_{k-1|k-1}^j, u_{k-1|k-1}^j \mu_{k-1}^j\}_{j=1}^N, \{A^i, C^i, Q^i, R^i\}_{i=1}^N, y_k,$
 m, n, λ, S

for $i=1:N$ **do**

Mixing:

for $j=1:N$ **do**

$$1. \mu_{k-1|k-1}^{ji} = \frac{\pi_{ji} \mu_{k-1}^i}{\sum_{l=1}^N \pi_{li} \mu_{k-1}^l}$$

end for

$$2. m_{k-1|k-1}^{0i} = \sum_{j=1}^N \mu_{k-1|k-1}^{ji} m_{k-1|k-1}^j$$

$$3. P_{k-1|k-1}^{0i} = \sum_{j=1}^N \mu_{k-1|k-1}^{ji} \times \left[P_{k-1|k-1}^j + (m_{k-1|k-1}^j - m_{k-1|k-1}^{0i})(m_{k-1|k-1}^j - m_{k-1|k-1}^{0i})^T \right]$$

$$4. u_{k-1|k-1}^{0i} = \arg \min_u \left\{ m \log(u - m - 1) - \sum_{l=1}^m \psi_0 \left(\frac{u - m - l}{2} \right) - \log \left| \sum_{j=1}^N \mu_{k-1|k-1}^{ji} (u - m - 1) \left(U_{k-1|k-1}^j \right)^{-1} \right| + \sum_{j=1}^N \sum_{l=1}^m \mu_{k-1|k-1}^{ji} \psi_0 \left(\frac{u - m - l}{2} \right) - \sum_{j=1}^N \mu_{k-1|k-1}^{ji} \log \left| U_{k-1|k-1}^j \right| \right\}$$

$$5. U_{k-1|k-1}^{0i} = (u_{k-1|k-1}^{0i} - m - 1) \times \left(\sum_{j=1}^N \mu_{k-1|k-1}^{ji} (u_{k-1|k-1}^{0i} - m - 1) \left(U_{k-1|k-1}^j \right)^{-1} \right)^{-1}$$

Time Update:

$$6. m_{k|k-1}^i = A^i m_{k-1|k-1}^{0i}$$

$$7. P_{k|k-1}^i = A^i P_{k-1|k-1}^{0i} (A^i)^T + Q^i$$

end for

(to be continued)

for i=1:N do

Measurement Update:

8. Initialization: $m_{k|k}^{i(0)} = m_{k|k-1}^i$, $P_{k|k}^{i(0)} = P_{k|k-1}^i$, $u_{k|k-1}^i = \lambda(u_{k-1|k-1}^{0i} - m - 1) + m + 1$, $U_{k|k-1}^i = \lambda U_{k-1|k-1}^{0i}$

for s=0:S-1 do

Update $q(\Sigma_k^i)^{(s+1)}$ given $q(x_k^i)^{(s)}$

9. $E_k^{i(s)} = \left(y_k - C^i m_{k|k}^{i(s)} \right) \left(y_k - C^i m_{k|k}^{i(s)} \right)^T + C^i P_{k|k}^{i(s)} (C^i)^T$

10. $u_k^{i(s+1)} = u_{k|k-1}^i + 1$, $U_k^{i(s+1)} = E_k^{i(s)} + U_{k|k-1}^i$

Update $q(x_k^i)^{(s+1)}$ given $q(\Sigma_k^i)^{(s+1)}$

11. $\mathbf{E} \left[(\Sigma_k^i)^{-1} \right]^{(s+1)} = \left(u_k^{i(s+1)} - m - 1 \right) \left(U_k^{i(s+1)} \right)^{-1}$

12. $\left[\tilde{\Sigma}_k^i \right]^{(s+1)} = \left(\mathbf{E} \left[(\Sigma_k^i)^{-1} \right]^{(s+1)} \right)^{-1}$

13. $S_k^{i(s+1)} = C^i P_{k|k-1}^i (C^i)^T + \left[\tilde{R}_k^i \right]^{(s+1)}$

14. $K_k^{i(s+1)} = P_{k|k-1}^i (C^i)^T \left(S_k^{i(s+1)} \right)^{-1}$

15. $m_{k|k}^{i(s+1)} = m_{k|k-1}^i + K_k^{i(s+1)} \left(y_k - C^i m_{k|k-1}^i \right)$

16. $P_{k|k}^{i(s+1)} = P_{k|k-1}^i - K_k^{i(s+1)} C^i P_{k|k-1}^i$

end for

17. $m_{k|k}^i = m_{k|k}^{i(S)}$, $P_{k|k}^i = P_{k|k}^{i(S)}$, $u_{k|k}^i = u_{k|k}^{i(S)}$, $U_{k|k}^i = U_{k|k}^{i(S)}$, $S_k^i = S_k^{i(S)}$

end for

for i=1:N do

18. $\mu_k^i = \frac{\mathcal{N} \left(y_k; C^i m_{k|k-1}^i, C^i P_{k|k-1}^i (C^i)^T + \frac{U_{k|k-1}^i}{u_{k|k-1}^i - m - 1} \right) \sum_{j=1}^N \pi_{ji} \mu_{k-1}^j}{\sum_{l=1}^N \mathcal{N} \left(y_k; C^l m_{k|k-1}^l, C^l P_{k|k-1}^l (C^l)^T + \frac{U_{k|k-1}^l}{u_{k|k-1}^l - m - 1} \right) \sum_{j=1}^N \pi_{jl} \mu_{k-1}^j}$

end for

19. $m_{k|k} = \sum_{i=1}^N \mu_k^i m_{k|k}^i$

20. $P_{k|k} = \sum_{i=1}^N \mu_k^i \left[P_{k|k}^i + (m_{k|k}^i - m_{k|k}) (m_{k|k}^i - m_{k|k})^T \right]$

Outputs: $\{m_{k|k}^i, P_{k|k}^i, u_{k|k}^i, U_{k|k}^i, \mu_k^i\}_{i=1}^N$, $m_{k|k}$, $P_{k|k}$

5.3 Performance Evaluation

In this section, IMM-VB-GG and IMM-VB-IW algorithms are tested on a simulation of a moving target in 2-D space that have Gaussian process noise and heavy-tailed measurement noise. The motion of the target is modeled by CV [8] and CT [40] models. These tests are conducted for five different characteristics of measurement noise. The state is the same with (3.41). The system matrices in (2.42) for CV and CT models are

$$A^{CV} = \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix}, A^{CT} = \begin{bmatrix} 1 & \frac{\sin(\omega T)}{\omega} & 0 & -\frac{1 - \cos(\omega T)}{\omega} \\ 0 & \cos(\omega T) & 0 & -\sin(\omega T) \\ 0 & \frac{1 - \cos(\omega T)}{\omega} & 1 & \frac{\sin(\omega T)}{\omega} \\ 0 & \sin(\omega T) & 0 & \cos(\omega T) \end{bmatrix}, \quad (5.43)$$

and

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}, \quad (5.44)$$

where $T = 1$ s is the sampling time and $\omega = \pi/15$ rad/s is the turn rate of CT model. A^{CV} and A^{CT} are the state transition matrices for CV and CT models, respectively. The process noise is distributed with normal distribution $w_k \sim \mathcal{N}(0, Q)$ where

$$Q = \begin{bmatrix} \frac{T^3}{3} & \frac{T^2}{2} & 0 & 0 \\ \frac{T^2}{2} & T & 0 & 0 \\ 0 & 0 & \frac{T^3}{3} & \frac{T^2}{2} \\ 0 & 0 & \frac{T^2}{2} & T \end{bmatrix}. \quad (5.45)$$

We consider five different cases for measurement noise generation. These cases are given below.

Case 1:

The measurement noise is generated according to

$$v_k \sim \begin{cases} \mathcal{N}(0, R_0) & \text{w.p. } p_0 \\ \mathcal{N}(\mu_1, R_1) & \text{w.p. } p_1 \\ \mathcal{N}(\mu_2, R_2) & \text{w.p. } p_2 \end{cases} \quad (5.46)$$

where $p_0 = 0.4$, $p_1 = 0.3$, $p_2 = 0.4$, $R_0 = 100 \times I_2$, $R_1 = R_2 = 50 \times I_2$, $\mu_1 = [70 \ 70]^T$ and $\mu_2 = [-70 \ -70]^T$. The nominal covariance, outlier covariances and the effective covariance is shown in Figure 4.1.

Case 2:

The measurement noise is generated according to (5.46) where $p_0 = 0.4$, $p_1 = 0.3$, $p_2 = 0.4$, $R_0 = 100 \times I_2$, $R_1 = R_2 = 50 \times I_2$, $\mu_1 = [0 \ 100]^T$ and $\mu_2 = [0 \ -100]^T$. The nominal covariance, outlier covariances and the effective covariance is shown in Figure 4.2.

Case 3:

The measurement noise is generated according to

$$v_k \sim \begin{cases} \mathcal{N}(0, R_0) & \text{w.p. } 0.9 \\ \mathcal{N}(0, 100R_0) & \text{w.p. } 0.1 \end{cases} \quad (5.47)$$

where $R_0 = 100 \times I_2$. The nominal covariance, outlier covariances and the effective covariance is shown in Figure 4.3.

Case 4:

The measurement noise is generated according to (5.46) where $p_0 = 0.8$, $p_1 = 0.1$, $p_2 = 0.1$, $R_0 = 100 \times I_2$, $R_1 = R_2 = 50 \times I_2$, $\mu_1 = [1000 \ 1000]^T$ and $\mu_2 = [-1000 \ -1000]^T$. The nominal covariance, outlier covariances and the effective covariance is shown in Figure 4.4.

Case 5:

The measurement noise is generated according to (5.46) where $p_0 = 0.8$, $p_1 = 0.1$, $p_2 = 0.1$, $R_0 = 100 \times I_2$, $R_1 = R_2 = 50 \times I_2$, $\mu_1 = [0 \ 1000]^T$ and $\mu_2 = [0 \ -1000]^T$. The nominal covariance, outlier covariances and the effective covariance is shown in Figure 4.5.

The target is simulated as switching between two modes which are CV mode and CT mode. The motion of the target is simulated according to

- CV mode between $k = 0s$ and $k = 100s$,
- CT mode between $k = 101s$ and $k = 150s$,
- CV mode between $k = 151s$ and $k = 250s$,

and the switching is controlled by TPM

$$\Pi = \begin{bmatrix} 0.95 & 0.05 \\ 0.05 & 0.95 \end{bmatrix}. \quad (5.48)$$

In this simulation, the position, the velocity and the effective covariance of the measurement noise are estimated with both IMM-VB-GG and IMM-VB-IW algorithms. The simulations are carried out for 250 MC runs, 250 time steps for each MC run and 10 iterations are performed for both VB algorithms. Moreover, the degrees of freedom parameter ν in IMM-VB-GG algorithm for each mode is taken as 3 [22]. The forgetting factor λ in IMM-VB-IW algorithm is taken as 1 [23]. For evaluating the estimation accuracy of the state, RMSE and ARMSE given in equations (3.16) and (3.17) are used as performance metrics.

Figure 5.3 and Figure 5.4 show respectively the RMSEs of position and velocity for Case 1. The ARMSEs of position and velocity for IMM-VB-GG and IMM-VB-IW for Case 1 are given in Table 5.1. In Figure 5.5, the effective measurement noise covariance, the estimated measurement noise covariance by IMM-VB-IW and the effective covariance calculated by IMM-VB-GG used in the measurement update of the state in VB iterations for Case 1 are shown for any time interval of a single MC run.

Furthermore, mode probability graphs of IMM-VB-GG and IMM-VB-IW algorithms for Case 1 are given in figures 5.6 and 5.7, respectively.

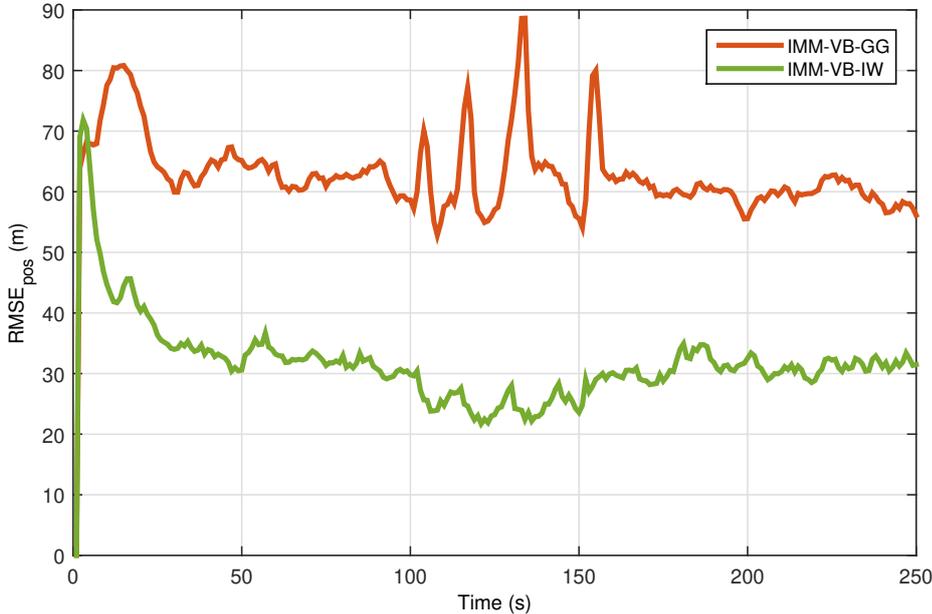


Figure 5.3: RMSEs of the position for 250 Monte Carlo run for Case 1. RMSEs of the position by IMM-VB-GG is shown by orange line and RMSEs of the position by IMM-VB-IW is shown by green line.

Table 5.1: ARMSEs of IMM-VB-GG and IMM-VB-IW for Case 1

	ARMSE of the position (m)	ARMSE of the velocity (m/s)
IMM-VB-GG	63.20	12.65
IMM-VB-IW	32.57	6.34

Figure 5.8 and Figure 5.9 show respectively the RMSEs of position and velocity for Case 2. The ARMSEs of position and velocity for IMM-VB-GG and IMM-VB-IW for Case 2 are given in Table 5.2. In Figure 5.10, the effective measurement noise covariance, the estimated measurement noise covariance by IMM-VB-IW and the effective covariance calculated by IMM-VB-GG used in the measurement update of the state in VB iterations for Case 2 are shown for any time interval of a single

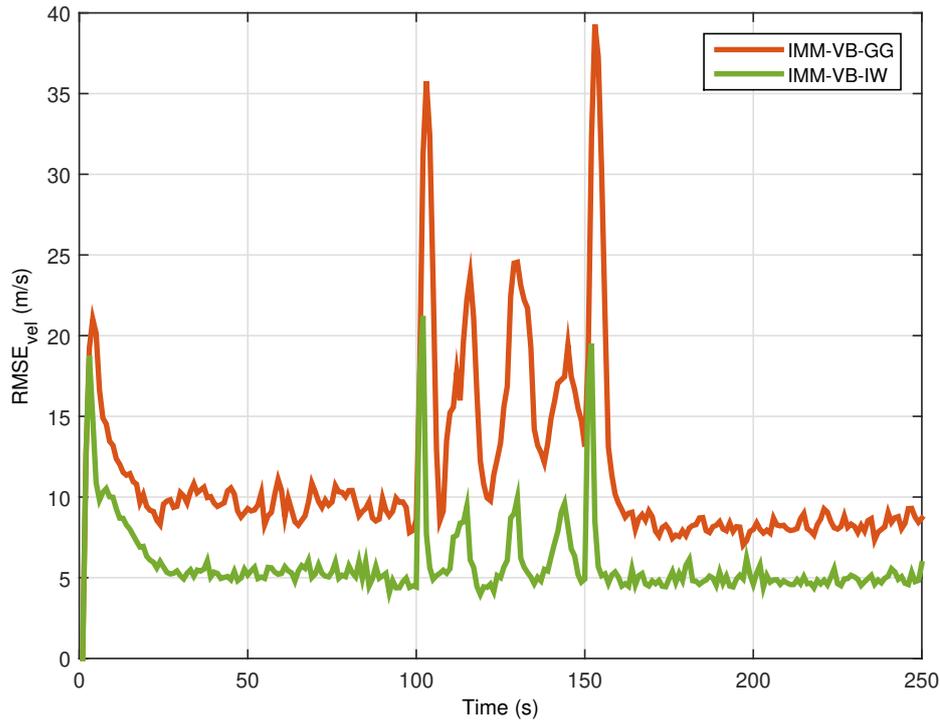


Figure 5.4: RMSEs of the velocity for 250 Monte Carlo run for Case 1. RMSEs of the velocity by IMM-VB-GG is shown by orange line and RMSEs of the velocity by IMM-VB-IW is shown by green line.

MC run. Furthermore, mode probability graphs of IMM-VB-GG and IMM-VB-IW algorithms for Case 2 are given in figures 5.11 and 5.12, respectively.

Figure 5.13 and Figure 5.14 show respectively the RMSEs of position and velocity for Case 3. The ARMSEs of position and velocity for IMM-VB-GG and IMM-VB-IW for Case 3 are given in Table 5.3. In Figure 5.15, the effective measurement noise covariance, the estimated measurement noise covariance by IMM-VB-IW and the effective covariance calculated by IMM-VB-GG used in the measurement update of the state in VB iterations for Case 3 are shown for any time interval of a single MC run. Furthermore, mode probability graphs of IMM-VB-GG and IMM-VB-IW algorithms for Case 3 are given in figures 5.16 and 5.17, respectively.

Figure 5.18 and Figure 5.19 show respectively the RMSEs of position and velocity for Case 4. The ARMSEs of position and velocity for IMM-VB-GG and IMM-VB-

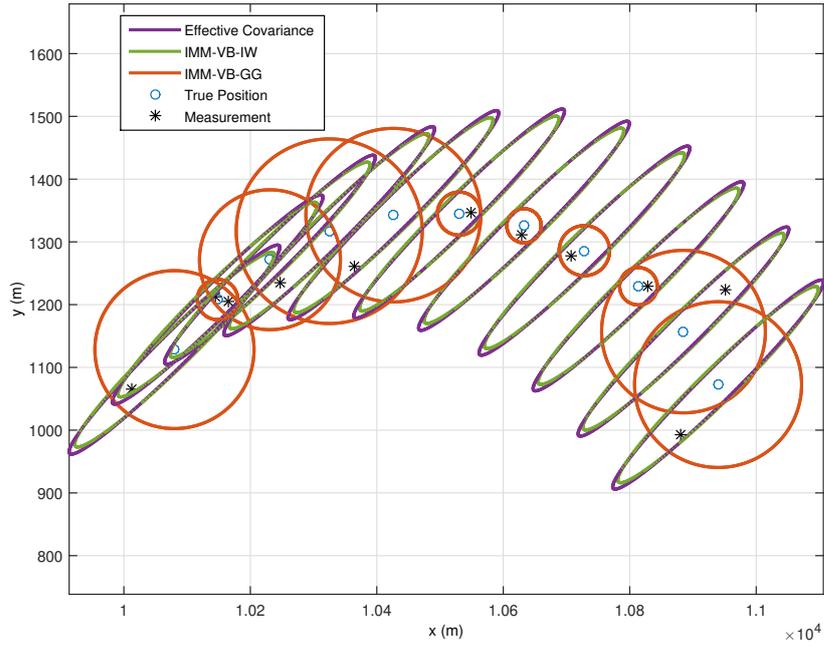


Figure 5.5: Measurement noise covariance tracking of the algorithms IMM-VB-GG and IMM-VB-IW for Case 1 (for any time interval of a single MC run). The effective measurement noise covariances are shown by purple ellipses, the estimated measurement noise covariances by IMM-VB-IW are shown by green ellipses and the effective covariance calculated by IMM-VB-GG used in the measurement update of the state in VB iterations are shown by orange circles. The blue circles and black stars show the true positions and the measurements, respectively.

IW for Case 4 are given in Table 5.4. In Figure 5.20, the effective measurement noise covariance, the estimated measurement noise covariance by IMM-VB-IW and the effective covariance calculated by IMM-VB-GG used in the measurement update of the state in VB iterations for Case 4 are shown for any time interval of a single MC run. Furthermore, mode probability graphs of IMM-VB-GG and IMM-VB-IW algorithms for Case 4 are given in figures 5.21 and 5.22, respectively.

Figure 5.23 and Figure 5.24 show respectively the RMSEs of position and velocity for Case 5. The ARMSEs of position and velocity for IMM-VB-GG and IMM-VB-IW for Case 5 are given in Table 5.5. In Figure 5.25, the effective measurement noise covariance, the estimated measurement noise covariance by IMM-VB-IW and

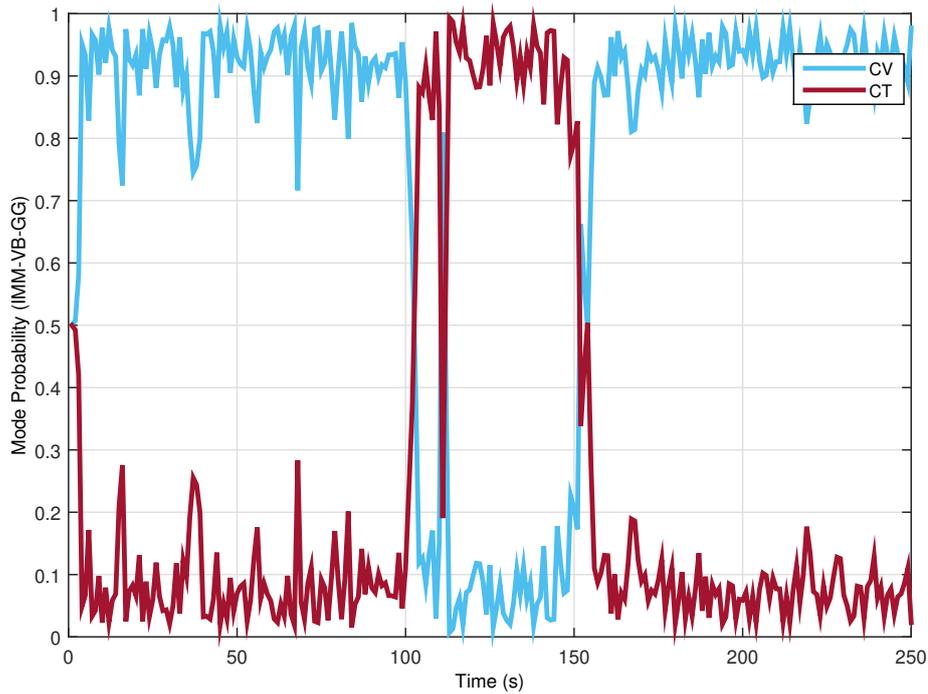


Figure 5.6: Mode probabilities of CV and CT models for IMM-VB-GG for Case 1. The mode probability of CV model is shown by blue line and the mode probability of CT model is shown by red line.

the effective covariance calculated by IMM-VB-GG used in the measurement update of the state in VB iterations for Case 5 are shown for any time interval of a single MC run. Furthermore, mode probability graphs of IMM-VB-GG and IMM-VB-IW algorithms for Case 5 are given in figures 5.26 and 5.27, respectively.

As seen in Figures 5.3, 5.4, 5.8 and 5.9, IMM-VB-IW algorithm has smaller RMSEs than IMM-VB-GG algorithm for the measurement noise characteristics in Case 1 and Case 2. It can be understood from Tables 5.1 and 5.2 that the ARMSEs of position and velocity from IMM-VB-IW algorithm are respectively decreased by approximately 50% and 60% with respect to IMM-VB-GG algorithm for Case 1 and Case 2. However, for Case 3-5, IMM-VB-GG algorithm has smaller RMSEs and ARMSEs, i.e., has better estimation accuracy, than IMM-VB-IW algorithm as seen in Figures 5.13, 5.14 and Table 5.3. Since the measurement noise characteristics in Case 1 and Case 2 show the behavior of unknown covariance, IMM-VB-IW provides

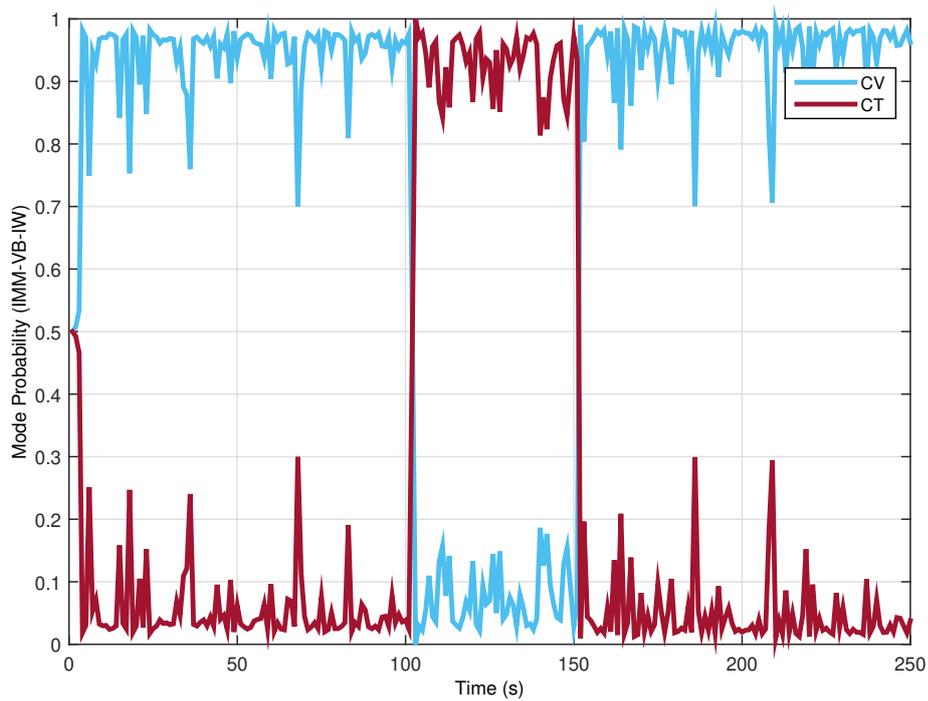


Figure 5.7: Mode probabilities of CV and CT models for IMM-VB-IW for Case 1. The mode probability of CV model is shown by blue line and the mode probability of CT model is shown by red line.

the best estimation accuracy for these cases. However, Case 3-5 show outlier behavior unlike Case 1 and Case 2 so IMM-VB-GG provides better estimation accuracy than IMM-VB-IW algorithm.

In the previous chapter, fixed prior VB-IW algorithm is proposed to increase the estimation accuracy of VB-IW algorithm for the presence of outliers and it is shown that by this approach, the RMSEs and ARMSEs of VB-IW algorithm approaches VB-GG algorithm. Therefore, if this approach is applied to IMM-VB-IW algorithm, it provides as good estimation accuracy as IMM-VB-GG algorithm.

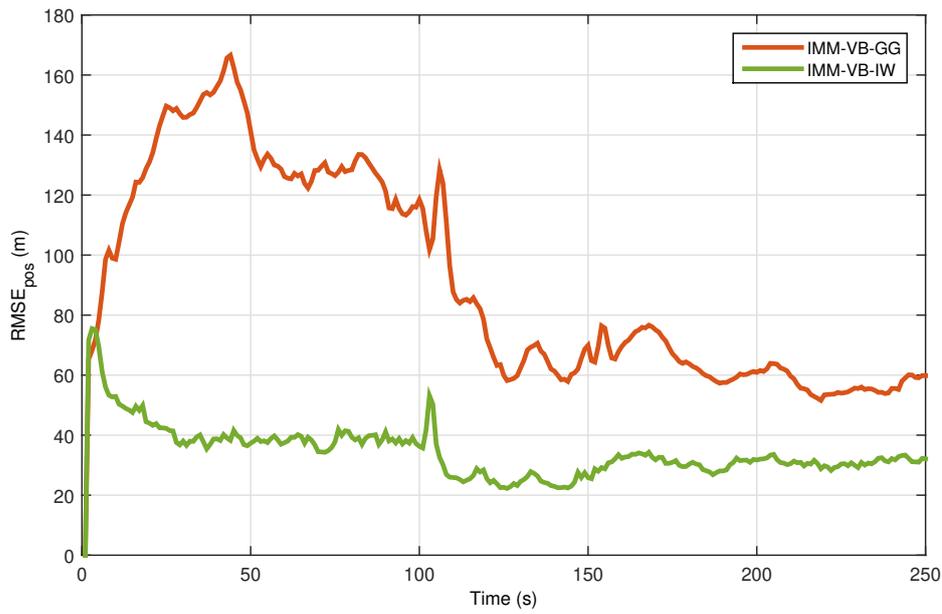


Figure 5.8: RMSEs of the position for 250 Monte Carlo run for Case 2. RMSEs of the position by IMM-VB-GG is shown by orange line and RMSEs of the position by IMM-VB-IW is shown by green line.

Table 5.2: ARMSEs of IMM-VB-GG and IMM-VB-IW for Case 2

	ARMSE of the position (m)	ARMSE of the velocity (m/s)
IMM-VB-GG	97.65	21.21
IMM-VB-IW	35.44	9.26

Table 5.3: ARMSEs of IMM-VB-GG and IMM-VB-IW for Case 3

	ARMSE of the position (m)	ARMSE of the velocity (m/s)
IMM-VB-GG	11.35	5.78
IMM-VB-IW	25.78	11.53

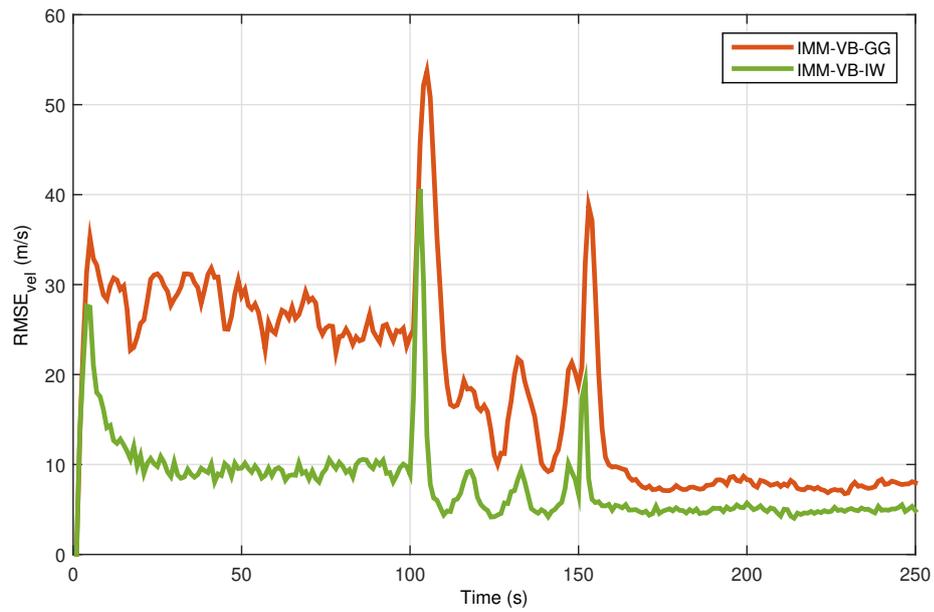


Figure 5.9: RMSEs of the velocity for 250 Monte Carlo run for Case 2. RMSEs of the velocity by IMM-VB-GG is shown by orange line and RMSEs of the velocity by IMM-VB-IW is shown by green line.

Table 5.4: ARMSEs of IMM-VB-GG and IMM-VB-IW for Case 4

	ARMSE of the position (m)	ARMSE of the velocity (m/s)
IMM-VB-GG	10.21	5.39
IMM-VB-IW	95.25	7.58

Table 5.5: ARMSEs of IMM-VB-GG and IMM-VB-IW for Case 5

	ARMSE of the position (m)	ARMSE of the velocity (m/s)
IMM-VB-GG	10.29	5.38
IMM-VB-IW	84.24	10.16

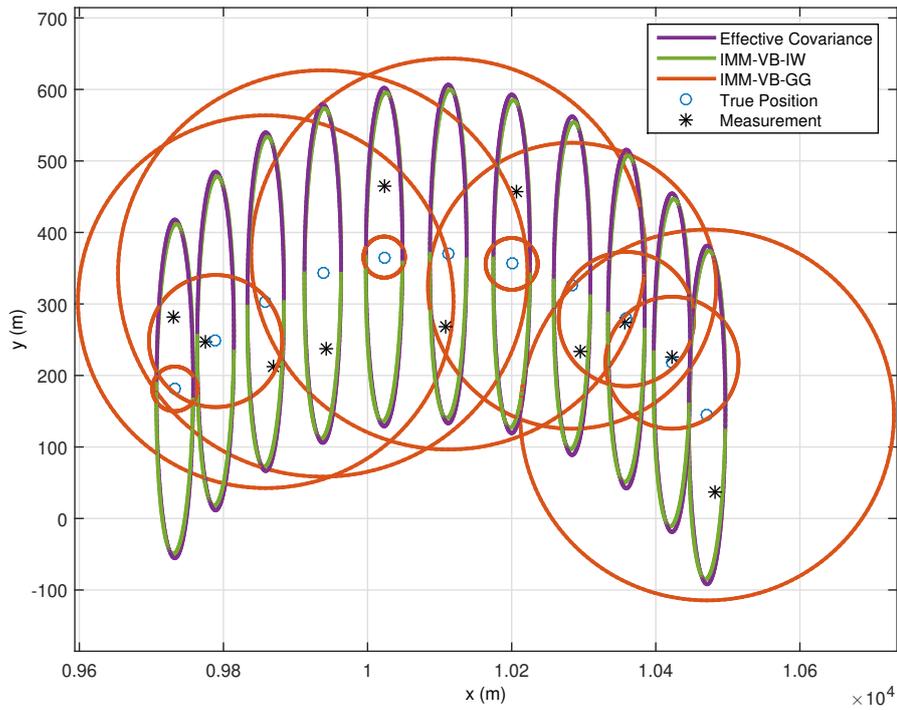


Figure 5.10: Measurement noise covariance tracking of the algorithms IMM-VB-GG and IMM-VB-IW for Case 2 (for any time interval of a single MC run). The effective measurement noise covariances are shown by purple ellipses, the estimated measurement noise covariances by IMM-VB-IW are shown by green ellipses and the effective covariance calculated by IMM-VB-GG used in the measurement update of the state in VB iterations are shown by orange circles.. The blue circles and black stars show the true positions and the measurements, respectively.

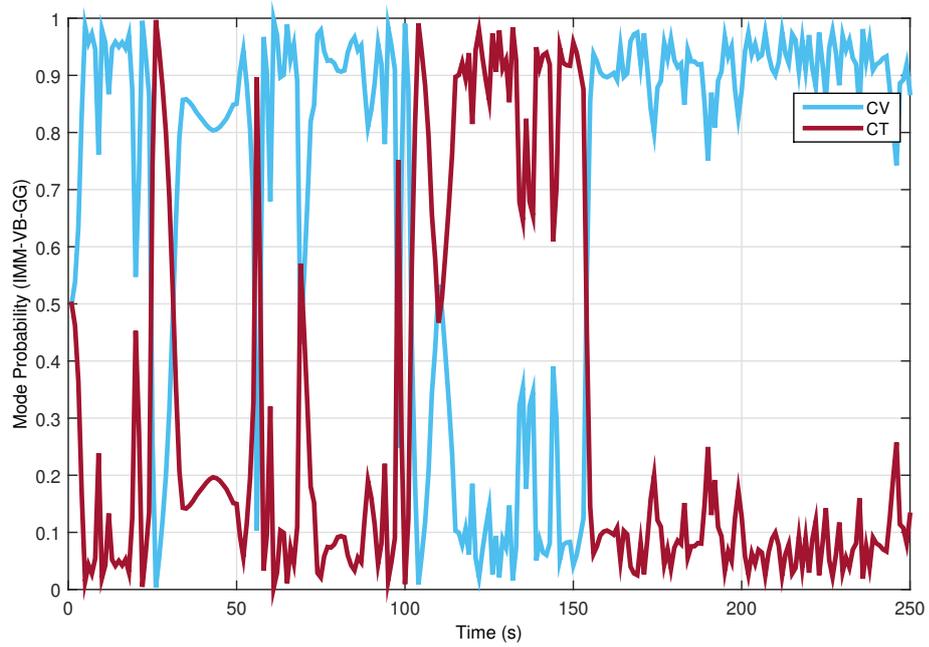


Figure 5.11: Mode probabilities of CV and CT models for IMM-VB-GG for Case 2. The mode probability of CV model is shown by blue line and the mode probability of CT model is shown by red line.

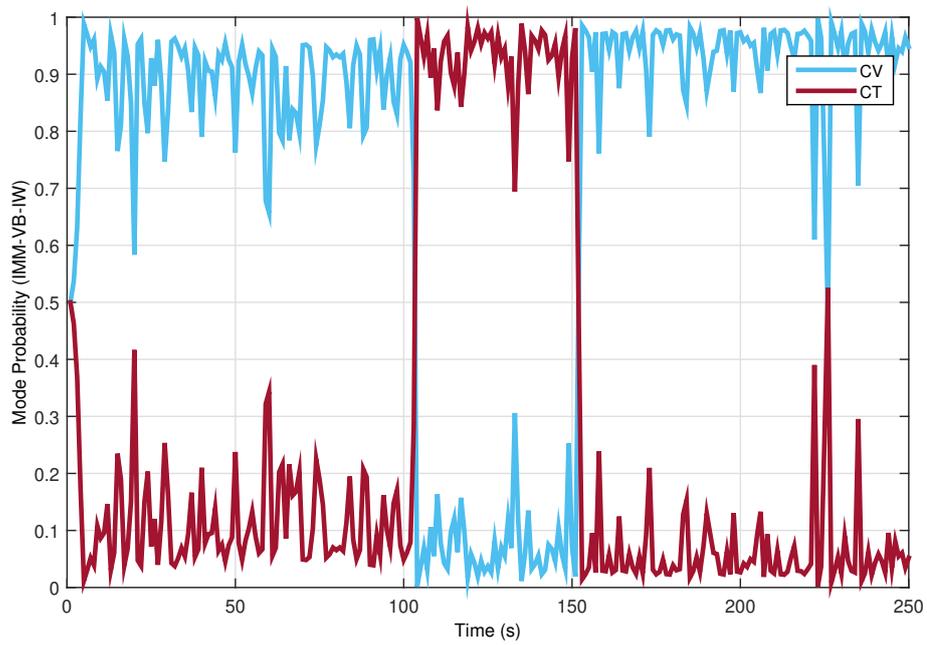


Figure 5.12: Mode probabilities of CV and CT models for IMM-VB-IW for Case 2. The mode probability of CV model is shown by blue line and the mode probability of CT model is shown by red line.

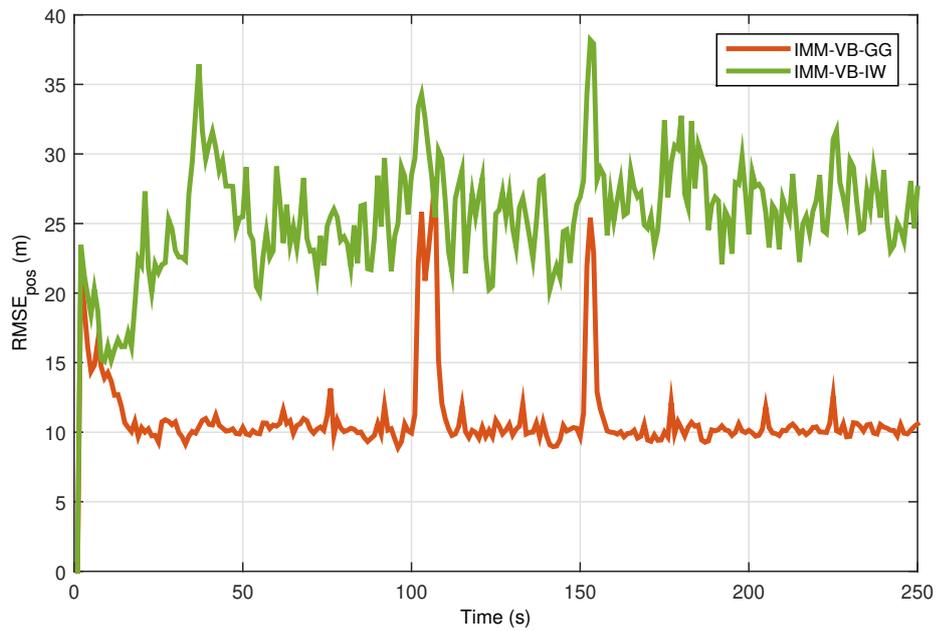


Figure 5.13: RMSEs of the position for 250 Monte Carlo run for Case 3. RMSEs of the position by IMM-VB-GG is shown by orange line and RMSEs of the position by IMM-VB-IW is shown by green line.

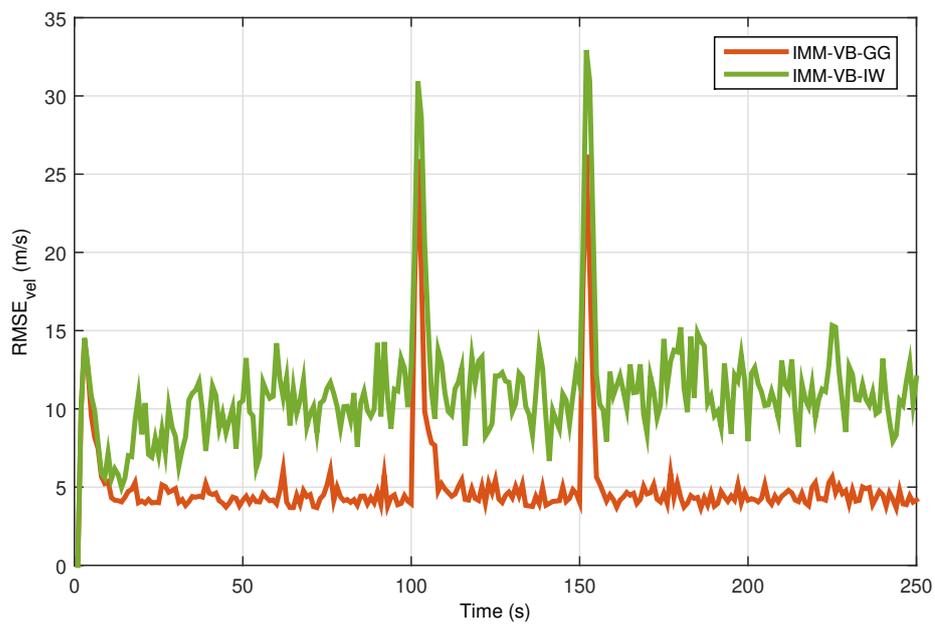


Figure 5.14: RMSEs of the velocity for 250 Monte Carlo run for Case 3. RMSEs of the velocity by IMM-VB-GG is shown by orange line and RMSEs of the velocity by IMM-VB-IW is shown by green line.

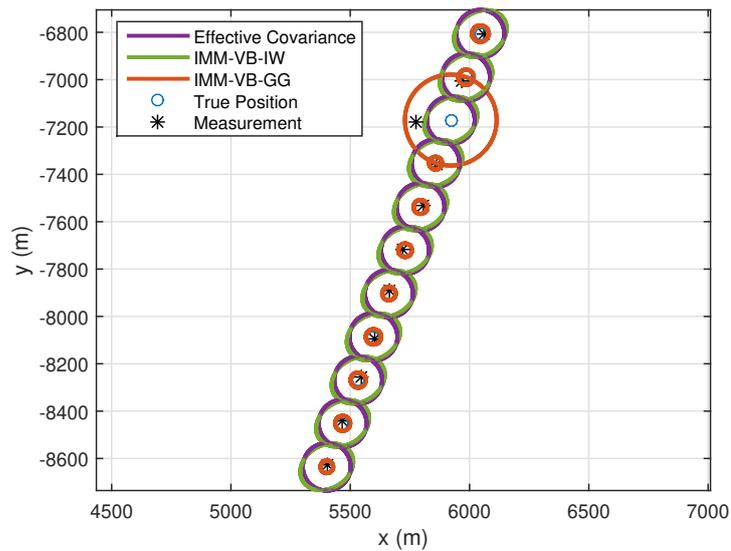


Figure 5.15: Measurement noise covariance tracking of the algorithms IMM-VB-GG and IMM-VB-IW for Case 3 (for any time interval of a single MC run). The effective measurement noise covariances are shown by purple ellipses, the estimated measurement noise covariances by IMM-VB-IW are shown by green ellipses and the effective covariance calculated by VB-GG used in the measurement update of the state in VB iterations are shown by orange circles. The blue circles and black stars show the true positions and the measurements, respectively.

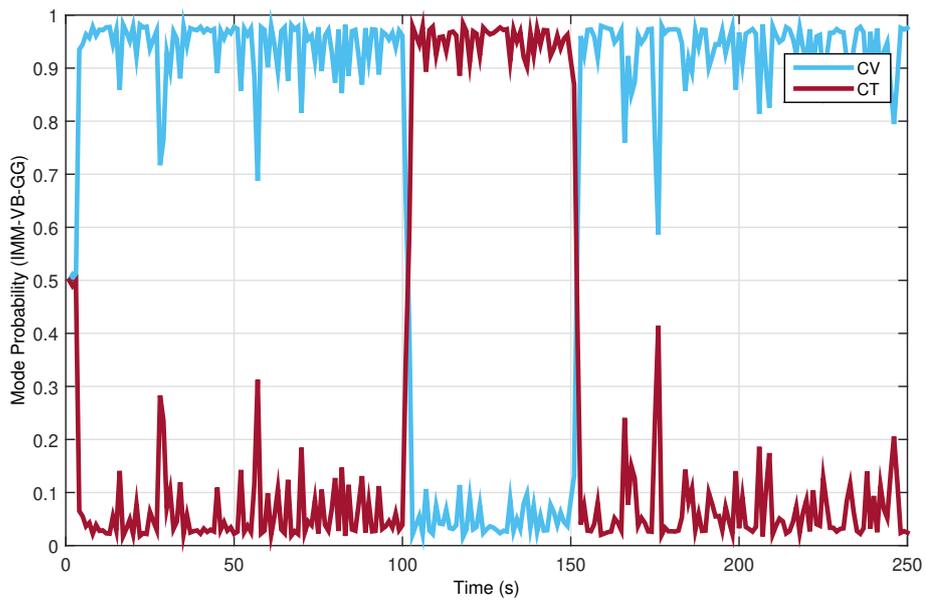


Figure 5.16: Mode probabilities of CV and CT models for IMM-VB-GG for Case 3. The mode probability of CV model is shown by blue line and the mode probability of CT model is shown by red line.

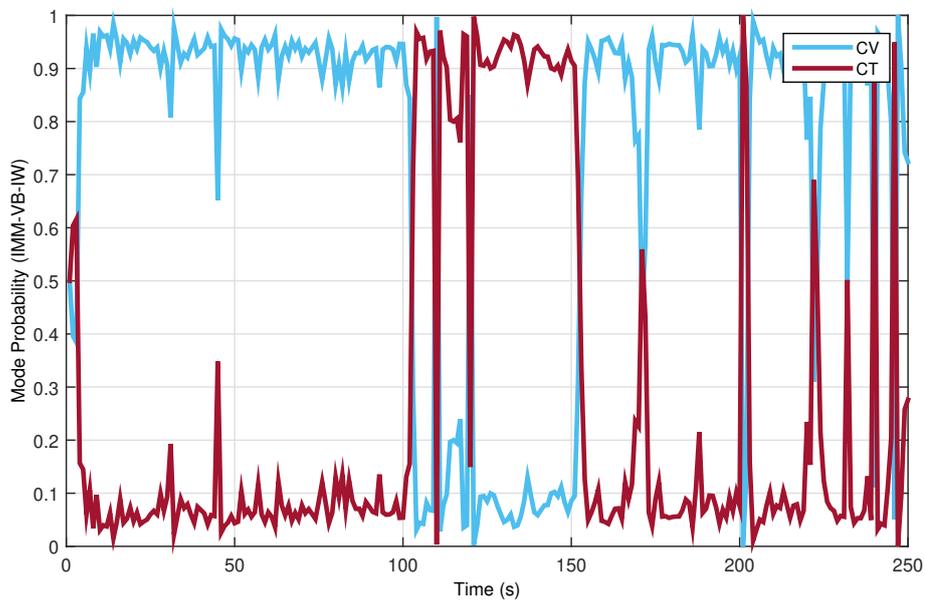


Figure 5.17: Mode probabilities of CV and CT models for IMM-VB-IW for Case 3. The mode probability of CV model is shown by blue line and the mode probability of CT model is shown by red line.

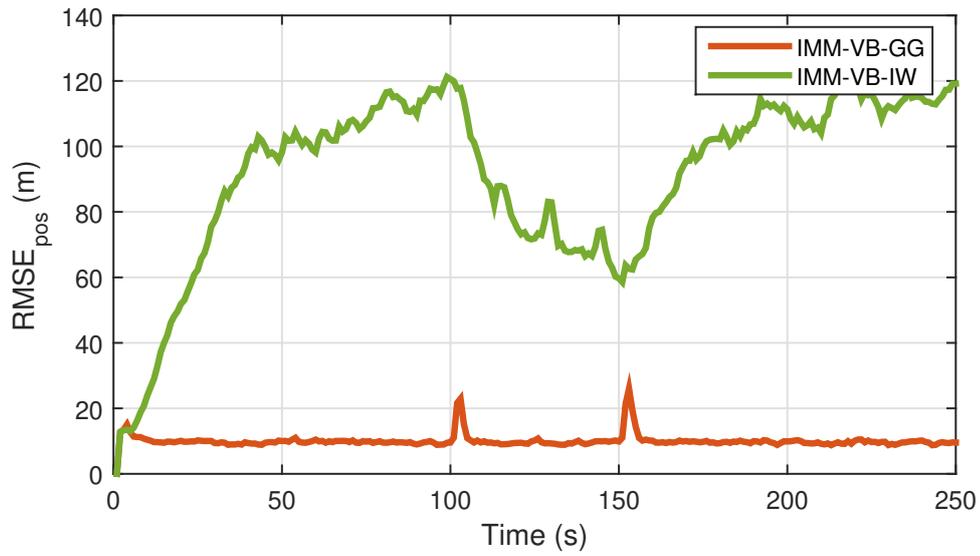


Figure 5.18: RMSEs of the position for 250 Monte Carlo run for Case 4. RMSEs of the position by IMM-VB-GG is shown by orange line and RMSEs of the position by IMM-VB-IW is shown by green line.

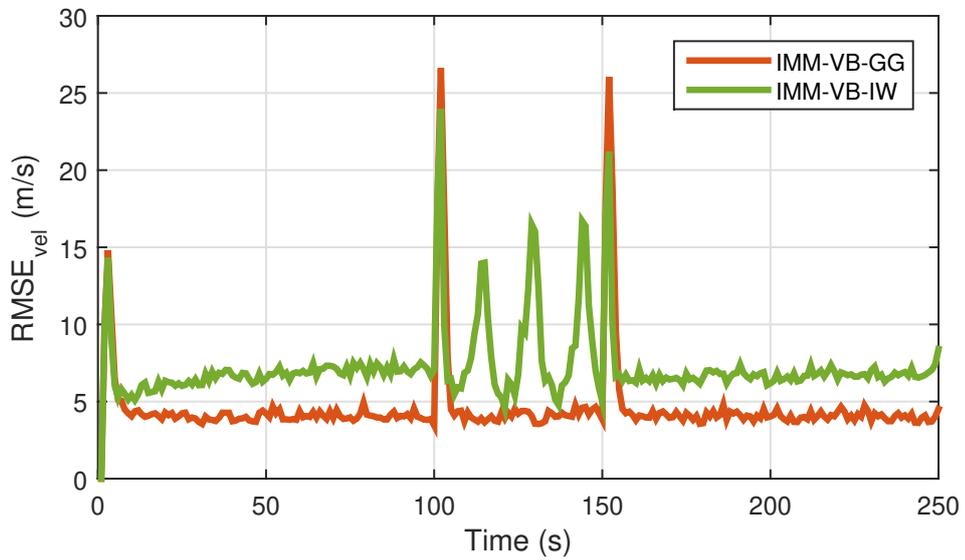


Figure 5.19: RMSEs of the velocity for 250 Monte Carlo run for Case 4. RMSEs of the velocity by IMM-VB-GG is shown by orange line and RMSEs of the velocity by IMM-VB-IW is shown by green line.

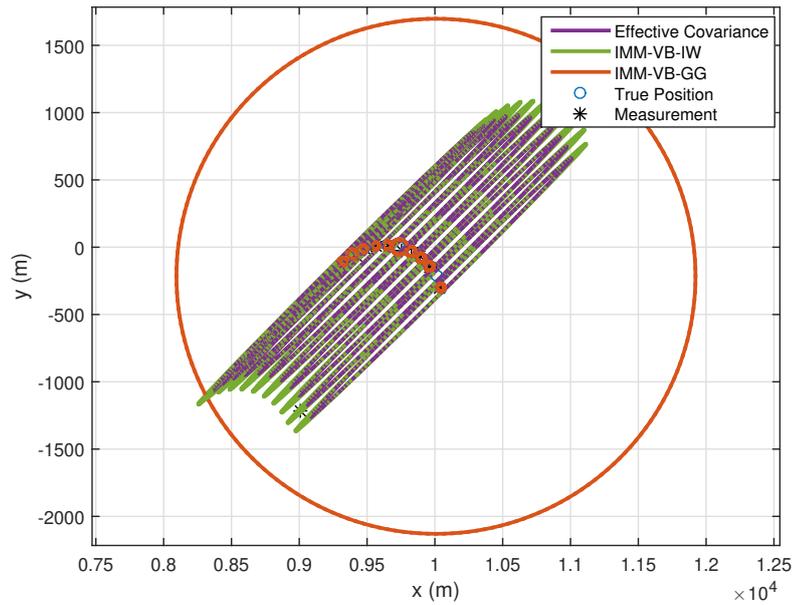


Figure 5.20: Measurement noise covariance tracking of the algorithms IMM-VB-GG and IMM-VB-IW for Case 4 (for any time interval of a single MC run). The effective measurement noise covariances are shown by purple ellipses, the estimated measurement noise covariances by IMM-VB-IW are shown by green ellipses and the effective covariance calculated by VB-GG used in the measurement update of the state in VB iterations are shown by orange circles. The blue circles and black stars show the true positions and the measurements, respectively.

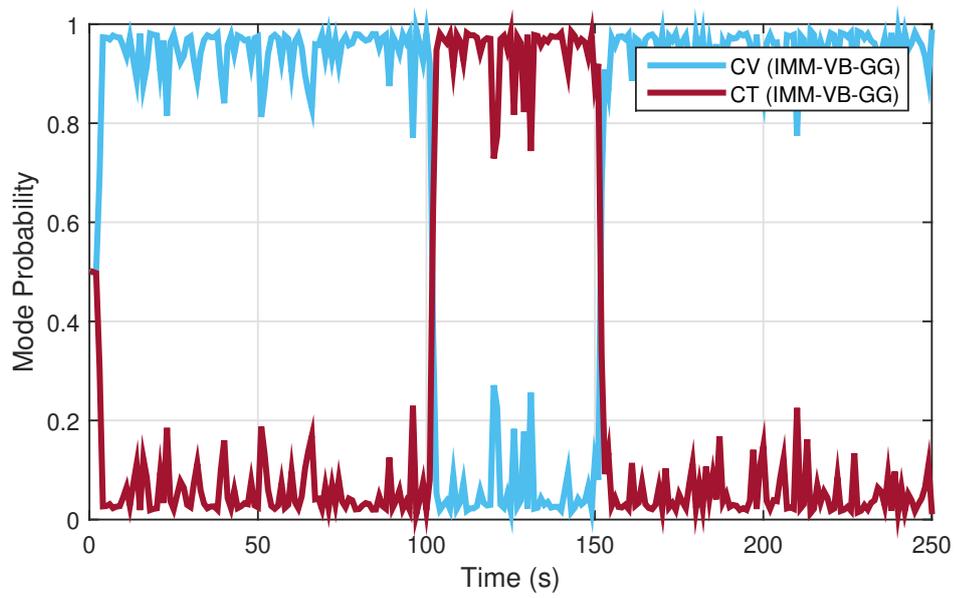


Figure 5.21: Mode probabilities of CV and CT models for IMM-VB-GG for Case 4. The mode probability of CV model is shown by blue line and the mode probability of CT model is shown by red line.

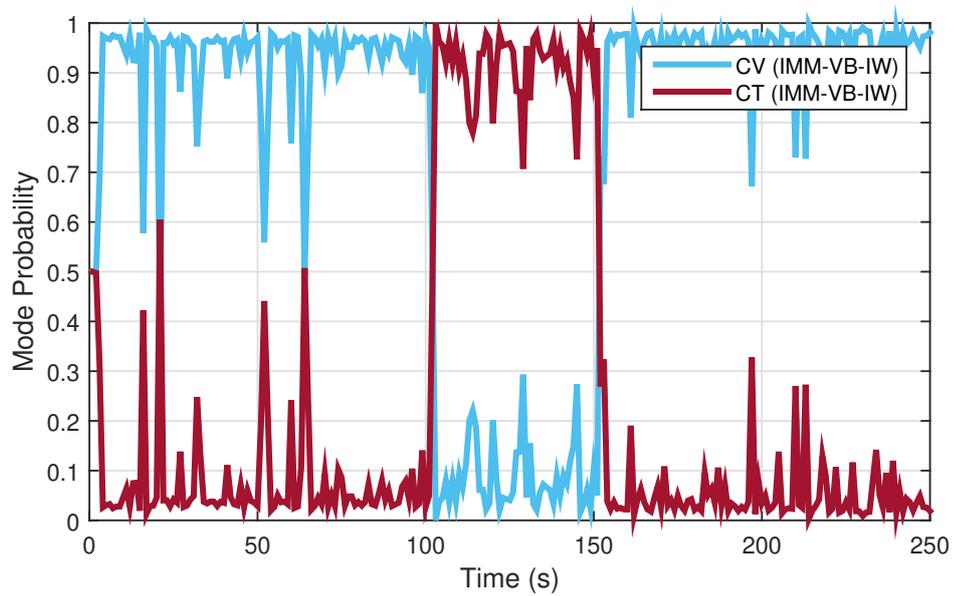


Figure 5.22: Mode probabilities of CV and CT models for IMM-VB-IW for Case 4. The mode probability of CV model is shown by blue line and the mode probability of CT model is shown by red line.

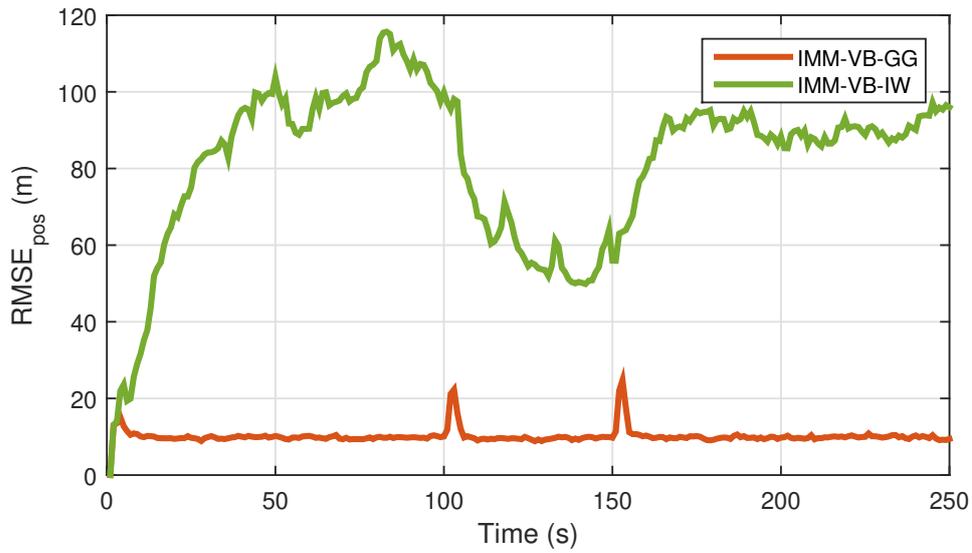


Figure 5.23: RMSEs of the position for 250 Monte Carlo run for Case 5. RMSEs of the position by IMM-VB-GG is shown by orange line and RMSEs of the position by IMM-VB-IW is shown by green line.

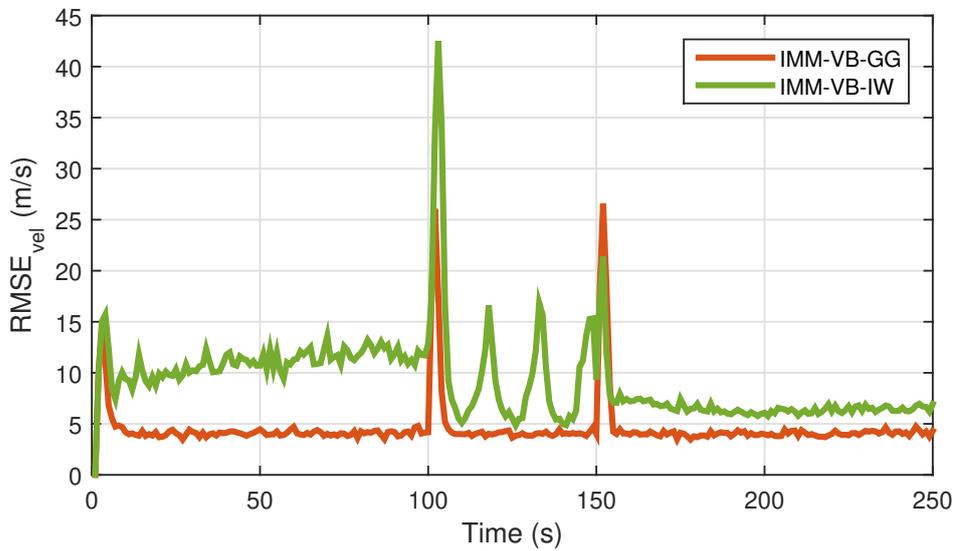


Figure 5.24: RMSEs of the velocity for 250 Monte Carlo run for Case 5. RMSEs of the velocity by IMM-VB-GG is shown by orange line and RMSEs of the velocity by IMM-VB-IW is shown by green line.

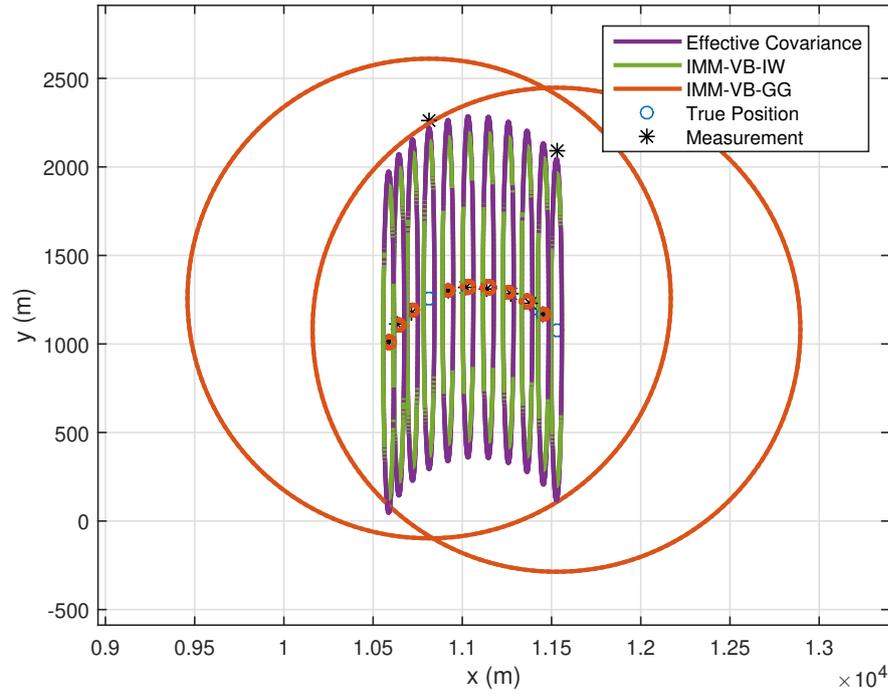


Figure 5.25: Measurement noise covariance tracking of the algorithms IMM-VB-GG and IMM-VB-IW for Case 5 (for any time interval of a single MC run). The effective measurement noise covariances are shown by purple ellipses, the estimated measurement noise covariances by IMM-VB-IW are shown by green ellipses and the effective covariance calculated by VB-GG used in the measurement update of the state in VB iterations are shown by orange circles. The blue circles and black stars show the true positions and the measurements, respectively.

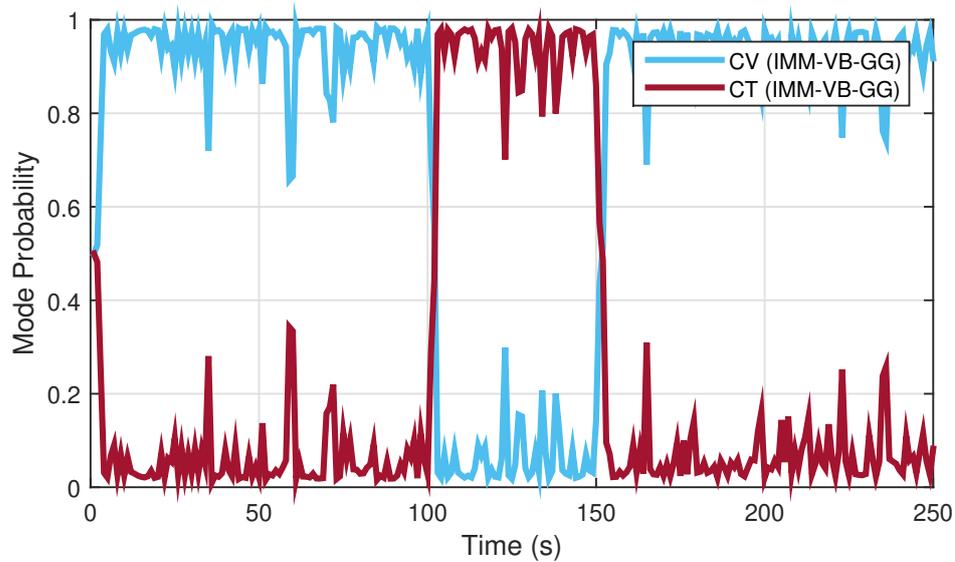


Figure 5.26: Mode probabilities of CV and CT models for IMM-VB-GG for Case 5. The mode probability of CV model is shown by blue line and the mode probability of CT model is shown by red line.

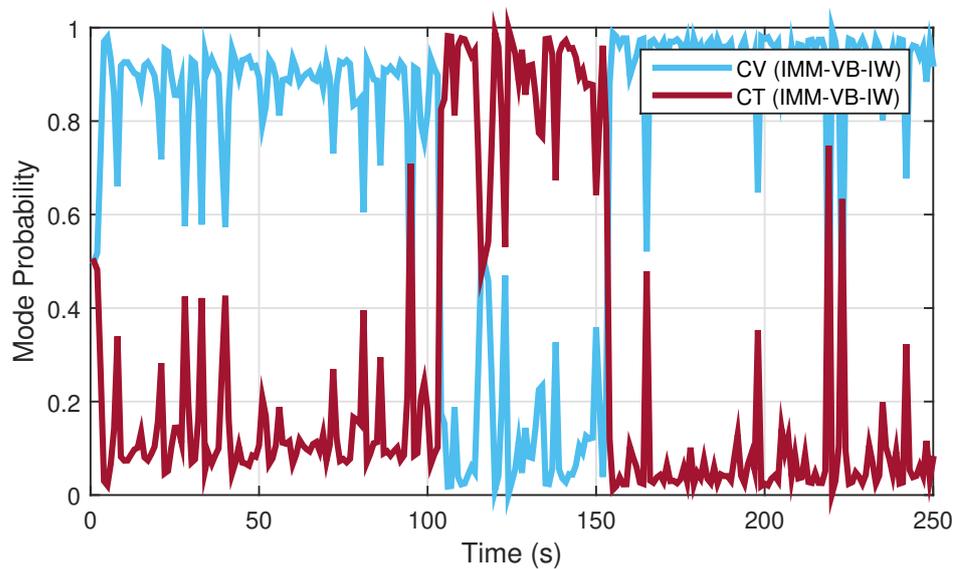


Figure 5.27: Mode probabilities of CV and CT models for IMM-VB-IW for Case 5. The mode probability of CV model is shown by blue line and the mode probability of CT model is shown by red line.

CHAPTER 6

CONCLUSION

Kalman filter (KF) is one of the most used algorithms in target tracking area due to its ease of application and low computational complexity. Although it provides the best linear unbiased estimate for linear Gaussian state-space models (SSMs), its estimation performance degrades when outliers occur. For such cases, different algorithms are dedicated to handle heavy-tailed noise. Until today, numerous studies have been carried out to solve the filtering problem of linear systems with heavy-tailed noises and to provide robustness towards outliers. In this thesis, a sub-class of outlier robust filters are investigated and their multiple-model extensions are derived based on interacting multiple model (IMM) approach.

In the first part of this study, Student's-t filter (STF) proposed in [44] is investigated. In this filter, the process and measurement noises are described as t-distributed that has heavier tails than Gaussian distribution. It is demonstrated that STF provides better estimation accuracy than KF under heavy-tailed process and measurement noise assumption. In addition, a multiple-model extension of STF (IMM-STF) is derived based on IMM approach and it is compared with conventional IMM algorithm on a simulation of a moving target in 2-D space. The simulation results show that IMM-STF has smaller RMSEs than the conventional IMM algorithm for a multiple-model system with heavy-tailed process and measurement noise.

In the second part, two VB algorithms which utilize Gamma-Gaussian (VB-GG) and Inverse Wishart (VB-IW) priors are examined in detail and their derivations are given. In VB-GG algorithm, the likelihood PDF is approximated as Student's-t distributed but posterior PDF is approximated as Gaussian distributed. The t-distributed likelihood PDF is expressed using Gamma-Gaussian approach which is the representation

of t distribution as an infinite mixture of Gaussians by defining auxiliary variable that is Gamma distributed. The posterior PDF and the auxiliary variable are inferred via VB approach. In VB-IW algorithm, VB approximations are used to estimate the state and the unknown measurement noise covariance by choosing Inverse Wishart prior for measurement noise covariance matrix. Since an outlier can be viewed as a result of inaccurate noise covariance, VB-IW algorithm can also be tested to handle outliers. VB-GG and VB-IW algorithms are implemented and simulated for a linear SSM for five different heavy-tailed measurement noise characteristics and the performances are compared. The simulation results show that the performance of VB-IW can match VB-GG in presence of outliers. Furthermore, by using its ability to estimate effective noise covariance, it can outperform VB-GG in a number of scenarios.

In the last part, multiple-model extensions of VB-GG (IMM-VB-GG) and VB-IW (IMM-VB-IW) algorithms are derived based on IMM approach. These algorithms are tested on a simulation for tracking a target moving according to linear multiple-model SSM for five different heavy-tailed measurement noise characteristics and the performances are compared. According to simulation results, it is shown that the multiple-model extension of VB-IW algorithm (IMM-VB-IW) provides robustness towards outliers for multiple model systems. In addition, it can outperform IMM-VB-GG algorithm in terms of RMSEs and ARMSEs.

The contributions of this thesis study are the derivation of multiple-model extension of STF, the comparison of VB-GG and VB-IW algorithms and the derivation of multiple-model extension of VB-IW algorithm.

REFERENCES

- [1] G. Agamennoni, J. Nieto, and E. Nebot. Approximate inference in state-space models with heavy-tailed noise. *IEEE Transactions on Signal Processing*, 60:5024–5037, October 2012.
- [2] G. Agamennoni, J. I. Nieto, and E. M. Nebot. An outlier-robust Kalman filter. *2011 IEEE International Conference on Robotics and Automation*, pages 1551–1558, 2011.
- [3] B. Anderson and J. Moore. *Optimal Filtering*. Prentice-Hall, 1979.
- [4] T. Ardeshiri and E. Özkan. An adaptive PHD filter for tracking with unknown sensor characteristics. In *Proceedings of the 16th International Conference on Information Fusion*, pages 1736–1743, July 2013.
- [5] T. Ardeshiri, E. Özkan, U. Orguner, and F. Gustafsson. Approximate Bayesian smoothing with unknown process and measurement noise covariances. *IEEE Signal Processing Letters*, 22(12):2450–2454, December 2015.
- [6] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp. A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. *IEEE Transactions on Signal Processing*, 50(2):174–188, February 2002.
- [7] Y. Bar-Shalom, K. C. Chang, and H. A. P. Blom. Tracking a maneuvering target using input estimation versus the interacting multiple model algorithm. *IEEE Transactions on Aerospace and Electronic Systems*, 25(2):296–300, March 1989.
- [8] Y. Bar-Shalom, T. Kirubarajan, and X.-R. Li. *Estimation with Applications to Tracking and Navigation*. John Wiley & Sons, Inc., New York, USA, 2002.
- [9] T. Bayes. An essay towards solving a problem in the doctrine of chances. *Phil. Trans. of the Royal Soc. of London*, 53:370–418, 1763.

- [10] J. M. Bernardo. Algorithm as 103: Psi (Digamma) function. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 25(3):315–317, 1976.
- [11] C. M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag, Berlin, Heidelberg, 2006.
- [12] H. Blom and Y. Bar-Shalom. The interacting multiple model algorithm for systems with Markovian switching coefficient. *IEEE Transactions on Automatic Control*, 33:780 – 783, September 1988.
- [13] S. Challa, M. R. Morelande, D. Mušicki, and R. J. Evans. *Fundamentals of Object Tracking*. Cambridge University Press, 2011.
- [14] Z. Chen. Bayesian filtering: From Kalman filters to particle filters, and beyond. *Statistics*, 182, January 2003.
- [15] A. Doucet, S. Godsill, and C. Andrieu. On sequential Monte Carlo sampling methods for Bayesian filtering. *Statistics and Computing*, 10(3):197–208, July 2000.
- [16] M. Feldmann, D. Fraenken, and W. Koch. Tracking of extended objects and group targets using random matrices. *Signal Processing, IEEE Transactions on*, 59:1409 – 1420, 05 2011.
- [17] A. Genovese. The interacting multiple model algorithm for accurate state estimation of maneuvering targets. *Johns Hopkins APL Technical Digest (Applied Physics Laboratory)*, 22:614–623, October 2001.
- [18] F. J. Giron and J. C. Rojano. Bayesian Kalman filtering with elliptically contoured errors. *Biometrika*, 81(2):390–395, June 1994.
- [19] K. Granström and U. Orguner. On the reduction of Gaussian inverse Wishart mixtures. In *2012 15th International Conference on Information Fusion*, pages 2162–2169, July 2012.
- [20] J. Hou, Y. Yang, H. He, and T. Gao. Adaptive dual extended Kalman filter based on variational Bayesian approximation for joint estimation of lithium-ion battery state of charge and model parameters. *Applied Sciences*, 9:1726, 04 2019.

- [21] Y. Huang, Y. Zhang, N. Li, and J. Chambers. A robust Gaussian approximate fixed-interval smoother for nonlinear systems with heavy-tailed process and measurement noises. *IEEE Signal Processing Letters*, 23(4):468–472, April 2016.
- [22] Y. Huang, Y. Zhang, Z. Wu, N. Li, and J. Chambers. A novel robust Student’s t-based Kalman filter. *IEEE Transactions on Aerospace and Electronic Systems*, 53:1545–1554, 2017.
- [23] Y. Huang, Y. Zhang, Z. Wu, N. Li, and J. A. Chambers. A novel adaptive Kalman filter with inaccurate process and measurement noise covariance matrices. *IEEE Transactions on Automatic Control*, 63(2):594–601, 2018.
- [24] R. E. Kalman. A new approach to linear filtering and prediction problems. *ASME Journal of Basic Engineering*, 1960.
- [25] S. F. Kara and E. Özkan. Multi-ellipsoidal extended target tracking using sequential Monte Carlo. *2018 21st International Conference on Information Fusion (FUSION)*, pages 1–8, 2018.
- [26] S. Kotz and S. Nadarajah. *Multivariate T-Distributions and Their Applications*. Cambridge University Press, 2004.
- [27] W. Li. State estimation for jump Markov linear systems by variational Bayesian approximation. *IET Control Theory & Applications*, 6:319–326(7), January 2012.
- [28] X. R. Li and Y. Bar-Shalom. Performance prediction of the interacting multiple model algorithm. *IEEE Transactions on Aerospace and Electronic Systems*, 29(3):755–771, July 1993.
- [29] X. R. Li and Y. Bar-Shalom. A recursive multiple model approach to noise identification. *IEEE Transactions on Aerospace and Electronic Systems*, 30(3):671–684, July 1994.
- [30] J. Ma, H. Lan, Z. Wang, X. Wang, Q. Pan, and B. Moran. Improved adaptive Kalman filter with unknown process noise covariance. pages 1–5, July 2018.

- [31] R. Martin and C. Masreliez. Robust estimation via stochastic approximation. *IEEE Transactions on Information Theory*, 21(3):263–271, May 1975.
- [32] J. Masreliez. Approximate non-Gaussian filtering with linear state and observation relation. *IEEE Transactions on Automatic Control*, 20:107 – 110, March 1975.
- [33] J. Masreliez and R. Martin. Robust Bayesian estimation for the linear model and robustifying the Kalman filter. *IEEE Transactions on Automatic Control*, 22:361 – 371, July 1977.
- [34] R. J. Meinhold and N. D. Singpurwalla. Robustification of Kalman filter models. *Journal of the American Statistical Association*, 84(406):479–486, 1989.
- [35] H. Nurminen, T. Ardeshiri, R. Piché, and F. Gustafsson. Robust inference for state-space models with skewed measurement noise. *IEEE Signal Processing Letters*, 22(11):1898–1902, Nov 2015.
- [36] S. W. Nydick. The Wishart and inverse Wishart distributions. In *Electron. J. Statist.*, volume 6, pages 1–19, May 2012.
- [37] U. Orguner. A variational measurement update for extended target tracking with random matrices. *IEEE Transactions on Signal Processing*, 60(7):3827–3834, July 2012.
- [38] H. A. P. Blom. An efficient filter for abruptly changing systems. In *The 23rd IEEE Conference on Decision and Control*, pages 656–658, Dec 1984.
- [39] Y. Peng, W. Panlong, and H. Shan. An IMM-VB algorithm for hypersonic vehicle tracking with heavy tailed measurement noise. In *2018 International Conference on Control, Automation and Information Sciences (ICCAIS)*, pages 169–174, Oct 2018.
- [40] X. Rong Li and V. P. Jilkov. Survey of maneuvering target tracking. part i. dynamic models. *IEEE Transactions on Aerospace and Electronic Systems*, 39(4):1333–1364, Oct 2003.
- [41] X. Rong Li and V. P. Jilkov. Survey of maneuvering target tracking. part v.

- multiple-model methods. *IEEE Transactions on Aerospace and Electronic Systems*, 41:1255 – 1321, 11 2005.
- [42] M. Roth. On the multivariate t distribution. Technical report, Automatic Control, Linköping University, Linköping, Sweden, April 2013.
- [43] M. Roth, T. Ardeshiri, E. Özkan, and F. Gustafsson. Robust Bayesian filtering and smoothing using Student’s t distribution. <https://arxiv.org/abs/1703.02428>, March 2017.
- [44] M. Roth, E. Özkan, and F. Gustafsson. A Student’s t filter for heavy tailed process and measurement noise. *Acoustics, Speech, and Signal Processing, 1988. ICASSP-88., 1988 International Conference on*, pages 5770–5774, October 2013.
- [45] S. Sarkka. *Bayesian Filtering and Smoothing*. Cambridge University Press, New York, NY, USA, 2013.
- [46] S. Sarkka and A. Nummenmaa. Recursive noise adaptive Kalman filtering by variational Bayesian approximations. *IEEE Transactions on Automatic Control*, 54(3):596–600, March 2009.
- [47] C. Shen, D. Xu, W. Huang, and F. Shen. An interacting multiple model approach for state estimation with non-Gaussian noise using a variational Bayesian method. *Asian Journal of Control*, 17(4):1424–1434, 2015.
- [48] D. J. Tylavsky and G. R. L. Sohie. Generalization of the matrix inversion lemma. *Proceedings of the IEEE*, 74(7):1050–1052, July 1986.
- [49] W. Wei. Interacting multiple model algorithm using skew-t measurement noise and variational Bayesian approach. pages 4384–4388, July 2018.
- [50] M. West. Robust sequential approximate Bayesian estimation. *Journal of the Royal Statistical Society (Ser. B)*, 43:157–166, 01 1981.
- [51] H. Zhu, H. Leung, and Z. He. A variational Bayesian approach to robust sensor fusion based on Student-t distribution. *Information Sciences*, 221:201–214, 02 2013.