State and Parameter Estimation for Dynamic Systems: Some Investigations

Thesis Submitted By

ARITRO DEY

DOCTOR OF PHILOSOPHY (ENGINEERING)

Department of Electrical Engineering
Faculty Council of Engineering & Technology
Jadavpur University
Kolkata, India
2016

JADAVPUR UNIVERSITY
KOLKATA – 700 032, INDIA

INDEX NO.: 150/13/E

1. Title of the Thesis: "State and Parameter Estimation for Dynamic Systems: Some Investigations".

2. Names, Designations and Institution of the Supervisor/s:

DR. SMITA SADHU,
Professor,
Department of Electrical Engineering,
Jadavpur University, Kolkata-700 032, India.

&

DR. TAPAN KUMAR GHOSHAL,
Honorary Emeritus Professor,
Department of Electrical Engineering,
Jadavpur University, Kolkata-700 032, India.

3. List of Publications:

Peer Reviewed International Journals:

1. A. Dey, M. Das, S. Sadhu, T. K. Ghoshal, "Adaptive Divided Difference Filter for Parameter and State Estimation of Nonlinear Systems," *IET Signal Processing,* 2015, vol. 9, issue 4, pp. 369-376
2. A. Dey, S. Sadhu, T. K. Ghoshal, "Adaptive Gauss Hermite Filter for Nonlinear Systems," *IET Science Measurement and Technology,* 2015, vol. 9, issue 8, pp. 1007-1015
3. M. Das, A. Dey, S. Sadhu, T. K. Ghoshal, "Central Difference Filter for Nonlinear State Estimation," *IET Science Measurement and Technology,* 2015, vol. 9, issue 6, pp. 728-735

Book Chapters:

1. A. Dey, S. Sadhu, T. K. Ghoshal, “Adaptive Nonlinear Information Filters for Multiple Sensor Fusion,” Revised Selected Papers of 12th International Conference on Informatics in Control, Automation and Robotics, *Lecture Notes in Electrical Engineering published by Springer Verlag*, 2016, vol. 383, pp. 371-390, DOI 10.1007/978-3-319-31898-1_21

Proceedings of Conferences:

1. A. Dey, S. Sadhu, T. K. Ghoshal, “Joint Estimation of Parameters and States of Nonlinear Systems using Adaptive Divided Difference Filter,” in the Proceedings of *2^{nd} Michael Faraday IET India Summit*, Kolkata, pp. CS.7-CS.11, 2013.

2. A. Dey, S. Sadhu, T. K. Ghoshal, “Adaptive Divided Difference Filter for Nonlinear Systems with Unknown Noise,” in the Proceedings of *International Conference on Control, Instrumentation, Energy and Communication*, University of Calcutta, Kolkata, pp. 640-644, 2014.

3. A. Dey, S. Sadhu, T. K. Ghoshal, “Adaptive Gauss Hermite Filter for Parameter varying Nonlinear Systems,” in the Proceedings of *International Conference on Signal Processing and Communication*, IISc Bangalore, , pp. 1-5, 2014.

4. A. Dey, M. Das, S. Sadhu, T. K. Ghoshal, “Adaptive Gauss Hermite Filter for Parameter and State Estimation of Nonlinear Systems,” in the Proceedings of in 11^{th} *International Conference on Informatics in Control, Automation and Robotics*, Vienna, Austria, vol. 1, pp. 583-589, 2014.

5. A. Dey, S. Sadhu, T. K. Ghoshal, “Adaptive Divided Difference Filter for Nonlinear Systems with Non additive Noise,” in the Proceedings of 3^{rd} *International Conference on Computer, Communication, Control and Information Technology (C3IT)*, Hooghly, WB, pp. 1-5, 2015.

6. A. Dey, S. Sadhu, T. K. Ghoshal, “Multiple Sensor Fusion using Adaptive Divided Difference Information Filter,” in the Proceeding of 12^{th}

International Conference on Informatics in Control, Automation and Robotics, Colmar, France, vol. 1, pp. 398-406, 2015.

7. A. Dey, M. Das, S. Sadhu, T. K. Ghoshal, "Adaptive Unscented Information Filter For Multiple Sensor Fusion," in the proceeding of *Michael Faraday IET India Summit 2015*, Kolkata, pp. 541-545, 2015.
8. M. Das, A. Dey, S. Sadhu, T. K. Ghoshal, "Adaptive Unscented Kalman Filter at the Presence of Non-additive Measurement Noise" in the proceeding of 12th *International Conference on Informatics in Control, Automation and Robotics*, Colmar, France, vol. 1, pp. 614-620, 2015.
9. M. Das, A. Dey, S. Sadhu, T. K. Ghoshal, "Joint Estimation of States and Parameters of a Reentry Ballistic Target Using Adaptive UKF," in the proceeding of Fifth International Symposium on Electronic System Design (ISED), Surathkal, pp. 99 – 103, 2014.

4. List of Patents: NIL

5. List of Presentations in National/ International Conferences/ Seminars/ Workshops:
Same as no. 1-7 of the proceedings of conferences

Certificate from Supervisors

This is to certify that the thesis entitled "**State and Parameter Estimation for Dynamic Systems: Some Investigations**" submitted by **Shri Aritro Dey**, who got his name registered on 28th March, 2013 (Index No.: 150/13/E, Ref. No. : D7/E/266/13), for the award of Ph. D. (Engineering) degree of Jadavpur University is absolutely based upon his own work under the supervision of Dr. Smita Sadhu and Dr. Tapan Kumar Ghoshal and that neither his thesis nor any part of the thesis has been submitted for any degree /diploma or any other academic award anywhere before.

Signature of the supervisor and date with Office Seal
DR. SMITA SADHU,
Professor,
Department of Electrical Engineering,
Jadavpur University, Kolkata-700 032,
India.

Signature of the supervisor and date with Office Seal
DR. TAPAN KUMAR GHOSHAL,
Honorary Emeritus Professor,
Department of Electrical Engineering,
Jadavpur University, Kolkata-700 032,
India.

ABSTRACT

This dissertation presents the outcome of investigations which envisaged to develop improved state and 'combined state and parameter' estimation algorithms for nonlinear signal models (during the contingent situations) where the complete knowledge of process and/or measurement noise covariance are not available.

Variants of "adaptive nonlinear estimators" capable of providing satisfactory estimation results in the face of unknown noise covariance have been proposed in this dissertation. The proposed adaptive nonlinear estimators incorporate adaptation algorithms with which they can implicitly or explicitly, estimate unknown noise covariances along with estimation of states and parameters.

Adaptation algorithms have been mathematically derived following different methods of adaptation which include Maximum Likelihood Estimation (MLE), Covariance Matching method and Maximum a Posteriori (MAP) method.

The adaptive nonlinear estimators which have been proposed in this dissertation are formulated with the help of a general framework for adaptive nonlinear estimators for both additive and non additive Gaussian noise.

The proposed new algorithms have been formulated and characterized with Monte Carlo simulation using nontrivial plant models.

The general framework mentioned as above, is extended to formulate alternative versions of adaptive nonlinear estimators (in the information filter form). Performance of such adaptive nonlinear information filters are demonstrated for multiple sensor fusion.

The contribution of this work may be categorized as follows:

- Proposing general frameworks for $\boldsymbol{Q}$ adaptive and $\boldsymbol{R}$ adaptive nonlinear state estimators (for respectively unknown process noise covariance $\boldsymbol{Q}$ and unknown measurement noise covariance $\boldsymbol{R}$) and demonstrating applicability of such filters with specific examples.

- Derivation of the nonlinear versions of adaptation algorithms for $\boldsymbol{R}$-adaptive nonlinear estimators following the Maximum Likelihood Estimation (MLE) method

respectively utilizing innovation and residual sequences. The latter version is important as it automatically ensures positive definiteness of the adapted ***R***- matrix.

- Modification of the existing Maximum a Posteriori (MAP) based algorithms for adaptive nonlinear estimators (both ***R***-adaptive and ***Q***-adaptive) with reasonable simplifying assumptions which illustrate that the adaptation algorithms after modification match well with those obtained by the MLE method and the intuitive Covariance Matching method.
- Proposing and characterizing algorithms for different versions of ***Q***-adaptive and ***R***-adaptive Divided Difference filter (ADDF).
- Proposing and formulation of algorithms for several new variety of ***Q*** and ***R*** adaptive nonlinear estimators, viz. Adaptive Gauss Hermite filters (AGHF), Adaptive Cubature Kalman filters (ACKF), Adaptive Cubature Quadrature Kalman filters (ACQKF).
- Extending the algorithms for Adaptive Divided Difference filter to suit signal models with non additive noise. Extension of the general framework for adaptive nonlinear estimators with non-additive noise and its demonstration by formulating Adaptive Cubature Kalman filter.
- Formulation of alternative general framework for adaptive nonlinear estimators (in presence of additive noise) with information filter configuration which are potentially suitable for multiple sensor fusion. Adaptive version of Divided Difference information filter, Gauss Hermite information filter, Cubature information filter, Cubature Quadrature information filter have been formulated from the general framework and validated using multi sensor estimation problems.
- Adopting the square root framework for formulation of adaptive versions of Gauss Hermite filter, Cubature Kalman filter and Cubature Quadrature Kalman filter both in the standard error covariance form and in the information filter form.

ACKNOWLEDGEMENTS

I express my deepest gratitude to my supervisors, Professor (Dr.) Smita Sadhu and Honorary Emeritus Professor (Dr.) Tapan Kumar Ghoshal, Department of Electrical Engineering, Jadavpur University. Without their scrupulous guidance, intellectual as well as moral support and continuing encouragement it would not be possible for me to come to the end of this journey. I am indebted to them for their valuable advices, comments, and suggestions during this research work and review of the manuscript at different stages which have substantially improved the quality of this dissertation.

I am grateful to my co-worker, Ms. Manasi Das, for research co-operation, comparing simulation results and several insightful discussions.

I thankfully acknowledge the Council of Scientific & Industrial Research (CSIR), Ministry of Human Resource Development, for the financial support during the tenure of this work.

I would like to thank the successive Heads of the Department of Electrical Engineering and the Coordinators of Center for Knowledge Based System for providing research infrastructure. I have great pleasure in expressing my thanks to all my colleagues at National Control Law Team, Bangalore, for their encouragement. I am also thankful to my co-worker, Mr. Nilanjan Patra, for his motivating words to keep my enthusiasm high.

Last, but not least, I would like to express my heart-felt gratitude to my parents for their love, affection, support and encouragement.

Aritro Dey,

Jadavpur University,

July, 2016

Offered at the Lotus Feet of
Sri Aurobindo & The Mother

CONTENTS

ABSTRACT .... **i**

ACKNOWLEDGEMENTS .... **iii**

CONTENTS .... **v**

LIST OF FIGURES .... **ix**

LIST OF TABLES .... **xiii**

CHAPTER 1: INTRODUCTION **1**

1.1 BACKGROUND AND MOTIVATION 1
1.2 BACKGROUND OF NONLINEAR STATE ESTIMATORS 4
1.3 AIMS, OBJECTIVES AND SCOPE OF THE RESEARCH 6
1.4 RESEARCH OBJECTIVES 7
1.5 APPROACH AND METHODOLOGY 8
1.6 SALIENT CONTRIBUTIONS 9
1.7 PUBLICATIONS GENERATED FROM THIS WORK 11
1.7.1 Peer Reviewed International Journals 11
1.7.2 Book Chapter 11
1.7.3 Proceedings of Conferences 11
1.8 CREDIT TO CO-WORKERS 12
1.9 ORGANIZATION OF THE DISSERTATION 13

CHAPTER 2: LITERATURE SURVEY **14**

2.1 CHAPTER INTRODUCTION 14
2.2 LITERATURE ON NONLINEAR (NON-ADAPTIVE) FILTERING 15
2.2.1 Unscented Kalman Filter 16
2.2.2 Interpolation based filters 17
2.2.3 Gauss Hermite Filter 19
2.2.4 Cubature Kalman Filter 21
2.2.5 Cubature Quadrature Kalman Filter 22
2.2.6 Nonlinear Information filters 23
2.2.6.1. Unscented Information Filter 23
2.2.6.2. Central Difference Information Filter 24
2.2.6.3. Cubature Information Filter 24
2.2.6.4. Higher degree Cubature & Gauss Hermite Information Filter 25
2.2.7 Nonlinear filters for non-additive noise 25
2.2.8 Optimality of nonlinear estimators 25
2.3 LITERATURE ON ADAPTIVE FILTERS 26
2.3.1 Adaptive filter with linear signal models 26
2.3.1.1. Bayesian Estimation of unknown noise covariance 27
2.3.1.2. Adaptation of noise covariance using Correlation Method 28
2.3.1.3. Covariance Matching method of adaptation 29
2.3.1.4. Maximum Likelihood Estimation of noise covariance 32
2.3.1.5. Expectation Maximization method of adaptation 33
2.3.1.6. Variational Bayesian approach of adaptation 34
2.3.1.7. Findings from the review on adaptive Kalman filters 35
2.3.2 Adaptive filter with nonlinear signal models 36
2.3.2.1. Adaptive Extended Kalman Filter 37
2.3.2.2. Adaptive Unscented Kalman Filter 39
2.3.2.3. Adaptive Divided Difference Filter 43
2.3.2.4. Adaptive Cubature Kalman Filter 44
2.3.3 Application of Adaptive filters 46
2.4 CONCLUSION 50

CHAPTER 3: TEST PROBLEMS **52**

3.1 CHAPTER INTRODUCTION 52
3.2 DESCRIPTION OF TEST PROBLEMS 52
3.2.1 State estimation of a first order nonlinear system *52*
3.2.2 Bearing only tracking problem *55*
3.2.3 Parameter and state estimation of Van der Pol's oscillator *58*
3.2.4 State estimation of a Lorentz attractor *61*
3.2.5 Object Tracking Problem *64*
3.2.5.1. Dynamic Model I 65
3.2.5.2. Dynamic Model II 68
3.2.6 State estimation of a fourth order nonlinear system *70*
3.2.7 Aircraft Tracking Problem *71*
3.2.7.1. Kinematic Model 71
3.2.7.2. Measurement Model 72
3.2.7.3. Simulation procedure 72

CHAPTER 4: A GENERAL FRAMEWORK FOR ADAPTIVE NONLINEAR FILTERS 75

4.1 CHAPTER INTRODUCTION 75
4.1.1 Problem Statement *76*
4.1.2 Different Approaches for Solution *77*
4.2 THE SOLUTION FRAMEWORK 78
4.2.1 Overview *78*
4.2.2 Part A: Underlying Framework of Non-adaptive Nonlinear Filter *79*
4.2.2.1. Implementation of Bayesian Filters 87
4.2.3 Part B: Derivation of Adaptation Algorithm *88*
4.2.3.1. Adaptation of the Process Noise Covariance (Q) 88
4.2.3.2. Adaptation of the Measurement Noise Covariance (R) 99
4.2.3.3. Analysis of Unbiasedness of Adapted Noise Covariance 108
4.2.3.4. Choice of Window Size 109
4.2.3.5. Notes on Adaptation Methods 110
4.3 ALGORITHMS FOR ADAPTIVE NONLINEAR FILTERS 111
4.3.1 Introduction to Algorithms *111*
4.3.2 Conventional Error Covariance form *112*
4.3.3 Square Root version *115*
4.3.3.1. Notes on Square Root version of Adaptive Nonlinear filters 118
4.4 DEMONSTRATION WITH ADAPTIVE UKF 120
4.4.1 Choice of Sigma Points *120*
4.4.2 Case Study: Object Tracking Problem *122*
4.5 DISCUSSIONS AND CONCLUSIONS 124

CHAPTER 5: ADAPTIVE DIVIDED DIFFERENCE FILTER 125

5.1 CHAPTER INTRODUCTION 125
5.2 STIRLING'S INTERPOLATION FORMULA 126
5.3 APPROXIMATION OF MEAN AND COVARIANCE OF A RANDOM VARIABLE 127
5.3.1 First order approximation *128*
5.3.2 Second order approximation *129*
5.4 ALGORITHM FOR ADAPTIVE DIVIDED DIFFERENCE FILTER 130
5.4.1 Underlying framework of non-adaptive DDF *130*
5.4.2 Algortihm for Q adaptation *132*
5.4.3 Algortihm for R adaptation *133*
5.5 NOTES ON ALGORITHM 134
5.6 CHARACTERIZATION OF ADAPTIVE DDF 135
5.6.1 Validation of Q adaptive DDF *136*
5.6.1.1. Estimation of the states of Van der Pol's oscillator 136
5.6.1.2. Object tracking problem 139
5.6.2 Validation of R adaptive DDF *142*
5.6.2.1. First order nonlinear problem 142
5.6.2.2. Object tracking problem 144

5.7 Discussions and Conclusions 149

CHAPTER 6: ADAPTIVE GAUSS HERMITE FILTER 151

6.1 Chapter Introduction 151
6.2 Gauss Hermite Quadrature Rule 152
6.2.1 Background 152
6.2.2 Generation of Quadrature Points and Weights 153
6.2.3 Extension for higher order systems 154
6.3 Algorithm for Adaptive Gauss Hermite Filter 155
6.4 Characterization of Adaptive GHF 155
6.4.1 Characterization of Q adaptive GHF 156
6.4.1.1. State and parameter estimation of Van der Pol's oscillator 156
6.4.2 Characterization of R adaptive GHF 158
6.4.2.1. State estimation of first order nonlinear system 158
6.4.2.2. Ballistic Object tracking Problem 165
6.5 Characterization of *R* Adaptive SR-GHF 167
6.6 Discussions and Conclusions 171

CHAPTER 7: ADAPTIVE CUBATURE KALMAN FILTER 173

7.1 Chapter Introduction 173
7.2 Spherical Radial Cubature Rule 174
7.2.1 Background 174
7.2.2 Spherical Rule 175
7.2.3 Radial Rule 176
7.2.4 Spherical Radial Cubature Rule 177
7.3 Cubature Quadrature Rule 179
7.4 Algorithm of ACKF and ACQKF 181
7.5 Characterization of proposed estimators 181
7.5.1 Demonstration of Q adaptive version of square root CKF 181
7.5.1.1. Ballistic object tracking problem 181
7.5.2 Demonstration of R adaptive estimators 184
7.5.2.1. First order nonlinear problem 184
7.5.2.2. State estimation of Lorenz attractors 187
7.5.2.3. Fourth order nonlinear estimation problem 190
7.5.3 Demonstration of R adaptive estimators in square root framework 192
7.6 Discussions and Conclusions 194

CHAPTER 8: ADAPTIVE NONLINEAR FILTERS FOR NON-ADDITIVE NOISE 197

8.1 Chapter Introduction 197
8.2 Problem Formulation 198
8.3 General framework for adaptive filter with non-additive noise 198
8.3.1 Underlying Framework of Non-adaptive filter 199
8.3.2 Adaptation algorithms 201
8.3.2.1. Background for statistical linearization 201
8.3.2.2. *R* adaptation algorithm 202
8.3.2.3. *Q* adaptation algorithm 204
8.3.3 Derivation of adaptation algorithm 205
8.3.3.1. *Q* adaptation algorithm 205
8.3.3.2. *R* adaptation algorithm 206
8.4 Formulation of ADDF with Non-additive Noise 208
8.4.1 Non-adaptive DDF framework 208
8.4.2 Adaptation algorithm 211
8.4.2.1. *R* adaptation algorithm 211
8.4.2.2. *Q* adaptation algorithm 212
8.5 Characterization of proposed estimators 213
8.5.1 Characterization of R-Adaptive estimators for non-additive noise 214
8.5.1.1. Object Tracking Problem 214

8.5.1.2. Bearing Only Tracking (BOT) Problem 217
8.5.2 Characterization of Q adaptive DDF for non-additive noise *221*
8.5.2.1. State estimation of Van der Pol's oscillator 221
8.6 DISCUSSIONS AND CONCLUSIONS 226

CHAPTER 9: ADAPTIVE NONLINEAR INFORMATION FILTERS FOR MULTIPLE SENSOR FUSION 228

9.1 CHAPTER INTRODUCTION 228
9.2 PROBLEM STATEMENT 229
9.3 FORMULATION OF ADAPTIVE NONLINEAR INFORMATION FILTER 230
9.3.1 Overview *230*
9.3.2 Non-adaptive Nonlinear Information filter *231*
9.3.3 Adaptation algorithms *233*
9.3.3.1. Adaptation of Process Noise Covariance 234
9.3.3.2. Adaptation of Measurement Noise Covariance 234
9.4 GENERAL FRAMEWORK 234
9.4.1 Algorithmic steps *235*
9.4.2 Multiple Sensor Fusion *238*
9.4.3 Choice of Sigma Points and Weights *238*
9.4.3.1. Unscented Transformation Rule 239
9.4.3.2. Gauss Hermite Quadrature Rule 240
9.4.3.3. Cubature Rule 240
9.4.3.4. Cubature Quadrature Rule 242
9.4.3.5. Divided Difference Interpolation formula 244
9.4.4 Notes *245*
9.5 SQUARE ROOT VERSION 245
9.5.1 Algorithm *246*
9.5.2 Multiple Sensor Fusion *249*
9.5.3 Notes *249*
9.6 CHARACTERIZATION OF PROPOSED ALGORITHMS 250
9.6.1 Ballistic Object Tracking Problem *250*
9.6.1.1. Demonstration of Q-adaptive UIF and DDIF 250
9.6.1.2. Demonstration of R-adaptive DDIF 253
9.6.2 Aircraft Tracking Problem *258*
9.6.2.1. Demonstration of Q adaptive information filters 258
9.6.2.2. Demonstration of R adaptive information filters 263
9.6.2.3. Demonstration of square root versions of R adaptive filters 265
9.7 DISCUSSIONS AND CONCLUSIONS 267

CHAPTER 10: CONCLUSIONS 269

BIBLIOGRAPHY 275

LIST OF FIGURES

Figure 3.1	Plot of measurement with respect to state	54
Figure 3.2	State trajectories for two different representative runs	54
Figure 3.3	True and estimated state for two different representative runs	55
Figure 3.4	Illustration of BOT problem with a schematic diagram	57
Figure 3.5	Plot of position of the target for a representative run	58
Figure 3.6	Plot of velocity of the target for a representative run	58
Figure 3.7	Plot of state x_1 for a representative run	60
Figure 3.8	Plot of state x_2 for a representative run	61
Figure 3.9	Phase plane plot for a representative run	61
Figure 3.10	Plot of state x_1 for a representative run	63
Figure 3.11	Plot of state x_2 for a representative run	63
Figure 3.12	Plot of state x_3 for a representative run	64
Figure 3.13	Phase plane plot for a representative run	64
Figure 3.14	Radar tracking of a ballistic object during reentry: A schematic diagram	65
Figure 3.15	Plot of altitude of the object for a representative run	67
Figure 3.16	Plot of velocity of the object for a representative run	68
Figure 3.17	Plot of trajectories of aircraft for different runs	73
Figure 3.18	A representative run where track loss has occurred	74
Figure 4.1	RMSE of altitude for 1000 Monte Carlo runs	122
Figure 4.2	RMSE of velocity for 1000 Monte Carlo runs	123
Figure 4.3	RMSE of ballistic parameter for 1000 Monte Carlo runs	123
Figure 5.1	RMSE of state (x_1) for 1000 Monte Carlo run	137
Figure 5.2	RMSE of state (x_2) for 1000 Monte Carlo run	137
Figure 5.3	RMSE of friction coefficient (parameter) for 1000 Monte Carlo run	137
Figure 5.4	True and adapted $\boldsymbol{Q}(1,1)$ for a representative run	138
Figure 5.5	True and adapted $\boldsymbol{Q}(2,2)$ for a representative run	138
Figure 5.6	True and adapted $\boldsymbol{Q}(3,3)$ for a representative run	139
Figure 5.7	Phase plane plot for a representative run	139
Figure 5.8	RMSE of altitude for 1000 Monte Carlo run	141
Figure 5.9	RMSE of velocity for 1000 Monte Carlo run	141
Figure 5.10	RMSE of ballistic parameter for 1000 Monte Carlo run	142
Figure 5.11	True and adapted $\boldsymbol{Q}(3,3)$ for a representative run	142
Figure 5.12	RMSE of state for 10000 Monte Carlo run	144
Figure 5.13	RMSE of altitude for 1000 Monte Carlo run	146
Figure 5.14	RMSE of velocity for 1000 Monte Carlo run	147
Figure 5.15	RMSE of ballistic parameter for 1000 Monte Carlo run	147
Figure 5.16	Plot of true and adapted $\boldsymbol{R}$ for a time varying noise covariance	148

Figure 5.17	Plot of true and adapted ***R*** for a time varying noise covariance	148
Figure 6.1	RMS error (friction coefficient estimation) of AGHF & GHF for 1000 MC runs	157
Figure 6.2	RMS error (state, x_2 estimation) of AGHF & GHF for 1000 MC runs	157
Figure 6.3	RMS error (state, x_1 estimation) of AGHF & GHF for 1000 MC runs	158
Figure 6.4	Friction coefficient estimation of AGHF & GHF for a representative run	158
Figure 6.5	Comparison of RMS error of AGHF (with and without re-computation step), Adaptive UKF and Non-adaptive GHF for 10000 MC run when $\boldsymbol{R}_{filter}(0)=10^{-3}*\boldsymbol{R}_{true}$.	162
Figure 6.6	Comparison of RMS error of AGHF (with and without re-computation step), Adaptive UKF and Non-adaptive GHF for 10000 MC run when $\boldsymbol{R}_{filter}(0)=10^{3}*\boldsymbol{R}_{true}$	162
Figure 6.7	True and estimated states for a representative run	163
Figure 6.8	True and adapted measurement noise covariance for a representative run	163
Figure 6.9	***R*** Tracking performance for time varying measurement noise covariance	164
Figure 6.10	Comparison of RMS error of AGHF (with and without re-computation step), Adaptive UKF and Non-adaptive GHF for 10000 MC run when $\boldsymbol{R}_{true}$.is time varying as shown in Fig. 6.9	164
Figure 6.11	Comparison of RMS error (altitude estimation) of AGHF, AUKF & GHF for 1000 MC runs	166
Figure 6.12	Comparison of RMS error (velocity estimation) of AGHF, AUKF & GHF for 1000 MC runs	166
Figure 6.13	Comparison of RMS error (ballistic parameter estimation) of AGHF, AUKF & GHF for 1000 MC runs	167
Figure 6.14	***R*** Tracking performance for time varying measurement noise covariance	167
Figure 6.15	Comparison of RMS error (position estimation) of AGHF & GHF for 10000 MC runs	169
Figure 6.16	Comparison of RMS error (velocity estimation) of AGHF & GHF for 10000 MC runs	169
Figure 6.17	Comparison of RMS error (turn rate estimation) of AGHF & GHF for 10000 MC runs	170
Figure 6.18	Plot of true and adapted value of ***R*** (1,1)	170
Figure 6.19	Plot of true and adapted value of ***R*** (2,2)	171
Figure 7.1	Comparison of RMS error (altitude estimation) of ASRCKF & SRCKF for 1000 MC runs	182
Figure 7.2	Comparison of RMS error (velocity estimation) of ASRCKF & SRCKF for 1000 MC runs	183
Figure 7.3	Comparison of percentage of RMS error (ballistic parameter estimation) of ASRCKF & SRCKF for 1000 MC runs	183
Figure 7.4	Plot of estimated process noise covariance ($\boldsymbol{Q}_{3,3}$) for a representative run	184
Figure 7.5	Comparison of RMS error of proposed filters with the existing filters for 10000 MC run	186
Figure 7.6	Plots of RMSE of first state of Lorentz attractor for different adaptive estimators	188
Figure 7.7	Plots of RMSE of second state of Lorentz attractor for different adaptive estimators	189
Figure 7.8	Plots of RMSE of third state of Lorentz attractor for different adaptive estimators	189
Figure 7.9	Plots of RMSE of 1^{st} state for different adaptive estimators	190
Figure 7.10	Plots of RMSE of 2^{nd} state for different adaptive estimators	191
Figure 7.11	Plots of RMSE of 3^{rd} state for different adaptive estimators	191
Figure 7.12	Plots of RMSE of 4^{th} state for different adaptive estimators	191
Figure 7.13	Comparison of RMS error (position estimation) of ASRCQKF, ASRGHF3 & SRCKF	193
Figure 7.14	Comparison of RMS error (velocity estimation) of ASRCQKF, ASRGHF3 & SRCKF	193
Figure 7.15	Comparison of RMS error (turn rate estimation) of ASRCQKF, ASRGHF3 & SRCKF	194

Figure 8.1	Comparison of RMS error (altitude estimation) of ADDF & DDF for 500 MC runs	215
Figure 8.2	Comparison of RMS error (velocity estimation) of ADDF & DDF for 500 MC runs	216
Figure 8.3	Comparison of RMS error (parameter estimation) of ADDF & DDF for 500 MC runs	216
Figure 8.4	Plot of adapted $\boldsymbol{R}$ when truth value is constant	217
Figure 8.5	Plot of adapted $\boldsymbol{R}$ when truth value is time varying	217
Figure 8.6	RMSE of position of ACKF, ADDF and non-adaptive CKF	218
Figure 8.7	RMSE of velocity of ACKF, ADDF and non-adaptive CKF	218
Figure 8.8	Plot of adapted $\boldsymbol{R}$ of ACKF (3rd degree) for element $\boldsymbol{R}(1,1)$	219
Figure 8.9	Plot of adapted $\boldsymbol{R}$ of ACKF (3rd degree) for element $\boldsymbol{R}(2,2)$	219
Figure 8.10	Plot of adapted $\boldsymbol{R}$ of ACKF (3rd degree) for element $\boldsymbol{R}(3,3)$	220
Figure 8.11	Comparison of RMS error (state, x_1 estimation) of ADDF & DDF for 1000 MC runs	222
Figure 8.12	Comparison of RMS error (state, x_2 estimation) of ADDF & DDF for 1000 MC runs	223
Figure 8.13	Comparison of RMS error (friction coefficient estimation) of ADDF & DDF for 1000 MC runs	223
Figure 8.14	Plot of true and adapted $\boldsymbol{Q}(1,1)$ for a typical run	223
Figure 8.15	Plot of true and adapted $\boldsymbol{Q}(2,2)$ for a typical run	224
Figure 8.16	Plot of true and adapted $\boldsymbol{Q}(3,3)$ for a typical run	224
Figure 8.17	Comparison of phase portrait of the DDF and ADDF estimates with that of true states	225
Figure 9.1	Comparison of RMSE (altitude) of AUIF, ADDIF & DDIF for 1000 MC runs	251
Figure 9.2	Comparison of RMSE (velocity) of AUIF, ADDIF & DDIF for 1000 MC runs	252
Figure 9.3	Comparison of RMSE (ballistic parameter) of AUIF, ADDIF & DDIF for 1000 MC runs	252
Figure 9.4	Plot of estimated process noise covariance ($\boldsymbol{Q}_{3,3}$) for a representative run	252
Figure 9.5	Comparison of RMS error (altitude estimation) of ADDF & DDF for 1000 MC runs	254
Figure 9.6	Comparison of RMS error (velocity estimation) of ADDF & DDF for 1000 MC runs	254
Figure 9.7	Comparison of RMS error (ballistic parameter estimation) of ADDF & DDF for 1000 MC runs	254
Figure 9.8	Plot of true and adapted $\boldsymbol{R}$ for a representative run	255
Figure 9.9	Comparison of RMS error (altitude estimation) of ADDF & DDF for 1000 MC runs	256
Figure 9.10	Comparison of RMS error (velocity estimation) of ADDF & DDF for 1000 MC runs	257
Figure 9.11	Comparison of RMS error (ballistic parameter estimation) of ADDF & DDF for 1000 MC runs	257
Figure 9.12	Plot of adapted $\boldsymbol{R}$ for the faulty measurement (for a representative run)	258
Figure 9.13	Comparison of RMSE (position estimation) of AGHIF & GHIF for 10000 MC runs	261
Figure 9.14	Comparison of RMSE (velocity estimation) of ACIF & CIF for 10000 MC runs	261
Figure 9.15	Comparison of RMSE (turn rate estimation) of ACQIF & CQIF for 10000 MC runs	261
Figure 9.16	Plot of adapted process noise co-variance (Q5,5) for a representative run	262
Figure 9.17	Comparison of RMS error (position estimation) of ACIF, AGHIF, ADDIF, ACQIF for 10000 MC runs	262
Figure 9.18	Comparison of RMS error (position estimation) of ACIF, AGHIF, ADDIF, AUIF, AHCQIF for 10000 MC runs	264
Figure 9.19	Comparison of RMS error (velocity estimation) of ACIF, AGHIF, ADDIF, AUIF, AHCQIF for 10000 MC runs	264
Figure 9.20	Comparison of RMS error (turn rate estimation) of ACIF, AGHIF, ADDIF, AUIF, AHCQIF for 10000 MC runs	265

Figure 9.21	Comparison of RMS error (position estimation) of ASRCQIF, ASRGHIF, ASRCIF for 10000 MC runs	266
Figure 9.22	Comparison of RMS error (velocity estimation) of ASRCQIF, ASRGHIF, ASRCIF for 10000 MC runs	266
Figure 9.23	Comparison of RMS error (turn rate estimation) of ASRCQIF, ASRGHIF, ASRCIF for 10000 MC runs	267

LIST OF TABLES

Table 3.1	Parameters for state estimation of Van der Pol's oscillator	60
Table 3.2	Parameters for state estimation of Lorenz attractor	62
Table 3.3	Parameters for object tracking problem (SI units)	67
Table 3.4	Parameters for object tracking problem (FPS units)	70
Table 6.1	Percentage of track loss cases computed from 10000 Monte Carlo runs	165
Table 7.1	Percentage of track loss and computation time presented for adaptive estimators	187
Table 7.2	Percentage of track loss and computation time for adaptive estimators	194
Table 9.1	Selection of Sigma Points & weights	239
Table 9.2	Selection of Quadrature Points & weights	240
Table 9.3.1	Selection of Cubature Points & weights	240
Table 9.3.2	Selection of Cubature Points & weights	241
Table 9.4.1	Selection of Cubature Quadrature Points & weights	242
Table 9.4.2	Selection of Cubature Quadrature Points & weights	243
Table 9.5	Selection of Sigma Points & weights	244
Table 9.6	Comparative study of %-age of track loss form $\boldsymbol{Q}$ adaptive information filters	263
Table 9.7	Comparative study of % -age of track loss for $\boldsymbol{R}$ adaptive information filters	265

Chapter 1: Introduction

1.1 Background and Motivation

Parameter and state estimation techniques find many uses in system modeling as well as in conventional, adaptive and optimal control applications. Parameter estimation is important for determining system parameters for yet to be modeled systems and also for systems where system parameters are not constants but vary with time or with environmental conditions. Apart from the uses in closed loop control as stated above, state and parameter estimation may also play a significant role in fault detection and identification. For example, state estimation may potentially reduce the cost of additional sensors to create analytical redundancy [Hwang2010], which in turn, may be useful in fault detection or continued operation in sensor failure conditions.

In our current state of knowledge, state estimation for linear signal models in presence of noisy measurements is comparatively easy because of the availability of precise estimation method, viz. Kalman filter [Anderson1979] which provides algebraic steps to arrive at a provably optimal estimation solution with minimal restrictions . However, state estimation for nonlinear system models becomes challenging primarily because analytical means to assess the optimality of performance and convergence of any (existing or conceivable) nonlinear filtering algorithms is not presently available. Consequently, a provable optimal closed form or iterative state estimation solution applicable to a wide class of nonlinear systems are not available and recourse is taken to approximate solutions. One of the earliest approximate state estimation technique goes by the name Extended Kalman Filter (EKF) [Brown1983].

As no nonlinear estimator has yet been reported which can guarantee optimality and/or convergence of an iterative estimate the scope of relative improvement of estimation accuracy compared to the already reported estimators or scope for developing an alternative algorithm for nonlinear signal models still remains open.

Knowledge in the domain of nonlinear estimation is still evolving (as would be evident from a literature survey, presented in a subsequent chapter) and new algorithms continue to be contributed in the field of nonlinear state estimation. Many such state estimation methods are collectively known as Post Kalman filtering [Ristic2004] which includes Unscented Kalman filter [Julier2000, Ristic2004], Divided and Central Difference filters [Ito2000, Norgaard2000, Schei1997], Gauss Hermite filters [Ito2000] and simulation based filters like Particle filters [Ristic2004] which are performance wise superior than EKF. Subsequently, newer filtering algorithms, e.g., Cubature Kalman filter [Arasaratnam2009], higher degree Cubature Kalman filter [Jia2013], Cubature Quadrature Kalman filter [Bhaumik2013] and its other version with higher degree quadrature points [Singh2015] and Fourier Hermite [Sarmavuori2012] filters have been contributed.

It would not be out of place to mention that even for joint estimation of parameters and states of linear systems nonlinear filters are often employed [Simon2006] where the system dynamics is expressed in terms of parameter augmented state vector.

Usually the nonlinear estimators are empirically validated before actual deployment in real life systems. The choice of suitable candidate from the pool of several competing algorithms for a specific mission requires experimentation as there is no direct analytical means which can indicate the best suited filter for the specific problem. A comparative study with respect to the accuracy and computational intensiveness often needs to be carried out. Scrutinising those results the designer can choose the most suitable filter for that particular class of nonlinear estimation problems. Nonlinear filtering, therefore, necessitates substantial effort in experimentation.

Linear as well as nonlinear estimators require prior knowledge of the system dynamics, the measurement equations and the values of noise covariances for producing satisfactory estimation results. Non-availability of knowledge of any of them deteriorates the performance of the filter. Therefore, the choice of the noise covariances which may often remain unknown in many situations is made after trial and error. This requires substantial effort in offline tuning of the estimators. Improper choice of noise covariances yields suboptimal estimation result and may cause even divergence for nonlinear estimation problem. In such situations, “adaptive estimators” may be employed. Such adaptive estimators implicitly or explicitly, estimate noise covariances (along with estimation of states and parameters).

Adaptive filters for linear signal models have been reported a few decades ago and validated by a large number of researchers. Substantial publications exist wherein these techniques are successfully employed in many real life applications.

However, Adaptive filtering for nonlinear systems is still an evolving area of knowledge which has drawn the attention of the researchers. It has been observed from the literature survey that there remains “knowledge gap” in the area of nonlinear filtering where one or more noise covariances are unknown. Unlike adaptive filters with linear signal model sufficient works on adaptive nonlinear estimator are yet to be available which have strong theoretical foundations. Moreover depending on the degree of nonlinearity in the signal model, the performance of adaptive nonlinear estimators may vary and choice of a suitable underlying structure for adaptive nonlinear estimator needs experimentation in the context of estimation accuracy and computation effort. The above discussed points motivated the present worker to investigate in the field of adaptive nonlinear estimators.

For parameter estimation several well known classical techniques e.g., least square, orthogonal least-squares, gradient-weighted least-squares methods exist. Apart from these classical methods, deterministic observers and, Post Kalman state estimators have been employed for parameter estimation as reported in previous works. For simultaneous estimation of parameters and states, any unknown parameter is modeled as an additional state and therefore augments the state vector. As mentioned before, the augmented state renders estimation problem as nonlinear even with linear signal models. Therefore, joint estimation of parameters and states essentially requires nonlinear estimators.

For the systems perturbed by unknown parameter variation the nature of the parameter variation also remains unknown as the parameters are unknown per se. During the parameter estimation of such systems assuming the simplest case the unknown parameters are modeled as random walk model while in reality the parameters may vary following certain trend. The uncertainty in the nature of parameter variation makes it difficult to assign a suitable process noise for the augmented system. Adaptive nonlinear filters, therefore, may be suitable estimators for parameter estimation as well apart from its use in nonlinear state estimation.

1.2 Background of Nonlinear State Estimators

The basics of nonlinear state estimation in presence of noisy measurements (with known process and measurement noise covariances) have been well covered in the oft referred textbooks e.g. [Anderson1979, Brown1983, Zarchan2000, Ristic2004]. A few significant points on nonlinear state estimation are reiterated in this section.

The estimation algorithms which are intended for nonlinear systems are usually termed as nonlinear estimators. State estimators require knowledge of the system dynamics which may be expressed in terms of state space models and the observation equation (alternatively termed as measurement model). Nonlinearity of any or both of the above models, makes the system model nonlinear by definition. The estimation algorithms can be presented in continuous time as well as discrete time. Throughout this dissertation the present worker has followed the approach of discrete time representation.

Usually, the system dynamics and measurement equation for a higher dimensional system in discrete time domain may be represented as given by (1.1) and (1.2) respectively.

$$\boldsymbol{x}_k = \boldsymbol{f}\left(\boldsymbol{x}_{k-1}, \boldsymbol{u}_k, \boldsymbol{w}_k\right) \tag{1.1}$$

$$\boldsymbol{y}_k = \boldsymbol{g}\left(\boldsymbol{x}_k, \boldsymbol{v}_k\right) \tag{1.2}$$

Here $\boldsymbol{x}_k$ is the state vector of the system, $\boldsymbol{u}_k$ is the vector of known input, $\boldsymbol{w}_k$ is the vector of random unknown input known as process noise, $\boldsymbol{y}_k$ is the measurement vector and $\boldsymbol{v}_k$ is the vector for measurement noise. In the above equations noises are expressed in non-additive form. However, for the simplest case the noises can be expressed as additive noise as below.

$$\boldsymbol{x}_k = \boldsymbol{f}\left(\boldsymbol{x}_{k-1}, \boldsymbol{u}_k\right) + \boldsymbol{w}_k \tag{1.3}$$

$$\boldsymbol{y}_k = \boldsymbol{h}\left(\boldsymbol{x}_k\right) + \boldsymbol{v}_k \tag{1.4}$$

In this dissertation both additive and non-additive cases have been considered.

State estimators also require the knowledge mean and covariance (first two moments) of the random noise which characterize such noise. In simplest cases the noises are assumed to be zero mean, white and Gaussian. The noises can also be non Gaussian, coloured and there may be correlation between process noise and measurement noise. Special treatments are necessary for such cases. However, such conditions have not been considered in this dissertation.

Most of the state estimation algorithms may be expressed in the Bayesian Estimation framework. The probability density function of the estimated state vector may be expressed in terms of Bayesian integrals. For zero mean Gaussian noise the mean and covariance of estimates can be computed by approximating the Bayesian integrals.

Generally the measurement noise is assumed to be zero mean white Gaussian noise sequence. The covariance matrix of the measurement noise is generally symbolized by '***R***' The measurement noise for well characterized sensing systems may be obtained from the experimental data or vendor specification. However in many situations the measurement noise covariance may not be known at the time of designing the filter.

The process noise in the above state equation signifies several things e.g., the unknown and random component of input which may be considered as disturbance, parametric uncertainty, modelling error etc. Like measurement noise, the process noise also can be considered to be zero mean, white and Gaussian for simplest case. The covariance matrix of the process noise is generally symbolized by '***Q***'.

The objective of the nonlinear state estimation is to obtain the optimal estimate of the states with known nominal models and known measures of process and measurement noise. However, even with the above knowledge, no nonlinear estimator has been reported as of now that can guarantee optimal estimation performance, as mentioned earlier.

After initialization the filters provide the state estimates recursively for the subsequent time using noise corrupted measurements as input. To begin the process of estimation, the state variables for the filter need to be initialised along with an initial choice of error covariance of those states. The error covariance indicates the degree of uncertainty in the initial choice of state.

In real life application of nonlinear state estimator situations may arise where the sensor characterisation is partially done or not done at all. The system dynamics may also suffer from unknown parameter variation. It becomes difficult then to assign the accurate values of ***Q*** and ***R***. In such a situation, adaptive estimation (where one or both noise covariances are also estimated along with the states and parameters) may be employed. When only the covariance $\boldsymbol{R}$ is not available the corresponding adaptive filter is often termed as $\boldsymbol{R}$-adaptive. Similarly, when only the process noise covariance $\boldsymbol{Q}$ is not available the corresponding adaptive filter is often termed as $\boldsymbol{Q}$-adaptive. The adaptive filter is often termed as $\boldsymbol{QR}$ – adaptive when both these covariances are unknown.

It may be recalled from the above discussions that adaptive estimation is required where one or both the noise covariances are unknown. Before adaptive estimation was properly understood and accepted (mostly in the context of linear signal models) system designers employed "manual tuning" which required exhaustive offline studies. Adaptive estimation obviates the need of such manual tuning.

An alternative to manual tuning of $\boldsymbol{Q}$ and or $\boldsymbol{R}$ is tuning a "scale factor" [Hide2004, Ding2007, Jwo2008]. In this approach the filter gain or error covariance matrices may be tuned with a scaling factor to get satisfactory estimation result.

Adaptive nonlinear state estimators also appeared in recent literature. The adaptive EKF [Busse2003, Bavdekar2011, Hajiyev2011] has been developed after Adaptive Kalman filter. However, AEKF incorporates all the short comings [Soken2012] of non-adaptive EKF as it happens to be the core of AEKF. Adaptive UKF [Chai2012, Hajiyev2014, Das2015] which has been recently introduced is performance wise superior to AEKF. AUKF too needs some tuning of parameters which control the locations and spread of sigma points.

Most of these non-linear state estimators (also known as nonlinear adaptive filters) use adaptive mechanisms which are analogous of corresponding linear adaptive filters. Such adaptation algorithms are based on intuitive methods without proper mathematical derivation.

1.3 Aims, Objectives and Scope of the Research

The dissertation aims to develop novel techniques for adaptive nonlinear state estimation. The focus is on introducing adaptive mechanisms to (non-adaptive) nonlinear estimators of recent origin. Many such improved (non-adaptive) nonlinear estimators of recent origin have been developed and reported in literature with the assumption that noise covariances are accurately known. It is needless to mention that even these new generation non-adaptive state estimators require manual tuning when the values of noise covariances remain unavailable.

The above inadequacy in the domain of nonlinear estimation gives an intellectual challenge to the present worker to develop improved adaptive nonlinear estimators which may be performance wise superior to most the existing adaptive nonlinear estimators and which may have some theoretical/analytical support or justification of the adaptation algorithms provided in such estimation techniques.

The other aim is demonstration of performance and quantitative characterization of the proposed nonlinear adaptive estimators in realistic situations.

State estimators had previously been employed for sensor data fusion. This prompted the present worker to explore whether the proposed estimators can also be employed foe similar purpose.

To keep the scope of work focused the following decisions were taken:

(1) Case studies for characterization of the proposed estimators should be chosen from domains where previous results are available. The aerospace tracking problem domain fitted this requirement.

(2) Signal models are to be chosen with additive zero mean Gaussian white noise model, primarily for facilitating comparison with previous work.

(3) Siganl models with non-additive Guassian noise is to be explored separately.

1.4 Research Objectives

In the perspective of the problem addressed in this dissertation, the research objective may be summarized as stated below:

(i) Formulation of improved algorithms for adaptive estimators for estimation of states as well as parameters of nonlinear systems.

(ii) Validation and characterization of such improved algorithms using non-trivial signal models.

(iii) Performance comparison of such improved algorithms with previously reported approaches of adaptive and non-adaptive nonlinear estimator.

(iv) Formulation of a computationally efficient alternative version of adaptive nonlinear estimators for multiple sensor fusion.

(v) Exploring the possibility of evolving some theoretical/analytical support or justification of the adaptation algorithms provided in the proposed estimation techniques

1.5 Approach and Methodology

From the research objective mentioned above it may be appreciated that the aim of this dissertation is to provide improved solution methods for adaptive nonlinear estimators.

Towards this goal existing literature about filters for nonlinear systems, adaptive filters for linear systems and previously proposed adaptive filters have been studied extensively. A summary of such studies is provided in the literature review section. Previously published ideas which have been found to be extendible to the proposed research had been closely examined. Similarly weaknesses of previously reported estimators were carefully studies so that such weaknesses are not inherited by the proposed estimators.

The new estimation algorithms which are developed in this dissertation are evaluated and quantitatively characterized using test problems. This approach had to be taken because (as previously mentioned) optimality and convergence of nonlinear estimators cannot be demonstrated in the usual analytical way.

For evaluation of consistency, convergence of the proposed algorithms and comparison of estimation the accuracy between different methods Monte Carlo simulations with adequate runs have been carried out. Each Monte Carlo run is simulated using true state trajectories and measurements which are obtained with random samples of process noise covariance and measurement noise covariance both having the correct value (truth value). However, for filters the covariance which remains unknown is initialized with an assumed value. In each Monte Carlo run this assumed value of covariance remains the same.

During the evaluation of the proposed estimators in simulation there remains the scope to evaluate the performance adaptive filter without the knowledge of one of the covariances with the non-adaptive filter in the ideal situation where noise covarinaces are known to it. This study can give a measure about how far the performance of adaptive filter, even in the adverse situation, approach to the ideal performance.

1.6 Salient Contributions

Salient contributions covered in this dissertation are:

(i) Proposing a fairly general framework for ***Q***-adaptive nonlinear state estimators and demonstrating the applicability of the framework with specific examples.

(ii) Proposing a fairly general framework for ***R***- adaptive nonlinear state estimators and demonstrating the applicability of the framework with specific examples.

(iii)Extending the above general frameworks for the square root version of adaptive nonlinear estimators.

(iv)Derivation of the nonlinear versions of adaptation algorithms for ***R***-adaptive nonlinear estimators following the Maximum Likelihood Estimation (MLE) method respectively utilizing innovation and residual sequences. The latter version is important as it automatically ensures positive definiteness of the adapted ***R***- matrix.

(v) Modification of the existing Maximum a Posteriori (MAP) based algorithms for adaptive nonlinear estimators (both ***R***-adaptive and ***Q***-adaptive) with reasonable simplifying assumptions which illustrate that the adaptation algorithms after modification match well with those obtained by the MLE method and the intuitive Covariance Matching method.

(vi)Proposing and characterizing algorithms for different versions of Adaptive Divided Difference filter (DDF) which include

- Scaling factor based ***Q*** adaptive DDF
- ***Q*** adaptive DDF using direct adaptation algorithm
- Residual based ***R*** adaptive DDF using direct adaptation algorithm with ensured positive definiteness of ***R***
- ***Q*** & ***R*** adaptive DDF for non-additive noises

(vii) Proposing and formulation of algorithms for several new types of adaptive nonlinear estimators, viz. Adaptive Gauss Hermite filters (GHF), Adaptive Cubature Kalman filters, Adaptive Cubature Quadrature Kalman filters.

- Derivation and demonstrations that the said algorithms may also be derived from the proposed general frameworks as mentioned above.

  - The algorithms for the said Adaptive Gauss Hermite filter include
    - Partially ***Q*** adaptive GHF where some of the elements of ***Q*** are known a priori
    - ***R*** adaptive GHF based on innovation sequence
    - An alternative ***R*** adaptive GHF based on residual sequence
    - Square Root versions of ***R*** adaptive GHF
- The algorithms for Adaptive Cubature Kalman filter comprise
  - ***R***-adaptive Cubature Kalman Filter with 3rd degree accuracy
  - ***R***-adaptive Cubature Kalman Filter with 5th degree accuracy
  - ***R***-adaptive Cubature Quadrature Filter with 3rd degree accuracy
  - ***R***-adaptive Cubature Quadrature Filter with 5th degree accuracy
  - Square root versions of adaptive Cubature Kalman Filter, adaptive Cubature Quadrature Kalman Filter with 3rd degree accuracy.
- Characterization of the said adaptive estimators with the help of non trivial case studies and Monte Carlo simulation

(viii) Extending the algorithms for adaptive nonlinear filters to suit signal models with non-additive noise for the following types of estimators:

- Adaptive Divided Difference filter
- Adaptive Cubature Kalman filter with 3rd degree and 5th degree accuracy

(ix) Formulation of alternative general algorithms for adaptive nonlinear estimators with information filter configuration which are potentially suitable for multiple sensor fusion. In particular information filter versions of the following estimators have been formulated and characterized:

- ***Q***-Adaptive Divided Difference information filter (DDIF)
- ***Q***-Adaptive Unscented information filter (UIF)
- ***R***-Adaptive DDIF
- ***R***-Adaptive Gauss Hermite information filter (GHIF)
- ***R***-Adaptive Cubature information filter (CIF)

- ***R***-Adaptive Cubature Quadrature information filter (CQIF)
- Square root version of CQIF, CIF, GHIF for ***R*** adaptation.

(x) Compilation of a comprehensive literature review for the benefit of future workers.

1.7 Publications generated from this work

1.7.1 Peer Reviewed International Journals

1. A. Dey, M. Das, S. Sadhu, T. K. Ghoshal, "Adaptive Divided Difference Filter for Parameter and State Estimation of Nonlinear Systems," *IET Signal Processing,* 2015, vol. 9, issue 4, pp. 369-376
2. A. Dey, S. Sadhu, T. K. Ghoshal, "Adaptive Gauss Hermite Filter for Nonlinear Systems," *IET Science Measurement and Technology,* 2015, vol. 9, issue 8, pp. 1007-1015
3. M. Das, A. Dey, S. Sadhu, T. K. Ghoshal, "Central Difference Filter for Nonlinear State Estimation," *IET Science Measurement and Technology,* 2015, vol. 9, issue 6, pp. 728-735

1.7.2 Book Chapter

1. A. Dey, S. Sadhu, T. K. Ghoshal, "Adaptive Nonlinear Information Filters for Multiple Sensor Fusion," Revised Selected Papers of 12th International Conference on Informatics in Control, Automation and Robotics, *Lecture Notes in Electrical Engineering published by Springer Verlag*, 2016, vol. 383, pp. 371-390, DOI 10.1007/978-3-319-31898-1_21

1.7.3 Proceedings of Conferences

1. A. Dey, S. Sadhu, T. K. Ghoshal, "Joint Estimation of Parameters and States of Nonlinear Systems using Adaptive Divided Difference Filter," in the Proceedings of *2nd Michael Faraday IET India Summit*, Kolkata, pp. CS.7-CS.11, 2013.
2. A. Dey, S. Sadhu, T. K. Ghoshal, "Adaptive Divided Difference Filter for Nonlinear Systems with Unknown Noise," in the Proceedings of *International Conference on Control, Instrumentation, Energy and Communication*, University of Calcutta, Kolkata, pp. 640-644, 2014.

3. A. Dey, S. Sadhu, T. K. Ghoshal, "Adaptive Gauss Hermite Filter for Parameter varying Nonlinear Systems," in the Proceedings of *International Conference on Signal Processing and Communication*, IISc Bangalore, , pp. 1-5, 2014.
4. A. Dey, M. Das, S. Sadhu, T. K. Ghoshal, "Adaptive Gauss Hermite Filter for Parameter and State Estimation of Nonlinear Systems," in the Proceedings of in 12th *International Conference on Informatics in Control, Automation and Robotics*, Vienna, Austria, vol. 1, pp. 583-589, 2014.
5. A. Dey, S. Sadhu, T. K. Ghoshal, "Adaptive Divided Difference Filter for Nonlinear Systems with Non-additive Noise," in the Proceedings of 12th *International Conference on Computer, Communication, Control and Information Technology (C3IT)*, Hooghly, WB, pp. 1-5, 2015.
6. A. Dey, S. Sadhu, T. K. Ghoshal, "Multiple Sensor Fusion using Adaptive Divided Difference Information Filter," in the Proceeding of *International Conference on Informatics in Control, Automation and Robotics*, Colmar, France, vol. 1, pp. 398-406, 2015.
7. A. Dey, M. Das, S. Sadhu, T. K. Ghoshal, "Adaptive Unscented Information Filter For Multiple Sensor Fusion," in the proceeding of *Michael Faraday IET India Summit 2015*, Kolkata, pp. 541-545, 2015.
8. M. Das, A. Dey, S. Sadhu, T. K. Ghoshal, "Adaptive Unscented Kalman Filter at the Presence of Non-additive Measurement Noise" in the proceeding of 12th *International Conference on Informatics in Control, Automation and Robotics*, Colmar, France, vol. 1, pp. 614-620, 2015.
9. M. Das, A. Dey, S. Sadhu, T. K. Ghoshal, "Joint Estimation of States and Parameters of a Reentry Ballistic Target Using Adaptive UKF," in the proceeding of Fifth International Symposium on Electronic System Design (ISED), Surathkal, pp. 99 – 103, 2014.

1.8 Credit to Co-workers

The present worker is indebted to Ms. Manasi Das who first identified the knowledge gap in the domain of adaptive nonlinear estimation and contributed the algorithm for Adaptive UKF first in [Das2013], a conference publication, and subsequently in [Das2015], a journal publication on adaptive UKF. Later, the algorithms for adaptive CDF and adaptive UKF with non-additive measurement noise have been published with joint authorship after several insightful discussions with Ms. Das.

1.9 Organization of the Dissertation

This dissertation comprises of ten chapters including the present one which have been arranged to maintain a continuity of discussion. A comprehensive literature survey on adaptive filters as well as nonlinear filtering follows this introductory chapter. The third chapter presents the test problems which are considered in the subsequent chapters for demonstration and performance analysis of proposed estimation algorithms.

A general algorithm for adaptive nonlinear estimators is presented in chapter 4 which incorporates derivation of several adaptation algorithms for process and measurement noise covariance. Both the square root and standard error covariance approaches for adaptive nonlinear filter are presented here.

In chapter 5 algorithms for Adaptive Divided Difference filter (ADDF) have been developed. In chapter 6 and chapter 7 respectively algorithms for Adaptive Gauss Hermite filter (AGHF) and Adaptive Cubature Kalman filter (ACKF) have been formulated using the general framework presented in chapter 4 and their performance is compared with the other competing algorithms.

In chapter 8 the algorithms for adaptive nonlinear filter and Adaptive Divided Difference filter have been reformulated for the situations with non-additive noise. The performance of ADDF and ACKF with non-additive noise is also demonstrated thereafter.

The general algorithm for adaptive nonlinear filter is extended in information filter configuration in chapter 9 for its possible use in multiple sensor fusion. A set of new algorithms of adaptive information filters are derived from this general framework and their relative advantage has been investigated in this chapter.

At the end of this dissertation the concluding comments are presented in chapter 10 along with the scope of further work.

Chapter 2: Literature Survey

2.1 Chapter Introduction

A brief literature survey related to adaptive state estimation for nonlinear systems, the central topic of the dissertation, has been presented in this chapter. It has been explained in the introduction that adaptive state estimation is called for in the contingent situations where knowledge of noise covariances remains incomplete. Significant contributions by previous workers up to the year 2015 have been reviewed in the above perspective along with notes, where applicable, how such previous work opened the scope of further research. This review is organised thematically rather than in conventional chronological order for the ease of interpretation. As the state estimators are often called "filters" (possibly since the days of Kalman Filters) both these expressions would be considered as synonymous in the context of this survey.

The first logical part of this chapter briefly reviews recent as well as foundational work on conventional (i.e non-adaptive) nonlinear state estimation, particularly, those which are popularly called as "Post Kalman filtering" techniques (i.e., EKF and its successors). A review on non-adaptive nonlinear filtering algorithms is important as these are used as the core of adaptive nonlinear filters. This part of the review is necessarily brief as issues of several nonlinear filters have been discussed rather elaborately in standard text books [Simon2006, Ristic2004, Zarchan2000, Brown1983, Anderson1979]. A summary of recent developments on nonlinear filtering after publications on Unscented Kalman filter has also been provided in this section. As the theme of this dissertation widely differs from the concept of simulation based (non-adaptive) filters like particle filter, the vast literature of simulation based filters have not been reviewed.

The next logical part reviews publications on adaptive filters after due categorization. Both linear and non linear systems have been covered with more emphasis on the latter. Possible real world applications of adaptive estimation algorithms are also reviewed in this part. Note that recently reported adaptive filters based on artificial intelligence has not been reviewed in this part as these approaches do not thematically match with the approach followed to develop adaptive estimators.

2.2 Literature on nonlinear (non-adaptive) filtering

For estimation of the states of a linear system in presence of noisy measurements Kalman filter has been extensively used for its optimal estimation performance even for the time varying signal model [Anderson1979]. The Linearized Kalman filters (LKF) and Extended Kalman filters (EKF) are first introduced in the solution domain of nonlinear estimation and have been widely reported in literature [Brown1983, Simon2006]. The LKF is implemented by linearizing the nonlinear system dynamics and measurement model about a nominal state trajectory of the system. However, EKF linearizes the system dynamics and measurement equation about *a posteriori* estimate of state of previous instant and *a priori* estimate of state of current instant respectively. Therefore, the operating point of linearization changes in each update cycle.

The Extended Kalman filter which can perform satisfactorily for systems with mild nonlinearity suffers from some negative aspects[Merwe2004]: (i) EKF considers only first order term for linearization and, therefore, approximation becomes inaccurate for the systems with significant nonlinearity (though higher order approximations had been introduced later [Simon2006]) , (ii) Computation of Jacobian increases complexity for higher order systems, (iii) Computation of Jacobian becomes impossible for the systems with discontinuity, (iv) Inaccuracy due to linear approximation can lead the filter to divergence.

The limitations of EKF reported in [Brown1983, Julier2004, Simon2006] have prepared the background of research on nonlinear estimation methods. One such class of nonlinear filters called “Sigma Point filters” [Merwe2004] can reportedly overcome the limitations of EKF by propagating the mean and covariance of the estimate by generating a set of “sigma points” and weights in a deterministic approach. These sigma point based filtering algorithms have been found to be performance wise superior to EKF. Despite the earlier widespread use of EKF [Merwe2004] the recent works on nonlinear state estimation are becoming more inclined to the sigma point filters. In the following subsections works on different sub types of sigma point filters have been reviewed and their advantages over one another are also reported. Many of the sigma point filters have also been called as “Linear Regression Kalman Filters” (LRKF) [Lefebvre 2002].

2.2.1 Unscented Kalman Filter

The Unscented Kalman Filter (UKF) is a popular member of the family of sigma point filters which is based on Unscented Transformation rule. This filter was first proposed in [Julier1995] and established as a derivative free sigma point filter in literature by a number of follow up papers [Julier1996, Julier1997, Julier2000, Julier2004, Merwe2004].

The mean and covariance of the estimates are computed as a weighted sum of a set of discrete points which are obtained after propagating sigma points through the nonlinear functions. The selection of sigma points is based on Unscented Transformation rule which is explained in [Julier1997]. This algorithm is consequently free from derivative calculation and presents superior performance compared to EKF as it incorporates second order approximation during mean and covariance estimation. Estimation accuracy of UKF and computational efficiency are also discussed in [Julier1997]. Superiority of estimation performance of UKF with respect to EKF is demonstrated with a case study.

The performance of UKF can be improved with scaled version of UKF discussed in [Julier1999, Julier2002, Merwe2004]. The spread of deterministically chosen sigma points can be controlled by suitable choice of scaling parameters. In an invited paper [Julier2004] contributed by the same authors the motivation, formulation, uses of unscented transformation method (UT) has been presented in a comprehensive way.

Reduced order UKF is proposed in [Julier1998, Julier2002] so that the computational load can be reduced compared to the algorithm presented in their earlier papers. Unlike the standard UKF which requires $2n+1$ (n is the order of the system) of sigma points only $n+1$ points may be sufficient for reduced order UKF.

The square root version of UKF (scaled) was reported in [Merwe2001]. This square root version of UKF is numerically stable and employs three algebraic techniques e.g.; QR-decomposition, Cholesky factor updating, and efficient least square to provide numerically efficient and stable filtering algorithm. The proposed method has been applied for both state and parameter estimation. In this work, it is reported that the SR-UKF requires the same computation effort as in EKF**.**

In the PhD dissertation [Merwe2004] the author summarizes the previous work on nonlinear filtering which uses sigma points and provides a general filtering algorithms based on

Bayesian Estimation framework where the Bayesian integrals are numerically evaluated using sigma points and weights. The varieties of sigma point filters which may be obtained from this general algorithm includes UKF and its corresponding square root versions namely SR-UKF. Simulation results are provided to show that these filters are better than EKF. This work also discusses a hybrid version of UKF and sequential Monte-Carlo filter.

[Wu2006] presented a general algorithm for Gaussian filters where the UKF can be obtained by the rule of exact monomials. There the author proposed to get a higher order UKF using the rule for higher precision. The author proposed to choose 5^{th} degree accuracy and name the algorithm as UKF5. It is demonstrated that it is performance wise superior to conventional UKF. This concept has similarity with the concept of [Julier2004] where the possibility of higher order UKF was first mentioned.

2.2.2 Interpolation based filters

In the family of sigma point filters some of the members are developed based on interpolation formula. These estimators are well known as Central Difference filters [Schei1997, Ito2000] and Divided Difference filter [Norgaard2000]. These algorithms are approximation based methods where the nonlinear function is approximated with the Taylor series with first and second order accuracy. However, computation of Jacobian and Hessian matrices are replaced by function evaluations based on interpolation formula.

The contribution of [Schei1997] is the earliest towards such an approach of filtering. In [Schei1997] the author has proposed a derivative free algorithm which approximates the square root of the error covariance matrix by the function evaluation based on central point finite difference approach. The author, however, did not mention Stirling Interpolation in his work. This new method suggested in [Schei1997], uses deterministically chosen points (sigma points) to linearize the nonlinear signal model without taking any derivatives. In this approach linearization of nonlinear function is done by considering the first order approximation only and therefore has less estimation accuracy compared to UKF. Nevertheless, this work has developed the background for the Central Difference filter [Ito2000] and the Divided Difference filter. [Norgaard2000].

Central Difference filter (CDF) proposed by [Ito2000] present a mathematically improved algorithm compared to [Schei1997] by considering polynomial interpolation methods for selection of sigma points. Here, linearization is done with second order approximation. In the same year [Norgaard2000] presented an interpolation based Divided Difference filter (DDF) which has a close similarity with that of [Ito2000].

The algorithm in [Norgaard2000] has been derived based on Stirling interpolation formula. The author of [Norgaard2000] has derived the algorithm for propagation of square root of error covariances of DDF and categorized the algorithm as DD1 and DD2 based on the order of approximation. The DD1 considers the first order terms during linearization while DD2, the improved algorithm, considers the second order terms also. It is understood from these works that the DD1 algorithm is similar to that of CDF proposed by [Schei1997] and DD2 is analogous to CDF [Ito2000]. The author of [Norgaard2000] presents the algorithm in a square root framework and the error covariance is obtained subsequently. The algorithm also accommodates the non-additive process and measurement noise. These features are missing in the algorithm of [Ito2000].

An important parameter in these algorithms is the normalized interval length [Norgaard2000]. The choice of the interval length is to be carefully made. An arbitrary choice of this parameter may degrade the performance. For example, with the value of interval length equal to 1, the DD2 algorithm get reduced to DD1 as it is obvious from the steps of algorithm that the contribution of second order approximation becomes nil with this choice. In [Norgaard2000] it is suggested to choose this parameter as the square root of the kurtosis of the distribution (4^{th} moment) of the distribution which happens to be equal to $\sqrt{3}$ for Gaussian noise.

In some situations the performance of DD1 and DD2 are comparably same as reported in an analytical work by [Simandl2009]. It is mentioned that for the systems with high measurement noise covariance (with a comparatively low process noise covariance) and having quasi linear measurement equation performance of DD1 and DD2 are comparatively equal. In [Simandl2009] it is also mentioned that the performance of UKF and DD2 are almost same and significantly improved compared to DD1. In the PhD thesis of Merwe [Merwe2004] the relation between the central difference filter (CDF), divided difference filter (DDF) and UKF has been presented. It can be concluded from the work of [Simandl2009] and [Merwe2004] that DDF/ CDF are performance wise equal with UKF and does not need a number of tuning parameters like UKF.

In [Ito2000], the performance of CDF is illustrated with the help of three nonlinear estimation problems. A first order test problem with strong nonlinearity has been considered which is well known for performance evaluation of nonlinear filter because of its strong nonlinearity in the system dynamics and measurement equation. It has been observed from the results that the performance of CDF is comparable to that obtained from UKF (also called as Julier Uhlman filter in [Ito2000]) and far better than that from EKF. Because of the significant nonlinearity in the system dynamics the estimation performance of both CDF and UKF are not satisfactory even though they are better than EKF. However, for the other case studies the performance of CDF is found to be quite satisfactory.

[Norgaard2000] demonstrated the performance of the proposed filters DD1 and DD2 with the help of a bench mark nonlinear estimation problem of object tracking. It is found that the performance of DD1 is comparable with EKF. As for DD2, it excels over DD1, EKF and even capable of producing better estimates compared to second order EKF.

DDF is also extended as iterated DDF [Shi2008] in the same vein of iterated EKF [BarShalom2001]. The Divided Difference operators have been used in recurrence after getting a recent estimate in each time instant. During iteration, the current mean and the covariance of the divided difference filter (DDF) were used to measurement update step to re-compute the estimate such that more refined value may be obtained. This algorithm shows its superiority over the non iterated algorithm at an additional computation burden.

2.2.3 Gauss Hermite Filter

Gauss-Hermite filter was first reported in [Ito2000] wherein the estimation algorithm is developed using the Gauss-Hermite quadrature rule to numerically evaluate the Bayesian integrals. The integrals encountered in nonlinear Bayesian filtering approach for estimation of nonlinear system perturbed with additive Gaussian noise can be replaced by the weighted sum of the function evaluation of quadrature points generated following the Gauss Hermite quadrature rule.

The Gauss Hermite quadrature rule is a special case of the Gauss quadrature rules for evaluating the Bayesian Integral in presence of Gaussian noise (also termed as Gaussian integrals in [Ito2000]). The Hermite polynomial is intended for Gaussian weighting function present in the integral. This rule is simplified further in [Golub1969] and followed in

[Ito2000] to develop GHF. A tri diagonal matrix is created in [Golub1969] from a three term recurrence formula using Hermite polynomial. Using this tri diagonal matrix the quadrature points and weights are obtained.

Though computationally intensive compared to traditional filters like Unscented Kalman Filters, Divided Difference Filters, Gauss Hermite filters perform commendably well in the situations where there is significant nonlinearity in the system dynamics and the measurement equation. In [Ito2000] the filter has been evaluated using a one dimensional problem as mentioned before. It is observed from the result that the performance of GHF is comparatively better than the other sigma point filters.

Many works contemporary with [Ito2000] seemed to be mentioning the possibility of Gauss-Hermite filter. Subhas Challa and others [Challa1999] contributed a nonlinear filtering algorithm using Generalized Edgeworth Series and Gauss–Hermite Quadrature rule followed by the demonstration of its superiority over EKF using bearing only tracking problem. Fred Daum in his work [Daum2005] has reported about different way of Bayesian filtering techniques wherein the Gauss quadrature rule is preferred for its higher accuracy. The higher accuracy of GHF is also acknowledged in [Wu2006] which provided a unified numerical-integration framework for Bayesian filtering and third order GHF is preferred for estimation at the disposal of sufficient computation power.

In [Arasaratnam2007] the possibilities of Gaussian sum like parallel filters are explored to take care of non Gaussian noise and superiority of such filters over other nonlinear filters is demonstrated. It is reported in [Arasaratnam2007] that use of Gauss Hermite quadrature rule can provide filtering accuracy comparable to that of much more computationally intensive simulation based filters like Particle filters in some applications specifically with relatively small value of measurement noise covariance.

The Square Root version of Gauss Hermite filter is also proposed by the same author in follow up paper [Arasaratnam2008] where it is observed that the performance is quite similar with the GHF with some additional numerical accuracy. A comprehensive description on Gauss Hermite quadrature rule along with necessary illustrations is provided in the in the master's thesis of N. K. Singh [Singh2012] and also in [Chalasani2012]. The work of [Chalasani2012] investigates the performance of GHF for Bearing only tracking problem and also rectified the printing mistake in GH quadrature rule which appears in [Ito2000].

It is realized that this sophisticated algorithm of Gauss Hermite filter suffers from the curse of dimensionality. The computation burden increases exponentially with the dimension of the system. To overcome this drawback the author of [Jia2012] proposed a sparse grid Gauss Hermite Filter. The filter uses weighted sparse-grid quadrature points to approximate the multi-dimensional integrals in the nonlinear Bayesian estimation algorithm. For this filter the number of sparse-grid quadrature points is a polynomial of the dimension of the system unlike the conventional GHF and overcomes the curse of dimensionality.

For more accurate estimation of the systems with higher order dynamic equation a modified GHF is proposed by [Singh2013] and named as transformed Gauss-Hermite filter. An orthogonal transformation has been applied on Gauss-Hermite quadrature points in order to obtain the newly transformed quadrature points using which more accurate estimates of states are obtained. However, this algorithm also suffers from the dimensionality problem like ordinary GHF.

2.2.4 Cubature Kalman Filter

The multi variable moment integrals encountered in nonlinear Bayesian filtering algorithm [Ito2000, Arasaratnam2009] are needed to be approximated using numerical method for the implementation of filtering algorithms. In [Arasaratnam2009] the authors propose a method of numerical integration with the help of a spherical radial cubature rule which is less computationally intensive compared to GHF. The authors proposed a third degree cubature rule which ensures satisfactory estimation performance of the filter. The third degree rule being accurate up to third degree polynomial, it computes the posterior mean accurately and posterior error covariance with an approximate accuracy. For this algorithm the number of quadrature points increases linearly with increase in the dimension of the system. Therefore, for higher order system CKF requires much less computational effort compared to GHF based on Gauss Hermite quadrature rule.

The Cubature Kalman filter algoritms resemble non scaled UKF algorithm and are claimed to be numerically more stable than UKF.. The square root version of CKF is also presented in [Arasaratnam2009] to retain the symmetry and positive definiteness of error covariance.

However, the proposed filter cannot outperform GHF in the context of estimation accuracy. For example, it cannot compute exactly the Gaussian weighted integrals of such simple polynomial functions as $x_1^2 x_2^2$, where x_1 and x_2 are two components of a Gaussian random vector [Jia2013a]. To increase the accuracy of CKF higher degree cubature points are required.

The authors of [Jia2013a] work further on this filter to extend it for arbitrary degree such that 3rd degree cubature rule is only a subset of this higher degree filter. The general algorithm of higher degree cubature Kalman filter is presented in this work which shows that the performance of this filter is comparable with GHF but requires a lower computation effort. Though the authors of [Jia2013a] use the term higher degree only results for fifth degree CKF have been provided.

Later a different method of higher order cubature filter is also presented in [Zhang2014].The authors present a new derivation of the CKFs, which easily lends itself to extension for higher-degree CKF compared to the method of [Jia2013a]. Three consistency conditions which have to be satisfied for fully symmetric cubature rules are introduced here for constructing desired CKFs. Additionally two different types of the fifth-degree CKFs are discussed in details.

2.2.5 Cubature Quadrature Kalman Filter

A new quadrature rule known as cubature quadrature rule has been proposed by [Bhaumik2013] which is published at the same time along with higher order cubature rule reported in [Jia2013a]. In the work of [Bhaumik2013] the author has proposed another version of cubature filter which increased accuracy compared to 3rd degree cubature rule and termed as cubature quadrature rule.

For the proposed quadrature rule the spherical integral is evaluated with 3rd degree spherical rule as in [Arasaratnam2009]. The radial integral is approximated using Gauss Laguerre quadrature rule unlike the moment matching method in [Jia2013a]. For the first degree quadrature rule the algorithm becomes same as that 3rd degree CKF given in [Arasaratnam2009]. The accuracy of CQKF reportedly increases with the increasing order of Gauss Laguerre quadrature rule The authors also extend their work in the square root framework [Bhaumik2014] which is also performance wise same with standard CQKF with its additional numerical advantages.

The same group of researchers also modified their work in [Singh2015] wherein the work of [Jia2013a] has been critically analyzed and the moment matching method followed in [Jia2013a] for computing the radial points and weights is reported to be analytically ambiguous. The authors proposed to combine higher degree spherical rule [Genz2003, Jia2013a] with higher order radial rule by solving higher order Chebyshev-Laguerre equation as mentioned in their previous paper [Bhaumik2013]. This new quadrature rule has been named as higher order cubature quadrature rule. The algorithm demonstrates its superiority over the algorithm of [Jia2013a] and [Bhaumik2013] for some numerical case studies.

2.2.6 Nonlinear Information filters

The nonlinear sigma point filters can also be extended with information filter configuration because this configuration facilitate initialization of the state, ensures positive definiteness of *a posteriori* error covariance and computationally economic with increasing number of measurements [Anderson1979].

It has been reported in literature that the Information filter variant of state estimators is widely recommended for multiple sensor estimation and plays a significant role in many real life applications (e.g. target tracking [Jia2013a], in collaborative sensor networks [Vercauteren2005] and decentralized guidance and control of UAV [Ragi2013]) because of the above mentioned advantages of the information filter configuration. [Anderson1979].

2.2.6.1.Unscented Information Filter

A generalized sigma point information filter algorithm was first reported in [Vercauteren2005] where only the performance of Unscented Information filter (UIF) was demonstrated for decentralized multi sensor fusion problem. The author also demonstrated the poor performance of Extended Information filter (EIF) as the drawbacks of EKF are also inherited by EIF. The sigma points are chosen using Unscented Transformation as given in the paper of Unscented Kalman Filter [Julier2004]. This work was also followed up in [Lee2008]. With the concept of unscented information filter square root unscented information filter was also proposed by [Liu2012] because of its enhanced numerical accuracy, double order precision and preservation of symmetry.

2.2.6.2. Central Difference Information Filter

Another version of sigma point information filter is proposed by [Liu2011] where the sigma points are selected using Stirling interpolation formula and the resulting algorithm is termed as Central Difference Information filter (CDIF). Stirling's interpolation formula is preferred over the unscented transformation as it does not require tuning parameters as in case of unscented information filter. Unless a careful tuning of UIF is achieved its performance may get deteriorated. CDIF is performance wise similar to UIF as demonstrated in [Liu2011] with reportedly low computation effort.

Square Root version of the Central Difference Information filter are also formulated in [Liu2012] together with square root UIF. The square-root central difference information filter (SR-CDIF) is readily available from CDIF. The square roots of error covariance that are computed in the filtering algorithm are updated by QR factorization and Cholesky update. The square-root central difference information filter (SR-CDIF) is numerically more stable than square-root unscented information filter (SR-UIF) as SR-CDIF has only positive weights and therefore a stable Cholesky update. In case of SR-UIF during Cholesky update the positive-definiteness may not be guaranteed due to the negative weights. These filtering algorithms are validated by object tracking problem using multiple radars in [Liu2012].

2.2.6.3. Cubature Information Filter

The concept of generating sigma points using spherical radial cubature rule is also extended for point based information filter. Cubature Information filter (CIF) is proposed in [Chandra2011] which has third degree accuracy as given in Cubature Kalman filter with standard error covariance form [Arasaratnam2009]. The performance of CIF is compared with UIF where it is demonstrated that the performance of CIF is comparable with the latter and sometimes better than that. The author has also extended CIF in the form of square root version of cubature information filter (SR-CIF) as the algorithm ensures numerical efficiency [Chandra2013]. Both CIF and SR-CIF are employed for multi-sensor state estimation. Cubature and Square Root information filter is also extended for correlated process and measurement noise by [Ge2014] and validated with a bearing only tracking problem.

2.2.6.4.Higher degree Cubature & Gauss Hermite Information Filter

The performance of cubature information filter changes substantially with increase in the degree of the cubature rule. As reported in [Jia2013a] the higher degree cubature filter, e.g., fifth degree CKF provides improved estimation result compared to third degree CKF. The author has also extended this concept and developed a high-degree cubature information filter [Jia2013b]. The performance of 5^{th} degree CIF is comparable with Gauss Hermite Information filter and better than CIF, UIF and CDIF. Note that the Gauss Hermite Information filter is a sigma point information filter where the sigma points are chosen using Gauss Hermite quadrature rule. GHIF has not been proposed in any other previous work. The author has used in [Jia2013b] for performance comparison with 5^{th} degree CIF.

2.2.7 Nonlinear filters for non-additive noise

In literature only a few publications exist on nonlinear filters which can accommodate non-additive process and measurement noise. The authors of [Wang2000, Merwe2004] proposed the augmented form of UKF which accommodate non-additive noise. In this algorithm the mean and covariance of noise vector are also computed with the help of sigma points as these cannot be computed directly algebraically as in the case of for additive noise.

In a contemporary paper [Norgaard2000] the authors propose interpolation based DDF which can also take care of the non-additive noise terms. However, augmentation is not required as 2^{nd} order approximation of the nonlinear function of process and measurement noise can be directly obtained using Stirling's interpolation formula.

The augmented form of UKF is also reported in [Sarkka2013a] and subsequently augmented form of GHF, CKF are also presented for non-additive process and measurement noise. Note that apart from [Sarkka2013a] no work exists to the best knowledge of the present worker where nonlinear filters with non-additive noise have been reported.

2.2.8 Optimality of nonlinear estimators

It may be noted from this brief survey as well as from the text and reference books cited that none of the nonlinear filters claim provable optimality in the general sense of the term, In other words estimators that can ensure optimal performance for nonlinear systems are yet to be developed. However, lack of provable optimality does not appear to deter fairly widespread application of such sub-optimal filters in tracking and other applications.

2.3 Literature on Adaptive filters

It has been discussed earlier that successful performance of the estimators (for linear as well as the nonlinear signal models can be ensured) requires. availability of the mathematical model of the system dynamics with adequate accuracy, the measurement equations and the prior knowledge about the distributions and covariances of the process noise and the measurement noise . Inappropriate assumption of any of these noise covariances can degrade the estimation accuracy of the filter and yields suboptimal estimation for both linear and nonlinear signal models [Mehra1972, Maybeck1982].

Process noise had been widely used to take care of modelling inaccuracy of system dynamics including parametric uncertainty, unknown disturbance etc [Simon2006, Zarchan2000]. The covariance of process noise is often unknown in many real time applications. As the objective of the present work includes estimation of unknown parameters along with the states, choice of process noise covariance becomes an important task.

Regarding measurement, the noise covariance can be assigned correctly only after a detailed characterization of the sensor data. It is difficult to assign the measurement noise covariance accurately where the sensor characterization has been partially done or not carried out at all.

Assignment of appropriate noise covariance requires a substantial experimentation and/or offline 'tuning' before employing a filter for real time estimation problem. Such procedures can be replaced by the use of adaptive filters which are capable to adapt process noise covariance ($\boldsymbol{Q}$-Adaptive filters) and measurement noise covariance ($\boldsymbol{R}$-Adaptive filters). This section presents the review on the existing adaptive estimators for linear as well as nonlinear signal models reported in literatures.

2.3.1 Adaptive filter with linear signal models

Adaptive filters for linear signal models have been reported in early works like [Mehra1970, Mehra1972, Myers1976, Maybeck1982] where the methods for adaptation of noise covariance have been presented. The work of [Mehra1972] is considered as a pioneer work in the history of adaptive filters where the methods of adaptation has been categorized into four categories. The rest of this section is organized on the basis of the categorization as mentioned in [Mehra1972] and the works on these particular methods of adaptation has been discussed.

2.3.1.1. Bayesian Estimation of unknown noise covariance

Bayesian estimation method can be employed for estimating the unknown parameters of the system in addition to the states of the system. In situations when the prior knowledge of noise covariances remains unavailable these matrices can also be considered as unknown parameters and can be estimated using the Bayesian method.

In this approach the objective is to find an optimal estimate of the unknown noise covariances ($\boldsymbol{Q}$ or $\boldsymbol{R}$) such that the *a posteriori* probability density of the state is maximized. Such an approach of estimation of noise statistics (both mean and covariance of the noise) was first proposed by Sage and Husa [Sage1969]. The method proposed by Sage and Husa is one of the algorithms for adaptive Bayesian estimation with linear signal models reported in [Sage1969] by the same authors and has been followed later in many works on adaptive Kalman filtering. In the method of Sage Husa [Sage1969] the algorithms for adaptation of the mean and covariance of unknown process noise and measurement noise have been derived by maximizing the *a posteriori* density function of state. Therefore this method can also be considered as Maximum a Posteriori (MAP) estimation. Adaptive Kalman filter based on Maximum a Posteriori (MAP) method estimate the noise statistics (mean and covariance) consistently with the changes of the innovation sequence.

It was admitted by the developers [Sage1969] that the algorithm for adaptation of mean and covariance of process noise and measurement noise based on MAP based method becomes computationally intensive with increase in the dimension of states and measurements. Therefore, the authors have suggested comparatively simpler algorithms on the basis of a few assumptions. However, the optimality of the estimated mean and covariances cannot be ensured with such assumptions. In addition to this it is obvious from the algorithmic steps that the positive definiteness of adapted $\boldsymbol{Q}$ and $\boldsymbol{R}$ cannot be ensured. In [Bavdekar2011] it is also reported that the MAP based estimation of noise statistics often provides biased estimate of true covariance. Expectation maximization method reportedly presents better performance compared to MAP in [Bavdekar2011] as would be discussed in the subsequent subsection.

The work of Sage Husa is referred in several recent contributions. In [Yang2003] same algorithm of $\boldsymbol{Q}$ and $\boldsymbol{R}$ adaptation as in [Sage1969] is used with an additional adaptation factor which is decided on the basis of variance component. In the work of [Narasimhappa2012] the adaptive filter using Sage Husa's method (same as MAP based method) is used to de-noise the fibre optics gyro signal. Here, fading memory approach using a forgetting factor is also incorporated in the basic MAP based algorithm.

To improve the estimation accuracy of MAP based adaptive filter, an AKF algorithm based on MAP estimation with one-step smoothing is proposed in a recent paper by [Gao2015a]. In the conventional MAP based algorithm an exponentially weighted fading memory approach is also incorporated to emphasize the recent measurements. However, in all the above referred modified MAP based estimation algorithms positive definiteness of adapted $\boldsymbol{Q}$ and $\boldsymbol{R}$ cannot be ensured as these are based on Sage Husa approach.

Amongst the algorithms for adaptive Bayesian estimation reported in [Sage1969] the Sage Husa method (MAP based method) of adaptive Kalman filtering has become popular and also extended for nonlinear state estimation. Therefore, MAP based algorithms have been emphasized during the literature review on adaptive Bayesian estimation. The other methods of adaptive Bayesian method reported in [Sage1969] have been rarely referred in literature later and not related to the present work. Therefore, those methods have not been reviewed.

2.3.1.2. Adaptation of noise covariance using Correlation Method

It is mentioned in [Anderson1979] that the innovation sequence from an accurately tuned Kalman filter is zero mean, white and Gaussian. Incorrect value of the system parameters or noise covariances results into the loss of zero mean nature and whiteness of innovation sequence. Consequently the autocorrelation of innovation sequence no longer remains a Kronecker delta function. The adaptation of noise covariance ($\boldsymbol{Q}$ or $\boldsymbol{R}$) using correlation method is developed on the basis of this concept. The correlation method was first introduced by [Mehra1970, Mehra1972] and followed up [Carew1973].

In [Mehra1970, Mehra1972] the author proposes an estimation method for the noise covariance using the autocorrelation function of the output of the system. Alternatively, autocorrelation function of the innovation sequence can also be considered instead of the output. It is mentioned in [Mehra1972] that use of the innovation sequence is preferable as the correlation method based on output becomes restrictive in some situations. The relation between the unknown noise covariance and the autocorrelation function of the output or innovation sequence are presented by a set of equations and are solved using least square method for the unknown noise covariance. The above method is restricted for the unknown noise covariance with constant value. In [Mehra1971] the author has proposed an algorithm where the system parameters (parameters of state transition matrix) are identified using this method.

In [Neethling1974] the authors proposed an alternative approach of estimating noise covariance based on weighted least square method and evaluated with the help of Monte Carlo simulation. The author also pointed out the drawbacks in the method of [Mehra1970, Carew1973] and demonstrated that this method may sometime present biased estimate of noise covariance. The authors of [Oussalah 2000] like [Neethling1974] estimate the unknown covariance with the help of weighted least square method instead of general least square method presented in [Mehra1970, Carew1973]. The rationale behind their proposal of using weighted least square method is to consider the quality of the autocorrelation function of the innovation sequence. The weights are determined using 'Bhattacharyya distance criterion' between the ideal probability and the distribution referring to the current first and second order statistics of autocorrelation functions. In [Oussalah2000] the authors demonstrated the superiority of the method proposed by them over the approach of [Mehra1970].

The authors of [Odelson2006] present a constrained Autocovariance Least Squares (ALS) method for estimation of $\boldsymbol{Q}$ and $\boldsymbol{R}$. The method ensures the positive semi-definiteness of the estimated noise covariance. It is mentioned in [Odelson2006] that the methods by [Mehra1970] follows a three-step procedure to compute the covariances while the method of [Odelson2006] follows only one-step procedure which yields covariance estimates with better accuracy compared to [Mehra1970]. The unbiasedness and better convergence (to the truth value) of adapted covariances with increase in the number of samples is demonstrated in this work. It is also demonstrated that the approach of Mehra present biased estimate and cannot ensure the positive definiteness of estimated noise covariance.

2.3.1.3. Covariance Matching method of adaptation

Covariance matching method which is also known as intuitive method of adaptation has been reported in early works [Mehra1972, Myers1976, Maybeck1982]. The innovation covariance computed during filtering steps (theoretical innovation covariance) is compared with the sample covariance of innovation estimated from the sliding window with a finite length. The expression of unknown noise covariance is computed such that the innovation covariance to be consistent with its theoretical value. This method has been followed to develop the algorithm for ***Q*** and ***R*** adaptation.

The author of [Meyers1976] presents an alternative approach of adaptation algorithm compared to [Mehra1972, Maybeck1982]. Here, both the mean and the covariance of process noise are adapted which has broadened the scope of the application compared to [Mehra1972, Maybeck1982]. The above authors have also introduced a fading memory weighting parameter by which recent observations are emphasized compared to the older observations. This method, however, incorporates another window to maintain the history of *a posteriori* error covariance of previous instants for use in algorithm and therefore computationally more intensive than the method of [Mehra1972, Maybeck1982]. The expression of adapted ***Q*** and ***R*** presented in [Meyers1976, Mehra1972] again cannot guarantee the positive definiteness of adapted matrices. However, [Maybeck1982] presents an alternate algorithm for ***R*** adaptation that can ensure the positive definiteness of adapted ***R*** matrix.

Because of the unavailability of the unique solution of adapted ***Q*** reported in [Mehra1972, Maybeck1982]**,** the scaling factor based ***Q*** adaptation techniques [Hide2003a, Hide2004, Ding2007, Almagbile2010] have also been explored for linear signal models. The work of [Hide2004] is a follow up of previous publication [Hide2003a] where the *a priori* error covariance is scaled rather than ***Q***. This method did not turn out to be promising for navigation problem and scaling of process noise covariance suggested to be a solution in this situation as reported in [Hide2004].

The scaling factor based ***Q*** adaptation was pursued again in [Ding2007] where some further modifications are made provided the measurement noise covariance is precisely known. The same method of ***Q*** adaptation is followed in the recent work of [Almagbile2010]. The scaling factor based method of ***Q*** adaptation does not have the straight forward proof and therefore, needs numerical experimentation before real time application with confidence.

R adaptation algorithm, unlike ***Q*** adaptation, presents a unique solution for adapted ***R*** [Mehra1972, Maybeck1982]. Two different approaches of ***R*** adaptation have been reported in [Maybeck1982] using the statistics of either the innovation or, the residual from filter. The distinction between innovation and residual is nontrivial. While innovation is defined as the difference between actual measurement and *a priori* measurement, residual uses *a posteriori* estimate of measurement. The expression of adapted ***R*** ensures positive definiteness when derived using the residual sequence. However, the possibility of residual based ***R*** adaptation for ensured positive definiteness of ***R*** is not mentioned in [Mehra1972].

The use of residual sequence is recommended in [Maybeck1982] because of its ensured positive definiteness and also preferred in [Almagbile2010]. In [Myers1976] adaptation algorithm for ***R*** is presented based on innovation sequence only. Here also the mean of measurement noise is adapted. However, due to innovation based adaptation positive definiteness of ***R*** cannot be guaranteed here too.

Following the method of covariance matching, a different algorithm adaptive filtering is presented in [Yang 2001a, 2001b, 2003, 2005]. The adaptation method although based on Sage Husa method of adaptation proposed in these publications the use an adaptive factor based on covariance matching method to adjust the contribution of the measurements and the predicted states. Usually the traditional covariance matching method of adaptation performs satisfactorily if the states and measurement errors are stable. When unstable prior states are predicted by the filter adaptive factor can balance the weights between the measurements and the predicted state and controls the ill effects of diverging predicted error. The choice of adaptive factor is a crucial part of design and plays a significant role in navigation.

Two of such adaptive factors were introduced first in [Yang 2001a, 2001b]. A different adaptive factor using the variance ratio of predicted states and observations was also developed in [Yang2003]. These adaptive factors are mostly developed by experimentation. Possibilities of other adaptive factors are mentioned in [Yang2005] and the influence of various adaptive factors on the filtering algorithm is investigated there.

Adaptive Kalman filter proposed by [Jwo2008] is also based on covariance matching method. The ***Q*** and ***R*** adaptation is done using a scaling factor. This factor is the ratio of window estimated and theoretical innovation covariance. However, for ***Q*** adaptation author has restricted its maximum value to be 1. The effect of ***Q*** adaptation can also be achieved by adapting the *a priori* error covariance matrix. The same authors also discussed about the scaling factor based ***P*** (*a priori* error covariance) adaptation in their book chapter [Jwo2009]. Basically, Zhou et al [Zhou1996] proposed this approach with the name of strong tracking Kalman filter. The advantages of this algorithm are: (i) robustness against model uncertainties and (ii) tracking of abrupt changes of states.

The covariance matching $\boldsymbol{Q}$ and $\boldsymbol{R}$ adaptation similar to [Jwo2008] are also employed for fault detection of the linear signal models in [Hajiyev2013a] so that acceptable estimation performance can be obtained. Depending on the residual statistics and using the Chi square method the choice is made between $\boldsymbol{Q}$ and $\boldsymbol{R}$ adaptation. The adaptation is done on the basis of a scaling factor which is restricted to be less than equal to one for $\boldsymbol{Q}$ adaptation like [Jwo2008].

2.3.1.4. Maximum Likelihood Estimation of noise covariance

Maximum Likelihood estimation is a well known approach and originated from Bayesian method of estimation only. This method can also be employed for adaptation of noise covariance. Adaptation based on Maximum Likelihood Estimation (MLE) method was first proposed in [Mehra1972] and followed up in [Maybeck1982] wherein the methods of $\boldsymbol{Q}$ and $\boldsymbol{R}$ adaptation is elaborated and mathematically derived.

From the ML method an unbiased estimate of the required parameter can be obtained with finite covariance using independent and identically distributed measurements. The estimates are so obtained that the computed value can maximize the probability density function of measurement which is expressed in terms of innovation or residual sequence. The innovation or residual of measurement from a sliding window is involved to obtain the expression of adapted $\boldsymbol{Q}$ or $\boldsymbol{R}$ with the assumptions that the elements of covariances are time invariant within the window (epoch) length and the innovation/residuals are white (not auto-correlated).

The method of $\boldsymbol{Q}$ adaptation based on maximum likelihood estimation is investigated in [Mohamed1999]. While [Maybeck1982] has presented the expression of adapted $\boldsymbol{Q}$ in terms of state residual, [Mohamed1999] has used innovation sequence. Use of innovation sequence in place of state residual may reduce the computation cost when the dimension of state vector is more than that of measurement vector. Another nontrivial contribution of [Mohamed1999] is to ensure the positive definiteness of adapted $\boldsymbol{Q}$. As the filter reaches steady state the *a posteriori* error covariance acquires a steady value (often low) and their effects can be overruled to get an expression of the symmetric, positive definite adapted $\boldsymbol{Q}$. This assumption is justified and satisfactory estimation result is obtained using this approximated adaptation algorithm that guarantees positive definiteness of $\boldsymbol{Q}$.

In the work of [Mohamed1999] and [Maybeck1982] two different $\boldsymbol{R}$ adaptation methods have been derived using MLE method. Here also the residual based method is preferred as the derived expression of adapted $\boldsymbol{R}$ automatically ensures positive definiteness of adapted $\boldsymbol{R}$ matrix. However, the innovation based $\boldsymbol{R}$ adaptation algorithm cannot ensure the positive definiteness of $\boldsymbol{R}$.

The approach of MLE based adaptation is followed in [Hide2003b]. The $\boldsymbol{Q}$ and $\boldsymbol{R}$ adaptation formula have been taken from [Mohamed1999]. The author also compared this approach with Multiple Model Adaptive Estimation where a bank of Kalman filters is used. For attitude estimation of a moving object the $\boldsymbol{Q}$ and $\boldsymbol{R}$ adaptation techniques by [Mohamed1999] is followed in [El-Mowafy2005]. However, the residual based $\boldsymbol{Q}$ adaptation algorithm presented in this publication seems to be incorrect. Additionally, a Gauss-Newton iteration method within a single epoch is employed to minimize the linearization bias in face of poor initial estimation or large disturbances.

The MLE method was reported in an earlier paper by [Kashyap1970] where the unknown parameters are computed using optimization techniques. $\boldsymbol{Q}$ and $\boldsymbol{R}$ can also be obtained using MLE method where the unknown covariance is obtained following the gradient-based numerical optimization methods so that the computed parameters maximize the likelihood function. These optimization based methods need more computation time compared to the other method as gradient based numerical optimisation are required to obtain the estimate of unknown covariance after convergence.

2.3.1.5. Expectation Maximization method of adaptation

Expectation Maximization is an alternative way of parameter estimation following the optimization based maximum likelihood estimation technique [Kashyap1970] where the usual gradient based numerical optimization approach of maximizing the likelihood function are replaced by iterative method as reported in [Shumway2000, MohanM.2015]. These works have been originated from an earlier work by [Dempster1977].

The expectation maximization (EM) technique was developed by Dempster and applied for estimation of covariance components of a linear signal models [Dempster1977]. On the availability of the complete data likelihood function the unknown parameters are estimated iteratively and this method does not require the derivative calculation.

The EM method, as its name suggests, consists of expectation and maximization steps. In EM method proposed by [Shumway2000] states are estimated using Kalman smoother with a guessed initial value of the unknown parameters. The unknown parameters are then estimated by Maximum Likelihood method. The above method of estimating the states with the help of a Kalman smoother and optimizing the parameters using ML method is repeated until the convergence of the parameters is achieved. The algorithm presents analytical expressions for the iterations of the parameters and therefore computations of the gradient are not required.

In a recent work by [Zagrobelny2014] the authors followed an optimization based MLE method for adaptation of process and measurement noise covariance. The measurements are presented with the help of a normal distribution where the variance of distribution is expressed in terms of the unknown process and measurement noise covariances. The likelihood function is then maximized for the optimal choice of unknown covariances using optimization techniques.

2.3.1.6. Variational Bayesian approach of adaptation

The variational Bayesian approach of adaptive Kalman filtering is first proposed by [Sarkka2009] which also comes under the category of Bayesian method of estimation of noise covariance. In this work the situation has been considered where the distribution of the measurement noise along with the noise statistics (parameters) of distribution remains unknown. The author has proposed an algorithm of adaptive Kalman filter using variational Bayesian approximation where joint estimation of states and the noise parameters is possible.

The method is based on variational approximation of the joint posterior distribution of states and noise parameters which has to be considered on each time step separately. Thus a recursive algorithm is obtained, where for every step, the state is estimated with Kalman filter and the diagonals for the measurement noise covariance are estimated with fixed point iteration. The m diagonals of the covariance of m-dimensional measurement noise are assumed stochastic and follow independent dynamics.

Approximation of the joint posterior distribution of the state and the noise variances by a "factorized free form distribution" is possible by the method reported in this work. This way of adaptation, though widely different from the other approaches of adaptive Kalman filter, has similarity with the basic concept of covariance matching technique [Maybeck1982] where the adapted covariance should remain consistent with observed innovation or residual.

Another work by the same author has been proposed in [Sarkka2013b] with nonlinear signal models for non-additive or nonlinear noise. The author of [Gao2011] used the above concept of estimating state and noise parameters for centralized sensor fusion where the complete knowledge of sensor noise remains unavailable. To reduce the computational burden of centralized fusion authors present sequential centralized fusion algorithm.

The variational Bayesian method is also employed in [Sun2012] to estimate the states in linear dynamic systems with unknown inputs. When the knowledge of noise remains unavailable the author considers system state, unknown inputs and time-varying noise parameters as hidden variables and presents an algorithm based on variational Bayesian method to learn the structure of hidden variables and approximate the joint posterior distribution of system state, unknown inputs and time-varying noise parameters.

2.3.1.7. Findings from the review on adaptive Kalman filters

In course of the review some significant points which are found common to several papers on adaptive Kalman filtering have been enumerated below.

For $\boldsymbol{Q}$ adaptation methods presented in [Mehra1972, Maybeck1982] both the authors admit that the uniqueness of the adapted $\boldsymbol{Q}$ is possible only for some restricted situations. Whenever the number of observation is greater than or equal to the number of states or the measurement matrix is of full rank unique solutions for *a priori* state error covariance and the process covariance can be obtained. When the above conditions do not hold, some restrictive approximations, e.g., $\boldsymbol{Q}$ as a diagonal matrix may be assumed. Alternatively, the pseudo inverse of measurement matrix needs to be computed. Above all, the filter is considered to be reached steady state during the most recent estimation window. However, for $\boldsymbol{R}$ adaptation no such constraints have been reported.

It is to be noted that in a few works [Myers1976, Almagbile2010] as reported above simultaneous adaptation of $\boldsymbol{Q}$ and $\boldsymbol{R}$ has been carried out and satisfactory results have also been demonstrated. A few researchers [Mehra1972, Maybeck1982, Mohamed1999] have admitted the drawback of simultaneous adaptation of $\boldsymbol{Q}$ and $\boldsymbol{R}$. This indicates that even though acceptable results are obtained for some cases use of algorithms with simultaneous adaptation of ***Q*** and ***R*** cannot be used with confidence and should be evaluated with offline simulations before real time application. The same comments would also be applicable for adaptive nonlinear filers.

In [MohanM.2015] the author reports on the review on adaptation method that the effect of ***Q*** and ***R*** are conflicting in nature which is the cause of poor estimation during simultaneous adaptation of these two covariances. In [MohanM.2015] the author mentioned that for simultaneous adaptation of ***Q*** and ***R*** using optimization based MLE method consideration of a special cost function is necessary based on the statistics of prior, post, and smoothed state estimates and their covariances. Improper combination of such statistics cannot lead to satisfactory performance of filter as. ***R*** is over estimated while ***Q*** is under estimated and vice versa. The authors cite a reference [Gemson1991] where ***Q*** and ***R*** are adapted alternately. To bypass the problems of simultaneous adaptation, the filter gain can also be adapted in place of adaptation of noise covariances [Mehra1972]. This method although may ensure the optimal performance of filter cannot present the adapted value of noise covariance which may be necessary for the further analysis.

Note also that most of the papers on AKF where the algorithm is presented using Kalman filters as its core have been demonstrated with nonlinear estimation problems. In that situation the need of linearization of the system and measurement equation is essential. Consequently the algorithm which is termed as AKF in the paper becomes equivalent to adaptive EKF during demonstration of that algorithm using nonlinear estimation problem. However, during review these works have been categorized as AKF following the theme of the paper.

2.3.2 Adaptive filter with nonlinear signal models

The concept of adaptive Kalman filters for linear signal models are also extended for nonlinear signal models using the non-adaptive nonlinear filters as the underlying framework. The adaptation algorithms are integrated in the non-adaptive nonlinear filtering algorithms so that the corresponding adaptive nonlinear filter can be formulated. In this way Adaptive version of EKF, UKF, DDF, CKF have been developed and reported in literature which are summarized in the following subsections.

2.3.2.1. Adaptive Extended Kalman Filter

The adaptation algorithm for linear signal models based on different methods can be applied for nonlinear systems using EKF as a base. An early work by [Maybeck1981] presents a ***Q*** adaptive EKF based on MLE method [Mehra1972]. In [Maybeck1981] adaptation algorithm additionally uses a modulating factor (may be termed as forgetting factor as well) to regulate adaptation speed. Use of MLE based ***Q***-Adaptive EKF is also found in [Busse2003]. In this work the same method of adaptation of ***Q*** [Maybeck1981, Maybeck1982] is followed and in the similar way a modulating factor is used which has been explained as a moving average method for refinement of adapted ***Q***. The only change in adaptation algorithm is that here the previous history of innovation sequence is not used by making the window length equal to 1, i.e., it considers only the value for the current instant. However, these algorithms for ***Q*** adaptation may suffer from singularity problem.

[Busse2003] also proposes ***R*** adaptive EKF based on MLE. However, the algorithm presents incorrect expression of adapted ***R***. While the residual based ***R*** adaptation method (as in [Maybeck1982]) has been followed here the expression of adapted ***R*** presented is similar to that of innovation based adaptation.

Q and ***R*** adaptive EKF is reported in [Han2009a]. MLE based approach of adaptation as in [Mohamed1999] has been followed to ensure the positive definiteness of adapted ***Q***. However, innovation based ***R*** adaptation again may suffer from loss of positive definiteness. The author of [Zeng2012] has presented an algorithm of adaptive EKF combining with particle swarm optimization technique to deal with the state constraints. As for the adaptation MLE approach is followed. The adapted ***Q*** and residual based adapted ***R*** ensure the positive definiteness. In the paper, however, the author uses the concept of residual but coins the term 'innovation' instead of 'residual'. The adaptive EKF in [He2015] uses the ***Q*** and ***R*** adaptation method of [Mohahmed1999] and appropriately modify the algorithm with the help of the computed derivatives of system and measurement model.

Bavdekar et al. [Bavdekar2011] follows two different approaches of adaptive EKF : (i) Optimization of Maximum Likelihood function, (ii) Expectation Maximization method for nonlinear systems based on the extended Kalman filter. In the ML based optimization approach the likelihood function based on the innovation sequence is directly optimized using a constrained nonlinear programming strategy, sequential quadratic programming. In the EM method the conditional density function of the states and measurements is maximized to compute the next iterate of the decision variables for the optimisation problem. In this method derivative calculation is replaced by function evaluation for estimation of the noise covariances.

Adaptive Extended Kalman filter with unknown input is proposed in [Cavusoglu2014] where the adaption is based on minimization of cost function which is selected as a quadratic function of innovation. Subsequently the computed scaling factor is obtained to tune the *a posteriori* error covariance and satisfactory state estimate is presented.

Based on covariance matching method $\boldsymbol{Q}$ and $\boldsymbol{R}$-adaptive EKF is developed in [Lippiello2007]. The adaptation method as in [Myer1976] has been followed. Note that the positive definiteness of adapted $\boldsymbol{Q}$ is not guaranteed here too. For $\boldsymbol{R}$ adaptation innovation based approach is followed and duly modified to suit for visual motion estimation. However, innovation based $\boldsymbol{R}$ adaptation algorithm cannot guarantee the positive definiteness of adapted $\boldsymbol{R}$ as said before. Note also that the apart from covariances, the mean of process and measurement noise are also adapted as it is recommended in [Myer1976].

In presence of faulty measurement, an Adaptive EKF is proposed by [Hajiyev2011] where the filter gain is adapted instead of noise covariances based on the evaluation of the *a posteriori* probability of fault free system. The adaptation of filter gain is carried out with the help of the posterior probability density of the normalized innovation sequence at the current estimation step.

An adaptive EKF based on strong tracking algorithm is proposed in [Xia1999] where the *a priori* error covariances is tuned using a scaling vector. The scaling factor is computed based on innovation sequence. This algorithm is found to perform well in the situation when system dynamics is affected by unknown disturbances.

Author of [Meng2000] has proposed both $\boldsymbol{Q}$ and $\boldsymbol{R}$ adaptive EKF where along with the covariances the mean of noises are also adapted. The covariance matching method [Myers1976] has been followed and innovation sequence is employed to obtain the adaptation algorithms. The speed of adaptation is regulated using a forgetting factor.

In [Jiancheng2011] EKF algorithm is presented where window estimated innovation-covariance is used to adapt the gain of the filter following the covariance matching method.

When measurement noise covariance remains unknown, gain of filter is adapted instead of adaptation of $\boldsymbol{R}$.

In [Wang2015] adaptive EKF following covariance matching method is also developed with a variable forgetting factor. The forgetting factor which is constrained to be less than equal to unity is decided on the basis of the comparison of window estimated and filter computed innovation covariance.

Following the MAP based estimation of noise statistics the authors of [He2011] proposed an AEKF algorithm where the mean and covariance of process and measurement noise are adapted. Here also the speed of adaptation is regulated with a forgetting factor.

However, the adaptive EKF suffers from the well known shortcomings of EKF. The adaptive filters based on sigma point filter can overcome the drawbacks of adaptive EKF as it is demonstrated in the papers on adaptive sigma point filter. In the following subsection the works on adaptive sigma point filters have been reviewed.

2.3.2.2. Adaptive Unscented Kalman Filter

Works on adaptive Unscented Kalman filters are plenty. These are presented below by categorizing them based on the different approaches of adaptation as discussed before for adaptive linear estimator.

MLE based AUKF

In [Lee2004, Lee2005] $\boldsymbol{Q}$ adaptive UKF is first proposed as one of the adaptive sigma point filters which has been applied on satellite attitude estimation problem. The proposed adaptation rule is an extension of MLE based $\boldsymbol{Q}$-adaptation formula for linear estimation problem. However, the mathematical derivation of the adaptation algorithm is not carried out in [Lee2004, Lee2005].

A method for appropriate choice of window size for the innovation window is also reported here which is another nontrivial contribution of [Lee2004, Lee2005]. An optimization based technique, namely, the Powell's method was followed for such appropriate choice of window size. The cost function which is a quadratic expression of innovation vector obtained from Monte Carlo simulation is to be minimized for the choice of appropriate window length. This work is followed up in [Soken2014] where the $\boldsymbol{Q}$ adaptive UKF is used for bias estimation. The author has also derived the mathematical expression of the adaptation algorithm following Maximum Likelihood Estimation method by defining a pseudo measurement matrix of nonlinear observation equation.

The complementary algorithm for ***R*** adaptation is reported in [Chai2012] which is inspired from the MLE based ***R*** adaptation [Mohamed1999] method for linear signal models. This work follows residual based ***R*** adaptation for the ensured positive definiteness of adapted ***R*** matrix. However, the adaptation algorithm has not been derived in this paper. The adaption algorithm has some errors which are pointed out in and rectified in [Das2013, Das2015]

Covariance matching method for AUKF

A scaling method based ***Q*** adaptive UKF by [Soken2011, Hajiyev2014] has been reported in the literature where the scaling method which has been reported in [Hide2004, Jwo2008, Hajiyev2013a] is extended for UKF framework. However, in place of innovation based adaptation residual based adaptation has been proposed for ***Q*** adaptation. However, there is no added advantage (e.g., ensured positive definiteness) of using residual in lieu of innovation sequence for ***Q*** adaptation. The adaptation approach is similar to intuitive covariance matching method and does not present a formal mathematical derivation like [Soken2014].

In the same vein scaling factor based ***R*** adaptive UKF are also proposed by the same authors [Hajiyev2014, Soken2012, Soken2011, Soken2009]. However, for ***R*** adaptation the innovation sequence is introduced and the maximum value of the scaling factor is restricted to be 1. In most of the cases these algorithms are employed for fault detection. Depending on the nature of fault ***Q*** or ***R*** adaptation algorithm is to be employed and this choice of adaptation algorithm is made by statistical measure of innovation or residual with the help of Chi square test.

Based on the covariance matching method of [Myers1976] an adaptive UKF has been formulated by [Jargani2009]. The ***Q*** adaptation algorithm has been modified from [Myers1976] with some reasonable approximation so that the expression of the adapted ***Q*** can assure the positive definiteness of ***Q***. In addition to ***Q*** adaptation, both innovation and residual based ***R*** adaptation methods are also proposed in work. The proposed algorithm is demonstrated to be superior to the fading memory based AUKF.

In [Cao2009] the *a priori* error covariance (***P***) is scaled instead of ***Q*** to make the UKF adaptive in face of unknown process noise statistics. Innovation covariance is introduced to compute the scaling factor (may also termed as forgetting factor) which consequently tunes the filter gain and presents a better estimate of state.

In [Fathabadi2009] new variety of AUKF is involved which is designed for asynchronous measurements. In absence of accurate dynamic model for the system a forgetting factor is chosen based on covariance matching method which scales *a priori* error covariance (***P***) like [Cao2009]. In addition to this when measurements are also uncharacterised the filter gain (***K***) is tuned by utilizing both the forgetting factor and a new scaling factor computed using innovation covariance. In this way the satisfactory state estimate is obtained here.

Another ***Q*** and ***R*** scaling methods are shown in [Huy2012], where the scaling factors are computed using optimal Downhill Simplex search technique so that the computed scaling factor minimizes a cost function which is an implicit function of the measurements. This cost function needs to be decided based on the application.

Strong tracking Unscented Kalman filter is proposed in [Li2010, Tao2014] for correlated process and measurement noise. The strong tracking filter is a kind of adaptive filter where the *a priori* error covariance is tuned with a scaling factor as mentioned before. This is useful when there is uncertainties in the system dynamics as discussed before. To compute the scaling factor estimated innovation covariance from the sliding window is used. This is also an intuitive method and may be categories under covariance matching method.

An adaptive UKF has been employed in [Wanxin2011] where the scaling factor based adaptation algorithm has been proposed. The error covariance matrix of *a priori* estimate of measurement and the cross covariance matrix are tuned using the scaling factor which is derived based on covariance matching technique.

In [Xia2014] an Adaptive UKF is proposed for both ***Q*** and ***R*** adaptation. Depending on the value of residual it is decided whether to increase or, decrease ***Q*** by multiplying a scaling factor. This algorithm tune ***Q*** or ***R*** in an ad hoc process which lacks rationale of using such scaling factors. The algorithm is employed for fast identification of a machine tool selected point temperature rise.

The algorithm of AUKF reported in [Chai2012] has a few errors in adaptation steps which have been rectified in [Das2015, Das2013]. As the adaptation algorithm in [Das2015, Das2013] is obtained analogous to that of the linear system without formal derivation, these $\boldsymbol{R}$ adaptation algorithms may also be categorized under intuition based covariance matching method. Both innovation and residual based adaptation methods are reported in [Das2015, Das2013]. Though these algorithms present comparable estimation result, the latter has an additional advantage of ensured positive definiteness as mentioned before. For innovation based $\boldsymbol{R}$ adaptation an ad hoc approach has been followed in [Das2015] before they are employed for nonlinear estimation.

MAP based AUKF

The Maximum a Posteriori method based noise statistics estimator is introduced in UKF by [Zhao2009] where the mean as well as covariance of process and measurement noise are estimated. The posterior density function is presented in terms of the innovation sequence which has to be maximized by correct choice of first two moments of noise. The proposed adaptive UKF is demonstrated to present satisfactory estimation performance by online adaptation of noise statistics when the measurement noise covariance varying with time. However, demonstration regarding $\boldsymbol{Q}$ adaptation is not found in [Zhao2009].

Following the work of [Zhao2009], the author of [Cheng2014] demonstrated the satisfactory estimation performance of both $\boldsymbol{Q}$ and $\boldsymbol{R}$ adaptation algorithm in UKF framework for MEMS/GPS integrated navigation. However, the adaptation algorithm for $\boldsymbol{Q}$ presents an approximated expression of adapted $\boldsymbol{Q}$ and correctness of the expression has not been justified in the paper.

For systems with time varying process and measurement noise covariance an adaptive UKF is proposed by [Gao2015b] which employs a random weighting technique along with Sage windowing approach (MAP) for estimating and tracking the unknown time varying process noise and measurement noise covariance. The innovation sequence is employed for deriving the expression of adapted noise covariances. The limitation of the windowing approach is that the appropriate choice of window size for adaptation of unknown time varying noise covariance has to be decided after experimentation. Infusing the random weighting technique this limitations are overcome and satisfactory tracking of time varying elements of $\boldsymbol{Q}$ and $\boldsymbol{R}$

is possible with the help of such automatically chosen weighting factor. The adaptation methods are presented in this work with the help of theorems along with proofs. The satisfactory performance of the proposed algorithm is demonstrated with numerical problems.

An adaptive UKF has been proposed by [Liu2009] for non white noise. The proposed algorithm is applied for GPS based position estimation problem where the system dynamics is perturbed with additive non-white Gaussian noise. The $\boldsymbol{Q}$ adaptation algorithm has been claimed to be derived from Sage Husa filter without derivation of the algorithm. A forgetting factor is also included to control the speed of adaptation. However, no further steps are incorporated to take care of the non white noise.

AUKF based on MIT Rule

The AUKF based on complex MIT adaptation rule is also found in [Jiang2007, Han2009b]. The implementation of this method is often impractical because of large number of partial derivative calculations.

2.3.2.3. Adaptive Divided Difference Filter

Adaptive Divided Difference Filter is first introduced along with AUKF in [Lee2004, Lee2005] as referred before wherein the $\boldsymbol{Q}$ adaptation was the focus of research. Adaptation is based on $\boldsymbol{Q}$-Adaptive Kalman filter as reported in [Maybeck1982].

The expression of adapted $\boldsymbol{Q}$ is modified from that of [Maybeck1982] and the modifications are made following Adaptive EKF [Busse2003]. In this work the final value of adapted $\boldsymbol{Q}$ of current instant is obtained from by the moving average of adapted $\boldsymbol{Q}$ of current instant and the previous instant with the help of a tuning parameter. The same idea is followed by [Lee2005, Lee2004] and the tuning parameter is obtained by optimization technique using Powell's method. However, the work of [Lee2005] can neither assure the positive definiteness of adapted $\boldsymbol{Q}$ nor can it guarantee the symmetry of adapted $\boldsymbol{Q}$. In addition to this the $\boldsymbol{Q}$ adaptive DDF proposed in [Lee2005] uses only first order approximation.

A Robust Adaptive second order Divided Difference Filter is proposed by [Karlgaard2010] which uses the concept of adaptation of $\boldsymbol{Q}$ reported by [Myer1976] and modified this concept of adaptation in presence of outlier. As developed from the methods proposed by [Myer1976] it inherits the drawbacks of this method as reported earlier. An innovation based $\boldsymbol{R}$ adaptation method based on [Myer1976] has also been presented in [Karlgaard2010]. The work of [Karlgaard2010] focuses on the robustness rather than adaptation. Therefore, Huber based Divided Difference filter is proposed for robust estimation in presence of outliers. Additionally when the noise covariance is unknown, they are adapted accordingly. However, this method is also suffers from computational burden and cannot ensure positive definiteness of adapted $\boldsymbol{R}$.

An adaptive formulation of second order DDF [Subrahmanya2009] has the emphasis on robustness instead of adaptation. The upper bound on error covariance matrix ($\boldsymbol{P}$) is derived so that the filter can be made robust to modelling uncertainties. In the algorithm of [Subrahmanya2009], (i) the parametric structure of upper bound has been evaluated and one of its parameters is determined following adaptive fading memory approach and (ii) tuning of the other parameters of filter is based on a combination of on-line and off line tuning, (iii) the measurement equation is constrained to be linear.

2.3.2.4. Adaptive Cubature Kalman Filter

Adaptive nonlinear filters are also formulated using Cubature Kalman filter as the underlying framework which is non-adaptive per se. The non-adaptive Cubature Kalman filter is a derivative free sigma point filter and comparable with its other relatives like UKF, DDF as discussed in the section on nonlinear filters. The adaptive cubature filter has several variants which may be classified based on adaptation approaches e.g., strong tracking approach, variational Bayesian approach, Maximum a Posterior (MAP) approaches and some combinations of these methods.

The adaptive CKF was first proposed in [Sarkka2013b] where the variational Bayesian method has been followed where the distribution of measurement noise remains unknown. The proposed method addresses an apparently similar estimation problem as discussed in AUKF, ADDF. However, in this work the distribution of the measurement noise is also unknown unlike the other adaptive nonlinear filters where the distribution of the measurement noise is assumed to be known (Gaussian). For such estimation problem a different solution methods has been proposed in [Sarkka2013b] based on variational Bayesian approach. This is basically the extension of the previous work [Sarkka2009] for state estimation of linear signal models. The measurement noise is adapted with the help of a different dynamic model. Though the adaptive CKF has been demonstrated in [Sarkka2013b] possibility of obtaining AUKF, AGHF using same method has also been mentioned.

Adaptive CKF (in square root framework) is also reported in [Ge2011] under the category of strong tracking filters for the situations when the system dynamics is susceptible to unknown parameter variations/ disturbance. In case of strong tracking filters a modulating factor is adaptively chosen based on innovation sequence and regulate the *a priori* error covariance. This would in turn adapt the filter gain so that satisfactory estimation performance may be obtained. Such an algorithm has been reported in where the adaptation algorithm is presented for a special situation where process and measurement noise are correlated.

A similar work has been reported by the same author in [Ge2014] where it has also been considered that the measurement noise (assumed Gaussian) covariance is unknown. The unknown noise covariance is adapted using the variational Bayesian approach here. However, in this work author did not consider the correlation between process and measurement noise. The information filter configuration is considered as an underlying framework for economic computation during multiple sensor fusion.

Adaptive CKF is also proposed in [Benzerrouk2013] which may be classified as strong tracking filter and applied for nonlinear state estimation with non Gaussian measurements. Adaptive iterated SR CKF is also proposed in [Chen2013] where for adaptation the strong tracking method has been applied to adapt the square root of predicted error covariance.

In [Tang2012] a $\boldsymbol{Q}$-adaptive CKF in square root approach is proposed where the adaptation steps of process noise covariance is presented which may be derived using MLE method. The work is a follow up of the work of [Lee2004]. The window length for adaptation has been automatically chosen using the simplex rule of optimization as in [Lee2004]. The estimation of accuracy of [Tang2012] based on square root CKF framework is expected to be comparable with $\boldsymbol{Q}$ adaptive UKF and improved compared to $\boldsymbol{Q}$ adaptive DDF (first order) [Lee2004]. Like the base paper here also derivation of the $\boldsymbol{Q}$ adaptation algorithms is not provided. Another adaptive CKF presented in [Xia2015] where the concept of $\boldsymbol{Q}$ adaptation and residual based $\boldsymbol{R}$ adaptation algorithm [Mohamed1999] has been implemented.

In [Yu2014] an adaptive cubature filter has been proposed for the situation where the first two moments of both the process and the measurement noise covariances are unknown. It is reported that following the Sage Husa method (MAP), the mean and covariances of the process and the measurement noise have been adapted. Additionally a fading memory factor is also been used. However, the adaptation formulae have not been derived here.

The adaptive CKF in square root approach is presented in [Liqiang2015] where maximum a posterior (MAP) based approach has been followed to adapt the unknown noise covariance of process as well as measurement noise. The square root version has been preferred for ensured positive definiteness of error covariance. The mathematical derivation of adaptation algorithms in square root approach have also been presented in the paper. To make the filter robust in the face of modelling uncertainty the strong tracking method has also been adopted in this work.

Convergence analysis of Cubature Kalman filter has been addressed in the work of [Zarei2014] where the effect of process noise covariance has been investigated and adaptation of process noise covariance is proposed to deal with large estimation error. For adaptation of process noise covariance MLE approach as in [Lee2004] is followed. It is demonstrated that adaptation of process noise covariance influence the convergence of the filter. Performance of an alternate algorithm of CKF modified with fuzzy logic is also compared with ACKF in the perspective of convergence by initializing a Gaussian prior (initial estimate) with large value of error covariance.

A different version of adaptive filter based on cubature rule is also reported in [Chen2012]. The authors propose an adaptive CKF for joint estimation of parameters and states without augmenting parameters. Here adaptation gain is chosen based on minimizing the recursive weighted least squares of the prediction error. To make the filter resistant to modeling uncertainties risk sensitive filtering algorithm has been followed.

2.3.3 Application of Adaptive filters

It is observed from the literature survey that the adaptive estimators have been prevalently used in attitude estimation, navigation of the land vehicles and the vehicles for aerospace and marine applications. For integrated navigation systems sensors used are inertial (accelerometers and gyroscopes) as the main system with external aid provided by GPS and GLONASS, Galileo or Beidou receivers. This method is known as Global Navigation Satellite System “GNSS” solutions. At present, inertial sensors are usually low cost where most sensors are MEMS (Micro Electrical Mechanical Systems) based.

Despite their widespread use in navigation systems following drawbacks have been identified which need special treatments e.g., use of estimators and filtering algorithms for reliable navigation. The inertial sensors are prone to have biases and drifts growing with time. Therefore another technology of sensors (e.g., GNSS receivers) is needed to bound these drift/ bias. Satellite-based systems such as GNSS although provide high precision measurements suffer interference from the other services in the respective frequency band. Moreover, while used for land vehicle navigation the accuracy of GNSS may deteriorate in urban area with high rise buildings and under dense foliage. Vehicular navigation is often characterized with dynamics changes in motion and exposed to unknown disturbances. The accuracy of the dynamic model gets affected as a consequence. The above discussed issues motivated researchers to investigate the performance of adaptive estimators in navigation.

Use of adaptive estimators for GPS/GNSS and INS fusion based navigation system is observed in many works including some recent papers. In [Hide2004] AKF is validated using a low cost Crossbow MEMS IMU integrated with carrier phase GPS integrated navigation system for a marine application. It was demonstrated that the time required to initialize the sensor errors and to align the INS has been reduced, navigation performance is improved using AKF. In [Hide2003b] AKF is demonstrated with the GPS and inertial data simulation software. A trajectory taken from a real marine trial is used to test the dynamic alignment of the inertial sensor errors.

The author of [El-Mowafy2005] suggests that the attitude of a moving vehicle can be determined using a GNSS multi-antennae system by rigidly mounting three antennae on the vehicle's external surface. Two antenna-to-antenna vectors can be used to represent the attitude change of the vehicle. AKF has been used by the author for the estimation of the attitude states. Experiments on the integrated Strap-down Inertial Navigation System/ Doppler Velocity Log (DVL) system for marine application have been performed in [Gao2015a]. It is demonstrated that the proposed AKF improves the estimation accuracy effectively and robustness in the presence of vigorous-maneuvers and rough sea conditions.

In [Jiancheng2011] AEKF is applied to In Flight Alignment for the SINS/GPS integrated Position and Orientation measurement System (POS) with a large initial heading error. Its performance is demonstrated under unstable GPS measurement, including the situations of the changes of the statistical characteristics of the measurement noise and the existence of isolated outliers. In [Meng2000] the author has used AEKF for Localization and Mapping of mobile robots.

Performance improvement using Adaptive UKF is demonstrated in [Akca2012] for tightly-coupled INS/GPS especially at the end of GPS outage periods. For indoor vehicle navigation integrating Wi-Fi measurements and MEMS based INS, superiority of AUKF has been demonstrated in [Chai2012]. The author of [Cheng2014] demonstrated MAP based AUKF in simulations that are conducted for MEMS/GPS integrated navigation system. The author of [Wanxin2011] has demonstrated the use of AUKF to improve the initial alignment accuracy and convergence rate of the Strap-down INS system.

Experimental results for AUKF are presented in [Gao2015b] for observation of an unmanned aerial vehicle (UAV) which uses a Strap-down Inertial Navigation System/Satellite Positioning System integrated navigation system. The author of [Liu2009] validates AUKF for the GPS based position estimation problem using real satellite data. In terms of the GPS system error characters, the proposed AUKF builds a model of the propagation error, and provide online estimate of its covariance. AUKF is also recommended for magnetometer calibration and attitude parameter estimation in [Soken2012]. The magnetometer biases are estimated as well as the attitude and gyro biases using Q-adaptative UKF in [Soken2014].

Adaptive sensor fusion of INS/GNSS CKF is considered in [Benzerrouk2013]. In the circumstances when GNSS outliers supposed to occur during specific interval of time, innovation based adaptive approach is selected and used to adapt the covariance of CKF and demonstrate satisfactory estimation performance. The author of [Georges2015] proposes variational Bayesian based Adaptive Cubature Smoothers and recommends its use in the presence of colored and variational process noise. VB-ACKS is able to provide a better position error in the presence of dynamic variation of the vehicle and the INS sensor error variation. In [Tang2012] authors demonstrate the performance of ACKF in simulation with a spacecraft attitude estimation problem. Filter is designed for quaternion based attitude estimation with the quaternion normalization constraint.

An adaptive SLAM based on the CKF method is proposed in [Yu2014] for Simultaneous localization and mapping of mobile robots. Maximum a Posterior (MAP) based adaptation method overcomes the SLAM problems, e.g., unknown and uncertain environment description and noise characteristics of sensors.

Despite the navigation application adaptive estimators are also employed for other real world problems to carry out estimation duty. Adaptive estimators may find application in target tracking. In an early work by [Maybeck1981] Q Adaptive EKF is employed for tracking of highly maneuverable targets. 'Bearing only tracking' problems are considered in [Ge2011, 2014] for validation of adaptive sigma point filters. In [Liqiang2015] the target tracking model is a 5-dimensional nonlinear system, where an aircraft executes maneuvering turn in a horizontal plane with unknown turn rate. ACKF is used for estimation of trajectory.

AEKF in [Busse2003] is employed for low earth orbit formation estimation. In [Kardgaard2010] ADDF is applied to the six-degree-of-freedom elliptical orbit rendezvous navigation of a satellite. Measurement data to the navigation filter are obtained from a sensor suite consisting of optical sensor, an inertial measurement unit (IMU), a star tracker, and a generic orbit sensor. In [Lee2004, Das2013] AUKF, ADDF are used for state estimation of the spacecraft trajectory in a low earth orbit.

Adaptive estimators have also been employed for state estimation and control of continuous stirred tank reactor (CSTR) [Cao2009, Fathabadi2009, Jargani2009]. AUKF has been validated in simulation for state estimation of a continuous stirred tank reactor (CSTR) [Jargani2009]. Simulation results demonstrate that the proposed algorithm can track and forecast fault processes accurately. AUKF has been applied in [Fathabadi2009] for state estimation CSTR plants with different communication delays in their sensors. Also decentralized multi sensor fusion has been carried out to estimate states in presence of multi-rate sensors.

Apart from CSTR plants performance of adaptive estimators is also demonstrated for a continuous fermenter in simulation and benchmark heater mixer setup in real time experiments in [Bavdekar2011]. In [Odelson2006] the authors have worked with a chemical company to apply adaptive estimation methods to data from a gas-phase reactor. The authors have also demonstrated the effectiveness of the proposed estimator with the help of experiments on a laboratory chemical reactor.

A recent trend of using adaptive estimators for state of charge estimation (SOC) of batteries is also observed. Use of AEKF for state of charge estimation of lead acid batteries is reported in [Han2009]. In [Chen2012], ACKF demonstrably outperforms the dual extended Kalman filter during state of charge estimation of a battery where the ambient temperature and the load of the battery fluctuate. ACKF based SOC estimation algorithm reported in [Xia2015] for lithium-ion batteries in electric vehicles.

Although adaptive estimators are not very popular for fault detection a few recent publications demonstrate promising performance of these estimators during fault detection:

In [Hajiyev2011] AEKF applied for the parameter identification process of an Electro Mechanical Actuator. The performance of the proposed filter is tested for the different types of measurement faults; instantaneous abnormal measurements, continuous bias at measurements, measurement noise increment and fault of zero output. For fault detection and isolation of Lithium ion battery AEKF is also applied in [He2015].

For the fault-tolerant attitude estimation of the pico satellites is [Hajiyev2014] proposes use of AUKF algorithm, which performs correction for the process noise covariance (Q-adaptation) or the measurement noise covariance (R-adaptation) depending on the type of the fault. The author of [Soken2009] tested AUKF for two different measurement malfunction scenarios, instantaneous abnormal measurements and continuous bias at measurements. A paper by the same author [Soken2011] demonstrates Q-adaptative UKF for unexpected events in space environment during satellite attitude determination.

Adaptive unscented Kalman filter (AUKF)-based fault detection and isolation (FDI) scheme is proposed in [Das2015] for a spacecraft attitude determination (AD) system. It is demonstrated that the fault detection efficacy as well as fault discrimination performance of AUKF is noticeably better than non-adaptive filters. AUKF is also utilized for nonlinear process fault prognostics in [Cao2009] for CSTR plants.

2.4 Conclusion

The review of the works on nonlinear estimation and adaptive filtering helped the present worker to appreciate the development in this domain. Use of adaptive estimators for variants of real time applications motivated the present worker to pursue research on this particular category of estimators. Below are provided a few significant findings which helped the present worker to define the objective of this dissertation and contribute improved algorithms for nonlinear estimation:

- Investigation of nonlinear estimators revealed their efficacies as well as limitations. Insight developed from the critical review of all these estimators helped during selection of a suitable candidate as an underlying structure for adaptive nonlinear estimators.
- The usefulness of the information filter configuration of some of sigma point filters are also appreciated from the literature review. Economic computation of this configuration encouraged the present worker to develop their adaptive versions for sensor fusion.
- During the review of existing adaptive nonlinear filters, some drawbacks are noticed.
    - AUKF requires careful choice of tuning parameters failing which the performance of estimators deteriorates. ACKF although free from such tuning parameters cannot perform equally well with AUKF for some applications.
    - The $\boldsymbol{Q}$ and $\boldsymbol{R}$ adaptation algorithms reported for adaptive sigma point filters lack theoretical foundation in most of the cases. Some of the adaptation algorithms cannot ensure the positive definiteness of $\boldsymbol{Q}$ and $\boldsymbol{R}$.
    - The “knowledge gap” identified during the review of adaptive nonlinear estimators motivated the present worker to contribute improved algorithms for adaptive nonlinear estimators.
- In earlier works on adaptive filters for linear signal models [Maybeck1982] authors have cautioned that simultaneous adaptation of $\boldsymbol{Q}$ and $\boldsymbol{R}$ may lead to unacceptable estimation results as $\boldsymbol{Q}$ and $\boldsymbol{R}$ are negatively correlated and their effects are conflicting. This is also acknowledged by the recent workers [Karlgaard2010, MohanM2015]. Therefore, the present worker has refrained from implementing simultaneous $\boldsymbol{Q}$ and $\boldsymbol{R}$ adaptation algorithm for the adaptive nonlinear estimators.

Chapter 3: Test Problems

3.1 Chapter Introduction

This chapter presents the description of the estimation problems which will be considered as test problems to evaluate the proposed estimation algorithms in the subsequent chapters. The estimation problems include some standard tracking problems which have been considered by previous workers for the validation of the estimators proposed in their works. In addition to these realistic tracking problems, estimation of well known nonlinear systems (viz., Van der Pol's oscillator, Lorentz attractor) and some numerical estimation problems are also considered. Proposed algorithms are validated and their relative performance analysis has been carried out with the help of these estimation problems. In this chapter the following test problems have been elaborated.

- State estimation of a benchmark first order nonlinear system
- Bearing only tracking using a on board tracker (2^{nd} order)
- Parameter and state estimation of Van der Pol's oscillator (3^{rd} order)
- State estimation of a 3^{rd} order Lorentz attractor
- Tracking of a ballistic object during reentry (3^{rd} order)
- State estimation of a benchmark fourth order nonlinear system
- Tracking of a maneuvering aircraft (5^{th} order)

3.2 Description of test problems

3.2.1 State estimation of a first order nonlinear system

A single dimensional estimation problem is considered where system dynamics and the measurement equation suffer from severe nonlinearity. Estimation of the state of this system is challenging task. In many previous works this estimation problem has been considered [Ito2000, Sadhu2004, Bhaumik2013] as this problem can critically analyze the performance of the candidate estimator and readily expose its shortcomings if any.

The process dynamics and the measurement equation are taken from [Ito2000]. The system has two stable equilibrium points at $1, -1$ and another unstable equilibrium at 0. The equilibrium point at the origin is unstable as the derivative of state equation is positive at

$x = 0$. The measurement equation has a strong bi-modal tendency and cannot decisively distinguish between the two stable equilibrium points. The system dynamics and measurement equation are presented below

$$x_k = \phi(x_{k-1}) + \theta_k \tag{3.1}$$

where the function ϕ is given by

$$\phi(x) = x + 5\tau x(1 - x^2) \tag{3.2}$$

θ_k is an additive Gaussian noise, $\theta_k \sim N(0, b^2\tau)$

The measurement equation is given by

$$y_k = \gamma(x_k) + \upsilon_k \tag{3.3}$$

$$\gamma(x) = \tau(x - 0.05)^2 \tag{3.4}$$

υ_k is an additive measurement noise(Gaussian), $\upsilon_k \sim N(0, d^2\tau)$. The parameters used to generate the true state trajectory have the values as given below. $\tau = 0.01\,\text{sec}$, $x_0 = -0.2$, $b = 0.5$, $d = 0.1$. For the filter, the initial values are chosen as $\hat{x}_0 = 0.8$, $\hat{\boldsymbol{P}}_{\boldsymbol{0}} = 4$. Note that the measurement equation is taken from [Ito2000] and has strong bi modal tendency compared to the measurement equation considered in [Sadhu2004, Bhaumik2013].

To illustrate the bi modal tendency of the measurement equation we present the state vs measurement plot in Fig. 3.1. It is observed that same measurement is obtained for two possible values of the state. Therefore, the measurement looses its uniqueness. Moreover, the rate of change of measurements for a rate of change in state is minimum near the origin (i.e., at $x = 0$).

Fig. 2 shows two trajectories, starting from the same equilibrium point, with small process noise. We see in Fig. 3.2 that the trajectories settle at two different steady equilibrium values. However, the measurement values hardly changes (illustrated in Fig. 3.1) for these radically different state trajectories.

This non uniqueness of measurement troubles the estimator to track the true state trajectory satisfactorily. The estimate settles at the one equilibrium point while the true trajectory settles on the other. This phenomenon may be termed as track loss. It can be said that the estimate has lost the correct track when estimation error is more than 0.8 at 4 sec.

It is observed from the previous works that for estimators with accurate knowledge of noise covariance estimation performance may degrade as the estimator is susceptible to numerous occurrence of track loss. In situations when the noise covariances are unknown and assumed with an arbitrary value it is needless to say that estimation performance would be degraded. Therefore, this case study may be an appropriate one for prima-facie validation of the proposed adaptive filters in face of unknown noise covariance.

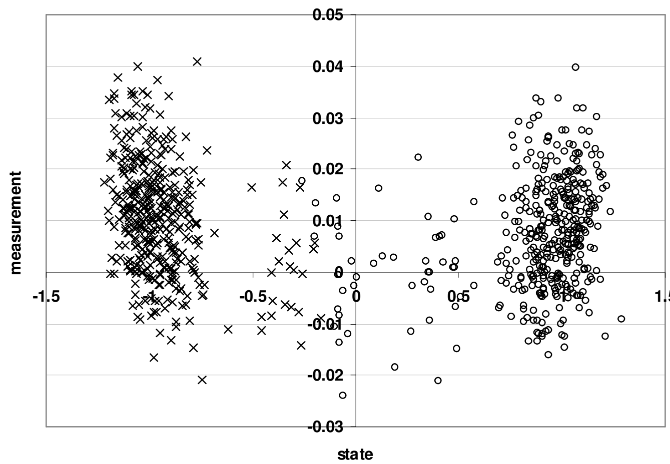


Fig. 3.1: Plot of measurement with respect to state

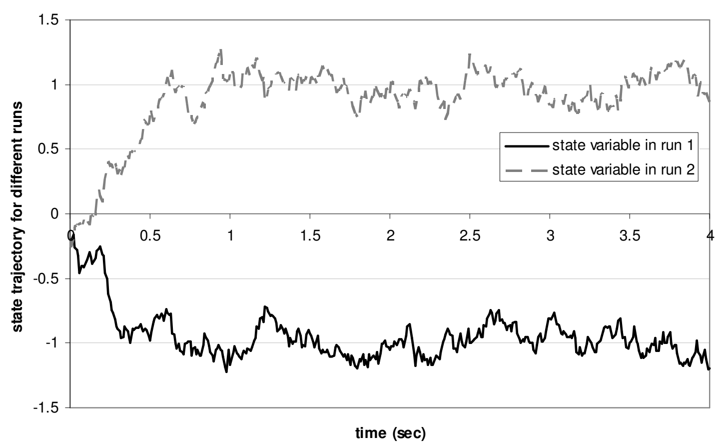


Fig. 3.2: State trajectories for two different representative runs

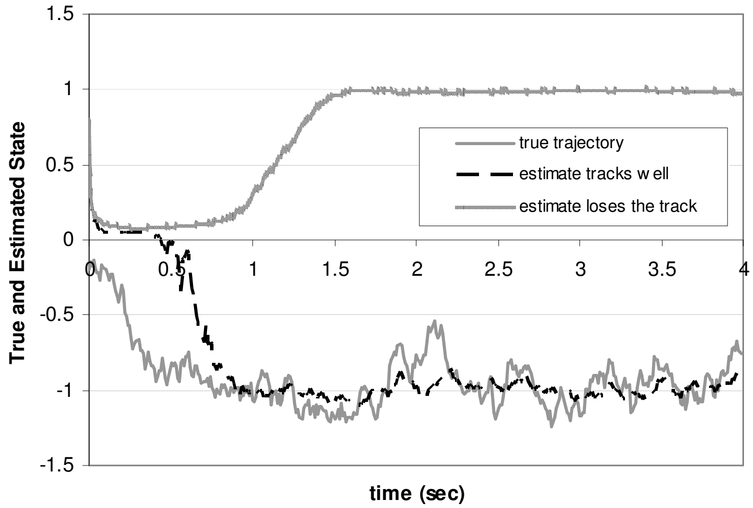


Fig. 3.3: True and estimated state for two different representative runs

3.2.2 Bearing only tracking problem

Bearing only tracking (BOT) problem has been considered from [Sadhu2006, Sadhu2004] where the system dynamics is linear but the measurement equation is nonlinear function of states and measurement noise (i.e., noise is non additive in nature). The target is moving on ground with a constant velocity (position and velocity are assumed to be perturbed by Gaussian noise). The target is assumed to move in a straight line (assumed to be along the positive direction of x-axis) in the horizontal plane. In BOT problem the target is tracked using an on board sensor. The tracking is carried out using angle of depression (bearing measurements) from the airborne platform (along with the sensor) moving parallel to the target in the same direction with constant velocity. A schematic diagram is provided by Fig. 3.4 for illustration. The platform is moving at a nearly fixed altitude in the same vertical plane. The kinematic equations of target as well as the platform are presented below. The bearing measurement for the target is noisy and measurement is nonlinear function of states and measurements as the platform motion noise also appears in the measurement equation.

Target motion is considered here as the process model and given by

$$\boldsymbol{x}_k = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix} \boldsymbol{x}_{k-1} + \begin{bmatrix} \frac{T^2}{2} \\ T \end{bmatrix} \boldsymbol{\omega}_k \tag{3.5}$$

where, $\boldsymbol{x_k} = [x_k^1 \quad x_k^2]^T$, x_k^1 is target position along x axis and x_k^2 is target velocity which is assumed to be constant. The initial choice of state vector is $\boldsymbol{x}_0 = [80m \quad 1ms^{-1}]^T$ The sampling time is considered as T=1 sec and $\boldsymbol{\omega}_k$ is process noise with covariance Q=0.01m^2/sec^4. Values of these parameters are taken from [Sadhu2006].

The line of sight of the target from the onboard sensor is obtained from the bearing angle measurement. The platform motion of sensor (along x and y axis) influence the bearing angle measurement. In this context the measurement equations are presented as given below:

$$y_k^1 = y_k^p = 20 + \upsilon_k^1 \tag{3.6}$$

$$y_k^2 = x_k^p = 4kT + \upsilon_k^2 \tag{3.7}$$

$$y_k^3 = \tan^{-1}\left(\frac{y_k^p}{x_k^1 - x_k^p}\right) + \upsilon_k^3 \tag{3.8}$$

or, $$y_k^3 = \tan^{-1}\left(\frac{20 + \upsilon_k^1}{x_k^1 - 4kT - \upsilon_k^2}\right) + \upsilon_k^3 \tag{3.9}$$

First two elements of measurement vector $\boldsymbol{y_k} = [y_k^1 \quad y_k^2 \quad y_k^3]^T$ are the platform positions along y axis and x axis respectively. 'k' is the current time instant. The measurement equation presented by (3.9) indicates that the third element of measurement vector is a nonlinear function of state as well as the measurement noises υ_k^1 and υ_k^2. However, υ_k^3 is the additive measurement noise. Measurement noise vector therefore may be formed as $\boldsymbol{\upsilon_k} = [\upsilon_k^1 \quad \upsilon_k^2 \quad \upsilon_k^3]^T$ with true covariance $\boldsymbol{R_k}$. Truth value of $\boldsymbol{R_k}$ is given by:

$$\boldsymbol{R_k} = \begin{bmatrix} \sigma_{v1}^2 & 0 & 0 \\ 0 & \sigma_{v2}^2 & 0 \\ 0 & 0 & \sigma_{v3}^2 \end{bmatrix}$$

σ_{v1}, σ_{v2} and σ_{v3} are the standard deviation of three measurement noises with values of 1 meter for the first two diagonal elements and 3° for the third respectively. Fig. 3.5 and Fig. 3.6 present the position and velocity of the target respectively for a representative run.

The tracking filters are usually initialized from first few measurements. The current bearing measurement defines the initial position estimate and the difference of two bearing measurements provides the estimation of initial velocity. The way authors of [Sadhu2006] has initialized the filter has been followed in this work.

The initial choice of position estimate and its covariance are considered from [Sadhu2006] as given below:

$$\hat{x}_0^1 = 4kT + \frac{20}{\tan y_0^3} \tag{3.10}$$

y_0^3 is the first measurement.

$$P_{11,0} = r_x + \frac{r_y}{\tan^2 y} + \frac{\bar{y}_p^{\,2}}{\sin^4 y} r_s \tag{3.11}$$

Where $y = y_0^3 - \upsilon_0^3$

As per [Sadhu2006] the initial velocity estimation is selected as $\hat{x}_0^2 = 0$ and associated variance as $P_{22,0} = 1$. The off diagonal terms $P_{12,0}$ and $P_{21,0}$ are taken as zero.

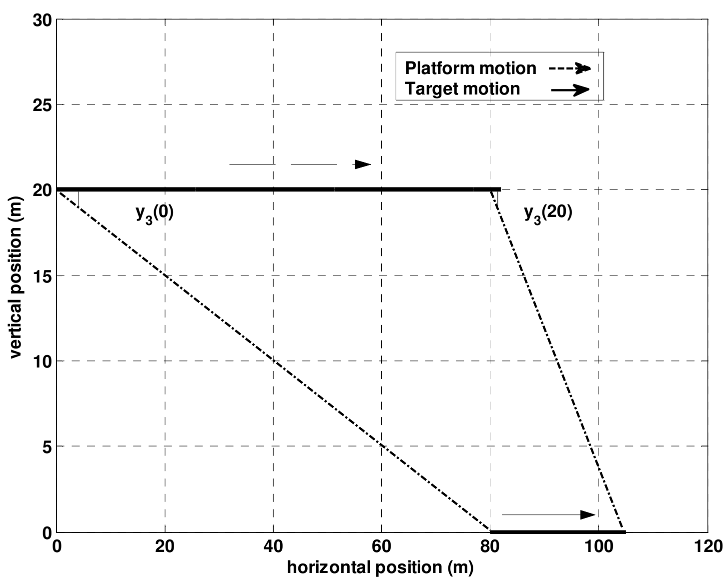


Fig.3.4: Illustration of BOT problem with a schematic diagram

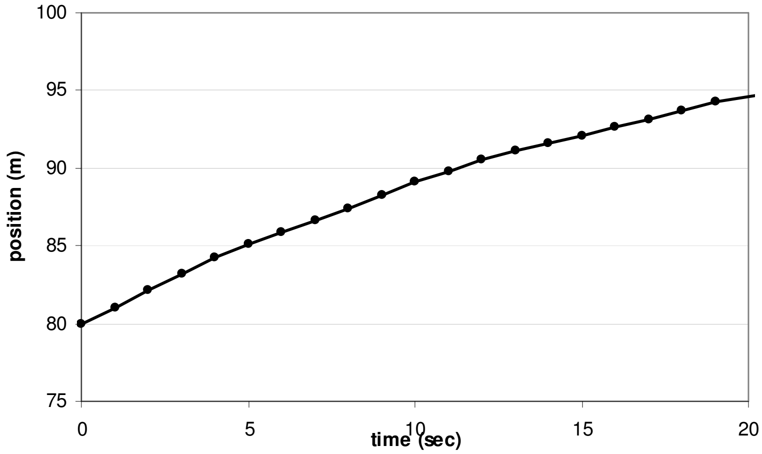


Fig. 3.5: Plot of position of the target for a representative run

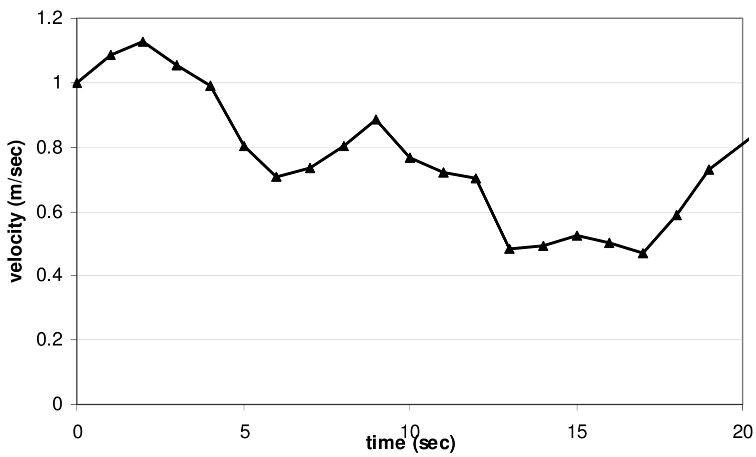


Fig. 3.6: Plot of velocity of the target for a representative run

3.2.3 Parameter and state estimation of Van der Pol's oscillator

The Van der Pol's oscillator is a noteworthy nonlinear oscillator and has been used by many workers for demonstration of performance of nonlinear estimators [Kandepu2008, Besançon2010]. Estimation of the friction coefficient along with the states of the oscillator is considered here as a test problem.

The dynamic equation of the Van der Pol oscillator which has been presented below exhibits a stable Limit Cycle oscillation. The oscillator irrespective of the initial condition always reaches the Limit Cycle and demonstrates sustained oscillations. The differential equation is presented as

$$\dot{x}_1 = x_2 \qquad (3.12)$$

$$\dot{x}_2 = \mu\left(1 - x_1^2\right)x_2 - x_1 \quad (3.13)$$

$$\dot{\mu} = 0 \quad (3.14)$$

μ represents a constant friction coefficient. The states of the oscillator are denoted by x_1 and x_2 respectively. Estimation of both friction coefficient and the states has been considered for this case study. The corresponding discrete state space model of the oscillator is obtained from Euler's approximation with a sampling time τ. The parameter (friction coefficient) is assumed to be unknown (which may also be time varying in some situation) for the estimators. For joint estimation the parameter is modeled as a state and augmented with the state vector. The system states and friction coefficient are corrupted by additive Gaussian noise with zero mean.

The discrete time model is given by:

$$x_k = f(x_{k-1}) + w_k \quad (3.15)$$

$f(x_{k-1})$ indicates the discrete nonlinear model of the oscillator.

$$f(x_{k-1}) = \phi x_{k-1} + G[D(x_{k-1})] \quad (3.16)$$

With a matrix $\phi = \begin{bmatrix} 1 & \tau & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$ with $x_{k-1} = \left[x_{1k-1} \quad x_{2k-1} \quad \mu_{k-1}\right]^T$

$G = \left[0 \quad \tau \quad 0\right]^T$, $D(x_{k-1})$ is defined by

$$D(x_{k-1}) = \left(x_{k-1}^T e_3\right)\left[1 - (x_{k-1}^T e_1)^2\right](x_{k-1}^T e_2) - (x_{k-1}^T e_1) \quad (3.17)$$

where e_i denotes the i^{th} unit vector for i=1,2,3.

w_k indicates an additive process noise which is independent of measurement noise. The state x_{1k-1} is directly available as a measurement and perturbed by a zero mean Gaussian noise v_k.

The necessary parameters for simulation are provided below in table 3.1

Table-3.1: Parameters for state estimation of Van der Pol's oscillator

Symbols	Value	Description
x_0	$[1.4 \quad 0 \quad 0.2]^T$	Initial value for true trajectories
$\hat{x}_0$	$[0 \quad 5 \quad 4]^T$	Initialization of filter estimates.
$\hat{P}(0)$	*diag* ([1.4, 5, 4])	Initial a posteriori error covariance
Q_k	*diag* ([10^{-3}, 10^{-3}, 10^{-5}])	True process noise covariance
R	10^{-3}	True measurement noise covariance
N	150	Window length for adaptation
τ	0.1 sec	Sampling time

For the time varying friction coefficient the nature of variation is assumed follow the equation $\mu_k = 0.5\sin(\omega\tau\, k) + 0.5$ to generate the true state trajectories. The window size is considered to be 30 time instants during the estimation of the time varying case.

In Fig. 3.7, Fig. 3.8 and Fig, 3.9 we present the state trajectories and the phase plane plot for a representative run. Plots are presented for two different cases where the friction coefficient is constant and for the other case when it is time varying.

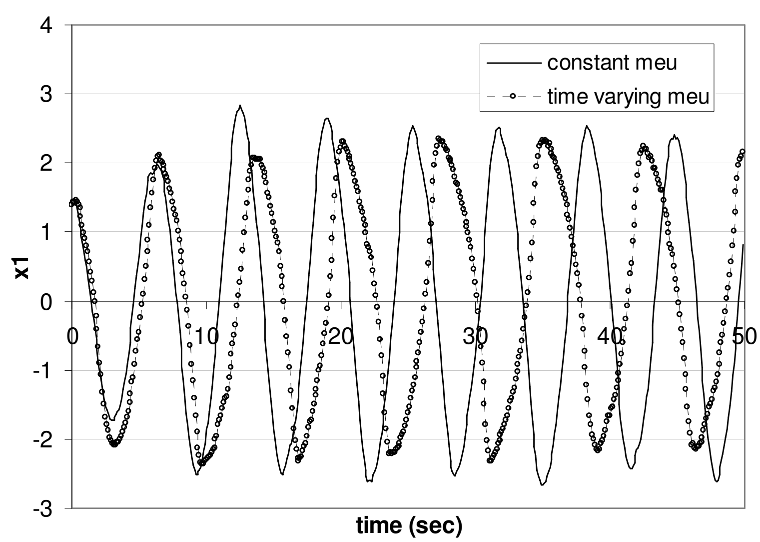


Fig. 3.7: Plot of state x_1 for a representative run

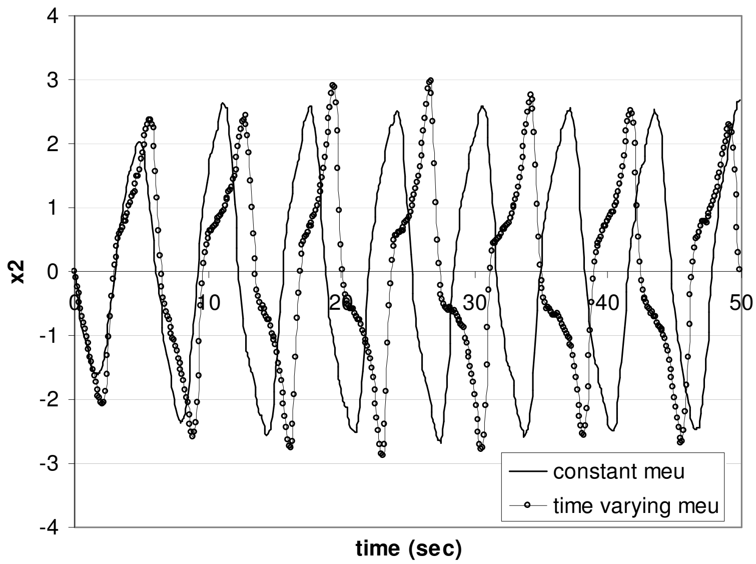


Fig. 3.8: Plot of state x_2 for a representative run

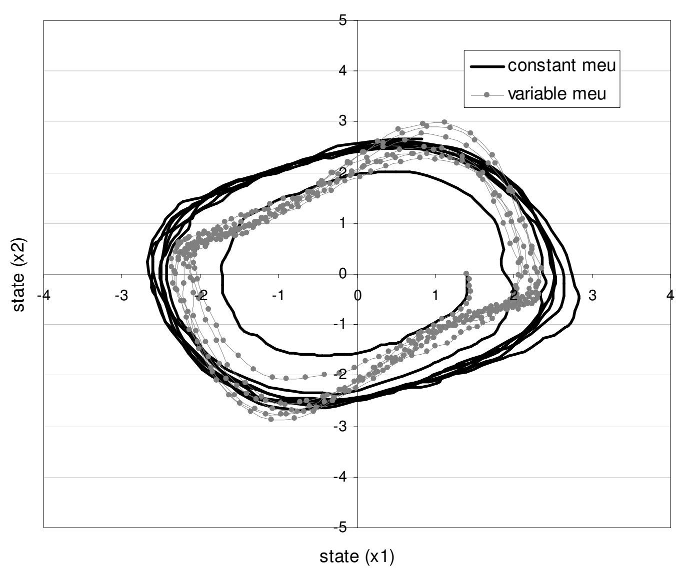


Fig. 3.9: Phase plane plot for a representative run

3.2.4 State estimation of a Lorentz attractor

In this case study we have considered the Lorentz attractor named after meteorological researcher Lorentz. The dynamics of the Lorentz attractor is significantly nonlinear which makes the system an appropriate one to evaluate the performance of nonlinear estimators. In many previous publications [Bhaumik2013, Ito2000] on nonlinear estimators this system has been considered as a case study for performance assessment. We present the discrete time three dimensional Lorentz attractor as presented in [Bhaumik2013].

The system dynamics in discrete time is presented as

$$\boldsymbol{x_k} = \boldsymbol{x_{k-1}} + \tau\left[\alpha(x_2 - x_1) \quad \beta x_1 - x_2 - x_1 x_3 \quad x_1 x_2 - \gamma x_3\right]^T + \boldsymbol{b} w_k \tag{3.18}$$

where the state vector of system $\boldsymbol{x_k} = \left[x_1 \quad x_2 \quad x_3\right]^T$. The measurement for this attractor is a scalar and the measurement equation is

$$y_k = \tau\sqrt{x_1^2 + x_2^2 + x_3^2} + d v_k \tag{3.19}$$

Both the process and the measurement equation are perturbed with Gaussian (white) noise and follows the distribution $N(0,\tau)$.

The system has three unstable equilibrium points as given below:

$\left[0 \quad 0 \quad 0\right]^T$, $\left[\sqrt{\gamma(\beta-1)} \quad \sqrt{\gamma(\beta-1)} \quad (\beta-1)\right]^T$ and $\left[-\sqrt{\gamma(\beta-1)} \quad -\sqrt{\gamma(\beta-1)} \quad (\beta-1)\right]^T$ while $\alpha \neq 0$ and $\sqrt{\gamma(\beta-1)} \geq 0$. The chosen values of these classical parameters along with other necessary parameters are given in the table 3.2 below.

Table-3.2: Parameters for state estimation of Lorenz attractor

Symbols	Values	Description
α	10	Prandtl number
β	28	Rayleigh number
γ	8/3	Parameter related to system dynamics
$\boldsymbol{b}$	$\left[0 \quad 0 \quad 0.5\right]^T$	Input matrix for process noise
d	0.065	Scaling factor for measurement noise
τ	0.01sec	Sampling time
$\boldsymbol{x_0}$	$\left[-0.2 \quad -0.3 \quad -0.5\right]^T$	Initial value for true trajectories
$\boldsymbol{\hat{P}_0}$	0.35**diag* ([1, 1, 1])	Initial updated error covariance
$\boldsymbol{\hat{x}_0}$	A Gaussian prior with mean $\boldsymbol{\bar{x}_0} = \left[1.35 \quad -3 \quad 6\right]^T$, covariance $\boldsymbol{\hat{P}_0}$	Initial choice of filter estimates.

In Fig. 3.10, Fig. 3.11 and Fig. 3.12 below the state trajectories of the attractor are presented for a representative run and the phase plane plot is also given in Fig. 3.13.

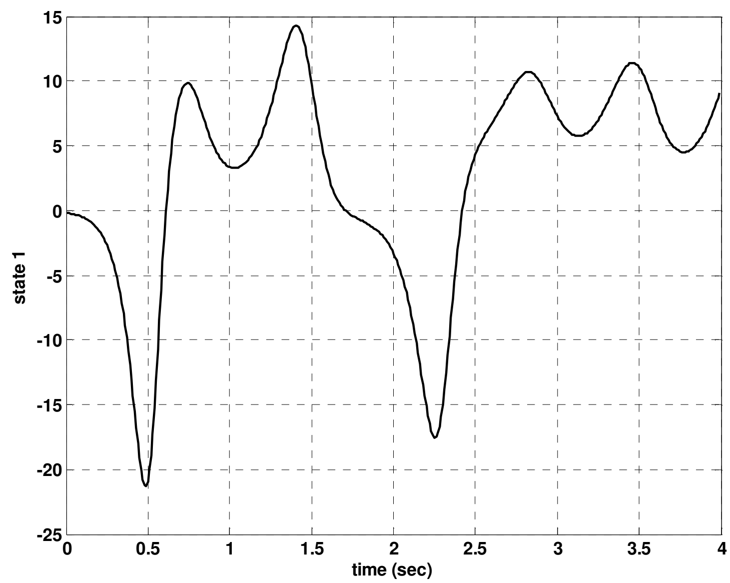


Fig. 3.10: Plot of state x_1 for a representative run

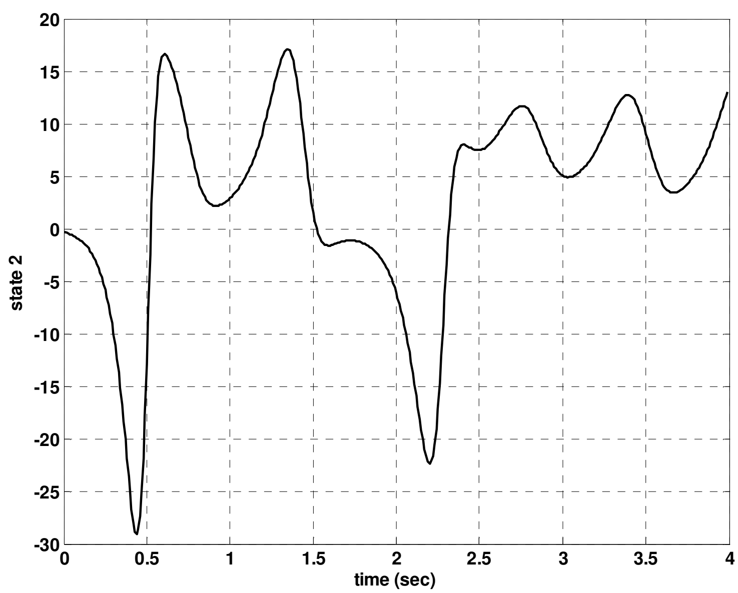


Fig. 3.11: Plot of state x_2 for a representative run

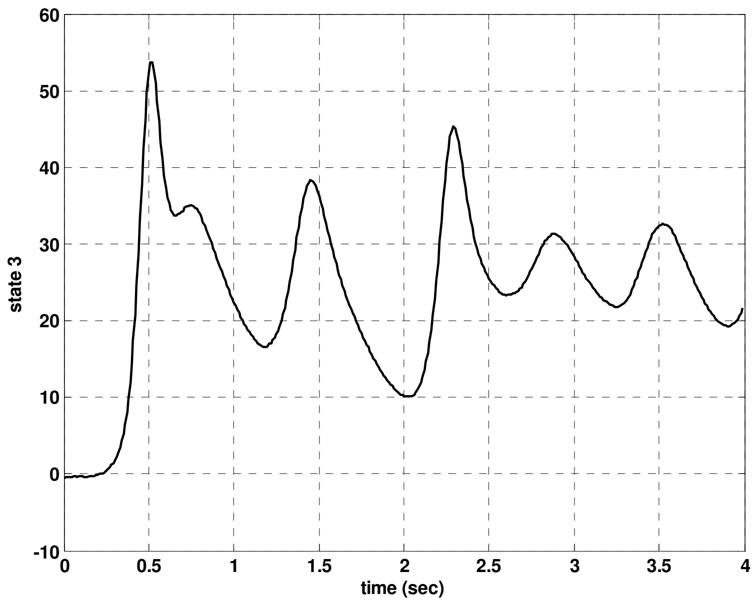


Fig. 3.12: Plot of state x_3 for a representative run

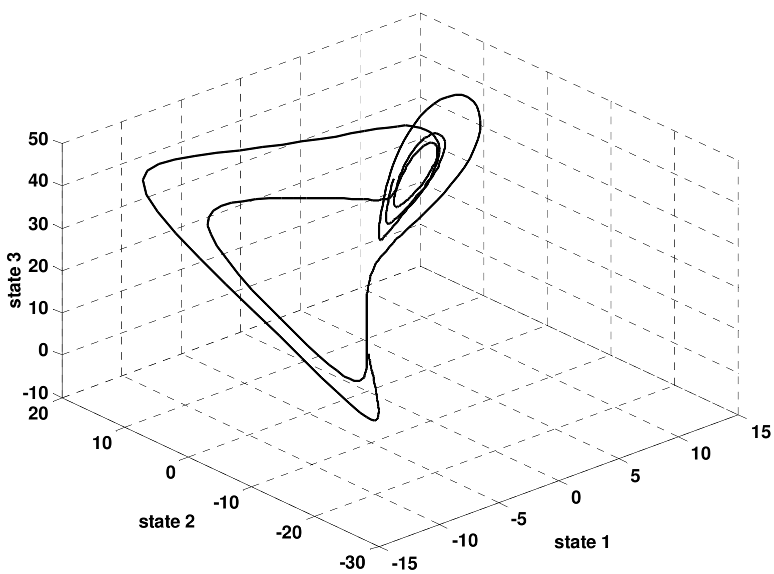


Fig. 3.13: Phase plane plot for a representative run

3.2.5 Object Tracking Problem

For performance evaluation of proposed filters a standard ballistic object tracking problem has been considered as a case study. Object tracking problems are well known tracking problems and has significant nonlinearity in the system dynamics when the object enters atmosphere. It is an important application area where different estimators can be employed so as to estimate the altitude, velocity of the object and ballistic coefficient/ballistic parameter from radar signals. It would be interesting to assess the estimation performance of the proposed estimators including one of such object tracking problems as a case study. Performance evaluation of estimators using this specific problem has been carried out in many previous works [Athans1968, Norgaard2000, Wu2006, Arasaratnam2011]. During initial phase when the object is in exo-atmospheric zone nonlinearity is not pronounced and the dynamics may be considered to be quasi linear. During endo-atmospheric phase the dynamics becomes extremely nonlinear as the object enters atmosphere and experiences drag.

The object is considered to be falling vertically in a single dimension. The tracking radar is assumed to provide the range of the tracked object as illustrated in Fig 3.14.

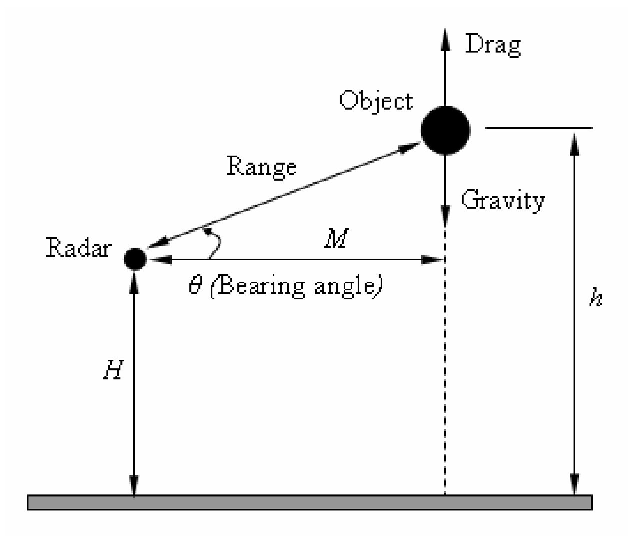


Fig 3.14: Radar tracking of a ballistic object during reentry: A schematic diagram

3.2.5.1. Dynamic Model I

The dynamic model for the object during reentry has been presented here in single dimension as presented in [Arasaratnam2011].

$$\dot{\boldsymbol{x}} = \begin{bmatrix} -x_2 & -\exp(-\gamma x_1) x_2^2 x_3 + g & 0 \end{bmatrix}^T \tag{3.20}$$

Where state vector $\boldsymbol{x} = [x_1 \quad x_2 \quad x_3]^T$; represents altitude, velocity and the ballistic parameter respectively. g is the acceleration due to gravity. The air density decays exponentially with height with the time constant $\gamma = 1.49 \times 10^{-4}$ m^{-1} [Arasaratnam2011].

The parameter to be estimated is called ballistic parameter instead of the term ballistic coefficient as used in [Arasaratnam2011]. This may avoid confusion with the standard definition of the term as provided in [Ristic2003].

The corresponding discretized model using Euler's method with a sampling time τ is given by:

$$\boldsymbol{x_k} = \boldsymbol{x_{k-1}} + \tau\left[-\boldsymbol{x}_{\boldsymbol{k-1}}^{\boldsymbol{T}}\boldsymbol{e_2} \quad -\exp(-\gamma \boldsymbol{x}_{\boldsymbol{k-1}}^{\boldsymbol{T}}\boldsymbol{e_1})(\boldsymbol{x}_{\boldsymbol{k-1}}^{\boldsymbol{T}}\boldsymbol{e_2})^2\left(\boldsymbol{x}_{\boldsymbol{k-1}}^{\boldsymbol{T}}\boldsymbol{e_3}\right) + g \quad 0\right] + \boldsymbol{w_k} \tag{3.21}$$

$\boldsymbol{x_{k-1}} = [x_{1k-1} \quad x_{2k-1} \quad x_{3k-1}]^T$ and $\boldsymbol{e_i}$ denotes the i^{th} unit vector.

$\boldsymbol{w_k}$ indicates an additive process noise which is independent of measurement noise $\boldsymbol{v_k}$. Noises are assumed to be zero mean, white (Gaussian).

The covariance of the process noise is given by $\boldsymbol{Q} = \begin{bmatrix} q_1\tau^3/3 & q_1\tau^2/2 & 0 \\ q_1\tau^2/2 & q_1\tau & 0 \\ 0 & 0 & q_2\tau \end{bmatrix}$ where q_1 and q_2 are parameters for describing the process noise [Ristic2003].

The range of the object from radar is obtained measured in a spherical reference frame and given by the measurement equation

$$y_k = \sqrt{M^2 + (\boldsymbol{x}_{\boldsymbol{k}}^{\boldsymbol{T}}\boldsymbol{e_1} - H)^2} + v_k \tag{3.22}$$

Where H denotes the altitude of radar and M is the shortest horizontal distance of radar form the flight path of the object as given in Fig. 3.14 and $\boldsymbol{e_1} = [1 \quad 0 \quad 0]^T$.

Necessary parameters for simulation are given in the table 3.3 below.

Table-3.3: Parameters for object tracking problem (SI units)

Symbols	Values	Description
$\boldsymbol{x_0}$	$[61000\text{ m} \quad 3048\text{ ms}^{-1} \quad 4.49\times10^{-4}\text{ m}^{-1}]^T$	Initial value for true trajectories
q_1	$1\text{ m}^2\text{s}^{-3}$	A parameter of true Q
q_2	$10^{-12}\text{m}^{-2}\text{s}^{-1}$	A parameter of true Q
M	10000 m	Distance of object from radar
H	1000 m	Height of the radar from ground
R	30^2 m^2	Measurement error covariance
$\hat{\boldsymbol{P}}_0$	*diag* ([10^6, 10^4, 10^{-4}])	Initial updated error covariance
$\hat{\boldsymbol{x}}_0$	A Gaussian prior with mean $\bar{\boldsymbol{x}}_0 = [62000 \quad 3400 \quad 10^{-5}]^T$, covariance $\hat{\boldsymbol{P}}_0$	Initialization of filter estimates.
$N_{\min}$	3	Initial choice of window length
N	60	Actual window length
τ	0.1sec	Sampling time

For a representative run we present the altitude and velocity of the reentry object in Fig. 3.15 and Fig. 3.16.

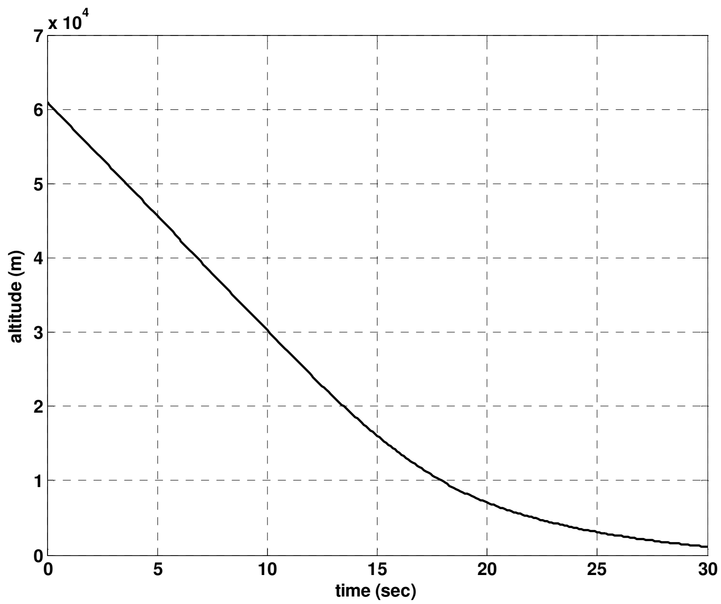


Fig. 3.15: Plot of altitude of the object for a representative run

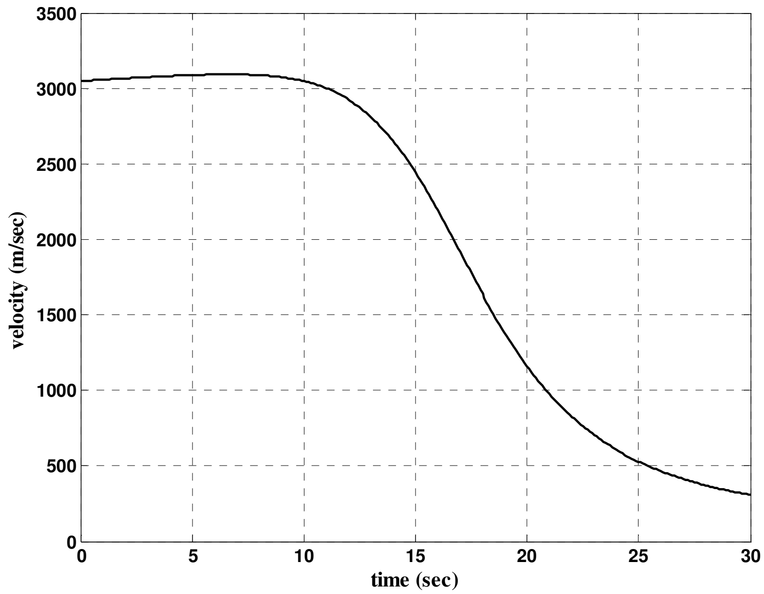


Fig. 3.16: Plot of velocity of the object for a representative run

3.2.5.2. Dynamic Model II

The Dynamic model of the object is presented in this section above is conceptually same with what discussed in the subsection above. Only an approximation is made in the system dynamics. In [Athans1968] for this object tracking problem it is assumed that the effect of gravity is negligible compared to drag force and omitted from the dynamics. This has also been followed in the succeeding works [Norgaard2000, Wu2006]. The measurements and necessary parameters for simulation are provided in FPS system unlike [Arasaratnam2011] where the parameters are presented in SI system.

The dynamic model is given by

$$\dot{h} = -V \tag{3.23}$$

$$\dot{V} = -\frac{C_D A \rho(h) V^2}{2m} \tag{3.24}$$

Note that in (3.24) gravity has been ignored.

The symbols used are defined as given below:

h :height of the object (ft), V : object velocity (ft/sec), C_D : drag coefficient (dimensionless), A : reference area for drag evaluation (sq. ft), ρ :air density(slugh/ft^3), m : mass of object(slugh)

Air density decays exponentially with height following the relation $\rho(h) = \rho_0 e^{-\gamma h}$ with $\gamma = 5\times10^{-5}$ ft^{-1}. We define $\xi \stackrel{\Delta}{=} \frac{C_D A \rho_0}{2m}$, a ballistic parameter as mentioned in [Athan1968]. However, the usual definition of ballistic coefficient given in [Ristic2003] as $\beta = \frac{mg}{C_D A}$ and related with the ballistic parameter as $\xi = \frac{\rho_0 g}{2\beta}$.

For estimation of ballistic parameter, it is augmented with state vector and assumed to be a constant. The differential equation of object dynamics is modified as given below:

$$\dot{h} = -V \tag{3.25}$$

$$\dot{V} = -e^{-\gamma h} V^2 \xi \tag{3.26}$$

$$\dot{\xi} = 0 \tag{3.27}$$

The corresponding discrete state space model of object dynamics is obtained from Euler's approximation as presented in (3.21) for Model I.

In some case studies it is also assumed that measurements are available from multiple radars. The radars are positioned at different locations of the atmosphere. The range only measurements are obtained from them which are a nonlinear functions of the system states. The interval of measurement is same as sampling interval, i.e., τ sec as discussed in previous model. The mathematical expression of the measurements obtained from the radars is presented as

$$y_k^s = r_k^s + v_k^s = \sqrt{\boldsymbol{M}_s^2 + \left(\boldsymbol{x}_k^T \boldsymbol{e}_1 - \boldsymbol{H}_s\right)^2} + v_k^s \tag{3.28}$$

for $s = 1,2,3$ where s indicate the radar at s^{th} position.

Here, $e_1 = [1 \quad 0 \quad 0]^T$ represents an unit vector.

H_s denotes the altitude of radar and M_s is the shortest horizontal distance from the flight path of the object during reentry. v_k^s indicates zero mean random noise with covariance R_s.

The parameters necessary for simulations are presented in table 3.4 below:

Table-3.4: Parameters for object tracking problem (FPS units)

Symbols	Values	Description
$\boldsymbol{x_0}$	$\begin{bmatrix}300000 & 20000 & 10^{-3}\end{bmatrix}^T$	Initial value for true trajectories
q_1	10^{-5} ft^2s^{-3}	A parameter of true Q
q_2	$10^{-9}ft^{-2}s^{-1}$	A parameter of true Q
M_1	100000 ft	Distance of object from radar 1
H_1	100000 ft	Height of the radar 1from ground
M_2	109540 ft	Distance of object from radar 2
H_2	100000 ft	Height of the radar 2from ground
M_3	89443 ft	Distance of object from radar 3
H_3	110000 ft	Height of the radar 3from ground
R_1	250^2	noise covariance for radar 1
R_2	100^2	noise covariance for radar 2
R_3	70^2 ft^2	noise covariance for radar 3
$\boldsymbol{\hat{P}_0}$	*diag* ([10^6, 4×10^6, 10^{-4}])	Initial a posteriori error covariance
$\boldsymbol{\hat{x}_0}$	Normal Random vector with mean $\boldsymbol{x_0}$, covariance $\boldsymbol{\hat{P}_0}$	Initialization of filter estimates.
$N_{\min}$	10	Initial choice of window length
N	100	Actual window length

3.2.6 State estimation of a fourth order nonlinear system

State estimation of a nonlinear system has been considered from the work of [Singh2015] with the dynamic equation and measurement equation as given below:

$$\boldsymbol{x_k} = 20\cos(\boldsymbol{x_{k-1}}) + \boldsymbol{w_k} \tag{3.29}$$

$$y_k = \sqrt{1 + \boldsymbol{x_k^T x_k}} + v_k \tag{3.30}$$

Here, $\boldsymbol{x_k}$ is the state vector , $\boldsymbol{x_k} \in \Re^4$. $\boldsymbol{w_k}$ is the Gaussian (white) noise, and $\boldsymbol{w_k} \in \Re^4$, $\boldsymbol{w_k} \sim N(\boldsymbol{0}_{4\times1}, \boldsymbol{I}_{4\times4})$.

y_k is a scalar measurement perturbed with measurement noise, v_k (Gaussian white noise) and $v_k \sim N(0,1)$

The true state trajectories are generated with the initial choice of state vector as $\boldsymbol{x_0} = 0.1 * [1 \quad 1 \quad 1 \quad 1]^T$. The states filters are initialized with a Gaussian prior as $\hat{\boldsymbol{x}}_{\boldsymbol{0}} \sim N(\boldsymbol{0}_{4\times1}, \boldsymbol{I}_{4\times4})$ with an initial error covariance $\hat{\boldsymbol{P}}_{\boldsymbol{0}} = \boldsymbol{I}_{4\times4}$

3.2.7 Aircraft Tracking Problem

The performance of proposed estimators has also been evaluated using a tracking problem where a maneuvering aircraft has to be tracked. The aircraft which is executing a maneuvering turn with unknown time varying turn rate has been considered to be tracked by multiple radars. This problem first appears in [BarShalom2001] and considered in many works [Arasaratnam2009, Jia2013a, Jia 2013b] for demonstration with different measurement equations. As the turn rate of the aircraft is considered to be unknown and time varying the kinematic model of the system becomes significantly nonlinear. Practically an aircraft while maneuvering with such unknown and time varying turn rate may escape radar stations and consequently the estimators may lose the track of the aircraft as would be explained in the following subsections.

3.2.7.1. Kinematic Model

The kinematic equation of the motion of the aircraft is presented below. The turn rate of the aircraft being unknown it is modelled as a state and augmented with the state vector of the kinematic model. This model appears in [BarShalom2001, Arasaratnam2009, Jia2013a, Jia2013b]

$$\xi_k = \begin{bmatrix} 1 & \frac{\sin(\omega_{k-1}\tau)}{\omega_{k-1}} & 0 & \frac{\cos(\omega_{k-1}\tau)-1}{\omega_{k-1}} & 0 \\ 0 & \cos(\omega_{k-1}\tau) & 0 & -\sin(\omega_{k-1}\tau) & 0 \\ 0 & \frac{1-\cos(\omega_{k-1}\tau)}{\omega_{k-1}} & 1 & \frac{\sin(\omega_{k-1}\tau)}{\omega_{k-1}} & 0 \\ 0 & \sin(\omega_{k-1}\tau) & 0 & \cos(\omega_{k-1}\tau) & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \xi_{k-1} + w_k \tag{3.31}$$

Here the state vector is $\xi_k = [p_{x_k} \quad v_{x_k} \quad p_{y_k} \quad v_{y_k} \quad \omega_k]^T$; p_{x_k} and p_{y_k} are the position in x and y coordinate respectively; v_{x_k} and v_{y_k} are the corresponding velocities at the instant k. ω_k is he unknown time varying turn rate. τ indicates the time interval between two consecutive measurements. w_k is zero mean Gaussian noise (white) which indicates the modelling uncertainty. The process noise for this noise sequence is considered as

$$Q_k = \begin{bmatrix} \frac{0.1\tau^3}{3} & \frac{0.1\tau^2}{2} & 0 & 0 & 0 \\ \frac{0.1\tau^2}{2} & 0.1\tau & 0 & 0 & 0 \\ 0 & 0 & \frac{0.1\tau^3}{3} & \frac{0.1\tau^2}{2} & 0 \\ 0 & 0 & \frac{0.1\tau^2}{2} & 0.1\tau & 0 \\ 0 & 0 & 0 & 0 & q\tau \end{bmatrix} \quad (3.32)$$

Note, that the element $Q_k(5,5)$ is the noise covariance of respective augmented parameter, i.e., turn rate of the aircraft. To generate the true state trajectories for this case study $Q_k(5,5)$ is selected as q_{true}.

3.2.7.2. Measurement Model

The bearing only tracking of aircraft as explained in [Jia2013b] is considered in this dissertation as a test problem.

The trajectory of the aircraft is tracked by the fusion of the bearing angle measurements from two tracking radars which are positioned in different locations of the atmosphere. The measurement equations can be represented as

$$\theta_k^{\zeta} = \tan^{-1}\left(\frac{p_{y_k} - p_{y_{ref}}^{\zeta}}{p_{x_k} - p_{x_{ref}}^{\zeta}}\right) + v_k^{\zeta} \qquad \zeta = 1,2 \quad (3.33)$$

ζ indicates position of the ζ^{th} radar. $p_{y_{ref}}^1 = -10^4 m;\ p_{x_{ref}}^1 = -10^4 m;\ p_{y_{ref}}^2 = 10^4 m;\ p_{x_{ref}}^2 = 10^4 m$. The zero mean measurement noise (Gaussian) sequences have covariances $R_1 = \left(\sqrt{30} mrad\right)^2$ and $R_2 = \left(\sqrt{40} mrad\right)^2$. The interval between two successive measurements is, $\tau = 1\sec$.

3.2.7.3. Simulation procedure

The proposed filtering algorithms are validated with the help of Monte Carlo simulation with 10000 runs. For generation of true state trajectories an initial choice of state is made as $x_0 = [1000\text{m} \quad 300\text{ms}^{-1} \quad 1000\text{m} \quad 0\text{ms}^{-1} \quad -0.05235\,\text{rad s}^{-1}]^T$. The unknown element of $\boldsymbol{Q}$ is chosen as $q_{true} = \left(1.323\times10^{-2}\,\text{rad s}^{-1}\right)^2$ to generate the true trajectories. The filters are initialized with a Gaussian prior with mean x_0 and $\hat{P}_0$, where $\hat{P}_0 = diag([100 \quad 10 \quad 100 \quad 10 \quad 10^{-4}])$.

Further investigation revealed the fact that such aircraft tracking problems are susceptible to track losses because of the non unique solution of the measurement equations. The possible set of true trajectories of aircraft has been provided for illustration in Fig. 3.17 which indicates the randomness of time varying turn rate as presented in the system equation. The values of q_{true} are high enough to induce random variations in turn rate. At some time instants the trajectory of the aircraft become such that the difference between the bearing angle from two different radars may either be negligibly small or become closer to π. Practically, at this moment the line of sight of two radars does not intersect each other and the aircraft can not be precisely located in the atmosphere. As a consequence of such non unique measurements the estimators fail to estimate the trajectory of the aircraft and track loss occurs. Fig. 3.18 has been presented for illustration where track loss has occurred for a non adaptive filter in the ideal situation with complete knowledge of noise covariances.

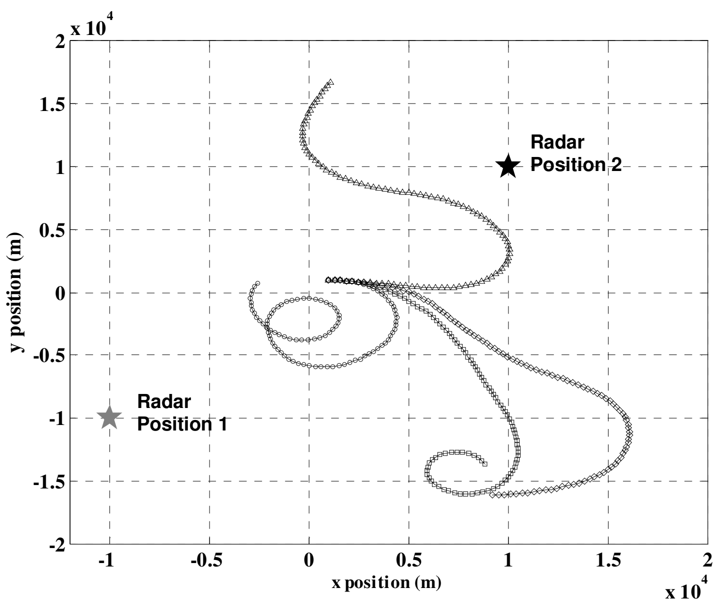


Fig. 3.17: Plot of trajectories of aircraft for different runs

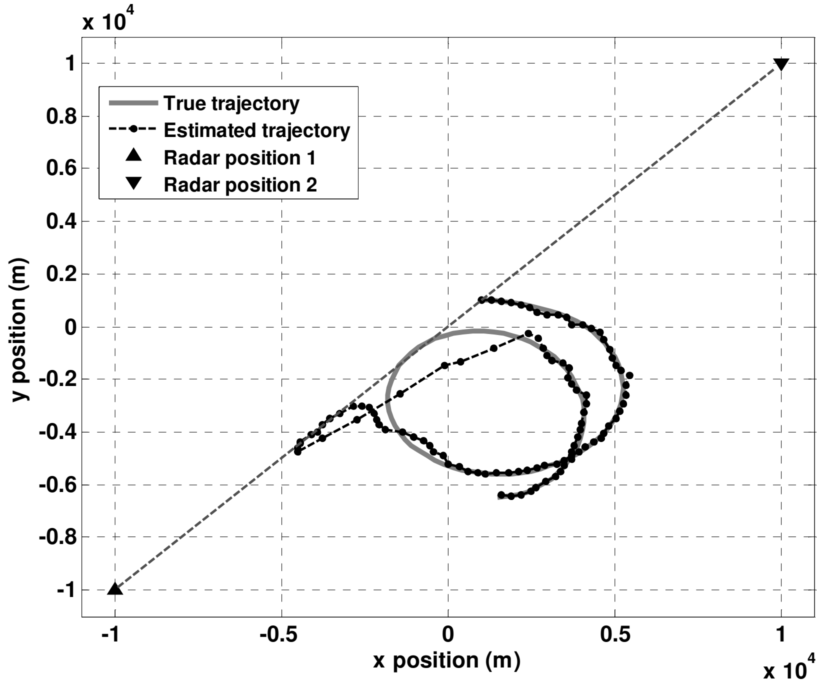


Fig. 3.18: A representative run where track loss has occurred

During the performance comparison of proposed estimators their performance is assessed in the context of RMS error and their susceptibility to the occurrence of track loss (i.e., percentage of track loss).

Root means square error (RMSE) of position, velocity and turn rate estimation are calculated using the formula given in [Jia2013b].

$$RMSE = \sqrt{\frac{1}{N_{mc}} \sum_{i=1}^{N_{mc}} \left(\left(\boldsymbol{\xi}_{k,i} \boldsymbol{e}_j - \hat{\boldsymbol{\xi}}_{k,i} \boldsymbol{e}_j \right)^2 + \left(\boldsymbol{\xi}_{k,i} \boldsymbol{e}_l - \hat{\boldsymbol{\xi}}_{k,i} \boldsymbol{e}_l \right)^2 \right)}$$

where $j=1$ and $l=3$ for RMSE of position estimation. For RMSE of velocity estimation $j=2$ and $l=4$. RMSE for turn rate estimation is obtained with $j=5$ and replacing the unit vector $\boldsymbol{e}_l$ by a zero vector.

To detect the occurrence of track loss the following condition has been considered. When the condition $\left\| \sqrt{(x_k - \hat{x}_k)^2 + (y_k - \hat{y}_k)^2} \right\|_\infty \geq 800m$ is true for any instant of Monte Carlo runs, it is understood that the estimated trajectory has failed to track the true trajectory of the aircraft.

Chapter 4: A General Framework for Adaptive Nonlinear Filters

4.1 Chapter Introduction

This chapter presents a general framework for adaptive nonlinear filters which would be useful to formulate variants of adaptive estimators for nonlinear signal models where the prior knowledge of process noise covariance ($\boldsymbol{Q}$) or measurement noise covariance ($\boldsymbol{R}$) remains unavailable. The adaptation algorithms for $\boldsymbol{Q}$ and $\boldsymbol{R}$ which have been incorporated in the proposed general framework have been mathematically derived in this chapter. The adaptation algorithms necessitate an underlying framework of conventional (non-adaptive) nonlinear filter wherein these algorithms have to be integrated. Therefore, for the proposed general framework the conventional Bayesian approach of filtering (in presence of additive Gaussian noise) has been used as the underlying framework. With the help of this general framework a class of adaptive nonlinear filters can be formulated by approximating the Bayesian integrals using several numerical methods.

The adaptive nonlinear estimators, in general, require statistics of the state residual or the measurement residual for adaptation. The adaptation algorithms depend on the underlying framework of non-adaptive nonlinear filters for the knowledge of state residual or measurement residual. Subsequently such adaptation algorithms provide the adapted value of the unknown noise covariance which is used by the underlying framework of non-adaptive nonlinear filters to compute the state/measurement residuals of the next time instant. In this way these two sets of algorithms work as the complement of one another. The general framework is presented in terms of such complementary sections based on Bayesian filtering algorithm and the adaptation algorithms respectively.

Adaptation algorithms which have been mathematically derived in this chapter are presented in the format of theorems along with their proofs. The methods for adaptation are inspired from the linear signal models and duly extended for nonlinear state estimation. Depending on the situations with unknown process noise covariance and measurement noise covariance the adaptation methods are broadly categorized as '$\boldsymbol{Q}$ adaptive' and '$\boldsymbol{R}$ adaptive' nonlinear filters. Different approaches followed for derivation of the adaptation algorithms include:

(i) Maximum Likelihood Estimation (MLE) based method (ii) Covariance Matching method (iii) Maximum a Posteriori Estimation (MAP) based method.

Depending of the nature of adaptation and choice of sigma points variants of adaptive nonlinear estimators can be formulated which would be demonstrated in the subsequent chapters of this dissertation. In this chapter only the $\boldsymbol{R}$ adaptive UKF has been presented to demonstrate the use of the proposed general framework. Performance of AUKF is also illustrated with help of a case study.

4.1.1 Problem Statement

We consider a nonlinear dynamic equation of system as given below

$$\boldsymbol{x_k} = \boldsymbol{f}(\boldsymbol{x_{k-1}}) + \boldsymbol{w_k} \tag{4.1}$$

where $\boldsymbol{x_k} \in \Re^{\boldsymbol{n}}$ is the state vector and $\boldsymbol{w_k}$ is the noise term which represents modelling uncertainties. $\boldsymbol{w_k}$ is white Gaussian noise $\boldsymbol{w_k} \in \Re^{\boldsymbol{n}} \sim (\boldsymbol{0}, \boldsymbol{Q})$. For joint estimation of parameters and state the state vector becomes a parameter augmented state. This implies that apart from the n_ψ proper states, n_ζ unknown parameters have been augmented such that $n = n_\psi + n_\zeta$. When the dynamics of parameter variation remains unknown it is assumed that unknown parameters vary following a simple random walk model $\boldsymbol{\zeta_k} = \boldsymbol{\zeta_{k-1}} + \boldsymbol{w_k^\zeta}$, where $\boldsymbol{w_k^\zeta}$ indicates a zero mean Gaussian noise sequence with its covariance symbolized by $\boldsymbol{Q_k^\zeta}$.

The observation equation is considered as

$$\boldsymbol{y_k} = \boldsymbol{g}(\boldsymbol{x_k}) + \boldsymbol{v_k} \tag{4.2}$$

$\boldsymbol{y_k} \in \Re^{\boldsymbol{m}}$ is the observed output vector.

$\boldsymbol{v_k}$ is the measurement noise (Gaussian) and $\boldsymbol{v_k} \in \Re^{\boldsymbol{m}} \sim (\boldsymbol{0}, \boldsymbol{R})$.

Situations are considered when the knowledge of the noise covariances remains incomplete, i.e., the knowledge of process noise covariance ($\boldsymbol{Q}$) or, measurement noise covariance ($\boldsymbol{R}$) remains unavailable.

The objective of adaptive nonlinear filtering is to find the conditional expectation $E(\boldsymbol{x_k}|\boldsymbol{Y_k})$ (where $\boldsymbol{Y_k}$ is the set of observed data, $\boldsymbol{Y_k} = y_j\ ; 1 \le j \le k$) along with online adaptation of unknown noise covariances. The initial choice of the state estimate $\hat{\boldsymbol{x}}_0$ is considered to be a Gaussian prior with mean $\boldsymbol{x}_o$ and covariance $\hat{\boldsymbol{P}}_0$.

When $\boldsymbol{Q}$ is unknown, the filter is initialized with an assumed value of $\overline{\boldsymbol{Q}}$. Alternatively, the filter is initialized with an assumed value of $\overline{\boldsymbol{R}}$ in the face of unknown $\boldsymbol{R}$. The assumed value of covariance has to be adapted using the proposed adaptation algorithms for obtaining satisfactory estimation results.

4.1.2 Different Approaches for Solution

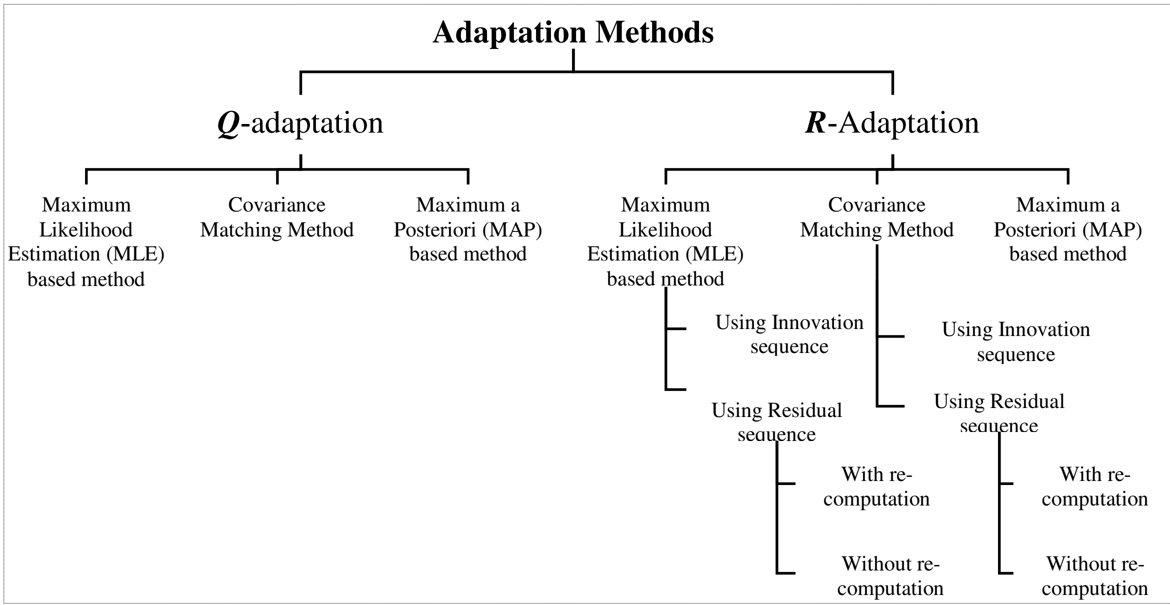
The objective of the adaptive nonlinear estimators is to provide satisfactory estimate of the states in the face of unknown noise covariances. Formal mathematical formulation of the estimation problem is presented in the previous section. It has been mentioned before that depending on the unavailability of process noise covariance or measurement noise covariance the adaptive filters are broadly classified into two classes: (i) $\boldsymbol{Q}$ adaptive filter (ii) $\boldsymbol{R}$ adaptive filter. $\boldsymbol{Q}$ or, $\boldsymbol{R}$ matrices are adapted by the adaptation algorithms which have been formulated with the help of three different approaches as stated below:

- Maximum Likelihood Estimation (MLE) based method
- Covariance Matching method
- Maximum a Posteriori (MAP) method

The adaptation algorithm for $\boldsymbol{Q}$ adaptive and $\boldsymbol{R}$ adaptive nonlinear estimators has been mathematically derived by Maximum Likelihood Estimation (MLE) based method, Covariance matching method and Maximum a Posteriori (MAP) based method for nonlinear signal models.

By modifying the existing MAP based algorithms for $\boldsymbol{R}$-adaptation and $\boldsymbol{Q}$- adaptation with reasonable simplifying assumptions it has been shown that the modified adaptation algorithms match well with those obtained by the MLE method and the intuitive Covariance Matching method.

Different methods of adaptation followed in this work are presented with the help of a tree diagram given below:



It is to be noted from the tree diagram that the ***R*** adaptation algorithms are further classified into two branches depending on the innovation based and residual based adaptation. The residual based method has some advantage over the innovation based method which would be discussed later in detail. In case of residual based adaptation, there exists a self referencing problem. This problem can be overcome by re-computation of measurement update steps with adapted ***R*** of current instant. The re-computation step is effective when there is transients in the adapted value of ***R*** or, truth value of ***R*** is time varying in nature. The residual based ***R***-adaptation is again divided into two categories: (i) with re-computation (ii) without re-computation. The method of re-computation has been explained in 4.2.3.5.

Note that the adaptive nonlinear filtering algorithms use the structure of non-adaptive nonlinear filters as their core. The variants of adaptation algorithms which may be derived following the above mentioned approaches are to be integrated in the underlying structure of non-adaptive nonlinear filtering algorithms so that their adaptive versions can be formulated.

4.2 The Solution Framework

4.2.1 Overview

The algorithm of adaptive nonlinear estimators in a general framework has been presented in this section. The adaptive nonlinear estimators are based on two major parts: (A) underlying framework of non-adaptive nonlinear filters which is the core for the adaptive nonlinear estimators, (B) The proposed adaptation algorithms.

The major steps of the proposed general framework are schematically presented below.

General Framework For Adaptive Nonlinear Estimators	
Part A: Underlying framework of non-adaptive estimators	
Gaussian integrals are approximated using numerical methods: • Unscented Transformation • Gauss Hermite quadrature rule • Spherical Radial Cubature rule • Cubature quadrature rule	
Part B: Adaptation algorithms	
Adaptation of Process Noise Covariance	Adaptation of Measurement Noise Covariance
Methods followed: • Maximum Likelihood Estimation based method • Covariance Matching method • Maximum a Posterior Estimation based method	

In part (A) the underlying framework of non-adaptive nonlinear estimators for Gaussian noise are presented following the Bayesian approach. The noises are considered to be Gaussian, white and additive in nature. Therefore, the estimation algorithm is expressed with the help of 'Gaussian integrals' as it is presented in many works [Sarkka2013a, Ito2000]. During implementation of these estimators such integrals are to be approximated with the help of numerical methods. Different approximation methods exist for approximation of the intractable integrals which would appear in the subsequent chapters of this dissertation. In part (B) different methods for adaptation of process noise and measurement noise covariance have been provided.

4.2.2 Part A: Underlying Framework of Non-adaptive Nonlinear Filter

The first part of the algorithm includes the non-adaptive nonlinear filtering algorithm which is to be used as an underlying framework with some modifications to ensure the compatibility with the adaptation portion. The adaptation methods are elaborated in the second section which is the significant portion of this algorithm.

In probabilistic terms the system and the measurement model with additive white noise (Gaussian) can be expressed as given by (4.1) and (4.2)

$$p\left(\boldsymbol{x}_k \middle| \boldsymbol{x}_{k-1}, \boldsymbol{Y}_{k-1}\right) = N\left(\boldsymbol{x}_k \middle| \boldsymbol{f}\left(\boldsymbol{x}_{k-1}\right), \overline{\boldsymbol{Q}}\right) \tag{4.3}$$

$$p(\boldsymbol{y}_k|\boldsymbol{x}_k,\boldsymbol{Y}_k) = N(\boldsymbol{y}_k|\boldsymbol{g}(\boldsymbol{x}_k),\overline{\boldsymbol{R}}) \tag{4.4}$$

For the above described system the conditional expectation of the state of the system can be denoted by $E(\boldsymbol{x}_k|\boldsymbol{Y}_k)$ where $\boldsymbol{Y}_k$ is the available measurements, $\boldsymbol{Y}_k = y_j\,; 1 \le j \le k$. To obtain the estimate of the state from the available set of measurements Gaussian approximation of probability density function of state (conditioned by the measurements) has to be made. This concept has been followed in previous works [Sarkka2013a, Wang2012, Arasaratnam2009, Haug2005, Ito2000] and termed as “Gaussian filters”. Here, the filtering algorithm is presented for the continuity of the adaptation algorithms presented in the next section.

<u>Assumptions 4.1</u>

The *a priori* (predictive) probability density function of the state $\boldsymbol{x}_k$ conditioned by $\boldsymbol{Y}_{k-1}$ is assumed to be Gaussian, i.e.,

$$p(\boldsymbol{x}_k|\boldsymbol{Y}_{k-1}) = N(\boldsymbol{x}_k;\overline{\boldsymbol{x}}_k,\overline{\boldsymbol{P}}_k)$$

where the first two moments, viz., *a priori* state estimate and corresponding error covariance are:

$$\overline{\boldsymbol{x}}_k = E(\boldsymbol{x}_k|\boldsymbol{Y}_{k-1})$$

$$\overline{\boldsymbol{P}}_k = E(\tilde{\boldsymbol{x}}_k\tilde{\boldsymbol{x}}_k^T|\boldsymbol{Y}_{k-1})$$

Here, the *a priori* error of state is defined as $\tilde{\boldsymbol{x}}_k \stackrel{\Delta}{=} \boldsymbol{x}_k - \overline{\boldsymbol{x}}_k$

Through out this dissertation $\boldsymbol{x} \sim N(\boldsymbol{x}|\boldsymbol{m},\boldsymbol{P})$ indicates that $\boldsymbol{x}$ is a Gaussian vector with mean $\boldsymbol{m}$ and covariance $\boldsymbol{P}$ and the probability density function of $\boldsymbol{x}$ is expressed as

$$p(\boldsymbol{x}) = \frac{1}{\sqrt{(2\pi)^n \boldsymbol{P}}}\exp\left((\boldsymbol{x}-\boldsymbol{m})\boldsymbol{P}^{-1}(\boldsymbol{x}-\boldsymbol{m})^T\right)$$

<u>Assumptions 4.2</u>

The *a priori* (predictive) probability density function of the measurement $\boldsymbol{y}_k$ conditioned by $\boldsymbol{Y}_{k-1}$ is to be Gaussian, i.e.,

$$p(\boldsymbol{y}_k|\boldsymbol{Y}_{k-1}) = N(\boldsymbol{y}_k;\overline{\boldsymbol{y}}_k,\boldsymbol{P}_k^y)$$

where the first two moments, viz., *a priori* estimate of measurement and corresponding error covariance are

$$\bar{\boldsymbol{y}}_k = E\left(\boldsymbol{y}_k \middle| \boldsymbol{Y}_{k-1}\right)$$

$$\boldsymbol{P}_k^y = E\left(\tilde{\boldsymbol{y}}_k \tilde{\boldsymbol{y}}_k^T \middle| \boldsymbol{Y}_{k-1}\right)$$

Here, *a priori* error of measurement is defined as $\tilde{\boldsymbol{y}}_k \overset{\Delta}{=} \boldsymbol{y}_k - \bar{\boldsymbol{y}}_k$

Assumptions 4.3

The *a posteriori* (updated) probability density function of the state $\boldsymbol{x}_k$ conditioned by $\boldsymbol{Y}_k$ is to be Gaussian, i.e.,

$$p\left(\boldsymbol{x}_k \middle| \boldsymbol{Y}_k\right) = N\left(\boldsymbol{x}_k ; \hat{\boldsymbol{x}}_k , \hat{\boldsymbol{P}}_k\right)$$

where the first two moments, i.e., *a posteriori* state estimate and corresponding error covariance are:

$$\hat{\boldsymbol{x}}_k = E\left(\boldsymbol{x}_k \middle| \boldsymbol{Y}_k\right)$$

$$\hat{\boldsymbol{P}}_k = E\left(\breve{\boldsymbol{x}}_k \breve{\boldsymbol{x}}_k^T \middle| \boldsymbol{Y}_{k-1}\right)$$

Here, the *a posteriori* error of state is defined as $\breve{\boldsymbol{x}}_k \overset{\Delta}{=} \boldsymbol{x}_k - \hat{\boldsymbol{x}}_k$

Lemma 4.1

If random variables $\boldsymbol{x} \in \Re^{\boldsymbol{n}}$ and $\boldsymbol{y} \in \Re^{\boldsymbol{m}}$ have the Gaussian probability densities as given below

$$\boldsymbol{x} \sim N\left(\boldsymbol{x} \middle| \boldsymbol{m}, \boldsymbol{P}\right)$$

$$\boldsymbol{y} \sim N\left(\boldsymbol{y} \middle| \boldsymbol{Hx} + \boldsymbol{u}, \boldsymbol{R}\right), \quad \boldsymbol{H} \in \Re^{m \times n}, \boldsymbol{R} \in \Re^{m \times m}$$

then the joint density of ***x, y*** and the marginal distribution of ***y*** are given as

$$\boldsymbol{x}, \boldsymbol{y} \sim N\left(\begin{bmatrix} \boldsymbol{m} \\ \boldsymbol{Hm} + \boldsymbol{u} \end{bmatrix}, \begin{bmatrix} \boldsymbol{P} & \boldsymbol{PH}^T \\ \boldsymbol{HP} & \boldsymbol{HPH}^T + \boldsymbol{R} \end{bmatrix}\right)$$

$$\boldsymbol{y} \sim N\left(\boldsymbol{y} \middle| \boldsymbol{Hm} + \boldsymbol{u},\ \boldsymbol{HPH}^T + \boldsymbol{R}\right)$$

Lemma 4.2

If the random variables $\boldsymbol{x}$ and $\boldsymbol{y}$ ($\boldsymbol{x} \in \Re^{\boldsymbol{n}}$, $\boldsymbol{y} \in \Re^{\boldsymbol{m}}$) have the joint Gaussian probability density

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

$\boldsymbol{a} \in \Re^{\boldsymbol{n}}$, $\boldsymbol{b} \in \Re^{m}$, $\boldsymbol{A} \in \Re^{\boldsymbol{n}\times n}$, $\boldsymbol{B} \in \Re^{m\times m}$, $\boldsymbol{C} \in \Re^{n\times m}$

then the marginal and conditional densities of $\boldsymbol{x}$ and $\boldsymbol{y}$ are given as follows:

$$\boldsymbol{x}|\boldsymbol{y} \sim N\left(\boldsymbol{a} + \boldsymbol{C}\boldsymbol{B}^{-1}(\boldsymbol{y}-\boldsymbol{b}), \boldsymbol{A} - \boldsymbol{C}\boldsymbol{B}^{-1}\boldsymbol{C}^T\right)$$

$$\boldsymbol{y}|\boldsymbol{x} \sim N\left(\boldsymbol{b} + \boldsymbol{C}^T\boldsymbol{B}^{-1}(\boldsymbol{x}-\boldsymbol{a}), \boldsymbol{B} - \boldsymbol{C}^T\boldsymbol{A}^{-1}\boldsymbol{C}\right)$$

Theorem 4.1

Considering the system dynamics given by (4.1) and (4.2) and using the assumptions from 4.1 to 4.3 and the Lemma 4.1 & 4.2 the *a posteriori* (also known as updated) estimate of state and the error covariance can be obtained using the following recursive formula:

$$\hat{\boldsymbol{x}}_k = \bar{\boldsymbol{x}}_k + \boldsymbol{K}_k(\boldsymbol{y}_k - \bar{\boldsymbol{y}}_k) \tag{4.5}$$

$$\hat{\boldsymbol{P}}_k = \bar{\boldsymbol{P}}_k - \boldsymbol{K}_k \boldsymbol{P}_k^y \boldsymbol{K}_k^T \tag{4.6}$$

Where,

$$\bar{\boldsymbol{x}}_k = \int_{\boldsymbol{R}^n} \boldsymbol{f}(\boldsymbol{x}_{k-1}) p(\boldsymbol{x}_{k-1}|\boldsymbol{Y}_{k-1}) d\boldsymbol{x}_{k-1} \tag{4.7}$$

$$\bar{\boldsymbol{P}}_k = \bar{\boldsymbol{Q}} + \int_{\boldsymbol{R}^n} (\boldsymbol{f}(\boldsymbol{x}_{k-1}) - \bar{\boldsymbol{x}}_k)(\boldsymbol{f}(\boldsymbol{x}_{k-1}) - \bar{\boldsymbol{x}}_k)^T p(\boldsymbol{x}_{k-1}|\boldsymbol{Y}_{k-1}) d\boldsymbol{x}_{k-1} \tag{4.8}$$

$$\bar{\boldsymbol{y}}_k = \int_{\boldsymbol{R}^n} \boldsymbol{g}(\boldsymbol{x}_k) p(\boldsymbol{x}_k|\boldsymbol{Y}_{k-1}) d\boldsymbol{x}_k \tag{4.9}$$

$$\boldsymbol{P}_k^{xy} = \int_{\boldsymbol{R}^n} (\boldsymbol{f}(\boldsymbol{x}_k) - \bar{\boldsymbol{x}}_k)(\boldsymbol{g}(\boldsymbol{x}_k) - \bar{\boldsymbol{y}}_k)^T p(\boldsymbol{x}_k|\boldsymbol{Y}_{k-1}) d\boldsymbol{x}_k \tag{4.10}$$

$$\boldsymbol{P}_k^y = \bar{\boldsymbol{R}} + \int_{\boldsymbol{R}^n} (\boldsymbol{g}(\boldsymbol{x}_k) - \bar{\boldsymbol{y}}_k)(\boldsymbol{g}(\boldsymbol{x}_k) - \bar{\boldsymbol{y}}_k)^T p(\boldsymbol{x}_k|\boldsymbol{Y}_{k-1}) d\boldsymbol{x}_k \tag{4.11}$$

$$\boldsymbol{K}_k = \boldsymbol{P}_k^{xy}\left(\boldsymbol{P}_k^y\right)^{-1} \tag{4.12}$$

Proof:

The *a priori* probability density function can be obtained using Chapman–Kolmogorov equation as:

$$p(\boldsymbol{x}_k|\boldsymbol{Y}_{k-1}) = \int_{\boldsymbol{R}^n} p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1})p(\boldsymbol{x}_{k-1}|\boldsymbol{Y}_{k-1})d\boldsymbol{x}_{k-1}$$

Where, $p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1},\boldsymbol{Y}_{k-1}) = N(\boldsymbol{x}_k;\boldsymbol{f}(\boldsymbol{x}_k),\overline{\boldsymbol{Q}})$

From the assumption 4.1 we have

$$\overline{\boldsymbol{x}}_k = E(\boldsymbol{x}_k|\boldsymbol{Y}_{k-1})$$

$$\Rightarrow \overline{\boldsymbol{x}}_k = \int_{\boldsymbol{R}^n} \boldsymbol{x}_k\, p(\boldsymbol{x}_k|\boldsymbol{Y}_{k-1})d\boldsymbol{x}_k$$

$$\Rightarrow \overline{\boldsymbol{x}}_k = \int_{\boldsymbol{R}^n} \boldsymbol{x}_k \left[\int_{\boldsymbol{R}^n} p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1})p(\boldsymbol{x}_{k-1}|\boldsymbol{Y}_{k-1})d\boldsymbol{x}_{k-1}\right]d\boldsymbol{x}_k$$

$$\Rightarrow \overline{\boldsymbol{x}}_k = \int_{\boldsymbol{R}^n} \boldsymbol{f}(\boldsymbol{x}_k) N(\boldsymbol{x}_k;\hat{\boldsymbol{x}}_{k-1},\hat{\boldsymbol{P}}_{k-1})d\boldsymbol{x}_{k-1}$$

$$\Rightarrow \overline{\boldsymbol{x}}_k = \int_{\boldsymbol{R}^n}\left[\int_{\boldsymbol{R}^n} \boldsymbol{x}_k N(\boldsymbol{x}_k;\boldsymbol{f}(\boldsymbol{x}_{k-1}),\overline{\boldsymbol{Q}})d\boldsymbol{x}_k\right]p(\boldsymbol{x}_{k-1}|\boldsymbol{Y}_{k-1})d\boldsymbol{x}_{k-1}$$

$$\Rightarrow \overline{\boldsymbol{x}}_k = \int_{\boldsymbol{R}^n} \boldsymbol{f}(\boldsymbol{x}_{k-1})p(\boldsymbol{x}_{k-1}|\boldsymbol{Y}_{k-1})d\boldsymbol{x}_{k-1}$$

Hence, (4.7) is proved. Equation (4.7) represents the *a priori* (predicted) estimate of state.

$$\overline{\boldsymbol{P}}_k = E(\tilde{\boldsymbol{x}}_k\tilde{\boldsymbol{x}}_k^T|\boldsymbol{Y}_{k-1})$$

$$\Rightarrow \overline{\boldsymbol{P}}_k = \int_{\boldsymbol{R}^n} \tilde{\boldsymbol{x}}_k\tilde{\boldsymbol{x}}_k^T\left[\int_{\boldsymbol{R}^n} p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1})p(\boldsymbol{x}_{k-1}|\boldsymbol{Y}_{k-1})d\boldsymbol{x}_{k-1}\right]d\boldsymbol{x}_k$$

$$\Rightarrow \overline{\boldsymbol{P}}_k = \int_{\boldsymbol{R}^n}\left[\int_{\boldsymbol{R}^n} \tilde{\boldsymbol{x}}_k\tilde{\boldsymbol{x}}_k^T N(\boldsymbol{x}_k;\boldsymbol{f}(\boldsymbol{x}_{k-1}),\overline{\boldsymbol{Q}})d\boldsymbol{x}_k\right]p(\boldsymbol{x}_{k-1}|\boldsymbol{Y}_{k-1})d\boldsymbol{x}_{k-1}$$

$$\Rightarrow \overline{\boldsymbol{P}}_k = \int_{\boldsymbol{R}^n}\left((\boldsymbol{f}(\boldsymbol{x}_{k-1})-\overline{\boldsymbol{x}}_k)(\boldsymbol{f}(\boldsymbol{x}_{k-1})-\overline{\boldsymbol{x}}_k)^T+\overline{\boldsymbol{Q}}\right)p(\boldsymbol{x}_{k-1}|\boldsymbol{Y}_{k-1})d\boldsymbol{x}_{k-1}$$

$$\Rightarrow \overline{\boldsymbol{P}}_k = \overline{\boldsymbol{Q}} + \int_{\boldsymbol{R}^n} (\boldsymbol{f}(\boldsymbol{x}_{k-1}) - \overline{\boldsymbol{x}}_k)(\boldsymbol{f}(\boldsymbol{x}_{k-1}) - \overline{\boldsymbol{x}}_k)^T p(\boldsymbol{x}_{k-1}|\boldsymbol{Y}_{k-1}) d\boldsymbol{x}_{k-1}$$

$$\Rightarrow \overline{\boldsymbol{P}}_k = \overline{\boldsymbol{Q}} + \int_{\boldsymbol{R}^n} (\boldsymbol{f}(\boldsymbol{x}_{k-1}) - \overline{\boldsymbol{x}}_k)(\boldsymbol{f}(\boldsymbol{x}_{k-1}) - \overline{\boldsymbol{x}}_k)^T N(\boldsymbol{x}_k; \hat{\boldsymbol{x}}_{k-1}, \hat{\boldsymbol{P}}_{k-1}) d\boldsymbol{x}_{k-1}$$

Hence, (4.8) is proved. Equation (4.8) represents the *a priori* (predicted) error covariance of state.

For future reference we denote $\overline{\boldsymbol{P}}_k^f$ as

$$\overline{\boldsymbol{P}}_k^f = \int_{\boldsymbol{R}^n} (\boldsymbol{f}(\boldsymbol{x}_{k-1}) - \overline{\boldsymbol{x}}_k)(\boldsymbol{f}(\boldsymbol{x}_{k-1}) - \overline{\boldsymbol{x}}_k)^T p(\boldsymbol{x}_{k-1}|\boldsymbol{Y}_{k-1}) d\boldsymbol{x}_{k-1}$$

Joint distribution of $\boldsymbol{y}_k$ and $\boldsymbol{x}_k$ can be obtained as

$$p(\boldsymbol{y}_k, \boldsymbol{x}_k|\boldsymbol{Y}_{k-1}) = p(\boldsymbol{y}_k|\boldsymbol{x}_k) p(\boldsymbol{x}_k|\boldsymbol{Y}_{k-1})$$

$$p(\boldsymbol{y}_k, \boldsymbol{x}_k|\boldsymbol{Y}_{k-1}) = p(\boldsymbol{y}_k|\boldsymbol{x}_k, \boldsymbol{Y}_{k-1}) p(\boldsymbol{x}_k|\boldsymbol{Y}_{k-1})$$

The disappearance of the measurement history $\boldsymbol{Y}_{k-1}$ is due to the conditional independence of $\boldsymbol{y}_k$ of the measurement history, given , $\boldsymbol{x}_k$.

The marginal distribution of $\boldsymbol{y}_k$ given $\boldsymbol{Y}_{k-1}$ can be obtained by integrating the distribution over $d\boldsymbol{x}_k$. The relation is obtained following Chapman–Kolmogorov equation as

$$p(\boldsymbol{y}_k|\boldsymbol{Y}_{k-1}) = \int_{R^n} p(\boldsymbol{y}_k|\boldsymbol{x}_k) p(\boldsymbol{x}_k|\boldsymbol{Y}_{k-1}) d\boldsymbol{x}_k$$

From assumption 4.2 we have

$$\overline{\boldsymbol{y}}_k = E(\boldsymbol{y}_k|\boldsymbol{Y}_{k-1})$$

$$\Rightarrow \overline{\boldsymbol{y}}_k = \int_{\boldsymbol{R}^n} \boldsymbol{y}_k\, p(\boldsymbol{y}_k|\boldsymbol{Y}_{k-1}) d\boldsymbol{x}_k$$

$$\Rightarrow \overline{\boldsymbol{y}}_k = \int_{\boldsymbol{R}^n} \boldsymbol{y}_k \left[\int_{\boldsymbol{R}^n} p(\boldsymbol{y}_k|\boldsymbol{x}_k) p(\boldsymbol{x}_k|\boldsymbol{Y}_{k-1}) d\boldsymbol{x}_k \right] d\boldsymbol{x}_k$$

$$\Rightarrow \overline{\boldsymbol{y}}_k = \int_{\boldsymbol{R}^n} \left[\int_{\boldsymbol{R}^n} \boldsymbol{y}_k\, p(\boldsymbol{y}_k|\boldsymbol{x}_k) d\boldsymbol{x}_k \right] p(\boldsymbol{x}_k|\boldsymbol{Y}_{k-1}) d\boldsymbol{x}_k$$

$$\Rightarrow \bar{\boldsymbol{y}}_k = \int_{R^n}\left[\int_{R^n} \boldsymbol{y}_k N\left(\boldsymbol{y}_k \middle| \boldsymbol{g}(\boldsymbol{x}_k), \overline{\boldsymbol{R}}\right) d\boldsymbol{x}_k\right] p\left(\boldsymbol{x}_k \middle| \boldsymbol{Y}_{k-1}\right) d\boldsymbol{x}_k$$

$$\Rightarrow \bar{\boldsymbol{y}}_k = \int_{R^n} \boldsymbol{g}(\boldsymbol{x}_k) p\left(\boldsymbol{x}_k \middle| \boldsymbol{Y}_{k-1}\right) d\boldsymbol{x}_k$$

$$\Rightarrow \bar{\boldsymbol{y}}_k = \int_{R^n} \boldsymbol{g}(\boldsymbol{x}_k) N\left(\boldsymbol{x}_k ; \bar{\boldsymbol{x}}_k, \overline{\boldsymbol{P}}_k\right) d\boldsymbol{x}_k$$

Hence, (4.9) is proved. Equation (4.9) represents the *a priori* (predicted) estimate of measurement.

$$\boldsymbol{P}_k^y = E\left(\tilde{\boldsymbol{y}}_k \tilde{\boldsymbol{y}}_k^T \middle| \boldsymbol{Y}_{k-1}\right)$$

$$\Rightarrow \boldsymbol{P}_k^y = \int_{R^n} (\boldsymbol{g}(\boldsymbol{x}_k) - \bar{\boldsymbol{y}}_k + \boldsymbol{v}_k)(\boldsymbol{g}(\boldsymbol{x}_k) - \bar{\boldsymbol{y}}_k + \boldsymbol{v}_k)^T p\left(\boldsymbol{y}_k \middle| \boldsymbol{Y}_{k-1}\right) d\boldsymbol{x}_k$$

Where $\boldsymbol{v}_k$ is defined as $\boldsymbol{v}_k = \boldsymbol{y}_k - \boldsymbol{g}(\boldsymbol{x}_k)$

$$\Rightarrow \boldsymbol{P}_k^y = \int_{R^n} (\boldsymbol{g}(\boldsymbol{x}_k) - \bar{\boldsymbol{y}}_k + \boldsymbol{v}_k)(\boldsymbol{g}(\boldsymbol{x}_k) - \bar{\boldsymbol{y}}_k + \boldsymbol{v}_k)^T \left[\int_{R^n} p\left(\boldsymbol{y}_k \middle| \boldsymbol{x}_k\right) p\left(\boldsymbol{x}_k \middle| \boldsymbol{Y}_{k-1}\right) d\boldsymbol{x}_k\right] d\boldsymbol{x}_k$$

$$\Rightarrow \boldsymbol{P}_k^y = \int_{R^n}\left[\int_{R^n} (\boldsymbol{g}(\boldsymbol{x}_k) - \bar{\boldsymbol{y}}_k + \boldsymbol{v}_k)(\boldsymbol{g}(\boldsymbol{x}_k) - \bar{\boldsymbol{y}}_k + \boldsymbol{v}_k)^T p\left(\boldsymbol{y}_k \middle| \boldsymbol{x}_k\right) d\boldsymbol{x}_k\right] p\left(\boldsymbol{x}_k \middle| \boldsymbol{Y}_{k-1}\right) d\boldsymbol{x}_k$$

$$\Rightarrow \boldsymbol{P}_k^y = \int_{R^n}\left[\int_{R^n} (\boldsymbol{g}(\boldsymbol{x}_k) - \bar{\boldsymbol{y}}_k + \boldsymbol{v}_k)(\boldsymbol{g}(\boldsymbol{x}_k) - \bar{\boldsymbol{y}}_k + \boldsymbol{v}_k)^T N\left(\boldsymbol{y}_k \middle| \boldsymbol{g}(\boldsymbol{x}_k), \overline{\boldsymbol{R}}\right) d\boldsymbol{x}_k\right] p\left(\boldsymbol{x}_k \middle| \boldsymbol{Y}_{k-1}\right) d\boldsymbol{x}_k$$

$$\Rightarrow \boldsymbol{P}_k^y = \int_{R^n}\left[\int_{R^n} (\boldsymbol{g}(\boldsymbol{x}_k) - \bar{\boldsymbol{y}}_k)(\boldsymbol{g}(\boldsymbol{x}_k) - \bar{\boldsymbol{y}}_k)^T N\left(\boldsymbol{y}_k \middle| \boldsymbol{g}(\boldsymbol{x}_k), \overline{\boldsymbol{R}}\right) d\boldsymbol{x}_k + \int_{R^n} \boldsymbol{v}_k \boldsymbol{v}_k^T N\left(\boldsymbol{y}_k \middle| \boldsymbol{g}(\boldsymbol{x}_k), \overline{\boldsymbol{R}}\right) d\boldsymbol{x}_k\right] p\left(\boldsymbol{x}_k \middle| \boldsymbol{Y}_{k-1}\right) d\boldsymbol{x}_k$$

$$\Rightarrow \boldsymbol{P}_k^y = \int_{R^n} \left[(\boldsymbol{g}(\boldsymbol{x}_k) - \bar{\boldsymbol{y}}_k)(\boldsymbol{g}(\boldsymbol{x}_k) - \bar{\boldsymbol{y}}_k)^T + \overline{\boldsymbol{R}}\right] p\left(\boldsymbol{x}_k \middle| \boldsymbol{Y}_{k-1}\right) d\boldsymbol{x}_k$$

$$\Rightarrow \boldsymbol{P}_k^y = \overline{\boldsymbol{R}} + \int_{R^n} (\boldsymbol{g}(\boldsymbol{x}_k) - \bar{\boldsymbol{y}}_k)(\boldsymbol{g}(\boldsymbol{x}_k) - \bar{\boldsymbol{y}}_k)^T p\left(\boldsymbol{x}_k \middle| \boldsymbol{Y}_{k-1}\right) d\boldsymbol{x}_k$$

$$\Rightarrow \boldsymbol{P}_k^y = \overline{\boldsymbol{R}} + \int_{R^n} (\boldsymbol{g}(\boldsymbol{x}_k) - \bar{\boldsymbol{y}}_k)(\boldsymbol{g}(\boldsymbol{x}_k) - \bar{\boldsymbol{y}}_k)^T N\left(\boldsymbol{x}_k ; \bar{\boldsymbol{x}}_k, \overline{\boldsymbol{P}}_k\right) d\boldsymbol{x}_k$$

Hence, (4.11) is proved.

For future reference we denote $\overline{\boldsymbol{P}}_k^g$ as:

$$\overline{\boldsymbol{P}}_k^g = \int_{\boldsymbol{R}^n} \left(\boldsymbol{g}(\boldsymbol{x}_k) - \overline{\boldsymbol{y}}_k\right)\left(\boldsymbol{g}(\boldsymbol{x}_k) - \overline{\boldsymbol{y}}_k\right)^T p\left(\boldsymbol{x}_k \middle| \boldsymbol{Y}_{k-1}\right) d\boldsymbol{x}_k$$

The cross covariance because of the correlation between $\boldsymbol{y}_k$ and $\boldsymbol{x}_k$ can be obtained as

$$\boldsymbol{P}_k^{xy} = \int_{\boldsymbol{R}^n} (\boldsymbol{x}_k - \overline{\boldsymbol{x}}_k)\left(\boldsymbol{g}(\boldsymbol{x}_k) - \overline{\boldsymbol{y}}_k\right)^T p\left(\boldsymbol{x}_k \middle| \boldsymbol{Y}_{k-1}\right) d\boldsymbol{x}_k$$

$$\Rightarrow \boldsymbol{P}_k^{xy} = \int_{\boldsymbol{R}^n} (\boldsymbol{x}_k - \overline{\boldsymbol{x}}_k)\left(\boldsymbol{g}(\boldsymbol{x}_k) - \overline{\boldsymbol{y}}_k\right)^T N\left(\boldsymbol{x}_k ; \overline{\boldsymbol{x}}_k, \overline{\boldsymbol{P}}_k\right) d\boldsymbol{x}_k$$

Hence, (4.10) is proved.

By Lemma 4.1, the joint distribution of $\boldsymbol{y}_k$ and $\boldsymbol{x}_k$ given $\boldsymbol{Y}_{k-1}$ is presented by

$$p(\boldsymbol{x}_k, \boldsymbol{y}_k | \boldsymbol{Y}_{k-1}) = p(\boldsymbol{y}_k | \boldsymbol{x}_k)\, p(\boldsymbol{x}_k | \boldsymbol{Y}_{k-1})$$

$$\Rightarrow p(\boldsymbol{x}_k, \boldsymbol{y}_k | \boldsymbol{Y}_{k-1}) = N\left(\boldsymbol{y}_k ; \boldsymbol{g}(\boldsymbol{x}_k), \overline{\boldsymbol{R}}\right) N\left(\boldsymbol{x}_k ; \overline{\boldsymbol{x}}_k, \overline{\boldsymbol{P}}_k\right)$$

$$\Rightarrow p(\boldsymbol{x}_k, \boldsymbol{y}_k | \boldsymbol{Y}_{k-1}) = N\left[\begin{pmatrix} \overline{\boldsymbol{x}}_k \\ \overline{\boldsymbol{y}}_k \end{pmatrix}, \begin{pmatrix} \overline{\boldsymbol{P}}_k & \boldsymbol{P}_k^{xy} \\ \left(\boldsymbol{P}_k^{xy}\right)^T & \boldsymbol{P}_k^{y} \end{pmatrix}\right] \tag{4.13}$$

By Lemma 4.2 the conditional distribution of $\boldsymbol{x}_k$ is obtained as

$$p(\boldsymbol{x}_k | \boldsymbol{Y}_k) = N(\boldsymbol{x}_k ; \hat{\boldsymbol{x}}_k, \hat{\boldsymbol{P}}_k)$$

Where,

$$\hat{\boldsymbol{x}}_k = \overline{\boldsymbol{x}}_k + \boldsymbol{P}_k^{xy}\left(\boldsymbol{P}_k^{y}\right)^{-1}\left(\boldsymbol{y}_k - \overline{\boldsymbol{y}}_k\right) \tag{4.14}$$

$$\hat{\boldsymbol{P}}_k = \overline{\boldsymbol{P}}_k - \boldsymbol{P}_k^{xy}\left(\boldsymbol{P}_k^{y}\right)^{-1}\left(\boldsymbol{P}_k^{xy}\right)^T \tag{4.15}$$

We define the filter gain $\boldsymbol{K}_k$ as given by (4.12)

$$\boldsymbol{K}_k \stackrel{\Delta}{=} \boldsymbol{P}_k^{xy}\left(\boldsymbol{P}_k^{y}\right)^{-1}$$

With this expression of $\boldsymbol{K}_k$ (4.14) and (4.15) can be expressed as

$$\hat{\boldsymbol{x}}_k = \overline{\boldsymbol{x}}_k + \boldsymbol{K}_k\left(\boldsymbol{y}_k - \overline{\boldsymbol{y}}_k\right)$$

$$\hat{\boldsymbol{P}}_k = \overline{\boldsymbol{P}}_k - \boldsymbol{K}_k\left(\boldsymbol{P}_k^{y}\right)\left(\boldsymbol{K}_k\right)^T$$

Hence, (4.5) and (4.6) are proved □

4.2.2.1. Implementation of Bayesian Filters

For linear signal models the filtering steps can be readily deduced using the Bayesian framework and the subsequent estimation algorithm is well known as Kalman filter. However, for nonlinear signal models where state and observation equations are nonlinear function of system states Bayesian integrals becomes intractable. As a consequence, Bayesian framework for nonlinear estimation with Gaussian noise, though theoretically justified, becomes difficult to implement in real life applications. For implementation of the Bayesian filters with Gaussian noise, the Gaussian weighted Integrals encountered in the filtering steps are to be evaluated with the help of numerical methods.

Consider a multi-dimensional weighted integral of the form

$$I(\boldsymbol{f}) = \int_{\boldsymbol{R}^n} \boldsymbol{f}(\boldsymbol{x})\omega(\boldsymbol{x})d\boldsymbol{x} \tag{4.16}$$

Where $\boldsymbol{f}(\boldsymbol{x})$ is a nonlinear function of $\boldsymbol{x} \in \Re^n$. $\omega(\boldsymbol{x})$ is a Gaussian density function present in the Gaussian weighted integral. Following the numerical methods the above intractable integral can be computed by a weighted sum of the function evaluations. The basic concept of numerical methods is to generate a set of points $\boldsymbol{x}_i$ (usually called as sigma points) and weights w_i such that

$$I(\boldsymbol{f}) \approx \sum_{i=1}^{m} \boldsymbol{f}(\boldsymbol{x}_i) w_i \tag{4.17}$$

Several methods are available for numerical approximation of the integrals from which a few methods has been considered in this dissertation to formulate a class of adaptive sigma point filters. Some discrete points (will be referred henceforth as 'sigma points') are generated in a deterministic approach using Unscented Transformation rule, Gauss Hermite Quadrature rule, Spherical Radial Cubature rule, Cubature Quadrature rule. The 'sigma points' selected using these rules can be plugged in the general framework and the corresponding adaptive estimator may be formulated. The adaptive nonlinear estimators developed using the referred numerical approximation methods in the general framework will be characterised in sequel in the succeeding chapters.

4.2.3 Part B: Derivation of Adaptation Algorithm

4.2.3.1. Adaptation of the Process Noise Covariance (Q)

Maximum Likelihood Estimation (MLE) based method

Assumption 4.4:

For adaptation of noise covariances a fixed-length memory (window) of innovation or, residual sequence has to be considered. Following *assumptions* are to be made for the MLE based adaptation algorithm [Mohamed1999, Maybeck1982]:

- The state vector $\boldsymbol{x}$ is independent of the adaptive parameters (noise covariances), $\boldsymbol{\alpha}$, i.e., $\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{\alpha}} = 0$
- The system dynamics $\boldsymbol{f}(\cdot)$ and the measurement equation $\boldsymbol{g}(\cdot)$ are time invariant and does not depend on the adaptive parameters $\boldsymbol{\alpha}$
- The innovation sequence is a white and ergodic sequence (time average is equal with ensemble average) within the estimation window with window length L.

The innovation, or, residual covariance matrix is dependent on adaptation parameters $\boldsymbol{\alpha}$. It is with the help of window estimated innovation/residual covariance using which adapted values of parameters are deduced.

Theorem 4.2:

For nonlinear estimators with unknown process noise covariance, the adapted $\boldsymbol{Q}$ is expressed as

$$\hat{\boldsymbol{Q}}_k = \boldsymbol{K}_k \hat{\boldsymbol{C}}_{\vartheta_k} \boldsymbol{K}_k^T \tag{4.18}$$

Where $\hat{\boldsymbol{C}}_{\vartheta_k}$ is estimated innovation covariance obtained from the sliding window of length L.

Proof:

The $\boldsymbol{Q}$ adaptation steps are derived following MLE method and using innovation sequence from a sliding window (also, known as estimation window) and. The steps for derivation of adapted $\boldsymbol{Q}$ are inspired from the work of [Mohamed1999, Maybeck1982] for linear signal models. The probability density function of the measurements conditioned on adaptive parameter, $\boldsymbol{\alpha}$ at specific epoch k is chosen based on innovation sequence as given below. The objective of the MLE method is to maximize the probability density function for the choice of adaptive parameter

$$\boldsymbol{P}_{(y|\alpha)_k} = \frac{1}{\sqrt{(2\pi)^m \left|\boldsymbol{C}_{\vartheta_k}\right|}} \exp\left(-\tfrac{1}{2}\boldsymbol{\vartheta}_{\boldsymbol{k}}^{\boldsymbol{T}} \boldsymbol{C}_{\vartheta_k}^{-1} \boldsymbol{\vartheta}_{\boldsymbol{k}}\right) \tag{4.19}$$

$$\text{or, } \ln\left(\boldsymbol{P}_{(y|\alpha)_k}\right) = -\frac{1}{2}\left\{m\ln(2\pi) + \ln\left|\boldsymbol{C}_{\vartheta_k}\right| + \boldsymbol{\vartheta}_{\boldsymbol{k}}^{\boldsymbol{T}} \boldsymbol{C}_{\vartheta_k}^{-1} \boldsymbol{\vartheta}_{\boldsymbol{k}}\right\} \tag{4.20}$$

Multiplying both sides with -2 and neglecting the constant term we get the equation modified as

$$\boldsymbol{P} = \ln\left|\boldsymbol{C}_{\vartheta_k}\right| + \boldsymbol{\vartheta}_{\boldsymbol{k}}^{\boldsymbol{T}} \boldsymbol{C}_{\vartheta_k}^{-1} \vartheta_k \tag{4.21}$$

The innovation sequence has been considered inside a window size L as the filter uses a fixed length memory. The innovations inside the window will be summed. Multiplication with a negative value inverts the maximization problem into a minimization problem. Therefore, the Maximum Likelihood condition becomes:

$$\min\left\{\sum_{j=j_0}^{k}\left(\ln\left|\boldsymbol{C}_{\vartheta_k}\right| + \boldsymbol{\vartheta}_{\boldsymbol{k}}^{\boldsymbol{T}} \boldsymbol{C}_{\vartheta_k}^{-1} \vartheta_k\right)\right\} \tag{4.22}$$

where $j_0 = k - L + 1$ and L is the window size.

After differentiation with respect to adaptive factor '$\boldsymbol{\alpha}$' the likelihood function in (4.22) is expressed as:

$$\sum_{j=j_0}^{k}\left[tr\left\{\boldsymbol{C}_{\vartheta_j}^{-1}\left(\frac{\partial \boldsymbol{C}_{\vartheta_j}}{\partial \boldsymbol{\alpha}_{\boldsymbol{k}}}\right)\right\} - \boldsymbol{\vartheta}_{\boldsymbol{j}}^{\boldsymbol{T}} \boldsymbol{C}_{\vartheta_j}^{-1}\left(\frac{\partial \boldsymbol{C}_{\vartheta_j}}{\partial \boldsymbol{\alpha}_{\boldsymbol{k}}}\right)\boldsymbol{C}_{\vartheta_j}^{-1}\boldsymbol{\vartheta}_{\boldsymbol{j}}\right] = 0 \tag{4.23}$$

$$\sum_{j=j_0}^{k}\left[tr\left\{\boldsymbol{C}_{\vartheta_j}^{-1}\left(\frac{\partial \boldsymbol{C}_{\vartheta_j}}{\partial \boldsymbol{\alpha}_{\boldsymbol{k}}}\right)\right\} - tr\left\{\boldsymbol{C}_{\vartheta_j}^{-1}\boldsymbol{\vartheta}_{\boldsymbol{j}}\boldsymbol{\vartheta}_{\boldsymbol{j}}^{\boldsymbol{T}}\boldsymbol{C}_{\vartheta_j}^{-1}\left(\frac{\partial \boldsymbol{C}_{\vartheta_j}}{\partial \boldsymbol{\alpha}_{\boldsymbol{k}}}\right)\right\}\right] = 0 \tag{4.24}$$

$$\sum_{j=j_0}^{k}\left[tr\left\{\left[\boldsymbol{C}_{\vartheta_j}^{-1} - \boldsymbol{C}_{\vartheta_j}^{-1}\boldsymbol{\vartheta}_{\boldsymbol{j}}\boldsymbol{\vartheta}_{\boldsymbol{j}}^{\boldsymbol{T}}\boldsymbol{C}_{\vartheta_j}^{-1}\right]\left(\frac{\partial \boldsymbol{C}_{\vartheta_j}}{\partial \boldsymbol{\alpha}_{\boldsymbol{k}}}\right)\right\}\right] = 0 \tag{4.25}$$

The following formulae for matrix operation have been used for deriving the above steps are presented below and also appear in [Maybeck1982, Mohamed1999].

$$\frac{\partial \ln \boldsymbol{P}}{\partial \boldsymbol{x}} = |\boldsymbol{P}|^{-1} \frac{\partial |\boldsymbol{P}|}{\partial \boldsymbol{x}} = tr\left(\boldsymbol{P}^{-1} \frac{\partial \boldsymbol{P}}{\partial \boldsymbol{x}} \right)$$

$$\frac{\partial \boldsymbol{P}^{-1}}{\partial \boldsymbol{x}} = \boldsymbol{P}^{-1} \frac{\partial \boldsymbol{P}}{\partial \boldsymbol{x}} \boldsymbol{P}^{-1}$$

$$\boldsymbol{x}^T \boldsymbol{P}^T \boldsymbol{Q}\, \boldsymbol{P}\, \boldsymbol{x} = tr\left(\boldsymbol{P}\, \boldsymbol{x}\boldsymbol{x}^T \boldsymbol{P}^T \boldsymbol{Q}\right)$$

The deduction of the relation between innovation covariance, $\boldsymbol{C}_{\vartheta_j}$ and the process noise covariance, $\boldsymbol{Q}_k$ necessitates the availability of the *pseudo measurement matrix* for the nonlinear measurement equation which can be obtained by statistical linearization. The concept of pseudo measurement matrix for nonlinear measurement model is introduced first in [Lee2008] and also followed in [Chandra2011, Jia2013b, Soken2014].

The cross covariance $\boldsymbol{P}_k^{xy}$ and *a priori* error covariance $\overline{\boldsymbol{P}}_k$ given by (4.10) and (4.8) respectively are now used to define the pseudo measurement matrix, $\boldsymbol{\Psi}_k$ following [Lee2008] as

$$\boldsymbol{\Psi}_k = \left(\overline{\boldsymbol{P}}_k^{-1} \boldsymbol{P}_k^{xy}\right)^T \tag{4.26}$$

Using the pseudo measurement matrix the innovation covariance can be represented as $\boldsymbol{C}_{\vartheta_j} = \boldsymbol{R}_k + \boldsymbol{\Psi}_k \overline{\boldsymbol{P}}_k \boldsymbol{\Psi}_k^T$ which is analogous to that for linear signal model. The innovation covariance expressed in terms of pseudo measurement matrix is approximately equal with $\boldsymbol{P}_k^y$, i.e., $\boldsymbol{P}_k^y \approx \boldsymbol{R}_k + \boldsymbol{\Psi}_k \overline{\boldsymbol{P}}_k \boldsymbol{\Psi}_k^T$

For adaptation of $\boldsymbol{Q}$, the adaptive parameter $\boldsymbol{\alpha}$ is chosen as $\boldsymbol{\alpha}_i = \boldsymbol{Q}_{ii}$.

Substituting this expression of innovation covariance we get:

$$\frac{\partial \boldsymbol{C}_{\vartheta_k}}{\partial \boldsymbol{Q}_{kk}} = \boldsymbol{\Psi}_k \frac{\partial}{\partial \boldsymbol{Q}_{kk}} \left(\overline{\boldsymbol{P}}_k\right) \boldsymbol{\Psi}_k^T$$

$$\Rightarrow \frac{\partial \boldsymbol{C}_{\vartheta_k}}{\partial \boldsymbol{Q}_{kk}} = \boldsymbol{\Psi}_k \frac{\partial}{\partial \boldsymbol{Q}_{kk}} \left(\overline{\boldsymbol{P}}_k^f + \boldsymbol{Q}_k\right) \boldsymbol{\Psi}_k^T$$

$$\Rightarrow \frac{\partial \boldsymbol{C}_{\vartheta_k}}{\partial \boldsymbol{Q}_{kk}} = \boldsymbol{\Psi}_k \boldsymbol{I}\, \boldsymbol{\Psi}_k^T$$

It is assumed following the work of [Mohamed1999, Maybeck1982] that the within the estimation window the filter is in steady state and, therefore, derivative of the term $\overline{\boldsymbol{P}}_{\boldsymbol{k}}^{f}$ (related with *a posteriori* error covariance) in the expression of $\overline{\boldsymbol{P}}_{\boldsymbol{k}}$ can be neglected.

Substituting this value in the ML equation given by (4.25) we get

$$\sum_{j=j_0}^{k}\left[tr\left\{\left[\boldsymbol{C}_{\vartheta_j}^{-1}-\boldsymbol{C}_{\vartheta_j}^{-1}\vartheta_j\vartheta_j^T\boldsymbol{C}_{\vartheta_j}^{-1}\right]\left(\boldsymbol{\Psi}_j\boldsymbol{I}\boldsymbol{\Psi}_j^T\right)\right\}\right]=0 \tag{4.27}$$

Alternatively,

$$\sum_{j=j_0}^{k}\left[tr\left\{\boldsymbol{\Psi}_j^T\left[\boldsymbol{C}_{\vartheta_j}^{-1}-\boldsymbol{C}_{\vartheta_j}^{-1}\vartheta_j\vartheta_j^T\boldsymbol{C}_{\vartheta_j}^{-1}\right]\boldsymbol{\Psi}_j\right\}\right]=0 \tag{4.28}$$

$$\sum_{j=j_0}^{k}\left[tr\left\{\left[\boldsymbol{\Psi}_j^T\boldsymbol{C}_{\vartheta_j}^{-1}\boldsymbol{\Psi}_j-\boldsymbol{\Psi}_j^T\boldsymbol{C}_{\vartheta_j}^{-1}\vartheta_j\vartheta_j^T\boldsymbol{C}_{\vartheta_j}^{-1}\boldsymbol{\Psi}_j\right]\right\}\right]=0 \tag{4.29}$$

Note that using $\boldsymbol{\Psi}_{\boldsymbol{k}}$ filter gain $\boldsymbol{K}_{\boldsymbol{k}}$ given by (4.12) can be expressed as $\boldsymbol{K}_{\boldsymbol{k}}=\overline{\boldsymbol{P}}_{\boldsymbol{k}}\boldsymbol{\Psi}_{\boldsymbol{k}}^T\boldsymbol{C}_{\vartheta_k}^{-1}$. Substituting this expression in (4.29) we get

$$\sum_{j=j_0}^{k}\left[tr\left\{\overline{\boldsymbol{P}}_j^{-1}\boldsymbol{K}_j\boldsymbol{\Psi}_j-\overline{\boldsymbol{P}}_j^{-1}\boldsymbol{K}_j\vartheta_j\vartheta_j^T\boldsymbol{K}_j^T\overline{\boldsymbol{P}}_j^{-1}\right\}\right]=0 \tag{4.30}$$

$$\sum_{j=j_0}^{k}\left[tr\left\{\overline{\boldsymbol{P}}_j^{-1}\left(\boldsymbol{K}_j\boldsymbol{\Psi}_j\overline{\boldsymbol{P}}_j-\boldsymbol{K}_j\vartheta_j\vartheta_j^T\boldsymbol{K}_j^T\right)\overline{\boldsymbol{P}}_j^{-1}\right\}\right]=0 \tag{4.31}$$

From (4.5) the term $\boldsymbol{K}_j\vartheta_j$ can also be represented as $\boldsymbol{K}_j\vartheta_j=\hat{\boldsymbol{x}}_j-\overline{\boldsymbol{x}}_j=\boldsymbol{\eta}_j$. The state residual is represented by $\boldsymbol{\eta}_j$.

$$\sum_{j=j_0}^{k}\left[tr\left\{\overline{\boldsymbol{P}}_j^{-1}\left(\boldsymbol{K}_j\boldsymbol{\Psi}_j\overline{\boldsymbol{P}}_j-\boldsymbol{\eta}_j\boldsymbol{\eta}_j^T\right)\overline{\boldsymbol{P}}_j^{-1}\right\}\right]=0 \tag{4.32}$$

The expression of $\overline{\boldsymbol{P}}_j$ in (4.8) ensures that $\overline{\boldsymbol{P}}_j$ is positive definite. Therefore, above expression vanishes only when

$$\sum_{j=j_0}^{k}\left[tr\left\{\left(\boldsymbol{K}_j\boldsymbol{\Psi}_j\overline{\boldsymbol{P}}_j-\boldsymbol{\eta}_j\boldsymbol{\eta}_j^T\right)\right\}\right]=0 \tag{4.33}$$

Using (4.6) and (4.12) we get

$$\sum_{j=j_0}^{k}\left[tr\{(\overline{\boldsymbol{P}}_j - \hat{\boldsymbol{P}}_j - \boldsymbol{\eta}_j\boldsymbol{\eta}_j^T)\}\right] = 0 \tag{4.34}$$

The matrices $\overline{\boldsymbol{P}}_j$ and $\overline{\boldsymbol{P}}_j$ becomes steady within the estimation window and therefore their average value have not been considered.

$$\overline{\boldsymbol{P}}_{\boldsymbol{k}} - \hat{\boldsymbol{P}}_{\boldsymbol{k}} = \frac{1}{N}\sum_{j=j_0}^{k}\left[\boldsymbol{\eta}_j\boldsymbol{\eta}_j^T\right] \tag{4.35}$$

From (4.8) expression of $\overline{\boldsymbol{P}}_{\boldsymbol{k}}$ is substituted in (4.35)

$$\overline{\boldsymbol{P}}_{\boldsymbol{k}}^f + \hat{\boldsymbol{Q}}_k - \hat{\boldsymbol{P}}_{\boldsymbol{k}} = \frac{1}{L}\sum_{j=j_0}^{k}\left[\boldsymbol{\eta}_j\boldsymbol{\eta}_j^T\right] \tag{4.36}$$

$$\hat{\boldsymbol{Q}}_k = \frac{1}{L}\sum_{j=j_0}^{k}\left[\boldsymbol{\eta}_j\boldsymbol{\eta}_j^T\right] + \hat{\boldsymbol{P}}_{\boldsymbol{k}} - \overline{\boldsymbol{P}}_{\boldsymbol{k}}^f \tag{4.37}$$

During steady state the term $\left(\hat{\boldsymbol{P}}_{\boldsymbol{k}} - \overline{\boldsymbol{P}}_{\boldsymbol{k}}^f\right)$ becomes often low and may negligible as it is recommended in [Mohamed1999].

$$\hat{\boldsymbol{Q}}_k \approx \frac{1}{L}\sum_{j=j_0}^{k}\left[\boldsymbol{\eta}_j\boldsymbol{\eta}_j^T\right] \tag{4.38}$$

$$\hat{\boldsymbol{Q}}_k \approx \frac{1}{L}\sum_{j=j_0}^{k}\left[\boldsymbol{K}_j\boldsymbol{\vartheta}_j\boldsymbol{\vartheta}_j^T\boldsymbol{K}_j^T\right] \tag{4.39}$$

$$\hat{\boldsymbol{Q}}_k \approx \boldsymbol{K}_k\frac{1}{L}\sum_{j=j_0}^{k}\left[\boldsymbol{\vartheta}_j\boldsymbol{\vartheta}_j^T\right]\boldsymbol{K}_k^T \tag{4.40}$$

$$\hat{\boldsymbol{Q}}_k \approx \boldsymbol{K}_k\hat{\boldsymbol{C}}_{\vartheta_k}\boldsymbol{K}_k^T$$

Hence, (4.18) is proved. □

For parameter augmented state vector it may so happen that the noise covariance of the states is known while the noise covariance for the parameters remains unknown. Partial adaptation of $\boldsymbol{Q}$ is possible in this situation. For partial adaptation we use the corresponding elements of filter gain which are related with n_ζ parameters. We select the last n_ζ rows from $\boldsymbol{K}_{\boldsymbol{k}}$ matrix and define as $\boldsymbol{K}_{\boldsymbol{k}}^\zeta$. Now the noise covariance of the corresponding parameter is adapted as $\hat{\boldsymbol{Q}}_{\boldsymbol{k}}^\zeta = \boldsymbol{K}_{\boldsymbol{k}}^\zeta\hat{\boldsymbol{C}}_{\boldsymbol{k}}^\vartheta\left(\boldsymbol{K}_{\boldsymbol{k}}^\zeta\right)^T$. Note that (4.22) differentiated with respect to unknown elements only.

As the covariance of the process noise of state vector of augmented vector is known, the overall noise covariance of augmented vector is partially adapted and presented as

$\hat{\boldsymbol{Q}}_k = \begin{bmatrix} \boldsymbol{Q}_k^{\psi} & \boldsymbol{0} \\ \boldsymbol{0} & \hat{\boldsymbol{Q}}_k^{\zeta} \end{bmatrix}$ where $\boldsymbol{Q}_k^{\psi}$ is a known constant matrix which is the noise covariance of the state.

Covariance Matching method

Covariance matching method is an intuitive method where the window estimate of the innovation or, the residual covariance is compared with its respective theoretical value computed by the filter for obtaining the adapted value of noise covariance. When the process or, measurement noise covariance is accurately initialized the estimated innovation covariance from the sliding window becomes consistent with the theoretical value computed in the filtering algorithm. However, any discrepancy between these two innovation/residual covariances indicates improper tuning of filter. While tuning parameters, $\boldsymbol{Q}$ or $\boldsymbol{R}$ are unknown, they can be deduced based on the comparison of window estimated and the filter computed innovation or residual covariance. For adaptation of process noise covariance the expression of $\boldsymbol{Q}$ is obtained comparing the window estimate of innovation covariance and the filter computed (theoretical) innovation covariance at each time instant assuming $\boldsymbol{R}$ is known to the filter.

Theorem 4.3: Same as theorem 4.2

Proof:

The online estimate of innovation sequence from the sliding window can be expressed as

$$\hat{\boldsymbol{C}}_{\vartheta_k} = \frac{1}{L}\sum_{j=j_0}^{j} \boldsymbol{\vartheta}_j \boldsymbol{\vartheta}_j^T \tag{4.41}$$

The filter computed innovation covariance is approximately equal to the window estimated innovation covariance when $\boldsymbol{Q}$ is properly tuned. It is expected that with the adapted $\hat{\boldsymbol{Q}}_k$ these two covariances will be consistent. Therefore we may write

$$\hat{\boldsymbol{C}}_{\vartheta_k} = \boldsymbol{P}_k^y$$

$$\Rightarrow \boldsymbol{K}_k \hat{\boldsymbol{C}}_{\vartheta_k} \boldsymbol{K}_k^T = \boldsymbol{K}_k \boldsymbol{P}_k^y \boldsymbol{K}_k^T$$

From (4.6) the expression of *a posteriori* error covariance is obtained as

$$\hat{P}_k = \overline{P}_k - K_k P_k^y K_k^T$$

This expression can be alternatively written as

$$\overline{P}_k - \hat{P}_k = K_k \hat{C}_{\vartheta_k} K_k^T \tag{4.42}$$

Using the expression of *a priori* error covariance given by (4.8) in (4.42)

$$\overline{P}_k^f + \hat{Q}_k - \hat{P}_k = K_k \hat{C}_{\vartheta_k} K_k^T$$

$$\Rightarrow \hat{Q}_k = K_k \hat{C}_{\vartheta_k} K_k^T + \hat{P}_k - \overline{P}_k^f \tag{4.43}$$

The expression $\left(\hat{P}_k - \overline{P}_k^f\right)$ becomes often negligible. As the filter reaches steady state $\hat{P}_{k-1}$ beomes steady and often low for satisfactory estimation.

Therefore, the expression for adapted Q after approximation becomes

$$\hat{Q}_k = K_k \hat{C}_{\vartheta_k} K_k^T$$

Hence, (4.18) is proved.

Theorem 4.4

For the adaptive nonlinear filters with unknown process noise covariance, Q can also be adapted with the help of a scaling factor and expressed as

$$\hat{Q}_k = \lambda_k \hat{Q}_{k-1}$$

Where, the scaling factor, $$\lambda_k = \sqrt{\frac{trace\left(\hat{C}_{\vartheta_k} - R\right)}{trace\left(P_k^y - R\right)}} \tag{4.45}$$

Where $\hat{C}_{\vartheta_k}$ is estimated innovation covariance obtained from the sliding window with window length L and P_k^y is the innovation covariance computed in the filtering algorithm.

Proof:

According to the covariance matching method the window estimated innovation covariance should match with the innovation covariance computed by the filtering algorithm. When measurement noise covariance is known, i.e., $\overline{R} = R$ it can be subtracted from innovation covariance and the remaining part is same as P_k^h which error covariance of *a priori* estimate of measurement. $\boldsymbol{P}_k^h$ is dependent on the *a priori* error covariance which subsequently depends on process noise covariance.

As it is considered that the process inside the window is in steady state the *a priori* error covariance is solely dependent of process noise covariance $\boldsymbol{Q}$ as *a posteriori* error covariance often converges to a small value and does not have significant contribution. The concept which was first introduced in the work of [Ding2007] for linear signal model has been extended here for nonlinear systems

With these concepts in mind process noise covariance can be adapted with help of a scaling factor which is empirically decided as $\hat{\boldsymbol{Q}}_k = \lambda_k \hat{\boldsymbol{Q}}_{k-1}$

$$\lambda_k = \sqrt{\frac{trace\left(\hat{\boldsymbol{C}}_{\vartheta_k} - \boldsymbol{R}\right)}{trace\left(\boldsymbol{P}_k^y - \boldsymbol{R}\right)}}$$

The square root has been considered to smooth out the changes in the scaling factor as recommended in [Ding2007]

Maximum a Posterior (MAP) based method

Theorem 4.4

For window length L and with the consideration that the noise statistics are constant within the window the process noise covariance can be expressed as:

$$\hat{\boldsymbol{Q}}_k = \frac{1}{k}\sum_{j=1}^{k}\left[\hat{\boldsymbol{P}}_j + \boldsymbol{K}_j \vartheta_j \vartheta_j^T \boldsymbol{K}_j^T - \overline{\boldsymbol{P}}_j^f\right] \tag{4.46}$$

Proof:

The algorithm for adapted process noise covariance is derived using the concept of Maximum a Posterior (MAP) estimation method provided in [Cheng2014, Gao2015b, Liqiang2015]. Usually both the mean and covariance of noise is estimated with the help of MAP noise statistic estimators reported in the above papers. In this case as zero mean Gaussian noises are considered. Therefore, estimation the mean of noise is not necessary.

The conditional distribution of interest based on measurements is expressed as

$$\boldsymbol{J}' = p\left[\boldsymbol{x}_k, \boldsymbol{Q}, \boldsymbol{R} \middle| \boldsymbol{Y}_k\right]$$

As $p[\boldsymbol{Y_k}]$ is not directly related to other parameters except the estimate of state, $\boldsymbol{x_k}$, the conditional distribution is expressed as

$$\boldsymbol{J}' = p[\boldsymbol{x_k}, \boldsymbol{Q}, \boldsymbol{R} | \boldsymbol{Y_k}]$$

$$\boldsymbol{J}' = \frac{p[\boldsymbol{x_k}, \boldsymbol{Q}, \boldsymbol{R}, \boldsymbol{Y_k}]}{p[\boldsymbol{Y_k}]}$$

For this optimization problem it seems that $p[\boldsymbol{Y_k}]$ is not related. Therefore, the objective is changed to compute the maximum of the following unconditional density function

$$\boldsymbol{J} = p[\boldsymbol{x_k}, \boldsymbol{Q}, \boldsymbol{R}, \boldsymbol{Y_k}]$$

$$\Rightarrow \boldsymbol{J} = p[\boldsymbol{x_k} | \boldsymbol{Q}, \boldsymbol{R}]\, p[\boldsymbol{Y_k} | \boldsymbol{x_k}, \boldsymbol{Q}, \boldsymbol{R}]\, p[\boldsymbol{Q}, \boldsymbol{R}] \qquad (4.47)$$

Note that among the conditional parameters present in (4.47) we have not considered the means of noise as the noises are zero mean. Here $p[\boldsymbol{Q}, \boldsymbol{R}]$ may be considered as a constant which signifies the prior information on noise statistic.

According to Gaussian distribution the condition distribution $p[\boldsymbol{x_k} | \boldsymbol{Q}, \boldsymbol{R}]$ could be expressed as:

$$p[\boldsymbol{x_k} | \boldsymbol{Q}, \boldsymbol{R}] = p[\boldsymbol{x_0}] \prod_{j=1}^{k} p[\boldsymbol{x_k} | \boldsymbol{Q}]$$

$$\Rightarrow p[\boldsymbol{x_k} | \boldsymbol{Q}, \boldsymbol{R}] =$$

$$\frac{1}{2\pi^{n/2} |\boldsymbol{P_0}|^{1/2}} \exp\left\{ -\frac{1}{2} (\boldsymbol{x_0} - \hat{\boldsymbol{x}}_0)^T \boldsymbol{P}_0^{-1} (\boldsymbol{x_0} - \hat{\boldsymbol{x}}_0) \right\}$$

$$\times \left[\prod_{j=1}^{k} \frac{1}{2\pi^{n/2} |\boldsymbol{Q}|^{1/2}} \exp\left\{ -\frac{1}{2} (\boldsymbol{x}_j - \boldsymbol{f}(\boldsymbol{x}_{j-1}))^T \boldsymbol{Q}^{-1} (\boldsymbol{x}_j - \boldsymbol{f}(\boldsymbol{x}_{j-1})) \right\} \right]$$

$$\Rightarrow p[\boldsymbol{x_k} | \boldsymbol{Q}, \boldsymbol{R}] = C_1 |\boldsymbol{P_0}|^{-1/2} |\boldsymbol{Q}|^{-n/2} \times \exp\left\{ -\frac{1}{2} \|\boldsymbol{x_0} - \hat{\boldsymbol{x}}_0\|^2_{\boldsymbol{P}_0^{-1}} + \sum_{j=1}^{k} \|\boldsymbol{x}_j - \boldsymbol{f}(\boldsymbol{x}_{j-1})\|^2_{\boldsymbol{Q}^{-1}} \right\} \qquad (4.48)$$

The notation $\|\boldsymbol{u}\|^2_{\boldsymbol{A}}$ signifies: $\|\boldsymbol{u}\|^2_{\boldsymbol{A}} = \boldsymbol{u}^T \boldsymbol{A} \boldsymbol{u}$

Assuming that measurements are known for *j=1,…,k* and unrelated to one another $p[\boldsymbol{Y_k} | \boldsymbol{x_k}, \boldsymbol{Q}, \boldsymbol{R}]$ can be expressed as

$$p\left[\boldsymbol{Y_k} \middle| \boldsymbol{x_k}, \boldsymbol{Q}, \boldsymbol{R}\right] = \prod_{j=1}^{k} p\left[\boldsymbol{y}_j \middle| \boldsymbol{x_k}, \boldsymbol{R}\right]$$

$$\Rightarrow p\left[\boldsymbol{Y_k} \middle| \boldsymbol{x_k}, \boldsymbol{Q}, \boldsymbol{R}\right] = \prod_{j=1}^{k} \frac{1}{(2\pi)^{m/2} |\boldsymbol{R}|^{1/2}} \exp\left\{ -\frac{1}{2} \left\| \boldsymbol{y}_j - \boldsymbol{g}(\boldsymbol{x}_j) \right\|_{\boldsymbol{R}^{-1}}^2 \right\}$$

$$\Rightarrow p\left[\boldsymbol{Y_k} \middle| \boldsymbol{x_k}, \boldsymbol{Q}, \boldsymbol{R}\right] = C_2 |\boldsymbol{R}|^{-k/2} \exp\left\{ -\frac{1}{2} \sum_{j=1}^{k} \left\| \boldsymbol{y}_j - \boldsymbol{g}(\boldsymbol{x}_j) \right\|_{\boldsymbol{R}^{-1}}^2 \right\} \tag{4.49}$$

Therefore the conditional distribution given by (4.47) can now be expressed using (4.48) and (4.49) as

$$\Rightarrow \boldsymbol{J} = C |\boldsymbol{Q}|^{-k/2} |\boldsymbol{R}|^{-k/2} \exp\left\{ -\frac{1}{2} \left[\sum_{j=1}^{k} \left\| \boldsymbol{x}_j - \boldsymbol{f}(\boldsymbol{x}_{j-1}) \right\|_{\boldsymbol{Q}^{-1}}^2 + \sum_{j=1}^{k} \left\| \boldsymbol{y}_j - \boldsymbol{g}(\boldsymbol{x}_j) \right\|_{\boldsymbol{R}^{-1}}^2 \right] \right\} \tag{4.50}$$

Where, the constant, $C = \exp\left\{ -\frac{1}{2} \left\| \boldsymbol{x_0} - \hat{\boldsymbol{x}}_{\boldsymbol{0}} \right\|_{\boldsymbol{P}_0^{-1}}^2 \right\} C_1 C_2 |\boldsymbol{P_0}|^{-1/2} p[\boldsymbol{Q}, \boldsymbol{R}]$

Taking logarithm on both sides of (4.50)

$$\ln \boldsymbol{J} = -\frac{k}{2} \ln|\boldsymbol{Q}| - \frac{k}{2} \ln|\boldsymbol{R}| + \ln|C| - \frac{1}{2} \sum_{j=1}^{k} \left\| \boldsymbol{x}_j - \boldsymbol{f}(\boldsymbol{x}_{j-1}) \right\|_{\boldsymbol{Q}^{-1}}^2 - \frac{1}{2} \sum_{j=1}^{k} \left\| \boldsymbol{y}_j - \boldsymbol{g}(\boldsymbol{x}_j) \right\|_{\boldsymbol{R}^{-1}}^2$$

For maximization of conditional density

$$\left. \frac{\partial \ln \boldsymbol{J}}{\partial \boldsymbol{Q}} \right|_{\boldsymbol{Q} = \hat{\boldsymbol{Q}}_k}^{x_{j-1} = \hat{x}_{j-1|k},\, x_j = \hat{x}_{j|k}} = 0$$

After differentiation of (4.50) with respect to $\boldsymbol{Q}$ after taking logarithm the expression of adapted $\hat{\boldsymbol{Q}}_k$ is obtained as

$$\hat{\boldsymbol{Q}}_k = \frac{1}{k} \sum_{j=1}^{k} \left\{ \left(\hat{\boldsymbol{x}}_{j|k} - \boldsymbol{f}\left(\hat{\boldsymbol{x}}_{j-1|k} \right) \right) \left(\hat{\boldsymbol{x}}_{j|k} - \boldsymbol{f}\left(\hat{\boldsymbol{x}}_{j-1|k} \right) \right)^T \right\} \tag{4.51}$$

Note that an assumption is made where $\hat{\boldsymbol{x}}_{\boldsymbol{j-1|k}}, \hat{\boldsymbol{x}}_{\boldsymbol{j|k}}$ are known. Moreover, in the expression of adapted $\boldsymbol{Q}$ the smoothed estimate $\hat{\boldsymbol{x}}_{j-1|k}$ and $\hat{\boldsymbol{x}}_{j|k}$ can be replaced by a posterori estimate $\hat{\boldsymbol{x}}_{j-1}$ and $\hat{\boldsymbol{x}}_j$ as calculation with smoothed estimate increases complexity. With this consideration (4.51) can be expressed as

$$\hat{\boldsymbol{Q}}_k = \frac{1}{k}\sum_{j=1}^{k}\left\{\left(\hat{\boldsymbol{x}}_j - \boldsymbol{f}\left(\hat{\boldsymbol{x}}_{j-1}\right)\right)\left(\hat{\boldsymbol{x}}_j - \boldsymbol{f}\left(\hat{\boldsymbol{x}}_{j-1}\right)\right)^T\right\} \tag{4.52}$$

However, in [Gao2015b] it is proved that the above expression yields a biased estimate of $\hat{\boldsymbol{Q}}_k$. The suboptimal estimation algorithm for $\hat{\boldsymbol{Q}}_k$ following [Gao2015b] can be obtained as given below.

$$\frac{1}{k}\sum_{j=1}^{k}\left\{\left(\hat{\boldsymbol{x}}_j - \boldsymbol{f}\left(\hat{\boldsymbol{x}}_{j-1}\right)\right)\left(\hat{\boldsymbol{x}}_j - \boldsymbol{f}\left(\hat{\boldsymbol{x}}_{j-1}\right)\right)^T\right\} = \frac{1}{k}\sum_{j=1}^{k}\left\{\boldsymbol{K}_j \vartheta_j \vartheta_j^{\boldsymbol{T}} \boldsymbol{K}_j^{\boldsymbol{T}}\right\} \tag{4.53}$$

$$\Rightarrow \frac{1}{k}\sum_{j=1}^{k}\left\{\boldsymbol{K}_j \vartheta_j \vartheta_j^{\boldsymbol{T}} \boldsymbol{K}_j^{\boldsymbol{T}}\right\} = \frac{1}{k}\sum_{j=1}^{k}\left\{\overline{\boldsymbol{P}}_j - \hat{\boldsymbol{P}}_j\right\} \tag{4.54}$$

$$\Rightarrow \frac{1}{k}\sum_{j=1}^{k}\left\{\boldsymbol{K}_j \vartheta_j \vartheta_j^{\boldsymbol{T}} \boldsymbol{K}_j^{\boldsymbol{T}}\right\} = \frac{1}{k}\sum_{j=1}^{k}\left\{\overline{\boldsymbol{P}}_j^f + \boldsymbol{Q} - \hat{\boldsymbol{P}}_j\right\} \tag{4.55}$$

$$\Rightarrow \hat{\boldsymbol{Q}}_{\boldsymbol{k}} = \frac{1}{k}\sum_{j=1}^{k}\left\{\boldsymbol{K}_j \vartheta_j \vartheta_j^{\boldsymbol{T}} \boldsymbol{K}_j^{\boldsymbol{T}} + \hat{\boldsymbol{P}}_j - \overline{\boldsymbol{P}}_j^f\right\}$$

Hence, (4.46) is proved. □

It has been discussed before that the filter gains and error covariance are often considered to be steady inside the window (specifically when the filter reaches steady state). Therefore, expression (4.46) can be modified as

$$\Rightarrow \hat{\boldsymbol{Q}}_{\boldsymbol{k}} = \boldsymbol{K}_{\boldsymbol{k}}\left(\frac{1}{k}\sum_{j=1}^{k}\vartheta_j \vartheta_j^{\boldsymbol{T}}\right)\boldsymbol{K}_{\boldsymbol{k}}^{\boldsymbol{T}} + \hat{\boldsymbol{P}}_{\boldsymbol{k}} - \overline{\boldsymbol{P}}_{\boldsymbol{k}}^{f} \tag{4.56}$$

As mentioned earlier, the contribution of the expression $\left(\hat{\boldsymbol{P}}_{\boldsymbol{k}} - \overline{\boldsymbol{P}}_{\boldsymbol{k}}^{f}\right)$ is often low and therefore can be neglected and (4.56) can be approximately written as

$$\hat{\boldsymbol{Q}}_{\boldsymbol{k}} \approx \boldsymbol{K}_{\boldsymbol{k}} \hat{\boldsymbol{C}}_{\vartheta_{\boldsymbol{k}}} \boldsymbol{K}_{\boldsymbol{k}}^{\boldsymbol{T}} \tag{4.57}$$

Therefore, the same expression of adapted $\boldsymbol{Q}$ given by (4.18) can also be obtained from (4.46).

4.2.3.2. Adaptation of the Measurement Noise Covariance (*R*)

Maximum Likelihood Estimation (MLE) based method

Theorem 4.5.1:

For the nonlinear filters with unknown measurement noise covariance ***R***, the adapted ***R*** at current instant is expressed as

$$\hat{\boldsymbol{R}}_k = \hat{\boldsymbol{C}}_k^{\boldsymbol{\rho}} + \hat{\boldsymbol{P}}_k^{\boldsymbol{g}} \tag{4.58}$$

Where $\hat{\boldsymbol{C}}_k^{\boldsymbol{\rho}}$ is estimated residual covariance obtained from the sliding window with length *L*.

Proof:

The ***R*** adaptation algorithm used in the proposed algorithm is derived using MLE technique with assumption 4.4. Derivation of adaptation algorithms for nonlinear systems are inspired from Maximum Likelihood based ***R*** adaptation approach for linear signal models [Mohamed1999]. The probability density function of the measurements conditioned on adaptive parameter, $\boldsymbol{\alpha}$ at specific epoch k is chosen based on residual sequence $\boldsymbol{\rho_k}$ unlike [Mohamed1999, Maybeck1982]

$$\boldsymbol{P}_{(y|\alpha)_k} = \frac{1}{\sqrt{(2\pi)^m \left|\boldsymbol{C}_k^{\boldsymbol{\rho}}\right|}} \exp\left(-\tfrac{1}{2}\boldsymbol{\rho}_k^{\boldsymbol{T}} \left(\boldsymbol{C}_k^{\boldsymbol{\rho}}\right)^{-1} \boldsymbol{\rho}_k\right) \tag{4.59}$$

$$\text{or, } \ln\left(\boldsymbol{P}_{(y|\alpha)_k}\right) = -\frac{1}{2}\left\{m\ln(2\pi) + \ln\left|\boldsymbol{C}_k^{\boldsymbol{\rho}}\right| + \boldsymbol{\rho}_k^{\boldsymbol{T}} \left(\boldsymbol{C}_k^{\boldsymbol{\rho}}\right)^{-1} \boldsymbol{\rho}_k\right\} \tag{4.60}$$

Multiplying both sides with -2 and neglecting the constant term we get

$$\boldsymbol{P} = \ln\left|\boldsymbol{C}_k^{\boldsymbol{\rho}}\right| + \boldsymbol{\rho}_k^{\boldsymbol{T}} \left(\boldsymbol{C}_k^{\boldsymbol{\rho}}\right)^{-1} \boldsymbol{\rho}_k \tag{4.61}$$

Residual sequence has been considered inside a window size L as the filter uses a fixed length memory. The residuals inside the window will be summed. Therefore, the Maximum Likelihood condition becomes:

$$\min\left\{\sum_{j=j_0}^{k}\left(\ln\left|\boldsymbol{C}_j^{\boldsymbol{\rho}}\right| + \boldsymbol{\rho}_j^{\boldsymbol{T}} \left(\boldsymbol{C}_j^{\boldsymbol{\rho}}\right)^{-1} \boldsymbol{\rho}_j\right)\right\} \tag{4.62}$$

Which results in

$$\sum_{j=j_0}^{k}\left[tr\left\{\left(\boldsymbol{C}_j^{\boldsymbol{\rho}}\right)^{-1}\left(\frac{\partial \boldsymbol{C}_j^{\boldsymbol{\rho}}}{\partial \boldsymbol{\alpha}_k}\right)\right\} - \boldsymbol{\rho}_j^{\boldsymbol{T}}\left(\boldsymbol{C}_j^{\boldsymbol{\rho}}\right)^{-1}\left(\frac{\partial \boldsymbol{C}_j^{\boldsymbol{\rho}}}{\partial \boldsymbol{\alpha}_k}\right)\left(\boldsymbol{C}_j^{\boldsymbol{\rho}}\right)^{-1}\boldsymbol{\rho}_j\right] = 0 \tag{4.63}$$

$$\sum_{j=j_0}^{k}\left[tr\left\{\left(\boldsymbol{C}_j^{\boldsymbol{\rho}}\right)^{-1}\left(\frac{\partial \boldsymbol{C}_j^{\boldsymbol{\rho}}}{\partial \boldsymbol{\alpha}_k}\right)\right\}-tr\left\{\left(\boldsymbol{C}_j^{\boldsymbol{\rho}}\right)^{-1}\boldsymbol{\rho}_j\boldsymbol{\rho}_j^{\boldsymbol{T}}\left(\boldsymbol{C}_j^{\boldsymbol{\rho}}\right)^{-1}\left(\frac{\partial \boldsymbol{C}_j^{\boldsymbol{\rho}}}{\partial \boldsymbol{\alpha}_k}\right)\right\}\right]=0 \tag{4.64}$$

$$\sum_{j=j_0}^{k}\left[tr\left\{\left[\left(\boldsymbol{C}_j^{\boldsymbol{\rho}}\right)^{-1}-\left(\boldsymbol{C}_j^{\boldsymbol{\rho}}\right)^{-1}\boldsymbol{\rho}_j\boldsymbol{\rho}_j^{\boldsymbol{T}}\left(\boldsymbol{C}_j^{\boldsymbol{\rho}}\right)^{-1}\right]\left(\frac{\partial \boldsymbol{C}_j^{\boldsymbol{\rho}}}{\partial \boldsymbol{\alpha}_k}\right)\right\}\right]=0 \tag{4.65}$$

The formulae necessary for matrix operation have been mentioned before.

The deduction of the relation between residual covariance, $\boldsymbol{C}_k^{\boldsymbol{\rho}}$ and the measurement noise covariance again necessitates the pseudo measurement matrix of the nonlinear measurement equation. The pseudo measurement matrix has been defined before in (4.26) as

$\boldsymbol{\Psi}_k=\left(\boldsymbol{P}_k^{xz}\right)^T\left(\hat{\boldsymbol{P}}_k\right)^{-1}$. Using the pseudo measurement matrix the residual covariance can be approximately represented as: $\boldsymbol{C}_k^{\boldsymbol{\rho}}\approx\boldsymbol{R}_k-\boldsymbol{\Psi}_k\hat{\boldsymbol{P}}_k\boldsymbol{\Psi}_k^{\boldsymbol{T}}$ which is derived analogously to that of linear signal model as the pseudo measurement matrix is now available. Readers are requested to consult the proof of theorem 4.6 for more details.

For adaptation of $\boldsymbol{R}$, the adaptive parameter $\boldsymbol{\alpha}$ is chosen as $\boldsymbol{\alpha}_i=\boldsymbol{R}_{ii}$. Hence,

$$\frac{\partial \boldsymbol{C}_k^{\boldsymbol{\rho}}}{\partial \boldsymbol{R}_{kk}}=\boldsymbol{I}-\boldsymbol{\Psi}_k\frac{\partial \hat{\boldsymbol{P}}_k}{\partial \boldsymbol{R}_{kk}}\boldsymbol{\Psi}_k^{\boldsymbol{T}} \tag{4.66}$$

$$\frac{\partial \hat{\boldsymbol{P}}_k}{\partial \boldsymbol{R}_{kk}}=\frac{\partial}{\partial \boldsymbol{R}_{kk}}\left[\left(\boldsymbol{I}-\boldsymbol{K}_k\boldsymbol{\Psi}_k\right)\overline{\boldsymbol{P}}_k\right] \tag{4.67}$$

$$\frac{\partial \hat{\boldsymbol{P}}_k}{\partial \boldsymbol{R}_{kk}}=\frac{\partial}{\partial \boldsymbol{R}_{kk}}\left[\overline{\boldsymbol{P}}_k-\boldsymbol{K}_k\boldsymbol{\Psi}_k\overline{\boldsymbol{P}}_k\right] \tag{4.68}$$

$$\frac{\partial \hat{\boldsymbol{P}}_k}{\partial \boldsymbol{R}_{kk}}=-\frac{\partial}{\partial \boldsymbol{R}_{kk}}\left[\overline{\boldsymbol{P}}_k\boldsymbol{\Psi}_k^T\left(\boldsymbol{\Psi}_k\overline{\boldsymbol{P}}_k\boldsymbol{\Psi}_k^T+\boldsymbol{R}_k\right)^{-1}\boldsymbol{\Psi}_k\overline{\boldsymbol{P}}_k\right] \tag{4.69}$$

$$\frac{\partial \hat{\boldsymbol{P}}_k}{\partial \boldsymbol{R}_{kk}}=-\boldsymbol{K}_k\boldsymbol{K}_k^T \tag{4.70}$$

Using the expression of $\frac{\partial \hat{\boldsymbol{P}}_k}{\partial \boldsymbol{R}_{kk}}$ in equation (4.66) we get

$$\frac{\partial \boldsymbol{C}_k^{\boldsymbol{\rho}}}{\partial \boldsymbol{R}_{kk}}=\boldsymbol{I}+\boldsymbol{\Psi}_k\boldsymbol{K}_k\boldsymbol{K}_k^{\boldsymbol{T}}\boldsymbol{\Psi}_k^{\boldsymbol{T}} \tag{4.71}$$

Substituting this value in the ML equation given by (4.65) we get

$$\sum_{j=j_0}^{k}\left[tr\left\{\left[\left(\boldsymbol{C}_j^{\rho}\right)^{-1}-\left(\boldsymbol{C}_j^{\rho}\right)^{-1}\boldsymbol{\rho}_j\boldsymbol{\rho}_j^{T}\left(\boldsymbol{C}_j^{\rho}\right)^{-1}\right]\left(\boldsymbol{I}+\boldsymbol{\Psi}_j\boldsymbol{K}_j\boldsymbol{K}_j^{T}\boldsymbol{\Psi}_j^{T}\right)\right\}\right]=0 \tag{4.72}$$

Alternatively,

$$\sum_{j=j_0}^{k}\left[tr\left\{\left(\boldsymbol{C}_j^{\rho}\right)^{-1}\left[\boldsymbol{C}_j^{\rho}-\boldsymbol{\rho}_j\boldsymbol{\rho}_j^{T}\right]\left(\boldsymbol{C}_j^{\rho}\right)^{-1}\left(\boldsymbol{I}+\boldsymbol{\Psi}_j\boldsymbol{K}_j\boldsymbol{K}_j^{T}\boldsymbol{\Psi}_j^{T}\right)\right\}\right]=0 \tag{4.73}$$

The above equation holds only when the expression within the square bracket vanishes as the residual covariance $\boldsymbol{C}_j^{\rho}$ and the expression $\left(\boldsymbol{I}+\boldsymbol{\Psi}_j\boldsymbol{K}_j\boldsymbol{K}_j^{T}\boldsymbol{\Psi}_j^{T}\right)$ are positive definite.

Assuming ergodic residual sequence inside the window the expression of estimated covariance of residual sequence becomes

$$\hat{\boldsymbol{C}}_k^{\rho}=\frac{1}{L}\sum_{j=j_0}^{k}\boldsymbol{\rho}_j\boldsymbol{\rho}_j^{T} \tag{4.74}$$

$$\hat{\boldsymbol{R}}_k-\boldsymbol{\Psi}_k\hat{\boldsymbol{P}}_k\boldsymbol{\Psi}_k^{T}=\frac{1}{L}\sum_{j=j_0}^{k}\boldsymbol{\rho}_j\boldsymbol{\rho}_j^{T} \tag{4.75}$$

Or, $$\hat{\boldsymbol{R}}_k=\frac{1}{L}\sum_{j=j_0}^{k}\boldsymbol{\rho}_j\boldsymbol{\rho}_j^{T}+\boldsymbol{\Psi}_k\hat{\boldsymbol{P}}_k\boldsymbol{\Psi}_k^{T} \tag{4.76}$$

The term $\boldsymbol{\Psi}_k\hat{\boldsymbol{P}}_k\boldsymbol{\Psi}_k^{T}$ is approximately equal to $\hat{\boldsymbol{P}}_k^{g}$. With the above consideration and using (4.74) and (4.76) the expression of the adapted $\boldsymbol{R}$ becomes

$$\hat{\boldsymbol{R}}_k=\hat{\boldsymbol{C}}_k^{\rho}+\hat{\boldsymbol{P}}_k^{g}$$

Hence, (4.58) is proved. □

Theorem 4.5.2:

For the nonlinear filters with unknown measurement noise covariance $\boldsymbol{R}$, the adapted $\boldsymbol{R}$ using the innovation sequence can be derived as

$$\hat{\boldsymbol{R}}_k=\hat{\boldsymbol{C}}_{\vartheta_k}-\overline{\boldsymbol{P}}_k^{g} \tag{4.77}$$

Where $\hat{\boldsymbol{C}}_{\vartheta_k}$ is estimated innovation covariance obtained from the sliding window with window length L.

Proof:

The adaptation algorithm for $\boldsymbol{R}$ can also be obtained using innovation sequence. The probability density function of the measurements conditioned on adaptive parameter, α at specific epoch k is chosen based on innovation sequence as given in maximum likelihood based $\boldsymbol{Q}$ adaptation method. Detailed steps are not shown to avoid repetition.

The Maximum Likelihood condition given by (4.25) is presented below:

$$\sum_{j=j_0}^{k}\left[tr\left\{\left[\boldsymbol{C}_{\vartheta_j}^{-1}-\boldsymbol{C}_{\vartheta_j}^{-1}\boldsymbol{\vartheta_j}\boldsymbol{\vartheta_j^T}\boldsymbol{C}_{\vartheta_j}^{-1}\right]\left(\frac{\partial \boldsymbol{C}_{\vartheta_j}}{\partial \boldsymbol{\alpha_k}}\right)\right\}\right]=0$$

$\boldsymbol{\alpha_k}$ is considered as $\boldsymbol{\alpha_k}=\boldsymbol{R_{kk}}$ for adaptation of $\boldsymbol{R}$. The innovation covariance can be expressed in terms of pseudo measurement matrix as $\boldsymbol{C}_{\vartheta_j}=\boldsymbol{R}_j+\boldsymbol{\Psi}_j\overline{\boldsymbol{P}}_j\boldsymbol{\Psi}_j^{\boldsymbol{T}}$

Therefore, $\left(\frac{\partial \boldsymbol{C}_{\vartheta_j}}{\partial \boldsymbol{\alpha_k}}\right)=\left(\frac{\partial \boldsymbol{C}_{\vartheta_j}}{\partial R_{kk}}\right)=\boldsymbol{I}$

The ML condition now becomes

$$\sum_{j=j_0}^{k}\left[tr\left\{\left[\boldsymbol{C}_{\vartheta_j}^{-1}-\boldsymbol{C}_{\vartheta_j}^{-1}\boldsymbol{\vartheta_j}\boldsymbol{\vartheta_j^T}\boldsymbol{C}_{\vartheta_j}^{-1}\right]\right\}\right]=0 \qquad (4.78)$$

$$\Rightarrow\sum_{j=j_0}^{k}\left[tr\left\{\left[\boldsymbol{C}_{\vartheta_j}^{-1}-\boldsymbol{C}_{\vartheta_j}^{-1}\boldsymbol{\vartheta_j}\boldsymbol{\vartheta_j^T}\boldsymbol{C}_{\vartheta_j}^{-1}\right]\right\}\right]=0 \qquad (4.79)$$

$$\Rightarrow\sum_{j=j_0}^{k}\left[tr\left\{\left[\boldsymbol{C}_{\vartheta_j}^{-1}\left(\boldsymbol{C}_{\vartheta_j}-\boldsymbol{\vartheta_j}\boldsymbol{\vartheta_j^T}\right)\boldsymbol{C}_{\vartheta_j}^{-1}\right]\right\}\right]=0 \qquad (4.80)$$

As $\boldsymbol{C}_{\vartheta_j}$ is non-singular the above condition holds true only when

$$\Rightarrow\sum_{j=j_0}^{k}\left[tr\left\{\left(\boldsymbol{C}_{\vartheta_j}-\boldsymbol{\vartheta_j}\boldsymbol{\vartheta_j^T}\right)\right\}\right]=0 \qquad (4.81)$$

$$\Rightarrow\sum_{j=j_0}^{k}\left[tr\left\{\left(\boldsymbol{R}_j+\boldsymbol{\Psi}_j\overline{\boldsymbol{P}}_j\boldsymbol{\Psi}_j^{\boldsymbol{T}}-\boldsymbol{\vartheta_j}\boldsymbol{\vartheta_j^T}\right)\right\}\right]=0 \qquad (4.82)$$

$$\Rightarrow\hat{\boldsymbol{R}}_k=\frac{1}{L}\sum_{j=j_0}^{k}\boldsymbol{\vartheta_j}\boldsymbol{\vartheta_j^T}-\boldsymbol{\Psi}_j\overline{\boldsymbol{P}}_j\boldsymbol{\Psi}_j^{\boldsymbol{T}} \qquad (4.83)$$

$$\Rightarrow \hat{\boldsymbol{R}}_k = \hat{\boldsymbol{C}}_{\vartheta_k} - \overline{\boldsymbol{P}}_{\boldsymbol{k}}^{\boldsymbol{g}}$$

Hence, (4.77) is proved. □

Covariance Matching method

Theorem 4.6: Same as theorem 4.5.1 for residual based ***R*** adaptation and 4.5.2 for innovation based ***R*** adaptation

Proof:

The adaptation algorithm for measurement noise covariance can also be obtained with the help of covariance matching method. The concept of the intuition based covariance matching technique has been discussed in ***Q*** adaptation method. In the same vein the adaptation algorithm for ***R*** is derived using both innovation and residual. Online estimate of innovation is obtained from the sliding window as given by (4.41)

$$\hat{\boldsymbol{C}}_{\vartheta_k} = \frac{1}{L} \sum_{j=k-L+1}^{k} \vartheta_j \vartheta_j^{\boldsymbol{T}}$$

The theoretical innovation covariance computed in filtering step is $\boldsymbol{C}_{\vartheta_k} = \boldsymbol{R}_{\boldsymbol{k}} + \overline{\boldsymbol{P}}_{\boldsymbol{k}}^{\boldsymbol{g}}$

With the correct value of adapted ***R*** these two innovation covariances should be consistent. Hence the adapted ***R*** can be expressed as

$$\hat{\boldsymbol{R}}_{\boldsymbol{k}} = \hat{\boldsymbol{C}}_{\vartheta_k} - \overline{\boldsymbol{P}}_{\boldsymbol{k}}^{\boldsymbol{g}}.$$

Note that the above expression of adapted ***R*** is same as (4.77)

When the statistics of residual instead of innovation is considered the estimate of residual covariance can be obtained as given by (4.74)

$$\hat{\boldsymbol{C}}_{\boldsymbol{k}}^{\boldsymbol{\rho}} = \frac{1}{L} \sum_{j=k-L+1}^{k} \boldsymbol{\rho}_j \boldsymbol{\rho}_j^{\boldsymbol{T}}$$

Ideally the residual covariance is given by $\boldsymbol{C}_{\boldsymbol{k}}^{\boldsymbol{\rho}} \approx \boldsymbol{R}_{\boldsymbol{k}} - \hat{\boldsymbol{P}}_{\boldsymbol{k}}^{\boldsymbol{g}}$ (4.84)

With the help of the pseudo measurement matrix the residual covariance can be represented by (4.85) and derived in the following steps:

$$\boldsymbol{C}_{\boldsymbol{k}}^{\boldsymbol{\rho}} \approx \hat{\boldsymbol{R}}_{\boldsymbol{k}} - \boldsymbol{\Psi}_{\boldsymbol{k}} \hat{\boldsymbol{P}}_{\boldsymbol{k}} \boldsymbol{\Psi}_{\boldsymbol{k}}^{\boldsymbol{T}} \tag{4.85}$$

where $\hat{\boldsymbol{P}}_{\boldsymbol{k}}^{\boldsymbol{g}} \approx \boldsymbol{\Psi}_{\boldsymbol{k}} \boldsymbol{P}_{\boldsymbol{k}}^{+} \boldsymbol{\Psi}_{\boldsymbol{k}}^{\boldsymbol{T}}$

The residual, $\boldsymbol{\rho}_k$ can be expressed as

$$\boldsymbol{\rho}_k = \boldsymbol{y}_k - \boldsymbol{g}(\hat{\boldsymbol{x}}_k) \tag{4.86}$$

$$\Rightarrow \boldsymbol{\rho}_k = \boldsymbol{g}(\boldsymbol{x}_k) + \boldsymbol{v}_k - \boldsymbol{g}(\hat{\boldsymbol{x}}_k) \tag{4.87}$$

Alternatively with the help of the pseudo measurement matrix (4.87) can be expressed as

$$\boldsymbol{\rho}_k = \boldsymbol{\Psi}_k(\boldsymbol{x}_k) + \boldsymbol{v}_k - \boldsymbol{\Psi}_k(\hat{\boldsymbol{x}}_k) \tag{4.88}$$

$$\Rightarrow \boldsymbol{\rho}_k = \boldsymbol{\Psi}_k(\boldsymbol{x}_k) + \boldsymbol{v}_k - \boldsymbol{\Psi}_k(\bar{\boldsymbol{x}}_k + \boldsymbol{K}_k \boldsymbol{\vartheta}_k)$$

$$\Rightarrow \boldsymbol{\rho}_k = (\boldsymbol{I} - \boldsymbol{\Psi}_k \boldsymbol{K}_k)\boldsymbol{\vartheta}_k$$

$$\Rightarrow \boldsymbol{\rho}_k = (\boldsymbol{I} - \boldsymbol{\Psi}_k \boldsymbol{K}_k)\boldsymbol{\Psi}_k \tilde{\boldsymbol{x}}_k + (\boldsymbol{I} - \boldsymbol{\Psi}_k \boldsymbol{K}_k)\boldsymbol{v}_k \tag{4.89}$$

The covariance of $\boldsymbol{\rho}_k$ is expressed as

$$\Rightarrow E(\boldsymbol{\rho}_k \boldsymbol{\rho}_k^T) = (\boldsymbol{I} - \boldsymbol{\Psi}_k \boldsymbol{K}_k)\boldsymbol{\Psi}_k E(\tilde{\boldsymbol{x}}_k \tilde{\boldsymbol{x}}_k^T)\boldsymbol{\Psi}_k^T (\boldsymbol{I} - \boldsymbol{\Psi}_k \boldsymbol{K}_k)^T + (\boldsymbol{I} - \boldsymbol{\Psi}_k \boldsymbol{K}_k) E(\boldsymbol{v}_k \boldsymbol{v}_k^T)(\boldsymbol{I} - \boldsymbol{\Psi}_k \boldsymbol{K}_k)^T$$

$$\Rightarrow E(\boldsymbol{\rho}_k \boldsymbol{\rho}_k^T) = (\boldsymbol{I} - \boldsymbol{\Psi}_k \boldsymbol{K}_k)\boldsymbol{\Psi}_k \bar{\boldsymbol{P}}_k \boldsymbol{\Psi}_k^T (\boldsymbol{I} - \boldsymbol{\Psi}_k \boldsymbol{K}_k)^T + (\boldsymbol{I} - \boldsymbol{\Psi}_k \boldsymbol{K}_k)\boldsymbol{R}_k (\boldsymbol{I} - \boldsymbol{\Psi}_k \boldsymbol{K}_k)^T$$

$$\Rightarrow E(\boldsymbol{\rho}_k \boldsymbol{\rho}_k^T) = \boldsymbol{\Psi}_k (\bar{\boldsymbol{P}}_k - \boldsymbol{K}_k \boldsymbol{\Psi}_k \bar{\boldsymbol{P}}_k)\boldsymbol{\Psi}_k^T (\boldsymbol{I} - \boldsymbol{\Psi}_k \boldsymbol{K}_k)^T + (\boldsymbol{I} - \boldsymbol{\Psi}_k \boldsymbol{K}_k)\boldsymbol{R}_k (\boldsymbol{I} - \boldsymbol{\Psi}_k \boldsymbol{K}_k)^T$$

$$\Rightarrow E(\boldsymbol{\rho}_k \boldsymbol{\rho}_k^T) = \boldsymbol{\Psi}_k \hat{\boldsymbol{P}}_k \boldsymbol{\Psi}_k^T (\boldsymbol{I} - \boldsymbol{\Psi}_k \boldsymbol{K}_k)^T + (\boldsymbol{R}_k - \boldsymbol{\Psi}_k \boldsymbol{K}_k \boldsymbol{R}_k)(\boldsymbol{I} - \boldsymbol{\Psi}_k \boldsymbol{K}_k)^T$$

$$\Rightarrow E(\boldsymbol{\rho}_k \boldsymbol{\rho}_k^T) = \boldsymbol{\Psi}_k \hat{\boldsymbol{P}}_k \boldsymbol{\Psi}_k^T - \boldsymbol{\Psi}_k \hat{\boldsymbol{P}}_k \boldsymbol{\Psi}_k^T \boldsymbol{K}_k^T \boldsymbol{\Psi}_k^T + \boldsymbol{R}_k - \boldsymbol{\Psi}_k \boldsymbol{K}_k \boldsymbol{R}_k - \boldsymbol{R}_k \boldsymbol{K}_k^T \boldsymbol{\Psi}_k^T + \boldsymbol{\Psi}_k \boldsymbol{K}_k \boldsymbol{R}_k \boldsymbol{K}_k^T \boldsymbol{\Psi}_k^T \tag{4.90}$$

The filter gain can be expressed using pseudo measurement matrix as $\boldsymbol{K}_k = (\hat{\boldsymbol{P}}_k \boldsymbol{\Psi}_k^T \boldsymbol{R}_k^{-1})$ following the expression of the gain of Kalman filter given in [Anderson1979]. Using this expression (4.90) can be expressed as

$$\Rightarrow E(\boldsymbol{\rho}_k \boldsymbol{\rho}_k^T) = \boldsymbol{\Psi}_k \hat{\boldsymbol{P}}_k \boldsymbol{\Psi}_k^T - \boldsymbol{\Psi}_k \hat{\boldsymbol{P}}_k \boldsymbol{\Psi}_k^T \boldsymbol{K}_k^T \boldsymbol{\Psi}_k^T + \boldsymbol{R}_k - \boldsymbol{\Psi}_k (\hat{\boldsymbol{P}}_k \boldsymbol{\Psi}_k^T \boldsymbol{R}_k^{-1})\boldsymbol{R}_k - \boldsymbol{R}_k (\boldsymbol{R}_k^{-1} \boldsymbol{\Psi}_k \hat{\boldsymbol{P}}_k)\boldsymbol{\Psi}_k^T + \boldsymbol{\Psi}_k (\hat{\boldsymbol{P}}_k \boldsymbol{\Psi}_k^T \boldsymbol{R}_k^{-1})\boldsymbol{R}_k \boldsymbol{K}_k^T \boldsymbol{\Psi}_k^T$$

$$\Rightarrow E(\boldsymbol{\rho}_k \boldsymbol{\rho}_k^T) = \boldsymbol{\Psi}_k \hat{\boldsymbol{P}}_k \boldsymbol{\Psi}_k^T - \boldsymbol{\Psi}_k \hat{\boldsymbol{P}}_k \boldsymbol{\Psi}_k^T \boldsymbol{K}_k^T \boldsymbol{\Psi}_k^T + \boldsymbol{R}_k - \boldsymbol{\Psi}_k \hat{\boldsymbol{P}}_k \boldsymbol{\Psi}_k^T - \boldsymbol{\Psi}_k \hat{\boldsymbol{P}}_k \boldsymbol{\Psi}_k^T + \boldsymbol{\Psi}_k \hat{\boldsymbol{P}}_k \boldsymbol{\Psi}_k^T \boldsymbol{K}_k^T \boldsymbol{\Psi}_k^T$$

$$\Rightarrow E(\boldsymbol{\rho}_k \boldsymbol{\rho}_k^T) = \boldsymbol{R}_k - \boldsymbol{\Psi}_k \hat{\boldsymbol{P}}_k \boldsymbol{\Psi}_k^T$$

$$\Rightarrow E(\boldsymbol{\rho}_k \boldsymbol{\rho}_k^T) = \boldsymbol{R}_k - \boldsymbol{\Psi}_k \hat{\boldsymbol{P}}_k \boldsymbol{\Psi}_k^T$$

$$\boldsymbol{C}_k^{\rho} = \boldsymbol{R}_k - \boldsymbol{\Psi}_k \hat{\boldsymbol{P}}_k \boldsymbol{\Psi}_k^T \tag{4.91}$$

In the ideal situation the window estimated residual covariance $\hat{\boldsymbol{C}}_k^{\rho}$ should be consistent with filter computed residual covariance for correctly adapted value of $\boldsymbol{R}$.

Hence, using the residual covariance adapted $\boldsymbol{R}$ can be expressed as

$$\hat{\boldsymbol{R}}_k = \hat{\boldsymbol{C}}_k^{\rho} + \boldsymbol{\Psi}_k \hat{\boldsymbol{P}}_k \boldsymbol{\Psi}_k^T \qquad (4.92)$$

Alternatively, with the relation $\hat{\boldsymbol{P}}_k^{g} \approx \boldsymbol{\Psi}_k \hat{\boldsymbol{P}}_k \boldsymbol{\Psi}_k^T$ the adapted $\boldsymbol{R}$ can be expressed as

$$\hat{\boldsymbol{R}}_k = \hat{\boldsymbol{C}}_k^{\rho} + \hat{\boldsymbol{P}}_k^{g}$$

Note that this expression is same as (4.58).

Maximum a Posterior (MAP) based method

Theorem 4.7

For window length L and with the consideration that the noise statistics are constant within the window length L the measurement noise covariance can be expressed as:

$$\hat{\boldsymbol{R}}_k = \frac{1}{k}\sum_{j=1}^{k}\left[\vartheta_j \vartheta_j^T - \overline{\boldsymbol{P}}_j^{g}\right] \qquad (4.93)$$

Proof:

The MAP based $\boldsymbol{R}$ adaptation algorithm can be derived in the same vein of $\boldsymbol{Q}$ adaptation method. From equation (4.50) we get of conditional density, $\boldsymbol{J}$ which is to be maximized for adapted value of $\boldsymbol{R}$.

$$\boldsymbol{J} = C|\boldsymbol{Q}|^{-k/2}|\boldsymbol{R}|^{-k/2}\exp\left\{-\frac{1}{2}\left[\sum_{j=1}^{k}\left\|\boldsymbol{x}_j - \boldsymbol{f}(\boldsymbol{x}_{j-1})\right\|_{\boldsymbol{Q}^{-1}}^2 + \sum_{j=1}^{k}\left\|\boldsymbol{y}_j - \boldsymbol{g}(\boldsymbol{x}_j)\right\|_{\boldsymbol{R}^{-1}}^2\right]\right\}$$

Where, the constant, $C = \exp\left\{-\frac{1}{2}\left\|\boldsymbol{x_0} - \hat{\boldsymbol{x}}_{\boldsymbol{0}}\right\|_{\boldsymbol{P}_0^{-1}}^2\right\}C_1C_2|\boldsymbol{P_0}|^{-1/2}\,p[\boldsymbol{Q},\boldsymbol{R}]$

Taking logarithm on both sides of (4.50)

$$\ln\boldsymbol{J} = -\frac{k}{2}\ln|\boldsymbol{Q}| - \frac{k}{2}\ln|\boldsymbol{R}| + \ln|C| - \frac{1}{2}\sum_{j=1}^{k}\left\|\boldsymbol{x}_j - \boldsymbol{f}(\boldsymbol{x}_{j-1})\right\|_{\boldsymbol{Q}^{-1}}^2 - \frac{1}{2}\sum_{j=1}^{k}\left\|\boldsymbol{y}_j - \boldsymbol{g}(\boldsymbol{x}_j)\right\|_{\boldsymbol{R}^{-1}}^2$$

For maximization of conditional density

$$\left.\frac{\partial\ln\boldsymbol{J}}{\partial\boldsymbol{R}}\right|_{\boldsymbol{R}=\hat{\boldsymbol{R}}_k}^{x_j=\bar{x}_{j|k}} = 0$$

Differentiating (4.50) with respect to $\boldsymbol{R}$ after taking logarithm the expression of adapted $\hat{\boldsymbol{R}}_k$ is obtained as

$$\hat{\boldsymbol{R}}_k = \sum_{j=1}^{k} \left[\left(\boldsymbol{y}_j - \boldsymbol{g}\left(\hat{\boldsymbol{x}}_{j|k} \right) \right) \left(\boldsymbol{y}_j - \boldsymbol{g}\left(\hat{\boldsymbol{x}}_{j|k} \right) \right)^T \right] \tag{4.94}$$

Note that an assumption is made where $\hat{\boldsymbol{x}}_{j|k}$ is known. Moreover, in the expression of adapted $\boldsymbol{R}$ the smoothed estimate $\hat{\boldsymbol{x}}_{j|k}$ is replaced by *a priori* estimate $\overline{\boldsymbol{x}}_j$ as calculation with smoothed estimate increases complexity. With this consideration (4.94) can be expressed as

$$\hat{\boldsymbol{R}}_k = \sum_{j=1}^{k} \left[\left(\boldsymbol{y}_j - \boldsymbol{g}\left(\overline{\boldsymbol{x}}_j \right) \right) \left(\boldsymbol{y}_j - \boldsymbol{g}\left(\overline{\boldsymbol{x}}_j \right) \right)^T \right] \tag{4.95}$$

However, in [Gao2015b] it is proved that the above expression of $\hat{\boldsymbol{R}}_k$ is inaccurate. The suboptimal estimation algorithm for $\hat{\boldsymbol{R}}_k$ is derived as given below following [Gao2015b].

$$\sum_{j=1}^{k} \left[\left(\boldsymbol{y}_j - \boldsymbol{g}\left(\overline{\boldsymbol{x}}_j \right) \right) \left(\boldsymbol{y}_j - \boldsymbol{g}\left(\overline{\boldsymbol{x}}_j \right) \right)^T \right] = \sum_{j=1}^{k} \left[\left(\boldsymbol{g}\left(\boldsymbol{x}_j \right) + v_j - \boldsymbol{g}\left(\overline{\boldsymbol{x}}_j \right) \right) \left(\boldsymbol{g}\left(\boldsymbol{x}_j \right) + v_j - \boldsymbol{g}\left(\overline{\boldsymbol{x}}_j \right) \right)^T \right]$$

$$\Rightarrow \frac{1}{k} \sum_{j=1}^{k} \vartheta_j \vartheta_j^{\boldsymbol{T}} = \frac{1}{k} \sum_{j=1}^{k} \left[\overline{\boldsymbol{P}}_j^{\boldsymbol{g}} + \boldsymbol{R} \right]$$

$$\Rightarrow \hat{\boldsymbol{R}}_k = \frac{1}{k} \sum_{j=1}^{k} \left[\vartheta_j \vartheta_j^{\boldsymbol{T}} - \overline{\boldsymbol{P}}_j^{\boldsymbol{g}} \right]$$

Hence, (4.93) is proved. □

During steady state $\overline{\boldsymbol{P}}_k^{\boldsymbol{g}}$ is considered to be steady within the estimation window and averaging of $\overline{\boldsymbol{P}}_k^{\boldsymbol{g}}$ may not be necessary. We can take $\overline{\boldsymbol{P}}_k^{\boldsymbol{g}}$ out of the estimation window and express adapted $\boldsymbol{R}$ as

$$\Rightarrow \hat{\boldsymbol{R}}_k = \hat{\boldsymbol{C}}_{\vartheta_k} - \overline{\boldsymbol{P}}_k^{\boldsymbol{g}}$$

Note that with this reasonable approximation expression of adapted $\boldsymbol{R}$ matches with (4.77)

<u>Theorem 4.8</u>

For window length L and with the consideration that the noise statistics are constant within the window the measurement noise covariance can be expressed as:

$$\hat{\boldsymbol{R}}_{\boldsymbol{k}} = \frac{1}{k}\sum_{j=1}^{k}\left[\boldsymbol{\rho}_j\boldsymbol{\rho}_j^T + \hat{\boldsymbol{P}}_j^g\right] \tag{4.96}$$

Proof:

For maximization of conditional density (4.50) with respect to $\boldsymbol{R}$ we can also think of using *a posteriori* estimate of state, $\hat{\boldsymbol{x}}_{\boldsymbol{k}}$ in place of *a priori* estimate of state, $\overline{\boldsymbol{x}}_{\boldsymbol{k}}$ in proof for theorem 4.7

$$\left.\frac{\partial \ln \boldsymbol{J}}{\partial \boldsymbol{R}}\right|_{\boldsymbol{R}=\hat{\boldsymbol{R}}_k}^{x_j=\hat{x}_j} = 0$$

Using this condition the expression of adapted $\hat{\boldsymbol{R}}_k$ is obtained as

$$\hat{\boldsymbol{R}}_k = \sum_{j=1}^{k}\left[\left(\boldsymbol{y}_j - \boldsymbol{g}\left(\hat{\boldsymbol{x}}_{j|k}\right)\right)\left(\boldsymbol{y}_j - \boldsymbol{g}\left(\hat{\boldsymbol{x}}_{j|k}\right)\right)^T\right] \tag{4.97}$$

In the expression of adapted $\boldsymbol{R}$, *a posteriori* estimate $\hat{\boldsymbol{x}}_j$ has been considered in place of smoothed estimate $\hat{\boldsymbol{x}}_{j|k}$. Replacing smoothed estimate by *a posteriori* estimate $\hat{\boldsymbol{x}}_j$ we present a different expression of adapted $\boldsymbol{R}$.

$$\hat{\boldsymbol{R}}_k = \sum_{j=1}^{k}\left[\left(\boldsymbol{y}_j - \boldsymbol{g}\left(\hat{\boldsymbol{x}}_j\right)\right)\left(\boldsymbol{y}_j - \boldsymbol{g}\left(\hat{\boldsymbol{x}}_j\right)\right)^T\right] \tag{4.98}$$

However, this adaptation algorithm again provides inaccurate estimate of $\hat{\boldsymbol{R}}_k$.

Let us consider the window estimate as

$$\sum_{j=1}^{k}\left[\left(\boldsymbol{y}_j - \boldsymbol{g}\left(\hat{\boldsymbol{x}}_j\right)\right)\left(\boldsymbol{y}_j - \boldsymbol{g}\left(\hat{\boldsymbol{x}}_j\right)\right)^T\right]$$

$$= \sum_{j=1}^{k}\left[\left(\boldsymbol{g}\left(\boldsymbol{x}_j\right) + v_j - \boldsymbol{g}\left(\hat{\boldsymbol{x}}_j\right)\right)\left(\boldsymbol{g}\left(\boldsymbol{x}_j\right) + v_j - \boldsymbol{g}\left(\hat{\boldsymbol{x}}_j\right)\right)^T\right]$$

$$= \sum_{j=1}^{k}\left[\left(\boldsymbol{g}\left(\boldsymbol{x}_j\right) + v_j - \boldsymbol{g}\left(\overline{\boldsymbol{x}}_j + \boldsymbol{K}_j\vartheta_j\right)\right)\left(\boldsymbol{g}\left(\boldsymbol{x}_j\right) + v_j - \boldsymbol{g}\left(\overline{\boldsymbol{x}}_j + \boldsymbol{K}_j\vartheta_j\right)\right)^T\right]$$

Here, $\boldsymbol{K}_j$ is the filter gain and ϑ_j is the innovation.

With the help of pseudo measurement matrix $\boldsymbol{\Psi}_{\boldsymbol{k}}$ the above expression can be simplified as done in covariance matching method from (4.88) to equation (4.90). Then, the window estimate can be expressed as

$$\sum_{j=1}^{k}\left[\left(\boldsymbol{y}_j-\boldsymbol{g}(\hat{\boldsymbol{x}}_j)\right)\left(\boldsymbol{y}_j-\boldsymbol{g}(\hat{\boldsymbol{x}}_j)\right)^T\right]=\sum_{j=1}^{k}\left[\boldsymbol{R}_j-\boldsymbol{\psi}_j\hat{\boldsymbol{P}}_j\boldsymbol{\psi}_j^T\right] \tag{4.99}$$

$$\hat{\boldsymbol{R}}_k=\sum_{j=1}^{k}\left[\left(\boldsymbol{y}_j-\boldsymbol{g}(\hat{\boldsymbol{x}}_j)\right)\left(\boldsymbol{y}_j-\boldsymbol{g}(\hat{\boldsymbol{x}}_j)\right)^T+\boldsymbol{\psi}_j\hat{\boldsymbol{P}}_j\boldsymbol{\psi}_j^T\right] \tag{4.100}$$

We can represent residual as $\boldsymbol{\rho}_j=\left(\boldsymbol{y}_j-\boldsymbol{g}(\hat{\boldsymbol{x}}_j)\right)$. Moreover, $\hat{\boldsymbol{P}}_j^{\boldsymbol{g}}\approx\boldsymbol{\psi}_j\hat{\boldsymbol{P}}_j\boldsymbol{\psi}_j^T$. With these considerations (4.100) becomes

$$\hat{\boldsymbol{R}}_k=\tfrac{1}{k}\sum_{j=1}^{k}\left[\boldsymbol{\rho}_j\boldsymbol{\rho}_j^T+\hat{\boldsymbol{P}}_j^{\boldsymbol{g}}\right]$$

Hence, (4.96) is proved. □

During steady state $\hat{\boldsymbol{P}}_k^{\boldsymbol{g}}$ is considered to be steady within the estimation window and averaging of $\hat{\boldsymbol{P}}_k^{\boldsymbol{g}}$ may not be necessary. We can take $\hat{\boldsymbol{P}}_k^{\boldsymbol{g}}$ out of the estimation window and express adapted $\boldsymbol{R}$ as

$$\Rightarrow\hat{\boldsymbol{R}}_k=\hat{\boldsymbol{C}}_k^{\boldsymbol{\rho}}+\hat{\boldsymbol{P}}_k^{\boldsymbol{g}}$$

Note that with this reasonable approximation expression of adapted $\boldsymbol{R}$ matches with (4.58)

4.2.3.3. Analysis of Unbiasedness of Adapted Noise Covariance

It can be said that an unbiased estimate of noise covariance has been presented by the adaptation algorithm when the expectation of adapted noise covariance would perfectly match with that of actual noise covariance. Alternatively it can be said that for an unbiased estimate the difference between the adapted value and the actual value should demonstrate a zero mean nature of variation.

For further discussions let us consider the expression of adapted measurement noise covariance from Theorem 4.5.2 as

$$\hat{\boldsymbol{R}}_k=\hat{\boldsymbol{C}}_{\vartheta_k}-\overline{\boldsymbol{P}}_k^h$$

The expectation of $\hat{\boldsymbol{R}}_k$ can be obtained as

$$E\left(\hat{\boldsymbol{R}}_k\right)=E\left(\hat{\boldsymbol{C}}_{\vartheta_k}-\overline{\boldsymbol{P}}_k^h\right)$$

Substituting the expression of window estimated innovation covariance we get

$$E\left(\hat{\boldsymbol{R}}_k\right) = E\left(\frac{1}{L}\sum_{j=j_0}^{k} \vartheta_j \vartheta_j^{\boldsymbol{T}} - \overline{\boldsymbol{P}}_k^{\,h}\right)$$

$$\Rightarrow E\left(\hat{\boldsymbol{R}}_k\right) = \frac{1}{L}\sum_{j=j_0}^{k} E\left(\vartheta_j \vartheta_j^{\boldsymbol{T}}\right) - \overline{\boldsymbol{P}}$$

$E\left(\hat{\boldsymbol{R}}_k\right)$, the expectation of $\hat{\boldsymbol{R}}_k$ becomes close to the true $\boldsymbol{R}_k$ i.e., $E\left(\hat{\boldsymbol{R}}_k\right) \to \boldsymbol{R}_k$ as the length of estimation window, L, is increased while satisfying the assumption 4.4. Theoretically the length may be infinite [Maybeck1982].

A larger estimation window is, therefore, recommended so long the noise statistics are stationary and ergodic within the estimation window. However, the situation may arise when the noise is non stationary with short time variation in covariance. In such situations, within a large estimation window, the statistics no longer remains stationary and a biased estimate of noise covariance is obtained.

To overcome this issue it is considered that the statistics are stationary within a small estimation window for the non stationary noises. In this way the unknown time varying noise covariances can be tracked. However, the estimate of the time varying noise covariances essentially becomes noisy with a choice of small window length. That is to say the difference between the estimated noise covariance and actual noise covariance will have high variances although the mean is zero. Sometimes the estimated covariance may also be biased (i.e., the mean of this value is different from the truth value) as the memory of only a few instants has been considered. Therefore, previous workers [Myers1976, Mohamed1999] have warned that for obtaining an unbiased and smooth estimate history of adequate instances has to be considered.

From the above discussions we can conclude that the unbiasedness of the adapted noise covariance depends significantly on the choice of epoch length or window size.

4.2.3.4. Choice of Window Size

With the above discussions it may be appreciated that the choice of window size for adaptive filtering is a major concern of the designer and needs experimentation. As the noise covariance remain unknown it is not easy to get the information whether the noise covariance is time varying. Therefore, the performance of filter needs to be investigated for different choice of window length on trial and error basis. For non stationary noise minimum choice of window length should be the order of system (for $\boldsymbol{Q}$ adaptation) or, the number of measurements (for $\boldsymbol{R}$ adaptation). Otherwise, divergence of filter cannot be over ruled [Mohamed1999].

The situation when the noise is stationary window length may be taken as the length up to the current time instant starting from initial time so that all the innovation/residual values of previous instants can be considered. In this approach the sliding window concept will not be applicable any more. This approach can initially present biased estimate of covariance when there remains large error between the assumed choice of the unknown noise covariance and its actual value. To converge on the truth value the adapted noise covariance may take more time as the innovation sequence retains the memory of all previous instants and consequently the estimation performance may get affected. In this situation the fixed length sliding window approach of adaptation would be appropriate.

For sliding window based adaptation the noise covariance is not adapted unless the time step index k is greater than or equal to the desired window length L. For small window length this will not affect much. However, for large window length the noise covariance remains frozen to the initial value for considerably long time which is not desirable.

To start adaptation early noise covariance is adapted initially based on a small window length L_{min} of innovation or, residual window. The window size is then gradually increased at each increment of step index k until the desire window length L is achieved. Afterwards noise covariance is adapted from sliding window.

4.2.3.5. Notes on Adaptation Methods

During the initial time steps the adapted value of process noise covariance may be less accurate or prone to be biased depending on the system dynamics and initial choice of $\boldsymbol{Q}$ as the $\boldsymbol{Q}$ adaptation algorithm is obtained after a few reasonable approximations. Afterwards, when the filter gradually attains steady state the assumption holds well and the adapted value of the process noise covariance becomes more accurate.

For $\boldsymbol{R}$ adaptation the residual sequence, instead of innovation, is preferred as residual based adaptation algorithm automatically ensures positive definiteness of adapted $\boldsymbol{R}$.

In case of innovation based $\boldsymbol{R}$ adaptation the adaptive filter often gets interrupted due to loss of positive definiteness of adapted $\boldsymbol{R}$. To avoid this singularity problem all the elements of the $\boldsymbol{R}$ matrix are forced to be positive by taking their absolute value. A similar ad hoc approach has been followed for innovation based $\boldsymbol{R}$ adaptation in [Almagbile2010]. In this perspective, the residual sequence based adaptation is preferable.

During implementation of $\boldsymbol{R}$ adaptive filters diagonalization of adapted $\boldsymbol{R}$ has to be carried out to ensure non-correlation when the measurement noises are not correlated. Otherwise, because of the window estimation the off diagonal terms may not turn up to be zero and its effect on the estimation may be detrimental for certain cases.

For residual based $\boldsymbol{R}$ adaptation it may be noted that the filter gain and the *a posteriori* error covariance depends on the adapted $\boldsymbol{R}$ implicitly. The adapted value of $\boldsymbol{R}$ subsequently depends on the gain and the error covariance. This self referencing problem can be circumvented following the alternative heuristic which uses some estimated value of the adapted measurement noise covariance. For the simplest case the adapted value of the previous instant can be used, i.e., $\overline{\boldsymbol{R}} = \hat{\boldsymbol{R}}_{k-1}$. Use of this approximation may not induce error in the estimate during steady state for the cases where the true value of $\boldsymbol{R}$ remains constant. However, during transients or if the true $\boldsymbol{R}$ is time varying tracking performance of adapted $\boldsymbol{R}$ may not be satisfactory due to the above approximation. The following iterative approach, though computationally intensive, can improve estimation accuracy. In this approach an intermediate variable $\boldsymbol{R}^{\#}$ is chosen as $\boldsymbol{R}^{\#} = \hat{\boldsymbol{R}}_k^{i^{\#}}$ where $\hat{\boldsymbol{R}}_k^{i}$ denotes the estimated value of $\boldsymbol{R}$ in the i^{th} iteration of the k^{th} step, and $i = 1, 2, 3, \cdots, i^{\#}$. In the iterative refinement $\boldsymbol{R}^{\#}$ takes the place of $\overline{\boldsymbol{R}}$ in the algorithm. If sufficient accuracy is achieved after the $i^{\#}$ iteration further iteration can be stopped. This method of iterative refinement has been termed as 're-computation' in the rest of the dissertation.

The similar approach of re-computation can also be followed for $\boldsymbol{Q}$ adaptation for iterative refinement of adapted $\boldsymbol{Q}$. However, this will also increase the computation burden.

4.3 Algorithms for Adaptive Nonlinear filters

4.3.1 Introduction to Algorithms

In this section general frameworks for adaptive nonlinear filters are presented. Two different frameworks for adaptive nonlinear filters have been proposed. One of them is in standard error covariance form and the other is in square root form. The motivation for the square root approach is also mentioned in the notes on the algorithm. With help of these frameworks a variety of adaptive filters are formulated depending on the choice of sigma points

4.3.2 Conventional Error Covariance form

This section presents the general framework of the adaptive Gaussian filter in standard error covariance form.

GENERAL FRAMEWORK FOR ADAPTIVE NONLINEAR FILTERS

(i) Initialization: Initialize $\hat{\boldsymbol{x}}_0, \hat{\boldsymbol{P}}_0, \overline{\boldsymbol{Q}}, \overline{\boldsymbol{R}}$

(ii) Time update step:

Compute Cholesky Factor such that $\hat{\boldsymbol{P}}_{k-1} = \hat{\boldsymbol{S}}_{k-1}\left(\hat{\boldsymbol{S}}_{k-1}\right)^T$ (4.101)

For the numerical method of integration select points and weights first for standard normal distribution and modify in the algorithmic steps as

$$\hat{\boldsymbol{\chi}}_i = \hat{\boldsymbol{S}}_{k-1}\boldsymbol{q}_i + \hat{\boldsymbol{x}}_{k-1} \tag{4.102}$$

Compute $\overline{\boldsymbol{x}}_k = \sum_{i=1}^{N} \boldsymbol{f}(\hat{\boldsymbol{\chi}}_i) w_i$ (4.103)

and $\overline{\boldsymbol{P}}_k = \overline{\boldsymbol{Q}} + \sum_{i=1}^{N} \left(\boldsymbol{f}(\hat{\boldsymbol{\chi}}_i) - \overline{\boldsymbol{x}}_k\right)\left(\boldsymbol{f}(\hat{\boldsymbol{\chi}}_i) - \overline{\boldsymbol{x}}_k\right)^T w_i$ (4.104)

$\overline{\boldsymbol{x}}_k$ is *a priori* estimate and $\overline{\boldsymbol{P}}_k$ is *a priori* error covariance

$\overline{\boldsymbol{Q}}$ denotes the assumed value of process noise covariance $\hat{\boldsymbol{Q}}_k$. In this algorithm, during $\boldsymbol{Q}$ adaptation, $\overline{\boldsymbol{Q}}$ is chosen as $\overline{\boldsymbol{Q}} = \hat{\boldsymbol{Q}}_{k-1}$ for the simplest case without re-computation.

(iii) Measurement update step:

Compute Cholesky Factor such that $\overline{\boldsymbol{P}}_k = \overline{\boldsymbol{S}}_k\left(\overline{\boldsymbol{S}}_k\right)^T$ (4.105)

Select the points as $\overline{\boldsymbol{\chi}}_i = \overline{\boldsymbol{S}}_k\boldsymbol{q}_i + \overline{\boldsymbol{x}}_k$ (4.106)

The *a priori* estimate of measurement becomes

$$\boldsymbol{z}_k = \sum_{i=1}^{N} \boldsymbol{g}(\overline{\boldsymbol{\chi}}_i) w_i \tag{4.107}$$

The following covariance can be computed as -

$$\boldsymbol{P}_k^{xz} = \sum_{i=1}^{N} \left(\overline{\boldsymbol{\chi}}_i - \overline{\boldsymbol{x}}_k\right)\left(\boldsymbol{g}(\overline{\boldsymbol{\chi}}_i) - \boldsymbol{z}_k\right)^T w_i \quad (4.108)$$

$$\boldsymbol{P}_k^{zz} = \sum_{i=1}^{N} \left(\boldsymbol{g}(\overline{\boldsymbol{\chi}}_i) - \boldsymbol{z}_k\right)\left(\boldsymbol{g}(\overline{\boldsymbol{\chi}}_i) - \boldsymbol{z}_k\right)^T w_i \quad (4.109)$$

The innovation covariance can be computed as the sum of $\boldsymbol{P}_k^{zz}$ and $\overline{\boldsymbol{R}}$. $\boldsymbol{P}_k^{zz}$ and $\boldsymbol{P}_k^{xz}$ is same as what is denoted before by $\overline{\boldsymbol{P}}_k^{g}$ and $\boldsymbol{P}_k^{xy}$ respectively. Here, $\overline{\boldsymbol{R}}$ is an approximation of adapted measurement noise covariance $\hat{\boldsymbol{R}}_k$. In this algorithm, during $\boldsymbol{R}$ adaptation, $\overline{\boldsymbol{R}}$ is chosen as the simplest case where $\overline{\boldsymbol{R}} = \hat{\boldsymbol{R}}_{k-1}$. Re-computation may be carried out as explained in 4.2.3.5.

The filter gain $\boldsymbol{K}_k$ is given by

$$\boldsymbol{K}_k = \boldsymbol{P}_k^{xz}\left(\boldsymbol{P}_k^{zz} + \overline{\boldsymbol{R}}\right)^{-1} \quad (4.110)$$

$$\hat{\boldsymbol{x}}_k = \overline{\boldsymbol{x}}_k + \boldsymbol{K}_k\left(\boldsymbol{y}_k - \boldsymbol{z}_k\right) \quad (4.111)$$

$$\hat{\boldsymbol{P}}_k = \overline{\boldsymbol{P}}_k - \boldsymbol{K}_k\left(\boldsymbol{P}_k^{zz} + \overline{\boldsymbol{R}}\right)\boldsymbol{K}_k^T \quad (4.112)$$

$\hat{\boldsymbol{x}}_k$ is *a posteriori* estimate of state and $\hat{\boldsymbol{P}}_k$ is *a posteriori* error covariance.

(iv) Q-Adaptation Steps:

When $\boldsymbol{Q}$ is unknown, on contrary, $\boldsymbol{R}$ is known (i.e., $\overline{\boldsymbol{R}} = \boldsymbol{R}$) the following steps are to be executed:

Compute the innovation sequence as

$$\vartheta_k = \boldsymbol{y}_k - \boldsymbol{z}_k \quad (4.113)$$

The estimated innovation covariance can be computed from a sliding window of epoch length L

$$\hat{\boldsymbol{C}}_{\vartheta_k} = \frac{1}{L} \sum_{j=k-L+1}^{k} \vartheta_k(j)\vartheta_k^T(j) \quad (4.114)$$

Direct adaptation algorithm for $\hat{\boldsymbol{Q}}_k$

$$\hat{\boldsymbol{Q}}_k = \boldsymbol{K}_k \hat{\boldsymbol{C}}_{\vartheta_k} \boldsymbol{K}_k \quad (4.115)$$

Alternative algorithm for adaptation of $\hat{\boldsymbol{Q}}_k$ with the help of scaling factor

$$\hat{\boldsymbol{Q}}_{\boldsymbol{k}} = \lambda_k \hat{\boldsymbol{Q}}_{\boldsymbol{k-1}} \tag{4.116}$$

Where, $$\lambda_k = \sqrt{\frac{trace\left(\hat{\boldsymbol{C}}_{\vartheta_k} - \boldsymbol{R}\right)}{trace\left(\boldsymbol{P}_{\boldsymbol{k}}^{y} - \boldsymbol{R}\right)}} \tag{4.117}$$

(v) R-Adaptation Steps:

When $\boldsymbol{R}$ is unknown, on contrary, $\boldsymbol{Q}$ is known (i.e., $\overline{\boldsymbol{Q}} = \boldsymbol{Q}$) the following steps are to be executed:

Innovation based R adaptation:

Compute the innovation sequence as

$$\vartheta_k = \boldsymbol{y}_{\boldsymbol{k}} - \boldsymbol{z}_{\boldsymbol{k}} \tag{4.118}$$

The estimated innovation covariance can be computed from a sliding window of epoch length L

$$\hat{\boldsymbol{C}}_{\vartheta_k} = \frac{1}{L} \sum_{j=k-L+1}^{k} \vartheta_k(j)\vartheta_k^T(j) \tag{4.119}$$

$$\hat{\boldsymbol{R}}_{\boldsymbol{k}} = \hat{\boldsymbol{C}}_{\vartheta_k} - \boldsymbol{P}_{\boldsymbol{k}}^{zz} \tag{4.120}$$

Residual based R adaptation:

Compute Cholesky Factor such that $$\hat{\boldsymbol{P}}_{\boldsymbol{k}} = \hat{\boldsymbol{S}}_{\boldsymbol{k}}\left(\hat{\boldsymbol{S}}_{\boldsymbol{k}}\right)^T \tag{4.121}$$

Select sigma points as $$\hat{\hat{\boldsymbol{\chi}}}_{\boldsymbol{i}} = \hat{\boldsymbol{S}}_{\boldsymbol{k}}\boldsymbol{q}_{\boldsymbol{i}} + \hat{\boldsymbol{x}}_{\boldsymbol{k}} \tag{4.122}$$

The *a posteriori* estimate of measurement and the respective error covariance becomes

$$\hat{\boldsymbol{z}}_{\boldsymbol{k}} = \sum_{i=1}^{N} \boldsymbol{g}(\hat{\hat{\boldsymbol{\chi}}}_{\boldsymbol{i}})w_i \tag{4.123}$$

$$\hat{\boldsymbol{P}}_{\boldsymbol{k}}^{zz} = \sum_{i=1}^{N}\left(\boldsymbol{g}(\hat{\hat{\boldsymbol{\chi}}}_{\boldsymbol{i}}) - \hat{\boldsymbol{z}}_{\boldsymbol{k}}\right)\left(\boldsymbol{g}(\hat{\hat{\boldsymbol{\chi}}}_{\boldsymbol{i}}) - \hat{\boldsymbol{z}}_{\boldsymbol{k}}\right)^T w_i \tag{4.124}$$

Compute the innovation sequence as

$$\boldsymbol{\rho}_{\boldsymbol{k}} = \boldsymbol{y}_{\boldsymbol{k}} - \hat{\boldsymbol{z}}_{\boldsymbol{k}} \tag{4.125}$$

The estimated residual covariance can be computed from a sliding window of epoch length L

$$\hat{\boldsymbol{C}}_k^{\rho} = \frac{1}{L} \sum_{j=k-L+1}^{k} \boldsymbol{\rho}_k(j)\boldsymbol{\rho}_k^T(j) \tag{4.126}$$

$$\hat{\boldsymbol{R}}_k = \hat{\boldsymbol{C}}_k^{\rho} + \hat{\boldsymbol{P}}_k^{zz} \tag{4.127}$$

(vi) Recursion: The time update and measurement update steps are repeated for estimates for the subsequent time steps starting from k=1.

4.3.3 Square Root version

This section presents the general framework of the adaptive nonlinear filter in the square root form.

GENERAL FRAMEWORK FOR ADAPTIVE NONLINEAR FILTERS IN SQUARE ROOT FORM

(i) Initialization: Initialize $\hat{\boldsymbol{x}}_0, \hat{\boldsymbol{S}}_0, \bar{\boldsymbol{S}}_k^{\boldsymbol{Q}}, \bar{\boldsymbol{S}}_k^{\boldsymbol{R}}$

(ii) Time update step:

For the particular mean and covariance modify the selected points for standard normal distribution as $\hat{\boldsymbol{\chi}}_i = \hat{\boldsymbol{S}}_{k-1}\boldsymbol{q}_i + \hat{\boldsymbol{x}}_{k-1}$ (4.128)

Compute $\bar{\boldsymbol{x}}_k = \sum_{i=1}^{N} \boldsymbol{f}(\hat{\boldsymbol{\chi}}_i)w_i$ (4.129)

Compute the weighted, centered (*a posteriori* estimate of previous instant is subtracted off) matrix $\boldsymbol{S}_k^x$ such that i^{th} element of $\boldsymbol{S}_k^x$:

$$\left(\boldsymbol{S}_k^x\right)_i = \left(\boldsymbol{f}(\hat{\boldsymbol{\chi}}_i) - \bar{\boldsymbol{x}}_k\right)\sqrt{w_i} \tag{4.130}$$

for $i = 1, 2, \cdots, N$

The estimate of the square root of *a priori* error covariance is obtained as

$$\bar{\boldsymbol{S}}_k = \text{Triangularize}\left(\begin{bmatrix}\boldsymbol{S}_k^x & \bar{\boldsymbol{S}}_k^Q\end{bmatrix}\right) \tag{4.131}$$

(iii) Measurement update step:

Select sigma points as $\bar{\boldsymbol{\chi}}_i = \bar{\boldsymbol{S}}_k \boldsymbol{q}_i + \bar{\boldsymbol{x}}_k$ (4.132)

The *a priori* estimate of measurement becomes

$$\boldsymbol{z_k} = \sum_{i=1}^{N} \boldsymbol{g}(\boldsymbol{\overline{\chi}_i}) w_i \tag{4.133}$$

Compute the weighted, centered (*a priori* estimate of measurement is subtracted off) matrix $\boldsymbol{S}_k^z$ such that i^{th} element of $\boldsymbol{S}_k^z$:

$$\left(\boldsymbol{S}_k^z\right)_i = \left(\boldsymbol{g}(\boldsymbol{\overline{\chi}_i}) - \boldsymbol{z_k}\right)\sqrt{w_i} \tag{4.134}$$

for $i = 1, 2, \cdots, N$

Compute the weighted, centered (*a priori* estimate of state is subtracted off) matrix $\boldsymbol{S}_k^{\bar{x}}$ such that i^{th} element of $\boldsymbol{S}_k^{\bar{x}}$:

$$\left(\boldsymbol{S}_k^{\bar{x}}\right)_i = \left(\boldsymbol{\overline{\chi}_i} - \boldsymbol{\overline{x}_k}\right)\sqrt{w_i} \tag{4.135}$$

for $i = 1, 2, \cdots, N$

The cross following covariance can be computed as -

$$\boldsymbol{P_k^{xz}} = \boldsymbol{S}_k^{\bar{x}}\left(\boldsymbol{S}_k^z\right)^T \tag{4.136}$$

The innovation covariance becomes

$$\boldsymbol{S}_k^{zz} = \text{Triangularize}\left(\left[\boldsymbol{S}_k^z \quad \boldsymbol{\overline{S}}_k^R\right]\right) \tag{4.132}$$

The filter gain is computed as

$$\boldsymbol{K_k} = \boldsymbol{P_k^{xz}}\left(\boldsymbol{S_k^{zz}}\right)^{-T}\left(\boldsymbol{S_k^{zz}}\right)^{-1} \tag{4.137}$$

The estimate of the square root of *a posteriori* error covariance is computed as

$$\hat{\boldsymbol{S}}_k = \text{Triangularize}\left(\left[\boldsymbol{S}_k^{\bar{x}} - \boldsymbol{K}_k\boldsymbol{S}_k^z \quad \boldsymbol{K}_k\boldsymbol{\overline{S}}_k^R\right]\right) \tag{4.138}$$

The *a posteriori* state estimate is given by

$$\boldsymbol{\hat{x}_k} = \boldsymbol{\overline{x}_k} + \boldsymbol{K_k}\left(\boldsymbol{y_k} - \boldsymbol{z_k}\right) \tag{4.139}$$

(iv) Q-Adaptation Steps:

When $\boldsymbol{Q}$ is unknown, on contrary, $\boldsymbol{R}$ is known, i.e., $\boldsymbol{\overline{S}_k^R} = \boldsymbol{S_k^R}$ the following steps are to be executed to adapt $\boldsymbol{\overline{S}_k^Q}$:

Compute the innovation sequence as

$$\vartheta_k = \boldsymbol{y_k} - \boldsymbol{z_k} \tag{4.140}$$

Compute the matrix from the innovation sequence as

$$\boldsymbol{S}_k^{\vartheta} = [\vartheta_k(k-L+1) \quad \cdots \quad \vartheta_k(k)] \tag{4.141}$$

The adaptated square root of process noise covariance $\hat{\boldsymbol{S}}_k^{Q}$ is obtained as

$$\hat{\boldsymbol{S}}_k^{Q} = \text{Triangularize}\left(\boldsymbol{K_k}\boldsymbol{S}_k^{\vartheta}\right) \tag{4.142}$$

(v) R-Adaptation Steps:

When $\boldsymbol{R}$ is unknown on contrary, $\boldsymbol{Q}$ is known, i.e., $\overline{\boldsymbol{S}}_k^{Q} = \boldsymbol{S}_k^{Q}$ the following steps are to be executed to adapt $\overline{\boldsymbol{S}}_k^{R}$:

Select sigma points as $\hat{\hat{\boldsymbol{\chi}}}_{\boldsymbol{i}} = \hat{\boldsymbol{S}}_{\boldsymbol{k}}\,\boldsymbol{q_i} + \hat{\boldsymbol{x}}_{\boldsymbol{k}}$ (4.143)

Compute $\hat{\boldsymbol{z}}_{\boldsymbol{k}} = \sum_{i=1}^{N} \boldsymbol{g}(\hat{\hat{\boldsymbol{\chi}}}_{\boldsymbol{i}})w_i$ (4.144)

Compute the weighted, centered (*a posteriori* estimate for measurement of previous instant is subtracted off) matrix $\boldsymbol{S}_k^{\hat{z}}$ such that i^{th} element of $\boldsymbol{S}_k^{\hat{z}}$:

$$\left(\boldsymbol{S}_k^{\hat{z}}\right)_i = \left(\boldsymbol{g}(\hat{\hat{\boldsymbol{\chi}}}_{\boldsymbol{i}}) - \hat{\boldsymbol{z}}_{\boldsymbol{k}}\right)\sqrt{w_i} \tag{4.145}$$

for $i = 1, 2, \cdots, N$

Compute the residual sequence as given by

$$\boldsymbol{\rho_k} = \boldsymbol{y_k} - \hat{\boldsymbol{z}}_{\boldsymbol{k}} \tag{4.146}$$

Compute the matrix from the innovation sequence as

$$\boldsymbol{S}_{\boldsymbol{k}}^{\rho} = [\boldsymbol{\rho_k}(k-L+1) \quad \cdots \quad \boldsymbol{\rho_k}(k)] \tag{4.147}$$

The estimate of the square root of *a priori* error covariance is obtained as

$$\hat{\boldsymbol{S}}_k^{R} = \text{Triangularize}\left(\left[\boldsymbol{S}_k^{\hat{z}} \quad \boldsymbol{S}_k^{\rho}\right]\right) \tag{4.148}$$

(vi) Recursion: The time update and measurement update steps are repeated for estimates for the subsequent time steps starting from *k*=1.

4.3.3.1. Notes on Square Root version of Adaptive Nonlinear filters

The square root framework of adaptive nonlinear filters are formulated and recommended over the standard error covariance form because of the reasons explained in [Anderson1979, Arasaratnam2008, Liu2012]. Those points have been reiterated below along with some significant attributes of the proposed general framework for adaptive nonlinear estimators in square root framework.

- The square root form preserves the symmetry of the error covariance matrix which is a significant aspect of the Kalman filter and its variants. Numerical stability of the filter is improved in the presence of the symmetric error covariance matrix.
- The condition number of a matrix is square of the condition number of its square root matrix. Because of smaller condition number of square roots, the estimates involving square roots will have more numerical accuracy compared to the estimates with the standard error covariance form.
- In standard error covariance form, the numerical approximation error creeps in because of the matrix square root computation in the time update and the measurement update step of the filter. This may degrade its accuracy. On contrary, in case of the square root approach the square root is directly updated instead of error covariance.
- In light of the above discussions it may also important to note that the results from the square root approach in single precision is comparably same as that of standard error covariance approach in double precision. In applications where the word length is limited, the square root version may be appropriate compared to the standard error covariance as it can provide estimates with improved precision compared to the error covariance form in such situations.
- The *a posteriori* error covariance matrix may suffer from the singularity problem for the application with limited precision in the situation where accuracy of measurements are high or a linear combination of state vector components is known with better accuracy while other combinations may not be well observable. Square root approach can overcome such singularity problems.

- In the adaptation step, the square root of the process noise covariance and the measurement noise covariance get adapted. Therefore, the positive definiteness of adapted process noise covariance and the measurement noise covariance is ensured by the square root version of adaptive nonlinear filers.
- Note that for adaptation of the square root of ***R*** matrix only residual based adaptation approach is followed as the innovation based approach is not straightforward for the square root framework. However, for adaptation of ***Q*** matrix innovation sequence has been employed as the adaptation algorithm can be conveniently modified for the square root framework.
- In the algorithm of adaptive filters with square root framework the matrix inversion steps can be replaced by backward substitution symbolized by '/' as the latter is computationally economic [Arasaratnam2008]. In case of square root approach the triangular matrix is obtained from the QR factorization unlike the standard error covariance form. On the availability of a square upper triangular matrix we can use the back substitution instead of matrix inversion to reduce the computational burden.
- The algorithm presented above is applicable only for the sigma point filters where the respective weights for the sigma points are non negative. Therefore, the algorithm can be applied with Gauss Hermite Quadrature Rule, 3rd degree Cubature Rule, Cubature Quadrature Rule, and non scaled version of Unscented Transformation Rule.
- However, the above algorithm cannot be applied with non scaled version of Unscented Transformation Rule and 5th degree Cubature rule as some of the weights becomes negative. In that case the "cholupdate" command has to be used as described in [Merwe2004]. This modification, however, does not change the adaptation part. The non-adaptive part of the above algorithm for the SR UKF (scaled) and SR ACKF5 can be obtained from the work of [Merwe2004].
- It has been reported in the dissertation of [Liu2012] that the SR UKF may also terminate due to loss of positive definiteness for some specific runs while updating the Cholesky matrix using "cholupdate". This may happen under the influence of the negative weights of SR UKF. The present worker has, therefore, avoided developing the square root version of adaptive UKF and CKF (5th degree).

4.4 Demonstration with Adaptive UKF

In this section we demonstrate the performance of adaptive Unscented Kalman Filter for adaptation of unknown measurement noise covariance which has been derived using the general framework for adaptive nonlinear estimators. For numerical approximation of the Bayesian integrals sigma points are chosen using Unscented Transformation rule proposed in the work of [Wan2000, Merwe2004]. For $\boldsymbol{R}$ adaptation covariance matching method incorporating residual sequence is followed as provided in theorem 4.5.1.The algorithm of AUKF has been discussed in details in the publications [Das2015, Das2013] by the co-worker, Ms. Manasi Das. Demonstration of the performance of AUKF for object tracking problem appears in a conference paper by Ms. Das with joint authorship with the present worker which has been referred in the list of conference papers with serial number '9', section 1.7.3, chapter 1.

4.4.1 Choice of Sigma Points

We consider an $\boldsymbol{n}$-element vector $\boldsymbol{x}$ with mean $\bar{\boldsymbol{x}}$ and covariance $\bar{\boldsymbol{P}}$.

Given a known nonlinear transformation $\boldsymbol{y} = \boldsymbol{h}(\boldsymbol{x})$, the mean and covariance of $\boldsymbol{y}$ can be obtained with the help of sigma points which are deterministically chosen following Unscented Transformation rule. The sigma points are be selected in the following way as reported in [Wan2000, Merwe2004]:

Generate *2n+1* sigma points as

$$\boldsymbol{x}_i = \bar{\boldsymbol{x}} + \tilde{\boldsymbol{x}}_i \tag{4.149}$$

for $i = 0, \cdots, 2n$

Where, $\tilde{\boldsymbol{x}}_0 = 0$;

$$\tilde{\boldsymbol{x}}_i = \left(\sqrt{(n+\lambda)\bar{\boldsymbol{P}}}\right)_i \tag{4.150}$$

for $i = 0, \cdots, n$

$$\tilde{\boldsymbol{x}}_{n+i} = \left(-\sqrt{(n+\lambda)\bar{\boldsymbol{P}}}\right)_i \tag{4.151}$$

for $i = 0, \cdots, n$

Here $\left(\sqrt{\overline{\boldsymbol{P}}}\right)_i$ denotes the i^{th} column of the cholesky factor (matrix square root) of covariance matrix $\overline{\boldsymbol{P}}$, n and λ represent the number of state variables and scaling factor respectively. The parameter α determines the spread of the sigma points and β denotes the prior knowledge about the noise distribution. The scale factor λ is determined as $\lambda = \alpha^2(n+\kappa) - n$, where κ is usually set to 0.

The weights corresponding to the sigma points are

$$W_0^m = \frac{\lambda}{n+\lambda};\ W_0^c = \frac{\lambda}{n+\lambda} + (1-\alpha^2+\beta) \tag{4.152}$$

for *i=0* and

$$W_i^m = W_i^c = \frac{1}{2(n+\lambda)} \quad \text{for}, i \neq 0. \tag{4.153}$$

Transform the sigma points as $\boldsymbol{y}_i = \boldsymbol{h}(\tilde{\boldsymbol{x}}_i)$

The mean and the covariance of the transformed vector:

$$\overline{\boldsymbol{y}} = \sum_{i=0}^{2n} \boldsymbol{h}(\tilde{\boldsymbol{x}}_i) W_i^m \tag{4.154}$$

$$\overline{\boldsymbol{P}}_{\boldsymbol{y}} = \sum_{i=0}^{2n} \left(\boldsymbol{h}(\tilde{\boldsymbol{x}}_i) - \overline{\boldsymbol{y}}\right)\left(\boldsymbol{h}(\tilde{\boldsymbol{x}}_i) - \overline{\boldsymbol{y}}\right)^T W_i^c \tag{4.155}$$

The adaptive UKF can be formulated using the general framework for adaptive nonlinear filter. The sigma points which are to be used in the general framework to obtain the algorithm of AUKF are chosen following the method presented above. The vector which is to be transformed through the nonlinear function is zero mean with covariance as identity matrix. Therefore the sigma points becomes

$$\boldsymbol{q}_i = \boldsymbol{e}_i \sqrt{n+\lambda} \ \text{ for } i = 0, \cdots, n \tag{4.156}$$

$$\boldsymbol{q}_{n+i} = -\boldsymbol{e}_i \sqrt{n+\lambda} \ \text{ for } i = 0, \cdots, n \tag{4.157}$$

Here, $\boldsymbol{e}_i$ is the i^{th} unit vector

The weights corresponding to the sigma points, $\boldsymbol{q}_i$ are obtained from (4.152) and (4.153).

4.4.2 Case Study: Object Tracking Problem

The object tracking problem in single dimension has been considered to illustrate the performance of AUKF. The estimation problem is discussed in the chapter 3. The true state trajectories of the reentry object is generated in simulation using the true values of the initial states and noise covariances ($\boldsymbol{x_0}, \boldsymbol{Q}_{true}$ & $\boldsymbol{R}_{true}$ respectively) as given in Chapter 3. In the face of unavailability of the measurement noise covariance both adaptive and non-adaptive filters are initialized with an assumed value of $\boldsymbol{R}$ ($\boldsymbol{R}_{filter} = \boldsymbol{R}_{true} \times 100$). The tuning parameters for AUKF are chosen as α=0.6, β=2 and $\kappa = 0$. The window size for $\boldsymbol{R}$ adaptation is chosen as 60.

The performance of AUKF is compared with non-adaptive UKF when truth value of $\boldsymbol{R}$ is unknown. The performance comparison is executed with the help of Monte Carlo Simulation with 1000 runs.

Fig. 4.1, 4.2 and Fig.4.3 represent RMS errors of the estimates from AUKF and non-adaptive UKF for altitude, velocity and the ballistic parameter respectively. The RMSE performance of AUKF is superior to that of non-adaptive UKF when initialized with an arbitrary choice of measurement noise covariance because of the unavailability of its accurate value. The RMS error of AUKF for states as well as the ballistic parameter converged to a lower steady state value compared to the non-adaptive UKF and the time to taken by RMS error to settle down is comparatively less in case of AUKF.

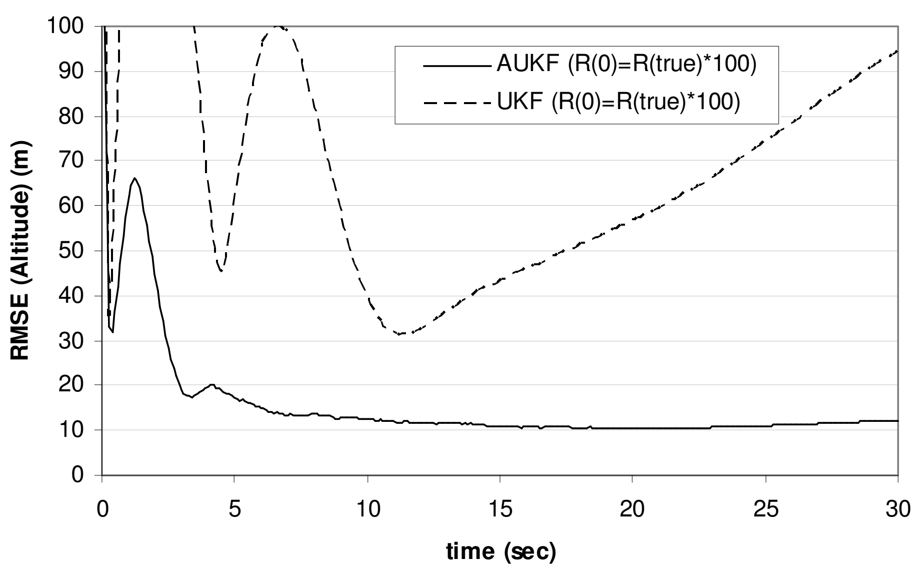


Fig. 4.1: RMSE of altitude for 1000 Monte Carlo runs.

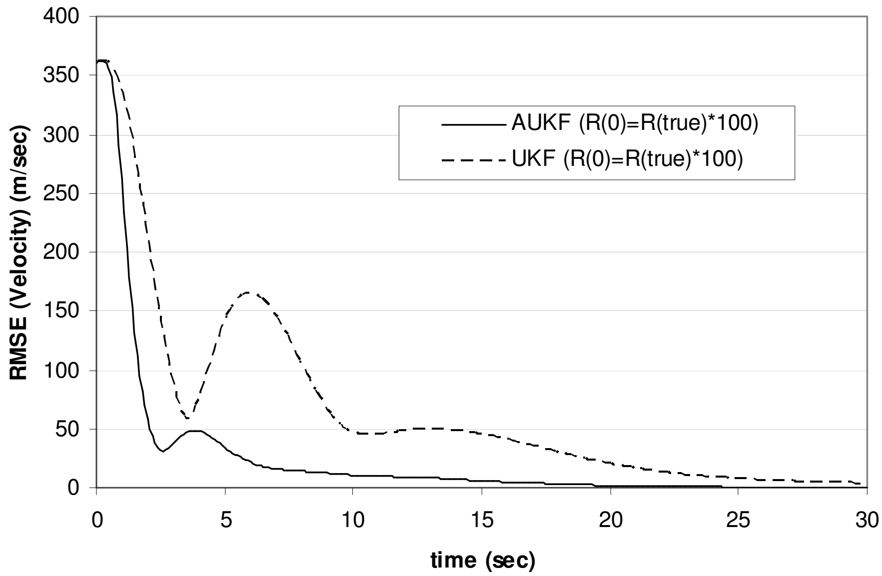


Fig. 4.1: RMSE of velocity for 1000 Monte Carlo runs.

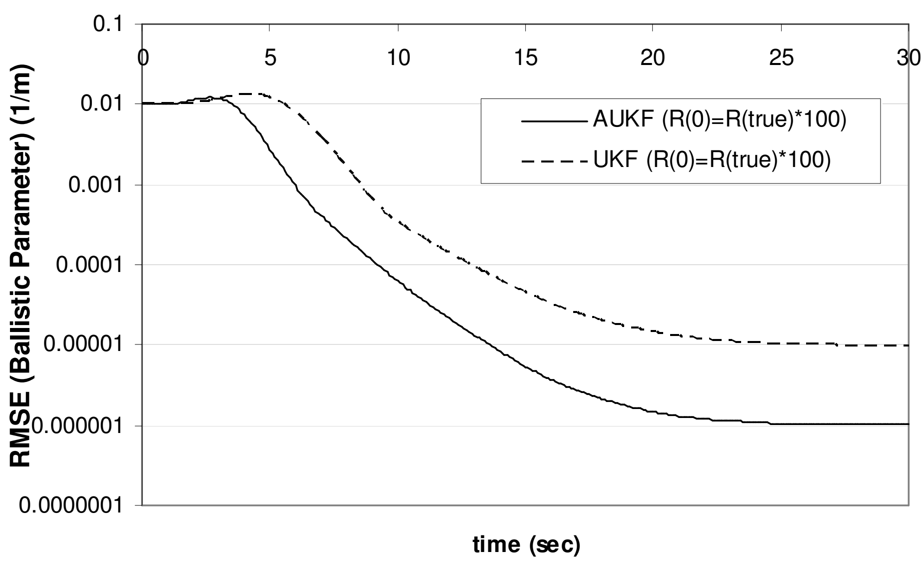


Fig. 4.2: RMSE of ballistic parameter for 1000 Monte Carlo runs

4.5 Discussions and Conclusions

A general framework for adaptive nonlinear estimators has been presented in this chapter. Depending on the unavailability of the knowledge of process noise or measurement noise covariance, the adaptation algorithms are classified and presented along with the mathematical derivations. It is interesting to note that the adaptation algorithms which have been derived following the method of MLE, covariance matching and MAP are closely similar provided some simplifying and reasonable assumptions are made.

On the availability of proposed general framework the scope for formulation of a class of adaptive nonlinear estimators remains open. Here it is demonstrated that the algorithm of adaptive UKF can be obtained using the proposed general framework. The algorithm of adaptive UKF is validated in simulation. The performance of $\boldsymbol{R}$ adaptive UKF is found to be superior compared to its non-adaptive version when the measurement noise covariance remains unknown. The same trend is expected to be followed by all the adaptive nonlinear estimators which can be formulated from the proposed general framework.

In the subsequent chapters of this dissertation a number of $\boldsymbol{Q}$ adaptive and $\boldsymbol{R}$ adaptive nonlinear estimators have been developed from the general framework using different numerical methods for approximation of the Bayesian integrals other than Unscented Transformation rule. The relative performance comparison of these competing algorithms of adaptive estimators has also been explored with the help of a number of case studies.

Chapter 5: Adaptive Divided Difference Filter

5.1 Chapter Introduction

In this chapter algorithms for Adaptive Divided Difference filter (ADDF) has been proposed and validated. Adaptive Divided Difference filter is based on the underlying framework of Divided Difference filter [Norgaard2000] which is non-adaptive on its own and formulated based on Taylor series approximation of nonlinear signal models. The Jacobian and Hessian matrices which appear in the Taylor series approximation are replaced by function evaluations using Stirling's interpolation formula. We have seen in chapter 4 that in the proposed general framework for adaptive nonlinear filter the Bayesian integrals are numerically approximated as summation using sigma points and weights. As non-adaptive DDF (which is the core of ADDF) is based on Taylor series approximation the algorithmic steps for ADDF cannot be directly derived the proposed general framework.

Non-adaptive version of Divided Difference filter (DDF) was first proposed in [Norgaard2000] and also reported in the contemporary work of [Ito2000] in the name of Central Difference filter. Both of these algorithms are based on second order approximation and the extended form of the Central Difference filter proposed by [Schei1997] which incorporates only first order approximation of nonlinear signal models. In [Norgaard2000, Ito2000] it is observed that the performance of non-adaptive DDF is demonstrably superior compared to second order EKF, Central Difference Filter [Schei1997] and comparable with Unscented Kalman filter. The interpolation formula used in DDF does not require careful choice of tuning parameters as it is essential for satisfactory performance of UKF. The adaptive versions of DDF are expected to inherit all these aspects of non-adaptive DDF.

Adaptation methods are incorporated in the algorithm of non-adaptive DDF to make estimation successful in absence of complete knowledge of the noise covariances. With concept of adaptation methods (discussed in chapter 4) $\boldsymbol{Q}$ and $\boldsymbol{R}$ adaptation algorithms are developed to suit DDF framework. The covariance matrices appear in the adaptation algorithms are computed with the help of Divided Difference interpolation formula. The algorithm for ADDF presented in this chapter is in standard error covariance form. The algorithm can also be expressed in the square root form by updating the square root of *a* *priori* and *a posteriori* error covariance matrix using QR factorization at the time update and measurement update steps respectively. In this chapter performance of $\boldsymbol{Q}$ and $\boldsymbol{R}$ adaptation algorithms are illustrated with several case studies.

Adaptive versions of DDF are seldom reported in literature. This chapter presents a set of new algorithms of ADDF. Before presenting the algorithms and the simulation results Stirling interpolation formula and its applicability for developing DDF have been discussed in brief for the ease of the interpretation of the algorithmic steps.

5.2 Stirling's Interpolation Formula

Let the multi dimensional variable $\boldsymbol{y}$ is a nonlinear vector function of $\boldsymbol{x}$ where $\boldsymbol{x} \in \boldsymbol{R}^{n}$. The vector $\boldsymbol{y}$ can be approximately expressed with the help of multi dimensional Taylor series approximation formula presented in [Norgaard2000] as given below:

$$\boldsymbol{y} = \boldsymbol{f}\left(\bar{\boldsymbol{x}} + \boldsymbol{\Delta x}\right) = \sum_{\boldsymbol{i}=0}^{\infty} \tfrac{1}{i!} D_{\boldsymbol{\Delta x}}^{\boldsymbol{i}} \boldsymbol{f} \tag{5.1}$$

$D_{\boldsymbol{\Delta x}}^{\boldsymbol{i}}$ as presented in [Norgaard2000] is interpreted as

$$D_{\boldsymbol{\Delta x}}^{\boldsymbol{i}} \boldsymbol{f} = \left(\Delta x_1 \tfrac{\partial}{\partial x_1} + \Delta x_2 \tfrac{\partial}{\partial x_2} + \cdots + \Delta x_n \tfrac{\partial}{\partial x_n}\right)^i \boldsymbol{f}\left(\boldsymbol{x}\right)\Big|_{\boldsymbol{x}=\bar{\boldsymbol{x}}} \tag{5.2}$$

Using Stirling's interpolation formula $\boldsymbol{y}$ can be approximated considering up to second order terms and requirement of computation of the Jacobian and Hessian matrices is replaced by function evaluations as presented in [Norgaard2000]. Following the interpolation formula $\boldsymbol{y}$ is expressed as

$$\boldsymbol{y} \approx \boldsymbol{f}\left(\bar{\boldsymbol{x}}\right) + \tilde{\boldsymbol{D}}_{\boldsymbol{\Delta x}} \boldsymbol{f} + \tfrac{1}{2!} \tilde{\boldsymbol{D}}_{\boldsymbol{\Delta x}}^{2} \boldsymbol{f} \tag{5.3}$$

Following Stirling's interpolation formula the Divided Difference operators are expressed as

$$\tilde{\boldsymbol{D}}_{\boldsymbol{\Delta x}} \boldsymbol{f} = \frac{1}{h}\left(\sum_{p=1}^{n} \Delta x_p \mu_p \delta_p\right) \boldsymbol{f}\left(\bar{\boldsymbol{x}}\right) \tag{5.4}$$

$$\tilde{\boldsymbol{D}}_{\boldsymbol{\Delta x}}^{2} \boldsymbol{f} = \frac{1}{h^2}\left(\sum_{p=1}^{n} \Delta x_p^2 \delta_p^2 + \sum_{p=1}^{n} \sum_{\substack{q=1 \\ q \neq p}}^{n} \Delta x_p \Delta x_q \left(\mu_p \delta_p\right)\left(\mu_q \delta_q\right)\right) \boldsymbol{f}\left(\bar{\boldsymbol{x}}\right) \tag{5.5}$$

where *h* denotes the interval length.

Here, δ_p is the partial difference operator and defined as

$$\delta_p \boldsymbol{f}(\bar{\boldsymbol{x}}) = \boldsymbol{f}\left(\bar{\boldsymbol{x}} + \tfrac{h}{2}\boldsymbol{e_p}\right) - \boldsymbol{f}\left(\bar{\boldsymbol{x}} - \tfrac{h}{2}\boldsymbol{e_p}\right) \tag{5.6}$$

The average operator, μ_p is defined as

$$\mu_p \boldsymbol{f}(\bar{\boldsymbol{x}}) = \tfrac{1}{2}\left(\boldsymbol{f}\left(\bar{\boldsymbol{x}} + \tfrac{h}{2}\boldsymbol{e_p}\right) - \boldsymbol{f}\left(\bar{\boldsymbol{x}} - \tfrac{h}{2}\boldsymbol{e_p}\right)\right) \tag{5.7}$$

In (5.6) and (5.7) $\boldsymbol{e_p}$ represents p^{th} unit vector.

With the linear transformation of the variable $\boldsymbol{x}$ as $\boldsymbol{z} = \boldsymbol{S}^{-1}\boldsymbol{x}$, $\tilde{f}(\cdot)$ can be defined by

$$\tilde{\boldsymbol{f}}(z) \overset{\Delta}{=} \boldsymbol{f}(\boldsymbol{S}z) = \boldsymbol{f}(\boldsymbol{x}) \tag{5.8}$$

Applying multi dimensional interpolation formula for $\tilde{\boldsymbol{f}}(z)$ we get

$$2\mu_p \delta_p \tilde{\boldsymbol{f}}(z) = \tilde{\boldsymbol{f}}\left(\bar{z} + h\boldsymbol{e_p}\right) - \tilde{\boldsymbol{f}}\left(\bar{z} - h\boldsymbol{e_p}\right) \tag{5.9}$$

(5.9) can also be expressed using (5.8) as

$$2\mu_p \delta_p \tilde{\boldsymbol{f}}(z) = \boldsymbol{f}\left(\bar{\boldsymbol{x}} + h\boldsymbol{s_p}\right) - \boldsymbol{f}\left(\bar{\boldsymbol{x}} - h\boldsymbol{s_p}\right) \tag{5.10}$$

Where $\boldsymbol{s_p}$ is p^{th} column of transformation matrix $\boldsymbol{S}$.

5.3 Approximation of mean and covariance of a random variable

For the above discussions we now consider $\boldsymbol{x}$ as a stochastic multi dimensional variable with mean, $\bar{\boldsymbol{x}} = E[\boldsymbol{x}]$ and covariance $\boldsymbol{P_x} = E\left[(\boldsymbol{x} - \bar{\boldsymbol{x}})(\boldsymbol{x} - \bar{\boldsymbol{x}})^T\right]$. For a transformed variable $\boldsymbol{y} = \boldsymbol{f}(\boldsymbol{x})$, its mean and covariance can be computed as

$$\bar{\boldsymbol{y}} = E[\boldsymbol{f}(\boldsymbol{x})] \tag{5.11}$$

$$\boldsymbol{P_y} = E\left[(\boldsymbol{f}(\boldsymbol{x}) - \bar{\boldsymbol{y}})(\boldsymbol{f}(\boldsymbol{x}) - \bar{\boldsymbol{y}})^T\right] \tag{5.12}$$

$$\boldsymbol{P_{xy}} = E\left[(\boldsymbol{x} - \bar{\boldsymbol{x}})(\boldsymbol{f}(\boldsymbol{x}) - \bar{\boldsymbol{y}})^T\right] \tag{5.13}$$

We introduce a new variable $\boldsymbol{z} = \boldsymbol{S}_x^{-1}\boldsymbol{x}$ such that $\boldsymbol{S}_x\boldsymbol{S}_x^T = \boldsymbol{P}_x$. The transformation matrix $\boldsymbol{S}_x$ is selected as the Cholesky factor of $\boldsymbol{P}_x$ chosen as triangular matrix for computational efficiency. The elements of z are decoupled (uncorrelated) as, $E\left[(z - E[z])(z - E[z])^T\right] = \boldsymbol{I}$. The relation (5.8) also holds here, i.e., $\tilde{\boldsymbol{f}}(z) = \boldsymbol{f}(\boldsymbol{S}_x z) = \boldsymbol{f}(\boldsymbol{x})$.

For the following subsections the following assumptions need to be considered.

We define $\Delta z = z - E[z]$ and Δz follows a zero mean Gaussian distribution.

5.3.1 First order approximation

The nonlinear function $\boldsymbol{y} = \tilde{\boldsymbol{f}}(z)$ is presented with first order approximation as

$$\boldsymbol{y} = \tilde{\boldsymbol{f}}(\bar{z} + \Delta z) = \tilde{\boldsymbol{f}}(\bar{z}) + \tilde{D}_{\Delta z}\tilde{\boldsymbol{f}} \tag{5.14}$$

Therefore, $E(\boldsymbol{y}) = E\left(\tilde{\boldsymbol{f}}(\bar{z}) + \tilde{D}_{\Delta z}\tilde{\boldsymbol{f}}\right) = \tilde{\boldsymbol{f}}(\bar{z}) = \boldsymbol{f}(\bar{\boldsymbol{x}})$ (5.15)

Here, the first-order moments are neglected since Δz is zero mean and $E(\Delta z) = \boldsymbol{0}$.

$$\boldsymbol{P}_y = E\left[\left(\tilde{\boldsymbol{f}}(\bar{z}) + \tilde{D}_{\Delta z}\tilde{\boldsymbol{f}} - \tilde{\boldsymbol{f}}(\bar{z})\right)\left(\tilde{\boldsymbol{f}}(\bar{z}) + \tilde{D}_{\Delta z}\tilde{\boldsymbol{f}} - \tilde{\boldsymbol{f}}(\bar{z})\right)^T\right] \tag{5.16}$$

$$\boldsymbol{P}_y = E\left[\left(\left(\sum_{p=1}^{n} \Delta z_p \mu_p \delta_p\right)\tilde{\boldsymbol{f}}(\bar{z})\right)\left(\left(\sum_{p=1}^{n} \Delta z_p \mu_p \delta_p\right)\tilde{\boldsymbol{f}}(\bar{z})\right)^T\right] \tag{5.17}$$

$$\boldsymbol{P}_y = \sum_{p=1}^{n}\left(\mu_p \delta_p \tilde{\boldsymbol{f}}(\bar{z})\right)\left(\mu_p \delta_p \tilde{\boldsymbol{f}}(\bar{z})\right)^T \tag{5.18}$$

Note that because of linear transformation the covariance of z becomes unity, i.e., $E\left[(\Delta z)(\Delta z)^T\right] = \boldsymbol{I}$. Now using the difference and average operators we get

$$\boldsymbol{P}_y = \tfrac{1}{4h^2}\sum_{p=1}^{n}\left(\tilde{\boldsymbol{f}}(\bar{z} + h\boldsymbol{e}_p) - \tilde{\boldsymbol{f}}(\bar{z} - h\boldsymbol{e}_p)\right)\left(\tilde{\boldsymbol{f}}(\bar{z} + h\boldsymbol{e}_p) - \tilde{\boldsymbol{f}}(\bar{z} - h\boldsymbol{e}_p)\right)^T \tag{5.19}$$

$$\boldsymbol{P}_y = \tfrac{1}{4h^2}\sum_{p=1}^{n}\left(\boldsymbol{f}(\bar{\boldsymbol{x}} + h\boldsymbol{s}_p) - \boldsymbol{f}(\bar{\boldsymbol{x}} - h\boldsymbol{s}_p)\right)\left(\boldsymbol{f}(\bar{\boldsymbol{x}} + h\boldsymbol{s}_p) - \boldsymbol{f}(\bar{\boldsymbol{x}} - h\boldsymbol{s}_p)\right)^T \tag{5.20}$$

$\boldsymbol{s}_p$ is the p^{th} column of $\boldsymbol{S}_x$

The expression for the cross covariance becomes

$$\boldsymbol{P}_{xy} = E\left[(\boldsymbol{x} - \bar{\boldsymbol{x}})\left(\tilde{\boldsymbol{f}}(\bar{z}) + \tilde{D}_{\Delta z}\tilde{\boldsymbol{f}} - \tilde{\boldsymbol{f}}(\bar{z})\right)^T\right] \tag{5.21}$$

$$\boldsymbol{P}_{xy} = E\left[\left(\sum_{p=1}^{n} \boldsymbol{s}_p \Delta z_p\right)\left(\left(\sum_{p=1}^{n} \Delta z_p \mu_p \delta_p\right)\tilde{\boldsymbol{f}}(\bar{z})\right)^T\right] \tag{5.22}$$

$$\boldsymbol{P}_{xy} = \tfrac{1}{2h}\sum_{p=1}^{n} \boldsymbol{s}_{x,p}\left(\boldsymbol{f}(\bar{\boldsymbol{x}} + h\boldsymbol{s}_p) - \boldsymbol{f}(\bar{\boldsymbol{x}} - h\boldsymbol{s}_p)\right)^T \tag{5.23}$$

5.3.2 Second order approximation

The nonlinear function $\boldsymbol{y} = \tilde{\boldsymbol{f}}(z)$ is presented with second order approximation as

$$\boldsymbol{y} = \tilde{\boldsymbol{f}}(\bar{z} + \Delta z) = \tilde{\boldsymbol{f}}(\bar{z}) + \tilde{D}_{\Delta z}\tilde{\boldsymbol{f}} + \tfrac{1}{2}\tilde{D}^2_{\Delta z}\tilde{\boldsymbol{f}} \tag{5.24}$$

$$\boldsymbol{y} = \tilde{\boldsymbol{f}}(\bar{z}) + \tfrac{1}{h}\left(\sum_{p=1}^{n} \Delta z_p \mu_p \delta_p\right)\tilde{\boldsymbol{f}}(\bar{z}) + \tfrac{1}{2h^2}\left(\sum_{p=1}^{n} \Delta z_p^2 \delta_p^2 + \sum_{p=1}^{n}\sum_{\substack{q=1\\ q\neq p}}^{n} \Delta z_p \Delta z_q (\mu_p\delta_p)(\mu_q\delta_q)\right)\tilde{\boldsymbol{f}}(\bar{z}) \tag{5.25}$$

The mean of transformed variable becomes

$$\bar{\boldsymbol{y}} = E\left[\tilde{\boldsymbol{f}}(\bar{z}) + \tfrac{1}{h}\left(\sum_{p=1}^{n} \Delta z_p^2 \delta_p^2\right)\tilde{\boldsymbol{f}}(\bar{z})\right] \tag{5.26}$$

$$\bar{\boldsymbol{y}} = \tilde{\boldsymbol{f}}(\bar{z}) + \tfrac{\sigma_2}{2}\sum_{p=1}^{n} \delta_p^2 \tilde{\boldsymbol{f}}(\bar{z}) \tag{5.27}$$

σ_2 is the covariance of arbitrary element Δz_p of $\boldsymbol{\Delta z}$ and $\sigma_2 = 1$. Using the difference operator we get

$$\bar{\boldsymbol{y}} = \tfrac{h^2 - n}{h^2}\boldsymbol{f}(\bar{\boldsymbol{x}}) + \tfrac{1}{2h^2}\sum_{p=1}^{n}\left(\boldsymbol{f}(\bar{\boldsymbol{x}} + h\boldsymbol{s}_p) - \boldsymbol{f}(\bar{\boldsymbol{x}} - h\boldsymbol{s}_p)\right) \tag{5.28}$$

Note that $\boldsymbol{\Delta z}$ being zero mean we can neglect $\tilde{D}_{\Delta z}\tilde{\boldsymbol{f}}$.

The covariance of $\boldsymbol{y}$

$$\boldsymbol{P}_y = E\left[\left(\tilde{D}_{\Delta z}\tilde{\boldsymbol{f}} + \tfrac{1}{2}\tilde{D}^2_{\Delta z}\tilde{\boldsymbol{f}}\right)\left(\tilde{D}_{\Delta z}\tilde{\boldsymbol{f}} + \tfrac{1}{2}\tilde{D}^2_{\Delta z}\tilde{\boldsymbol{f}}\right)^T\right] - E\left[\left(\tilde{D}_{\Delta z}\tilde{\boldsymbol{f}} + \tfrac{1}{2}\tilde{D}^2_{\Delta z}\tilde{\boldsymbol{f}}\right)\right]E\left[\left(\tilde{D}_{\Delta z}\tilde{\boldsymbol{f}} + \tfrac{1}{2}\tilde{D}^2_{\Delta z}\tilde{\boldsymbol{f}}\right)^T\right] \tag{5.29}$$

For computation of covariance with second order terms following steps are to be considered. As per the linear transformation the transformed variable becomes uncorrelated. Therefore, the cross differences are discarded. The following expectations (as given in [Norgaard2000]) are required to compute (5.29)

$$E\left[\left(\Delta z_i^2 \delta_i^2 \tilde{\boldsymbol{f}}\right)\left(\Delta z_i^2 \delta_i^2 \tilde{\boldsymbol{f}}\right)^T\right] = \left(\delta_i^2 \tilde{\boldsymbol{f}}\right)\left(\delta_i^2 \tilde{\boldsymbol{f}}\right)^T \sigma_4 \tag{5.30}$$

$$E\left[\left(\Delta z_i^2 \delta_i^2 \tilde{\boldsymbol{f}}\right)\left(\Delta z_j^2 \delta_j^2 \tilde{\boldsymbol{f}}\right)^T\right] = \left(\delta_i^2 \tilde{\boldsymbol{f}}\right)\left(\delta_j^2 \tilde{\boldsymbol{f}}\right)^T \sigma_2^2 \tag{5.31}$$

$$E\left[\left(\Delta z_i^2 \delta_i^2 \tilde{\boldsymbol{f}}\right)\right]E\left[\left(\Delta z_i^2 \delta_i^2 \tilde{\boldsymbol{f}}\right)^T\right] = \left(\delta_i^2 \tilde{\boldsymbol{f}}\right)\left(\delta_i^2 \tilde{\boldsymbol{f}}\right)^T \sigma_2^2 \tag{5.32}$$

$$E\left[\left(\Delta z_i^2 \delta_i^2 \tilde{\boldsymbol{f}}\right)\right]E\left[\left(\Delta z_j^2 \delta_j^2 \tilde{\boldsymbol{f}}\right)^T\right] = \left(\delta_i^2 \tilde{\boldsymbol{f}}\right)\left(\delta_j^2 \tilde{\boldsymbol{f}}\right)^T \sigma_2^2 \tag{5.33}$$

Here, σ_4 and σ_2 denotes 4th moment and 2nd moment of the distribution respectively. In [Norgaard2000] it is reported that for optimal choice of interval length h, square of h should be equal to the kurtosis of the distribution. For Gaussian distribution $\sigma_4 = h^2$ and $\sigma_2 = 1$. Using these values $\boldsymbol{P}_y$ is obtained as

$$\boldsymbol{P}_y = \sum_{p=1}^{n} \left(\mu_p \delta_p \tilde{\boldsymbol{f}}(\bar{z})\right)\left(\mu_p \delta_p \tilde{\boldsymbol{f}}(\bar{z})\right)^T + \tfrac{\sigma_4 - 1}{4} \sum_{p=1}^{n} \left(\delta_p^2 \tilde{\boldsymbol{f}}(\bar{z})\right)\left(\delta_p^2 \tilde{\boldsymbol{f}}(\bar{z})\right)^T \sigma_4 \tag{5.34}$$

Using the average and difference operator $\boldsymbol{P}_y$ is expressed as

$$\begin{aligned} \boldsymbol{P}_y &= \tfrac{1}{4h^2} \sum_{p=1}^{n} \left(\boldsymbol{f}(\bar{\boldsymbol{x}} + h\boldsymbol{s}_p) - \boldsymbol{f}(\bar{\boldsymbol{x}} - h\boldsymbol{s}_p)\right)\left(\boldsymbol{f}(\bar{\boldsymbol{x}} + h\boldsymbol{s}_p) - \boldsymbol{f}(\bar{\boldsymbol{x}} - h\boldsymbol{s}_p)\right)^T \\ &+ \tfrac{h^2 - 1}{4h^4} \sum_{p=1}^{n} \left(\boldsymbol{f}(\bar{\boldsymbol{x}} + h\boldsymbol{s}_p) + \boldsymbol{f}(\bar{\boldsymbol{x}} - h\boldsymbol{s}_p) - 2\boldsymbol{f}(\bar{\boldsymbol{x}})\right)\left(\boldsymbol{f}(\bar{\boldsymbol{x}} + h\boldsymbol{s}_p) + \boldsymbol{f}(\bar{\boldsymbol{x}} - h\boldsymbol{s}_p) - 2\boldsymbol{f}(\bar{\boldsymbol{x}})\right)^T \end{aligned} \tag{5.35}$$

The cross covariance $\boldsymbol{P}_{xy}$ remains the same as in the case of first order approximation.

5.4 Algorithm for Adaptive Divided Difference Filter

The algorithm for Adaptive Divided Difference filter intended for nonlinear state estimation problem (described in chapter 4) is presented here. The algorithm is presented in two subsections. At first the algorithm for non-adaptive DDF is presented which has been used as the core of ADDF. The $\boldsymbol{Q}$ and $\boldsymbol{R}$ adaptation algorithms are provided in the succeeding section.

5.4.1 Underlying framework of non-adaptive DDF

The first part of the adaptive DFF i.e., its non-adaptive part is presented in the standard error covariance form which is an alternative form the algorithm presented in [Norgaard2000] based on square root approach. Suitable modifications are made to make it compatible for the adaptation algorithms described in following subsection.

(i) ***Filter Initialization:*** Initialize $\hat{\boldsymbol{x}}_0, \hat{\boldsymbol{P}}_0, \bar{\boldsymbol{Q}}, \bar{\boldsymbol{R}}$

(ii) ***Time Update Steps:***

Given $\bar{\boldsymbol{P}}_{k-1}$, compute the Cholesky Factor $\bar{\boldsymbol{S}}_x(\mathrm{k}-1)$ such that $\bar{\boldsymbol{P}}_k = \bar{\boldsymbol{S}}_x(\mathrm{k}-1)\bar{\boldsymbol{S}}_x^T(\mathrm{k}-1)$

Propagation of a priori estimate of state:

$$\bar{\boldsymbol{x}}_k = \tfrac{h^2-n}{h^2} \boldsymbol{f}(\hat{\boldsymbol{x}}_{k-1}) + \tfrac{1}{2h^2} \sum_{p=1}^{n} \left\{ \boldsymbol{f}\left(\hat{\boldsymbol{x}}_{k-1} + h\hat{\boldsymbol{s}}_{x,p}\right) + \boldsymbol{f}\left(\hat{\boldsymbol{x}}_{k-1} - h\hat{\boldsymbol{s}}_{x,p}\right)\right\} \tag{5.36}$$

where $\hat{s}_{x,p}$ is p^{th} column of $\hat{\boldsymbol{S}}_x(\mathrm{k}-1)$ and h is the appropriately chosen interval length ($h=\sqrt{3}$ for Gaussian distribution).

Propagation of a priori error covariance:

The *a priori* error covariance becomes

$$\bar{\boldsymbol{P}}_k = \boldsymbol{S}^{(1)}_{x\hat{x}}(\mathrm{k})\left(\boldsymbol{S}^{(1)}_{x\hat{x}}(\mathrm{k})\right)^T + \boldsymbol{S}^{(2)}_{x\hat{x}}(\mathrm{k})\left(\boldsymbol{S}^{(2)}_{x\hat{x}}(\mathrm{k})\right)^T + \overline{\boldsymbol{Q}} \tag{5.37}$$

$\boldsymbol{S}^{(1)}_{x\hat{x}}(\mathrm{k})$ and $\boldsymbol{S}^{(2)}_{x\hat{x}}(\mathrm{k})$ are the first order and the second order approximation of the square root matrix of *a priori* error covariance. The elements of these matrices are obtained from (5.38) and (5.39) for *i=1,…,n* and *j=1,…,n*.

$$\boldsymbol{S}^{(1)}_{x\hat{x}}(\mathrm{k})_{(i,j)} = \tfrac{1}{2h}\left(\boldsymbol{f}_i\left(\hat{\boldsymbol{x}}_{k-1} + h\hat{\boldsymbol{s}}_{x,j}\right) - \boldsymbol{f}_i\left(\hat{\boldsymbol{x}}_{k-1} - h\hat{\boldsymbol{s}}_{x,j}\right)\right) \tag{5.38}$$

$$\boldsymbol{S}^{(2)}_{x\hat{x}}(\mathrm{k})_{(i,j)} = \tfrac{\sqrt{h^2-1}}{2h^2}\left(\left(\boldsymbol{f}_i\left(\hat{\boldsymbol{x}}_{k-1} + h\hat{\boldsymbol{s}}_{x,j}\right) + \boldsymbol{f}_i\left(\hat{\boldsymbol{x}}_{k-1} - h\hat{\boldsymbol{s}}_{x,j}\right) - 2\boldsymbol{f}_i\left(\hat{\boldsymbol{x}}_{k-1}\right)\right)\right) \tag{5.39}$$

Here $\boldsymbol{S}^{(1)}_{x\hat{x}}(\mathrm{k})_{(i,j)}$ and $\boldsymbol{S}^{(2)}_{x\hat{x}}(\mathrm{k})_{(i,j)}$ indicate the element s_{ij} of $\boldsymbol{S}^{(1)}_{x\hat{x}}(\mathrm{k})$ and $\boldsymbol{S}^{(2)}_{x\hat{x}}(\mathrm{k})$ respectively.

(iii) ***Measurement Update Steps:***

Given $\bar{\boldsymbol{P}}_k$,compute the Cholesky Factor $\bar{\boldsymbol{S}}_x(\mathrm{k})$ such that $\bar{\boldsymbol{P}}_k = \bar{\boldsymbol{S}}_x^T(\mathrm{k})\bar{\boldsymbol{S}}_x(\mathrm{k})$

Propagation of a priori estimate of measurement:

The expression of the *a priori* estimate of the measurement is similar to (5.36). Only the nonlinear measurement equation $\boldsymbol{g}(\cdot)$ has to be used in place of dynamics equation $\boldsymbol{f}(\cdot)$ of the system.

$$\bar{\boldsymbol{y}}_k = \tfrac{h^2-n}{h^2} \boldsymbol{g}(\bar{\boldsymbol{x}}_k) + \tfrac{1}{2h^2} \sum_{p=1}^{n} \left\{ \boldsymbol{g}\left(\bar{\boldsymbol{x}}_k + h\bar{\boldsymbol{s}}_{x,p}\right) + \boldsymbol{g}\left(\bar{\boldsymbol{x}}_k - h\bar{\boldsymbol{s}}_{x,p}\right)\right\} \tag{5.40}$$

Propagation of Innovation Covariance:

The innovation covariance is computed using the following expression

$$\boldsymbol{P}^y_k = \boldsymbol{S}^{(1)}_{y\bar{x}}(\mathrm{k})\left(\boldsymbol{S}^{(1)}_{y\bar{x}}(\mathrm{k})\right)^T + \boldsymbol{S}^{(2)}_{y\bar{x}}(\mathrm{k})\left(\boldsymbol{S}^{(2)}_{y\bar{x}}(\mathrm{k})\right)^T + \overline{\boldsymbol{R}} \tag{5.41}$$

where

$$\boldsymbol{S}^{(1)}_{y\bar{x}}(\mathrm{k})_{(i,j)} = \tfrac{1}{2h}\left(\boldsymbol{g}_i\left(\bar{\boldsymbol{x}}_k + h\bar{\boldsymbol{s}}_{x,j}\right) - \boldsymbol{g}_i\left(\bar{\boldsymbol{x}}_k - h\bar{\boldsymbol{s}}_{x,j}\right)\right) \tag{5.42}$$

$$S_{y\bar{x}}^{(2)}(\text{k})_{(i,j)} = \frac{\sqrt{h^2-1}}{2h}\left(\left(g_i\left(\bar{x}_k + h\bar{s}_{x,j}\right) + g_i\left(\bar{x}_k - h\bar{s}_{x,j}\right) - 2g_i\left(\bar{x}_k\right)\right)\right) \tag{5.43}$$

$S_{y\bar{x}}^{(1)}(\text{k})$ and $S_{y\bar{x}}^{(2)}(\text{k})$ are the first and the second order approximation of the square root matrix of error covariance of *a priori* estimate of measurement. These are obtained from (5.42) to (5.43) for $i=1,\dots,m$ and $j=1,\dots,n.$, and analogous to those of the time update steps.

The cross covariance is computed as

$$P_k^{xy} = \left[\bar{S}_x(\text{k})\right]\left[S_{y\bar{x}}^{(1)}(\text{k})\right]^T \tag{5.44}$$

The filter gain K_k becomes

$$K_k = P_k^{xy}\left(P_k^y\right)^{-1} \tag{5.45}$$

The *a posteriori* estimate of state is given by

$$\hat{x}_k = \bar{x}_k + K_k\left(y_k - \bar{y}_k\right) \tag{5.46}$$

The expression of the *a posteriori* error covariance is presented by (47). The formula ensures the positive definiteness of $\hat{P}_k$

$$\hat{P}_k = S_y(\text{k})S_y^T(\text{k}) + K_k\,\bar{R}_k\,K_k^T \tag{5.47}$$

where $S_y(\text{k}) = \left[\bar{S}_x(\text{k}) - K_k S_{y\bar{x}}^{(1)} \quad K_k S_{y\bar{x}}^{(2)}\right]$ (5.48)

(iv) ***Recursion:*** Estimates for the subsequent steps is to be computed by repeating the time update and measurement update steps as given above for the required number of time steps starting from k =1. Note that in the non-adaptive DDF noise covariance are denoted as $\bar{Q}$ and $\bar{R}$. When $\bar{Q}$ is known and $\bar{R}$ is unknown, $\bar{Q}$ is replaced by true Q and $\bar{R}$ is replaced by $\hat{R}_{k-1}$, i.e., adapted R of previous instant. Reverse is the case when $\bar{Q}$ is unknown and $\bar{R}$ is known. The adaptation algorithms are presented in the following sections.

5.4.2 Algorithm for Q adaptation

For adaptation of process noise covariance the innovation covariance needs to be estimated from the sliding window as discussed in Q adaptation methods in chapter 4.

Computation of window estimated innovation covariance:

The innovation is defined as

$$\vartheta_k = y_k - \bar{y}_k \tag{5.49}$$

The estimated innovation covariance from sliding window with length L is obtained as

$$\hat{\boldsymbol{C}}_{\boldsymbol{k}}^{\vartheta} = \frac{1}{L}\sum_{j=k-L+1}^{k} \vartheta(j)\vartheta^{\boldsymbol{T}}(j) \tag{5.50}$$

Adaptation of process noise covariance:

Two different $\boldsymbol{Q}$ adaptation methods have been presented here.

For MLE based $\boldsymbol{Q}$ adaptation method as given by (4.18) in chapter 4 adapted $\boldsymbol{Q}$ is presented below

$$\hat{\boldsymbol{Q}}_{\boldsymbol{k}} = \boldsymbol{K}_{\boldsymbol{k}}\hat{\boldsymbol{C}}_{\boldsymbol{k}}^{\vartheta}\boldsymbol{K}_{\boldsymbol{k}}^{\boldsymbol{T}} \tag{5.51}$$

Following the scaling factor based $\boldsymbol{Q}$ adaptation (covariance matching method) as given by (4.45) in chapter 4 the adaptation steps are presented below.

Compute the scaling Factor:

$$\alpha_k = \frac{trace\left(\hat{\boldsymbol{C}}_{\vartheta}^{\boldsymbol{k}} - \boldsymbol{R}\right)}{trace\left(\left[\boldsymbol{S}_{\boldsymbol{y}\bar{\boldsymbol{x}}}^{(1)}(\mathrm{k}) \quad \boldsymbol{S}_{\boldsymbol{y}\bar{\boldsymbol{x}}}^{(2)}(\mathrm{k})\right]\left[\boldsymbol{S}_{\boldsymbol{y}\bar{\boldsymbol{x}}}^{(1)}(\mathrm{k}) \quad \boldsymbol{S}_{\boldsymbol{y}\bar{\boldsymbol{x}}}^{(2)}(\mathrm{k})\right]^{\boldsymbol{T}}\right)} \tag{5.52}$$

Adapted of process noise covariance at current instant is therefore given by

$$\hat{\boldsymbol{Q}}_{\boldsymbol{k}} = \hat{\boldsymbol{Q}}_{\boldsymbol{k-1}}\sqrt{\alpha_k} \tag{5.53}$$

5.4.3 Algorithm for R adaptation

For adaptation of measurement noise covariance either the innovation covariance or the residual covariance needs to be estimated from the sliding window as discussed in $\boldsymbol{R}$ adaptation methods in chapter 4. The present worker has opted for residual based $\boldsymbol{R}$ adaptation for its additional advantaged of ensured positive definiteness of $\boldsymbol{R}$ matrix.

Given $\hat{\boldsymbol{P}}_{\boldsymbol{k}}$, compute the Cholesky Factor $\hat{\boldsymbol{S}}_{\boldsymbol{x}}(\mathrm{k})$ such that such that $\hat{\boldsymbol{P}}_{\boldsymbol{k}} = \hat{\boldsymbol{S}}_x(\mathrm{k})\,\hat{\boldsymbol{S}}_x^T(\mathrm{k})$

Propagation of a posteriori estimate of measurement:

$$\hat{\boldsymbol{y}}_{\boldsymbol{k}} = \frac{h^2-n}{h^2}\boldsymbol{g}(\hat{\boldsymbol{x}}_{\boldsymbol{k}}) + \frac{1}{2h^2}\sum_{p=1}^{n}\left\{\boldsymbol{g}\left(\hat{\boldsymbol{x}}_{\boldsymbol{k}} + h\hat{\boldsymbol{s}}_{\boldsymbol{x,p}}\right) + \boldsymbol{g}\left(\hat{\boldsymbol{x}}_{\boldsymbol{k}} - h\hat{\boldsymbol{s}}_{\boldsymbol{x,p}}\right)\right\} \tag{5.54}$$

This step is similar to (4.40). Only the Cholesky factor of *a priori* error covariance is replaced by the *a posteriori* one.

Propagation of error covariance of a posteriori estimate of measurement:

The elements of the error covariance of *a posteriori* estimate of measurement are obtained in a similar approach like (5.42) and (5.43).

$$\boldsymbol{S}^{(1)}_{y\hat{x}}(\mathrm{k})_{(i,j)} = \tfrac{1}{2h}\left(\boldsymbol{g}_i\left(\hat{\boldsymbol{x}}_k + h\hat{\boldsymbol{s}}_{x,j}\right) - \boldsymbol{g}_i\left(\hat{\boldsymbol{x}}_k - h\hat{\boldsymbol{s}}_{x,j}\right)\right) \tag{5.55}$$

$$\boldsymbol{S}^{(2)}_{y\hat{x}}(\mathrm{k})_{(i,j)} = \tfrac{\sqrt{h^2-1}}{2h}\left(\left(\boldsymbol{g}_i\left(\hat{\boldsymbol{x}}_k + h\hat{\boldsymbol{s}}_{x,j}\right) + \boldsymbol{g}_i\left(\hat{\boldsymbol{x}}_k - h\hat{\boldsymbol{s}}_{x,j}\right) - 2\boldsymbol{g}_i\left(\hat{\boldsymbol{x}}_k\right)\right)\right) \tag{5.56}$$

$\boldsymbol{S}^{(1)}_{y\hat{x}}(\mathrm{k})$ and $\boldsymbol{S}^{(2)}_{y\hat{x}}(\mathrm{k})$ are the first and the second order approximation of the error covariance of *a posteriori* estimate of measurement.

Computation of estimated residual covariance:

The residual is defined as the difference between the actual measurement and the *a posteriori* estimate of measurement, i.e.,

$$\boldsymbol{\rho}_k = \boldsymbol{y}_k - \hat{\boldsymbol{y}}_k \tag{5.57}$$

The estimated residual covariance can be computed from a sliding window (size L) as

$$\hat{\boldsymbol{C}}^{\rho}_k = \frac{1}{L}\sum_{j=k-L+1}^{k} \boldsymbol{\rho}(j)\boldsymbol{\rho}^T(j) \tag{5.58}$$

Adaptation of measurement noise covariance:

The adapted measurement noise covariance is obtained based on the covariance matching method and given by (4.58) in chapter 4. The adapted $\boldsymbol{R}$ is presented by (5.59) given below. As formulated on the basis of residual sequence the expression of adapted $\hat{\boldsymbol{R}}_k$ ensures positive definiteness.

$$\hat{\boldsymbol{R}}_k = \boldsymbol{S}^{(1)}_{y\hat{x}}(\mathrm{k})\left(\boldsymbol{S}^{(1)}_{y\hat{x}}(\mathrm{k})\right)^T + \boldsymbol{S}^{(2)}_{y\hat{x}}(\mathrm{k})\left(\boldsymbol{S}^{(2)}_{y\hat{x}}(\mathrm{k})\right)^T + \hat{\boldsymbol{C}}^{\rho}_k \tag{5.59}$$

5.5 Notes on algorithm

- The algorithm is presented here in standard error covariance form. However, on availability of the square root of the error covariances and noise covariances in the intermediate steps of the proposed algorithm, it can be easily extended in square root approach. In that case for updating the square root of error covariances QR factorization method has to be followed and instead of the noise covariance its square root has to be adapted. The square root version for ADDF may be formulated consulting the general algorithm in square root version presented in chapter 4.

- Note that only residual based ***R*** adaptation algorithm has been presented because of its additional advantage of ensured positive definiteness of adapted ***R*** matrix. The innovation based ***R*** adaptation algorithm can be obtained using window estimated innovation covariance in place of residual covariance. This covariance can be obtained replacing $\hat{\boldsymbol{y}}_k$ by $\bar{\boldsymbol{y}}_k$ in equation (57). Then the expression of adapted ***R*** can be obtained by subtracting $\boldsymbol{S}^{(1)}_{y\bar{x}}(\mathrm{k})\left(\boldsymbol{S}^{(1)}_{y\bar{x}}(\mathrm{k})\right)^T$ and $\boldsymbol{S}^{(2)}_{y\bar{x}}(\mathrm{k})\left(\boldsymbol{S}^{(2)}_{y\bar{x}}(\mathrm{k})\right)^T$ from the window estimated innovation covariance given by (5.50).
- On availability of sufficient computation power, the measurement update step and both the time and measurement update steps can be recomputed using the adapted value of ***R*** and ***Q*** respectively.
- The algorithm of adaptive Central Difference filter can be readily obtained from the algorithm presented here. The non-adaptive version of Central Difference filter [Schei1997] considers only first order approximation of interpolation formula. Therefore, for obtaining its adaptive version second order approximation terms should be ignored from the present adaptation algorithm and also from the underlying framework. Moreover, the length of the interval, h is to be chosen as $h=1$ as in [Schei1997]. For some estimation problem, specifically with high measurement covariance and low process noise covariance, Central Difference filter perform comparably same with DDF [Simandl2009]. For those cases, use of Adaptive Central Difference filter is preferred over ADDF as it is computationally economic compared to ADDF. The algorithm for ***R*** adaptive Central Difference filter has not been presented here as this algorithm can be readily derived from the proposed algorithm for ADDF with the above referred simplifying steps. Algorithmic steps for ACDF are elaborately presented in journal paper contributed by the present worker as a co-author.

5.6 Characterization of Adaptive DDF

The ***Q*** and ***R*** adaptive DDF algorithms are validated in this chapter using different nonlinear estimation problems. Here two different ***Q*** and ***R*** adaptation algorithms viz., direct adaptation and scaling factor based adaptation have been compared. Performance of scaling factor based

Q adaptive DDF proposed in this chapter has been demonstrated and compared with the direct ***Q*** adaptation algorithm. Performance of ***R*** adaptive DDF based on direct ***R*** adaptation is demonstrated and also compared with scaling factor based ***R*** adaptation algorithm.

5.6.1 Validation of Q adaptive DDF

5.6.1.1. Estimation of the states of Van der Pol's oscillator

For evaluation of scaling factor based ***Q*** adaptive DDF this case study has been considered. States and the friction coefficient of a Van der Pol's oscillator have to be estimated in this problem. The friction coefficient remains unknown here. The system dynamics, measurement equation and all the necessary parameters for simulation are provided in chapter 3.

The performance of scaling factor based ***Q*** adaptive DDF has been evaluated from the RMS error analysis. RMS error of the proposed QA-DDF and its respective non-adaptive version has been obtained from Monte Carlo simulation with 1000 runs. Both the filters are initialized with an assumed value of ***Q*** (500 times higher than the truth value) because of the unavailability of the knowledge of ***Q***. as it remains unavailable for this problem. The higher value of ***Q*** is chosen to induce sufficient uncertainty in initialization of ***Q***. To generate the true state trajectories of the oscillator, the initial kinematic states and truth value of friction coefficient and other parameters are chosen as given in chapter 3. The measurement equation is linear and the noise statistics is considered to be known. For adaptation the length of sliding window is considered to be 50 time instants. Adaptation does not begin till the length of innovation sequence is less than the desired window length.

Fig. 5.1 – 5.3 present the RMSE performance of the filters. From the plots of RMS errors it is observed that despite improper choice of ***Q*** matrix ADDF indicates lower RMS errors that converge within a less time while compared to non-adaptive DDF. The results also indicate that ADDF can accommodate large error in the initial choice of ***Q*** and capable of producing reliable estimation although initialized with wrong ***Q***.

The adapted ***Q*** gradually approach to the truth value of ***Q*** and continues to track that after converging on it. Fig 5.4 – 5.6 illustrates the ***Q*** adaptation performance.

In fig 5.7 a representative phase portrait of system states is illustrated where it is observed that estimated state trajectories reaches true Limit cycle in less time in case of ADDF as compared with non-adaptive DDF.

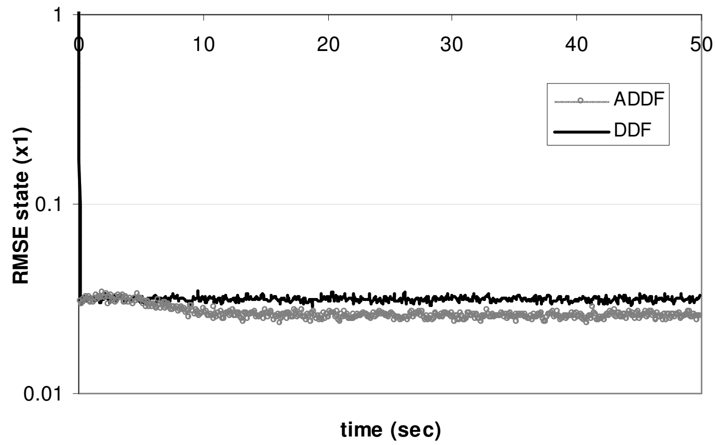


Fig. 5.1: RMSE of state (x_1) for 1000 Monte Carlo run

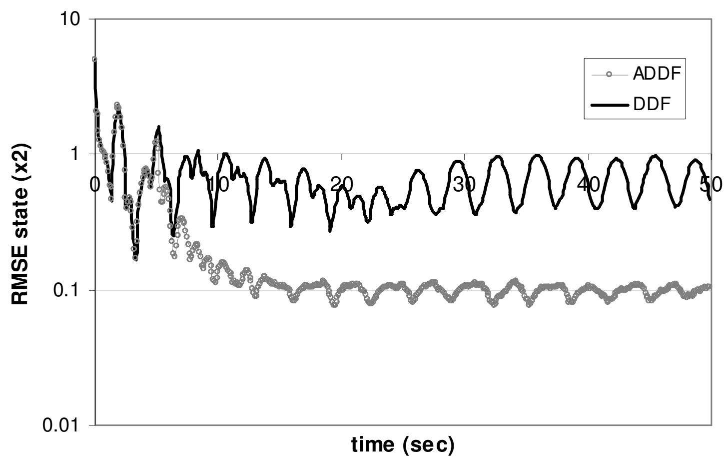


Fig. 5.2: RMSE of state (x_2) for 1000 Monte Carlo run

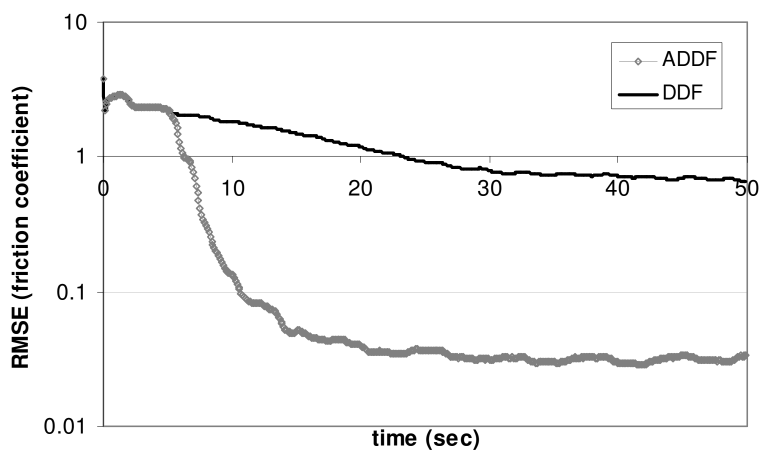


Fig. 5.3: RMSE of friction coefficient (parameter) for 1000 Monte Carlo run

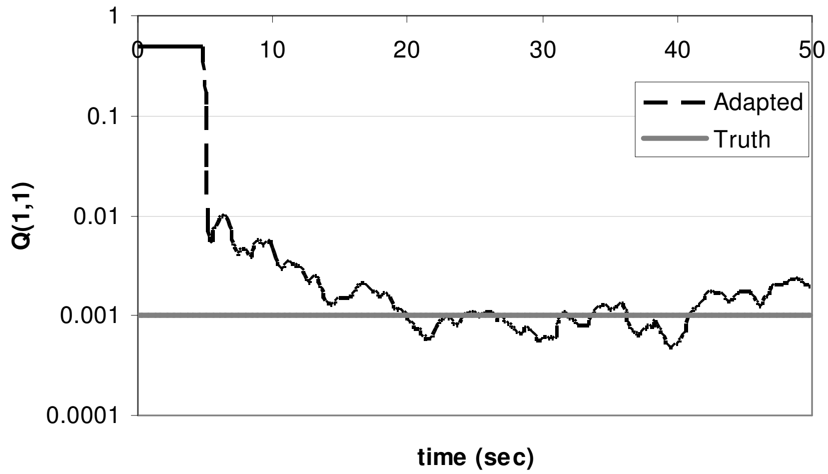


Fig. 5.4: True and adapted $\boldsymbol{Q}(1,1)$ for a representative run

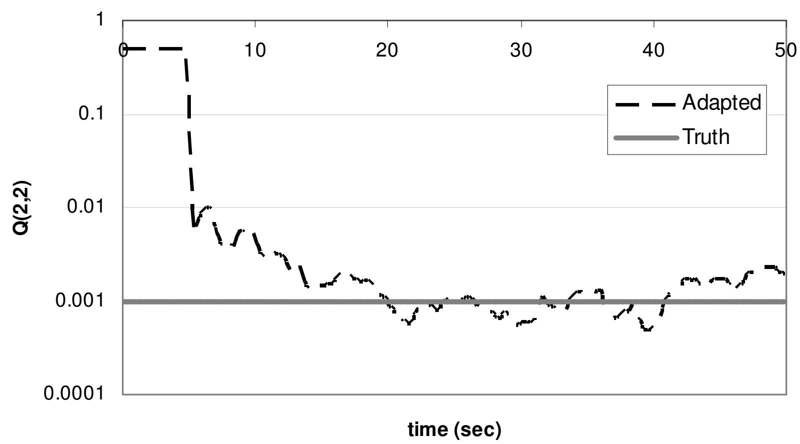


Fig. 5.5: True and adapted $\boldsymbol{Q}(2,2)$ for a representative run

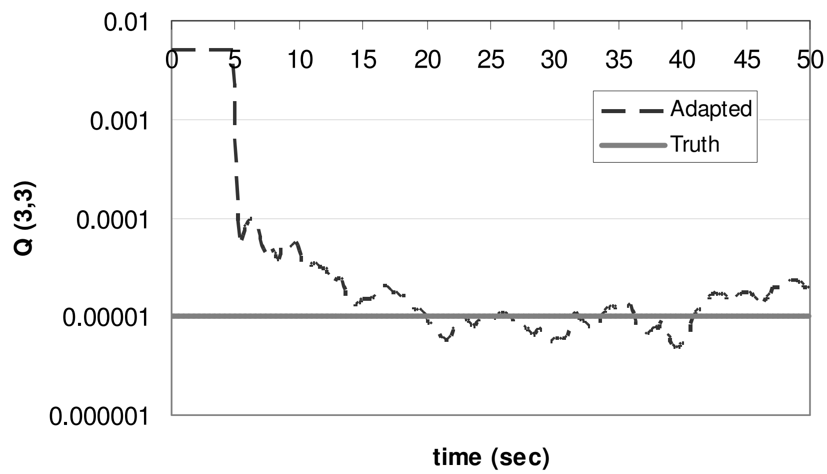


Fig. 5.6: True and adapted $\boldsymbol{Q}(3,3)$ for a representative run

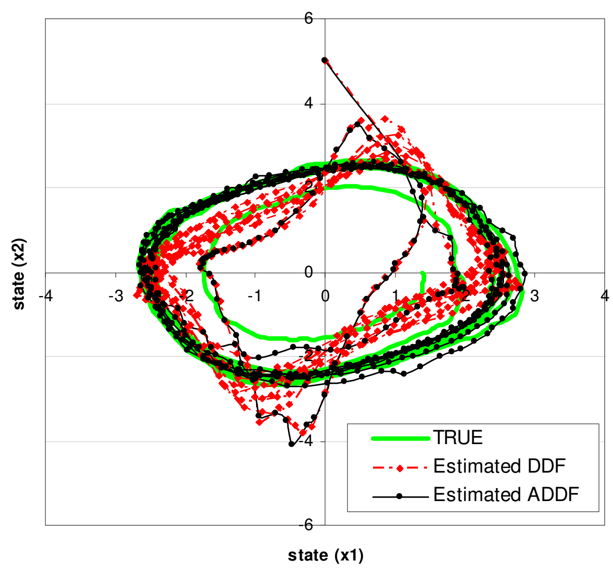


Fig. 5.7: Phase plane plot for a representative run

5.6.1.2. Object tracking problem

The task of joint estimation of ballistic parameter and states of an object during reentry phase using radar signal is addressed in this section. The ballistic object is considered to be falling vertically and tracked by radar which provides the range measurements of the object tracked by the radar. The description of system dynamics and observation equations are provided in chapter 3. In this case study emphasis is given on the performance analysis of ADDF with direct ***Q*** adaptation, ADDF with scaling factor based ***Q*** adaptation. In the situation when process noise covariance remains unavailable ***Q*** is initialized arbitrarily with an assumed value which is five decades higher than the truth value. The window size for adaptation is considered to be 10 time instants and adaptation does not begin till the length of innovation sequence exceeds the desired window length. Rest of the parameters for simulation are provided in chapter 3.

The results of Monte Carlo simulation (with 1000 runs) in terms of RMS errors of ADDF with direct ***Q*** adaptation, ADDF with scaling factor based tuning of ***Q*** and non-adaptive DDF for all the state variables and the ballistic parameter are presented by Fig. 5.8, Fig. 5.9, and Fig. 5.10.

The most significant finding is that the RMSE performance of the scaling factor based ***Q*** adaptive DDF even though found to work satisfactorily for the previous case study is considerably deteriorated while compared to ADDF with the direct ***Q*** adaptation algorithm for this estimation problem. The parameter as well as state estimates from ADDF with scaling factor based ***Q*** adaptation are showing a tendency of divergence while those for ADDF with the direct ***Q*** adaptation have been adequately converged.

In presence of nonlinear measurement equation performance of scaling factor based algorithm deteriorates because of the fact that the approximations made in the adaptation algorithmic do not hold well for nonlinear measurement equation and consequently cannot adapt ***Q*** satisfactorily. On contrary it is observed in case of direct ***Q*** adaptation that this estimator ensures reasonably well estimation performance by satisfactory adaptation of ***Q***. Figure 5.11 is presented to compare the ***Q*** adaptation performance of both the adaptive filters. Plot for an element of adapted ***Q*** is presented for a representative run. It is illustrated for ADDF with direct ***Q*** adaptation that the elements of ***Q*** tries to approach the corresponding truth value. For the scaling factor based the same element of ***Q*** cannot converge on its truth value.

Although initialized with an assumed value of ***Q*** (3 decade higher than the true value) performance of ADDF with direct adaptation algorithm is comparably closer to that of the non-adaptive DDF in the ideal case when the process noise covariance is known to the latter. This validates the algorithm of ***Q*** adaptive DDF with direct adaptation method.

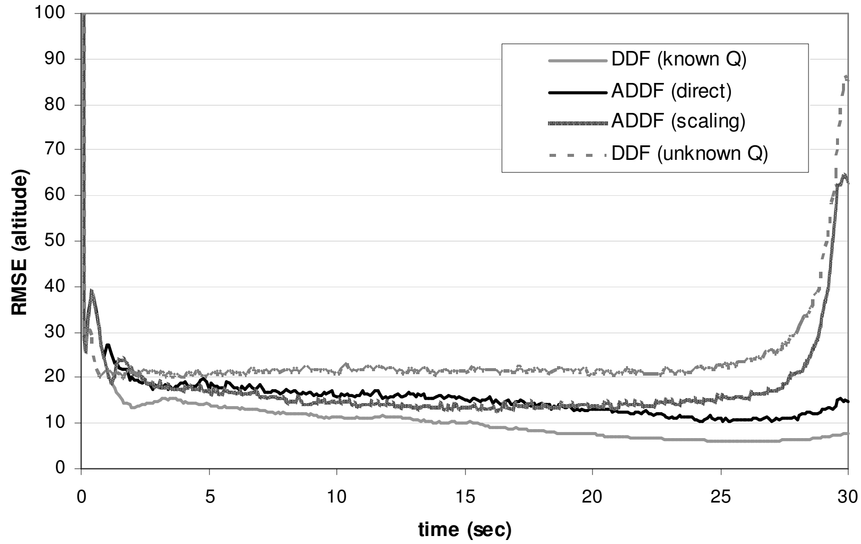


Fig. 5.8: RMSE of altitude for 1000 Monte Carlo run

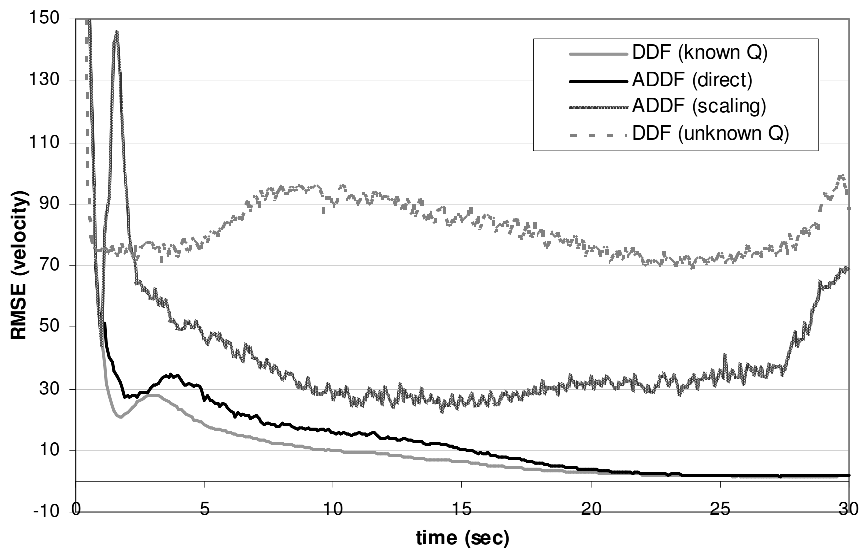


Fig. 5.9: RMSE of velocity for 1000 Monte Carlo run

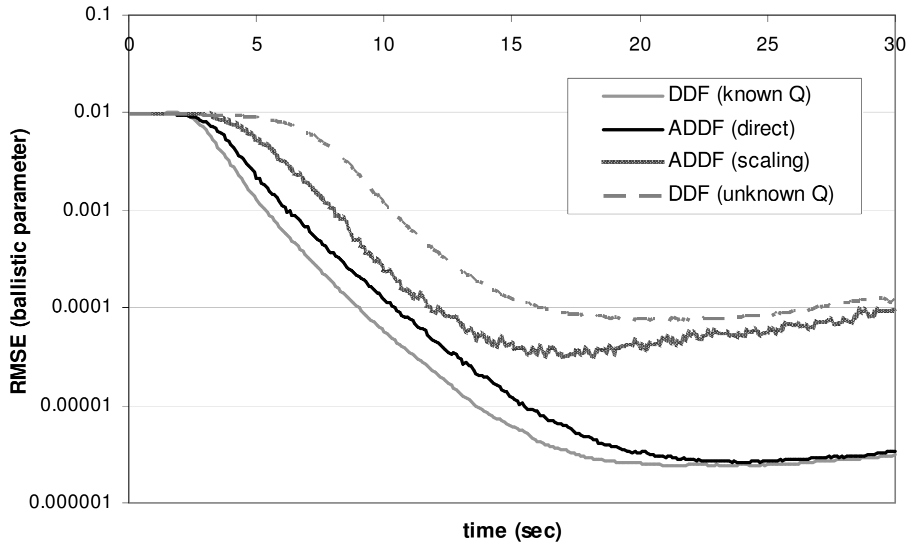


Fig. 5.10: RMSE of ballistic parameter for 1000 Monte Carlo run

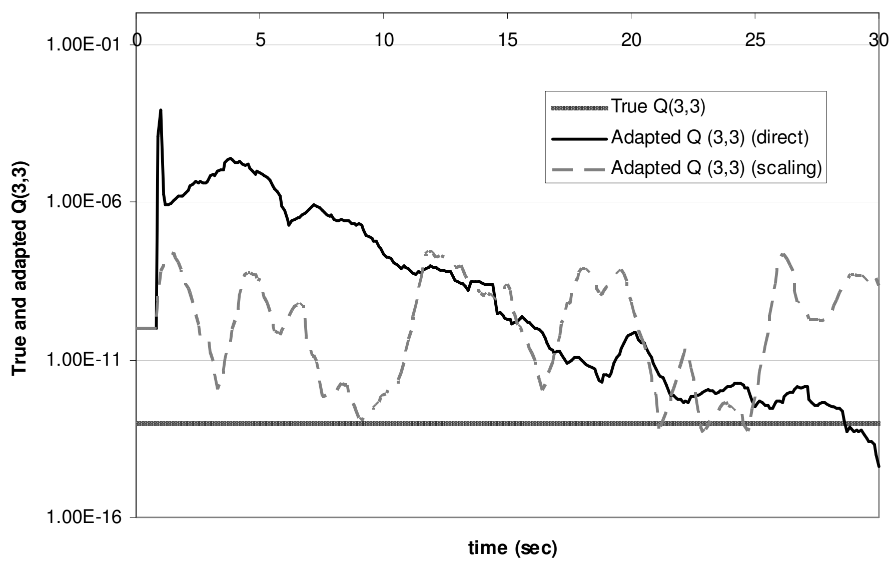


Fig. 5.11: True and adapted $\boldsymbol{Q}(3,3)$ for a representative run

5.6.2 Validation of R adaptive DDF

5.6.2.1. First order nonlinear problem

A single dimensional estimation problem is considered to evaluate the performance of $\boldsymbol{R}$ adaptive DDF in face of unknown measurement noise covariance. This estimation problem is an effective case study which can critically judge the performance of candidate estimator in presence of strong nonlinearity in the system dynamics and the measurement equation. The problem description can be found in chapter 3. Here, the performance of ADDF with direct ***R*** adaptation is compared with ADDF with a scaling factor based ***R*** adaptation (the adaptation method proposed in [Hajiyev2014] for linear signal models). For direct ***R*** adaptation residual sequence has been employed. Performance of Adaptive CDF and adaptive UKF with direct ***R*** adaptation (RA-CDF) algorithm has also been compared with proposed RA-DDF. The underlying framework of non-adaptive CDF is taken from [Schei1997].

Performance comparison is carried out on the basis of Monte Carlo study with 10000 runs. Due to the unavailability of measurement noise covariance it is assumed as three decades higher than the truth value of ***R***. The window length is taken as 100 time instants and adaptation begins from the very first time instants with available size of residual window till it attains the desired length. The other necessary parameters are provided in chapter 3. In Fig. 5.12 the plots of the RMS errors for the above referred estimators are presented. It is observed the RMSE of ADDF with direct ***R*** adaptation algorithm settles to a lower value compared to the scaling factor based algorithm. For validation of the direct ***R*** adaptive algorithm we have also compared its performance with non-adaptive DDF in the ideal case when the non-adaptive filter has the knowledge of ***R***. It is observed that despite the unavailability of the knowledge of ***R*** matrix performance RA-DDF (direct adaptation) is closely comparable with non-adaptive DDF in the ideal case. Performance of RA-DDF, as expected, is superior to non-adaptive DDF when ***R*** matrix remains unknown.

Performance of RA-DDF is also compared with RA-CDF, RA-UKF where all the estimators include direct ***R*** adaptation algorithm. For this case study RA-DDF shows considerably improved estimation performance compared to RA-CDF and RA-UKF.

Performance of all the candidate estimators is numerically compared on the basis of the percentage of track loss. For this case study the system has 2 stable and one unstable equilibrium points. Because of the nonlinearity in the observation equation the estimators even with known ***R*** may fail to track the true trajectory and get settled to incorrect equilibrium point. It is understood from the percentage of track loss that ADDF with direct ***R*** adaptation method is less susceptible to track losses among the other alternative adaptive filters. Percentage of track loss for RA-DDF with direct adaptation (24.26%) is a little higher than the non-adaptive DDF with known ***R*** (24.02%) which signifies that RA-DDF without knowledge of ***R*** ensures estimation performance close enough to that of the non-adaptive DDF in the ideal case. ADDF with scaling factor based ***R*** adaptation, RA-UKF and RA-CDF with direct adaptation algorithm are observed to be more susceptible to track loss as the percentage of track for these estimators are 30.23%, 93.91% and 36.8% respectively. Percentage of track is observed to be quite high for RA-UKF and non-adaptive DDF without knowledge of ***R*** (90.12%) compared to other estimators.

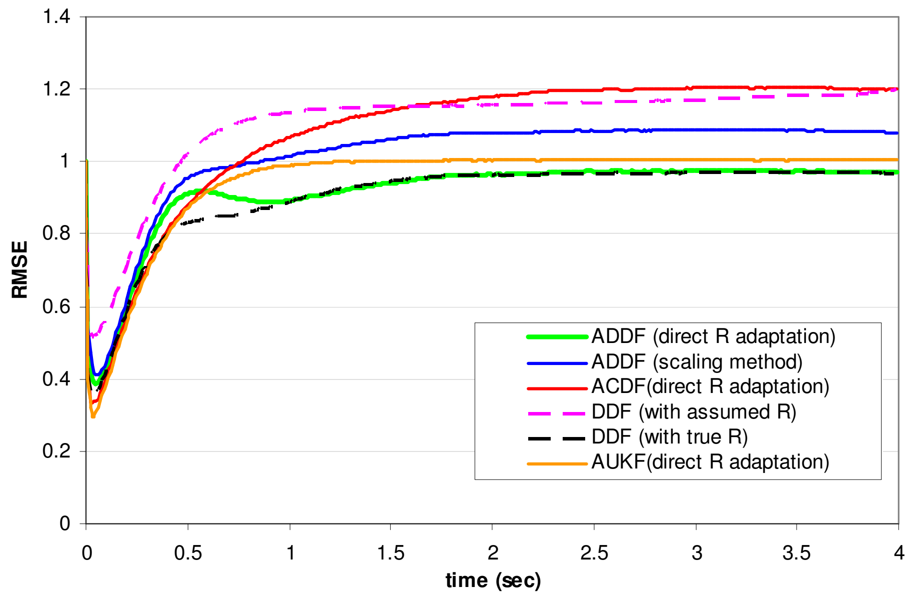


Fig. 5.12 RMSE of state for 10000 Monte Carlo run

5.6.2.2. Object tracking problem

The object tracking problem is revisited for demonstration of state and parameter estimation performance of ***R*** adaptive DDF. In the situation when measurement noise covariance remains unavailable ***R*** is initialized arbitrarily with an assumed value which is two decades higher than the truth value for this case study. The window length is taken as 100 time instants and adaptation begins from the first time instants with available size of residual window till it attains the desired length. For the performance analysis Monte Carlo simulation is carried out with 1000 run. Fig. 5.13 – 5.15 represents the of RMS errors of RA-DDF, RA-UKF, RA-CDF and non-adaptive DDF for all the state variables and the ballistic parameter.

Here also RA-DDF is validated by demonstrating that the RMSE performance of the proposed Adaptive DDF regardless of initialized with an assumed value of measurement noise covariance (2 decade higher than the true value) is comparable to that of the non-adaptive DDF with known (true) value of ***R*** matrix. In the situation when the knowledge of the measurement noise covariance is unavailable to the non-adaptive DDF the performance of RA-DDF is substantially improved compared to non-adaptive DDF.

On the plots of RMS errors the square roots of the corresponding diagonal elements of the *a posteriori* error covariance matrix have been super imposed. It is to be noted that the RMS errors obtained from RA-DDF almost retrace the square root of the respective diagonal elements of the *a posteriori* error covariance and are consistent with them.

The performance of RA-DDF while compared with a carefully tuned ***R*** adaptive UKF is found to be nearly identical to that of the latter. RA-CDF although found to be performance wise less accurate compared to RA-DDF in the previous case study demonstrates estimation performance comparable with RA-DDF for this case study. In the context it may be noted that the non-adaptive CDF is reported to present comparable estimation performance with UKF and DDF for the systems with high measurement noise covariance (with a comparatively low process noise covariance) and having quasi linear measurement equation [Simandl2009]. This may be reason behind the above observations for ***R*** Adaptive versions of DDF, UKF and CDF for this case study as the true measurement noise covariance is considerably high compared to the process noise covariance.

The tracking performance of RA-DDF is also demonstrated for the situation with non stationary measurement noise where the measurement noise covariance is unknown and time varying. Here the objective is to investigate how far the adapted value of ***R*** matches to that of the truth value of time varying ***R***. While investigating this performance the effect of window size is also explored. The ***R***-adaptation performance of RA-DDF is illustrated considering two different situations, (i) truth value of ***R*** exponentially decreases before saturating to a steady value, and (ii) truth value of ***R*** exponentially increases before saturating to a steady value. In all these cases the adapted ***R*** is initialized with values which are different from the true initial values (twice of the truth value). Fig. 5.16 and Fig. 5.17 present the ***R***-adaptation performance of the proposed filter with the effect of different choice of window size.

From Fig. 5.16 where the value of ***R*** is decaying it is observed that adapted ***R*** tends to be noisy for smaller window size (*N*=30). With a bigger window size (*N*=90) the adapted value, although smoother, takes longer time to reach the corresponding truth value. For a reasonable choice of window size (*N*=60) the performance is found to be acceptable.

In Fig. 17 where the value of ***R*** is increasing till it saturates to the maximum value it is observed that a smaller window size (*N*=30) speeds up the tracking performance of the adapted ***R*** at the expense of more oscillations. On contrary, negligible oscillations are observed for a larger window size (*N*=90). However, a delay of 25 sec is noted to reach the truth value. A moderate choice of window size (*N*=60) shows an acceptable performance.

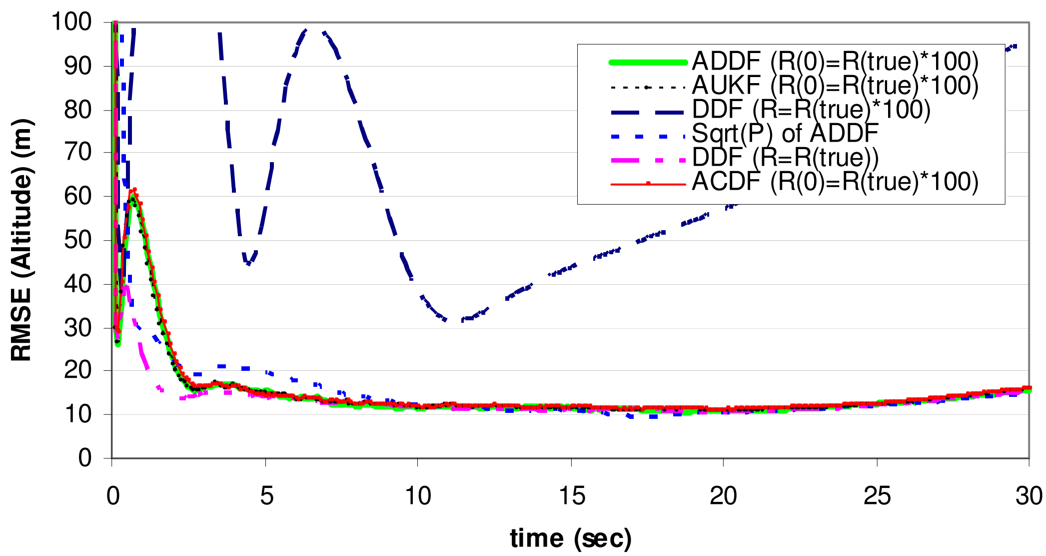


Fig. 5.13: RMSE of altitude for 1000 Monte Carlo run

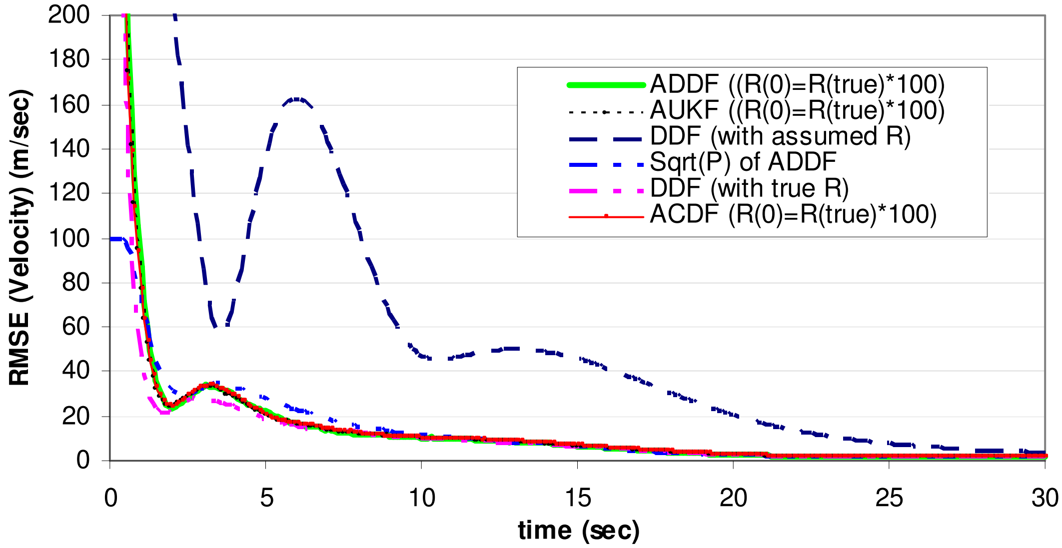


Fig. 5.14: RMSE of velocity for 1000 Monte Carlo run

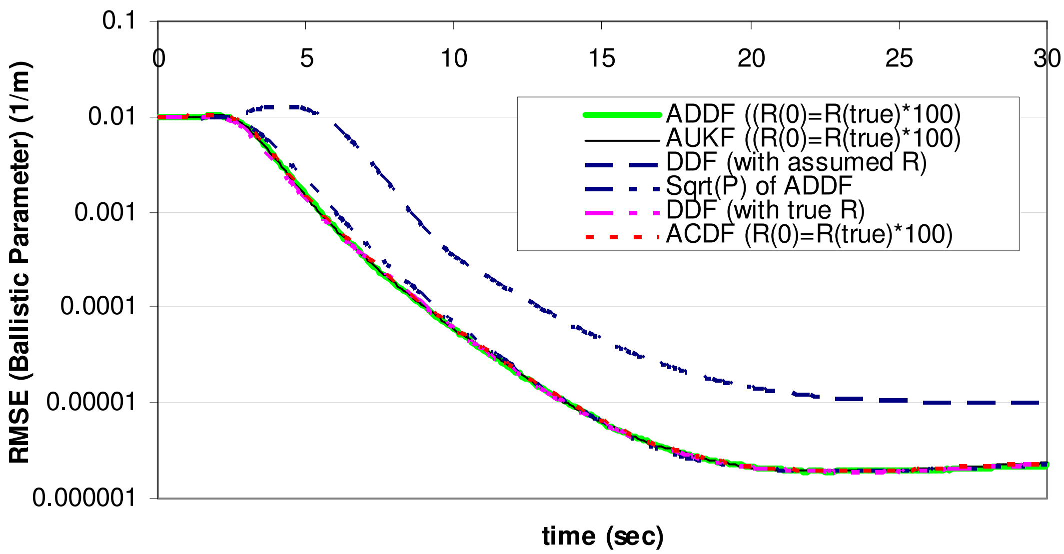


Fig. 5.15: RMSE of ballistic parameter for 1000 Monte Carlo run

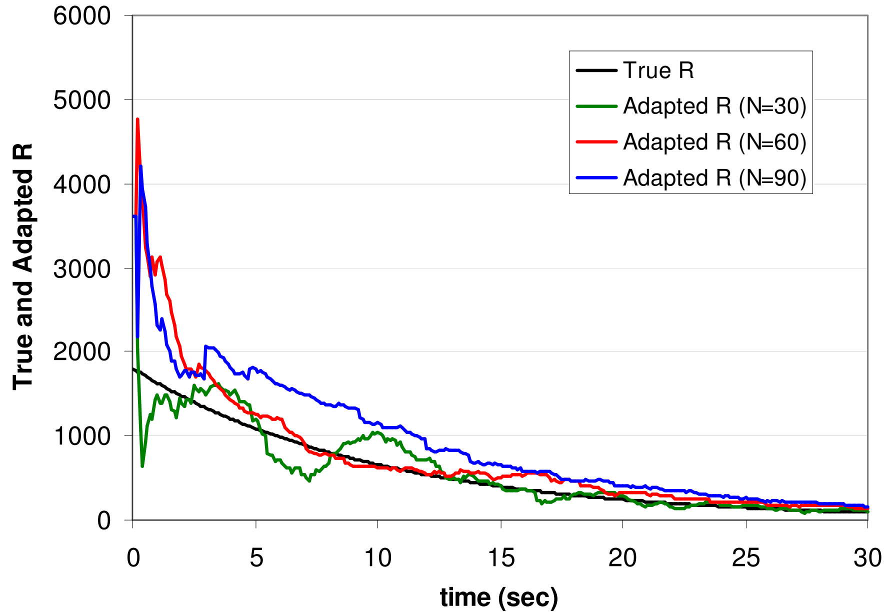


Fig. 5.16: Plot of true and adapted ***R*** for a time varying noise covariance

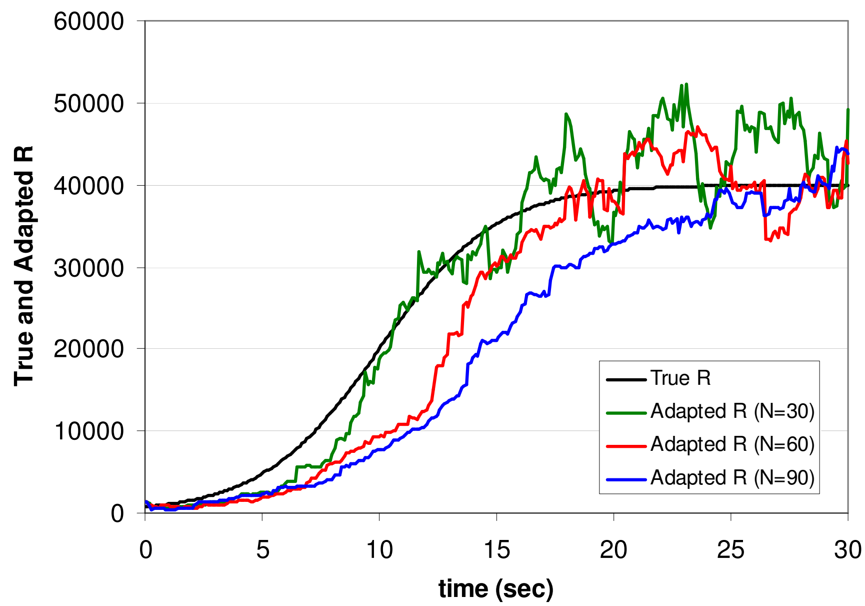


Fig. 5.17: Plot of true and adapted ***R*** for a time varying noise covariance

5.7 Discussions and Conclusions

In this chapter algorithms for $\boldsymbol{Q}$ and $\boldsymbol{R}$ adaptive Divided Difference filters have been presented and exemplified. Significant findings are enumerated as follows:

- In general performance of both $\boldsymbol{Q}$ and $\boldsymbol{R}$ adaptive DDF are observed to be demonstrably superior to their non-adaptive counterparts in each case study.
- The proposed algorithms of $\boldsymbol{Q}$ and $\boldsymbol{R}$ adaptive DDF in the face of unknown noise covariance are validated by comparing their performance with non-adaptive DDF in the ideal situation with full knowledge of process and measurement noise covariance. Despite the improper initial choice of noise covariance ($\boldsymbol{Q}$ or $\boldsymbol{R}$ depending on the unavailability) ADDF performs satisfactorily and its performance is closely comparable with non-adaptive DDF in ideal situation with known noise covariance.
- $\boldsymbol{Q}$ adaptive DDF with scaling factor based adaptation shows satisfactory estimation result for linear measurement equations. Nevertheless, it fails to produce acceptable estimation performance with nonlinear measurement equation. Performance of scaling factor based algorithm is compared with ADDF with direct $\boldsymbol{Q}$ adaptation algorithm. ADDF with direct $\boldsymbol{Q}$ adaptation algorithm is found to outperform the former and perform equally well with both linear and nonlinear measurements. For $\boldsymbol{R}$ adaptation also direct $\boldsymbol{R}$ adaptation algorithm shows its superiority over scaling factor based algorithm. Therefore, adaptive estimators with direct adaptation algorithms are preferred over the scaling factor based algorithm for nonlinear state estimation.
- ADDF with direct $\boldsymbol{R}$ adaptation algorithm is compared with corresponding $\boldsymbol{R}$ adaptive version of UKF and CDF. It is found that $\boldsymbol{R}$ adaptive DDF performs considerably better compared to other two competing estimators when system dynamics is significantly nonlinear. However their performance is observed to be closely comparable for another estimation problem.
- It is also observed that plots of RMSE for RA-DDF (with direct adaptation method) retrace the square root of the respective diagonal element of a posterior error covariance. This demonstrates the consistency of the adaptive DDF. For direct $\boldsymbol{R}$ adaptation residual sequence is employed as the residual based $\boldsymbol{R}$ adaptation algorithm automatically ensures the positive definiteness of adapted $\boldsymbol{R}$ matrix. The

direct ***R*** adaptation algorithm also shows its efficacy to track the time varying measurement noise covariance.

- It is interesting to note from the previous work [Simandl2009] that for some specific situations where the measurement noise covariance is high (with a comparatively low process noise covariance) or the measurement equation is quasi linear adaptive CDF may present satisfactory estimation results and its performance may be comparable with ADDF and AUKF. In one of case studies similar results are demonstrated. However, ACDF cannot always present acceptable estimation performance for significantly nonlinear systems.
- Note also that although AUKF performs comparably same as ADDF in several estimation problems it performance is attributed to careful choice of tuning of parameters that regulate the spread of sigma points. Such discerning tuning is not required in case of ADDF. Therefore, ADDF can be an apposite alternative of AUKF for nonlinear state estimation.

Chapter 6: Adaptive Gauss Hermite Filter

6.1 Chapter Introduction

In this chapter adaptive Gauss Hermite filter has been formulated from the general framework for adaptive nonlinear filters and characterized using several estimation problems. When the intractable Bayesian integrals present in the general framework are numerically approximated with the help of Gauss Hermite quadrature rule [Golub1969, Ito2000] the algorithm for adaptive Gauss Hermite filter is obtained.

The non-adaptive version of Gauss Hermite filter was first proposed in [Ito2000] where it is demonstrated that on the availability of the complete knowledge of noise covariances performance of non-adaptive version of Gauss Hermite filter is superior compared to its competing algorithms, viz., Divided Difference filter, Unscented Kalman filter and Extended Kalman filter particularly during the state estimation of dynamic systems with significant nonlinearity. Apart from high performance accuracy Gauss Hermite filter has several advantages. Being a point based algorithm Gauss Hermite filter does not require computation of Jacobian and Hessian matrices. Unlike the Unscented Transformation rule choice of Gauss Hermite quadrature points does not depend on tuning parameters.

The above advantages of Gauss Hermite filter motivate the present worker to formulate its adaptive version using the proposed general framework. It is expected that this new algorithm will inherit all the advantages of non-adaptive Gauss Hermite filter along with its essential aspect of adaptation. However, the adaptive Gauss Hermite filter suffers from the curse of dimensionality like its non-adaptive counterpart. The number of quadrature points rises exponentially with the order of the system to be estimated. Therefore, this newly proposed adaptive filter demands sufficient computation effort.

In this chapter the Gauss Hermite quadrature rule is briefly discussed. The quadrature points and weights which are computed using the quadrature rule can be directly plugged into the general framework to formulate the adaptive Gauss Hermite filter. The square root version of adaptive Gauss Hermite filter is also formulated in this chapter from the general framework in square root approach as the Gauss Hermite quadrature rule ensures non negative weights.

The superiority of the proposed algorithm is demonstrated in simulation with the help of non trivial case studies.

6.2 Gauss Hermite Quadrature Rule

6.2.1 Background

In this section we provide the basic concepts of Gauss Hermite quadrature rule with which the Bayesian integrals of the general framework can be numerically approximated. The Gauss Hermite quadrature rule is a special form of Gaussian quadrature rules where weighting function is a unit Gaussian function. In case of Gaussian quadrature rules the weights and sigma points are chosen in such a way that with a suitable polynomial integrand the approximation becomes exact. The Hermite polynomial is chosen in case of the Gaussian weighting function. For evaluation of one dimensional integral with standard Gaussian weighting function, the integral can be expressed as

$$\int_{-\infty}^{\infty} g(x)N(0,1)dx = \int_{-\infty}^{\infty} g(x)e^{-x^2/2}\, dx \qquad (6.1)$$

To evaluate the above integral with Gauss Hermite quadrature rule, the p^{th} order Hermite polynomial is obtained as

$$H_p(z) = (-1)^p e^{z^2} \frac{d^p}{dx^p}\left(e^{-z^2}\right) \qquad (6.2)$$

where $x = z\sqrt{2}$.

Following the definition of $H_p(z)$ the polynomials can be found in a recursive way as given below:

$$H_{p+1}(z) = zH_p(z) - p\,H_{p-1}(z) \qquad (6.3)$$

The quadrature points, ξ_i , for Gauss Hermite quadrature rule are the roots of p^{th} order Hermite polynomial, $H_p(z)$, i.e., $H_p(\xi_i) = 0$. The weights are computed using the formula

$$w_i = \frac{p!}{p^2\left[H_{p-1}(\xi_i)\right]^2} \qquad (6.4)$$

Note that the p^{th} order Hermite polynomial $H_p(z)$ makes the quadrature rule exact for polynomials up to $2p-1$ degree, i.e., the integration is exact for the linear combination of monomials $x_1^{\alpha_1} x_2^{\alpha_2} \cdots x_n^{\alpha_n}$ with total degree up to $2p-1$.

6.2.2 Generation of Quadrature Points and Weights

The selection of quadrature points and the weights as explained above can be done in a simpler alternative way which was first reported in [Golub1969] and followed in [Ito2000] wherein the algorithm of Gauss Hermite filter appears for the first time. A tri-diagonal matrix is formed from a three term recurrence formula as reported in [Golub1969]. The recurrence formula which can be obtained for any orthogonal polynomial is expressed as

$$P_j(x) = (a_j x + b_j) P_{j-1}(x) - c_j P_{j-2}(x) \tag{6.5}$$

for $j = 1,2,\cdots,N$ with $a_j > 0$ and $c_j > 0$. Given that $P_{-1}(x) = 0$ and $P_0(x) = 1$.

The above formula can also be expressed as

$$x\,\boldsymbol{p}(x) = \boldsymbol{Tp}(x) + (1/a_N) p_N(x) \boldsymbol{e}_N \tag{6.6}$$

Where $\boldsymbol{p}(x) = [p_0(x) \quad p_1(x) \quad \cdots \quad p_N(x)]^T$ and $\boldsymbol{e}_N$ is N^{th} unit vector.

Now, $p_N(\lambda) = 0$ when $\lambda\, \boldsymbol{p}(\lambda) = \boldsymbol{T}\, \boldsymbol{p}(\lambda)$ (6.7)

Alternatively, the condition can be written as the roots of the above equation to be equal with the eigen values of tri-diagonal matrix $\boldsymbol{T}$. If the polynomials are not orthonormal, matrix $\boldsymbol{T}$ is not symmetric. The matrix $\boldsymbol{T}$ can be made symmetric by diagonal similarity transformation as

$$\boldsymbol{J} = \boldsymbol{DTD}^{-1} = \begin{bmatrix} \alpha_1 & \beta_1 & 0 & \ldots & 0 \\ \beta_1 & \alpha_2 & \beta_2 & \cdots & 0 \\ 0 & \beta_2 & \alpha_3 & \cdots & 0 \\ 0 & \cdots & \beta_3 & \cdots & 0 \\ 0 & \cdots & \cdots & \beta_{n-1} & \alpha_n \end{bmatrix} \tag{6.8}$$

Where, $\alpha_i = -\dfrac{b_i}{a_i}$ and $\beta_i = \left(\dfrac{c_{i+1}}{a_i a_{i+1}} \right)^{1/2}$ (6.9)

Depending on the choice of quadrature rule the values of a_i, b_i, c_i varies. For Gauss Hermite quadrature rule, $a_i = 2$, $b_i = 0$, $c_i = 2i$. Using these value we get $\alpha_i = 0$ and $\beta_i = \sqrt{i/2}$.

Steps for selection of quadrature points and respective weights are listed below:

- Compute $\boldsymbol{J}$, a symmetric tri-diagonal, defined as $\boldsymbol{J}_{i,i} = 0$ and $\boldsymbol{J}_{i,i+1} = \sqrt{\frac{i}{2}}$ for $1 \le i \le N-1$ for N-quadrature points.
- The quadrature points are chosen as $q_i = \sqrt{2}x_i$ where x_i are the eigen values of $\boldsymbol{J}$ matrix.
- The corresponding weights (w_i) of q_i is computed as $\left|(v_i)_1\right|^2$ where $(v_i)_1$ is the first element of the i^{th} normalized eigenvector of $\boldsymbol{J}$

6.2.3 Extension for higher order systems

The above described quadrature rule is appropriate only for the single dimensional integral. This needs modification to apply for higher order integration space. The single dimensional quadrature rule can be extended for approximating higher order integral of Gaussian filters with the help of product rule. The n^{th} order Gaussian integral $\boldsymbol{I_N} = \int_{R^n} \tilde{\boldsymbol{F}}(\boldsymbol{s}) \frac{1}{(2\pi)^{n/2}} e^{-(1/2)|\boldsymbol{s}|^2} \boldsymbol{ds}$ can be equivalently expressed as

$$\boldsymbol{I_N} \approx \sum_{i_1=1}^{N} \dots \sum_{i_n=1}^{N} \tilde{\boldsymbol{F}}\left(\xi_{i_1,\cdots,i_n}\right) w_{i_1,\cdots,i_n}$$

Or,

$$\boldsymbol{I_N} \approx \sum_{i_1=1}^{N} \dots \sum_{i_n=1}^{N} \tilde{\boldsymbol{F}}\left(q_{i_1}, q_{i_2}, \dots, q_{i_n}\right) w_{i_1} w_{i_2} \dots w_{i_n} \tag{6.10}$$

Where we generate multi-dimensional weights as the products of one-dimensional weights as $w_{i_1,\cdots,i_n} = w_{i_1} w_{i_2} \cdots w_{i_n}$ and generate multi-dimensional unit sigma points as Cartesian product of the one-dimensional unit sigma points as

$$\xi_{i_1,\cdots,i_n} = \begin{bmatrix} \xi_{i_1} \\ \vdots \\ \xi_{i_n} \end{bmatrix} \tag{6.11}$$

In order to evaluate $\boldsymbol{I}_N$ for n^{th} order system with the help of Gauss Hermite quadrature rule, N^n number of quadrature points and weights are required. This indicates that the number of quadrature points rises exponentially with the order of system. A comprehensive description on Gauss Hermite quadrature rule along with necessary illustrations is provided in the in the master's thesis of N. K. Singh [Singh2012].

6.3 Algorithm for Adaptive Gauss Hermite Filter

Algorithm for Adaptive Gauss Hermite filter can be obtained plugging in the method of sigma point selection in the general algorithm for adaptive nonlinear filters in chapter 4. The $\boldsymbol{Q}$ adaptive and the $\boldsymbol{R}$ adaptive GHF can be formulated using this general algorithm. Apart from the method of sigma point generation rest of the algorithm is same as that of the general framework for adaptive nonlinear filter. The algorithm has, therefore, not been reproduced again.

The root version of adaptive GHF can also be obtained using the general framework in Square Root approach presented in chapter 4 because of its additional advantages compared to the standard error covariance form. The usefulness of Square Root approach has already been discussed there. It is to be noted that all the weights in case of Gauss Hermite quadrature rule are non negative. Therefore, the algorithm of Adaptive Square Root Gauss Hermite Filter can be formulated directly using the general algorithm presented without further modifications.

6.4 Characterization of Adaptive GHF

The performance of proposed algorithms of adaptive Gauss Hermite filter has been evaluated with different case studies in this section. A partially $\boldsymbol{Q}$ adaptive GHF is demonstrated using an estimation problem where a time varying parameter has to be estimated along with the states of nonlinear system. $\boldsymbol{R}$ adaptive GHF has also been demonstrated with two case studies and its performance has been compared with the competing algorithms. The $\boldsymbol{R}$ adaptive GHF in square root framework is also validated with a non trivial case study.

6.4.1 Characterization of Q adaptive GHF

6.4.1.1. State and parameter estimation of Van der Pol's oscillator

Q adaptive GHF is validated with the help of an estimation problem where the states and the unknown, time varying friction coefficient of a Van der Pol's oscillator have to be estimated. As the friction coefficient is unknown and also time varying the system model suffers from parametric uncertainty. The parameter cannot be modeled correctly as the dynamics of the parameter variation remains unknown to the designer. In presence of such unknown time varying parameters some elements of the process noise covariance of the system (modeled in terms of parameter augmented states) become unknown. Here it is assumed that only the element of ***Q*** related to friction coefficient is unknown. Therefore, this estimation problem is appropriate to evaluate the applicability of partially ***Q*** adaptive GHF. As the elements of ***Q*** related to the parameter remains unknown only that element needs to be adapted. The known elements of ***Q*** remain frozen to their respective truth values. The concept of partial ***Q*** adaptation is presented in chapter 4. For generation of true state trajectories the true friction coefficient is assumed to vary following the equation

$$\mu_k = 0.5\sin(\tfrac{\pi}{600}k) + 0.5 \tag{6.12}$$

The details of system dynamics, measurement equation and necessary parameters are provided in chapter 3. For adaptation window length is considered to be 30 time instants. Adaptation does not begin till the length of innovation sequence is less than the desired window length. As the element of ***Q*** related to the friction coefficient remains unknown both the adaptive and non-adaptive GHF is initialized with ***Q*** with that particular element assumed arbitrarily as 0.5. A comparative study of the RMS error of the proposed adaptive GHF with that of a non-adaptive GHF has been presented from Monte Carlo study with 1000 runs. The results of Monte Carlo simulation have been presented in Fig. 6.1 – 6.3. It is observed that the RMS error of AGHF for both parameters and states is lower than that of non-adaptive GHF. This indicates that the partially ***Q*** adaptive GHF is capable of tuning ***Q*** to track satisfactorily the unknown time varying parameter. For a representative run the parameter estimation performance of ***Q*** adaptive GHF is provided in Fig. 6.4 which supports the above statement. It is also important to note that although the elements of process noise covariance is known the RMSE of state estimates for non-adaptive GHF is higher than that of AGHF. This is because of the implicit effect of inadequately estimated parameter.

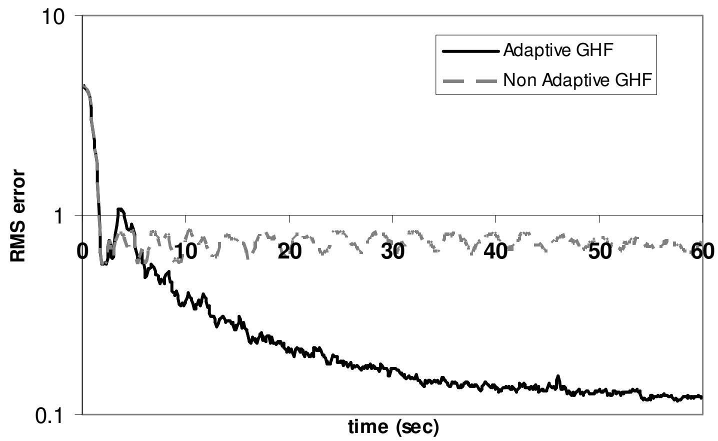


Fig. 6.1: RMS error (friction coefficient estimation) of AGHF & GHF for 1000 MC runs

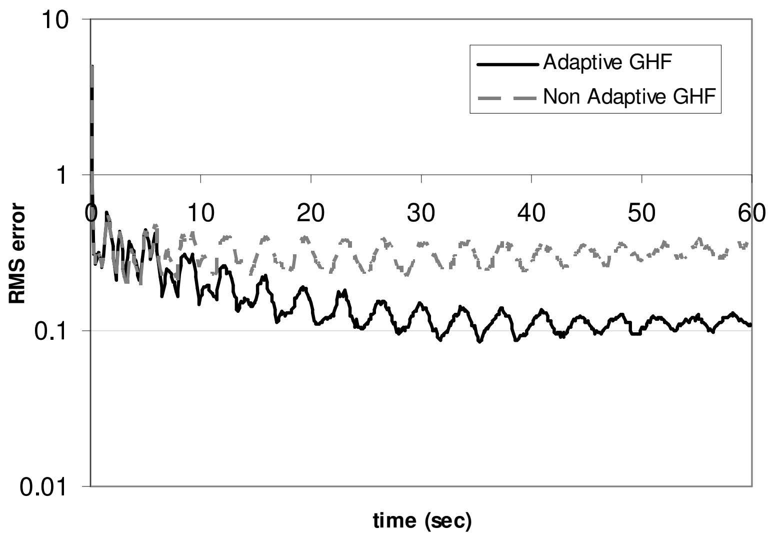


Fig. 6.2: RMS error (state, x_2 estimation) of AGHF & GHF for 1000 MC runs

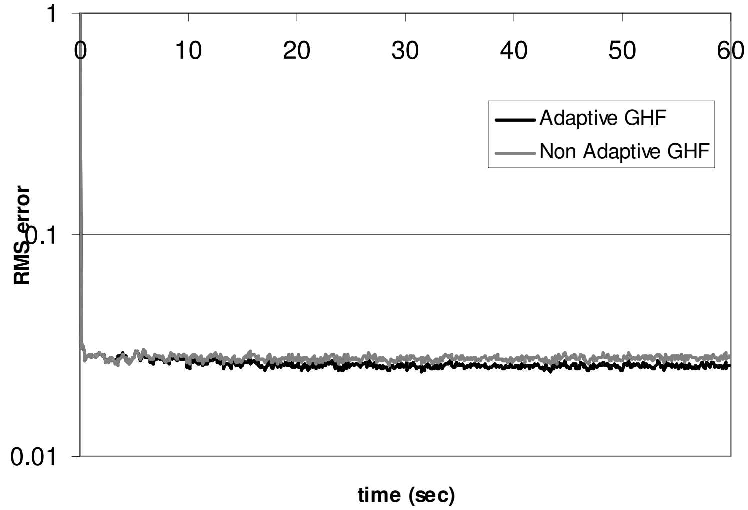


Fig. 6.3: RMS error (state, x_1 estimation) of AGHF & GHF for 1000 MC runs

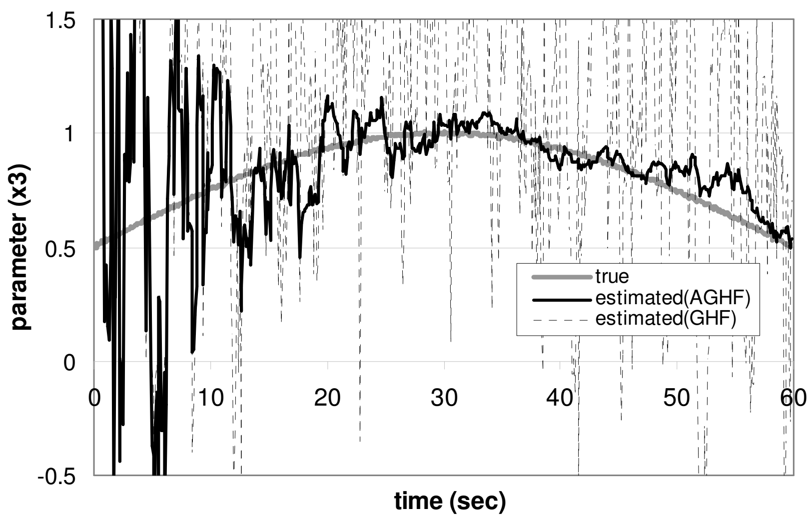


Fig. 6.4: Friction coefficient estimation of AGHF & GHF for a representative run

6.4.2 Characterization of R adaptive GHF

6.4.2.1. State estimation of first order nonlinear system

A single dimensional estimation problem described in chapter 3 and also considered in chapter 5 is revisited again to evaluate the performance of the proposed filter in face of severe nonlinearity in the system dynamics and the measurement equation. In this particular problem tracking of the state of the system is a difficult task and such a problem can readily expose the shortcoming of the estimator involved specifically when the measurement noise covariance is unknown to the estimator. Therefore, the case study may be an appropriate one to evaluate the performance of the proposed filter.

As the measurement noise covariance, ($\boldsymbol{R}_{filter}$) is unknown to the filter we assume a value of measurement noise arbitrarily. The Monte Carlo study is executed initializing different assumed value of $\boldsymbol{R}_{filter}(0)$ to judge the performance of proposed filter. For adaptation the length of the sliding window length is chosen to be equal to 100. Adaptation initiated from the beginning with available window size. When desired length is achieved the sliding window concept becomes appropriate.

It has been discussed in chapter 4 that the recomputation of measurement update step with adapted value of $\boldsymbol{R}$ at a particular time instant improves the estimation accuracy of the adaptive filter. This is also demonstrated with this case study.

The RMS errors of AGHF (with and without re-computation steps), and non-adaptive GHF have been compared with the help of Monte Carlo study with 10,000 runs. Both adaptive and non-adaptive GHF are based on 5 quadrature points. The performance of AGHF is also compared with AUKF with tuning parameters mentioned in [Das2015]. As stated before the unknown measurement noise covariance is initialized with an assumed value as $\boldsymbol{R}_{filter}(0) = 10^{\lambda} * \boldsymbol{R}_{true}$ where λ is chosen as a positive or negative real number depending on the case where a higher value or a lower value of $\boldsymbol{R}_{filter}(0)$ compared to the truth value is considered respectively. In each set of MC simulation the assumed value of $\boldsymbol{R}_{filter}(0)$ remains the same. For each of the candidate estimators same sequence of noises are considered in each Monte Carlo run by appropriate seeding. The observations from the simulation are enumerated below.

Fig. 6.5 indicates that the RMSE of adaptive Gauss Hermite filter is low compared to its non-adaptive version for the choice of $\boldsymbol{R}_{filter}(0) = 10^{-3} * \boldsymbol{R}_{true}$ i.e., where the filter overweighs the measurements. It is also observed from the Fig. 6.5 that the RMSE of AGHF is lower than that of Adaptive UKF and non-adaptive GHF for this assumed value of $\boldsymbol{R}_{filter}(0)$. This indicates that AGHF is performance wise superior compared AUKF when there is significant nonlinearity in the system dynamics. Additionally, it is interesting to note that the RMSE of AGHF with re-computation step is lower compared to AGHF without re-computation.

The similar profiles of RMS errors like Fig. 6.5 are observed in Fig. 6.6 for the choice of $\boldsymbol{R}_{filter}(0) = 10^3 * \boldsymbol{R}_{true}$ where the filter overweighs the measurements. RMSE of AGHF with and without re-computation step is observed to be lower compared to AUKF and non-adaptive GHF for this assumed value of $\boldsymbol{R}_{filter}(0)$.

Monte Carlo simulations are also carried out with other assumed initial choices of $\boldsymbol{R}_{filter}(0)$ such that the filter underweighs or overweighs the measurements. Mathematically it can be represented as, $\boldsymbol{R}_{filter}(0) = 10^{\lambda} * \boldsymbol{R}_{true}$ with $\lambda = 2,1,-1,-2$. Because of the similar trend of the profiles of RMSE for these batches of MC simulations plots are not included. However, their performance has been analysed in Table-6.1.

For a representative run the estimation performance of AGHF, AUKF and non-adaptive GHF is illustrated in Fig. 6.7. It is observed that despite initializing with an assumed value of $\boldsymbol{R}_{filter}(0)$ with large error, the proposed AGHF can satisfactorily track the true trajectory and settle at the true equilibrium point ($x = -1$). However, the non-adaptive GHF fails to keep the track and settles at one of the other equilibrium points ($x = 1$). The AUKF also loses the track and settles at $x = 0$.

Percentage of track loss is computed from MC study for each of the candidate estimators, where track loss is said to occur when estimation error is more than 0.8 at 4 sec. Table -6.1 presents a comparative study of the percentage of track loss for each estimators. A number of batches of Monte Carlo simulations are carried out for various assumed value of $\boldsymbol{R}_{filter}(0)$ as has been discussed before. It is observed from the table that AGHF with the re-computation steps shows the lowest percentage of track loss compared to the other estimators. Here we have also presented the percentage of for the $\boldsymbol{R}$ adaptive GHF based on innovation sequence. The performance of innovation based GHF is found to be comparable with that of residual based AGHF with out re-computation. However, the innovation based algorithm encountered the singularity problem during simulation. Therefore, the simulation is carried out after considering the absolute values of the elements of adapted $\boldsymbol{R}$ matrix.

The plots of the adapted measurement noise covariance obtained from the adaptive GHF are presented in Fig. 6.8 when the truth value of measurement noise covariance is constant. It is observed that despite such assumed initial choice of $\boldsymbol{R}_{filter}(0)$ the adapted measurement noise covariance converges on the truth value (in about 0.1 sec) and subsequently tracks the value. Hence, it may be inferred that the proposed filter can accommodate a wide range of uncertainty while initializing $\boldsymbol{R}_{filter}(0)$.

Another situation is considered when the truth value of the measurement noise covariance is time varying. For this situation also proposed filter can also perform satisfactorily as demonstrated in Fig. 6.9. It has been observed that for such a time varying measurement noise covariance the proposed filter can successfully track the truth value of the measurement noise covariance by online adaptation. To investigate the consequence of different choice of sliding window length for adaptation, the window length (L) is chosen as 25, 50 and 100. For L=25 the adapted $\boldsymbol{R}$ tracks the truth value but tend to overshoot the truth value. For a relatively high window length, L=100, a smoothly varying value of adapted $\boldsymbol{R}$ is obtained. However, for L=100 adapted $\boldsymbol{R}$ cannot track the short term variation satisfactorily. For a choice of window length L=50, tracking of truth value $\boldsymbol{R}$ has been observed to be satisfactory.

Fig. 6.10 illustrates the RMS errors of AGHF (with and without re-computation steps), AUKF and non-adaptive GHF for the case when truth value of $\boldsymbol{R}$ is time varying. The nature of variation of $\boldsymbol{R}$ is already presented in Fig. 6.5. For this case study the window length for the adaptive filters are considered as L=50 which shows a satisfactory tracking in Fig. 6.5. It is also observed that for time varying measurement noise covariance the RMSE of AGHF with re-computation is lower compared to AGHF without re-computation and the other candidate estimators. In this case the percentage of track loss for AGHF with re-computation is 12.39% which is the lowest compared to AGHF without re-computation (percentage of track loss is 16.24%) , AUKF (percentage of track loss is 93.12 %) and non-adaptive GHF (percentage of track loss is 36.01%).

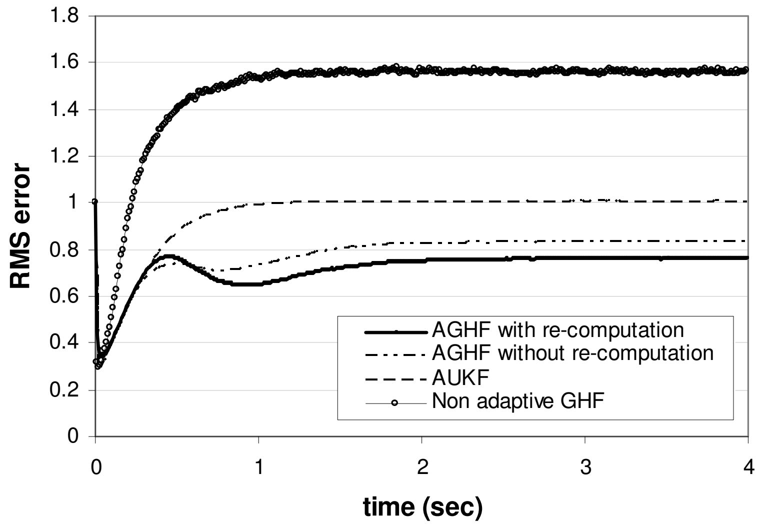


Fig. 6.5: Comparison of RMS error of AGHF (with and without re-computation step), Adaptive UKF and Non-adaptive GHF for 10000 MC run when $\boldsymbol{R}_{filter}(0)=10^{-3}*\boldsymbol{R}_{true}$.

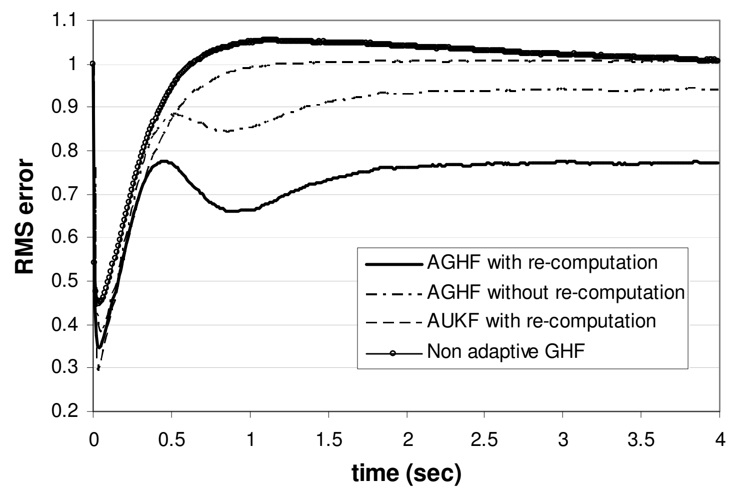


Fig. 6.6: Comparison of RMS error of AGHF (with and without re-computation step), Adaptive UKF and Non-adaptive GHF for 10000 MC run when $\boldsymbol{R}_{filter}(0)=10^{3}*\boldsymbol{R}_{true}$.

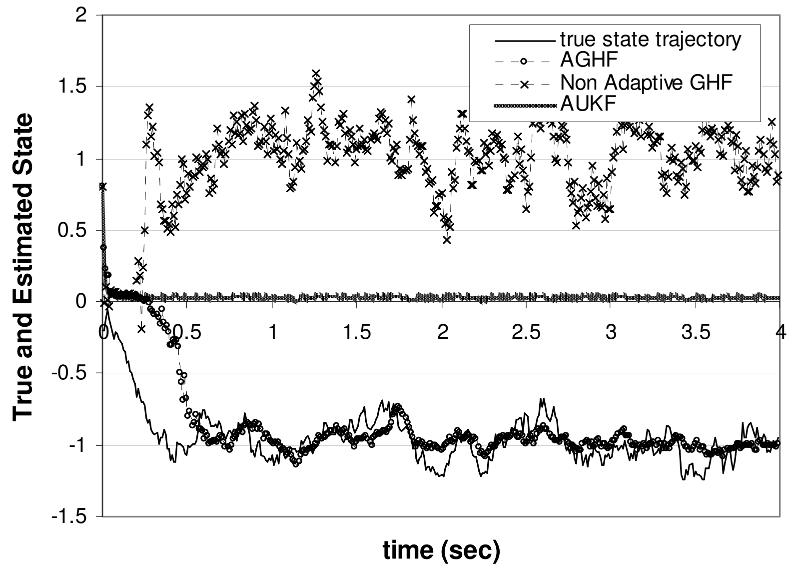


Fig. 6.7: True and estimated states for a representative run

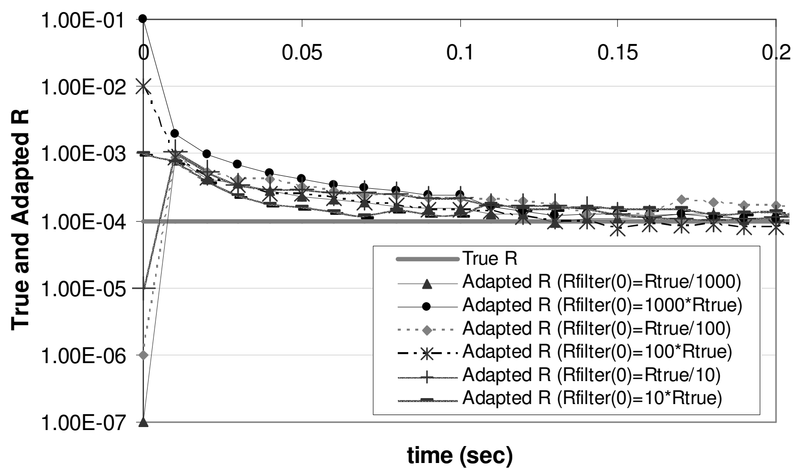


Fig. 6.8: True and adapted measurement noise covariance for a representative run

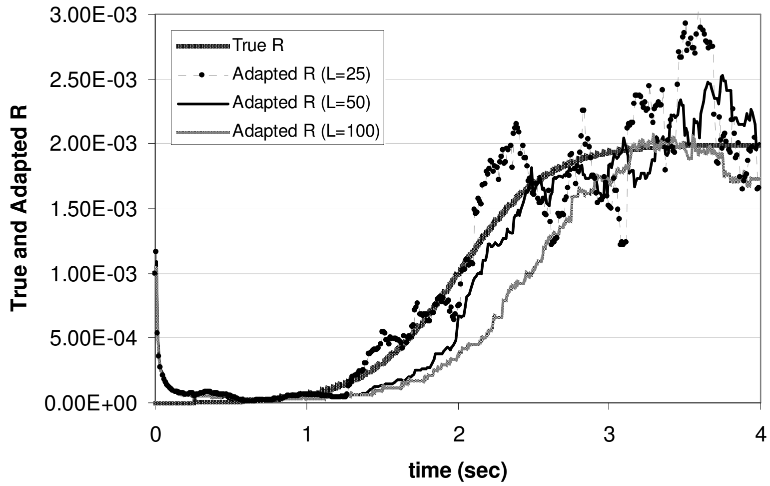


Fig. 6.9: ***R*** Tracking performance for time varying measurement noise covariance (truth value of ***R*** is rising with time)

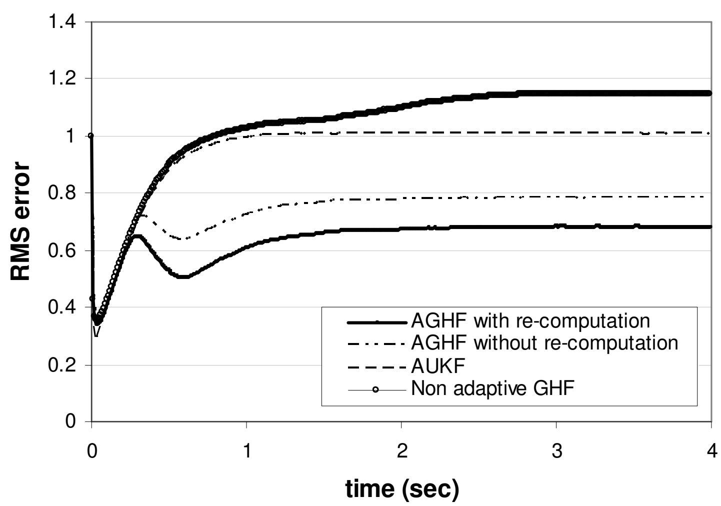


Fig. 6.10: Comparison of RMS error of AGHF (with and without re-computation step), Adaptive UKF and Non-adaptive GHF for 10000 MC run when $\boldsymbol{R}_{true}$.is time varying as shown in Fig. 6.9

Table-6.1: Percentage of track loss cases computed from 10000 Monte Carlo runs

Initial value of R_{filter} $(R_{filter}(0))$	AGHF				Non-adaptive GHF	AUKF
	Re-computation with 2 iterations	Re-computation with 1 iteration	Without re-computation	Innovation based $\boldsymbol{R}$ adaptation		
$10^{-3} * R_{true}$	14.94%	15.11%	17.88%	21.54%	54.71%	93.91%
$10^{-2} * R_{true}$	14.07%	15.15%	17.77%	21.34%	50.50%	93.88%
$10^{-1} * R_{true}$	13.62%	14.80%	17.25%	21.13%	49.65%	93.87%
$10^{1} * R_{true}$	14.87%	15.37%	20.12%	23.51%	35.10%	93.85%
$10^{2} * R_{true}$	14.92%	15.86%	24.62%	24.75%	94.60%	93.87%
$10^{3} * R_{true}$	15.15%	15.42%	22.65%	23.73%	94.32%	93.91%

6.4.2.2. Ballistic Object tracking Problem

The ballistic object tracking problem in single dimension described in chapter 3 and considered in chapter 5 is also addressed here to illustrate the performance of the proposed filter for joint estimation of parameters and states. The ballistic object is considered to be falling vertically and tracked by radar which provides the range of the tracked object. The dynamics of the ballistic object is dependent on aerodynamic drag and gravity during reentry phase. As the object enters atmosphere and experiences drag, the dynamics becomes highly nonlinear. The RMS error of position, velocity and ballistic parameters estimation obtained from the RA-GHF (residual based adaptation), RA-UKF and non-adaptive GHF from 1000 Monte Carlo run are presented below. The unknown measurement noise covariance is assumed to be two decades higher than the truth value to underweight the measurement. Rests of the parameters are specified in chapter 3.

RMS errors of RA-GHF, RA UKF and non-adaptive GHF for the estimates of altitude, velocity of the object and the ballistic parameter have been presented in Fig. 6.11, Fig. 6.12, and Fig. 6.13 respectively. The plots indicate that the RMSE performance of RA GHF is superior to that of non-adaptive GHF for each estimate. The RMS error of AGHF for states as well as the parameter converged quickly (in about 7 sec) to a lower steady state value compared to the non-adaptive GHF.

For this specific case study performance of RA GHF, although not superior to RA UKF, is found to be comparably same to that of competing algorithm of adaptive UKF for all the states (including the ballistic parameter).

It has been observed from simulation that for a single run an average computation time for AUKF is 32.635% of that for AGHF. The simulations are carried out using MATLAB (version 7.9.0.529) in a computer with specifications Intel®, Core (TM) 2 Duo CPU, 2.8 GHz, 2 GB RAM.

For this object tracking problem the noise covariance of the radar measurement is also considered to be time varying to demonstrate the ***R*** tracking performance of AGHF. Fig. 6.14 illustrates the plots of adapted ***R*** for different choice of window length (*L*). The plots indicate that a moderately high value of window length (*L*=90) is appropriate for a smoothed estimate of ***R*** while for tracking of short term changes a small length (*L*=30) is appropriate. For this case the satisfactory performance of AGHF is obtained for *L*=60.

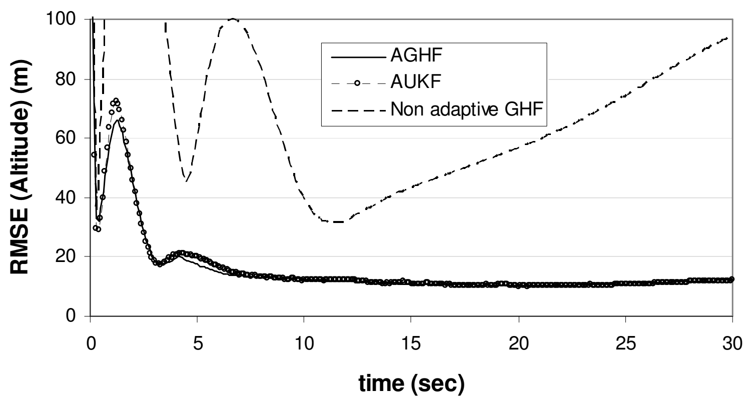


Fig. 6.11: Comparison of RMS error (altitude estimation) of AGHF, AUKF & GHF for 1000 MC runs

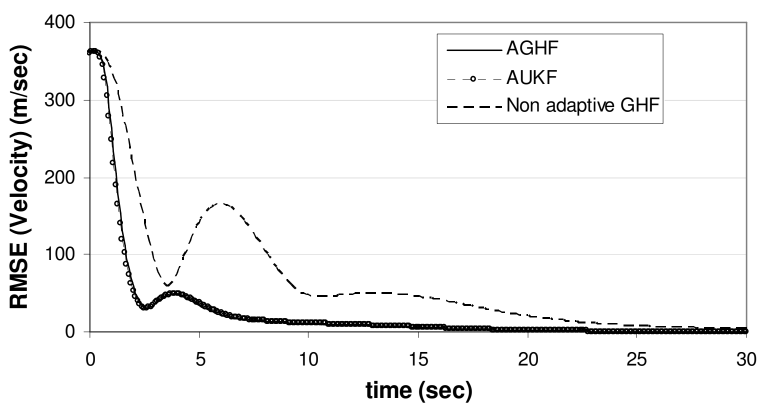


Fig. 6.12: Comparison of RMS error (velocity estimation) of AGHF, AUKF & GHF for 1000 MC runs

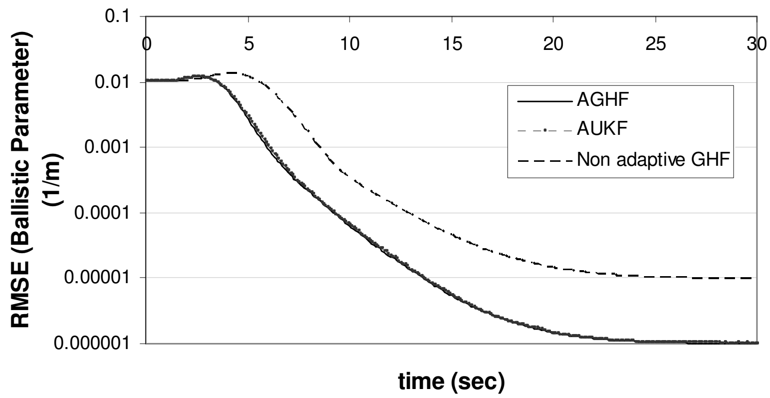


Fig. 6.13: Comparison of RMS error (ballistic parameter estimation) of AGHF, AUKF & GHF for 1000 MC runs

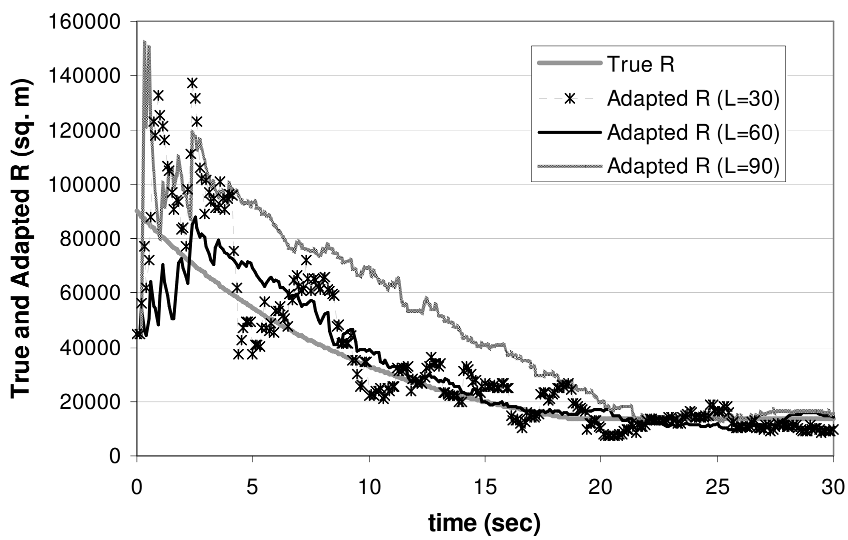


Fig. 6.14: ***R*** Tracking performance for time varying measurement noise covariance (truth value of ***R*** is decaying with time)

6.5 Characterization of *R* Adaptive SR-GHF

The algorithm for adaptive nonlinear filters in standard error covariance form may suffer from loss of positive definiteness of error covariance for some specific estimation problems. For such estimation problems the ***R*** adaptive GHF in square root form has been formulated in this chapter. The algorithm of ***R*** adaptive SR-GHF is validated using an aircraft tracking problem. The aircraft which is executing a maneuvering turn is tracked using bearing only measurements from two tracking radars as described in chapter 3. In the face of unavailability of complete knowledge of ***R*** it is assumed that ***R***(2,2) is unknown. Therefore, ***R***(2,2) is assumed arbitrarily as 5 times more than the truth values of ***R***(2,2). For adaptation the window length is assumed as 25 time instants. Rest of the necessary parameters are provided in chapter 3.

From Monte Carlo simulation with 10000 run RMS error performance of ***R*** adaptive SR-GHF has been compared with its non-adaptive versions in two different situations: (i) when ***R***(2,2) remains unknown for both adaptive and non-adaptive filter, (ii) ***R***(2,2) is known to the non-adaptive filter only. It has been investigated from this case study that the non-adaptive estimators even with the true knowledge of noise covariance cannot always successfully track the maneuvering aircraft because of the non uniqueness of measurement equation as discussed before in chapter 3. Consequently the estimator loses the track of the aircraft. The performance of adaptive and non-adaptive estimators is also compared on the basis of percentage of track loss.

From the RMSE of position, velocity and the turn rate estimation presented by Fig. 6.15, Fig. 6.16, Fig. 6.17 respectively it has been observed that the ***R*** adaptive SR-GHF is superior compared to its non-adaptive version in face of unknown measurement noise covariance. Note that the RMSE are presented excluding the cases where track loss occurs. The RMSE of RA-SR-GHF which is lower compared to non-adaptive SR-GHF excluding track loss case indicates the superiority of RA-SR-GHF over its non-adaptive version irrespective of the cases of track loss. Also the percentage of track loss is significantly low for RA-SR-GHF (percentage of track loss is 1.34%) compared to its non-adaptive version (percentage of track loss is 45.46%).

It is also observed that the adaptive version without the knowledge of ***R***(2,2) can perform nearly close to the non-adaptive version in ideal case with known ***R***(2,2). However percentage of track loss is less for the non-adaptive version in ideal case (percentage of track loss is 1.14%). This also validates the satisfactory performance of RA-SR-GHF.

In Fig. 6.18 and Fig. 6.19 plot of the adapted value of the ***R*** has been presented. It is found from Fig. 6.19 that despite improper initial choice with large error the square of the adapted value of the square root of ***R*** converged to its corresponding truth value and continues to keep it in track. Fig. 6.18 demonstrates that ***R***(1,1) which is known and initialized with the truth value is satisfactorily adapted by RA-SR-GHF to hold on to the truth value. One may opt to keep $\boldsymbol{R}(1,1)$ frozen at its known truth value instead of adapting it.

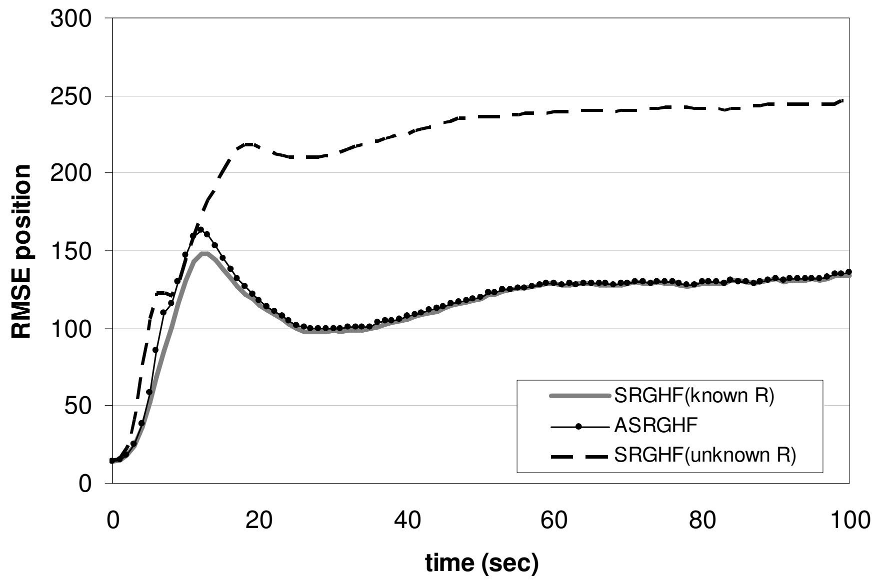


Fig. 6.15: Comparison of RMS error (position estimation) of AGHF & GHF for 10000 MC runs

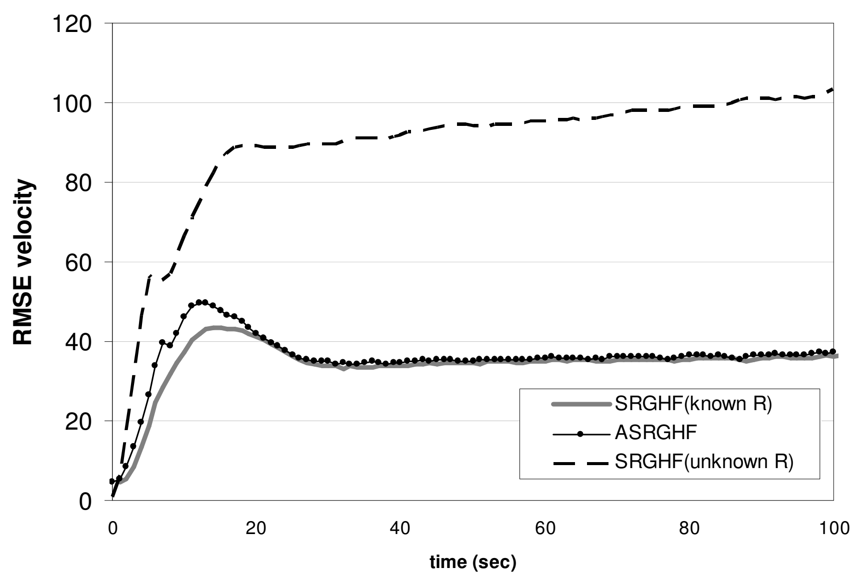


Fig. 6.16: Comparison of RMS error (velocity estimation) of AGHF & GHF for 10000 MC runs

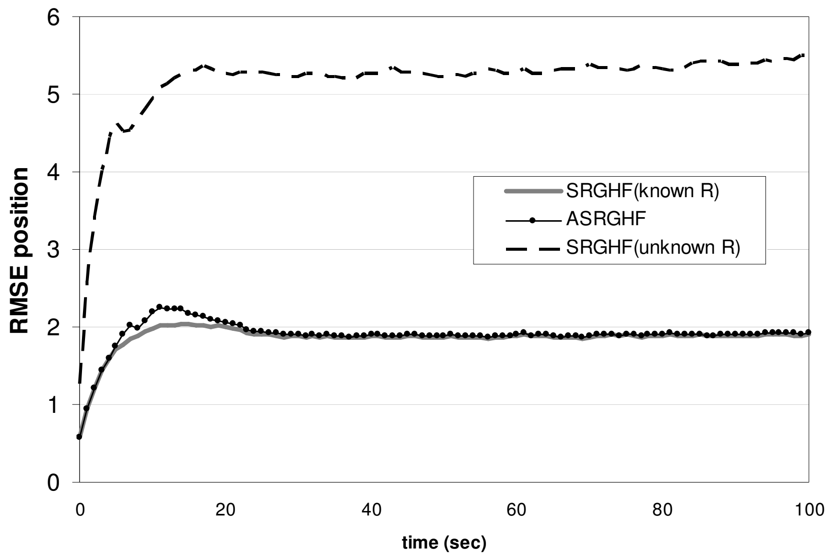


Fig. 6.17: Comparison of RMS error (turn rate estimation) of AGHF & GHF for 10000 MC runs

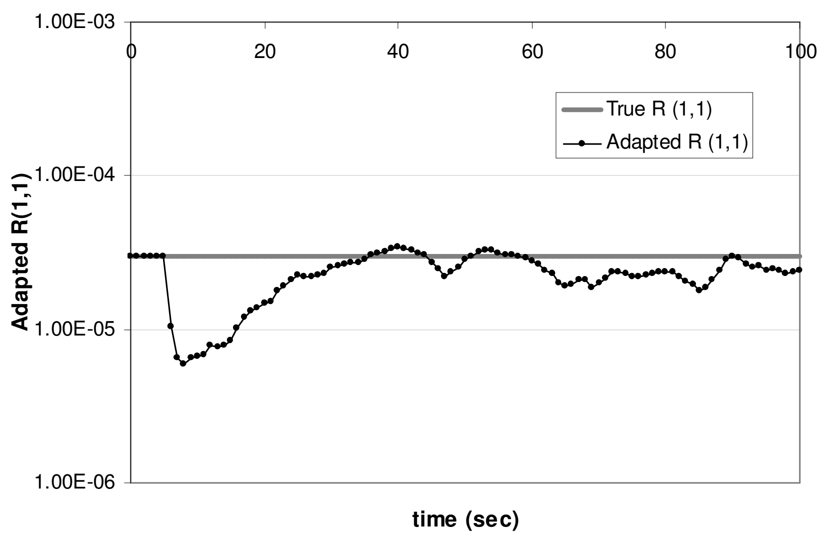


Fig. 6.18: Plot of true and adapted value of ***R*** (1,1)

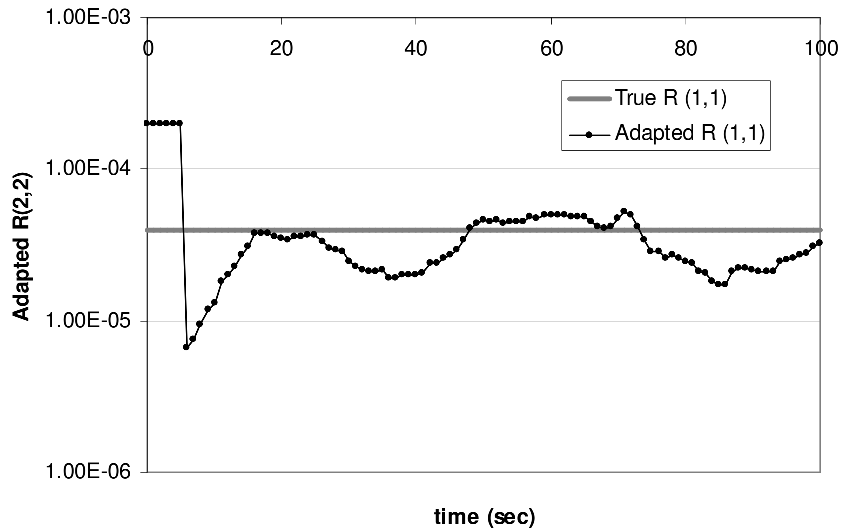


Fig. 6.19: Plot of true and adapted value of ***R*** (2,2)

6.6 Discussions and Conclusions

In this chapter the algorithm of adaptive Gauss Hermite filter is presented and exemplified. It is illustrated that this algorithm can be readily formulated from the general framework for adaptive nonlinear filter using the Gauss Hermite quadrature points and corresponding weights. The estimation performance of both ***Q*** and ***R*** adaptive GHF are validated using different case studies. The significant observations are listed below.

- The superiority of ***Q*** adaptive GHF is demonstrated with the help of the case study which concerns the state and parameter estimation of parameter varying nonlinear system. By satisfactory adaptation (partial ***Q*** adaptation) of the unknown element of ***Q*** the unknown time varying parameter is observed to be estimated satisfactorily. The RMSE from the Monte Carlo studies are also demonstrably superior for ***Q*** adaptive AGHF compared to its non-adaptive version.
- ***R*** adaptive GHF is validated with a number of case studies. From the first order estimation problem with strong nonlinearity it is found that estimation performance of ***R*** adaptive GHF is substantially improved over its non-adaptive version for widely uncertain choice of unknown ***R*** matrix. The superiority of ***R*** adaptive GHF over the non-adaptive counterpart is also demonstrated in the context of joint estimation of parameters

and states with the help of ballistic object tracking problem. Another important finding from these simulation results is that the performance of the proposed adaptive GHF is superior to the adaptive UKF for signal models with strong nonlinearity.

- It has also been observed from the above case studies that the proposed adaptive GHF can successfully adapt the unknown measurement noise covariance and converge on its truth value when it is constant. Furthermore, the adapted $\boldsymbol{R}$ is capable of tracking a time varying truth value of $\boldsymbol{R}$.
- Advantage of considering re-computation for residual based $\boldsymbol{R}$ adaptive GHF is also demonstrated for both constant and time varying $\boldsymbol{R}$. With a few cycle of re-computation estimation performance of the proposed estimator is substantially improved.
- It is also found that performance of innovation as well as residual based $\boldsymbol{R}$ adaptive GHF is comparably same. However, innovation based algorithm suffers from singularity problem and needs an ad hoc method to over come the issue.
- The algorithm of adaptive GHF in square root framework is also formulated in this chapter which is observed to present satisfactorily estimation results and outperforms its non-adaptive version.

However, it is also demonstrated that the algorithm of adaptive GHF provides improved estimation accuracy at the cost of high computation effort specifically for higher order systems. Therefore, adaptive Gauss Hermite filter along with its square root versions are recommended for nonlinear estimation in the face of unknown noise covariance on availability of sufficient computation power.

To overcome the curse of dimensionality adaptive nonlinear filters can also be formulated using spare grid GH quadrature rule [Jia2012] which the present worker has left as his future work. However, in the following chapters other point based adaptive filters are also formulated which are found to be performance wise comparable with AGHF at a lower computation effort.

Chapter 7: Adaptive Cubature Kalman Filter

7.1 Chapter Introduction

In this chapter several versions of adaptive cubature filters have been formulated from the general framework for adaptive nonlinear filter and characterized using non trivial case studies. Bayesian integrals appear in the general algorithm for adaptive nonlinear filter can be numerically approximated with the help of spherical radial cubature rule. Adaptive versions of Cubature Kalman filter, higher degree Cubature Kalman filter, Cubature Quadrature Kalman filter and higher degree Cubature Quadrature Kalman filter are formulated from the general framework when the intractable Bayesian integrals are numerically approximated with help of variants of spherical radial cubature rule proposed in recent publications [Arasaratnam2009, Jia2013a, Bhaumik2013, Singh2015].

The accuracy of the Cubature Kalman filter (CKF) varies depending on the degree of the spherical rule and the order of radial rule. Third degree and fifth degree Cubature rules have been reported in literature by [Arasaratnam2009] and [Jia2013a] respectively with which the Bayesian integrals are numerically approximated. However, possibility of higher degree (higher than 5^{th} degree) Cubature rule is also reported in [Jia2013a].

A variant of Cubature rules are also proposed and renamed as Cubature Quadrature rule in [Bhaumik2013, Singh2015]. This new Cubature rule is different from that of [Jia2013a] in perspective of the approximation of radial integrals. While in higher degree cubature rule [Jia2013a] emphasis is on increasing the degree of accuracy of spherical rule, Cubature Quadrature rule attempts to improve the accuracy by increasing the order of radial rule with the spherical rule of 3^{rd} degree accuracy. Cubature Quadrature rule Kalman filter (CQKF) is presented with 3^{rd} degree spherical rule and higher order radial rule in [Bhaumik2013]. Later the authors of [Bhaumik2013] also extended their work in [Singh2015] with higher degree (5^{th} degree) spherical rule.

However, performance of these advanced cubature point based estimators deteriorates in face of the common problem of unknown noise covariance. Performance of their adaptive versions is needed to be investigated in such contingent situations. In this chapter the spherical radial cubature rule has been briefly discussed. Following the 3^{rd} and 5^{th} degree cubature rule and cubature quadrature rule cubature/quadrature points and their corresponding weights are selected and used in the general framework of adaptive nonlinear filter to develop the corresponding adaptive versions of CKF and CQKF. These new algorithms have been validated with the help of different case studies and the relative advantages of these newly proposed estimation algorithms over the competing algorithms are also investigated.

7.2 Spherical Radial Cubature Rule

7.2.1 Background

Cubature rule reported in [Jia2013a, Arasaratnam2009] can numerically approximate an integral represented by (7.1) using a set of cubature points and the corresponding weights. The d^{th} degree cubature rule accurately approximates a nonlinear function $\boldsymbol{g(x)}$ which is linear combination of monomials with total degree up to d or less.

$$\int_{\boldsymbol{R}^n} \boldsymbol{g}(\boldsymbol{x})\omega(\boldsymbol{x})d\boldsymbol{x} \approx \sum_{i=1}^{m} w_i \boldsymbol{g}(\boldsymbol{x}_i) \tag{7.1}$$

Where $\boldsymbol{x} \in \boldsymbol{R}^n$ and $\omega(\boldsymbol{x})$ is the weighting function. The above expression is accurate for d^{th} degree rule when $\boldsymbol{g}(\boldsymbol{x})$ is defined as $\boldsymbol{g}(\boldsymbol{x}) = a_1 x_1^{\alpha_1} + a_2 x_2^{\alpha_2} + \cdots + a_n x_n^{\alpha_n}$ i.e., linear combination of monomials $x_1^{\alpha_1} x_2^{\alpha_2} \cdots x_n^{\alpha_n}$ with total degree up to d. Here, $\alpha_1, \alpha_2, \ldots, \alpha_n$ are nonnegative integers, $a_1, a_2, \ldots, a_n$ are real values such that $\sum_{i=1}^{n} \alpha_i \le d$. It is to be noted that accuracy of solution is dependent on the degree, d. For cubature rule with higher accuracy, degree of polynomial should be increased.

For Gaussian weighting function, i.e., $\omega(\boldsymbol{x}) = \exp(-\boldsymbol{x}^T \boldsymbol{x})$, the integral is expressed as

$$I(\boldsymbol{g}) = \int_{\boldsymbol{R}^n} \boldsymbol{g}(\boldsymbol{x}) \exp(-\boldsymbol{x}^T \boldsymbol{x}) d\boldsymbol{x} \tag{7.2}$$

To convert it in a spherical radical integration form it is assumed that $\boldsymbol{x} = r\boldsymbol{s}$ with $\boldsymbol{s}^T \boldsymbol{s} = 1$ and $r = \sqrt{\boldsymbol{x}^T \boldsymbol{x}}$ for $r = [0, \infty)$. Then (7.2) will be transformed in the spherical radial coordinate system as

$$I(\boldsymbol{g}) = \int_0^{\infty} \int_{U_n} \boldsymbol{g}(r\boldsymbol{s})\, r^{n-1} \exp(-r^2) d\sigma(\boldsymbol{s}) dr \tag{7.3}$$

Here, $U_n = \{\boldsymbol{s} \in \boldsymbol{R}^n : \boldsymbol{s}^T \boldsymbol{s} = 1\}$ and $\sigma(\cdot)$ is the spherical circle measure or the area element on U_n.

The equation (7.3) consists of two integrals, (i) the radial integral (ii) spherical integral. These two integrals can be numerically approximated and combining these two approximation methods the spherical radial cubature rule can be obtained.

The N_r point radial rule is approximated as

$$\int_0^{\infty} \boldsymbol{g}_r(r) r^{n-1} \exp(-r^2) dr \approx \sum_{i=1}^{N_r} w_{r_i} \boldsymbol{g}_r(r_i) \tag{7.4}$$

The N_s point spherical rule is approximated as

$$\int_{U_n} \boldsymbol{g}_s(\boldsymbol{s}) d\sigma(\boldsymbol{s}) \approx \sum_{i=1}^{N_s} w_{s_i} \boldsymbol{g}_s(\boldsymbol{s}_i) \tag{7.5}$$

Combining the spherical radial approximation rule $I(\boldsymbol{g})$ can be expressed as

$$I(\boldsymbol{g}) \approx \sum_{i=1}^{N_r} \sum_{i=1}^{N_s} w_{r_i} w_{s_i} \boldsymbol{g}(r_i s_i) \tag{7.6}$$

r_i and w_{r_i} are points and weights for approximation of radial integral and s_i and w_{s_i} are points and weights for approximation of spherical integral. The total number of points for computation of $I(\boldsymbol{g})$ is $N_s N_r$ when $r_i \neq 0$ and $N_s(N_r - 1) + 1$ if one of the r_i is zero. The proof has been provided in [Jia2013a].

7.2.2 Spherical Rule

The spherical rule is used to numerically approximate the spherical integral as represented in (7.5). The numerical approximation method for spherical integral with arbitrary degree is presented in the work of [Genz2003] and also followed by [Jia2013a]. In [Jia2013a] following the method of [Genz2003] 3rd degree and 5th degree spherical rules have been derived.

From the 3rd degree spherical rule the spherical integration can be approximated as

$$I_{U_n,3}(\boldsymbol{g}_s) = \frac{A_n}{2n}\sum_{i=1}^{n}\left(\boldsymbol{g}_s(\boldsymbol{e}_i) + \boldsymbol{g}_s(-\boldsymbol{e}_i)\right) \tag{7.7}$$

Where $\boldsymbol{e}_i$ is the i^{th} unit vector and the weight $\omega_i = \frac{A_n}{2n}$ with $A_n = \frac{2\sqrt{\pi^n}}{\Gamma\left(\frac{n}{2}\right)}$

Following the 5th degree spherical rule the spherical integration can be approximated as

$$\begin{aligned} I_{U_n,5}(\boldsymbol{g}_s) = \frac{A_n}{n(n+2)}\sum_{i=1}^{n(n-1)/2}\left(\boldsymbol{g}_s(\boldsymbol{s}_i^+) + \boldsymbol{g}_s(-\boldsymbol{s}_i^+) + \boldsymbol{g}_s(\boldsymbol{s}_i^-) + \boldsymbol{g}_s(-\boldsymbol{s}_i^-)\right) \\ + \frac{(4-n)A_n}{2n(n+2)}\sum_{i=1}^{n}\left(\boldsymbol{g}_s(\boldsymbol{e}_i) + \boldsymbol{g}_s(-\boldsymbol{e}_i)\right) \end{aligned} \tag{7.8}$$

where $\boldsymbol{s}_i^+$ and $\boldsymbol{s}_i^-$ are selected as

$$\left\{\boldsymbol{s}_i^+\right\} \stackrel{\Delta}{=} \left\{\sqrt{\tfrac{1}{2}}(\boldsymbol{e}_k + \boldsymbol{e}_l) : k < l, \quad k,l = 1,2,\cdots,n\right\} \tag{7.9}$$

$$\left\{\boldsymbol{s}_i^-\right\} \stackrel{\Delta}{=} \left\{\sqrt{\tfrac{1}{2}}(\boldsymbol{e}_k - \boldsymbol{e}_l) : k < l, \quad k,l = 1,2,\cdots,n\right\} \tag{7.10}$$

7.2.3 Radial Rule

The radial rule is used to numerically approximate the radial integral as represented in (7.4).The radial integral is transformed with $t = r^2$ so that it can be approximated using Generalized Gauss Laguerre quadrature (GGLQ)rule as given below

$$\int_0^\infty \boldsymbol{g}_r(r) r^{n-1}\exp(-r^2)dr = \int_0^\infty \boldsymbol{g}_{\tilde{r}}(t) t^{\frac{n}{2}-1}\exp(-t)dt \tag{7.11}$$

where $\boldsymbol{g}_{\tilde{r}}(t) = \boldsymbol{g}_r(\sqrt{t})$

The GGLQ rule can now be applied on the right hand side of (7.11). Therefore, (7.11) can be approximated as

$$\int_0^\infty \boldsymbol{g}_{\tilde{r}}(t) t^{\frac{n}{2}-1}\exp(-t)dt = \sum_{i=1}^{N_r} w_{g_i}\tilde{\boldsymbol{g}}_r(r_{g,i}) \tag{7.12}$$

Because of the transformation $t = r^2$ the points and weights are obtained by

$$r_i = \sqrt{r_{g,i}} \text{ with weights } w_{r_i} = w_{g_i}/2 \tag{7.13}$$

However, in [Jia2013a] the author has followed an alternative method of moment matching for computing the points and weights. Using the moment matching method the author of [Jia2013a] has presented the points and weights for 3rd degree and 5th degree radial rule as provided below. It has been reported in [Jia2013a] that the number of points are less for moment matching method compared to the GGLQ approach when *m* is even for the radial rule with (*2m+1*) degree accuracy. On contrary, when *m* is odd, the radial rule gives same points and weight as obtained from GGLQ rule. Therefore, for 3rd degree radial rule both these approach provide same points and weights as

$r_1 = \sqrt{\frac{n}{2}}$ and the corresponding weight $w_{r_1} = \frac{\Gamma(n/2)}{2}$ (7.14)

For 5th degree radial rule the author of [Jia2013a] presents less number of points and weights which are generated using moment matching method as mentioned below.

$r_1 = 0$ and the corresponding weight $w_{r_1} = \frac{\Gamma(n/2)}{n+2}$ (7.15)

$r_2 = \sqrt{\frac{n}{2}+1}$ and the corresponding weight $w_{r_2} = \frac{n\Gamma(n/2)}{2(n+2)}$ (7.16)

7.2.4 Spherical Radial Cubature Rule

Combining the spherical and the radial rule presented above the 3rd degree and 5th degree spherical radial cubature are obtained as given below.

Using 3rd degree rule the integral is approximated using *2n* number of points and weights as

$$\int_{\boldsymbol{R}^n} \boldsymbol{g}(\boldsymbol{x}) N(\boldsymbol{x};\boldsymbol{0},\boldsymbol{I}) = \frac{1}{2n}\sum_{i=1}^{n}\left[\boldsymbol{g}\left(\sqrt{n}\,\boldsymbol{e}_i\right) + \boldsymbol{g}\left(-\sqrt{n}\,\boldsymbol{e}_i\right)\right] \tag{7.17}$$

Using 5th degree rule the integral is approximated using ($2n^2$+1) number of points and weights as

$$\begin{aligned}\int_{\boldsymbol{R}^n} \boldsymbol{g}(\boldsymbol{x}) N(\boldsymbol{x};\boldsymbol{0},\boldsymbol{I}) = {} & \frac{2}{n+2}\boldsymbol{g}(\boldsymbol{0}) + \frac{4-n}{2(n+2)^2}\sum_{i=1}^{n}\left[\boldsymbol{g}\left(\sqrt{n+2}\,\boldsymbol{e}_i\right) + \boldsymbol{g}\left(-\sqrt{n+2}\,\boldsymbol{e}_i\right)\right] \\ & + \frac{1}{(n+2)^2}\sum_{i=1}^{n(n-1)/2}\left[\boldsymbol{g}\left(\sqrt{n+2}\,\boldsymbol{s}_i^+\right) + \boldsymbol{g}\left(-\sqrt{n+2}\,\boldsymbol{s}_i^+\right)\right] \\ & + \frac{1}{(n+2)^2}\sum_{i=1}^{n(n-1)/2}\left[\boldsymbol{g}\left(\sqrt{n+2}\,\boldsymbol{s}_i^-\right) + \boldsymbol{g}\left(-\sqrt{n+2}\,\boldsymbol{s}_i^-\right)\right]\end{aligned} \tag{7.18}$$

The derivations in details have been provided in [Jia2013a].

Notes:

The cubature rules with third degree and fifth degree of accuracy have been considered for numerical approximation of intractable Gaussian integrals. The merits and demerits of these two cubature rules are provided below:

- It is evident from the 3rd degree cubature rule that it is a special case of Unscented Transformation (UT) rule. 3rd degree cubature rule uses $2n$ number of points for n^{th} order system whereas UT rule uses $2n+1$ number of points. The non scaled version of Unscented Transformation [Simon2006] with zero weight for the mean about which the sigma points are selected is exactly matches with 3rd degree cubature rule.
- However, the accuracy of the UT rule can be improved compared to 3rd degree cubature rule with careful choice of tuning parameters with which the spread of sigma points can be controlled. Because of the scaling the weights of UT rule becomes negative for some cases. For example, with the scaling parameter, $\kappa = 3 - n$ suggested in [Julier2000] the weights become negative when applied for higher order the integration space. This may increase the tendency to loss of positive definiteness of error covariance and reduce the stability of the filtering algorithm. The 3rd degree cubature rule having non negative weights is more stable and ensures the positive definiteness of error covariance.
- It has been reported in [Arasaratnam2009, Jia2013a] that the accuracy of Cubature rule can be improved with rise in degree of the cubature rule. The number of cubature points increases polynomially with increase in degree. For the 5th order cubature rule the required number of cubature points is found to be $2n^2 + 1$.
- However, like UT rule, there exists stability issue regarding the 5th order cubature rule. With increase in the dimension of the integration space the weights of 5th degree cubature become negative unlike 3rd degree cubature rule. For 5th degree cubature rule sigma points which are same as that for 3rd degree cubature rule have weights selected as, $\frac{4-n}{2(n+2)^2}$. Therefore for the dimension of the integration space greater than 4 this weight becomes negative. However, for $n \to \infty$, the negative weight of UT rules tends to $-\infty$ whereas the negative weight of 5th degree cubature rule tends to zero. Consequently, 5th degree cubature rule is relatively more stable than UT rule as reported in [Jia2013a].
- The n^{th} degree Cubature rule is exact for monomials up to n^{th} order whereas n^{th} order GHQ rule has $(2n-1)^{th}$ order accuracy [Sarkka2013]. It is, therefore, well understood that

the accuracy of 3rd degree cubature rule is lower than that of 3rd degree Gauss Hermite quadrature rule. On contrary the 5th degree cubature rule has same accuracy of 3rd order GHQ rule.

7.3 Cubature Quadrature Rule

A variant of cubature rule known as cubature quadarture rule has been formulated and reported in [Bhaumik2013], contemporary work of [Jia2013a] which proposes higher order cubature rule. In the work of [Bhaumik2013] the author has proposed another version of cubature filter which has increased accuracy compared to 3rd degree cubature rule and termed as cubature quadrature rule. For the proposed quadrature rule the spherical integral is evaluated with 3rd degree spherical rule only. The radial integral is approximated using Generalized Gauss Laguerre quadrature (GGLQ) rule with accuracy of n'^{th} order by solving n'^{th} order Chebyshev-Laguerre equation. The approximation method is stated below:

$$\int_{\boldsymbol{R}^n} \boldsymbol{g}(\boldsymbol{x}) N(\boldsymbol{x};\boldsymbol{0},\boldsymbol{I}) = \sum_{i=1}^{2nn'} \boldsymbol{g}(\xi_i) w_i \tag{7.19}$$

Where $\boldsymbol{\xi_i} = \sqrt{2\lambda_j}\, \boldsymbol{e_k}$ (7.20)

λ_j is the solution of n'^{th} order Chebyshev-Laguerre polynomial with $\alpha = n/2 - 1$:

$$L_{n'}^{\alpha} = \lambda^{n'} - \tfrac{n'}{1!}(n'+\alpha)\lambda^{n'-1} + \tfrac{n'(n'-1)}{2!}(n'+\alpha)(n'+\alpha-1)\lambda^{n'-2} - \cdots = 0 \tag{7.21}$$

The weights corresponding to the above sigma points are obtained as

$$w_i = \frac{1}{2n\Gamma(n/2)} \frac{n'!\Gamma(\alpha+n'+1)}{\lambda_j \left[\dot{L}_{n'}^{\alpha}(\lambda_j)\right]^2} \tag{7.22}$$

Here, $i = 1,2,\ldots,2nn'$, $j = 1,2,\ldots,n'$ and $k = 1,2,\ldots,2n$

The quadrature points and weights are obtained by combining the 3rd degree spherical rule and n'^{th} order radial rule where n' radial points are selected and the solution is accurate for $(2n'-1)$ degree. The work of [Bhaumik2013] differs from [Jia2013a] on the perspective of the numerical approximation method of radial integral. The authors of [Bhaumik2013] follows GGLQ rule where as [Jia2013a] follows moment matching method.

The authors of [Bhaumik2013] has also extended their work in [Singh2015] wherein the work of [Jia2013a] has been critically analyzed and the moment matching method followed in [Jia2013a] for computing the radial points and weights is reported to be analytically ambiguous. The authors proposed to combine higher degree spherical rule [Genz2003, Jia2013a] with higher order radial rule by solving n'^{th} order Chebyshev-Laguerre equation as mentioned [Bhaumik2013]. This new cubature rule has been named as higher order cubature quadrature rule. The steps for generation of higher order cubature quadrature points and corresponding weights are enumerated in [Singh2015]. Below we present only the steps for computing cubature quadrature points based on 5th degree spherical rule which has been presented in (7.23) and (7.24)

$$\begin{aligned}\int_{R^n} \boldsymbol{g}(\boldsymbol{x})N(\boldsymbol{x};\boldsymbol{0},\boldsymbol{I}) = {} & \tfrac{4-n}{2n(n+2)} A_n \sum_{j=1}^{n'} \sum_{i=1}^{n} \left[\boldsymbol{g}\left(\boldsymbol{e}_i\sqrt{2\lambda_j}\right)+\boldsymbol{g}\left(-\boldsymbol{e}_i\sqrt{2\lambda_j}\right)\right]\varpi_j \\ & + \tfrac{1}{n(n+2)} A_n \sum_{j=1}^{n'} \sum_{i=1}^{n(n-1)/2} \left[\boldsymbol{g}\left(\boldsymbol{s}_i^+\sqrt{2\lambda_j}\right)+\boldsymbol{g}\left(-\boldsymbol{s}_i^+\sqrt{2\lambda_j}\right)\right]\varpi_j \\ & + \tfrac{1}{n(n+2)} A_n \sum_{j=1}^{n'} \sum_{i=1}^{n(n-1)/2} \left[\boldsymbol{g}\left(\boldsymbol{s}_i^-\sqrt{2\lambda_j}\right)+\boldsymbol{g}\left(-\boldsymbol{s}_i^-\sqrt{2\lambda_j}\right)\right]\varpi_j\end{aligned} \tag{7.23}$$

Where $A_n = 2\sqrt{\pi^n}\big/\Gamma\left(\tfrac{n}{2}\right)$ and λ_j is the solution of n'^{th} order Chebyshev-Laguerre polynomial with $\alpha = n/2 - 1$ as given below:

$$L_{n'}^{\alpha} = \lambda^{n'} - \tfrac{n'}{1!}(n'+\alpha)\lambda^{n'-1} + \tfrac{n'(n'-1)}{2!}(n'+\alpha)(n'+\alpha-1)\lambda^{n'-2} - \cdots = 0 \tag{7.24}$$

The corresponding weights are obtained as

$$\varpi_j = \frac{1}{2\sqrt{\pi^n}} \frac{n'!\Gamma(\alpha+n'+1)}{\lambda_j\left[\dot{L}_{n'}^{\alpha}(\lambda_j)\right]^2} \tag{7.25}$$

Notes:

- It has been observed by the present worker while implementing the non-adaptive version of higher degree Cubature Quadrature Kalman filter that the 3rd degree CQKF is more stable than 5th degree CQKF as negative weights do not appear for the former. 5th degree CQKF having negative weights like 5th degree cubature filter [Jia2013a] may not always guarantee the positive definiteness of error covariance.
- It is also to be noted from the simulation results in [Singh2015] that the performance of higher degree CQKF (5th degree) is improved over the higher degree CKF (5th degree) and CQKF(3rd degree) for higher order systems. However, during

performance comparison with CQKF [Bhaumik2013] it is observed that the performance is not substantially improved for higher degree CQKF (5^{th} degree) with its additional computation effort.

7.4 Algorithm of ACKF and ACQKF

For numerical approximation of integrals with unit Gaussian weighting function the methods using spherical radial cubature rule has been presented in previous sections. Using these approximation methods the integrals can be expressed as weighted sum of sigma points. After computation of sigma points and corresponding weights they can be directly plugged into the General framework of adaptive nonlinear filters to formulate adaptive cubature filters and adaptive Cubature Quadrature Kalman filters. Using 3^{rd} and 5^{th} degree cubature rule and cubature quadrature rule and depending on the nature of adaptation (i.e., $\boldsymbol{Q}$ or, $\boldsymbol{R}$ adaptation) different algorithms can be formulated with the help of general framework given in chapter 4. The algorithmic steps are not repeated in this chapter as those are the same as that of general framework.

7.5 Characterization of proposed estimators

The adaptive Cubature filters are validated using different case studies. At first the algorithm of $\boldsymbol{Q}$ Adaptive CKF (3^{rd} degree) in square root framework is validated using ballistic object tracking problem. The other case studies are considered to illustrate the superiority of $\boldsymbol{R}$ Adaptive versions of CKF (5^{th} degree) and CQKF (both 3^{rd} and 5^{th} degree). The $\boldsymbol{R}$ adaptive versions CQKF (3^{rd} degree) and CKF (3^{rd} degree) in square root framework are also validated using a case study and their performance is compared with $\boldsymbol{R}$ adaptive square root GHF(3^{rd} degree).

7.5.1 Demonstration of Q adaptive version of square root CKF

7.5.1.1. Ballistic object tracking problem

The ballistic object tracking problem as described in chapter 3 is considered for validation of the algorithm of $\boldsymbol{Q}$ Adaptive version of Square root CKF with 3^{rd} degree accuracy (QA-SR-CKF 3^{rd} degree). This case study has been used before for validation of ADDF and AGHF. We consider the same case study again for validation of QA-SR-CKF (3^{rd} degree). The algorithm can be obtained using the general framework with square root approach presented in chapter 4 as the weights for 3^{rd} degree cubature rule are non negative. When the process noise covariance is unknown and the measurement noise covariance is known performance of QA-SR-CKF (3^{rd} degree) is compared with non-adaptive SR-CKF (3^{rd} degree). $\boldsymbol{Q}$ being unknown has been assigned arbitrarily with 5 decades higher than the truth value. The window length is considered to be 10 time instants. RMSE of states and the ballistic parameter obtained from 1000 Monte Carlo run are presented from Fig. 7.1 – 7.3. The plots indicates that the performance of QA-SR-CKF (3^{rd} degree) in face of unknown $\boldsymbol{Q}$, is superior to its non-adaptive version as expected. The RMSE of QA-SR-CKF is low and its convergence is better. However, for the non-adaptive SR-CKF with unknown $\boldsymbol{Q}$ RMSE for states are high. In Fig. 7.3 it is observed that RMSE of parameter for non-adaptive SR-CKF tends to diverge.

The performance of QA-SR-CKF (without the knowledge of $\boldsymbol{Q}$) is also compared with its non-adaptive version when $\boldsymbol{Q}$ is known (an ideal situation). It is observed also from the plots of RMSE that the RMSE for non-adaptive version is lower than that of the adaptive version in the ideal case. However, RMSE of QA-SR-CKF shows a tendency to come close to that of the ideal case during the steady state.

The adapted $\boldsymbol{Q}$ gradually settles on the actual value. But it takes some time takes time to converge. This is the reason behind the initial mismatch of RMSE of QA-SR-CKF (unknown $\boldsymbol{Q}$) and non-adaptive SR-CKF (known $\boldsymbol{Q}$ in ideal situation). In Fig. 7.4 we provide the plot of an element of adapted $\boldsymbol{Q}$ for a representative run.

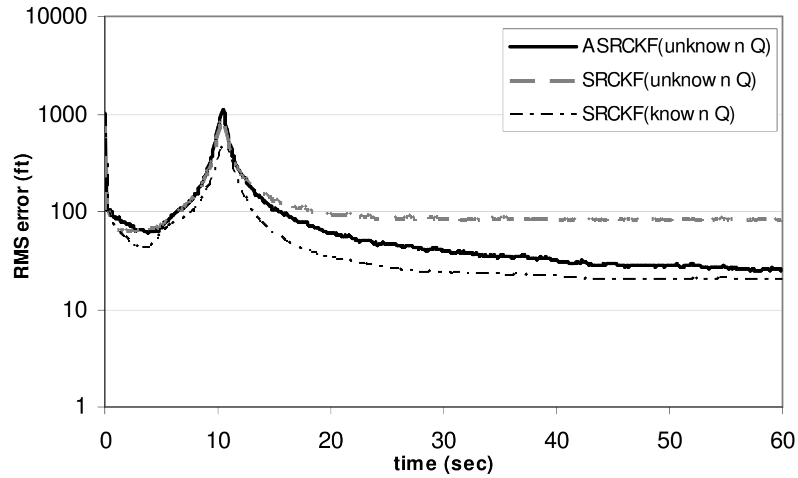


Fig. 7.1: Comparison of RMS error (altitude estimation) of ASRCKF & SRCKF for 1000 MC runs

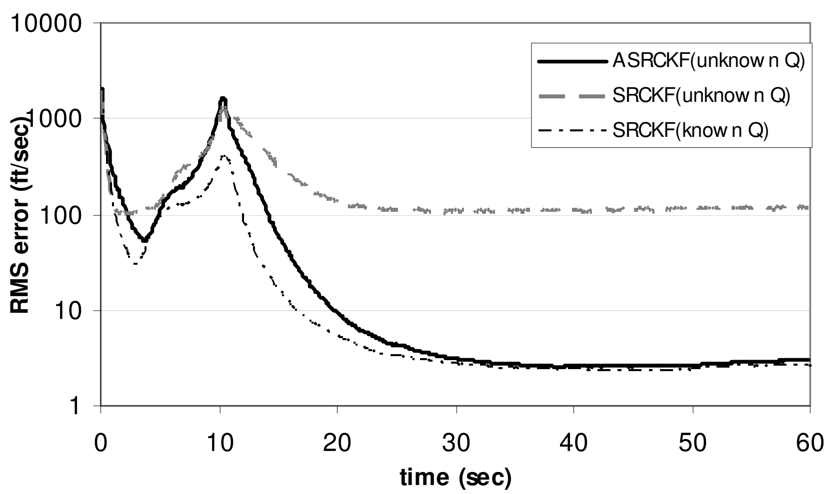


Fig. 7.2: Comparison of RMS error (velocity estimation) of ASRCKF & SRCKF for 1000 MC runs

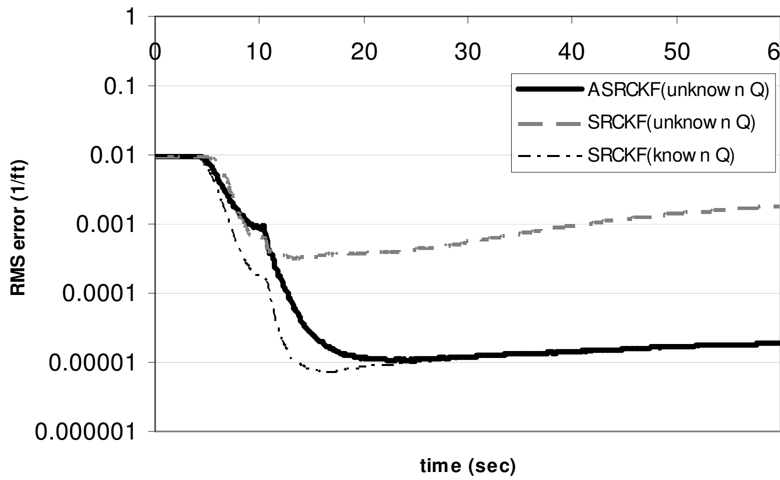


Fig. 7.3: Comparison of percentage of RMS error (ballistic parameter estimation) of ASRCKF & SRCKF for 1000 MC runs

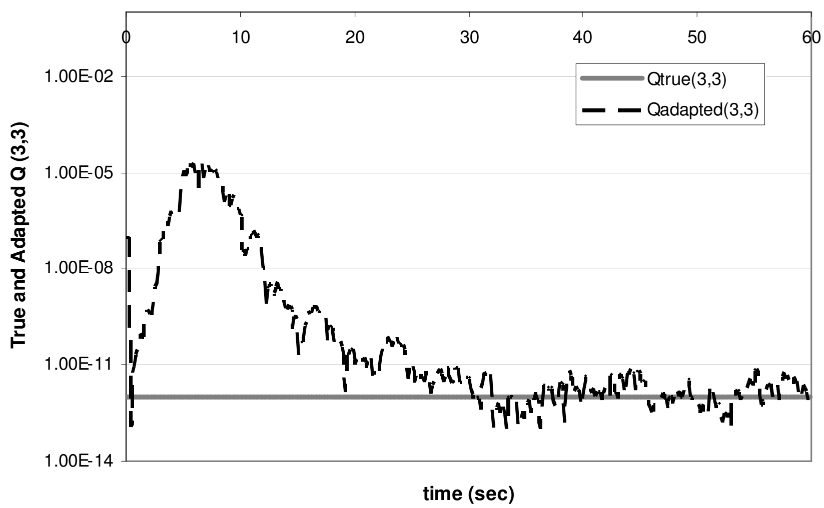


Fig. 7.4: Plot of estimated process noise covariance ($\boldsymbol{Q}_{3,3}$) for a representative run

7.5.2 Demonstration of R adaptive estimators

7.5.2.1. First order nonlinear problem

The first order nonlinear estimation problem which has been considered in chapter 5 and 6 is employed once again for a comparative study of ***R*** adaptive versions of CKF, CQKF and the previously proposed ***R*** adaptive filtering algorithms. It is assumed that the process noise covariance is known to the filter where as the knowledge of measurement noise covariance is unavailable. Because of the unavailability of the measurement noise covariance it is assumed arbitrarily with a value three decades lower than the truth value. Window size is chosen as 100 time instants.

It is observed earlier that due to strong bi modal tendency of the measurement equation the state estimate may settle on a wrong equilibrium point. Even non-adaptive filters with complete knowledge of noise covariances may lose the track. The susceptibility of track loss increases when noise covariances are initialized with inaccurate knowledge of noise covariance. The performance of adaptive CKF and CQKF are analyzed with other adaptive nonlinear filters with respect to the percentage of track loss. The RMS error from Monte Carlo study with 10000 run is also furnished in addition to this. Note that in each Monte Carlo run same noise sequences have been considered for all the candidate estimators.

It has been observed from the simulation results (see Table 7.1) that percentage of track loss in case of ***R*** adaptive CKF (5^{th} degree) is less compared to ***R*** adaptive CKF (3^{rd} degree), ***R*** adaptive UKF. It is also important to note that almost same percentage of track loss is observed for RA-CKF (5^{th} degree) and RA-GHF (3^{rd} order). However, track loss percentage for RA-GHF (5^{th} order) is lower compared to RA-CKF (5^{th} degree).

Performance of ***R*** adaptive CQKF based on 3^{rd} degree cubature rule and 2^{nd} order radial rule, denoted as RA-CQKF (3^{rd} degree, 2^{nd} order) and adaptive higher degree CQKF based on 5^{th} degree cubature rule and 3^{rd} order radial rule, denoted as RA-CQKF (5^{th} degree, 3^{rd} order) is also assessed using this case study. It is demonstrated that the RA-CQKF (3^{rd} degree, 2^{nd} order) performs equally well as compared with computationally intensive RA-GHF (5^{th} order). The percentage of track loss for RA-CQKF (3^{rd} degree, 2^{nd} order) is significantly less as compared to RA-CKF (5^{th} degree) and the computational cost of RA-CQKF(3^{rd} degree, 2^{nd} order) is also less than that of RA-CKF (5^{th} degree) and RA-GHF (5^{th} order) as the former uses less number of quadrature points compared to the other two estimators.

Performance of the RA-CQKF (5^{th} degree, 3^{rd} order) is also found comparable with RA+CQKF (3^{rd} degree, 2^{nd} order) and RA-GHF (5^{th} order). But percentage of track loss of this estimator is slightly higher than other tow. Moreover, as applied for a first order system the number of points for this estimator is more compared to all other estimator.

For each of the candidate estimators in addition to the percentage of track loss, number of points required and the computation time are also presented in Table 7.1 for comparison on the basis of computation time.

The effect track loss is reflected in the RMS error performance presented in Fig. 7.5. RMSE of RA-CKF (5^{th} degree) retrace that of RA-GHF (3^{rd} order) as it is expected from the percentage of track loss. RMSE of RA-CKF (5^{th} degree) is lower than that of RA-CKF (3^{rd} degree) but higher than RA-GHF (5^{th} order). RMSE of RA-CQKF (3^{rd} degree, 2^{nd} order) and RA-CQKF (5^{th} degree, 3^{rd} order) almost retrace the RMSE of RA-GHF(5^{th} order).

To demonstrate how the performance of non-adaptive filter degrades in the face of unknown $\boldsymbol{R}$, RMSE of non-adaptive CQKF (3^{rd} degree, 2^{nd} order) is also presented along with the plots of RMS error of $\boldsymbol{R}$ adaptive filters. The RMSE of non-adaptive CQKF (3^{rd} degree, 2^{nd} order) despite its higher accuracy cannot resist numerous occurrences of track loss (55.36%) with improper choice of $\boldsymbol{R}$ matrix. Consequently, the RMSE for this case is the higher compared to all the adaptive nonlinear estimators.

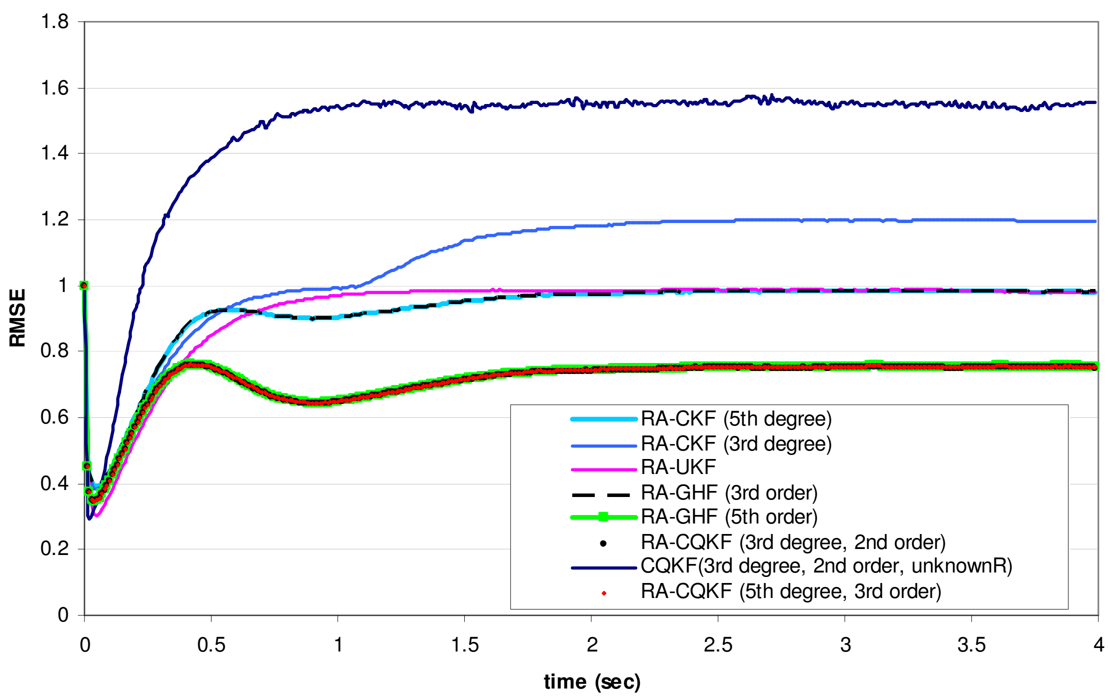


Fig. 7.5: Comparison of RMS error of proposed filters with the existing filters for 10000 MC run

Table-7.1: Percentage of track loss and computation time presented for adaptive estimators

Adaptive estimators	Percentage of track loss	Computation time #	No. of points	Expression*
ACKF(5th degree)	24.78%	83.4%	3	$1+2n^2$
ACKF(3rd degree)	36.48%	48.9%	2	$2n$
AUKF	91.85%	82.5%	3	$2n+1$
AGHF(3rd order)	24.79%	84.8%	3	3^n
AGHF(5th order)	14.64%	100%	5	5^n
ACQKF(3rd degree, 2nd order)	14.65%	94.2%	4	$2nn_r$
AHCQKF (5th degree, 3rd order)	14.72%	113.4%	6	$2n^2n_r$

* Expression of number of points required for n^{th} order system and n_r^{th} degree radial rule
\# The computation time for each estimator is expressed as a percentage of that for AGHF (5th order). The simulation are carried out using MATLAB (version 7.9.0.529) in a computer with specifications Intel®, Core (TM) 2 Duo CPU, 2.8 GHz, 2 GB RAM.

7.5.2.2. State estimation of Lorenz attractors

The Lorenz attractors are a special class of chaotic system with nonlinear signal models which are considered in [Ito2000, Bhaumik2013] for evaluation of nonlinear estimators. System dynamics of a third order Lorenz attractor and the respective observation equation have been provided in chapter 3.

In the face of unknown measurement noise covariance RA-CKF (5th degree), RA-GHF (3rd order), RA-CQKF (5th degree, 3rd order) and RA-UKF have been employed to estimate the states of Lorenz system. The estimation accuracy of the estimators is illustrated with the help of RMSE plot for each state. Following [Bhaumik2013] Monte Carlo simulation with 100 runs has been carried out with same set of noise sequences for each estimator.

Form Fig. 7.6 and Fig. 7.7 it is observed that the RMSE plot of RA-CKF (5th degree) almost retrace that of RA-GHF (3rd order) for state 1 and state 2. It is also important to note here that the RMSE for these two estimators is considerably low compared to than that for RA-UKF.

For this case study performance of RA-CQKF (5th degree, 3rd order) is also compared with RA-CKF (5th degree). For this specific case study it is observed that its performance does not seem to be promising as per the expectation from its performance of the previous case study.

RMSE of this estimator for both the states is higher than RA-CKF (5^{th} degree). Nevertheless, its performance is comparable with that of RA-UKF.

Superiority of none of the adaptive estimators can be inferred from the RMSE for state 3. All the RMSE profiles are comparable as observed in Fig. 7.8.

During the performance comparison the non-adaptive CKF (5^{th} degree) is also considered which shows that performance of non-adaptive CKF (5th degree) degrades drastically with inaccurately chosen ***R***. The assumed value of ***R*** is 3 decades lower than the truth value for this case study. Window size for adaptation is taken as 100 time instants.

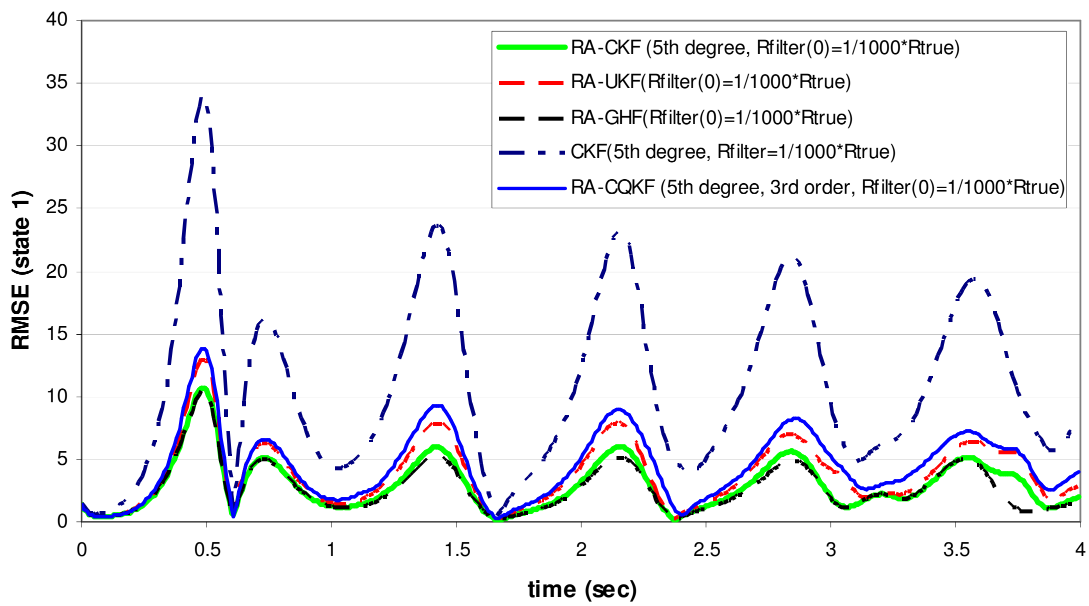


Fig. 7.6: Plots of RMSE of first state of Lorentz attractor for different adaptive estimators

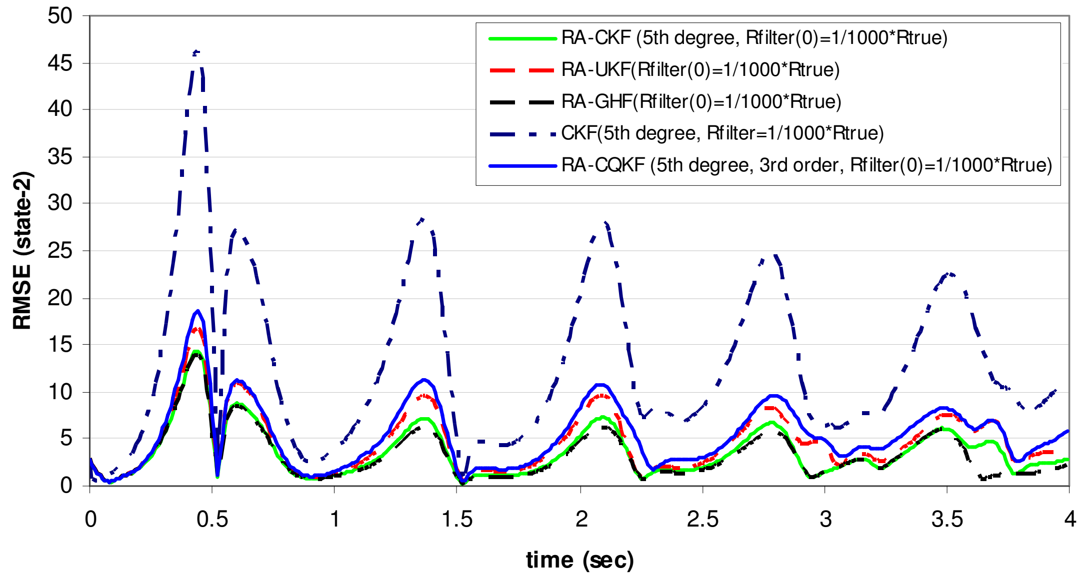


Fig. 7.7: Plots of RMSE of second state of Lorentz attractor for different adaptive estimators

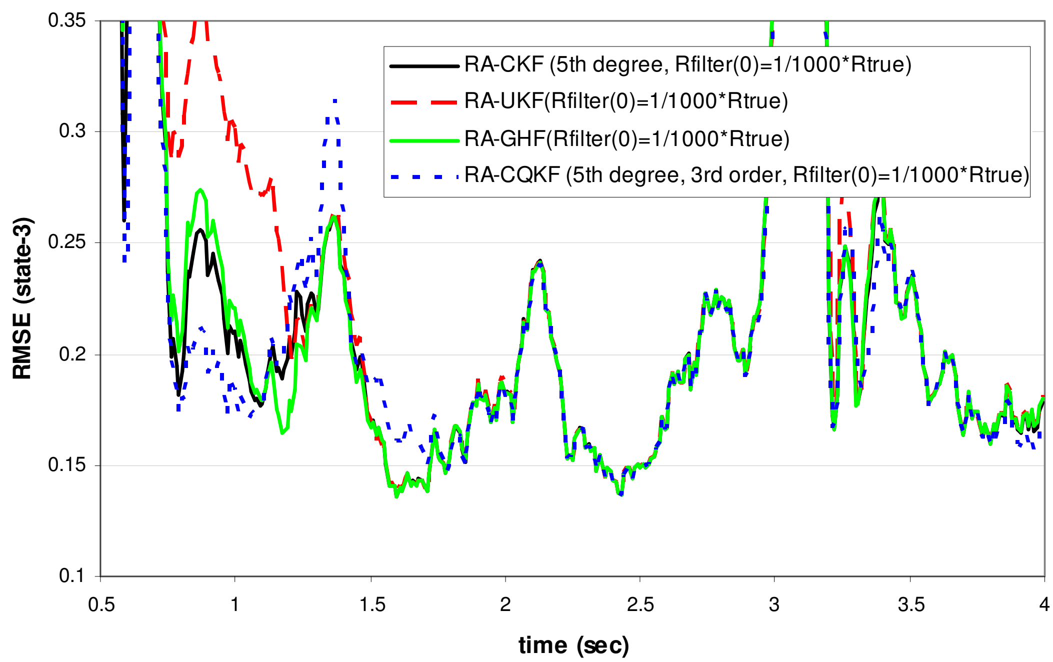


Fig. 7.8: Plots of RMSE of third state of Lorentz attractor for different adaptive estimators

7.5.2.3. Fourth order nonlinear estimation problem

This case study has been considered to demonstrate the superiority of ***R*** adaptive higher order CQKF over its competing algorithms. This numerical problem has been used by [Singh2015] for the validation of non-adaptive version of higher order CQKF. We have considered it to demonstrate the superiority of its adaptive version. Here performance of RA-CQKF (5^{th} degree, 3^{rd} order) has been compared with RA-CQKF (3^{rd} degree, 3^{rd} order) and RA-CKF (5^{th} degree). ***R*** is chosen with large error compared to the truth value (two decades lower than the truth value) for this case study. Window size for adaptation is taken as 100 time instants. Plots for RMS error of four states from 1000 Monte Carlo run are presented from Fig. 7.9 – 7.12. It is observed that for all four states RMS error for RA-CQKF (5^{th} order, 3^{rd} degree) is marginally lower compared to that for the other estimators. However, computation effort for RA-CQKF (5^{th} order, 3^{rd} degree) is also higher compared to the other estimators is more as it incorporates higher number of points. Note that for this case study computation time for RA-CKF (5^{th} degree), RA-CQKF (3^{rd} degree, 3^{rd} order) are 36.55% and 29.75% of that for RA-CQKF (5^{th} degree, 3^{rd} order) respectively. Simulation is carried out using MATLAB (version 7.9.0.529) in a computer with specifications Intel®, Core (TM) 2 Duo CPU, 2.8 GHz, 2 GB RAM.

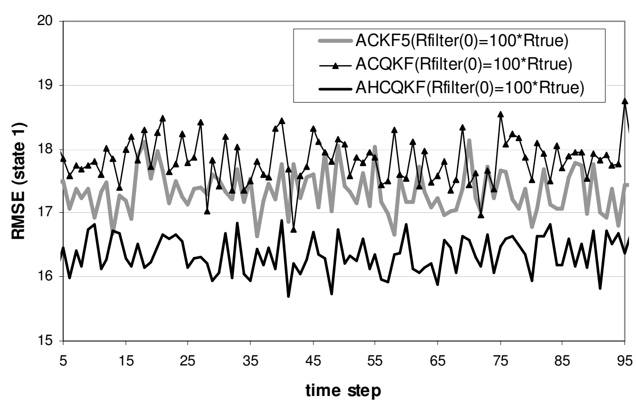


Fig. 7.9: Plots of RMSE of 1^{st} state for different adaptive estimators

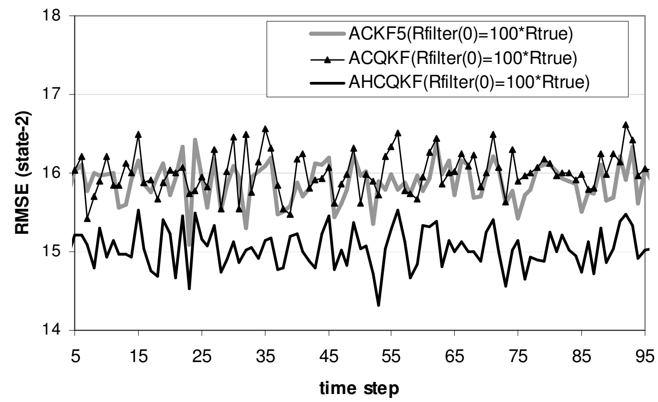


Fig. 7.10: Plots of RMSE of 2nd state for different adaptive estimators

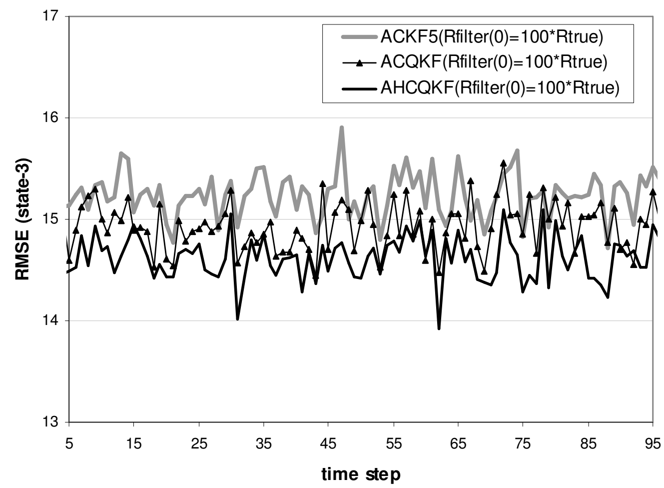


Fig. 7.11: Plots of RMSE of 3rd state for different adaptive estimators

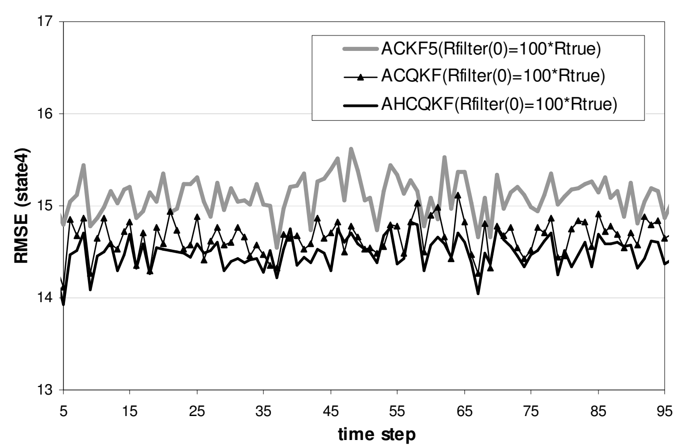


Fig. 7.12: Plots of RMSE of 4th state for different adaptive estimators

7.5.3 Demonstration of R adaptive estimators in square root framework

Here we consider an aircraft tracking problem which has been used to validate adaptive Gauss Hermite filter in square root framework. The states and turn rate of an aircraft executing a maneuvering turn with an unknown and time varying turn rate has to be estimated. The adaptive versions of cubature filter and Cubature Quadrature Kalman filters in square root framework which are formulated in this chapter have been validated using this case study.

Performance comparison of $\boldsymbol{R}$ adaptive versions of square root GHF with 3rd order accuracy (RA-SR-GHF), square root CKF with 3rd degree accuracy (RA-SR-CKF) and square root CQKF (based on 3rd degree spherical rule, 2nd order radial rule and denoted as RA-SR-CQKF) have been carried out for the above estimation problem where the measurement noise covariance of one of the radar remains unknown.

The candidate filters are evaluated with the help of a Monte Carlo study with 10000 Monte Carlo runs. The element of measurement noise covariance, $\boldsymbol{R}(2,2)$ is considered to be unknown as in chapter 6. The unknown element $\boldsymbol{R}(2,2)$ is assumed with an arbitrary value which is 5 times less than the truth value for this case study. The RMS error of position, velocity estimation and also the RMS error of estimated turn rate are presented for all three filters. In addition to the plots of RMS errors the percentage of track loss is also provided for each estimator. The track loss occurs because of non uniqueness of the measurements as explained in chapter 3 and chapter 6. The RMSE are presented excluding the track loss cases.

From the Fig. 7.13 – 7.15 RMSE for all three estimators are found comparably same. However, in Fig. 7.13 the RMSE of RA-SR-CKF is marginally higher than the other two plots. Therefore, the performance of these estimators cannot be analyzed on the basis of RMSE only. In this perspective percentage of track loss gives us a cue to judge the performance of these competing algorithms. It is to be noted that the percentage of track loss for RA-SR-GHF is considerably low compared to RA-SR-CKF and RA-SR-CQKF. However, percentage of track loss for RA-SR-CQKF is lower than RA-SR-CKF. It is interesting to note that RA-SR-CQKF is performance wise as good as RA-SR-GHF at a less computation effort. Table 6.2 is provided in support of the above statements.

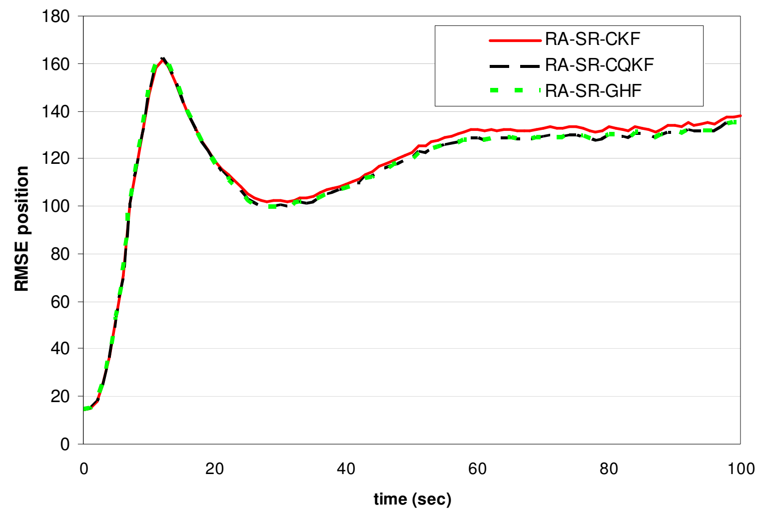


Fig. 7.13: Comparison of RMS error (altitude estimation) of ASRCQKF, ASRGHF3 & SRCKF

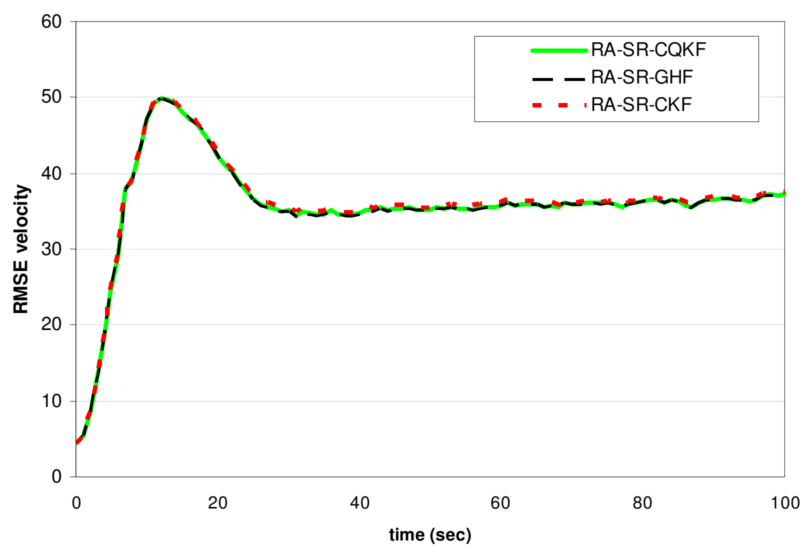


Fig. 7.14: Comparison of RMS error (velocity estimation) of ASRCQKF, ASRGHF3 & SRCKF

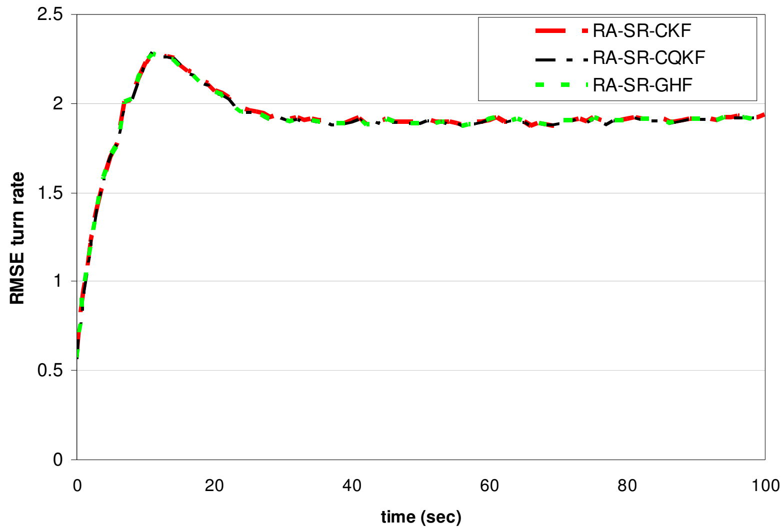


Fig. 7.15: Comparison of RMS error (turn rate estimation) of ASRCQKF, ASRGHF3 & SRCKF

Table-7.2: Percentage of track loss and computation time for adaptive estimators

Estimators	Percentage of track loss	Computation time*	Points for n=5, n_r=2
RA-SR-CKF	1.55%	6.30%	10 (2n)
RA-SR-CQKF	1.44%	8.13%	20 (2nn_r)
RA-SR-GHF	1.34%	100%	243 (3^n)

* Computation time is presented as the percentage of the computation time of RA-SR-GHF. Simulation is carried out using MATLAB (version 7.9.0.529) in a computer with specifications Intel®, Core (TM) 2 Duo CPU, 2.8 GHz, 2 GB RAM.

7.6 Discussions and Conclusions

In this chapter adaptive versions of Cubature filters (3rd degree and 5th degree) and Cubature Quadrature Kalman filters (3rd degree and 5th degree) have been formulated from the general framework. Relative performance comparison with previously reported adaptive nonlinear filters is carried out with the help of several nonlinear state estimation problems. Simulation results from all the case studies indicate that the performance of adaptive CKF (5th degree) is demonstrably better than Adaptive CKF (3rd degree) and AUKF. Performance of ACQKF (3rd degree and 5th degree) is found to comparable with ACKF (5th degree) and sometimes superior to it. Below we present in detail the significant findings from the case studies.

- $\boldsymbol{Q}$ adaptive CKF (3rd degree) in square root framework is found to outperform its non-adaptive version in the face of unknown noise covariance. When the filters reaches steady state performance of the proposed filter without knowledge of $\boldsymbol{Q}$ is observed to be comparable with that of its non-adaptive version in the ideal situation when it has the complete knowledge of $\boldsymbol{Q}$.

- $\boldsymbol{R}$ adaptive version of CKF and CQKF are demonstrated using a number of case studies. For the first order estimation problem it is observed that the performance of RA-CKF (5th degree) has the same level of accuracy with that of RA-GHF (3rd order). However, RA-CQKF (3rd degree, 2nd order) is performance wise superior to RA-CKF (5th degree) and have equivalent accuracy of RA-GHF (5th order) at less computational cost. Performance of adaptive higher order Cubature Quadrature Kalman filter, RA-CQKF (5th degree, 3rd order) is comparable with RA-CQKF (3rd degree, 2nd order). However, the computation cost of the higher degree version of RA-CQKF is more in presence of higher number of cubature points. Performance of all other adaptive estimators excels over RA-UKF for this specific case study where system dynamics suffers from significant nonlinearity.
- In the second case study the superiority of RA-CKF (5th degree) over RA-UKF is exhibited using RMSE analysis of the state estimates of the Lorenz attractor. However, for this estimation problem RA-CQKF (5th degree, 3rd order) could not show improved performance compared to RA-CKF (5th degree). Nevertheless, its performance is comparable with RA-UKF.
- To demonstrate the superiority of adaptive version of higher order Cubature Quadrature Kalman filter, RA-CQKF (5th degree, 3rd order), over the competing algorithms of RA-CKF (5th degree) and RA-CQKF (3rd degree, 3rd order) a fourth order estimation problem is considered. It is observed from the RMS error analysis that RA-CQKF (5th degree, 3rd order) can present marginally improved estimation performance compared to the other competing algorithms at the cost of additional computation effort.
- $\boldsymbol{R}$ adaptive versions GHF (3rd order), CKF (3rd degree), CQKF (3rd degree, 2nd order) in the square root framework is also formulated in this chapter and validated using aircraft tracking problem. Superiority of RA-SR-CQKF and RA-SR-GHF over RA-SR-CKF is demonstrated. It is interesting to note that RA-SR-CQKF can present satisfactory estimation performance (performance is nearly close to RA-SR-GHF) at significantly less computational burden compared to RA-SR-GHF. The square root version of other adaptive nonlinear filters viz. AUKF, ACKF (5th degree) and ACQKF (5th degree) have

not been formulated due to presence of the negative weights for some of the points. These algorithms cannot be formulated directly from the proposed general framework in chapter4 as the presence of negative weights demand modifications in the algorithm. This work may be considered as the future scope of this dissertation.

From the above findings it may be concluded that Adaptive CKF (5^{th} degree) and Adaptive CQKF are strong candidates during estimation of multi dimensional systems with significant nonlinearity for their better accuracy and economic computation (compared to the computationally intensive Adaptive GHF). The square root version of Adaptive CQKF (3^{rd} degree) is also recommended for the estimation problems where their standard error covariance forms may terminate due to loss of positive definiteness.

Chapter 8: Adaptive Nonlinear Filters for Non-additive Noise

8.1 Chapter Introduction

This chapter presents a general framework for adaptive nonlinear filters to suit nonlinear signal models where the system dynamics and observation equations are nonlinear function of states as well as noise. Situations have been considered where knowledge of the covariance of non-additive noise remains unavailable. For state estimation of the systems with such signal models the proposed algorithms for adaptive nonlinear filter with additive noise no longer remain appropriate. The adaptation algorithms for the additive noise derived in chapter 4 necessitate substantial modifications for non-additive noise. In addition to this the underlying framework of non-adaptive nonlinear filters also needs to be modified to suit non-additive noise.

The redesigning of the adaptation algorithms which are essential for non-additive noise have been presented in this chapter and subsequently a new general framework for adaptive nonlinear filters with non-additive noise has been developed. The derivation of the adaptation algorithms are based on Maximum Likelihood Estimation (MLE) method. The general framework proposed in this chapter uses the non-adaptive nonlinear filters with the "augmented form" as its core. In this particular form the process noise and measurement noise are augmented with the state vector in time update steps and measurement update steps respectively to make the estimation algorithm suitable for the state and the observation equations which are nonlinear function of states and the noise terms. The non-adaptive nonlinear filtering algorithm has been reported in [Sarkka2013a]. The same algorithm is considered here as an underlying framework.

The general framework includes algorithms for adaptation of the covariance of process noise as well as the measurement noise which are non-additive in nature. With the help of general framework proposed in this chapter formulation of variants of $\boldsymbol{Q}$ adaptive and $\boldsymbol{R}$ adaptive sigma point filters for non-additive noise may be possible following different methods of sigma point selection discussed in the previous chapters. For illustration $\boldsymbol{R}$ adaptive estimators have been derived from the general framework incorporating sigma points and weights which are chosen following 3rd degree and 5th degree cubature rule. The derived algorithms are validated in simulation using a realistic estimation problem where the measurement equation is indeed a nonlinear function of states and measurement noise.

Additionally algorithms for adaptive Divided Difference filter (ADDF) with non-additive process and measurement noise are also formulated. These algorithms, although conceptually same, cannot be directly obtained using the proposed general framework. This is because of the fact that non-adaptive DDF is based on Taylor series approximation and uses Stirling's interpolation formula to replace Jacobian and Hessian matrices present in the Taylor series approximation. On the other hand the general framework includes Bayesian integrals which need to be approximated by numerical methods. The proposed algorithms based on DDF are validated with the help of different case studies where the noises are considered to be non-additive in nature.

8.2 Problem Formulation

We consider nonlinear dynamic equations of a system as given below

$$\boldsymbol{x}_k = \boldsymbol{f}(\boldsymbol{x}_{k-1}, \boldsymbol{\theta}_k) \tag{8.1}$$

$$\boldsymbol{y}_k = \boldsymbol{g}(\boldsymbol{x}_k, \boldsymbol{v}_k) \tag{8.2}$$

where $\boldsymbol{x}_k \in \Re^{n}$ is a state vector, $\boldsymbol{y}_k \in \Re^{p}$ is output vector. The zero mean process and measurement noises (assumed Gaussian) are denoted as $\boldsymbol{\theta}_k \in \Re^{q} \sim (\boldsymbol{0}, \boldsymbol{Q})$, $\boldsymbol{v}_k \in \Re^{m} \sim (\boldsymbol{0}, \boldsymbol{R}_k)$. The system dynamics and the observation equation are considered as the nonlinear function of the noise and state vectors. Consequently the noises are non-additive in nature. During the unavailability of the knowledge of covariance of process or the measurement noise it needs to be adapted at every instant of time. For joint estimation of parameters and states parameters has to be augmented with state vector as explained before in chapter 4.

For the above described estimation problem, the general frameworks for Adaptive nonlinear filters and algorithms for Adaptive Divided Difference filter are presented below which can adapt the unknown noise covariance of non-additive process or measurement noise.

8.3 General framework for adaptive filters with non-additive noise

In this section a general framework for adaptive nonlinear filters with non-additive noise is proposed. The general framework presented here is conceptually similar to that presented in chapter 4 for adaptive nonlinear filters with additive noise. However, in the presence of non-additive noise, the "augmented form" (the noise vector is augmented with the state vector) of non-adaptive nonlinear filtering algorithm [Wan2000, Sarkka2013a] is considered as an underlying framework. A detailed discussion on the non-adaptive framework is provided in above referred publications. The adaptation algorithms which have been designed for the non-additive noises are to be incorporated in the underlying framework.

8.3.1 Underlying Framework of Non-adaptive filter

In this section we present the algorithm of non-adaptive nonlinear filter with non-additive noise as given in [Wan2000, Sarkka2013]. This algorithm is used as underlying framework for the proposed general framework. The adaptation algorithms which have to be integrated in the underlying framework have been designed in the succeeding subsection.

Initialization: Initialize $\hat{\boldsymbol{x}}_0, \hat{\boldsymbol{P}}_0, \overline{\boldsymbol{Q}}_k, \overline{\boldsymbol{R}}_k$

Time update step:

For the numerical method of integration select n no. of points ($\boldsymbol{\xi}^i$) and weights (w_i) for standard normal distribution and modify in the algorithmic steps as provided below. For selection of the modified sigma points use augmented vector concatenating state and process noise vector (i.e., $[\boldsymbol{x} \quad \boldsymbol{\theta}]^T$) as

$$\tilde{\boldsymbol{\chi}}_k^i = \boldsymbol{\alpha}_k + \sqrt{\boldsymbol{P}_k^\alpha}\,\xi^i \tag{8.3}$$

Where $\boldsymbol{\alpha}_k = [\hat{\boldsymbol{x}}_{k-1} \quad \boldsymbol{0}]^T$ and $\boldsymbol{P}_k^\alpha = diag\left(\hat{\boldsymbol{P}}_{k-1}, \overline{\boldsymbol{Q}}_k\right)$

These points are propagated through nonlinear dynamic equation as $f\left(\tilde{\boldsymbol{\chi}}_k^{i,x}, \tilde{\boldsymbol{\chi}}_k^{i,q}\right)$ where $\tilde{\boldsymbol{\chi}}_k^{i,x}$ is the first n_x elements of vector $\tilde{\boldsymbol{\chi}}_k^i$ linked with states and $\tilde{\boldsymbol{\chi}}_k^{i,q}$ is the rest of the elements of $\tilde{\boldsymbol{\chi}}_k^i$ leaving the first n_x elements of vector $\tilde{\boldsymbol{\chi}}_k^i$. The vector $\tilde{\boldsymbol{\chi}}_k^{i,q}$ is linked with the process noise.

The *a priori* estimate of state is obtained as

$$\overline{\boldsymbol{x}}_k = \sum_{i=1}^{n} f\left(\tilde{\boldsymbol{\chi}}_k^{i,x}, \tilde{\boldsymbol{\chi}}_k^{i,q}\right) w_i \tag{8.4}$$

The respective *a priori* error covariance becomes

$$\overline{\boldsymbol{P}}_k = \sum_{i=1}^{n} w_i \left(f\left(\tilde{\boldsymbol{\chi}}_k^{i,x}, \tilde{\boldsymbol{\chi}}_k^{i,q}\right) - \overline{\boldsymbol{x}}_k \right) \left(f\left(\tilde{\boldsymbol{\chi}}_k^{i,x}, \tilde{\boldsymbol{\chi}}_k^{i,q}\right) - \overline{\boldsymbol{x}}_k \right)^T \tag{8.5}$$

Measurement update step:

Select the sigma points using augmented vector concatenating state and measurement noise vector (i.e., $[\boldsymbol{x} \quad \boldsymbol{\upsilon}]^T$) as

$$\overline{\boldsymbol{\chi}}_k^i = \boldsymbol{\beta}_k + \sqrt{\boldsymbol{P}_k^{\beta}}\,\xi^i \tag{8.6}$$

Where $\boldsymbol{\beta}_k = [\overline{\boldsymbol{x}}_k \quad \boldsymbol{0}]^T$ and $\boldsymbol{P}_k^{\beta} = diag\left(\overline{\boldsymbol{P}}_k, \overline{\boldsymbol{R}}_k\right)$

The *a priori* estimate of measurement is given by

$$\overline{\boldsymbol{y}}_k = \sum_{i=1}^{n} g\left(\overline{\boldsymbol{\chi}}_k^{i,x}, \overline{\boldsymbol{\chi}}_k^{i,\upsilon}\right) w_i \tag{8.7}$$

where $\overline{\boldsymbol{\chi}}_k^{i,x}$ is the first n_x elements of vector $\overline{\boldsymbol{\chi}}_k^i$ and $\overline{\boldsymbol{\chi}}_k^{i,\upsilon}$ is the rest of the elements of $\overline{\boldsymbol{\chi}}_k^i$ leaving the first n_x elements of vector $\overline{\boldsymbol{\chi}}_k^i$.The vector $\overline{\boldsymbol{\chi}}_k^{i,\upsilon}$ is linked with the measurement noise.

The innovation covariance is obtained as

$$\boldsymbol{P}_k^y = \sum_{i=1}^{n} w_i \left(g\left(\overline{\boldsymbol{\chi}}_k^{i,x}, \overline{\boldsymbol{\chi}}_k^{i,\upsilon}\right) - \overline{\boldsymbol{y}}_k \right) \left(g\left(\overline{\boldsymbol{\chi}}_k^{i,x}, \overline{\boldsymbol{\chi}}_k^{i,\upsilon}\right) - \overline{\boldsymbol{y}}_k \right)^T \tag{8.8}$$

The cross covariance is given by

$$\boldsymbol{P}_k^{xy} = \sum_{i=1}^{n} w_i \left(\overline{\boldsymbol{\chi}}_k^{i,x} - \overline{\boldsymbol{x}}_k \right) \left(g\left(\overline{\boldsymbol{\chi}}_k^{i,x}, \overline{\boldsymbol{\chi}}_k^{i,\upsilon}\right) - \overline{\boldsymbol{y}}_k \right)^T \tag{8.9}$$

The filter gain $\boldsymbol{K}_k$ can be computed using

$$\boldsymbol{K}_k = \boldsymbol{P}_k^{xy} \left(\boldsymbol{P}_k^y \right)^{-1} \tag{8.10}$$

Having computed the gain $\boldsymbol{K}_k$ the *a posteriori* estimate of the state and the error covariance can be obtained by (8.11) and (8.12) respectively.

$$\hat{\boldsymbol{x}}_k = \overline{\boldsymbol{x}}_k + \boldsymbol{K}_k \left(\boldsymbol{y}_k - \overline{\boldsymbol{y}}_k \right) \tag{8.11}$$

$$\hat{\boldsymbol{P}}_k = \overline{\boldsymbol{P}}_k - \boldsymbol{K}_k \boldsymbol{P}_k^y \boldsymbol{K}_k^T \tag{8.12}$$

8.3.2 Adaptation algorithms

The adaptation algorithms for the covariance of non-additive noise necessitate substantial modification of the algorithms that are derived for cases of additive noise. Matrix approximations of the nonlinear function of noises are required for obtaining the adapted value of process or the measurement noise covariance. To derive the matrix approximation of the nonlinear function the method of statistical linearization has been followed instead of differentiation. The concept of statistical linearization is reported in [Geist2010, Sarkka2011] and has not been elaborated here. Only a few significant points are reiterated as a background for statistical linearization for the ease of interpretation of adaptation algorithms.

8.3.2.1. Background for statistical linearization

The system dynamics and the measurement equations are expressed as the nonlinear function of the vectors of system states and noise. About a particular operating point this nonlinear relation can be expressed in terms of the linearized relation as presented in (8.13).

Let us consider the nonlinear system dynamics $\boldsymbol{x}_k = \boldsymbol{f}(\boldsymbol{x}_{k-1}, \boldsymbol{\theta}_k)$ as given by (8.1). The linearlized model of (8.1) about a nominal point $\left(\hat{\boldsymbol{x}}_{k-1}, \hat{\boldsymbol{\theta}}_k\right)$ is expressed as

$$\boldsymbol{x}_k = \boldsymbol{f}\left(\hat{\boldsymbol{x}}_{k-1}, \hat{\boldsymbol{\theta}}_k\right) + \tfrac{\partial f}{\partial x} \Delta \boldsymbol{x}_{k-1} + \tfrac{\partial f}{\partial \theta} \Delta \boldsymbol{\theta}_k \qquad (8.13)$$

In the same vein the linearlized model of the observation equation given by (8.2) can also be presented about a nominal point, $(\bar{\boldsymbol{x}}_k, \bar{\boldsymbol{v}}_k)$, as

$$\boldsymbol{y}_k = \boldsymbol{g}(\bar{\boldsymbol{x}}_k, \bar{\boldsymbol{v}}_k) + \tfrac{\partial g}{\partial x} \Delta \boldsymbol{x}_k + \tfrac{\partial g}{\partial v} \Delta \boldsymbol{v}_k \qquad (8.14)$$

Here the 'Δ' linked terms are perturbation of state and noise vectors about the nominal point. However, for the sigma point based filtering algorithms the Jacobians matrices indicated in the above expressions cannot be obtained as they readily appear in EKF. Therefore the matrix approximations are done on the basis of statistical linearization. The matrices obtained by the statistical linearization [Geist2010] can be referred as 'pseudo matrix' and this concept is also introduced in [Lee2008]. These matrices are obtained by executing the following steps using statistical linearization [Geist2010] and accordingly used in the algorithm. Below we present the basic steps of statistical linearization which have been used to derive the adaptation algorithms.

Let us consider a vector which is nonlinear function of two vectors following Gaussian distribution as

$$\boldsymbol{y} = \boldsymbol{f}(\boldsymbol{x},\boldsymbol{w}), \quad \boldsymbol{y} \in \Re^m,\ \boldsymbol{x} \in \Re^n,\ \boldsymbol{w} \in \Re^n \tag{8.15}$$

where $\boldsymbol{x} \sim N(\boldsymbol{m}_x, \boldsymbol{P}_x)$ and $\boldsymbol{w} \sim N(\boldsymbol{m}_w, \boldsymbol{P}_w)$. We can express the nonlinear relation in (8.15) by the linear approximation as

$$\boldsymbol{f}(\boldsymbol{x},\boldsymbol{w}) = \boldsymbol{f}(\boldsymbol{m}_x,\boldsymbol{m}_w) + \boldsymbol{\Phi}\delta\boldsymbol{x} + \boldsymbol{\Omega}\delta\boldsymbol{w} \tag{8.16}$$

where

$\boldsymbol{\Phi}$ and $\boldsymbol{\Omega}$ are matrix approximation of $\boldsymbol{f}(\boldsymbol{x},\boldsymbol{w})$ with respect to $\boldsymbol{x}$ and $\boldsymbol{w}$ respectively. $\boldsymbol{\Phi}$ and $\boldsymbol{\Omega}$ are obtained as

$$\boldsymbol{\Phi} = E\left[(\boldsymbol{f}(\boldsymbol{x},\boldsymbol{m}_w) - \boldsymbol{f}(\boldsymbol{m}_x,\boldsymbol{m}_w))\delta\boldsymbol{x}^T\right]\boldsymbol{P}_x^{-1} \tag{8.17}$$

and

$$\boldsymbol{\Omega} = E\left[(\boldsymbol{f}(\boldsymbol{m}_x,\boldsymbol{w}) - \boldsymbol{f}(\boldsymbol{m}_x,\boldsymbol{m}_w))\delta\boldsymbol{w}^T\right]\boldsymbol{P}_w^{-1} \tag{8.18}$$

For the situations with zero mean unity variance noise vector i.e., when $\boldsymbol{w} \sim N(\boldsymbol{0},\boldsymbol{I})$, $\boldsymbol{\Omega}$ can be expressed as

$$\boldsymbol{\Omega} = E\left[(\boldsymbol{f}(\boldsymbol{m}_x,\boldsymbol{w}) - \boldsymbol{f}(\boldsymbol{m}_x,\boldsymbol{m}_w))\delta\boldsymbol{w}^T\right] \tag{8.19}$$

Alternatively, we can also write

$$\boldsymbol{\Omega}\boldsymbol{\Omega}^T = E\left[(\boldsymbol{f}(\boldsymbol{m}_x,\boldsymbol{w}) - \boldsymbol{f}(\boldsymbol{m}_x,\boldsymbol{m}_w))(\boldsymbol{f}(\boldsymbol{m}_x,\boldsymbol{w}) - \boldsymbol{f}(\boldsymbol{m}_x,\boldsymbol{m}_w))^T\right]$$

Therefore, $\boldsymbol{\Omega}$ can also be obtained as

$$\boldsymbol{\Omega} = \textit{Cholesky factorization}\left(E\left[(\boldsymbol{f}(\boldsymbol{m}_x,\boldsymbol{w}) - \boldsymbol{f}(\boldsymbol{m}_x,\boldsymbol{m}_w))(\boldsymbol{f}(\boldsymbol{m}_x,\boldsymbol{w}) - \boldsymbol{f}(\boldsymbol{m}_x,\boldsymbol{m}_w))^T\right]\right) \tag{8.20}$$

The advantage of the above approach is that the approximated matrix is always a square matrix and does not require matrix pseudo inverse when matrix inversion is required. This approach is followed in the adaptation algorithm as presented in the next subsection.

8.3.2.2. *R* adaptation

For $\boldsymbol{R}$ adaptation replace $\overline{\boldsymbol{R}}_k$ with adapted $\hat{\boldsymbol{R}}_{k-1}$ of previous instant in (8.6) of the measurement update step. The expression of adapted $\boldsymbol{R}$ can be obtained as follows. Only for

k=1 the value of $\hat{\boldsymbol{R}}_{k-1}$ has to be assumed arbitrarily. Here we propose residual based $\boldsymbol{R}$ adaptation algorithm in the following steps.

Select the sigma points using augmented random variable $[\boldsymbol{x} \quad \boldsymbol{\upsilon}]^T$ as

$$\hat{\boldsymbol{\chi}}_k^i = \hat{\boldsymbol{\beta}}_k + \sqrt{\hat{\boldsymbol{P}}_k^{\beta}}\,\xi^i \tag{8.21}$$

Where $\hat{\boldsymbol{\beta}}_k = [\hat{\boldsymbol{x}}_k \quad \boldsymbol{0}]^T$ and $\hat{\boldsymbol{P}}_k^{\beta} = diag\left(\hat{\boldsymbol{P}}_k, \hat{\boldsymbol{R}}_{k-1}\right)$ where $\hat{\boldsymbol{\chi}}_k^{i,x}$ is the first n_x elements of vector $\hat{\boldsymbol{\chi}}_k^i$ and $\hat{\boldsymbol{\chi}}_k^{i,\upsilon}$ is the rest of the elements of $\hat{\boldsymbol{\chi}}_k^i$ linked with the measurement noise.

The *a posteriori* estimate of measurement is obtained by

$$\hat{\boldsymbol{y}}_k = \sum_{i=1}^{n} g\left(\hat{\boldsymbol{\chi}}_k^{i,x}, \hat{\boldsymbol{\chi}}_k^{i,\upsilon}\right) w_i \tag{8.22}$$

The residual is obtained as

$$\boldsymbol{\rho}_k = \boldsymbol{y}_k - \hat{\boldsymbol{y}}_k \tag{8.23}$$

The residual covariance from a sliding window is computed as

$$\hat{\boldsymbol{C}}_{\rho} = \frac{1}{L} \sum_{j=j-L+1}^{j} \boldsymbol{\rho}_k \boldsymbol{\rho}_k^T \tag{8.24}$$

Select another set of sigma points using augment random variable as

$$\breve{\boldsymbol{\chi}}_k^i = \boldsymbol{\gamma}_k + \sqrt{\boldsymbol{P}_k^{\gamma}}\,\xi^i \tag{8.25}$$

Where $\boldsymbol{\gamma}_k = [\hat{\boldsymbol{x}}_k \quad \boldsymbol{0}]^T$ and $\boldsymbol{P}_k^{\gamma} = diag\left(\hat{\boldsymbol{P}}_k, \boldsymbol{I}\right)$

Note that here we introduce a dummy variable which is following a standard normal distribution. Augmentation of this dummy variable in (8.25) enables us to obtain the matrix approximation of the nonlinear function of noise following (8.26) and (8.28).

Transform the sigma points through the function $\boldsymbol{g}(\cdot)$ as $\boldsymbol{g}\left(\hat{\boldsymbol{x}}_k, \breve{\boldsymbol{\chi}}_k^{i,\upsilon}\right)$ and $\boldsymbol{g}\left(\breve{\boldsymbol{\chi}}_k^{i,x}, \boldsymbol{0}\right)$ where $\breve{\boldsymbol{\chi}}_k^{i,x}$ is the first n elements of vector $\breve{\boldsymbol{\chi}}_k^i$ and $\breve{\boldsymbol{\chi}}_k^{i,\upsilon}$ consists rest of the elements. This indicates that for the first case the sigma points are used only for the matrix approximation of the nonlinear function with respect to the measurement noise keeping the state vector constant at $\hat{\boldsymbol{x}}_k$. For the second case the reverse has been done, i.e., sigma points are used only for the matrix approximation of the nonlinear function with respect to state estimate keeping the measurement noise vector constant at its mean value. Using this two differently transformed set of sigma point we get the following equations

$$\hat{\boldsymbol{y}}_{\boldsymbol{k}}^{\hat{x}} = \sum_{i=1}^{n} \boldsymbol{g}\left(\hat{\boldsymbol{x}}_{\boldsymbol{k}}, \breve{\boldsymbol{\chi}}_{\boldsymbol{k}}^{i,\boldsymbol{v}}\right) w_i \tag{8.26}$$

$$\hat{\boldsymbol{y}}_{\boldsymbol{k}}^{\overline{v}} = \sum_{i=1}^{n} \boldsymbol{g}\left(\breve{\boldsymbol{\chi}}_{\boldsymbol{k}}^{i,x}, \boldsymbol{0}\right) w_i \tag{8.27}$$

To calculate the matrix approximation of the nonlinear function of measurement noise, first compute the matrix as given below. It is to be noted here that (8.18) and (8.20) have been used for statistical linearization.

$$\hat{\boldsymbol{P}}_{\boldsymbol{k}}^{\boldsymbol{v}} = \sum_{i=1}^{n} \left(\boldsymbol{g}\left(\hat{\boldsymbol{x}}_{\boldsymbol{k}}, \breve{\boldsymbol{\chi}}_{\boldsymbol{k}}^{i,\boldsymbol{v}}\right) - \hat{\boldsymbol{y}}_{\boldsymbol{k}}^{\hat{x}}\right)\left(\boldsymbol{g}\left(\hat{\boldsymbol{x}}_{\boldsymbol{k}}, \breve{\boldsymbol{\chi}}_{\boldsymbol{k}}^{i,\boldsymbol{v}}\right) - \hat{\boldsymbol{y}}_{\boldsymbol{k}}^{\hat{x}}\right)^T w_i \tag{8.28}$$

Now find the required matrix approximation as the Cholesky factor of $\hat{\boldsymbol{P}}_{\boldsymbol{k}}^{\boldsymbol{v}}$ such that $\hat{\boldsymbol{P}}_{\boldsymbol{k}}^{\boldsymbol{v}} = \hat{\boldsymbol{S}}_{\boldsymbol{k}}^{\boldsymbol{v}}\left(\hat{\boldsymbol{S}}_{\boldsymbol{k}}^{\boldsymbol{v}}\right)^T$ (8.29)

The error covariance of *a posteriori* estimate of measurement is obtained using

$$\hat{\boldsymbol{P}}_{\boldsymbol{k}}^{\boldsymbol{g}} = \sum_{i=1}^{n} \left(\boldsymbol{g}\left(\breve{\boldsymbol{\chi}}_{\boldsymbol{k}}^{i,x}, \boldsymbol{0}\right) - \hat{\boldsymbol{y}}_{\boldsymbol{k}}^{\overline{v}}\right)\left(\boldsymbol{g}\left(\breve{\boldsymbol{\chi}}_{\boldsymbol{k}}^{i,x}, \boldsymbol{0}\right) - \hat{\boldsymbol{y}}_{\boldsymbol{k}}^{\overline{v}}\right)^T w_i \tag{8.30}$$

The adapted measurement noise covariance is obtained using the following relation

$$\hat{\boldsymbol{R}}_{\boldsymbol{k}} = \left(\hat{\boldsymbol{S}}_{\boldsymbol{k}}^{\boldsymbol{v}}\right)^{-1}\left(\hat{\boldsymbol{C}}_{\boldsymbol{\rho}} + \hat{\boldsymbol{P}}_{\boldsymbol{k}}^{\boldsymbol{g}}\right)\left(\hat{\boldsymbol{S}}_{\boldsymbol{k}}^{\boldsymbol{v}}\right)^{-\boldsymbol{T}} \tag{8.31}$$

8.3.2.3. *Q* adaptation

For $\boldsymbol{Q}$ adaptation replace $\overline{\boldsymbol{Q}}_{\boldsymbol{k}}$ with adapted $\hat{\boldsymbol{Q}}_{\boldsymbol{k}-1}$ of previous instant in (8.3) of time update step. The expression of adapted $\boldsymbol{Q}$ can be obtained as follows. Only for k=1 the value of $\hat{\boldsymbol{Q}}_{\boldsymbol{k}-1}$ has to be assumed arbitrarily.

Find the innovation sequence as

$$\vartheta_{\boldsymbol{k}} = \boldsymbol{y}_{\boldsymbol{k}} - \overline{\boldsymbol{y}}_{\boldsymbol{k}} \tag{8.32}$$

The innovation covariance from a sliding window is computed as

$$\hat{\boldsymbol{C}}_{\boldsymbol{k}}^{\vartheta} = \frac{1}{L} \sum_{j=j-L+1}^{j} \vartheta_{\boldsymbol{k}} \vartheta_{\boldsymbol{k}}^{\boldsymbol{T}} \tag{8.33}$$

Select another set of sigma points using the steps given below

$$\widehat{\boldsymbol{\chi}}_{k}^{i} = \boldsymbol{\delta}_{k} + \sqrt{\boldsymbol{P}_{k}^{\delta}}\,\boldsymbol{\xi}^{i} \quad (8.34)$$

Where $\boldsymbol{\delta}_{k} = [\hat{\boldsymbol{x}}_{k-1} \quad \boldsymbol{0}]^{T}$ and $\boldsymbol{P}_{k}^{\delta} = diag\left(\hat{\boldsymbol{P}}_{k-1}, \boldsymbol{I}\right)$

Following the same approach of $\boldsymbol{R}$ adaptation here also we introduce a dummy variable which is following a standard normal distribution.

Transform the sigma points through the function $\boldsymbol{f}(\cdot)$ as $\boldsymbol{f}\left(\hat{\boldsymbol{x}}_{k-1}, \widehat{\boldsymbol{\chi}}_{k}^{i,\boldsymbol{\theta}}\right)$ and $\boldsymbol{f}\left(\widehat{\boldsymbol{\chi}}_{k}^{i,x}, \boldsymbol{0}\right)$. These are similar steps as explained in the $\boldsymbol{R}$ adaptation algorithm. For the first case the sigma points are used only for the matrix approximation of the nonlinear function with respect to the process noise keeping the state vector constant at $\hat{\boldsymbol{x}}_{k-1}$. For the second case the reverse has been done, i.e., sigma points are used only for the matrix approximation of the nonlinear function with respect to state estimate keeping the process noise vector constant at its mean value. However, for Q adaptation the second set of sigma points is not necessary. Using only the first set of sigma point we get the following equations.

$$\hat{\boldsymbol{x}}_{k}^{\hat{x}} = \sum_{i=1}^{n} \boldsymbol{f}\left(\hat{\boldsymbol{x}}_{k-1}, \widehat{\boldsymbol{\chi}}_{k}^{i,\boldsymbol{\theta}}\right) w_{i} \quad (8.35)$$

To calculate the matrix approximation of the nonlinear function of process noise, first compute the matrix as given below using (8.18) and (8.20) for statistical linearization

$$\hat{\boldsymbol{P}}_{k}^{\boldsymbol{\theta}} = \sum_{i=1}^{n} \left(\boldsymbol{f}\left(\hat{\boldsymbol{x}}_{k-1}, \widehat{\boldsymbol{\chi}}_{k}^{i,\boldsymbol{\theta}}\right) - \hat{\boldsymbol{x}}_{k}^{\hat{x}}\right)\left(\boldsymbol{f}\left(\hat{\boldsymbol{x}}_{k-1}, \widehat{\boldsymbol{\chi}}_{k}^{i,\boldsymbol{\theta}}\right) - \hat{\boldsymbol{x}}_{k}^{\hat{x}}\right)^{T} w_{i} \quad (8.36)$$

Now find the required matrix approximation by taking the matrix square root of $\hat{\boldsymbol{P}}_{k}^{\boldsymbol{\theta}}$ by Cholesky factorization as $\hat{\boldsymbol{P}}_{k}^{\boldsymbol{\theta}} = \hat{\boldsymbol{S}}_{k}^{\boldsymbol{\theta}}\left(\hat{\boldsymbol{S}}_{k}^{\boldsymbol{\theta}}\right)^{T}$ (8.37)

The adapted process noise covariance is then obtained using the following relation

$$\hat{\boldsymbol{Q}}_{k} = \left(\hat{\boldsymbol{S}}_{k}^{\boldsymbol{\theta}}\right)^{-1} \boldsymbol{K}_{k}\hat{\boldsymbol{C}}_{\vartheta}\boldsymbol{K}_{k}^{T} \left(\hat{\boldsymbol{S}}_{k}^{\boldsymbol{\theta}}\right)^{-\boldsymbol{T}} \quad (8.38)$$

8.3.3 Derivation of adaptation algorithm

8.3.3.1. *Q* adaptation algorithm

On the availability of the matrix approximations of nonlinear function with respect to the noise terms the adaptation algorithms can be derived as follows.

For derivation of the innovation based $\boldsymbol{Q}$ adaptation algorithm following MLE method refer equation (4.35) in chapter 4. We have derived already derived the relation in (4.35) by MLE method as presented below:

$$\overline{\boldsymbol{P}}_{\boldsymbol{k}} - \hat{\boldsymbol{P}}_{\boldsymbol{k}} = \boldsymbol{K}_k \frac{1}{L} \sum_{j=j_0}^{k} \left[\vartheta_j \vartheta_j^T \right] \boldsymbol{K}_k^T \tag{8.39}$$

where $j_0 = k - L + 1$

The expression of *a priori* estimate of $\overline{\boldsymbol{P}}_{\boldsymbol{k}}$ is gets modified with the matrix approximation by statistical linearization. If $\hat{\boldsymbol{S}}_{\boldsymbol{k}}^{\boldsymbol{\theta}}$, the matrix approximation of the nonlinear function with respect to the process noise vector is available the expression of $\overline{\boldsymbol{P}}_{\boldsymbol{k}}$ becomes

$$\overline{\boldsymbol{P}}_{\boldsymbol{k}} = \sum_{i=1}^{n} \left(\boldsymbol{f}\left(\widehat{\chi}_{\boldsymbol{k}}^{i,x}, \boldsymbol{0}\right) - \hat{\boldsymbol{x}}_{\boldsymbol{k}}^{\overline{\theta}} \right) \left(\boldsymbol{f}\left(\widehat{\chi}_{\boldsymbol{k}}^{i,x}, \boldsymbol{0}\right) - \hat{\boldsymbol{x}}_{\boldsymbol{k}}^{\overline{\theta}} \right)^T w_i + \hat{\boldsymbol{S}}_{\boldsymbol{k}}^{\boldsymbol{\theta}} \hat{\boldsymbol{Q}}_{\boldsymbol{k}} \left(\hat{\boldsymbol{S}}_{\boldsymbol{k}}^{\boldsymbol{\theta}} \right)^T \tag{8.40}$$

$$\overline{\boldsymbol{P}}_{\boldsymbol{k}} - \hat{\boldsymbol{P}}_{\boldsymbol{k}} = \sum_{i=1}^{n} \left(\boldsymbol{f}\left(\widehat{\chi}_{\boldsymbol{k}}^{i,x}, \boldsymbol{0}\right) - \hat{\boldsymbol{x}}_{\boldsymbol{k}}^{\overline{\theta}} \right) \left(\boldsymbol{f}\left(\widehat{\chi}_{\boldsymbol{k}}^{i,x}, \boldsymbol{0}\right) - \hat{\boldsymbol{x}}_{\boldsymbol{k}}^{\overline{\theta}} \right)^T w_i - \hat{\boldsymbol{P}}_{\boldsymbol{k}} + \hat{\boldsymbol{S}}_{\boldsymbol{k}}^{\boldsymbol{\theta}} \hat{\boldsymbol{Q}}_{\boldsymbol{k}} \left(\hat{\boldsymbol{S}}_{\boldsymbol{k}}^{\boldsymbol{\theta}} \right)^T \tag{8.41}$$

Here, $\hat{\boldsymbol{x}}_{\boldsymbol{k}}^{\overline{\theta}} = \sum_{i=1}^{n} \boldsymbol{f}\left(\widehat{\chi}_{\boldsymbol{k}}^{i,x}, \boldsymbol{0}\right) w_i$. Note that the expression $\sum_{i=1}^{n} \left(\boldsymbol{f}\left(\widehat{\chi}_{\boldsymbol{k}}^{i,x}, \boldsymbol{0}\right) - \hat{\boldsymbol{x}}_{\boldsymbol{k}}^{\overline{\theta}} \right) \left(\boldsymbol{f}\left(\widehat{\chi}_{\boldsymbol{k}}^{i,x}, \boldsymbol{0}\right) - \hat{\boldsymbol{x}}_{\boldsymbol{k}}^{\overline{\theta}} \right)^T w_i$ is equivalent to the matrix $\overline{\boldsymbol{P}}_{\boldsymbol{k}}^{\boldsymbol{f}}$ referred in chapter 4. It is mentioned earlier in chapter 4 that during the steady state reached by the filter the difference between $\overline{\boldsymbol{P}}_{\boldsymbol{k}}^{\boldsymbol{f}}$ and $\hat{\boldsymbol{P}}_{\boldsymbol{k}}$ are negligible. Therefore, both the terms can be ignored from (8.41) leaving only the term $\hat{\boldsymbol{S}}_{\boldsymbol{k}}^{\boldsymbol{\theta}} \hat{\boldsymbol{Q}}_{\boldsymbol{k}} \left(\hat{\boldsymbol{S}}_{\boldsymbol{k}}^{\boldsymbol{\theta}} \right)^T$. Therefore, the expression in (8.39) can be represented as

$$\hat{\boldsymbol{S}}_{\boldsymbol{k}}^{\boldsymbol{\theta}} \hat{\boldsymbol{Q}}_{\boldsymbol{k}} \left(\hat{\boldsymbol{S}}_{\boldsymbol{k}}^{\boldsymbol{\theta}} \right)^T = \boldsymbol{K}_k \frac{1}{N} \sum_{j=j_0}^{k} \left[\vartheta_j \vartheta_j^T \right] \boldsymbol{K}_k^T \tag{8.42}$$

Hence the adapted $\boldsymbol{Q}$ matrix is obtained as

$$\hat{\boldsymbol{Q}}_{\boldsymbol{k}} = \left(\hat{\boldsymbol{S}}_{\boldsymbol{k}}^{\boldsymbol{\theta}} \right)^{-1} \boldsymbol{K}_{\boldsymbol{k}} \hat{\boldsymbol{C}}_{\vartheta} \boldsymbol{K}_{\boldsymbol{k}}^{\boldsymbol{T}} \left(\hat{\boldsymbol{S}}_{\boldsymbol{k}}^{\boldsymbol{\theta}} \right)^{-\boldsymbol{T}} \tag{8.43}$$

8.3.3.2. ***R*** **adaptation algorithm**

For derivation of the residual based $\boldsymbol{R}$ adaptation algorithm using MLE method refer equation (4.73) in chapter 4. We have already derived the relation in (4.73) by MLE method as presented below:

$$\sum_{j=j_0}^{k}\left[tr\left\{\left(\boldsymbol{C}_j^{\rho}\right)^{-1}\left[\boldsymbol{C}_j^{\rho}-\boldsymbol{\rho}_j\boldsymbol{\rho}_j^T\right]\left(\boldsymbol{C}_j^{\rho}\right)^{-1}\left(\boldsymbol{I}+\boldsymbol{\Psi}_j\boldsymbol{K}_j\boldsymbol{K}_j^T\boldsymbol{\Psi}_j^T\right)\right\}\right]=0 \tag{8.44}$$

where $j_0 = k - L + 1$

For non-additive measurement noise with the availability of $\hat{\boldsymbol{S}}_k^{\boldsymbol{v}}$, the matrix approximation of nonlinear function with respect to the measurement noise, (4.73) gets changed and represented as

$$\sum_{j=j_0}^{k}\left[tr\left\{\left(\boldsymbol{C}_j^{\rho}\right)^{-1}\left[\boldsymbol{C}_j^{\rho}-\boldsymbol{\rho}_j\boldsymbol{\rho}_j^T\right]\left(\boldsymbol{C}_j^{\rho}\right)^{-1}\left(\boldsymbol{I}+\boldsymbol{\Psi}_j\boldsymbol{K}_j\hat{\boldsymbol{S}}_k^{\boldsymbol{v}}\left(\hat{\boldsymbol{S}}_k^{\boldsymbol{v}}\right)^T\boldsymbol{K}_j^T\boldsymbol{\Psi}_j^T\right)\right\}\right]=0 \tag{8.45}$$

The expression $\left(\boldsymbol{I}+\boldsymbol{\Psi}_j\boldsymbol{K}_j\hat{\boldsymbol{S}}_k^{\boldsymbol{v}}\left(\hat{\boldsymbol{S}}_k^{\boldsymbol{v}}\right)^T\boldsymbol{K}_j^T\boldsymbol{\Psi}_j^T\right)$ being positive definite, the above expression becomes zero only when

$$\sum_{j=j_0}^{k}\left[\boldsymbol{C}_j^{\rho}-\boldsymbol{\rho}_j\boldsymbol{\rho}_j^T\right]=0 \tag{8.46}$$

$$\hat{\boldsymbol{C}}_k^{\rho}=\frac{1}{L}\sum_{j=j_0}^{k}\boldsymbol{\rho}_j\boldsymbol{\rho}_j^T \tag{8.47}$$

If the nonlinear function of measurement noise is approximated by the approximated matrix $\hat{\boldsymbol{S}}_k^{\boldsymbol{v}}$ the residual covariance can be presented as

$\boldsymbol{C}_k^{\rho}=\hat{\boldsymbol{S}}_k^{\boldsymbol{v}}\boldsymbol{R}_k\left(\hat{\boldsymbol{S}}_k^{\boldsymbol{v}}\right)^T-\hat{\boldsymbol{P}}_k^{g}$, alternatively, $\boldsymbol{C}_k^{\rho}=\hat{\boldsymbol{S}}_k^{\boldsymbol{v}}\boldsymbol{R}_k\left(\hat{\boldsymbol{S}}_k^{\boldsymbol{v}}\right)^T-\boldsymbol{\Psi}_k\hat{\boldsymbol{P}}_k\boldsymbol{\Psi}_k^T$.

See the derivation of (4.91) in chapter 4 for reference. The matrix $\hat{\boldsymbol{P}}_k^{g}$, error covariance of *a posteriori* estimate of state, is equivalent to $\boldsymbol{\Psi}_k\hat{\boldsymbol{P}}_k\boldsymbol{\Psi}_k^T$ where $\boldsymbol{\Psi}_k$ is the matrix approximation of the nonlinear measurement equation with respect to state estimate. Note also that $\hat{\boldsymbol{P}}_k^{g}$ is equivalent to the matrix $\hat{\boldsymbol{P}}_k^{g}$ referred in chapter 4. Here, $\hat{\boldsymbol{P}}_k^{g}$ is expressed in the algorithm by (8.30).

From (4.91) in chapter 4 we have already obtained the relation

$$\hat{\boldsymbol{S}}_k^{\boldsymbol{v}}\hat{\boldsymbol{R}}_k\left(\hat{\boldsymbol{S}}_k^{\boldsymbol{v}}\right)^T-\boldsymbol{\Psi}_k\hat{\boldsymbol{P}}_k\boldsymbol{\Psi}_k^T=\frac{1}{L}\sum_{j=j_0}^{k}\boldsymbol{\rho}_j\boldsymbol{\rho}_j^T \tag{8.48}$$

This can now be expressed as

$$\hat{S}_k^v \hat{R}_k \left(\hat{S}_k^v\right)^T - \hat{P}_k^g = \frac{1}{L}\sum_{j=j_0}^{k} \rho_j \rho_j^T \tag{8.49}$$

$$\hat{S}_k^v \hat{R}_k \left(\hat{S}_k^v\right)^T = \frac{1}{L}\sum_{j=j_0}^{k} \rho_j \rho_j^T + \hat{P}_k^g \tag{8.50}$$

Finally the adapted $\boldsymbol{R}$ can be derived as

$$\hat{R}_k = \left(\hat{S}_k^v\right)^{-1}\left(\frac{1}{L}\sum_{j=j_0}^{k} \rho_j \rho_j^T + \hat{P}_k^g\right)\left(\hat{S}_k^v\right)^{-T} \tag{8.51}$$

As formulated on the basis of residual sequence the expression of adapted $\hat{R}_k$ ensures positive definiteness.

It can be noted from the expression of adapted $\boldsymbol{Q}$ and $\boldsymbol{R}$ matrix that the matrix approximation of nonlinear function of noise which is obtained using statistical linearization has been used along with the window estimated state/measurement residual covariance matrix.

8.4 Formulation of ADDF with Non-additive Noise

The algorithm of non-adaptive Divided Difference filter is based on Taylor series approximation where the Jacobian and Hessian matrices are replaced with function evaluations with the help of Strings interpolation formula. In the work of [Norgaard2000] the algorithm for non-adaptive Divided Difference filter with non-additive noises has been presented. As the algorithm has been formulated on the basis of Taylor series approximation it is therefore not required to augment noises with state vector as done in the general framework presented above which is based on Bayesian approach. The steps for adaptive Divided Difference filter are different from that of general framework although the basic concept of filtering remains the same. The algorithmic steps are provided below.

8.4.1 Non-adaptive DDF framework

Initialization: Initialize $\hat{x}_0, \hat{P}_0, \overline{Q}_0, \overline{R}_0$

Time update step:

Given $\hat{P}_{k-1}$, compute the Cholesky Factor $\hat{S}_x(\mathrm{k}-1)$ such that

$$\hat{P}_k = \hat{S}_x(\mathrm{k}-1)\,\hat{S}_x^T(\mathrm{k}-1) \tag{8.52}$$

Given $\overline{\boldsymbol{Q}}_k$, compute the Cholesky Factor $\boldsymbol{S}_w(\mathrm{k})$ such that

$$\boldsymbol{Q} = \boldsymbol{S}_w(\mathrm{k})\boldsymbol{S}_w^T(\mathrm{k}) \tag{8.53}$$

Propagation of a priori estimate of state:

The expression of *a priori* state is

$$\begin{aligned}\overline{\boldsymbol{x}}_k = \tfrac{h^2-n-n_w}{h^2} \boldsymbol{f}(\hat{\boldsymbol{x}}_{k-1}, \overline{\boldsymbol{w}}_k) + \tfrac{1}{2h^2}\sum_{p=1}^{n}\left\{\boldsymbol{f}\left(\hat{\boldsymbol{x}}_{k-1} + h\hat{\boldsymbol{s}}_{x,p}, \overline{\boldsymbol{w}}_k\right) + \boldsymbol{f}\left(\hat{\boldsymbol{x}}_{k-1} - h\hat{\boldsymbol{s}}_{x,p}, \overline{\boldsymbol{w}}_k\right)\right\} \\ + \tfrac{1}{2h^2}\sum_{p=1}^{n_w}\left\{\boldsymbol{f}\left(\hat{\boldsymbol{x}}_{k-1}, \overline{\boldsymbol{w}}_k + h\hat{\boldsymbol{s}}_{w,p}\right) + \boldsymbol{f}\left(\hat{\boldsymbol{x}}_{k-1}, \overline{\boldsymbol{w}}_k - h\hat{\boldsymbol{s}}_{w,p}\right)\right\}\end{aligned} \tag{8.54}$$

$\overline{w}_k$, the mean of the process noise is zero for the consideration of a zero mean noise. $\hat{\boldsymbol{s}}_{x,p}$ is p^{th} column of $\hat{\boldsymbol{S}}_x(\mathrm{k}-1)$, $\hat{\boldsymbol{s}}_{w,p}$ is p^{th} column of $\hat{\boldsymbol{S}}_w(\mathrm{k})$ and h is the appropriately chosen interval length ($h = \sqrt{3}$ for Gaussian distribution [Norgaard2000]).

Propagation of a priori error covariance:

The *a priori* error covariance become

$$\overline{\boldsymbol{P}}_k = \boldsymbol{S}_{x\hat{x}}^{(1)}(\mathrm{k})\left(\boldsymbol{S}_{x\hat{x}}^{(1)}(\mathrm{k})\right)^T + \boldsymbol{S}_{x\hat{x}}^{(2)}(\mathrm{k})\left(\boldsymbol{S}_{x\hat{x}}^{(2)}(\mathrm{k})\right)^T + \boldsymbol{S}_{xw}^{(1)}(\mathrm{k})\left(\boldsymbol{S}_{xw}^{(1)}(\mathrm{k})\right)^T + \boldsymbol{S}_{xw}^{(2)}(\mathrm{k})\left(\boldsymbol{S}_{xw}^{(2)}(\mathrm{k})\right)^T \tag{8.55}$$

$\boldsymbol{S}_{x\hat{x}}^{(1)}(\mathrm{k})$, $\boldsymbol{S}_{xw}^{(1)}(\mathrm{k})$ and $\boldsymbol{S}_{x\hat{x}}^{(2)}(\mathrm{k})$, $\boldsymbol{S}_{xw}^{(2)}(\mathrm{k})$ are the first order and the second order approximation of the square root matrix of *a priori* error covariance. The elements of these matrices are obtained from (8.56) to (8.59) for i=1,…,n and j=1,…,n.

$$\boldsymbol{S}_{x\hat{x}}^{(1)}(\mathrm{k})_{(i,j)} = \tfrac{1}{2h}\left(\boldsymbol{f}_i\left(\hat{\boldsymbol{x}}_{k-1} + h\hat{\boldsymbol{s}}_{x,j}, \overline{\boldsymbol{w}}_k\right) - \boldsymbol{f}_i\left(\hat{\boldsymbol{x}}_{k-1} - h\hat{\boldsymbol{s}}_{x,j}, \overline{\boldsymbol{w}}_k\right)\right) \tag{8.56}$$

$$\boldsymbol{S}_{x\hat{x}}^{(2)}(\mathrm{k})_{(i,j)} = \tfrac{\sqrt{h^2-1}}{2h^2}\left(\left(\boldsymbol{f}_i\left(\hat{\boldsymbol{x}}_{k-1} + h\hat{\boldsymbol{s}}_{x,j}, \overline{\boldsymbol{w}}_k\right) + \boldsymbol{f}_i\left(\hat{\boldsymbol{x}}_{k-1} - h\hat{\boldsymbol{s}}_{x,j}, \overline{\boldsymbol{w}}_k\right) - 2\boldsymbol{f}_i\left(\hat{\boldsymbol{x}}_{k-1}, \overline{\boldsymbol{w}}_k\right)\right)\right) \tag{8.57}$$

$$\boldsymbol{S}_{xw}^{(1)}(\mathrm{k})_{(i,j)} = \tfrac{1}{2h}\left(\boldsymbol{f}_i\left(\hat{\boldsymbol{x}}_{k-1}, \overline{\boldsymbol{w}}_k + h\hat{\boldsymbol{s}}_{w,j}\right) - \boldsymbol{f}_i\left(\hat{\boldsymbol{x}}_{k-1}, \overline{\boldsymbol{w}}_k - h\hat{\boldsymbol{s}}_{w,j}\right)\right) \tag{8.58}$$

$$\boldsymbol{S}_{xw}^{(2)}(\mathrm{k})_{(i,j)} = \tfrac{\sqrt{h^2-1}}{2h^2}\left(\left(\boldsymbol{f}_i\left(\hat{\boldsymbol{x}}_{k-1}, \overline{\boldsymbol{w}}_k + h\hat{\boldsymbol{s}}_{w,j}\right) + \boldsymbol{f}_i\left(\hat{\boldsymbol{x}}_{k-1}, \overline{\boldsymbol{w}}_k - h\hat{\boldsymbol{s}}_{w,j}\right) - 2\boldsymbol{f}_i\left(\hat{\boldsymbol{x}}_{k-1}, \overline{\boldsymbol{w}}_k\right)\right)\right) \tag{8.59}$$

Measurement update step:

Given $\overline{\boldsymbol{P}}_k$, Compute the Cholesky Factor $\overline{\boldsymbol{S}}_x(\mathrm{k})$

such that $\overline{\boldsymbol{P}}_k = \overline{\boldsymbol{S}}_x(\mathrm{k})\overline{\boldsymbol{S}}_x^T(\mathrm{k})$ (8.60)

Compute $\boldsymbol{S}_v(\mathrm{k})$ such that $\hat{\boldsymbol{R}}_{k-1} = \boldsymbol{S}_v(\mathrm{k})\boldsymbol{S}_v^T(\mathrm{k})$ (8.61)

The a priori estimate of measurement:

The expression of the *a priori* estimate of the measurement is

$$\bar{\boldsymbol{y}}_k = \tfrac{h^2-n-n_v}{h^2}\boldsymbol{g}(\bar{\boldsymbol{x}}_k,\bar{\boldsymbol{v}}_k) + \tfrac{1}{2h^2}\sum_{p=1}^{n}\left\{\boldsymbol{g}\left(\bar{\boldsymbol{x}}_k + h\bar{\boldsymbol{s}}_{x,p},\bar{\boldsymbol{v}}_k\right) + \boldsymbol{g}\left(\bar{\boldsymbol{x}}_k - h\bar{\boldsymbol{s}}_{x,p},\bar{\boldsymbol{v}}_k\right)\right\} + \tfrac{1}{2h^2}\sum_{p=1}^{n_v}\left\{\boldsymbol{g}\left(\bar{\boldsymbol{x}}_k,\bar{\boldsymbol{v}}_k + h\boldsymbol{s}_{v,p}\right) + \boldsymbol{g}\left(\bar{\boldsymbol{x}}_k,\bar{\boldsymbol{v}}_k - h\boldsymbol{s}_{v,p}\right)\right\} \quad (8.62)$$

$\bar{\boldsymbol{v}}_k$, the mean of the measurement noise is zero for the consideration of a zero mean Gaussian noise.

Propagation of Innovation Covariance:

The innovation covariance is computed using the following expression

$$\boldsymbol{P}_k^y = \boldsymbol{S}_{y\bar{x}}^{(1)}(\mathrm{k})\left(\boldsymbol{S}_{y\bar{x}}^{(1)}(\mathrm{k})\right)^T + \boldsymbol{S}_{y\bar{x}}^{(2)}(\mathrm{k})\left(\boldsymbol{S}_{y\bar{x}}^{(2)}(\mathrm{k})\right)^T + \boldsymbol{S}_{yv}^{(1)}(\mathrm{k})\left(\boldsymbol{S}_{yv}^{(1)}(\mathrm{k})\right)^T + \boldsymbol{S}_{yv}^{(2)}(\mathrm{k})\left(\boldsymbol{S}_{yv}^{(2)}(\mathrm{k})\right)^T \quad (8.63)$$

$\boldsymbol{S}_{y\bar{x}}^{(1)}(\mathrm{k})$, $\boldsymbol{S}_{yv}^{(1)}(\mathrm{k})$ and $\boldsymbol{S}_{y\bar{x}}^{(2)}(\mathrm{k})$, $\boldsymbol{S}_{yv}^{(2)}(\mathrm{k})$ are the first and second order approximation of the square root matrix of innovation covariance. These elements are obtained from (8.64) to (8.67) for $i=1,\ldots,p$ and $j=1,\ldots,n$.

$$\boldsymbol{S}_{y\bar{x}}^{(1)}(\mathrm{k})_{(i,j)} = \tfrac{1}{2h}\left(\boldsymbol{g}_i\left(\bar{\boldsymbol{x}}_k + h\bar{\boldsymbol{s}}_{x,j},\bar{\boldsymbol{v}}_k\right) - \boldsymbol{g}_i\left(\bar{\boldsymbol{x}}_k - h\bar{\boldsymbol{s}}_{x,j},\bar{\boldsymbol{v}}_k\right)\right) \quad (8.64)$$

$$\boldsymbol{S}_{y\bar{x}}^{(2)}(\mathrm{k})_{(i,j)} = \tfrac{\sqrt{h^2-1}}{2h}\left(\left(\boldsymbol{g}_i\left(\bar{\boldsymbol{x}}_k + h\bar{\boldsymbol{s}}_{x,j},\bar{\boldsymbol{v}}_k\right) + \boldsymbol{g}_i\left(\bar{\boldsymbol{x}}_k - h\bar{\boldsymbol{s}}_{x,j},\bar{\boldsymbol{v}}_k\right) - 2\boldsymbol{g}_i\left(\bar{\boldsymbol{x}}_k,\bar{\boldsymbol{v}}_k\right)\right)\right) \quad (8.65)$$

$$\boldsymbol{S}_{yv}^{(1)}(\mathrm{k})_{(i,j)} = \tfrac{1}{2h}\left(\boldsymbol{g}_i\left(\bar{\boldsymbol{x}}_k,\bar{\boldsymbol{v}}_k + h\boldsymbol{s}_{v,j}\right) - \boldsymbol{g}_i\left(\bar{\boldsymbol{x}}_k,\bar{\boldsymbol{v}}_k - h\boldsymbol{s}_{v,j}\right)\right) \quad (8.66)$$

$$\boldsymbol{S}_{yv}^{(2)}(\mathrm{k})_{(i,j)} = \tfrac{\sqrt{h^2-1}}{2h}\left(\left(\boldsymbol{g}_i\left(\bar{\boldsymbol{x}}_k,\bar{\boldsymbol{v}}_k + h\boldsymbol{s}_{v,j}\right) + \boldsymbol{g}_i\left(\bar{\boldsymbol{x}}_k,\bar{\boldsymbol{v}}_k - h\boldsymbol{s}_{v,j}\right) - 2\boldsymbol{g}_i\left(\bar{\boldsymbol{x}}_k,\bar{\boldsymbol{v}}_k\right)\right)\right) \quad (8.67)$$

The cross covariance is computed as

$$\boldsymbol{P}_k^{xy} = \left[\bar{\boldsymbol{S}}_x(\mathrm{k})\right]\left[\boldsymbol{S}_{y\bar{x}}^{(1)}(\mathrm{k})\right]^T \quad (8.68)$$

In [Norgaard2000] it has been demonstrated that the cross covariance of second order DDF is same as that for first order DDF.

The filter gain $\boldsymbol{K}_k$ becomes

$$\boldsymbol{K}_k = \boldsymbol{P}_k^{xy}\left(\boldsymbol{P}_k^y\right)^{-1} \quad (8.69)$$

The *a posteriori* estimate of state is given by

$$\hat{\boldsymbol{x}}_k = \bar{\boldsymbol{x}}_k + \boldsymbol{K}_k\left(\boldsymbol{y}_k - \bar{\boldsymbol{y}}_k\right) \quad (8.70)$$

The *a posteriori* error covariance given by (8.70) ensures the positive definiteness

$$\hat{\boldsymbol{P}}_k = \boldsymbol{S}_y(\mathrm{k})\boldsymbol{S}_y^T(\mathrm{k}) \quad (8.71)$$

where

$$\boldsymbol{S}_y(\mathrm{k}) = \left[\overline{\boldsymbol{S}}_x(\mathrm{k}) - \boldsymbol{K}_k \boldsymbol{S}_{y\bar{x}}^{(1)} \quad \boldsymbol{K}_k \boldsymbol{S}_{y\bar{x}}^{(2)} \quad \boldsymbol{K}_k \boldsymbol{S}_{yv}^{(1)} \quad \boldsymbol{K}_k \boldsymbol{S}_{yv}^{(2)}\right] \quad (8.72)$$

8.4.2 Adaptation algorithm

8.4.2.1. *R* adaptation algorithm

For $\boldsymbol{R}$ adaptation replace $\overline{\boldsymbol{R}}_k$ with adapted $\hat{\boldsymbol{R}}_{k-1}$ of previous instant in (8.61) of the measurement update step. The algorithms for $\boldsymbol{R}$ adaptation are provided in the following steps.

Given the *a posteriori* error covariance $\hat{\boldsymbol{P}}_k$, compute the Cholesky Factorization $\hat{\boldsymbol{S}}_x(\mathrm{k})$ as

$$\hat{\boldsymbol{P}}_k = \hat{\boldsymbol{S}}_x(\mathrm{k})\hat{\boldsymbol{S}}_x^T(\mathrm{k}) \quad (8.73)$$

The *a posteriori* estimate of measurement is obtained as

$$\begin{aligned}\hat{\boldsymbol{y}}_k = \tfrac{h^2-n-n_v}{h^2}\boldsymbol{g}(\hat{\boldsymbol{x}}_k,\bar{\boldsymbol{v}}_k) + \tfrac{1}{2h^2}\sum_{p=1}^{n}\left\{\boldsymbol{g}(\hat{\boldsymbol{x}}_k + \boldsymbol{h}\hat{\boldsymbol{s}}_{x,p},\bar{\boldsymbol{v}}_k) + \boldsymbol{g}(\hat{\boldsymbol{x}}_k - \boldsymbol{h}\hat{\boldsymbol{s}}_{x,p},\bar{\boldsymbol{v}}_k)\right\} \\ + \tfrac{1}{2h^2}\sum_{p=1}^{n}\left\{\boldsymbol{g}(\hat{\boldsymbol{x}}_k,\bar{\boldsymbol{v}}_k + \boldsymbol{h}\hat{\boldsymbol{s}}_{x,p}) + \boldsymbol{g}(\hat{\boldsymbol{x}}_k,\bar{\boldsymbol{v}}_k - \boldsymbol{h}\hat{\boldsymbol{s}}_{x,p})\right\}\end{aligned} \quad (8.74)$$

This step is similar to the step for *a priori* estimate of measurement. $\bar{\boldsymbol{v}}_k$ is the mean of the measurement noise which is zero for the consideration of a zero mean noise.

The error covariance of *a posteriori* estimate of measurement is obtained in a similar approach for computation of $\boldsymbol{P}_k^y$.

$$\boldsymbol{S}_{y\hat{x}}^{(1)}(\mathrm{k})_{(i,j)} = \tfrac{1}{2h}\left(\boldsymbol{g}_i(\hat{\boldsymbol{x}}_k + h\hat{\boldsymbol{s}}_{x,j},\bar{\boldsymbol{v}}_k) - \boldsymbol{g}_i(\hat{\boldsymbol{x}}_k - h\hat{\boldsymbol{s}}_{x,j},\bar{\boldsymbol{v}}_k)\right) \quad (8.75)$$

$$\boldsymbol{S}_{y\hat{x}}^{(2)}(\mathrm{k})_{(i,j)} = \tfrac{\sqrt{h^2-1}}{2h}\left(\left(\boldsymbol{g}_i(\hat{\boldsymbol{x}}_k + h\hat{\boldsymbol{s}}_{x,j},\bar{\boldsymbol{v}}_k) + \boldsymbol{g}_i(\hat{\boldsymbol{x}}_k - h\hat{\boldsymbol{s}}_{x,j},\bar{\boldsymbol{v}}_k) - 2\boldsymbol{g}_i(\hat{\boldsymbol{x}}_k,\bar{\boldsymbol{v}}_k)\right)\right) \quad (8.76)$$

$\boldsymbol{S}_{y\hat{x}}^{(1)}(\mathrm{k})$ and $\boldsymbol{S}_{y\hat{x}}^{(2)}(\mathrm{k})$ are first and second order approximations of the error covariance of *a posteriori* estimate of measurement respectively.

Residual is defined as the difference between the actual measurement and the *a posteriori* estimate of measurement and expressed as

$$\boldsymbol{\rho}_k = \boldsymbol{y}_k - \hat{\boldsymbol{y}}_k \quad (8.77)$$

Estimated residual covariance can be computed from a sliding window (size *L*) as

$$\hat{\boldsymbol{C}}_k^{\rho} = \frac{1}{L}\sum_{j=k-L+1}^{k}\boldsymbol{\rho}(j)\boldsymbol{\rho}^T(j) \tag{8.78}$$

Compute the Divided Difference operator $\boldsymbol{G}_{\bar{v},k}$ from equation (8.79) as

$$\boldsymbol{G}_{\bar{v},k} = \text{Cholesky Factor of } \left(\boldsymbol{S}_{y\bar{v}}^{(1)}(\mathrm{k})\left(\boldsymbol{S}_{y\bar{v}}^{(1)}(\mathrm{k})\right)^T + \boldsymbol{S}_{y\bar{v}}^{(2)}(\mathrm{k})\left(\boldsymbol{S}_{y\bar{v}}^{(2)}(\mathrm{k})\right)^T\right) \tag{8.79}$$

where

$$\boldsymbol{S}_{y\bar{v}}^{(1)}(\mathrm{k})_{(i,j)} = \tfrac{1}{2h}\left(\boldsymbol{g}_i\left(\hat{\boldsymbol{x}}_k, \bar{\boldsymbol{v}}_k + h\boldsymbol{e}_j\right) - \boldsymbol{g}_i\left(\hat{\boldsymbol{x}}_k, \bar{\boldsymbol{v}}_k - h\boldsymbol{e}_j\right)\right) \tag{8.80}$$

$$\boldsymbol{S}_{y\bar{v}}^{(2)}(\mathrm{k})_{(i,j)} = \tfrac{\sqrt{h^2-1}}{2h}\left(\left(\boldsymbol{g}_i\left(\hat{\boldsymbol{x}}_k, \bar{\boldsymbol{v}}_k + h\boldsymbol{e}_j\right) + \boldsymbol{g}_i\left(\hat{\boldsymbol{x}}_k, \bar{\boldsymbol{v}}_k - h\boldsymbol{e}_j\right) - 2\boldsymbol{g}_i\left(\hat{\boldsymbol{x}}_k, \bar{\boldsymbol{v}}_k\right)\right)\right) \tag{8.81}$$

$\boldsymbol{e}_j$ is the j^{th} unit vector.

The divided difference operator $\boldsymbol{G}_{\bar{v},k}$ is basically the matrix approximation of the nonlinear measurement equation with respect to the measurement noise component. The above steps are similar with the matrix approximation of the nonlinear function of noise. The similar approach is followed for the matrix approximation as explain before. Here also we need to introduce a dummy variable following a standard normal distribution with zero mean unity covariance. First the square of the approximated matrix is computed using the steps of DDF. Subsequently the Cholesky factorization is taken. Alternatively, the square matrix can also be obtained by matrix triangularization as $\boldsymbol{G}_{\bar{v},k} = Triangularize\left(\left[\boldsymbol{S}_{y\bar{v}}^{(1)}(\mathrm{k}) \quad \boldsymbol{S}_{y\bar{v}}^{(2)}(\mathrm{k})\right]\right)$.

The adapted measurement noise covariance is finally derived following (8.31) as

$$\hat{\boldsymbol{R}}_k = \boldsymbol{G}_{\bar{v},k}^{-1}\left(\boldsymbol{S}_{y\hat{x}}^{(1)}(\mathrm{k})\left(\boldsymbol{S}_{y\hat{x}}^{(1)}(\mathrm{k})\right)^T + \boldsymbol{S}_{y\hat{x}}^{(2)}(\mathrm{k})\left(\boldsymbol{S}_{y\hat{x}}^{(2)}(\mathrm{k})\right)^T + \hat{\boldsymbol{C}}_k^{\rho}\right)\left(\boldsymbol{G}_{\bar{v},k}^T\right)^{-1} \tag{8.82}$$

8.4.2.2. *Q* adaptation algorithm

For $\boldsymbol{Q}$ adaptation replace $\overline{\boldsymbol{Q}}_k$ with adapted $\hat{\boldsymbol{Q}}_{k-1}$ of previous instant in (8.53) of the time update steps. The expression of adapted $\boldsymbol{Q}$ is provided in the following steps.

Consider the innovation sequence defined before as

$$\vartheta_k = \boldsymbol{y}_k - \bar{\boldsymbol{y}}_k \tag{8.83}$$

Estimated innovation covariance can be computed from a sliding window (size L) as

$$\hat{\boldsymbol{C}}_k^{\vartheta} = \frac{1}{L}\sum_{j=k-L+1}^{k}\vartheta_j\vartheta_j^T \tag{8.84}$$

Computation of Divided Difference operators of the nonlinear function of process noise:

Compute the Divided Difference operator $\boldsymbol{F}_{\overline{w},k}$ from equation (8.85)

$$\boldsymbol{F}_{\overline{w},k} = \text{Cholesky Factor of } \left(\boldsymbol{S}^{(1)}_{x\overline{w}}(\mathrm{k})\left(\boldsymbol{S}^{(1)}_{x\overline{w}}(\mathrm{k})\right)^T + \boldsymbol{S}^{(2)}_{x\overline{w}}(\mathrm{k})\left(\boldsymbol{S}^{(2)}_{x\overline{w}}(\mathrm{k})\right)^T\right) \tag{8.85}$$

where

$$\boldsymbol{S}^{(1)}_{x\overline{w}}(\mathrm{k})_{(i,j)} = \tfrac{1}{2h}\left(f_i\left(\hat{\boldsymbol{x}}_{k-1}, \overline{w}_k + h\boldsymbol{e}_j\right) - f_i\left(\hat{\boldsymbol{x}}_{k-1}, \overline{w}_k - h\boldsymbol{e}_j\right)\right) \tag{8.86}$$

$$\boldsymbol{S}^{(2)}_{x\overline{w}}(\mathrm{k})_{(i,j)} = \tfrac{\sqrt{h^2-1}}{2h}\left(\left(f_i\left(\hat{\boldsymbol{x}}_{k-1}, \overline{w}_k + h\boldsymbol{e}_j\right) + f_i\left(\hat{\boldsymbol{x}}_{k-1}, \overline{w}_k - h\boldsymbol{e}_j\right) - 2f_i\left(\hat{\boldsymbol{x}}_k, \overline{w}\right)\right)\right) \tag{8.87}$$

$\boldsymbol{e}_j$ is the j^{th} unit vector.

The divided difference operator $\boldsymbol{F}_{\overline{w},k}$ is basically the matrix approximation of the nonlinear process equation with respect to the process noise component. The above steps are similar with the matrix approximation of the nonlinear function of noise. Here also we need to introduce a dummy variable following a standard normal distribution with zero mean unity covariance. First the square of the approximated matrix is computed using the steps of DDF. Subsequently the Cholesky factorization is taken. Alternatively, the square matrix can also be obtained by matrix triangularization method as $\boldsymbol{F}_{\overline{w},k} = Triangularize\left(\left[\boldsymbol{S}^{(1)}_{x\overline{w}}(\mathrm{k}) \quad \boldsymbol{S}^{(2)}_{x\overline{w}}(\mathrm{k})\right]\right)$.

The adapted measurement noise covariance is finally derived following (8.38) as

$$\hat{\boldsymbol{Q}}_k = \boldsymbol{F}^{-1}_{\overline{w},k}\boldsymbol{K}_k\hat{\boldsymbol{C}}_{\vartheta_k}\boldsymbol{K}^T_k\left(\boldsymbol{F}^T_{\overline{w},k}\right)^{-1} \tag{8.88}$$

8.5 Characterization of proposed estimators

Algorithms for adaptive estimators for non-additive noise which are proposed in this chapter have been demonstrated in this section. The ***R*** adaptive DDF for non-additive noise has been validated with the help of two tracking problems. In the first case study a single dimensional object tracking problem is considered with a measurement equation which is nonlinear function of state and measurement noise. Note that such a measurement equation may not be a realistic one. Nevertheless, we have considered this as a toy problem to validate the performance of ***R*** adaptive DDF for non-additive process noise The other one is a bearing only tracking problem where an object moving with a constant velocity and tracked by bearing only measurement from an on board sensor perturbed by platform disturbance. For the second tracking problem the performance of ***R*** adaptive DDF has been compared with ***R*** adaptive CKF (both 3rd degree and 5th degree accuracy) which has been formulated using the proposed general framework. This estimation problem demonstrates that the measurement equation may be nonlinear function of noise in reality.

The performance of ***Q*** adaptive DDF for non-additive process noise is demonstrated with the help of another estimation problem where the states and the friction coefficient of Van der pol's oscillators are to be estimated. The system noise is considered to be non-additive in nature only for one of the states of the oscillator. Note also that such a system dynamics may not be a realistic one. Nevertheless, we have considered this again as a toy problem to validate the performance of ***Q*** adaptive DDF for non-additive process noise.

8.5.1 Characterization of R-Adaptive estimators for non-additive noise

8.5.1.1. Object Tracking Problem

In this section, the performance of proposed filter has been evaluated using the object tracking problem described in chapter 3 for the situation when measurement noise covariance remains unknown and the noise is non-additive in nature. The equation for range measurement presented by (3.22) in chapter 3 is changed as $y_k = \sqrt{M^2 + (\boldsymbol{x}_k^T \boldsymbol{e}_1 + v_k - H)^2}$. However, process noise statistics is considered to be known and additive. With the help of Monte Carlo study with 500 runs the RMS error for parameter and state estimates from the proposed ***R*** adaptive Divided Difference filter for the non-additive measurement noise has been presented. The performance comparison between the adaptive and non-adaptive version of DDF is carried out in the situations when (a) both filters do not have the knowledge of ***R***, (b) ADDF does not have the knowledge of ***R*** while the non-adaptive DDF have knowledge of ***R*** as in the ideal situation. For the case when ***R*** is unknown both the filters are initialized with an assumed value of ***R***. This value is chosen as 100 times of true ***R*** to induce sufficient uncertainty in the choice of ***R***. For sliding window based adaptation the window length is chosen as 30 time instants.

To generate the true state trajectories of target, the truth value of initial kinematic states and truth value of ballistic parameter are chosen as specified in chapter 3. Each filter has also been initialized with Gaussian prior as an initial state vector.

Fig. 8.1, Fig. 8.2, Fig. 8.3 are provided to illustrate the RMS error for the state and the parameter estimation obtained from ADDF and the non-adaptive DDF. When the measurement noise covariance is unknown performance of ADDF is significantly better than the non-adaptive DDF as the RMSE for parameter and the states settle down to a lower steady state value. ADDF can also accommodate an initial choice of ***R*** with a large error and can make the adapted value of ***R*** to converge on the truth value.

It is also important to note that during steady state the RMSE performance of ADDF with assumed ***R*** is comparably similar with DDF with known ***R*** (ideal situation). The initial mismatch between the RMSE of ADDF and DDF with known ***R*** (ideal case) is because of the time taken by the adapted ***R*** to converge on the truth value. This can be verified from Fig. 8.4.

Fig. 8.5 demonstrates the tracking performance of Adapted ***R*** when the truth value is time varying. It can also be demonstrated from this observation that the satisfactory tracking of unknown time varying ***R*** is also ensured by the algorithm of ADDF even when the measurement equation is a nonlinear function of noise.

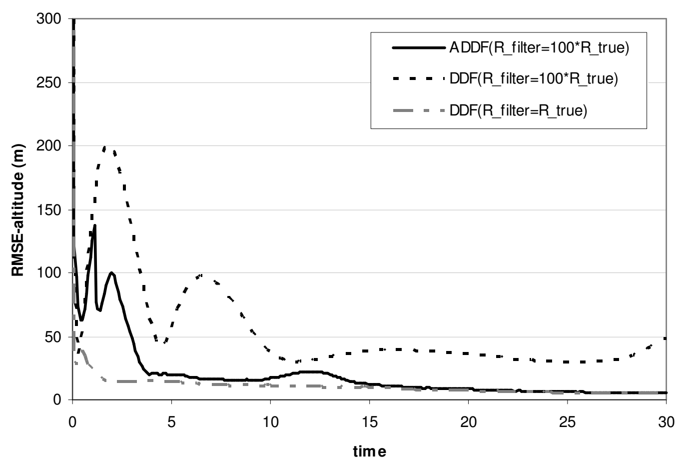


Fig. 8.1: Comparison of RMS error (altitude estimation) of ADDF & DDF for 500 MC runs

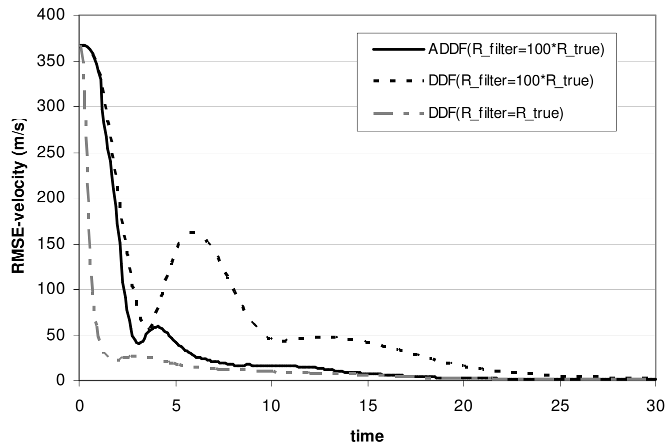


Fig. 8.2: Comparison of RMS error (velocity estimation) of ADDF & DDF for 500 MC runs

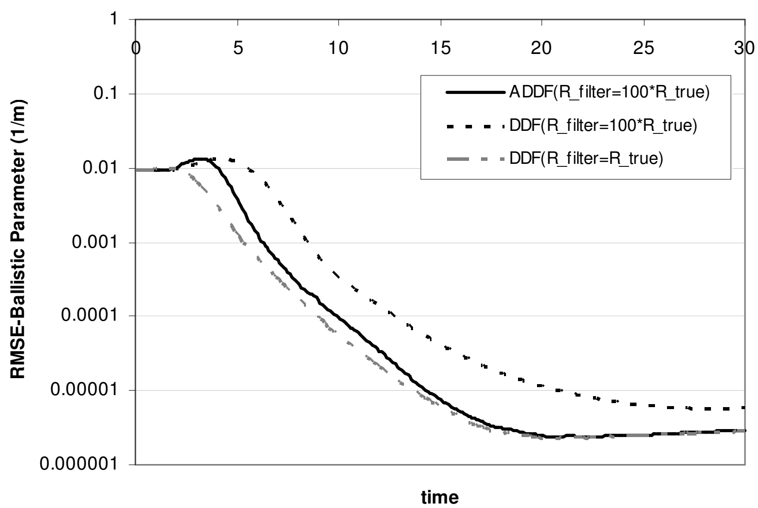


Fig. 8.3: Comparison of RMS error (parameter estimation) of ADDF & DDF for 500 MC runs

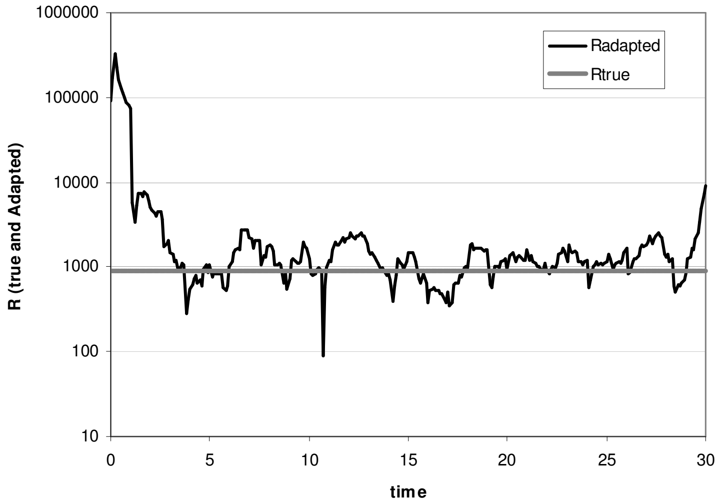


Fig. 8.4: Plot of adapted ***R*** when truth value is constant

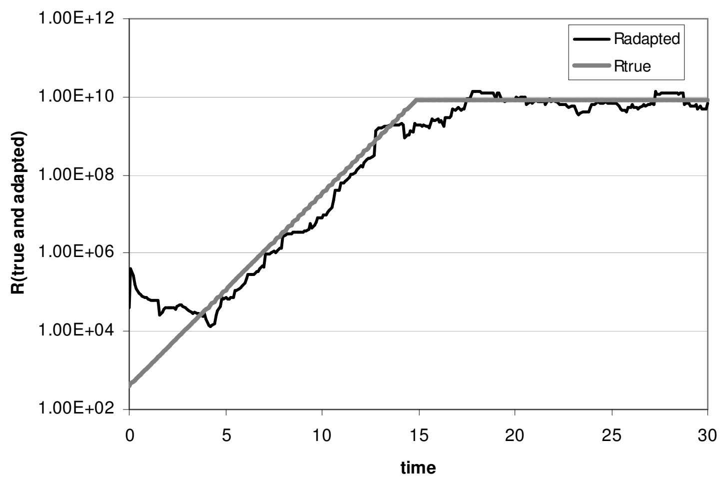


Fig. 8.5: Plot of adapted ***R*** when truth value is time varying

8.5.1.2. Bearing Only Tracking (BOT) Problem

The bearing only tracking problem which is described in chapter 3 demonstrates the situation where the measurement noise becomes non-additive in nature. The non-additive noise is approximated as an additive noise in the previous work [Sadhu2006]. This case study also gives a scope to validate the ***R*** adaptive DDF and CKF with non-additive noise when the measurement noise covariance remains unknown. The ***R*** is unknown and therefore has to be assumed arbitrarily. Here, for simulation ***R*** for the filter is chosen as a value two decades higher than the truth value of ***R*** to induce uncertainty in the initial choice. However, truth value of ***R*** is used to simulate the true measurements. Both the adaptive and the non-adaptive filter are initialized with the assumed value of measurement noise covariance i.e., $\boldsymbol{R}_{filter}(0) = 100 * \boldsymbol{R}_{true}$. The performances of the filters are evaluated on the basis of percentage of track loss and the RMS errors. The track loss case is defined as the situation when the estimation error of position at the time instant 20 is greater than 15 meter [Sadhu2006]. The track loss cases are excluded from Monte Carlo runs during the calculation of RMS errors.

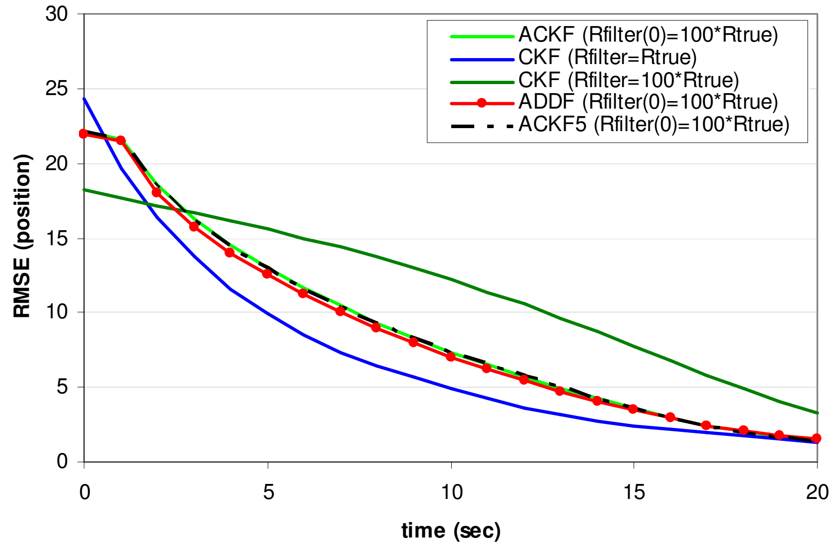


Fig. 8.6: RMSE of position of ACKF, ADDF and non-adaptive CKF

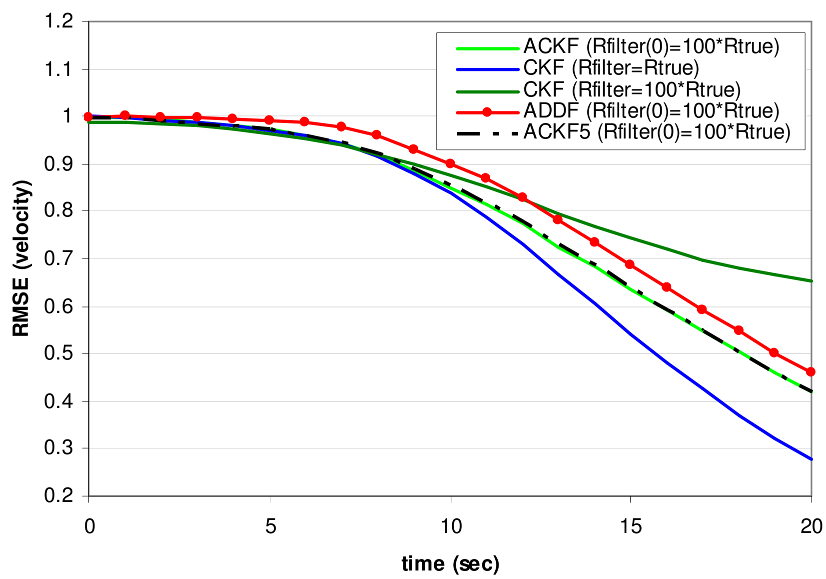


Fig. 8.7: RMSE of velocity of ACKF, ADDF and non-adaptive CKF

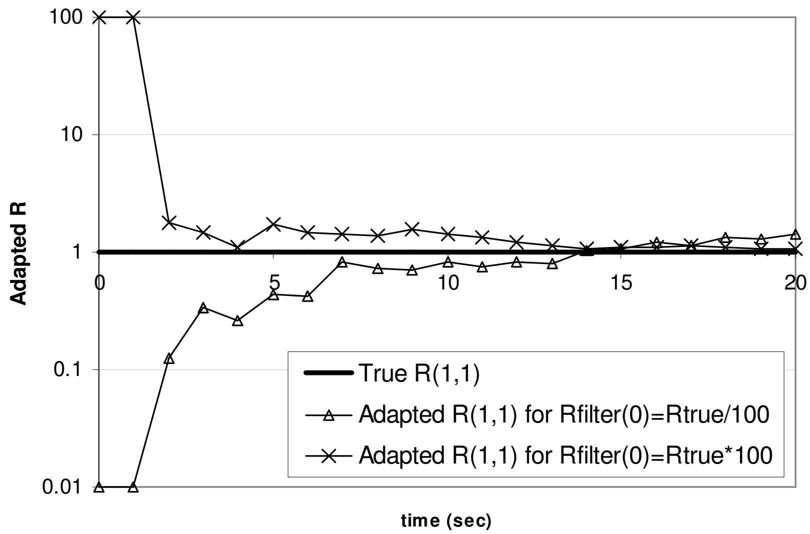


Fig. 8.8: Plot of adapted ***R*** of ACKF (3rd degree) for element ***R***(1,1)

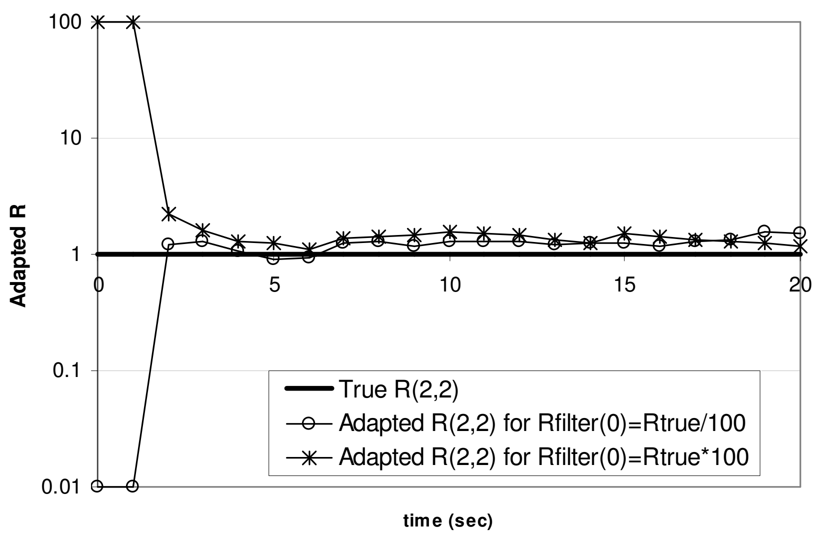


Fig. 8.9: Plot of adapted ***R*** of ACKF (3rd degree) for element ***R***(2,2)

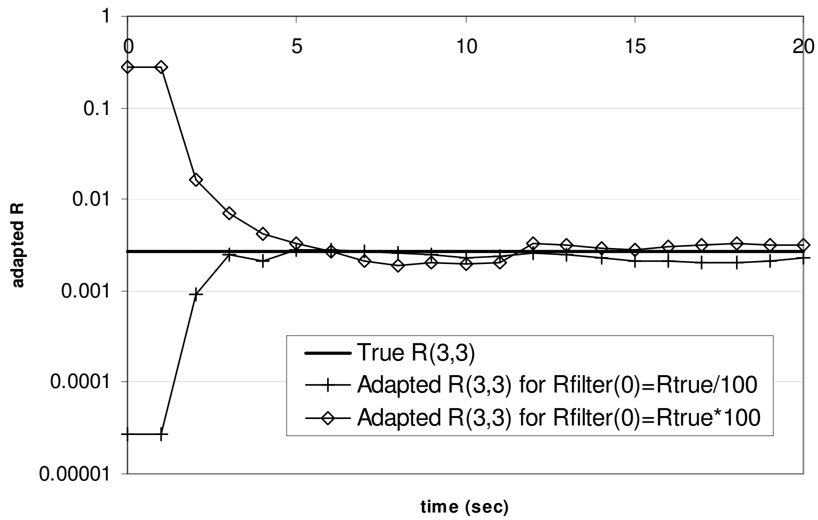


Fig. 8.10: Plot of adapted ***R*** of ACKF (3rd degree) for element ***R***(3,3)

It has been observed from the simulation results that the percentage of track loss is significantly lower in case of ***R*** adaptive Cubature Kalman filter with 3rd degree accuracy (RA-CKF 3rd degree) compared to its non-adaptive version when ***R*** remains unknown. The RMSE (excluding the track loss cases) for the proposed filter converged to lower values within lesser time compared to non-adaptive filter as shown in Fig. 8.6 and Fig. 8.7 for position and velocity estimation respectively. This indicates that even when the track loss cases are excluded from the non-adaptive filter cannot provide satisfactory estimation results for other cases when track loss does not occur. As the track loss cases are excluded the RMSE did not start from the same initial points in the respective figures.

The performance of RA-CKF (3rd) is compared with ***R*** adaptive Cubature Kalman filter with 5th degree accuracy (RA-CKF 5th degree) and ***R*** adaptive DDF. Their RMSE (excluding the track loss cases) are found comparable.

During the analysis of track loss cases it has been observed that the percentage of track loss is 0.36% for RA-CKF (3rd) while that for its non-adaptive version is 2.32%. The percentage of track loss for RA-CKF (5th) is 0.34% which is comparably same with that for RA-CKF (3rd). The track loss percentage for ***R*** adaptive DDF with non-additive noise is found as 0.4% which is slightly higher than RA-CKF (both 3rd 5th degree).

The performance of the RA-CKF (3^{rd}) with unknown noise covariance is also compared with the non-adaptive filter in the ideal situation when the $\boldsymbol{R}$ is known. This comparison is carried out to investigate how close the adaptive filter (with unknown $\boldsymbol{R}$) can perform compared to the non-adaptive filter in ideal situation with knowledge of $\boldsymbol{R}$. It is found that the track loss has not occurred at all for non-adaptive filter in ideal situation with known $\boldsymbol{R}$. Note also that the non-adaptive estimators (UKF and EKF) with known $\boldsymbol{R}$ reported in [Sadhu2006] track loss has occurred as the additive approximation of measurement noise has been made. The track loss cannot be overruled also for the proposed adaptive estimators with non-additive noise. However, susceptibility of track loss is less for the adaptive estimators. So it may be said that proposed adaptive filter without knowledge of $\boldsymbol{R}$ try to perform as good as the non-adaptive filter in the ideal situation.

The plots of adapted $\boldsymbol{R}$ is presented in Fig. 8.8 to Fig. 8.10 where it has been observed that the diagonal elements of adapted $\boldsymbol{R}$ converged to the corresponding truth values staring from the assumed value and continue to track that value for the subsequent times. To investigate the effect of the assumed initial choice of $\boldsymbol{R}$ on the adaptation performance of the proposed RA-CKF (3^{rd}), $\boldsymbol{R}_{filter}(0)$ is chosen deliberately with higher and lower values with large errors such as $\boldsymbol{R}_{filter}(0) = 10^{2} * \boldsymbol{R}_{true}$ and $\boldsymbol{R}_{filter}(0) = 10^{-2} * \boldsymbol{R}_{true}$ respectively. Although the assumed values have a large error the adapted $\boldsymbol{R}$ has converged to the truth value. This indicates that the proposed filters may accommodate a wide range of uncertainty in the initial choice of $\boldsymbol{R}$.

The Bearing only tracking problem has been considered in a publication (mentioned in the list of conference papers with serial number ‘8’, section 1.7.3, chapter 1) of the co worker, Ms. Manasi Das wherein the present worker is a co-author. Adaptive UKF with non-additive measurement noise has been proposed in that work and its performance has been compared with Adaptive DDF for the non-additive measurement noise. More discussions on AUKF for non-additive measurement noise are provided in the referred paper.

8.5.2 Characterization of Q adaptive DDF for non-additive noise

8.5.2.1. State estimation of Van der Pol’s oscillator

The $\boldsymbol{Q}$ adaptive DDF for non-additive process noise is validated with an estimation problem wherein the states and the friction coefficient of Van der Pol’s oscillator are to be estimated. The process noise is assumed to be unknown and also non-additive in nature. Instead of considering the noise term added directly with the state vector, square of the noise terms are considered to be added with the 2nd state ($\boldsymbol{x}(2)$) of the augmented state vector ($\boldsymbol{x}$). This makes the system dynamics a nonlinear function of process noise. This consideration may not be realistic. Because of the paucity of the realistic model of systems with non-additive process noise, such a model is considered as a toy problem for the validation of $\boldsymbol{Q}$ adaptive DDF with non-additive noise. In reality the process noise are often non-additive in nature which are difficult to model mathematically and therefore approximated as additive noise. The process noise covariance is considered as $\boldsymbol{Q}_{true} = diag\left(\begin{bmatrix}10^{-4} & 10^{-6} & 10^{-6}\end{bmatrix}\right)$ to simulate the true state trajectories. The process noise covariance being unknown to the filter it is initialized here with an assumed value for the filter. For this case study $\boldsymbol{Q}_{filter}$ is chosen as $\boldsymbol{Q}_{filter} = 10^5 \times \boldsymbol{Q}_{true}$

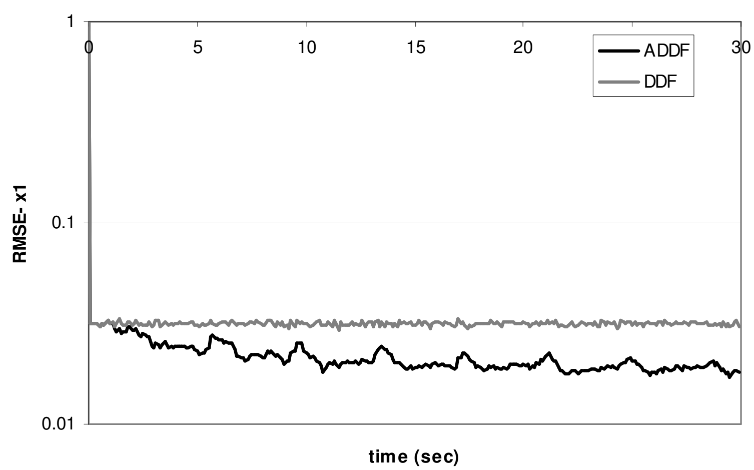


Fig. 8.11: Comparison of RMS error (state, x_1 estimation) of ADDF & DDF for 1000 MC runs

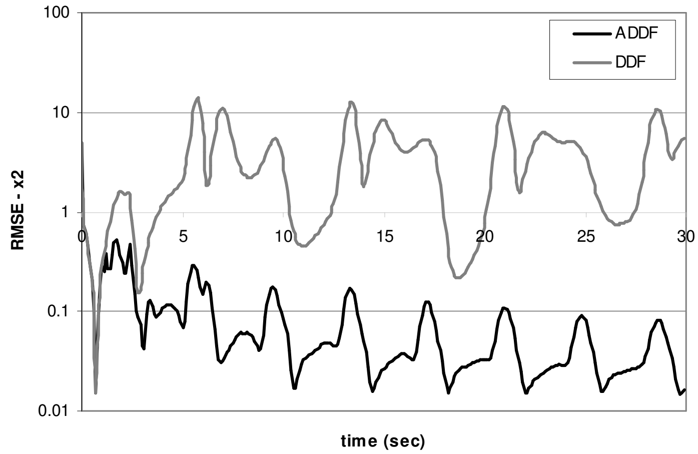


Fig. 8.12: Comparison of RMS error (state, x_2 estimation) of ADDF & DDF for 1000 MC runs

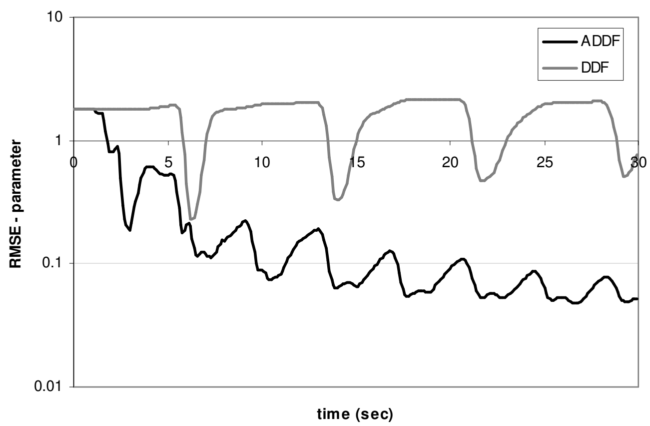


Fig. 8.13: Comparison of RMS error (friction coefficient estimation) of ADDF & DDF for 1000 MC runs

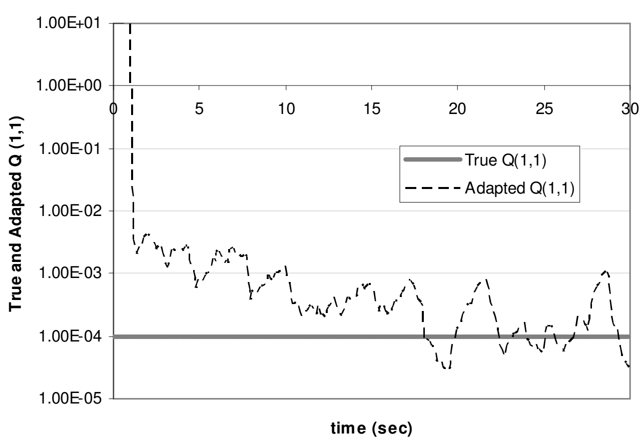


Fig. 8.14: Plot of true and adapted $\boldsymbol{Q}(1,1)$ for a typical run

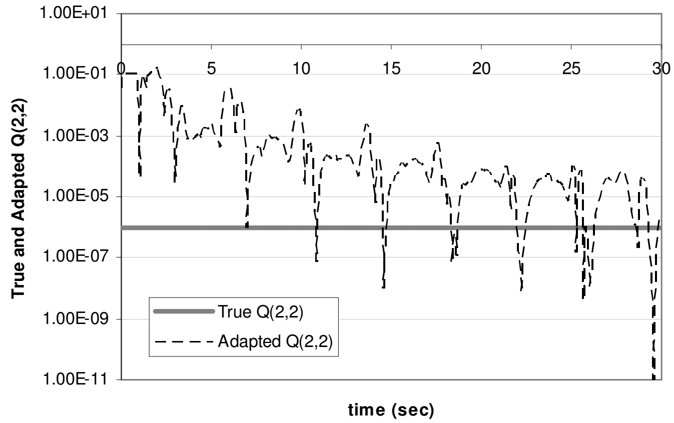


Fig. 8.15: Plot of true and adapted $\boldsymbol{Q}(2,2)$ for a typical run

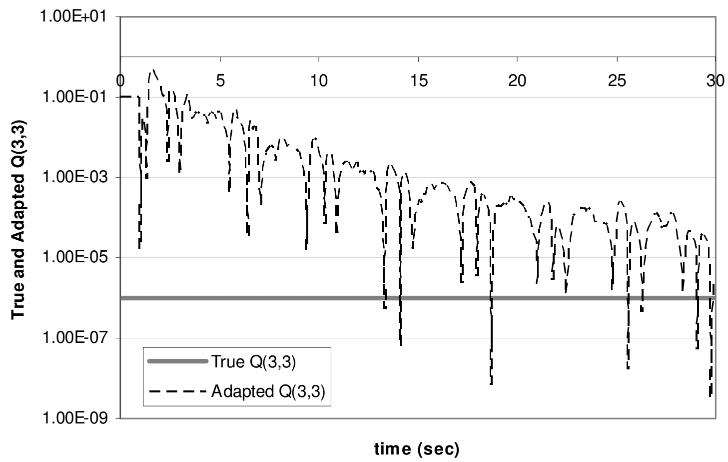


Fig. 8.16: Plot of true and adapted $\boldsymbol{Q}(3,3)$ for a typical run

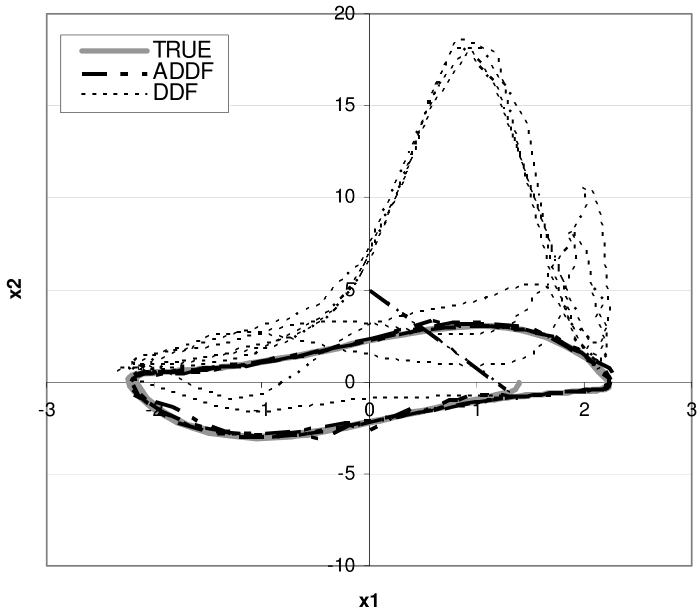


Fig. 8.17: Comparison of phase portrait of the DDF and ADDF estimates with that of true states

From the simulation results it is found that RMS error for both the states and the parameter obtained from Adaptive DDF is much lower than its non-adaptive version. Fig. 8.11 – 8.13 are provided for illustration.

It is also observed that the tracking performance of ***Q*** is less accurate compared to the ***R*** adaptive filters. The elements of adapted ***Q*** tend to converge on the corresponding truth value, however, cannot track truth value satisfactorily. Fig. 8.14 – 8.16 are presented to support the above statement.

In Fig. 8.17 the phase plane plot for a representative run is provided. It is found that the estimate of ADDF although initialized with a point outside the limit cycle gradually converges and retraces the limit cycle of the true state trajectory. The non-adaptive DDF, on contrary, cannot converge on the actual limit cycle and shows a diverging tendency.

8.6 Discussions and Conclusions

In this chapter algorithms for adaptive nonlinear filters have been proposed where the system dynamics or the measurements are nonlinear function of states and noise with unknown noise covariance. The adaptation algorithms for non-additive noise have been derived after substantial modifications of the adaptation algorithm for additive noise. Proposed algorithms are validated in simulation. Significant finding are enumerated below:

- ***R*** adaptive DDF for non-additive measurement noise proposed in this chapter has been validated with the object tracking problem in the presence of non-additive measurement noise. RA-DDF is found to outperform its non-adaptive version when the knowledge of the measurement noise covariance remains unavailable. During steady state the performance of RA-DDF without knowledge of ***R*** is found to be nearly comparable with non-adaptive DDF in the ideal situation where accurate value of ***R*** is known only to the non-adaptive filter.
- Performance of RA-DDF also has been compared with ***R*** Adaptive CKF (both 3rd and 5th degree) which are formulated from the proposed general framework for adaptive nonlinear filter with non-additive noise. The performance comparison is carried out for the bearing only tracking problem where the measurement equation is indeed a nonlinear function of state and measurement noise. It is observed that for this case study that the performance of RA-CKF (both 3rd and 5th degree) is comparable with that RA-DDF on the basis of RMSE and percentage of track loss.
- ***Q*** adaptive DDF for non-additive process noise proposed in this chapter is validated with the estimation problem of Van der Pol's oscillator. QA-DDF also found to outperform its non-adaptive version in the face of unknown process noise covariance.
- It has been observed from the simulation results that the adapted value of unknown measurement noise covariance satisfactorily converges on the truth value and continues to track it for subsequent time for non-additive measurement noise. However, for non-additive process noise accuracy of ***Q*** tracking performance is not satisfactory. Approximations made while deriving adaptation algorithms for ***Q*** may be the reason behind such inaccuracy in the ***Q*** tracking performance. Nevertheless,

QA-DDF is better than non-adaptive DDF while compared on the basis of estimation performance.

From the above observations it may be concluded that these newly designed adaptive nonlinear filters for non-additive noise demonstrate superior estimation performance compared to their non-adaptive counterpart and may be recommended as a suitable candidate for estimation when noises are non-additive in nature and the noise covariance remain unknown.

Chapter 9: Adaptive Nonlinear Information Filters for Multiple Sensor Fusion

9.1 Chapter Introduction

In this chapter a general filtering algorithm for adaptive nonlinear information filters is proposed which is an extension of general framework for adaptive nonlinear estimators presented in chapter 4. The Information filter variant of state estimators is widely recommended for multiple sensor estimation as this particular variant of estimators is computationally economic, supports decentralized sensor fusion, and easy to initialize [Anderson1979, Whyte2008].

Sensor fusion is a conventional process where measurements from multiple sensors are integrated to obtain sufficiently reliable and enriched estimate of the unmeasured states of the system. Formulation of an estimation problem and its solution is one of the central aspects of successful sensor fusion. Publications on non-adaptive nonlinear filters with information filter configuration for nonlinear state estimation are plenty and indicate continued interest in this form of filter (e.g. Unscented information filters [Lee2008], Central Difference information filters [Liu2011], Cubature information filters [Chandra2011]). More discussions have been provided in the literature survey.

However, like other nonlinear filters with standard error covariance form, successful performance of multiple sensor data fusion using information filters presupposes complete knowledge of the covariance of the sensor noise and the system noise. An inaccurately chosen initial value of noise covariance degrades the performance of the filter and the estimates of state get deteriorated even after multiple sensor fusion.

The information filter configuration for adaptive nonlinear filters has not yet received much attention. Formulation of adaptive information filtering techniques for nonlinear signal models is, therefore, an evolving area of knowledge as it is mentioned in a recent review paper on sensor fusion [Khalegi2013]. Only a few works exists in literature where adaptive versions of nonlinear information filters have been reported. Among them, adaptive Cubature Information filters [Tao2014, Ge2014] are noteworthy as discussed in the chapter on literature survey.

In this chapter the general framework of adaptive nonlinear filter in standard error covariance form has been extended with information filter configuration. The same concept for adaptation of process or measurement noise covariance as discussed in chapter 4 has been followed here. The adaptation algorithms are integrated in the information filter algorithms so that the unknown noise covariances can be adapted and the sensor fusion may become successful. With this general framework a number of adaptive nonlinear information filters have been formulated which include adaptive versions of (i) Divided Difference information filter, (ii) Cubature information filter (3^{rd} & 5^{th} degree), (iii) Unscented information filter, (iv) Gauss Hermite information filter, (v) Cubature Quadrature information filter (3^{rd} & 5^{th} degree).

Performances of these newly proposed algorithms have been validated with the help of multi sensor estimation problem in contingent situations where the system (process) noise covariance remains unknown; system dynamics suffers from unknown parameter variations. Alternatively, situations are also considered where the knowledge of the noise covariances of some of the sensors remains unavailable where the sensor characterization has been partially done. During characterization of these estimators, parameter as well as state estimation performance of the proposed estimators has been demonstrated.

The adaptive information filtering algorithms in square root framework are also proposed in this chapter. The advantages of square root approach have already been explained in chapter 4 and are also applicable in case of information filters. With the $\boldsymbol{R}$ adaptation algorithms the square root versions of (i) adaptive Cubature Quadrature information filters (RA-SR-CQIF), (ii) adaptive Cubature information filters (RA-SR-CQF) and (iii) adaptive Gauss Hermite information filters (RA-SR-GHIF) are demonstrated.

9.2 Problem Statement

We consider an augmented nonlinear dynamic system as given below

$$\boldsymbol{x}_k = \boldsymbol{f}(\boldsymbol{x}_{k-1}) + \boldsymbol{w}_k \tag{9.1}$$

$$\boldsymbol{y}_k^{\zeta} = \boldsymbol{g}^{\zeta}(\boldsymbol{x}_k) + \boldsymbol{v}_k^{\zeta} \tag{9.2}$$

Here $\boldsymbol{x}_k \in \Re^{\boldsymbol{n}}$ is an augmented state vector, By the term *augmented state vector*, it is meant that the unknown parameters have been concatenated with the state vector such that dimension of the augmented state vector is n. The difference equations corresponding to unknown parameters $\boldsymbol{\theta}_k$ are considered to obey the random walk model, i.e., $\boldsymbol{\theta}_k = \boldsymbol{\theta}_{k-1} + \boldsymbol{w}_k^{\theta}$, where $\boldsymbol{w}_k^{\theta}$ is the noise term. $\boldsymbol{w}_k \in \Re^n \sim (\boldsymbol{0}, \boldsymbol{Q}_k)$ indicates zero mean process noise (Gaussian white noise). $\boldsymbol{y}_k^{\zeta} \in \Re^m$ is the measurement available from the ζ^{th} sensor among M different sensors where $\zeta = 1, \cdots, M$. The measurement noise of each sensor is also considered to be white (Gaussian) and denoted as, $\boldsymbol{v}_k^{\zeta} \in \Re^m \sim (\boldsymbol{0}, \boldsymbol{R}_k^{\zeta})$.

In the situation when the system dynamics suffers from modeling uncertainties or unknown parameter variation complete knowledge of the process noise covariance often remains unavailable. Some of the elements of $\boldsymbol{Q}_k$ remain unknown and therefore need to be adapted.

In some situations the noise covariances of some of the sensors may remain unavailable where noise characterization has not been carried out. When $\boldsymbol{R}_k^{\zeta}$ of ζ^{th} sensor among the available M sensors remians unknown, adaptation of $\boldsymbol{R}_k^{\zeta}$ becomes necessary for satisfactory estimation.

9.3 Formulation of Adaptive Nonlinear Information Filter

9.3.1 Overview

Information filter for linear signal models is the inverse covariance form of Kalman filter where the information vector and information matrix are propagated instead of state estimate and its error covariance. The information matrix (Also known as Fisher's information matrix) is the inverse of the error covariance matrix. This specific form of estimation algorithm is characterised by information matrix and information vector (termed as canonical parameters in [Liu2012]). With the help of matrix inversion identities [Anderson1979] the information filter variants of Kalman filter and Extended Kalman filter can be readily obtained. However, the information filter form of the sigma point filter cannot be obtained using such matrix inversion identities. The algorithms for information filters using sigma points are reported in literature [Vercauteren2005, Lee2008] where some significant modifications of the algorithm are essential.

Unlike the sigma point filters in standard error covariance form information filter variants require availability of matrix approximation of nonlinear measurement equation which is not readily available with nonlinear signal models other than Extended information filter (EIF). Therefore, [Lee2008] has recommended computation of a pseudo measurement matrix with the help of Statistical Linearization which enables to formulate the information filter variants of nonlinear filters. In the following section algorithm of non-adaptive nonlinear information filters has been presented.

9.3.2 Non-adaptive Nonlinear Information filter

The algorithm for non-adaptive nonlinear information filter has been provided this section. Theorem 4.1 provided in chapter 4 regarding the non-adaptive nonlinear filter in standard error covariance form has been considered again to derive the information filter variant of estimator. The matrix inversion identities from [Anderson1979] are provided below and have been used for obtaining the information filter variant of nonlinear estimators.

Matrix Inversion Identity 9.1:

$$\left(\boldsymbol{A}+\boldsymbol{B}\boldsymbol{D}^{-1}\boldsymbol{C}\right)^{-1}=\boldsymbol{A}^{-1}-\boldsymbol{A}^{-1}\boldsymbol{B}\left(\boldsymbol{D}+\boldsymbol{C}\boldsymbol{A}^{-1}\boldsymbol{B}\right)^{-1}\boldsymbol{C}\boldsymbol{A}^{-1}$$

where $\boldsymbol{A}\in\Re^{m\times m}$, $\boldsymbol{B}\in\Re^{m\times n}$, $\boldsymbol{C}\in\Re^{n\times m}$, $\boldsymbol{D}\in\Re^{n\times n}$

Matrix Inversion Identity 9.2:

$$\left(\boldsymbol{A}+\boldsymbol{B}\boldsymbol{D}^{-1}\boldsymbol{C}\right)^{-1}\boldsymbol{B}\boldsymbol{D}^{-1}=\boldsymbol{A}^{-1}\boldsymbol{B}\left(\boldsymbol{D}+\boldsymbol{C}\boldsymbol{A}^{-1}\boldsymbol{B}\right)^{-1}$$

where $\boldsymbol{A}\in\Re^{m\times m}$, $\boldsymbol{B}\in\Re^{m\times n}$, $\boldsymbol{C}\in\Re^{n\times m}$, $\boldsymbol{D}\in\Re^{n\times n}$

From Theorem 4.1 given in chapter 4 we have:

$$\hat{\boldsymbol{x}}_k=\overline{\boldsymbol{x}}_k+\boldsymbol{K}_k\left(\boldsymbol{y}_k-\overline{\boldsymbol{y}}_k\right) \tag{9.3}$$

$$\hat{\boldsymbol{P}}_k=\overline{\boldsymbol{P}}_k-\boldsymbol{K}_k\boldsymbol{P}_k^{y}\boldsymbol{K}_k^{T} \tag{9.4}$$

where

$$\overline{\boldsymbol{x}}_k=\int_{\boldsymbol{R}^n}\boldsymbol{f}\left(\boldsymbol{x}_{k-1}\right)p\left(\boldsymbol{x}_{k-1}\middle|\boldsymbol{Y}_{k-1}\right)d\boldsymbol{x}_{k-1} \tag{9.5}$$

$$\overline{\boldsymbol{P}}_k=\overline{\boldsymbol{Q}}+\int_{\boldsymbol{R}^n}\left(\boldsymbol{f}\left(\boldsymbol{x}_{k-1}\right)-\overline{\boldsymbol{x}}_k\right)\left(\boldsymbol{f}\left(\boldsymbol{x}_{k-1}\right)-\overline{\boldsymbol{x}}_k\right)^{T}p\left(\boldsymbol{x}_{k-1}\middle|\boldsymbol{Y}_{k-1}\right)d\boldsymbol{x}_{k-1} \tag{9.6}$$

$$\overline{\boldsymbol{y}}_k=\int_{\boldsymbol{R}^n}\boldsymbol{g}\left(\boldsymbol{x}_k\right)p\left(\boldsymbol{x}_k\middle|\boldsymbol{Y}_{k-1}\right)d\boldsymbol{x}_k \tag{9.7}$$

$$P_k^{xy} = \int_{R^n} (f(x_k) - \bar{x}_k)(g(x_k) - \bar{y}_k)^T p(x_k | Y_{k-1}) dx_k \tag{9.8}$$

$$P_k^{y} = \bar{R} + \int_{R^n} (g(x_k) - \bar{y}_k)(g(x_k) - \bar{y}_k)^T p(x_k | Y_{k-1}) dx_k \tag{9.9}$$

$$K_k = P_k^{xy} \left(P_k^{y}\right)^{-1} \tag{9.10}$$

Here, we consider the pseudo measurement matrix of the nonlinear measurement equation as defined in [Lee2008, Liu2011, Chandra2011] following the method of Statistical Linearization. The pseudo measurement matrix Ψ_k is defined as

$$\Psi_k = \left(\bar{P}_k^{-1} P_k^{xy}\right)^T \tag{9.11}$$

Using Ψ_k , P_k^{xy} and P_k^{y} can be expressed as:

$$P_k^{xy} = \bar{P}_k \Psi_k^T \tag{9.12}$$

$$P_k^{y} = \bar{R} + \Psi_k \bar{P}_k \Psi_k^T \tag{9.13}$$

Therefore, the expression of filter gain becomes similar to that of Kalman filter, i.e.,

$$K_k = P_k^{xy} \left(P_k^{y}\right)^{-1} \tag{9.14}$$

$$K_k = \bar{P}_k \Psi_k^T \left(\bar{R} + \Psi_k \bar{P}_k \Psi_k^T\right)^{-1} \tag{9.15}$$

We have

The *a posteriori* error covariance , $\hat{P}_k = \bar{P}_k - K_k P_k^{y} K_k^T$, which can alternatively presented as

$$\hat{P}_k = \bar{P}_k - P_k^{xy} \left(P_k^{y}\right)^{-1} \left(P_k^{xy}\right)^T \tag{9.16}$$

It can also be expressed in terms of Ψ_k as

$$\hat{P}_k = \bar{P}_k - \bar{P}_k \Psi_k^T \left(\bar{R} + \Psi_k \bar{P}_k \Psi_k^T\right)^{-1} \Psi_k \bar{P}_k \tag{9.17}$$

The *a posteriori* (updated) information matrix corresponding to $\hat{P}_k$ is given by

$$\hat{Z}_k = \hat{P}_k^{-1} \tag{9.18}$$

$$\Rightarrow \hat{Z}_k = \left(\bar{P}_k - \bar{P}_k \Psi_k^T \left(\bar{R} + \Psi_k \bar{P}_k \Psi_k^T\right)^{-1} \Psi_k \bar{P}_k \right)^{-1} \tag{9.19}$$

Using matrix inversion identity 9.1, the expression $\overline{\boldsymbol{P}}_k - \overline{\boldsymbol{P}}_k \boldsymbol{\Psi}_k^T \left(\overline{\boldsymbol{R}} + \boldsymbol{\Psi}_k \overline{\boldsymbol{P}}_k \boldsymbol{\Psi}_k^T\right)^{-1} \boldsymbol{\Psi}_k \overline{\boldsymbol{P}}_k$ can be written as $\left(\overline{\boldsymbol{P}}_k^{-1} + \boldsymbol{\Psi}_k^T \overline{\boldsymbol{R}}^{-1} \boldsymbol{\Psi}_k\right)^{-1}$ and $\hat{\boldsymbol{Z}}_k$ becomes

$$\hat{\boldsymbol{Z}}_k = \left(\overline{\boldsymbol{Z}}_k + \boldsymbol{\Psi}_k^T \overline{\boldsymbol{R}}^{-1} \boldsymbol{\Psi}_k\right) \tag{9.20}$$

Where $\overline{\boldsymbol{Z}}_k$ indicates *a priori* information matrix,

$$\overline{\boldsymbol{Z}}_k = \overline{\boldsymbol{P}}_k^{-1} \tag{9.21}$$

The *a posteriori* information vector is derived as presented below:

$$\hat{\boldsymbol{z}}_k = \hat{\boldsymbol{P}}_k^{-1} \hat{\boldsymbol{x}}_k$$

$$\Rightarrow \hat{\boldsymbol{z}}_k = \left(\overline{\boldsymbol{P}}_k^{-1} + \boldsymbol{\Psi}_k^T \overline{\boldsymbol{R}}^{-1} \boldsymbol{\Psi}_k\right)\left(\overline{\boldsymbol{x}}_k + \boldsymbol{K}_k\left(\boldsymbol{y}_k - \overline{\boldsymbol{y}}_k\right)\right)$$

$$\Rightarrow \hat{\boldsymbol{z}}_k = \left(\overline{\boldsymbol{P}}_k^{-1} + \boldsymbol{\Psi}_k^T \overline{\boldsymbol{R}}^{-1} \boldsymbol{\Psi}_k\right)\overline{\boldsymbol{x}}_k + \left(\overline{\boldsymbol{P}}_k^{-1} + \boldsymbol{\Psi}_k^T \overline{\boldsymbol{R}}^{-1} \boldsymbol{\Psi}_k\right)\boldsymbol{K}_k\left(\boldsymbol{y}_k - \overline{\boldsymbol{y}}_k\right)$$

$$\Rightarrow \hat{\boldsymbol{z}}_k = \overline{\boldsymbol{P}}_k^{-1}\overline{\boldsymbol{x}}_k + \boldsymbol{\Psi}_k^T \overline{\boldsymbol{R}}^{-1} \boldsymbol{\Psi}_k \overline{\boldsymbol{x}}_k + \left(\overline{\boldsymbol{P}}_k^{-1} + \boldsymbol{\Psi}_k^T \overline{\boldsymbol{R}}^{-1} \boldsymbol{\Psi}_k\right)\overline{\boldsymbol{P}}_k \boldsymbol{\Psi}_k^T \left(\overline{\boldsymbol{R}} + \boldsymbol{\Psi}_k \overline{\boldsymbol{P}}_k \boldsymbol{\Psi}_k^T\right)^{-1}\left(\boldsymbol{y}_k - \overline{\boldsymbol{y}}_k\right)$$

$$\Rightarrow \hat{\boldsymbol{z}}_k = \overline{\boldsymbol{P}}_k^{-1}\overline{\boldsymbol{x}}_k + \boldsymbol{\Psi}_k^T \overline{\boldsymbol{R}}^{-1} \boldsymbol{\Psi}_k \overline{\boldsymbol{x}}_k + \left(\overline{\boldsymbol{P}}_k^{-1} + \boldsymbol{\Psi}_k^T \overline{\boldsymbol{R}}^{-1} \boldsymbol{\Psi}_k\right)\left(\overline{\boldsymbol{P}}_k^{-1} + \boldsymbol{\Psi}_k^T \overline{\boldsymbol{R}}^{-1} \boldsymbol{\Psi}_k\right)^{-1} \boldsymbol{\Psi}_k^T \overline{\boldsymbol{R}}^{-1}\left(\boldsymbol{y}_k - \overline{\boldsymbol{y}}_k\right)$$

$$\Rightarrow \hat{\boldsymbol{z}}_k = \overline{\boldsymbol{z}}_k + \boldsymbol{\Psi}_k^T \overline{\boldsymbol{R}}^{-1} \boldsymbol{\Psi}_k \overline{\boldsymbol{x}}_k + \boldsymbol{\Psi}_k^T \overline{\boldsymbol{R}}^{-1} \boldsymbol{y}_k - \boldsymbol{\Psi}_k^T \overline{\boldsymbol{R}}^{-1} \overline{\boldsymbol{y}}_k$$

Therefore, *a posteriori* information vector is obtained as

$$\hat{\boldsymbol{z}}_k = \overline{\boldsymbol{z}}_k + \boldsymbol{\Psi}_k^T \overline{\boldsymbol{R}}^{-1}\left(\boldsymbol{y}_k - \overline{\boldsymbol{y}}_k + \boldsymbol{\Psi}_k \overline{\boldsymbol{x}}_k\right) \tag{9.22}$$

where $\overline{\boldsymbol{z}}_k$, the *a priori* information vector is defined as $\overline{\boldsymbol{z}}_k = \overline{\boldsymbol{Z}}_k \overline{\boldsymbol{x}}_k$ (9.23)

The algorithm of non-adaptive information filter is presented (9.5) from (9.23) considering $\boldsymbol{g}^{\zeta}(\cdot) = \boldsymbol{g}\ (\cdot)$

9.3.3 Adaptation algorithms

The methods of adaptation for process noise covariance and measurement noise covariance have been presented in detail in chapter 4. The same adaptation algorithms can be reproduced with the following considerations.

For the derivation of adaptation algorithms all the available measurements are to be augmented to get an augmented measurement vector as $\boldsymbol{y}_k = \left[\boldsymbol{y}_k^1 \quad \boldsymbol{y}_k^2 \quad \dots \quad \boldsymbol{y}_k^M\right]^T$ with order *mM*. Therefore, the corresponding measurement noise covariance becomes $\boldsymbol{R}_k = diag\left(\boldsymbol{R}_k^1, \boldsymbol{R}_k^2, \dots, \boldsymbol{R}_k^M\right)$ and the augmented pseudo measurement matrix would appear as

$\boldsymbol{\Psi}_k = diag(\boldsymbol{\Psi}_k^1, \boldsymbol{\Psi}_k^2, \ldots, \boldsymbol{\Psi}_k^M)$. Now, employing the innovation or residual sequence the following adaptation steps can be derived in the same way as presented in chapter 4. In this chapter we present only the adaptation algorithms instead of re-deriving them to avoid repetition.

9.3.3.1. Adaptation of Process Noise Covariance

The process noise covariance can be adapted using the window estimate of the state residual where the state residual is defined as $\boldsymbol{\eta}_k = \hat{\boldsymbol{x}}_k - \bar{\boldsymbol{x}}_k$ or, $\boldsymbol{\eta}_k = \hat{\boldsymbol{Z}}_k^{-1}\hat{\boldsymbol{z}}_k - \bar{\boldsymbol{Z}}_k^{-1}\bar{\boldsymbol{z}}_k$

The adapted process noise covariance, after some approximation, becomes

$$\hat{\boldsymbol{Q}}_k = \frac{1}{L}\sum_{j=k-L+1}^{k}\left[\boldsymbol{\eta}_k\boldsymbol{\eta}_k^T\right] \tag{9.24}$$

9.3.3.2. Adaptation of Measurement Noise Covariance

The measurement noise covariance as discussed in chapter 4 can be adapted incorporating innovation sequence as well as the residual sequence. However, the latter is preferred because of its additional advantage of ensured positive definiteness. The innovation or, the residual sequence for ζ^{th} measurement is used for adaptation of ζ^{th} measurement noise covariance.

The adaptation step for measurement noise covariance using innovation sequence can be presented as

$$\hat{\boldsymbol{R}}_k^\zeta = \frac{1}{L}\sum_{j=k-L+1}^{k}\vartheta_j^\zeta\left(\vartheta_j^\zeta\right)^T - \boldsymbol{\psi}_k\bar{\boldsymbol{P}}_k\boldsymbol{\psi}_k^T \tag{9.25}$$

Where the innovation is defined as $\vartheta_k^\zeta = \boldsymbol{y}_k^\zeta - \bar{\boldsymbol{y}}_k^\zeta$

Alternatively, for residual based $\boldsymbol{R}$ adaptation the expression of the adapted measurement noise covariance can be presented as

$$\hat{\boldsymbol{R}}_k^\zeta = \frac{1}{L}\sum_{j=k-L+1}^{k}\boldsymbol{\rho}_j^\zeta\left(\boldsymbol{\rho}_j^\zeta\right)^T + \boldsymbol{\psi}_k\hat{\boldsymbol{P}}_k\boldsymbol{\psi}_k^T \tag{9.26}$$

where the residual is defined as $\boldsymbol{\rho}_k^\zeta = \boldsymbol{y}_k^\zeta - \hat{\boldsymbol{y}}_k^\zeta$

9.4 General Framework

The algorithm for adaptive nonlinear information filters is presented below in a general framework so that different methods of sigma point and weight selection can be applied and subsequently the respective adaptive information filtering algorithm can be formulated. Following the numerical methods sigma points are generated which are represented as $\boldsymbol{q_i}$ and w_i in the algorithm where N denotes the number of points.

9.4.1 Algorithmic steps

GENERAL FRAMEWORK FOR ADAPTIVE NONLINEAR INFORMATION FILTER

(i) Initialization: Initialize $\hat{\boldsymbol{x}}_0, \hat{\boldsymbol{P}}_0, \overline{\boldsymbol{Q}}, \overline{\boldsymbol{R}}_k^{\zeta}$

(ii) Time update step (propagation):

Compute Cholesky Factor such that $\hat{\boldsymbol{P}}_{k-1} = \hat{\boldsymbol{S}}_{k-1}\left(\hat{\boldsymbol{S}}_{k-1}\right)^T$ (9.27)

The points selected for propagation of mean and covariance are given below as

$$\hat{\boldsymbol{\chi}}_i = \hat{\boldsymbol{S}}_{k-1}\boldsymbol{q}_i + \hat{\boldsymbol{x}}_{k-1} \tag{9.28}$$

Compute *a priori* estimate of state as

$$\overline{\boldsymbol{x}}_k = \sum_{i=1}^{N} \boldsymbol{f}(\hat{\boldsymbol{\chi}}_i) w_i \tag{9.29}$$

and respective *a priori* error covariance is obtained as

$$\overline{\boldsymbol{P}}_k = \overline{\boldsymbol{Q}} + \sum_{i=1}^{N} \left(\boldsymbol{f}(\hat{\boldsymbol{\chi}}_i) - \overline{\boldsymbol{x}}_k\right)\left(\boldsymbol{f}(\hat{\boldsymbol{\chi}}_i) - \overline{\boldsymbol{x}}_k\right)^T w_i \tag{9.30}$$

The *a priori* information matrix is obtained as

$$\overline{\boldsymbol{Z}}_k = \overline{\boldsymbol{P}}_k^{-1} \tag{9.31}$$

The *a priori* information vector becomes $\overline{\boldsymbol{z}}_k = \overline{\boldsymbol{Z}}_k \overline{\boldsymbol{x}}_k$ (9.32)

(iii) Measurement update step:

Compute the Cholesky Factor such that $\overline{\boldsymbol{P}}_k = \overline{\boldsymbol{S}}_k\left(\overline{\boldsymbol{S}}_k\right)^T$ (9.33)

Select sigma points as $\overline{\boldsymbol{\chi}}_i = \overline{\boldsymbol{S}}_k \boldsymbol{q}_i + \overline{\boldsymbol{x}}_k$ (9.34)

The *a priori* estimate of measurement becomes

$$\overline{\boldsymbol{y}}_k^{\zeta} = \sum_{i=1}^{N} \boldsymbol{g}^{\zeta}(\overline{\boldsymbol{\chi}}_i) w_i \tag{9.35}$$

The cross covariance can be computed as

$$\boldsymbol{P}_k^{xy} = \sum_{i=1}^{N} \left(\overline{\boldsymbol{\chi}}_i - \overline{\boldsymbol{x}}_k\right)\left(\boldsymbol{g}^{\zeta}(\overline{\boldsymbol{\chi}}_i) - \overline{\boldsymbol{y}}_k^{\zeta}\right)^T w_i \tag{9.36}$$

The pseudo measurement matrix becomes

$$\boldsymbol{\Psi}_k^{\zeta} = \left(\overline{\boldsymbol{P}}_k\right)^{-1} \boldsymbol{P}_k^{xy} \tag{9.37}$$

The *a posteriori* of information matrix is

$$\hat{\boldsymbol{Z}}_k = \hat{\boldsymbol{P}}_k^{-1} \tag{9.38}$$

Where $\hat{\boldsymbol{P}}_k^{-1} = \overline{\boldsymbol{P}}_k^{-1} + \left(\boldsymbol{\Psi}_k^{\zeta}\right)^T \left(\overline{\boldsymbol{R}}_k^{\zeta}\right)^{-1} \boldsymbol{\Psi}_k^{\zeta}$ (9.39)

The *a posteriori* estimate of information vector is obtained as

$$\hat{\boldsymbol{z}}_k = \overline{\boldsymbol{z}}_k + \left(\boldsymbol{\Psi}_k^{\zeta}\right)^T \left(\overline{\boldsymbol{R}}_k^{\zeta}\right)^{-1} \left(\boldsymbol{y}_k^{\zeta} - \overline{\boldsymbol{y}}_k^{\zeta} + \boldsymbol{\Psi}_k^{\zeta} \overline{\boldsymbol{x}}_k\right) \tag{9.40}$$

Therefore, the *a posteriori* estimate of state becomes

$$\hat{\boldsymbol{x}}_k = \hat{\boldsymbol{Z}}_k^{-1} \hat{\boldsymbol{z}}_k \tag{9.41}$$

(iv) Q-Adaptation Steps:

When $\boldsymbol{Q}$ is unknown, on contrary, $\boldsymbol{R}$ is known, i.e., $\overline{\boldsymbol{R}}_k^{\zeta} = \boldsymbol{R}_k^{\zeta}$ the steps for $\boldsymbol{Q}$ adaptation have to be executed after replacing $\overline{\boldsymbol{Q}} = \hat{\boldsymbol{Q}}_{k-1}$ in (9.30) of the time update steps

Compute the state residual sequence as

$$\boldsymbol{\eta}_k = \hat{\boldsymbol{x}}_k - \overline{\boldsymbol{x}}_k \tag{9.42}$$

The estimated residual covariance can be computed from a sliding window of length L

$$\hat{\boldsymbol{C}}_k^{\eta} = \frac{1}{L} \sum_{j=k-L+1}^{k} \boldsymbol{\eta}_k(j) \boldsymbol{\eta}_k^T(j) \tag{9.43}$$

The adaptation step for $\hat{\boldsymbol{Q}}_k$ is given by

$$\hat{\boldsymbol{Q}}_k = \hat{\boldsymbol{C}}_k^{\eta} \tag{9.44}$$

(v) R-Adaptation Steps:

When noise covariance of ζ^{th} sensor is unknown, on contrary, $\boldsymbol{Q}$ is known, i.e., $\overline{\boldsymbol{Q}} = \boldsymbol{Q}$ the steps for $\boldsymbol{R}$ adaptation are to be executed replacing $\overline{\boldsymbol{R}}_k^{\zeta} = \hat{\boldsymbol{R}}_{k-1}^{\zeta}$ in (9.39) and (9.40)

Innovation based R adaptation:

Compute the innovation sequence as

$$\boldsymbol{\vartheta}_k^{\zeta} = \boldsymbol{y}_k^{\zeta} - \overline{\boldsymbol{y}}_k^{\zeta} \tag{9.45}$$

The estimated residual covariance can be computed from a sliding window of length L

$$\hat{\boldsymbol{C}}_{\vartheta_k} = \frac{1}{L} \sum_{j=k-L+1}^{k} \boldsymbol{\vartheta}_k^{\zeta}(\boldsymbol{j}) \left(\boldsymbol{\vartheta}_k^{\zeta}(\boldsymbol{j}) \right)^T \tag{9.46}$$

The expression of adapted $\boldsymbol{R}$ is given by

$$\hat{\boldsymbol{R}}_k^{\zeta} = \hat{\boldsymbol{C}}_{\vartheta_k} - \overline{\boldsymbol{P}}_k^{g} \tag{9.47}$$

where

$$\overline{\boldsymbol{P}}_k^{g} = \sum_{i=1}^{N} \left(\boldsymbol{g}^{\zeta}(\overline{\boldsymbol{\chi}}_i) - \overline{\boldsymbol{y}}_k^{\zeta} \right) \left(\boldsymbol{g}^{\zeta}(\overline{\boldsymbol{\chi}}_i) - \overline{\boldsymbol{y}}_k^{\zeta} \right)^T w_i \tag{9.48}$$

Residual based R adaptation:

Compute the residual sequence as

$$\boldsymbol{\rho}_k^{\zeta} = \boldsymbol{y}_k^{\zeta} - \hat{\boldsymbol{y}}_k^{\zeta} \tag{9.49}$$

where

$$\hat{\boldsymbol{y}}_k^{\zeta} = \sum_{i=1}^{N} \boldsymbol{g}^{\zeta}(\hat{\boldsymbol{\chi}}_i) w_i \tag{9.50}$$

The estimated residual covariance can be computed from a sliding window of length L

$$\hat{\boldsymbol{C}}_k^{\rho} = \frac{1}{L} \sum_{j=k-L+1}^{k} \boldsymbol{\rho}_k^{\zeta}(j) \left(\boldsymbol{\rho}_k^{\zeta}(j) \right)^T \tag{9.51}$$

The expression of adapted $\boldsymbol{R}$

$$\hat{\boldsymbol{R}}_k^{\zeta} = \hat{\boldsymbol{C}}_k^{\rho} + \hat{\boldsymbol{P}}_k^{g} \tag{9.52}$$

$$\hat{\boldsymbol{P}}_k^{g} = \sum_{i=1}^{N} \left(\boldsymbol{g}^{\zeta}(\hat{\boldsymbol{\chi}}_i) - \hat{\boldsymbol{y}}_k^{\zeta} \right) \left(\boldsymbol{g}^{\zeta}(\hat{\boldsymbol{\chi}}_i) - \hat{\boldsymbol{y}}_k^{\zeta} \right)^T w_i \tag{9.54}$$

(vi) Recursion: The time update and measurement update steps are repeated for obtaining the estimates for the subsequent time steps starting from k=1.

9.4.2 Multiple Sensor Fusion

Estimation using multiple measurements is preferred for obtaining a reliable estimate using the algorithm of adaptive information filter. Information filters supports decentralized approach of multiple sensor estimation which does not require central processing unit and increases the reliability of sensor fusion even in face of sensor failures.

The updated information vector and information matrix after fusion is obtained as a linear combination of the local information contribution terms as given by:

$$\hat{z}_k = \bar{z}_k + \sum_{\zeta=1}^{M} \varphi_k^{\zeta} \tag{9.55}$$

$$\hat{Z}_k = \bar{Z}_k + \sum_{\zeta=1}^{M} \Phi_k^{\zeta} \tag{9.56}$$

Here, for ζ^{th} sensor, the local contribution for information vector and information matrix are obtained from the above algorithm as

$$\varphi_k^{\zeta} = \left(\Psi_k^{\zeta}\right)^T \left(\bar{R}_k^{\zeta}\right)^{-1} \left(\vartheta_k^{\zeta} + \Psi_k^{\zeta} \bar{x}_k\right) \tag{9.57}$$

$$\Phi_k^{\zeta} = \left(\Psi_k^{\zeta}\right)^T \left(\bar{R}_k^{\zeta}\right)^{-1} \Psi_k^{\zeta} \tag{9.58}$$

Finally the updated estimates of systems state and error covariance matrix after multi sensor data fusion are obtained as

$$\hat{x}_k = \hat{Z}_k^{-1} \hat{z}_k \tag{9.59}$$

where,

$$\hat{P}_k = \hat{Z}_k^{-1} \tag{9.60}$$

9.4.3 Choice of Sigma Points and Weights

In this section the possible set of sigma points and weights are presented which can be selected based on numerical methods. These sigma points and weights can be applied in the general algorithm to formulate variants of adaptive sigma point information filters. Depending on the choice of sigma points and weights, the general algorithm presents adaptive versions of (i) Unscented Information filter (ii) Divided Difference Information filter (iii) Gauss Hermite Information filters and (iv) Cubature Information filter (3^{rd} degree and 5^{th} degree) (v) Cubature Quadrature Information filter (3^{rd} degree and 5^{th} degree). Adaptive sigma point filters developed in standard error covariance have been presented in the previous chapters where the numerical methods for selection of sigma points have been discussed in detail. Here we present only the steps for selection of the sigma point using different approaches.

9.4.3.1. Unscented Transformation Rule

Following the unscented transformation rule [Julier2004], $2n+1$ number of sigma points and the corresponding weights can be generated as given below. We select the points for a Gaussian distribution with zero mean and unity covariance. These points will undergo a scale change and origin shift to suit the steps of general algorithm.

TABLE-9.1: SELECTION OF SIGMA POINTS & WEIGHTS
Steps for generation of sigma points:
For i=0, $\boldsymbol{q_0} = \boldsymbol{0}$; $\boldsymbol{q_i} = \boldsymbol{e_i}\sqrt{n+\lambda}$ for $i = 1,\cdots,n$ $\boldsymbol{q_{n+i}} = -\boldsymbol{e_i}\sqrt{n+\lambda}$ for $i = 1,\cdots,n$ Here, $\boldsymbol{e_i}$ is the i^{th} unit vector, Where (i) $\lambda = (\alpha^2 - 1)n$, $\alpha = 0.6$, $\beta = 2$ [Merwe2003]or (ii) $\lambda = 3 - n$ [Julier2000]
Steps for weight selection:
The weights corresponding to the sigma points, $\boldsymbol{q_i}$ are $w_0^m = \frac{\lambda}{n+\lambda}$; $w_0^c = \frac{\lambda}{n+\lambda} + (1-\alpha^2+\beta)$ and $w_i^m = w_i^c = \frac{1}{2(n+\lambda)}$ for, i $\neq$ 0. Where (i) $\lambda = (\alpha^2 - 1)n$, $\alpha = 0.6$, $\beta = 2$ [Merwe2003]or (ii) $\lambda = 3 - n$ [Julier2000]

9.4.3.2. Gauss Hermite Quadrature Rule

Gauss Hermite quadrature rule as in [Ito2000] is presented in Table 9.2.

TABLE-9.2: SELECTION OF QUADRATURE POINTS & WEIGHTS
Steps for generation of quadrature Points:
Compute $\boldsymbol{J}$, a symmetric tri-diagonal, defined as $\boldsymbol{J}_{i,i}=0$ and $\boldsymbol{J}_{i,i+1}=\sqrt{\frac{i}{2}}$ for $1\le i\le N-1$ for N -quadrature points. The quadrature points are chosen as $q_i=\sqrt{2}x_i$ where x_i are the eigen values of $\boldsymbol{J}$ matrix.
Steps for weight selection:
The corresponding weights (w_i) of q_i is computed as $\lvert(v_i)_1\rvert^2$ where $(v_i)_1$ is the first element of the i^{th} normalized eigenvector of $\boldsymbol{J}$ For multi dimensional system the quadrature points and weights are obtained with the help of direct tensor product rule as mentioned in chapter 6.

9.4.3.3. Cubature Rule

The third degree cubature rule as in [Arasaratnam2009] is presented in Table 9.3.1.

TABLE-9.3.1: SELECTION OF CUBATURE POINTS & WEIGHTS
Steps for generation of cubature points:
2n number of cubature points have been selected as $\boldsymbol{q_i}=\boldsymbol{e_i}\sqrt{n}$ for $i=1,\cdots,n$ $\boldsymbol{q_{n+i}}=-\boldsymbol{e_i}\sqrt{n}$ for $i=1,\cdots,n$ Here, $\boldsymbol{e_i}$ is the i^{th} unit vector
Steps for weight selection:
The weights of the corresponding cubature points are considered as $w_i=\frac{1}{2n}$ for, $i\neq 0$.

The above points are selected based on cubature rule with third degree accuracy.

We can also get the points from the fifth degree cubature rule following [Jia2013] as given in Table 9.3.2.

TABLE 9.3.2: SELECTION OF CUBATURE POINTS & WEIGHTS

Cubature Points (q_i)	Weights (w_i)
$\boldsymbol{0}$	$w_0 = \frac{2}{2+n}$ for $n=0$
$\boldsymbol{e}_i\sqrt{n+2}$	For each case $w_i = \frac{4-n}{2(2+n)^2}$ for $i=1, \dots, n$
$-\boldsymbol{e}_i\sqrt{n+2}$	
$\boldsymbol{s}_i^+\sqrt{n+2}$	For each case $w_i = \frac{1}{(2+n)^2}$ $i = 1, \cdots, \frac{n(n-1)}{2}$
$-\boldsymbol{s}_i^+\sqrt{n+2}$	
$\boldsymbol{s}_i^-\sqrt{n+2}$	
$-\boldsymbol{s}_i^-\sqrt{n+2}$	

$\boldsymbol{s}_i^+$ and $\boldsymbol{s}_i^-$ are generated as

$$\boldsymbol{s}_i^+ = \sqrt{\tfrac{1}{2}}\left(\boldsymbol{e}_i + \boldsymbol{e}_l\right):\ j < l,\ j, l = 1, \cdots, n$$

$$\boldsymbol{s}_i^- = \sqrt{\tfrac{1}{2}}\left(\boldsymbol{e}_i - \boldsymbol{e}_l\right):\ j < l,\ j, l = 1, \cdots, n$$

9.4.3.4. Cubature Quadrature Rule

Cubature Quadrature rule (third degree accuracy) as in [Bhaumik2013] can be employed to obtain points and weights as given in Table 9.4.1.

TABLE-9.4.1: SELECTION OF CUBATURE QUADRATURE POINTS & WEIGHTS

Steps for generation of cubature points:

$2nn'$ number of cubature quadrature points are to be selected as

Where $\boldsymbol{\xi_i} = \sqrt{2\lambda_j}\,\boldsymbol{e_i}$

The cubature points located at the intersection of the unit hyper-sphere and its axes.

For 3rd degree approximation rule $\boldsymbol{e_k}$ can be obtained as:

$\boldsymbol{e}_i = \boldsymbol{e}_k$ for $k=1, \ldots, n$ and $\boldsymbol{e}_i = -\boldsymbol{e}_k$ for $k=n+1, \ldots, 2n$ where $\boldsymbol{e}_k$ is the k^{th} unit vector.

λ_j is the solution of n'^{th} order Chebyshev-Laguerre polynomial with $\alpha = n/2 - 1$:

$$L_{n'}^{\alpha} = \lambda^{n'} - \tfrac{n'}{1!}(n'+\alpha)\lambda^{n'-1} + \tfrac{n'(n'-1)}{2!}(n'+\alpha)(n'+\alpha-1)\lambda^{n'-2} - \cdots = 0$$

Here, $i = 1, 2, \ldots, 2nn'$, $j = 1, 2, \ldots, n'$ and $k = 1, 2, \ldots, 2n$

Steps for weight selection:

The corresponding weights are obtained as

$$w_i = \frac{1}{2n\Gamma(n/2)} \frac{n'!\Gamma(\alpha + n' + 1)}{\lambda_j \left[\dot{L}_{n'}^{\alpha}(\lambda_j)\right]^2}$$

Cubature Quadrature rule (with fifth degree accuracy) as in [Singh2015] can be employed to obtain points and weights as given in Table 9.4.2.

TABLE-9.4.2: SELECTION OF CUBATURE QUADRATURE POINTS & WEIGHTS

Intermediate Points (q_i)	Intermediate Weights (w_i)
$\boldsymbol{e}_i\,,\,-\boldsymbol{e}_i$	For each case $w_i^e=\frac{(4-n)A_n}{2n(n+2)}$ for $i=1,\ldots,n$ where $A_n=\frac{2\sqrt{\pi^n}}{\Gamma\left(\frac{n}{2}\right)}$
$\boldsymbol{s}_i^+\,,\,-\boldsymbol{s}_i^+$ $\boldsymbol{s}_i^-\,,\,-\boldsymbol{s}_i^-$	For each case $w_i^s=\frac{A_n}{n(n+2)}$ $i=1,\cdots,\frac{n(n-1)}{2}$ where $A_n=\frac{2\sqrt{\pi^n}}{\Gamma\left(\frac{n}{2}\right)}$

$\boldsymbol{s}_i^+$ and $\boldsymbol{s}_i^-$ are generated as

$\boldsymbol{s}_i^+=\sqrt{\tfrac{1}{2}}(\boldsymbol{e}_i+\boldsymbol{e}_l):\ j<l,\ j,l=1,\cdots,n$

$\boldsymbol{s}_i^-=\sqrt{\tfrac{1}{2}}(\boldsymbol{e}_i-\boldsymbol{e}_l):\ j<l,\ j,l=1,\cdots,n$

Cubature Quadrature Points (ξ_k)	Weights (ω_k)
$\boldsymbol{e}_i\sqrt{2\lambda_j}\,,\,-\boldsymbol{e}_i\sqrt{2\lambda_j}$	$w_i^e\,\varpi_j$
$\boldsymbol{s}_i^+\sqrt{2\lambda_j}\,,\,-\boldsymbol{s}_i^+\sqrt{2\lambda_j}$	$w_i^s\,\varpi_j$
$\boldsymbol{s}_i^-\sqrt{2\lambda_j}\,,\,-\boldsymbol{s}_i^-\sqrt{2\lambda_j}$	$w_i^s\,\varpi_j$

λ_j is the solution of n'^{th} order Chebyshev-Laguerre polynomial with $\alpha=n/2-1$.

$$L_{n'}^{\alpha}=\lambda^{n'}-\tfrac{n'}{1!}(n'+\alpha)\lambda^{n'-1}+\tfrac{n'(n'-1)}{2!}(n'+\alpha)(n'+\alpha-1)\lambda^{n'-2}-\cdots=0$$

And the corresponding weights are obtained as

$$\varpi_j=\frac{1}{2\sqrt{\pi^n}}\,\frac{n'!\Gamma(\alpha+n'+1)}{\lambda_j\left[\dot{L}_{n'}^{\alpha}(\lambda_j)\right]^2}$$

Here, $i=1,\cdots,2n^2$ $j=1,\cdots,n'$ and $k=1,\cdots,2n^2n'$

9.4.3.5. Divided Difference Interpolation formula

The Divided Difference filter is based on Taylor series approximation of nonlinear functions using Stirling's Divided Difference Interpolation formula. Therefore the algorithmic steps are different from that of the Bayesian approach where the integrals are numerically approximated. Nevertheless the concept of Divided Difference filter has similarity with the Bayesian approach. The algorithm based on Divided Difference rule cannot be obtained directly from the general algorithm presented above. However, the algorithm of DDF can be adjusted and expressed in terms of points and weights as presented in [Liu2012] which partially matches with the general algorithm apart from the step for the computation of error covariance. The step for the propagation of covariance should be followed as stated below.

TABLE-9.5: SELECTION OF SIGMA POINTS & WEIGHTS

Sigma Points and weights for computation of Mean	
Sigma Points $(\boldsymbol{q}_i)$	Weights (w_i)
$\boldsymbol{0}$	$w_0 = \frac{h^2 - n}{h^2}$ for $n=0$
$h\boldsymbol{e}_i$	$w_i = \frac{1}{2h^2}$ for $i=1, \dots, n$
$-h\boldsymbol{e}_i$	
Sigma Points and weights for computation of Covariance	
Sigma Points $(\boldsymbol{q}_i)$	Weights (w_i)
$\boldsymbol{s}_i^1 = \boldsymbol{g}\left(\overline{\boldsymbol{x}} + h\sqrt{\overline{\boldsymbol{P}}}\boldsymbol{e}_i\right) - \boldsymbol{g}\left(\overline{\boldsymbol{x}} - h\sqrt{\overline{\boldsymbol{P}}}\boldsymbol{e}_i\right)$	$w_i^1 = 1/4h^2$
$\boldsymbol{s}_i^2 = \boldsymbol{g}\left(\overline{\boldsymbol{x}} + h\sqrt{\overline{\boldsymbol{P}}}\boldsymbol{e}_i\right) + \boldsymbol{g}\left(\overline{\boldsymbol{x}} - h\sqrt{\overline{\boldsymbol{P}}}\boldsymbol{e}_i\right) - 2\boldsymbol{g}(\overline{\boldsymbol{x}})$	$w_i^2 = (h^2 - 1)/4h^4$

A random variable with mean $\overline{\boldsymbol{x}}$ and covariance $\overline{\boldsymbol{P}}$ when propagated through a nonlinear function $\boldsymbol{g}(\cdot)$ the covariance is obtained after transformation using the step (different from general algorithm) as given by, $\hat{\boldsymbol{P}} = \sum_{i=1}^{n} w_i^1 \boldsymbol{s}_i^1 \left(\boldsymbol{s}_i^1\right)^T + \sum_{i=1}^{n} w_i^2 \boldsymbol{s}_i^2 \left(\boldsymbol{s}_i^2\right)^T$

These steps are to be used in the general algorithm while computing the error covariances. The mean on contrary can be obtained using the formula given in general approach.

9.4.4 Notes

- The general framework for adaptive nonlinear information filter is suitable for Unscented Transform, Gauss Hermite quadrature rule, Cubature rule, Cubature quadrature rule which are for numerical approximation of Gaussian integrals encountered in general framework developed on the basis of Bayesian approach. The Divided Difference interpolation formula, on contrary, follows the approach of Taylor series approximation and therefore differs in the algorithmic steps from that of the other methods. In the chapter on adaptive divided difference filter estimation algorithms for the standard error covariance form have been presented individually. However, in this chapter we have included the algorithm of adaptive divided difference information filter in the general algorithm with some significant modification in the step for computation of the error covariances.
- It is to be noted that the adaptation of measurement noise covariance is to be executed separately ($\hat{\boldsymbol{R}}_k^{\zeta}, \zeta = 1, \cdots, M$) for each sensors when the noise covariance of all of the sensors are unavailable. However, when the designer has the knowledge of the noise covariances of some of the sensors, adaptation steps for those sensors need not to be executed and only those covariances which are unknown should be adapted.

9.5 Square Root version

The square root approach for conventional error covariance form has been proposed in chapter 4. In this section general framework for adaptive nonlinear information filters in the square root framework has been proposed. Use of square root approach ensures positive definiteness of error covariance and adapted noise covariances. The rationale for following the square root approach has been discussed before and also applicable for the information filter configuration. The algorithmic steps are provided below.

9.5.1 Algorithm

GENERAL FRAMEWORK FOR ADAPTIVE NONLINEAR FILTERS IN SQUARE ROOT FORM

(i) Initialization: Initialize $\hat{\boldsymbol{x}}_{\boldsymbol{0}}, \hat{\boldsymbol{S}}_{\boldsymbol{0}}, \overline{\boldsymbol{S}}_{\boldsymbol{k}}^{\boldsymbol{Q}}, \left(\overline{\boldsymbol{S}}_{\boldsymbol{k}}^{\boldsymbol{R}}\right)^{\zeta}$

(ii) Time update step:

For the particular mean and covariance modify the selected points for standard normal distribution as $\hat{\boldsymbol{\chi}}_{\boldsymbol{i}} = \hat{\boldsymbol{S}}_{\boldsymbol{k-1}} \boldsymbol{q}_{\boldsymbol{i}} + \hat{\boldsymbol{x}}_{\boldsymbol{k-1}}$ (9.61)

Compute $\overline{\boldsymbol{x}}_{\boldsymbol{k}} = \sum_{i=1}^{N} \boldsymbol{f}(\hat{\boldsymbol{\chi}}_{\boldsymbol{i}}) w_i$ (9.62)

Compute the weighted, centred (*a posteriori* estimate of previous instant is subtracted off) matrix $\boldsymbol{S}_{\boldsymbol{k}}^{\boldsymbol{x}}$ such that for $i = 1, 2, \cdots, N$, i^{th} element of $\boldsymbol{S}_{\boldsymbol{k}}^{\boldsymbol{x}}$ becomes

$$\left(\boldsymbol{S}_{\boldsymbol{k}}^{\boldsymbol{x}}\right)_i = \left(\boldsymbol{f}(\hat{\boldsymbol{\chi}}_{\boldsymbol{i}}) - \hat{\boldsymbol{x}}_{\boldsymbol{k}-1}\right)\sqrt{w_i} \tag{9.63}$$

The estimate of the square root of *a priori* error covariance is obtained as

$$\overline{\boldsymbol{S}}_{\boldsymbol{k}} = \text{Triangularize}\left(\left[\boldsymbol{S}_{\boldsymbol{k}}^{\boldsymbol{x}} \quad \overline{\boldsymbol{S}}_{\boldsymbol{k}}^{\boldsymbol{Q}}\right]\right) \tag{9.64}$$

The information vector can be obtained as

$$\bar{\boldsymbol{z}}_{\boldsymbol{k}} = \overline{\boldsymbol{S}}_k^{-T} \overline{\boldsymbol{S}}_k^{-1} \overline{\boldsymbol{x}}_{\boldsymbol{k}} \tag{9.65}$$

The square root of the *a priori* information matrix is obtained as

$$\overline{\boldsymbol{S}}_k^{Z} = \text{Triangularize}\left(\overline{\boldsymbol{S}}_k^{-1}\right) \tag{9.66}$$

(iii) Measurement update step:

Select sigma points as $\overline{\boldsymbol{\chi}}_{\boldsymbol{i}} = \overline{\boldsymbol{S}}_{\boldsymbol{k}} \boldsymbol{q}_{\boldsymbol{i}} + \overline{\boldsymbol{x}}_{\boldsymbol{k}}$ (9.67)

The *a priori* estimate of measurement becomes

$$\overline{\boldsymbol{y}}_{\boldsymbol{k}}^{\boldsymbol{\zeta}} = \sum_{i=1}^{N} \boldsymbol{g}^{\zeta}(\overline{\boldsymbol{\chi}}_{\boldsymbol{i}}) w_i \tag{9.68}$$

Compute the weighted, centred (*a priori* estimate of measurement is subtracted off) matrix $\boldsymbol{S}_{\boldsymbol{k}}^{\boldsymbol{Y}}$ such that for $i = 1, 2, \cdots, N$, i^{th} element of $\boldsymbol{S}_{\boldsymbol{k}}^{\boldsymbol{Y}}$ becomes

$$\left(\boldsymbol{S}_{\boldsymbol{k}}^{\boldsymbol{Y}}\right)_i = \left(\boldsymbol{g}^{\zeta}(\overline{\boldsymbol{\chi}}_{\boldsymbol{i}}) - \overline{\boldsymbol{y}}_{\boldsymbol{k}}^{\boldsymbol{\zeta}}\right)\sqrt{w_i} \tag{9.69}$$

Compute the weighted, centred (*a priori* estimate of state is subtracted off) matrix $S_k^{\bar{x}}$ such that for $i = 1, 2, \cdots, N$ i^{th} element of $S_k^{\bar{x}}$ becomes

$$\left(\boldsymbol{S}_k^{\bar{x}}\right)_i = \left(\boldsymbol{\chi}_i - \bar{\boldsymbol{x}}_k\right)\sqrt{w_i} \tag{9.70}$$

The cross covariance can be computed as follows

$$\left(\boldsymbol{P}_k^{xz}\right)^{\zeta} = \boldsymbol{S}_k^{\bar{x}}\left(\boldsymbol{S}_k^{Y}\right)^T \tag{9.71}$$

Define the matrix, $\boldsymbol{\Lambda}_k^{\zeta}$ as

$$\boldsymbol{\Lambda}_k^{\zeta} = \bar{\boldsymbol{S}}_k^{-T}\bar{\boldsymbol{S}}_k^{-1}\left(\boldsymbol{P}_k^{xz}\right)^{\zeta}\left[\left(\bar{\boldsymbol{S}}_k^{R}\right)^{\zeta}\right]^{-T} \tag{9.72}$$

The *a posteriori* estimate of information vector is obtained as

$$\hat{\boldsymbol{z}}_k = \bar{\boldsymbol{z}}_k + \boldsymbol{\Lambda}_k^{\zeta}\left[\left(\bar{\boldsymbol{S}}_k^{R}\right)^{\zeta}\right]^{-1}\left(\boldsymbol{y}_k^{\zeta} - \bar{\boldsymbol{y}}_k^{\zeta} + \left[\left(\boldsymbol{P}_k^{xz}\right)^{\zeta}\right]^T\bar{\boldsymbol{z}}_k\right) \tag{9.73}$$

The square root of the *a posteriori* estimate of information matrix becomes

$$\hat{\boldsymbol{S}}_k^{Z} = \text{cholupdate}\left(\bar{\boldsymbol{S}}_k, \boldsymbol{\Lambda}_k^{\zeta}, +\right) \tag{9.74}$$

The square root of the corresponding error covariance matrix

$$\hat{\boldsymbol{S}}_k = \text{Triangularize}\left(\left(\hat{\boldsymbol{S}}_k^{Z}\right)^{-1}\right) \tag{9.75}$$

Hence the *a posteriori* estimate of state becomes

$$\hat{\boldsymbol{x}}_k = \hat{\boldsymbol{S}}_k\hat{\boldsymbol{z}}_k \tag{9.76}$$

(iv) Q-Adaptation Steps:

When $\boldsymbol{Q}$ is unknown, on contrary, $\boldsymbol{R}$ is known, i.e., $\bar{\boldsymbol{S}}_k^{R} = \boldsymbol{S}_k^{R}$, replace $\bar{\boldsymbol{S}}_k^{Q}$ by $\hat{\boldsymbol{S}}_{k-1}^{Q}$ (the adapted standard deviation of $\boldsymbol{Q}$ of previous instant) in (9.64). Then the following steps are to be executed for adaptation of square root of process noise covariance.

Compute the state residual sequence as

$$\boldsymbol{\eta}_k = \hat{\boldsymbol{x}}_k - \bar{\boldsymbol{x}}_k \tag{9.77}$$

Compute the matrix from the state residual sequence as

$$\boldsymbol{S}_k^{\eta} = \sqrt{1/L}\,\left[\boldsymbol{\eta}_k(k-L+1) \quad \cdots \quad \boldsymbol{\eta}_k(k)\right] \tag{9.78}$$

where L denotes the window length.

The adapted square root of process noise covariance $\hat{\boldsymbol{S}}_k^{Q}$ is obtained as

$$\hat{\boldsymbol{S}}_{\boldsymbol{k}}^{\boldsymbol{Q}} = \text{Triangularize}\left(\boldsymbol{S}_{\boldsymbol{k}}^{\boldsymbol{\eta}}\right) \tag{9.80}$$

(v) R-Adaptation Steps:

When $\boldsymbol{R}$ is unknown, on contrary, $\boldsymbol{Q}$ is known, i.e., $\overline{\boldsymbol{S}}_{\boldsymbol{k}}^{\boldsymbol{Q}} = \boldsymbol{S}_{\boldsymbol{k}}^{\boldsymbol{Q}}$ replace $\overline{\boldsymbol{S}}_{\boldsymbol{k}}^{\boldsymbol{R}}$ by $\hat{\boldsymbol{S}}_{\boldsymbol{k-1}}^{\boldsymbol{R}}$ (the adapted standard deviation of $\boldsymbol{R}$ of previous instant) in (9.72) and (9.73). Then the following steps are to be executed for adaptation of the square root of measurement noise covariance.

Select sigma points as $\hat{\boldsymbol{\chi}}_i^+ = \hat{\boldsymbol{S}}_{\boldsymbol{k}}\, \boldsymbol{q}_{\boldsymbol{i}} + \hat{\boldsymbol{x}}_{\boldsymbol{k}}$ (9.81)

Compute *a posteriori* estimate of measurement as

$$\hat{\boldsymbol{y}}_{\boldsymbol{k}}^{\boldsymbol{\zeta}} = \sum_{i=1}^{N} \boldsymbol{g}^{\zeta}(\hat{\boldsymbol{\chi}}_i^+) w_i \tag{9.82}$$

Compute the weighted, centered (*a posteriori* estimate of measurement is subtracted off) matrix $\left(\boldsymbol{S}_{\boldsymbol{k}}^{\hat{y}}\right)^{\zeta}$ such that for $i = 1, 2, \cdots, N$, i^{th} element of $\left(\boldsymbol{S}_{\boldsymbol{k}}^{\hat{y}}\right)^{\zeta}$ becomes

$$\left(\boldsymbol{S}_{\boldsymbol{k}}^{\hat{y}}\right)_i^{\zeta} = \left(\boldsymbol{g}^{\zeta}(\hat{\boldsymbol{\chi}}_i^+) - \hat{\boldsymbol{y}}_{\boldsymbol{k}}^{\boldsymbol{\zeta}}\right)\sqrt{w_i} \tag{9.83}$$

Compute the residual sequence as given by

$$\boldsymbol{\rho}_{\boldsymbol{k}}^{\boldsymbol{\zeta}} = \boldsymbol{y}_{\boldsymbol{k}}^{\boldsymbol{\zeta}} - \hat{\boldsymbol{y}}_{\boldsymbol{k}}^{\boldsymbol{\zeta}} \tag{9.84}$$

Compute the matrix from the residual sequence as

$$\left(\boldsymbol{S}_{\boldsymbol{k}}^{\boldsymbol{\rho}}\right)^{\zeta} = \sqrt{1/L}\,\left[\boldsymbol{\rho}_{\boldsymbol{k}}^{\boldsymbol{\zeta}}(k - L + 1) \quad \cdots \quad \boldsymbol{\rho}_{\boldsymbol{k}}^{\boldsymbol{\zeta}}(k)\right] \tag{9.85}$$

where L denotes the window length.

The estimate of the square root of measurement noise covariance is obtained as

$$\hat{\boldsymbol{S}}_{\boldsymbol{k}}^{\boldsymbol{R}} = \text{Triangularize}\left(\left[\left(\boldsymbol{S}_{\boldsymbol{k}}^{\hat{y}}\right)^{\zeta} \quad \left(\boldsymbol{S}_{\boldsymbol{k}}^{\boldsymbol{\rho}}\right)^{\zeta}\right]\right) \tag{9.86}$$

(vi) Recursion: The time update and measurement update steps are repeated for obtaining the estimates for the subsequent time steps starting from k=1.

9.5.2 Multiple Sensor Fusion

The information contribution of ζ^{th} sensor is denoted as

$$\boldsymbol{\varphi}_k^{\zeta} = \boldsymbol{\Lambda}_k^{\zeta}\left[\left(\overline{\boldsymbol{S}}_k^{\boldsymbol{R}}\right)^{\zeta}\right]^{-1}\left(\boldsymbol{y}_k^{\zeta} - \overline{\boldsymbol{y}}_k^{\zeta} + \left[\left(\boldsymbol{P}_k^{xz}\right)^{\zeta}\right]^{T}\overline{\boldsymbol{z}}_k\right) \tag{9.87}$$

The contributions of all the sensors starting from $\zeta = 1,\cdots,M$ are fused to obtain a more reliable estimate as

$$\hat{\boldsymbol{z}}_k = \overline{\boldsymbol{z}}_k + \sum_{\zeta=1}^{M}\boldsymbol{\varphi}_k^{\zeta} \tag{9.88}$$

The information matrix contribution for the ζ^{th} sensor is $\boldsymbol{\Lambda}_k^{\zeta}$. After multiple sensor fusion the square root of the *a posteriori* information matrix is obtained as

$$\hat{\boldsymbol{S}}_k^{\boldsymbol{Z}} = \text{cholupdate}\left(\overline{\boldsymbol{S}}_k,\ \left[\boldsymbol{\Lambda}_k^{1} \quad \cdots \quad \boldsymbol{\Lambda}_k^{M}\right], +\right) \tag{9.89}$$

9.5.3 Notes

In the algorithm of adaptive information filters in square root approach the matrix inversion steps may be replaced by backward substitution symbolized by '/' as the latter is computationally economic. Unlike the standard error covariance form in case of square root approach the triangular matrix is obtained from the QR factorization. On the availability of the square upper triangular matrix one can follow the method of back substitution instead of matrix inversion to reduce the computational burden.

Note that only residual based $\boldsymbol{R}$ adaptation algorithm is presented in the square root approach as it supports adaptation of the standard deviation of $\boldsymbol{R}$ by 'triangularization'. This cannot be done for innovation based $\boldsymbol{R}$ adaptation as 'cholupdate' is required for subtraction. The intuition of the present worker says that here also the singularity problems cannot be avoided.

The algorithm provided above is for the sigma point rule which have non negative weights like, Gauss Hermite quadrature rule, 3rd degree Cubature rule, Cubature quadrature rule. However, for scaled Unscented transformation rule, 5th degree cubature rule this algorithm cannot be applied and a modification using the "cholupdate" has to be incorporated for the negative weights. In this dissertation square root version for UT rule and 5th degree cubature rule are not presented as these algorithm may suffer from the loss of positive definiteness in some situations [Liu2012].

9.6 Characterization of proposed algorithms

The proposed filtering algorithms have been evaluated with the help of two different multiple sensor estimation problems in simulation. The situations have been considered where either noise covariances of some sensors are unavailable or the process noise covariance unavailable due to modelling uncertainty or unknown parameter variation. In the first case a ballistic object tracking problem has been considered which is to be tracked by multiple radars. In second case a tracking problem is considered where a maneuvering aircraft has to be tracked using measurements from multiple tracking radars.

9.6.1 Ballistic Object Tracking Problem

The tracking of a ballistic object during re-entry phase has been described in chapter 3. $\boldsymbol{Q}$ adaptive versions of UIF, DDIF and $\boldsymbol{R}$ adaptive DDIF have been evaluated considering this problem where the object is being tracked by multiple tracking radars at different locations in the atmosphere. Only range measurements have been considered to be available from these radars. The simulation results illustrate that the proposed adaptive information filters work satisfactorily and make sensor fusion successful in the face of unknown process noise covariance.

9.6.1.1. Demonstration of $\boldsymbol{Q}$ adaptive UIF and DDIF

To generate the true state trajectories of object, the truth value of initial kinematic states and the ballistic parameter are chosen following [Norgaard2000] as specified in chapter 3. The necessary parameters for the filters are also provided in same table. The process noise covariance is considered to be unknown in this problem. Therefore $\hat{\boldsymbol{Q}}_{0}$ is initialized arbitrarily with an assumed value with large error.

The plots of RMS values of estimation errors of altitude, velocity and ballistic parameter of the object have been compared for ADDIF, AUIF and non-adaptive UIF from Fig. 9.1, Fig. 9.2 and Fig. 9.3 respectively. Because of the assumption of unavailability of process noise covariance both the filters are initialized with a value of $\hat{\boldsymbol{Q}}_{0}$ with a large error compared to the true $\boldsymbol{Q}$ as mentioned above. Specifically, for this case study initial value of $\boldsymbol{Q}$ is chosen as $\hat{\boldsymbol{Q}}_{0} = 10^{5} \times \boldsymbol{Q}_{true}$. For adaptive filters choice of sliding window length is considered to be 10 time instants. From the Monte Carlo simulation it has been found that for all three cases convergence of the RMS errors of AUIF is better compared to its non-adaptive version and the steady state value is also lower for the proposed AUIF. The performance of ADDIF is comparably same with AUIF. ADDIF like AUIF presents satisfactory estimation and capable of online adaptation of unknown noise process covariance.

For a representative run, Fig. 9.4 shows the plot of $\boldsymbol{Q}_{true}(3,3)$ and $\hat{\boldsymbol{Q}}_k(3,3)$ for AUIF. It is observed that even though $\boldsymbol{Q}$ is initialized with an arbitrary initial choice with large error, ($\hat{\boldsymbol{Q}}_0 = 10^5 \times \boldsymbol{Q}_{true}$), the Adapted $\hat{\boldsymbol{Q}}_k(3,3)$ converges on the truth value and continues to track it. However, time for this convergence is considerably high, near about 30 sec.

Note that the RMSE of adaptive information filters are also compared with Non-adaptive UIF in the ideal situation when $\boldsymbol{Q}$ is accurately known. It is observed that the plots of RMSE for adaptive information filters become comparable with those for non-adaptive UIF after 30 sec. The delay in convergence of adapted $\boldsymbol{Q}$ is the reason for the initial mismatch in the plots of RMSE.

The initial rise in the RMSE about 10 sec is because of the influence of the drag during reentry. As the object enters the atmosphere it experiences drag and system nonlinearity becomes pronounced. As a consequence, the RMSE of all the filters (both adaptive and non-adaptive) tend to rise temporarily before they finally settle down to a lower value.

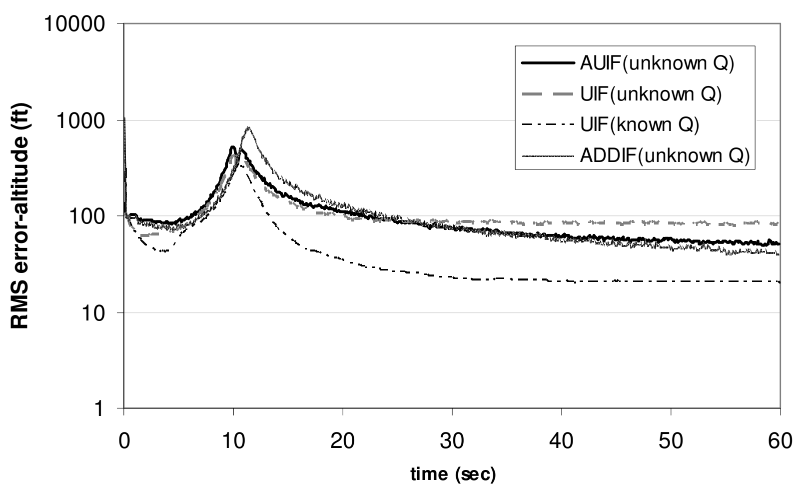


Fig. 9.1: Comparison of RMSE (altitude) of AUIF, ADDIF & DDIF for 1000 MC runs

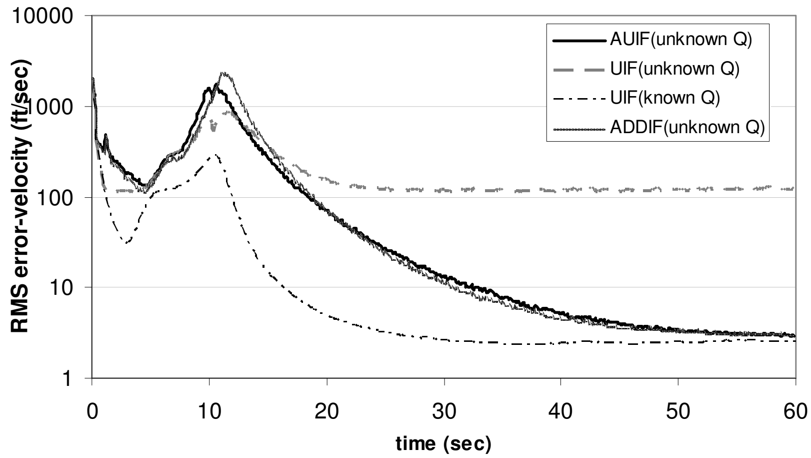


Fig. 9.2: Comparison of RMSE (velocity) of AUIF, ADDIF & DDIF for 1000 MC runs

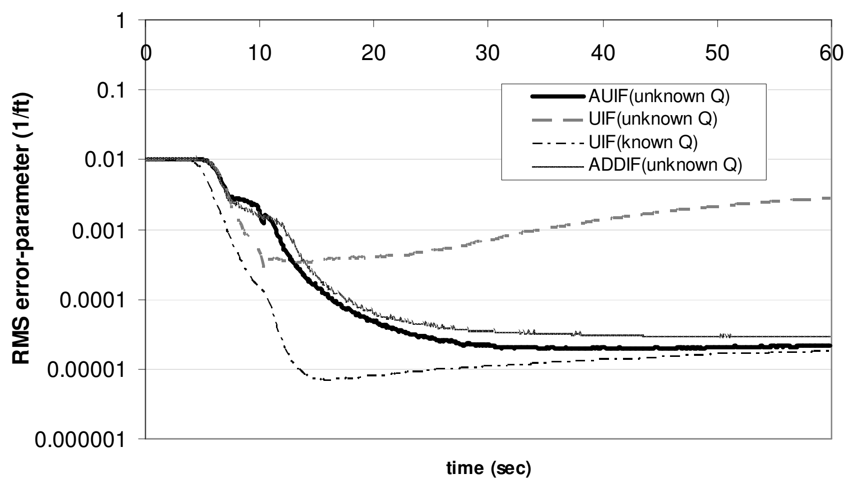


Fig. 9.3: Comparison of RMSE (ballistic parameter) of AUIF, ADDIF & DDIF for 1000 MC runs

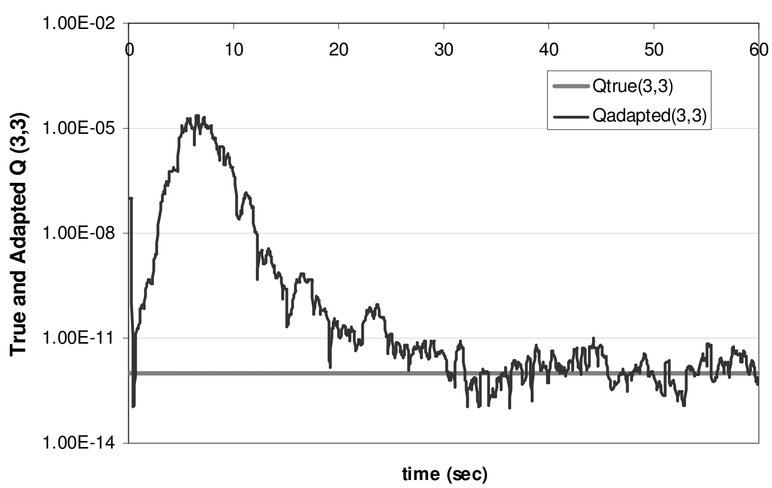


Fig. 4: Plot of estimated process noise covariance ($\boldsymbol{Q}_{3,3}$) for a representative run

9.6.1.2. Demonstration of *R*- Adaptive DDIF

In this section the performance of ***R*** adaptive DDIF is presented. The object tracking problem is considered again and assumed that it is assumed that some sensor remains uncharacterised and their noise covariances remain unknown. When the measurement noise covariance of any of the sensors remains unknown, that particular $\boldsymbol{R}_k^{\zeta}$ is initialized arbitrarily with an optimistic choice of $\boldsymbol{R}_k^{\zeta}$. The term optimistic choice signifies the initialization of the filter with a sufficient low value of $\boldsymbol{R}_k^{\zeta}$ so that the measurement of that particular sensor may not get underweighted. This optimistic choice, however, may affect the estimation performance when there is sufficient discrepancy between the true value of $\boldsymbol{R}_k^{\zeta}$ and the optimistic choice of $\boldsymbol{R}_k^{\zeta}$ ($\boldsymbol{R}_{filter}$) for filter initialization. However, initialization of $\boldsymbol{R}_k^{\zeta}$ ($\boldsymbol{R}_{filter}$) with a higher value will not always deteriorate the estimation performance during multi sensor estimation as this choice underweights that particular measurement and consequently ignore that measurement. It is, therefore, not recommended to initialize the unknown covariance with a higher value as such a choice contradicts the concept of the reliability of sensor fusion underweighting those measurements.

The $\boldsymbol{R}_{filter}$ for the unknown sensor noise covariance is assigned with a sub multiple of ten so that the choice is an optimistic one. For this particular case study $\boldsymbol{R}_{filter}$ for radar 1 is considered to be two decade lower than the true ***R***. Window length is chosen as 100. For this case study an initial stop time equal with window length is set. The justification for this stop time is given below. Initially (during first 10 second of the descend) the object remains in exo atmospheric zone. In this zone the influence of drag is negligible and the ballistic parameter, therefore, cannot be estimated satisfactorily. Consequently the residual or innovation from filter loses the whiteness and adaptation may not be accurate. A stop time in adaptation may therefore be suggested for ballistic object tracking problem. Otherwise because of fading memory due to sliding window based adaptation, this error can it will affect detrimentally the adapted value. An alternative way is to choose a low window size so that the memory gets faded quickly.

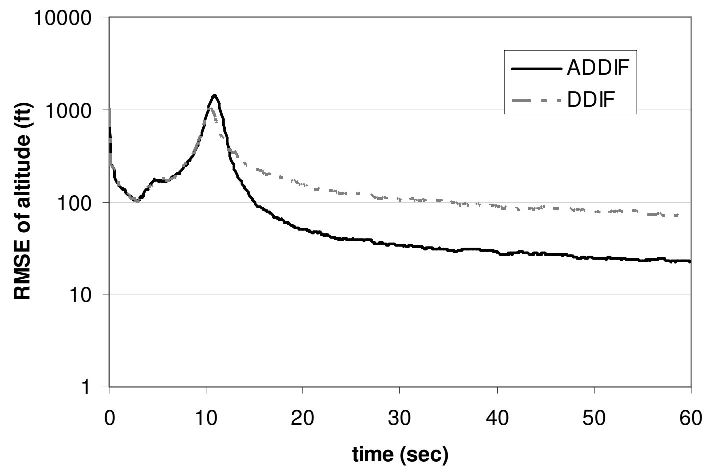


Fig. 9.5: Comparison of RMS error (altitude estimation) of ADDF & DDF for 1000 MC runs

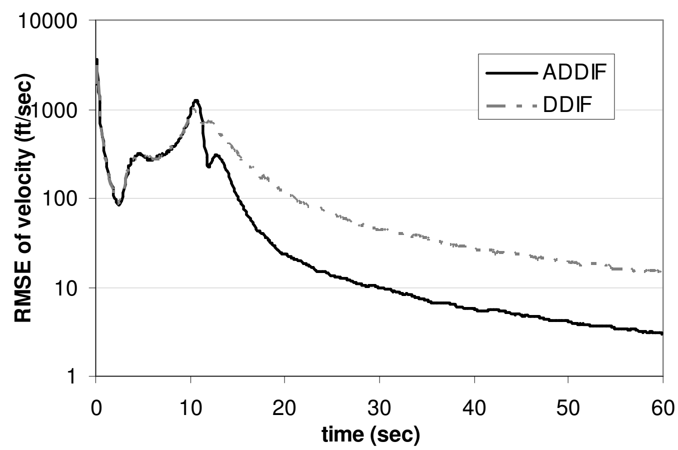


Fig. 9.6: Comparison of RMS error (velocity estimation) of ADDF & DDF for 1000 MC runs

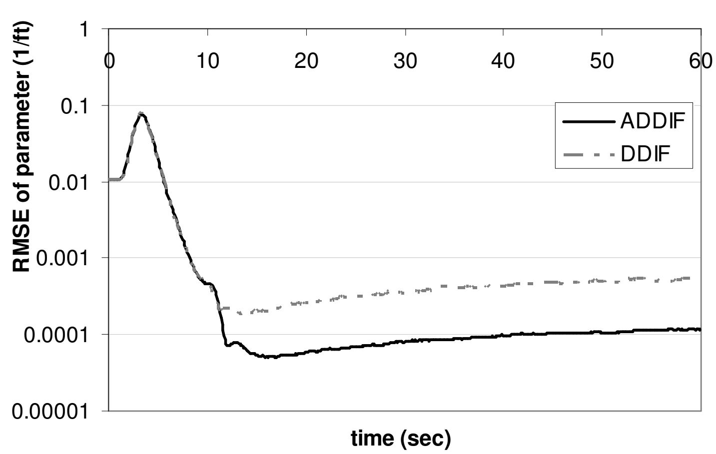


Fig. 9.7: Comparison of RMS error (ballistic parameter estimation) of ADDF & DDF for 1000 MC runs

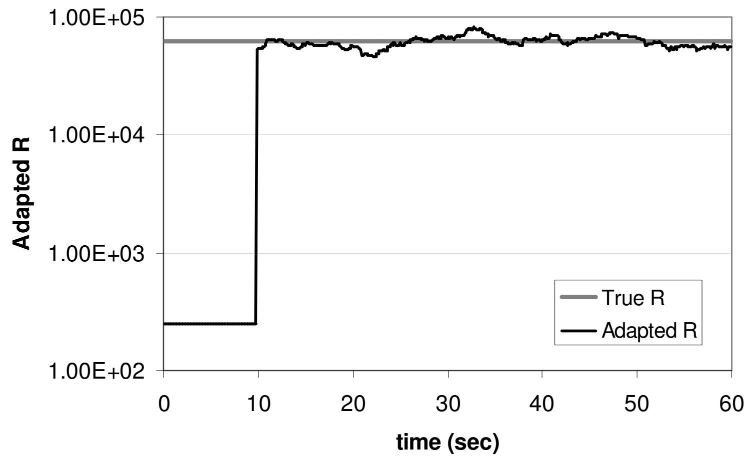


Fig. 9.8: Plot of true and adapted $\boldsymbol{R}$ for a representative run

The plots of RMS values of estimation errors of altitude, velocity and ballistic parameter of the object have been compared for both adaptive and non-adaptive DDIF. From 1000 Monte Carlo simulation it has been found that the RMSE of ADDIF has converged to a lower value compared to that of non-adaptive DDIF. Both the filter have been initialized with measurement noise covariance different from that of the truth value. Fig. 9.5 – 9.7 are presented in support of the above statement.

In case of multiple sensor fusion problem situations may arise when the noise covariances of one or more sensors remain unknown. The above results are presented for the case when the noise covariance of only one of the measurements is unknown. In this context it can be inferred from the observation that the non-adaptive DDIF cannot present satisfactory estimation performance even when the measurement noise covariance of only one of the measurements is unknown and initialized with an assumed choice of $\boldsymbol{R}$. This limitation can be overcome by employing ADDIF which can adapt the unknown noise covariance and ensures satisfactory estimation.

The same study may be repeated for the situation when prior knowledge about measurement noise covariances for more than one measurement remains unavailable. In those situations it would be observed that ADDIF excels over non-adaptive DDIF.

For a representative run the true and adapted measurement noise covariance for the particular measurement has been presented by Fig. 9.8 to demonstrate that even though initialized with an assumed choice of ***R***, the Adapted ***R*** converges on the truth value and tracks it for subsequent times.

The ballistic object tracking problem is considered again to characterize ADDIF in presence of a biased measurement. Here also it would be shown that ADDIF outperforms non-adaptive DDIF. A level bias of 10000 ft has been introduced in the measurement received from the second radar at the instant of 20 sec. The performance of the filter is presented by the Fig. 9.9 to Fig. 9.12.

It is observed from Fig. 9.9- 9.11 that the RMSE of ADDIF is at a lower value than that of non-adaptive DDIF. It is also to be noted from Fig. 9.12 that when the bias is introduced in the measurement at the instant of 20 sec the adapted ***R*** corresponding to the biased measurement rises to a higher value so that the biased measurement gets under weighted until the bias present in that measurement. Thus ADDIF always ensures improved estimation performance ignoring the biased measurement. The non-adaptive version cannot underweight that measurement as the scope of ***R*** adaptation is not present there.

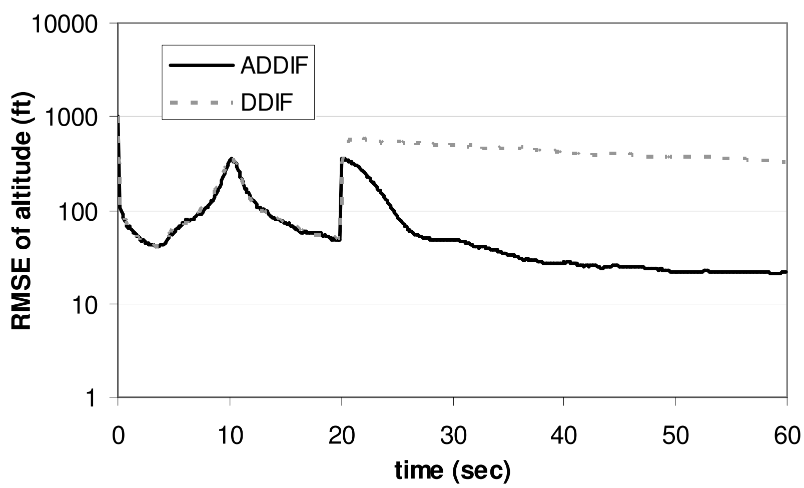


Fig. 9.9: Comparison of RMS error (altitude estimation) of ADDF & DDF for 1000 MC runs

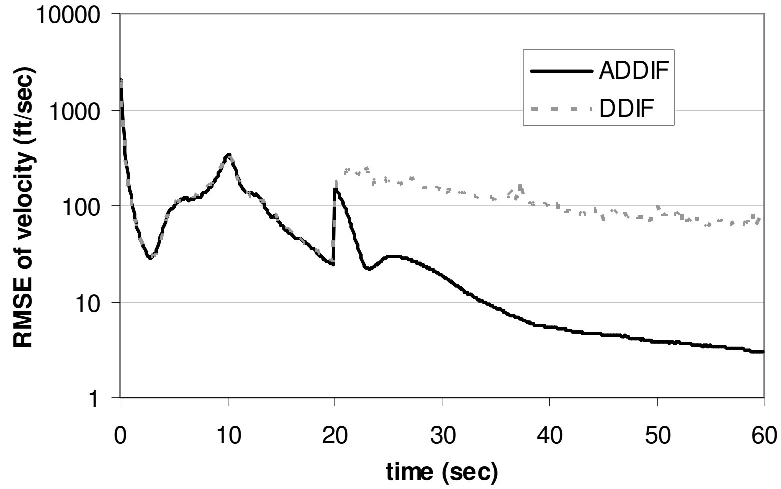


Fig. 9.10: Comparison of RMS error (velocity estimation) of ADDF & DDF for 1000 MC runs

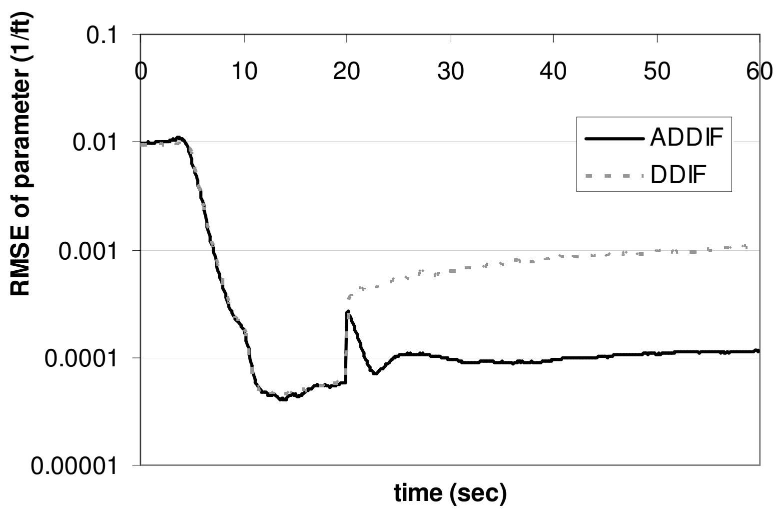


Fig. 9.11: Comparison of RMS error (ballistic parameter estimation) of ADDF & DDF for 1000 MC runs

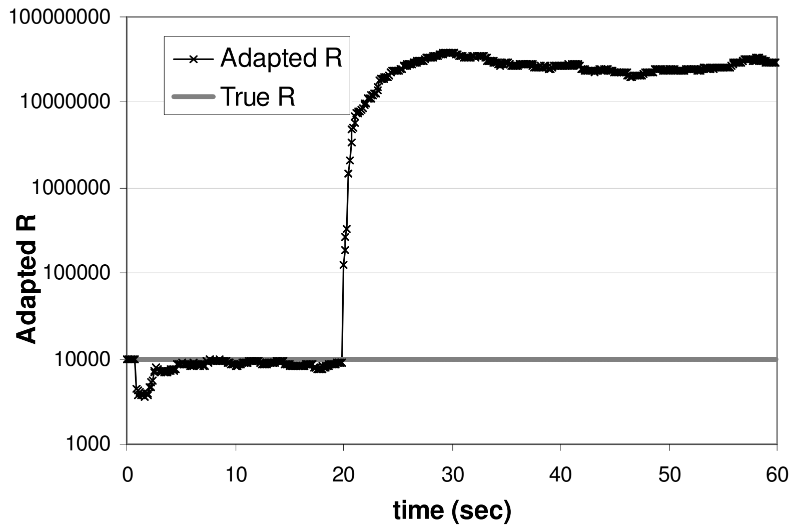


Fig. 9.12: Plot of adapted $\boldsymbol{R}$ for the faulty measurement (for a representative run)

9.6.2 Aircraft Tracking Problem

9.6.2.1. Demonstration of $\boldsymbol{Q}$ adaptive information filters

The aircraft tracking problem described in chapter 3 has been considered as a case study for evaluation of $\boldsymbol{Q}$ adaptive information filers. The situation is considered where an aircraft maneuvering with an unknown time varying turn rate has to be tracked with multiple radars. The unknown variation of turn rate cannot be modelled appropriately and therefore induces parametric uncertainty in the system dynamics. It may also be considered that element of the process noise covariance associated with turn rate remains unknown for this situation. $\boldsymbol{Q}$ adaptive information filers are employed for successful sensor fusion in this case study. As the other elements of $\boldsymbol{Q}$ are known, only the unknown element needs to be adapted while other should remain same as the truth value. This may be executed by the partial adaptation of $\boldsymbol{Q}$ which has also been illustrated in chapter 6. For this case study $\boldsymbol{Q}(5,5)$ is adapted. The $\boldsymbol{Q}(5,5)$ being unknown, it is arbitrarily initialized as 20 times of $\boldsymbol{Q}_{true}(5,5)$. The window size is taken as 10.

From the Monte Carlo simulation with 10000 runs, performance of $\boldsymbol{Q}$ adaptive versions of GHIF, CQIF, CIF and DDIF is compared with that of their non-adaptive version in the situation when the turn rate of the aircraft is unknown. In each case it is observed that estimation performance of adaptive filter is superior compared to the non-adaptive version.

RMSE plots for different estimators are presented for illustration. Note that the RMSE plots are presented excluding the cases where track loss occurs. The occurrence of track loss has been explained and demonstrated in chapter 3. During tracking of the aircraft which is executing a maneuvering turn the estimators are susceptible to lose the track for bearing only measurements in some cases. Note that track losses cannot be over ruled even for the ideal situations when the filter is properly tuned with prior knowledge of all the noise covariances. The percentage of track losses helps to analyze the performances of the filters in a quantitative way. Lesser the tendency of track loss more accuracy the estimator has. Table 9.7 presents the percentage of track losses for the above estimators.

It has been observed from Fig. 9.13, Fig. 9.14, Fig. 9.15 that the performance of adaptive information filters are substantially superior to that of non-adaptive counter parts as the RMSE for all three states converged to a lower steady state value within reasonably less time. Fig. 9.13 presents RMSE of adaptive and non-adaptive GHIF for position estimation. Fig. 9.14 presents the RMSE of adaptive and non-adaptive CIF (3rd degree) for velocity estimation and Fig. 9.15 depicts the RMSE of adaptive and non-adaptive CQIF for turn rate estimation. It is important to note that although the elements of $\boldsymbol{Q}$ related to position and velocity are known RMSE of position and velocity for the non-adaptive filters are deteriorated because of the implicit effect of inadequately estimated turn rate.

As the proposed filters are validated in simulation, it is also possible to compare the RMSE performance of proposed filters with their non-adaptive counterpart in the ideal situation where $\boldsymbol{Q}_{true}(5,5)$ is known only to the non-adaptive version. This comparison illumines how far the performance of adaptive filters (without complete knowledge of $\boldsymbol{Q}$) is close to that of conventional non-adaptive filter in ideal situation with known $\boldsymbol{Q}$. It is demonstrated that the RMSE of adaptive filters for all the states are nearly comparable to the nature of RMSE of non-adaptive filter in ideal condition. The initial mismatch in RMSE is because of the time taken for adapted element of $\boldsymbol{Q}$ to converge on the truth value which has been shown in Fig. 9.16.

Fig. 9.16 illustrates the $\boldsymbol{Q}$ adaptation performance of the adaptive filters. For all of the proposed information filters the adapted value of $\boldsymbol{Q}_{true}(5,5)$ converged to the truth value even though initialized with an erroneous assumed value. The adapted value converges with 30 sec approximately and continues to track the truth value.

The performance comparison of the proposed ***Q*** adaptive information filters, viz., QA-DDIF, QA-CIF (3rd degree), QA-GHIF, and QA-CQIF (3rd degree) has been carried out on the basis of RMS errors and percentage of track loss out of 10000 Monte Carlo runs.

It has been observed from the RMSE plot of position estimation in Fig. 9.17 (excluding the track loss case) that the RMSE of proposed adaptive filters are performance wise comparable. Same trend is observed for velocity and turn rate estimation and are not presented to avoid repetition. However, on the basis of track loss performance given by Table 9.6 it may be commented that AGHIF and ACQIF are better alternatives than ADDIF and ACIF as they indicate less percentage of track loss. It is also to be noted here that although AGHIF and ACQIF are performance wise equivalent the latter is computationally less expensive as it uses comparatively less number of quadrature points. It is mentioned in the previous chapters that the quadrature points for AGHIF rises exponentially with the system dimension. On contrary, the quadrature points for ACQIF rises linearly with the system dimension.

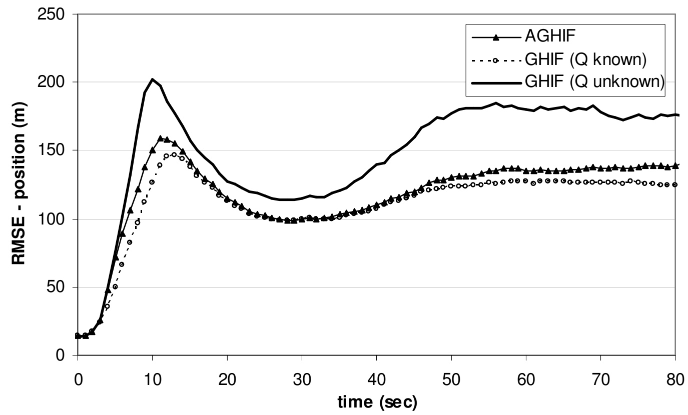


Fig. 9.13: Comparison of RMSE (position estimation) of AGHIF & GHIF for 10000 MC runs

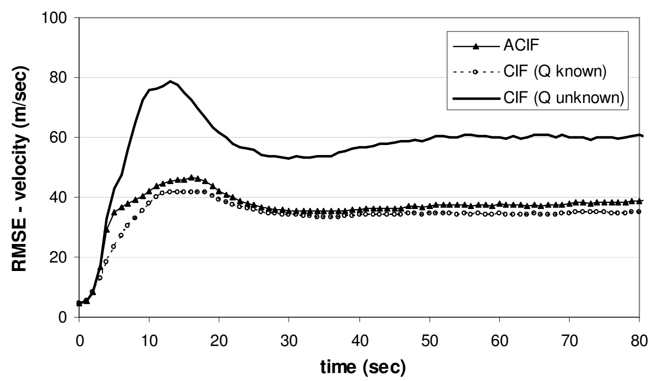


Fig. 9.14: Comparison of RMSE (velocity estimation) of ACIF & CIF for 10000 MC runs

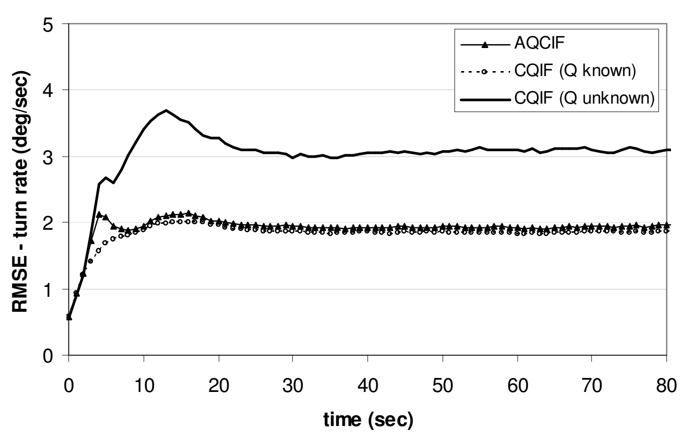


Fig. 9.15: Comparison of RMSE (turn rate estimation) of ACQIF & CQIF for 10000 MC runs

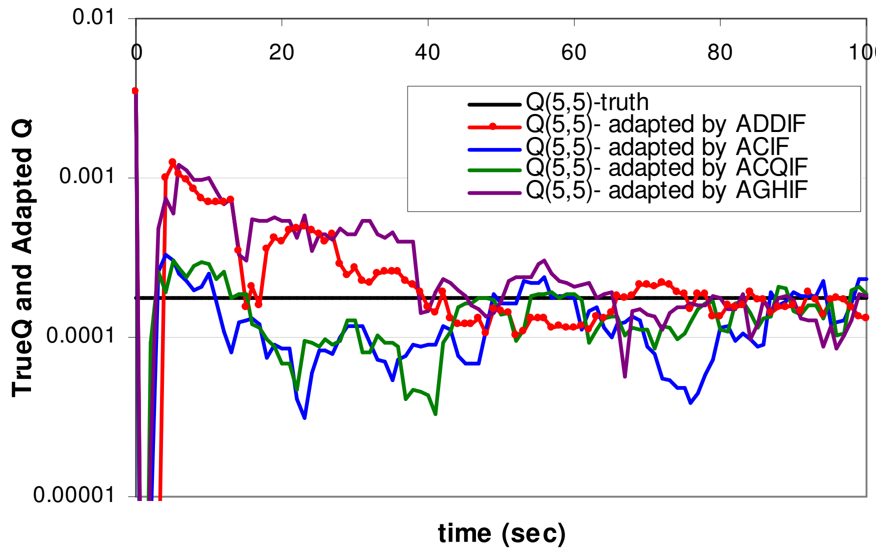


Fig. 9.16: Plot of estimated process noise co-variance (Q5,5) for a representative run

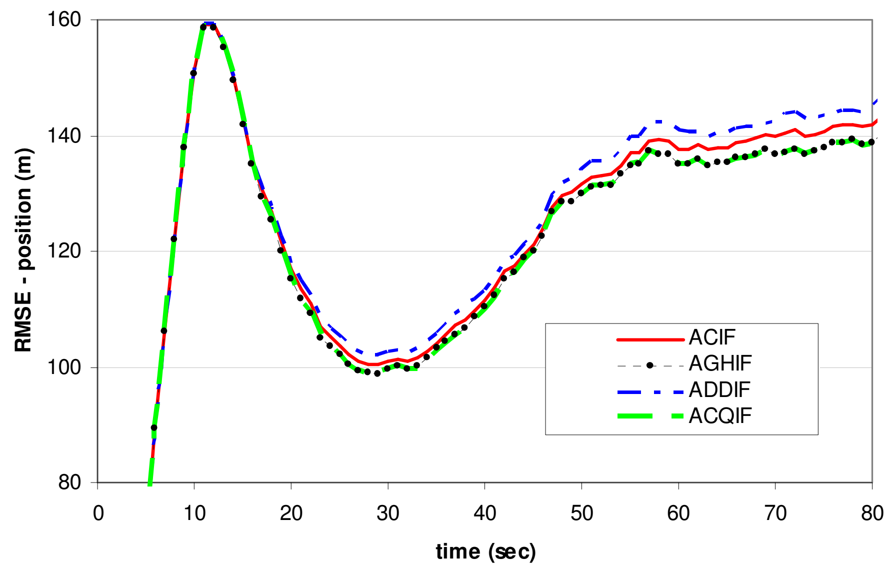


Fig. 9.17: Comparison of RMS error (position estimation) of ACIF, AGHIF, ADDIF, ACQIF for 10000 MC runs

TABLE-9.6 : Comparative study of %-age of track loss form ***Q*** adaptive information filters

Underlying framework used	Percentage of track loss for		
	Adaptive filter with unknown ***Q***	Non-adaptive filter with unknown ***Q***	Non-adaptive filter with known ***Q*** (ideal)
CIF	1.02%	4.10%	0.91%
DDIF	1.14%	4.17%	0.97%
GHIF	0.84%	3.67%	0.64%
CQIF	0.86%	3.69%	0.64%

9.6.2.2. Demonstration of *R* adaptive information filters

The aircraft tracking problem has also been considered where the knowledge of noise covariance of the bearing measurement from one of the tracking radar remains unavailable. The filter is initialized with $\boldsymbol{R}_{filter}$ (an optimistic choice of ***R*** is made as discussed before). $\boldsymbol{R}_{filter}$ for radar 1 is considered as 20 times lower than the truth value and the window size is taken as 25. Rest of the parameters remain the same as given in chapter 3.

In this case study we have compared the performance of ***R*** adaptive sigma point information filters, viz., RA-DDIF, RA-UIF, RA-GHIF, RA-CIF (5^{th} degree) and RA-HCQIF (5^{th} degree). The performance is compared on the basis of and percentage of track loss. RMSE are presented in Fig. 9.18 -9.20 excluding the cases where track losses had occurred.

It has been observed from the RMSE of position and velocity that the RMSE of RA-DDIF and RA-UIF are performance wise equivalent. However, RMSE of RA-GHIF, RA-CIF (5^{th} degree) and RA-CQIF (5^{th} degree) is slightly less than that for other filters for both position and velocity. Nevertheless RMSE the plots are found comparable from the figures for all the estimators. The performance accuracy of the estimators can therefore be compared on the basis of percentage of track loss given in Table 9.7. The percentage of track loss for RA-GHIF, RA-CQIF (5^{th} degree) and RA-CIF (5^{th} degree) are comparably same and significantly less than that for RA-UIF and RA-DDIF among 10000 Monte Carlo runs.

It is also to be noted that among RA-GHIF, RA-HCQIF (5^{th} degree) and RA-CIF (5^{th} degree) the latter is computationally less expensive as it uses less number of points compared to the other two estimators. For a single run an average computation time for RA-CIF (5^{th} degree) and RA-HCQIF (5th degree) are 26.13% and 64.71% of the average computation time for RA-GHIF respectively. The simulations are carried out using MATLAB (version 7.9.0.529) in a computer with specifications Intel®, Core (TM) 2 Duo CPU, 2.8 GHz, 2 GB RAM.

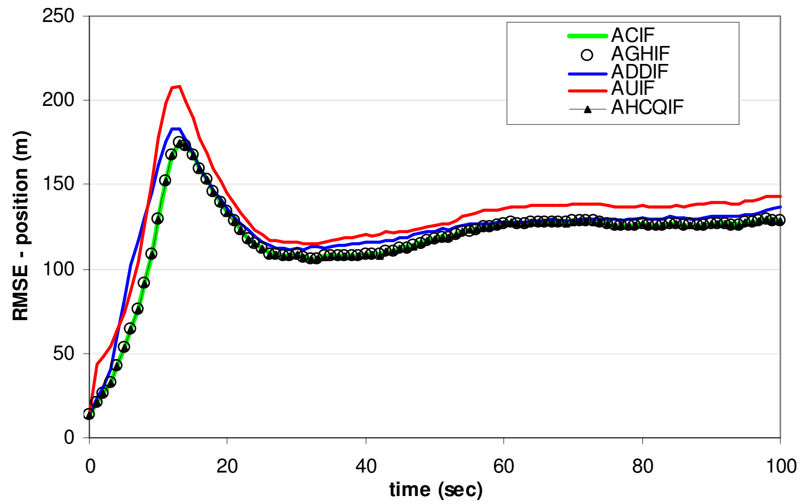


Fig. 9.18: Comparison of RMS error (position estimation) of ACIF, AGHIF, ADDIF, AUIF, AHCQIF for 10000 MC runs

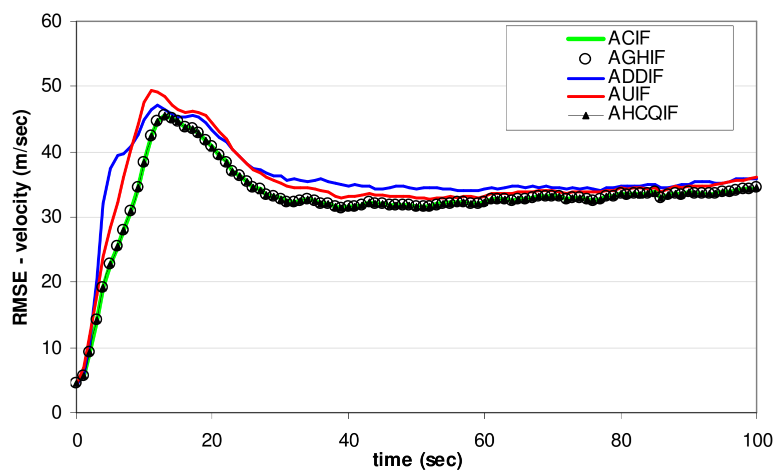


Fig. 9.19: Comparison of RMS error (velocity estimation) of ACIF, AGHIF, ADDIF, AUIF, AHCQIF for 10000 MC runs

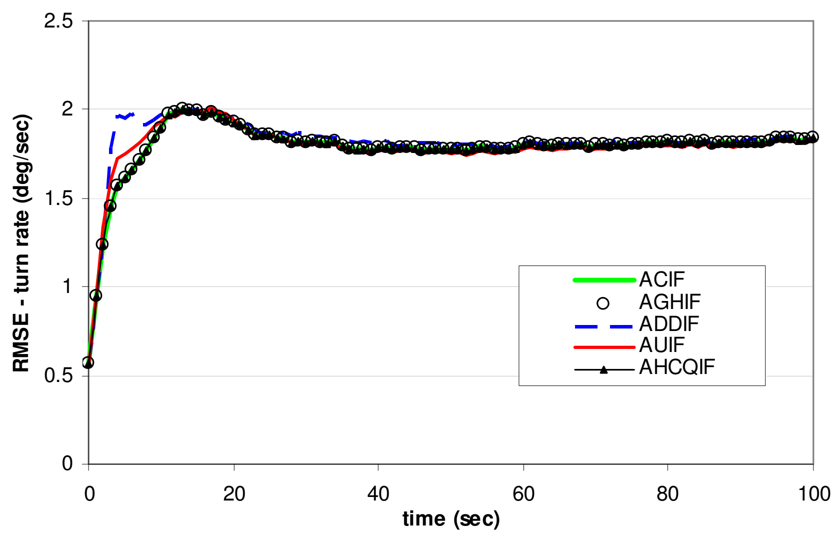


Fig. 9.20: Comparison of RMS error (turn rate estimation) of ACIF, AGHIF, ADDIF, AUIF, AHCQIF for 10000 MC runs

TABLE-9.7 : Comparative study of % -age of track loss for ***R*** adaptive information filters

Estimation algorithms	Percentage of track loss
ADDIF	5.56%
AUIF	6.10%
ACIF (5^{th} degree)	2.11%
AHCQIF (5^{th} degree)	2.26%
AGHIF (3^{rd} order)	2.17%

9.6.2.3. Demonstration of square root versions of *R* adaptive filters

The aircraft tracking problem is considered again to validate the square root versions of GHIF, CIF(3^{rd} degree), CQIF (3^{rd} degree) for ***R*** adaptation. The performance of the algorithms has been compared in the same vein as discussed before. It has been observed from Fig. 9.21-9.23 that the RMSE of RA-SR-CQIF(3^{rd} degree) and RA-SR-GHIF are comparably same and sometimes slightly less than RA-SR-CIF (3^{rd} degree). The track loss cases have been excluded from the RMSE.

The percentage of track loss for RA-SR-CIF (3^{rd} degree), RA-SR-CQIF(3^{rd} degree) and RA-SR-GHIF are 2.47%, 2.07% and 2.06% respectively. The percentage track loss for RA-SR-

CQIF(3rd degree) and RA-SR-GHIF, as expected, are comparable and low compared to that for RA-SR-CIF (3rd degree).

The number of points required for RA-SR-CQIF(3rd degree) is less compared to RA-SR-GHIF and computation cost for the former is also less as a consequence. For a single run an average computation time for RA-SR-CIF (3rd degree) and RA-SR-CQIF (3rd degree) are 9.83% and 4.48% of the average computation time for RA-SR-GHIF respectively. The simulations are carried out using MATLAB (version 7.9.0.529) in a computer with specifications Intel®, Core (TM) 2 Duo CPU, 2.8 GHz, 2 GB RAM.

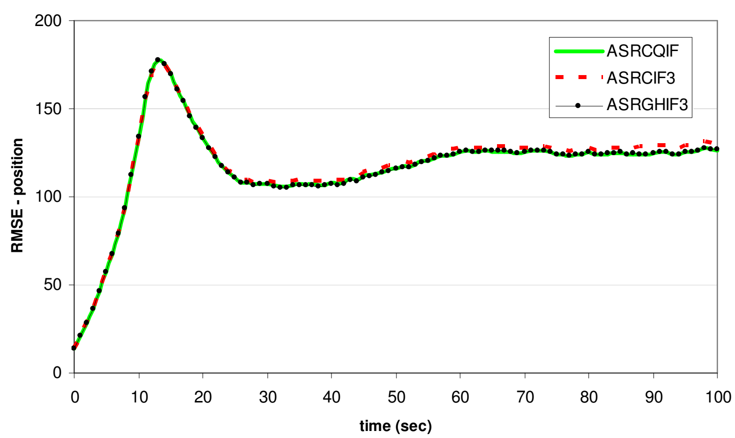


Fig. 9.21: Comparison of RMS error (position estimation) of ASRCQIF, ASRGHIF, ASRCIF for 10000 MC runs

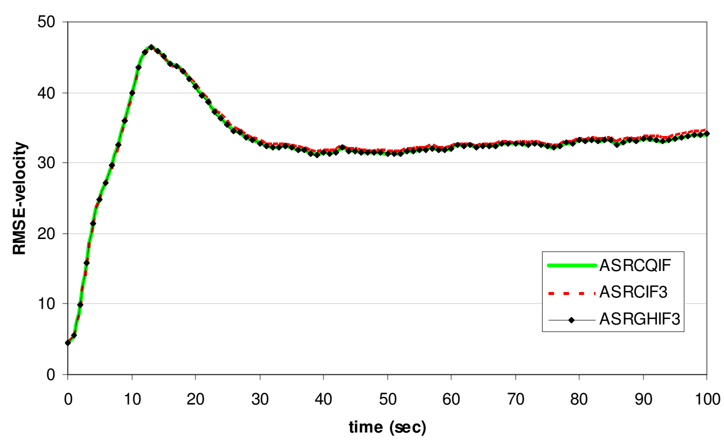


Fig. 9.22: Comparison of RMS error (velocity estimation) of ASRCQIF, ASRGHIF, ASRCIF for 10000 MC runs

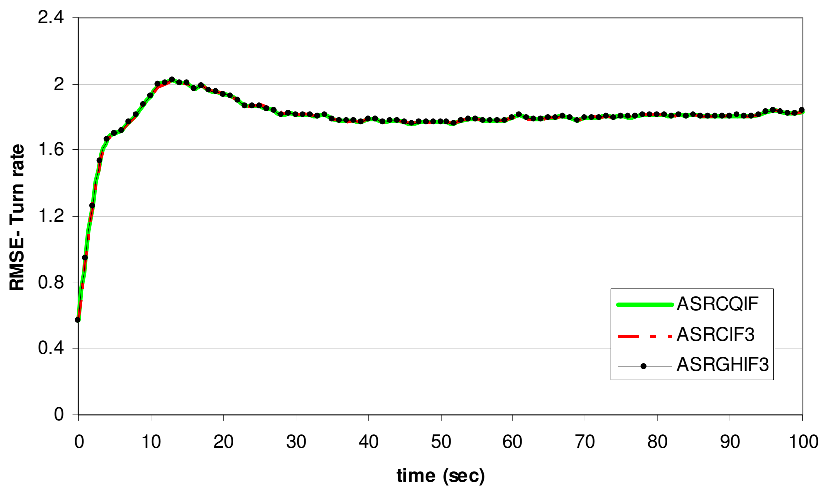


Fig. 9.23: Comparison of RMS error (turn rate estimation) of ASRCQIF, ASRGHIF, ASRCIF for 10000 MC runs

9.7 Discussions and Conclusions

In this chapter a class of $\boldsymbol{Q}$ and $\boldsymbol{R}$ adaptive sigma point information filters have been formulated from the proposed general algorithm and demonstrated with the help of case studies based on multi sensor estimation problems. The newly proposed adaptive nonlinear information filters are found to produce satisfactory estimation results in following contingent situations when (i) one or more sensor noise covariances are unknown, (ii) one of the sensors provides biased measurement, (iii) the system dynamics suffers from unknown parameter variation and the knowledge of process noise covariance remains incomplete as a consequence. A few significant findings have been enumerated below.

- The proposed $\boldsymbol{Q}$ adaptive nonlinear information filters are observed to present satisfactory estimation performance by online adaptation of $\boldsymbol{Q}$ where the complete knowledge of $\boldsymbol{Q}$ remains unavailable. For each of the proposed adaptive filters it is observed that the adapted value of the unknown element of $\boldsymbol{Q}$ converges on its truth value and subsequently tracks it.
- The results from Monte Carlo study demonstrate that the RMSE of each of the proposed filters settles down to a lower value compared to the respective non-adaptive counterpart. Another important finding for $\boldsymbol{Q}$ adaptive information filters is that for each filter the

RMSE is closely comparable with that of non-adaptive filters in ideal situation with full knowledge of ***Q***.

- The performance comparison of alternative ***Q*** adaptive information filters revealed that the estimation performance of QA-CQIF and QA-GHIF is superior to QA-DDIF and QA-CIF. Although the estimation performance of QA-CQIF and QA-GHIF are comparably same use of the former is preferable as it uses less number of sigma points and supposed to be computationally economic.
- Superiority of ***R*** adaptive DDIF is demonstrated over its non-adaptive versions in presence of bias in one of the measurements and also in face of unknown measurement noise covariance. Relative performance comparison of variants of ***R*** adaptive information filters has also been carried out. ***R*** adaptive versions of GHIF, CQIF (5th degree) and CIF (5th degree) demonstrate superiority over the competing algorithms of RA-DDIF and RA-UIF.
- For the aircraft tracking problem considered in this chapter performance of RA-GHIF, RA-CQIF (5th degree) and RA-CIF (5th degree) are found to be comparably same. Note that RA-CQIF (5th degree) and RA-CIF (5th degree) are computationally economic compared to RA-GHIF.
- The square root versions of ***R*** adaptive CQIF, CIF (3^{rd} degree) and GHIF have also been formulated and validated with the same tracking problem. Following the same trend RA-SR-GHIF and RA-SR-CQIF (3^{rd} degree) are found to outperform RA-SR-CIF (3^{rd} degree).

Considering the above findings adaptive nonlinear information filters are advocated for multiple sensor fusion because of their dual aspect of information filter configuration and adaptation performance.

Chapter 10: Conclusions

10.1 Concluding comments

The objective of the present work had been to develop improved estimation methods for state as well as parameter estimation of nonlinear systems. Towards this overall objective the present work focused on Adaptive state estimation for nonlinear signal models. The findings of this dissertation and the concluding comments are presented below.

1. Algorithms for a class of adaptive state estimators for plants with nonlinear dynamics have been proposed and their characteristics have been evaluated. Such a set of adaptive estimators include a fair number of adaptive filters, viz. Adaptive Divided Difference filters, Adaptive Gauss Hermite filters, Adaptive Cubature Kalman filters, Adaptive Cubature Quadrature Kalman filters.

2. The proposed nonlinear state estimators have been found to be superior to their corresponding non-adaptive versions in every case where process or measurement noise covariance remains unknown, demonstrating successful adaptation.

3. Regarding the performance of the proposed filter, the numerical simulations for all the case studies indicate that the performances of the adaptive filters without the knowledge of any one of the noise covariances (viz. $\boldsymbol{Q}$ or $\boldsymbol{R}$) are comparable to that of their respective non-adaptive nonlinear filter in the ideal situation where the noise covariances are accurately known. This indicates that the proposed adaptive nonlinear filters may be strong candidates for state estimation with nonlinear signal models in the face of unknown noise covariance.

4. A general framework for nonlinear state estimators had been proposed and it is demonstrated that algorithms for most of the proposed state estimators can be deduced from this general framework.

5. Regarding the $\boldsymbol{Q}$-adaptive Divided Difference filters the following specific concluding comments apply:

   - A new algorithm for $\boldsymbol{Q}$ adaptive second order Divided Difference filter developed for the joint estimation of parameters and states of nonlinear dynamic systems with unknown process noise covariance was found to outperform the non-adaptive version as evident from the Monte Carlo simulation of an object tracking problem.

- A variant of ***Q*** adaptive Divided Difference filter which is based on the method of automatic tuning of ***Q*** with the help of a scaling factor is also observed to be providing satisfactory estimation results for some specific estimation problems with linear measurement equations.
- Performance of the QA-DDF with direct adaptation algorithm was found to be superior compared to the scale factor based QA-DDF for the same object tracking problem where the measurement equation is nonlinear.

6. Regarding the ***R*-**adaptive Divided Difference filters the following specific concluding comments apply:

- The new algorithm for ***R*** adaptive Divided Difference filter for joint estimation of parameters and states of nonlinear signal models has been demonstrated in situations when prior knowledge of the measurement noise covariance remains unavailable.
- It has been theoretically established and also validated in simulation that the filter guarantees positive definiteness of adapted ***R*** matrix.
- It has been possible to demonstrate that the estimation performance of this filter is superior to non-adaptive filters even for the assumed initial values of ***R*** matrix with large errors.
- The estimation performance of direct ***R***-Adaptation is found to be superior to the scaling factor based ***R*** adaptation for some specific case studies.
- The ***R*** adaptive DDF (with direct ***R***-Adaptation) had been found to perform satisfactorily for unknown time varying ***R***. The adapted ***R*** adequately converged to the truth value of ***R*** and subsequently tracked the time varying truth value of ***R*** in all the cases considered in this case study.
- The ***R*** adaptive Central Difference filter which is a subset of the above algorithm is also validated. Although this algorithm cannot perform as well as ***R*** adaptive DDF on the ground of estimation accuracy it may be advocated for its economic computation and less complexity.

7. Regarding the joint estimation of states and parameters , the following specific concluding comments apply:
   - It is fairly well known that during the joint estimation of states and parameters, assigning appropriate process noise covariance corresponding to the parameters to be estimated requires substantial effort in tuning and experimentation. This is further exacerbated when such unknown parameters are time-varying. Assuming that the elements of the $\boldsymbol{Q}$ matrix corresponding to the states are known, one needs to adapt only the terms of $\boldsymbol{Q}$ which corresponds to the unknown parameters. It has been shown that the so-named partially $\boldsymbol{Q}$-adaptive filters may be employed for such cases.
8. Novel algorithms viz. $\boldsymbol{Q}$ and $\boldsymbol{R}$ adaptive Gauss Hermite filters have been developed by using Gauss Hermite quadrature rule for numerical approximation of the Bayesian Integrals (present in the general algorithm for nonlinear filter). A few variants of such filters have been developed which provide improved estimation performance over their non-adaptive versions (when evaluated with the help of benchmark estimation problems).Regarding adaptive Gauss Hermite state estimators/filters, the following specific concluding comments apply:
   - Algorithm for partially adaptive GHF have been formulated and it has been shown that such filters satisfactorily estimate and track unknown time varying parameters successfully.
   - Both innovation based and residual based $\boldsymbol{R}$-adaptive GHF are observed to provide improved estimation performance in the face of unknown $\boldsymbol{R}$. However, use of residual based $\boldsymbol{R}$ adaptation algorithm is recommended for the ensured positive definiteness of adapted $\boldsymbol{R}$ matrix.
   - For improved numerical accuracy and ensured positive definiteness of error covariance Square Root versions of $\boldsymbol{R}$ Adaptive GHF is formulated and validated.
9. Algorithms for adaptive nonlinear filters are also formulated using the Spherical Radial Cubature rule (third and fifth degree) and Spherical Radial Cubature Quadrature rule (third and fifth degree) for numerical evaluation of Bayesian integrals. Regarding such proposed adaptive versions of state estimators which are

based on recently proposed (non-adaptive) quadrature-cubature KF family, the following specific concluding comments apply:

- The third degree ***R***-adaptive Cubature filter (RA-CKF), as expected, is found to be superior to its non-adaptive version. However, RA-CKF (third degree) could not excel RA-UKF or RA-GHF with respect to performance accuracy.
- The ***R*** adaptive version of higher degree cubature filter (fifth degree accuracy) is found to be comparable in performance with ***R*** adaptive GHF (third order) and computationally economic compared to the latter. RA-CKF (fifth degree) is also performance wise superior to RA-UKF.
- The performance comparison of ***R*** adaptive Cubature Quadrature Kalman filter (3^{rd} degree) with the above adaptive filters revealed that this estimator outperforms many of the above cited estimators and its performance is found to be comparable with ***R*** adaptive GHF (fifth order) for some estimation problems. Although performance wise equivalent, the RA-CQKF requires less computational effort compared to RA-GHF (5^{th} order). It is observed from the case studies that RA-CQKF (5^{th} degree) can present marginally improved estimation performance compared to RA-CQKF (3^{rd} degree) and the other competing algorithms at the cost of additional computation effort.
- The square root version of ***R*** adaptive CQKF, which guarantees positive definiteness of error covariance, has also been developed and characterized.

10. Algorithms for adaptive nonlinear filters are also validated for non-additive noise models. ***Q*** and ***R*** adaptive DDF for non-additive noise are found to be demonstrably superior to the respective non-adaptive versions. ***R*** adaptive CKF (3^{rd} and 5^{th} degree) for non-additive measurement noise is derived from the general framework extended for non-additive noise and its performance is also compared with ***R*** adaptive DDF. Their performance has been found to be equivalent for the bearing only tracking problem.

    - In the case of non-additive noise models the tracking performance of the proposed ***R***-adaptive filters were found to be satisfactory whereas the same for the ***Q*** adaptive versions is not so.

11. Several new algorithms for ***Q*** and ***R*** adaptive nonlinear information filters are formulated and their use in multiple sensor fusion for nonlinear signal models has been demonstrated. The efficacy of these algorithms over their corresponding non-adaptive information filters are demonstrated in presence of parametric uncertainty, biased measurements or unavailability of the knowledge of any of the sensor noise covariance.

    - Like adaptive nonlinear filters their information filter configurations with inaccurately assumed value of noise covariance is comparably equal with that of their non-adaptive version in ideal situation when the non-adaptive version have the knowledge of the all the noise covariances. This has been demonstrated for the case of ***Q*** adaptive version of information filters.
    - Information filter versions viz. AGHIF (3^{rd} order), ACQIF (3^{rd} degree and 5^{th} degree) and ACIF (5^{th} degree) demonstrate an improved performance over the competing algorithms.
    - The square root versions of several nonlinear adaptive information filters viz. CQIF, CIF (3^{rd} degree) and GHIF (3^{rd} order) have also been formulated and characterized with the help of case studies..

10.2 Scope for future work

The possibilities for further work which have been recognized during the entire tenure of this research work are enumerated below:

- With the help of the general algorithm for adaptive nonlinear estimator new algorithms of adaptive nonlinear filters can be formulated considering non-adaptive versions of sparse grid Gauss Hermite filter, modified Cubature filter and Fourier Hermite filter as an underlying framework. It would be interesting to carry out a relative performance analysis of these estimators with the existing ones which have been presented in this dissertation.
- Like nonlinear filters nonlinear smoothers also require the knowledge of noise covariance for their satisfactory performance. The general algorithm may also be

extended for nonlinear smoother which may result a set of novel adaptive nonlinear smoothers.

- In this dissertation the noises are restricted to be Gaussian white noise. The redesigning of adaptation algorithms for the colour noise is yet to be explored.
- Moreover, there also remains a scope of investigation in formulating adaptive nonlinear filters for non Gaussian noise by extending the general algorithm for Gaussian sum noise.
- Nevertheless, the scope of mathematical analysis for the optimality and convergence of adaptive nonlinear filters always remains open as this work has not attempted in this dissertation as well as in the works reported in literature as of now.

Bibliography

[AkÇa2012] AkÇa, T., Demİrekler, M.: 'An adaptive unscented Kalman filter for tightly coupled INS/GPS integration', Proc. Position Location and Navigation Symposium (PLANS), IEEE, 2012, pp. 389-395

[Almagbile2010] Almagbile, A., Wang, J., Ding, W.: 'Evaluating the performances of adaptive Kalman filter methods in GPS/INS integration', Journal of Global Positioning Systems, 2010, **9**, (1), pp. 33-40

[Anderson1979] Anderson, B.D.O, Moore, J.B.: 'Optimal filtering', (Prentice-Hall, Englewood Cliffs, New Jersey, 1979, 1st edn.)

[Arasaratnam2007] Arasaratnam, I., Haykin, S., Elliott, R. J.: 'Discrete-time nonlinear filtering algorithms using Gauss–Hermite quadrature', 2007,. Proceedings of the IEEE, **95**, (5), pp. 953-977

[Arasaratnam2008] Arasaratnam, I., Haykin, S.: 'Square-root quadrature Kalman filtering', IEEE Trans. Signal Processing, 2008, **56**, (6), pp. 2589-93

[Arasaratnam2009] Arasaratnam I, Haykin S.: 'Cubature kalman filters',. IEEE Trans Automatic Control, 2009, **54**, (6), pp. 1254-1269

[Arasaratnam2011] Arasaratnam, I., Haykin, S.: 'Cubature Kalman smoothers', Automatica, 2011, **47**, (10), pp. 2245-2250

[Athans1968] Athans, M., Wishner, R., Bertolini, A.: 'Suboptimal state estimation for continuous-time nonlinear systems from discrete noisy measurements',. IEEE Trans. Automatic Control, 1968, **13**, (5), pp. 504-514

[BarShalom2001] Bar-Shalom, Y., Rong Li, X., Kirubarajan, T.: 'Estimation with Applications to Tracking and Navigation' (John Wiley & Sons, 2001, 1st edn.)

[Bavdekar2011] Bavdekar, V. A., Deshpande, A. P., Patwardhan, S. C.: 'Identification of process and measurement noise covariance for state and parameter estimation using extended Kalman filter', Journal of Process Control, 2011, **21**, (4), pp. 585-601

[Benzerrouk2013] Benzerrouk, H., Salhi, H., Nebylov, A.: 'Adaptive "Cubature and Sigma Points" Kalman Filtering Applied to MEMS IMU/GNSS Data Fusion during Measurement Outlier', Journal of Sensor Technology, 2013, **3**, (4), pp. 115-125

[Besançon2010] Besançon, G., Voda, A., Jouffroy, G.: 'A note on state and parameter estimation in a Van der Pol oscillator', Automatica, 2010, **46**, (10), pp. 1735-1738

[Bhaumik2013] Bhaumik, S., Swati: 'Cubature quadrature Kalman filter', IET Signal Processing, 2013, **7**, (7), pp. 533-541

[Brown1983] Brown, R. G.: 'Introduction to random signal analysis and Kalman filtering' (Vol. 8, Wiley, New York, 1983, 1st edn.).

[Busse2003] Busse, F.D., How, J.P., Simpson, J.: 'Demonstration of adaptive extended Kalman filter for low earth orbit formation estimation using CDGPS', Journal of Institute of. Navigation, 2003, **50**, (2), pp. 79–93

[Cao2009] Cao, Y., Tian, X.: 'An adaptive UKF algorithm for process fault prognostics', Proc. 2nd International Conference on Intelligent Computation Technology and Automation, IEEE, 2009, Vol. 2, pp. 487-490

[Carew1973] Carew, B., Belanger, P., 'Identification of optimum filter steady-state gain for systems with unknown noise covariances', IEEE Trans. Automatic Control, 1973, **18**, (6), pp. 582-587

[Cavusoglu2014] Cavusoglu, B., Oral, E. A.: 'Estimation of available bandwidth share by tracking unknown cross-traffic with adaptive extended Kalman filter', Computer Communications, 2014, **47**, pp. 34-50

[Chai2012] Chai, W., Chen, C., Edwan, E., Zhang, J., Loffeld, O.: 'INS/Wi-Fi based indoor navigation using adaptive Kalman filtering and vehicle constraints'. Proc. 9th IEEE Workshop on Positioning Navigation and Communication, Dresden, March, 2012, pp. 36-41

[Chalasani2012] Chalasani, G., Bhaumik, S.: 'Bearing only tracking using Gauss-Hermite filter', Proc. Int. Conf. Industrial Electronics and Applications, IEEE, Singapore, July 2012, pp. 1549-1554

[Challa1999] Challa, S., Bar-Shalom, Y., Krishnamurthy, V.: 'Nonlinear filtering via generalized edgeworth series and Gauss-Hermite quadrature', IEEE Trans. Signal Processing, 2000, **48**, (6), pp. 1816-1820

[Chandra2011] Chandra, K. P. B., Gu, D. W., Postlethwaite, I.: 'Cubature information filter and its applications', Proc. American Control Conference (ACC), IEEE, 2011, San Francisco, pp. 3609-3614

[Chandra2013] Chandra, K. P. B., Gu, D. W., Postlethwaite, I: 'Square root cubature information filter', IEEE Sensors Journal, 2013, **13**, (2), pp. 750-758

[Chen2012] Chen, H.: 'Adaptive cubature Kalman filter for nonlinear state and parameter estimation', Proc. 15th International Conference on Information Fusion (FUSION), IEEE, 2012, pp. 1413-1420

[Chen2013] Chen, Z., Dai, X., Jiang, L., Yang, C., Cai, B.: 'Adaptive iterated square-root cubature Kalman filter and its application to SLAM of a mobile robot', TELKOMNIKA Indonesian Journal of Electrical Engineering, 2013, **11**, (12), pp. 7213-7221

[Cheng2014] Cheng, J., Chen, D., Landry, J. R., Zhao, L., Guan, D.: 'An adaptive unscented kalman filtering algorithm for MEMS/GPS integrated navigation systems', Journal of Applied Mathematics, 2014, DOI:10.1155/2014/451939, pp. 1-8

[Das2013] Das, M., Sadhu, S., Ghoshal, T. K.: 'An Adaptive Sigma Point Filter for Nonlinear Filtering Problems', International Journal of Electrical, Electronics and Computer Engineering, 2013, **2**, (2), pp. 13-19

[Das2014] Das, M., Dey, A., Sadhu, S., Ghoshal, T.K.: 'Joint Estimation of States and Parameters of a Reentry Ballistic Target using Adaptive UKF', Proc. 5th IEEE International Symposium on Electronic System Design (ISED), Dec 2014, pp. 99-103

[Das2015] Das, M., Sadhu, S., Ghoshal, T. K.: 'Fault Detection and Isolation of Satellite Inertial Instrumentation Using Nonlinear Adaptive Filters', Journal of Aerospace Information Systems, 2015;**12**, (2), pp. 284-289

[Daum2005] Daum, F. 'Nonlinear filters: beyond the Kalman filter', IEEE Magazine Aerospace and Electronic Systems, 2005, **20**, (8), pp. 57-69

[Dempster1977] Dempster, A. P., Laird, N. M., Rubin, D. B.: 'Maximum likelihood from incomplete data via the EM algorithm', Journal of the Royal Statistical Society, 1977, Series B (methodological), B3, pp. 1-38

[Ding2007] Ding, W., Wang, J., Rizos, C., Kinlyside, D.: 'Improving adaptive Kalman estimation in GPS/INS integration', Journal of Navigation, 2007; **60**, (03), pp. 517-529

[El-Mowafy2005] El-Mowafy, A., Mohamed, A.: 'Attitude determination from GNSS using adaptive Kalman filtering', Journal of Navigation, 2005, **58**, (1), pp. 135-148

[Fathabadi2009] Fathabadi, V., Shahbazian, M., Salahshour, K., Jargani, L.: 'Comparison of adaptive Kalman filter methods in state estimation of a nonlinear system using asynchronous measurements', Proc. World Congress on Engineering and Computer Science, 2009, Vol. 2

[Gao2011] Gao, X., Chen, J., Tao, D., Li, X.: 'Multi-sensor centralized fusion without measurement noise covariance by variational Bayesian approximation', IEEE Trans. Aerospace and Electronic Systems, 2011, **47**, (1), pp.718-722

[Gao2015a] Gao, W., Li, J., Zhou, G., Li, Q.: 'Adaptive Kalman Filtering with Recursive Noise Estimator for Integrated SINS/DVL Systems', Journal of Navigation, 2015, **68**, (01), pp. 142-161

[Gao2015b] Gao, S., Hu, G., Zhong, Y.: 'Windowing and random weighting based adaptive unscented Kalman filter', International Journal of Adaptive Control and Signal Processing, 2015, **29**, (2), pp. 201-23

[Ge2011] Ge, Q. B., Li, W. B., Wen, C. L.: 'SCKF-STF-CN: a universal nonlinear filter for maneuver target tracking', Journal of SCIENCE, Zhejiang University, 2011, **12**, (8), pp. 678-686

[Ge2014] Ge. Q., Wen, C., Chen, S., Sun, R., Li, Y.: 'Adaptive cubature strong tracking information filter using variational Bayesian method', Proc.

19th World Congress of the International Federation of Automatic Control, Cape Town, South Africa, 2014, pp. 24-29

[Geist2010] Geist, M.: 'Statistical linearization for value function approximation in reinforcement learning', Proc. NIPS Workshop on Learning and Planning from Batch Time Series Data, 2010, pp. 1-6

[Genz2003] Genz, A.: 'Fully symmetric interpolatory rules for multiple integrals over hyper-spherical surfaces', Journal of Computational and Applied Mathematics, 2003, **157**, (1), pp. 187–195

[Georges2015] Georges, H. M., Wang, D., Xiao, Z.: 'GNSS/Low-Cost MEMS-INS integration using variational Bayesian adaptive cubature Kalman smoother and ensemble regularized ELM', Mathematical Problems in Engineering, 2015, Article ID: 682907, DOI: http://dx.doi.org/10.1155/2015/682907

[Golub1969] Golub, G. H., Welsch, J. H.: 'Calculation of Gauss quadrature rules', Mathematics of Computation, 1969; **23**, (106), pp. 221-230

[Hajiyev2011] Hajiyev, C.: 'Nonlinear robust adaptive EKF for identification of EMAs parameters in the presence of sensor faults', Proc. 5th International Conference on Physics and Control(PHYSCON), Leon, Spain, September, 2011

[Hajiyev2013a] Hajiyev, C., Soken, H. E.: 'Robust adaptive Kalman filter for estimation of UAV dynamics in the presence of sensor/actuator faults,' Aerospace Science and Technology, 2013, **28**, (1), pp.376-383

[Hajiyev2014] Hajiyev, C., Soken, H. E.: 'Robust adaptive unscented Kalman filter for attitude estimation of pico satellites', International Journal of Adaptive Control and Signal Processing, 2014, **28**, (2), pp. 107-120

[Han2009a] Han, J., Kim, D., Sunwoo, M.: 'State-of-charge estimation of lead-acid batteries using an adaptive extended Kalman filter', Journal of Power Sources, 2009, **188**, (2), pp. 606-612

[Han2009b] Han, J., Song, Q., He, Y.: 'Adaptive unscented Kalman filter and its applications in nonlinear control', in Victor M. Moreno, Alberto Pigazo (Eds.): 'Kalman Filter: Recent Advances and Applications' (Tech, 2009, 1st edn.), pp. 1-24

[Haug2005] Haug ,A. J.: 'A tutorial on Bayesian estimation and tracking techniques applicable to nonlinear and non-Gaussian processes',. MITRE Corporation, McLean. Virginia, Contract No.: W15P7T-04-D199, Project No.: 01MSR0115RT, 2005 Jan, pp.1-55

[He2011] He, H., Xiong, R., Zhang, X., Sun, F., Fan, J.: 'State-of-charge estimation of the lithium-ion battery using an adaptive extended Kalman filter based on an improved Thevenin model', IEEE Trans. Vehicular Technology, 2011, **60**, (4), pp. 1461-1469

[He2015] He, H., Liu, Z., Hua, Y.: 'Adaptive extended Kalman filter based fault detection and isolation for a lithium-ion battery pack', Energy Procedia, 2015, **75**, pp. 1950-1955

[Hide2003a] Hide, C., 'Integration of GPS and low cost INS measurements:' Ph.D. dissertation, University of Nottingham, September 2003.

[Hide2003b] Hide, C., Moore, T., Smith, M.: 'Adaptive Kalman filtering for low-cost INS/GPS', 2003, The Journal of Navigation, **56**, (1), pp. 143-152

[Hide2004] Hide, C., Moore, T., Smith, M.: 'Adaptive Kalman filtering algorithms for integrating GPS and low cost INS', Proc. IEEE Position Location and Navigation Symposium, 2004, pp. 227-233

[Huy2012] Huy, L., Matunaga, S.: 'Fine-tuning Kalman filters using star trackers data for micro satellite attitude estimation,' Proc. International symposium on Artificial Intelligence, Robotics and Automation in Space, European Space Agency, 2012

[Hwang2010] Hwang, I., Kim, S., Kim, Y., Seah, C. E.: 'A survey of fault detection, isolation, and reconfiguration methods', IEEE Trans Control Systems Technology, 2010, **18**, (3), pp. 636-53

[Ito20000] Ito, K., Xiong, K.: 'Gaussian filters for nonlinear filtering problems', IEEE Trans Automatic Control, 2000, **45**, (5), pp. 910-927

[Jargani2009] Jargani, L., Shahbazian, M., Salashoor, K., Fathabadi, V.: 'Performance monitoring of nonlinear CSTR using novel adaptive unscented Kalman filter', Proc. World Congress on engineering and Computer Science, 2009, Vol. 2, pp. 20-22

[Jia2012] Jia, B., Xin, M., Cheng, Y.: 'Sparse-grid quadrature nonlinear filtering', Automatica, 2012, **48**, (2), pp. 327-341

[Jia2013a] Jia, B., Xin, M., Cheng, Y.: 'High-degree cubature Kalman filter', Automatica, 2013, **49**, (2), pp. 510-518

[Jia2013b] Jia, B., Xin, M., Pham, K., Blasch, E., Chen, G.: 'Multiple sensor estimation using a high-degree cubature information filter', Proc. SPIE 8739, Sensors and Systems for Space Applications VI, Baltimore, Maryland, USA, April 2013, pp. 87390T-87390T, doi: 10.1117/12.2015546

[Jiancheng2011] Jiancheng, F., Sheng, Y.: 'Study on innovation adaptive EKF for in-flight alignment of airborne POS', IEEE Trans. Instrumentation and Measurement, 2011, **60**, (4), pp. 1378-1388

[Jiang2007] Jiang, Z., Song, Q., He, Y., Han, J.: 'A novel adaptive unscented Kalman filter for nonlinear estimation', Proc. 46th IEEE Conference on Decision and Control, 2007, pp. 4293-4298

[Julier1995] Julier, S. J., Uhlmann, J. K., Durrant-Whyte, H. F.: 'A new approach for filtering nonlinear systems', Proc. American Control Conference (IEEE), 1995, Vol. 3, pp. 1628-1632

[Julier1996] Julier, S. J., Uhlmann, J. K.: 'A general method for approximating nonlinear transformations of probability distributions', Technical report, Robotics Research Group, Department of Engineering Science, University of Oxford, 1996.

[Julier1997] Julier, S. J., Uhlmann, J. K.: 'New extension of the Kalman filter to nonlinear systems', Proc. 11th International Symposium on AeroSense/Defence Sensing Simulation and Control, Oralando, Florida, USA, 1997, Vol. Multi Sensor Fusion, pp. 182-193

[Julier2000] Julier, S. J., Uhlmann, J. K., Durrant-Whyte, H. F.: 'A new method for the nonlinear transformation of means and covariances in filters and estimators', IEEE Trans Automatic Control, 2000, **45**, (3), pp. 477-482

[Julier2002a] Julier, S. J., Uhlmann, J. K.: 'The scaled unscented transformation,' in Proc. American Control Conference, 2002, pp. 4555–4559

[Julier2002b] Julier, S. J., Uhlmann, J. K.: 'Reduced sigma point filters for the propagation of means and covariances through nonlinear transformations', in Proc. American. Control Conference, 2002, pp. 887–892

[Julier2004] Julier, S. J., Uhlmann, J. K.: 'Unscented filtering and nonlinear estimation', Proceedings of the IEEE, 2004, **92**, (3), pp. 401-422

[Jwo2008] Jwo, D.J., Weng, T.P.: 'An adaptive sensor fusion method with applications in integrated navigation', Journal of Navigation, 2008, **61**, (4), pp. 705-721

[Jwo2009] Jwo, D.J., Chen, M. Y., Tseng, C. H., Cho, T. S.: 'Adaptive and nonlinear Kalman filtering for GPS navigation processing', Victor M. Moreno, Alberto Pigazo (Eds.): 'Kalman Filter: Recent Advances and Applications' (Tech, 2009), pp. 321-346

[Kandepu2008] Kandepu, R., Foss, B., Imsland, L.: 'Applying the unscented Kalman filter for nonlinear state estimation', Journal of Process Control, 2008, **18**, (7), pp. 753-768

[Karlgaard2010] Karlgaard, C. D.: 'Robust adaptive estimation for autonomous rendezvous in elliptical orbit'. PhD thesis, Virginia Polytechnic Institute and State University, 2010

[Kashyap1970] Kashyap, R. L.: 'Maximum likelihood identification of stochastic linear systems', 1970, IEEE Trans. Automatic Control, **15**, (1), pp. 25-34

[Khaleghi2013] Khaleghi, B., Khamis, A., Karray, F. O., Razavi, S. N.: 'Multisensor data fusion: A review of the state-of-the-art', Information Fusion, 2013, **14**, (1), pp. 28-44

[Lee2004] Lee, D. J., Alfriend, K. T.: 'Adaptive sigma point filtering for state and parameter estimation', Proc. AIAA/AAS Astrodynamics specialist conference and exhibit, Rhode Island, 2004, pp. 1-20

[Lee2005] Lee, D. J.: 'Nonlinear Bayesian filtering with applications to estimation and navigation'. PhD thesis, Texas A&M University, 2005

[Lee2008] Lee, D. J.: 'Nonlinear estimation and multiple sensor fusion using unscented information filtering', IEEE Signal Processing Letters, 2008; **15**, pp. 861-864

[Lefebvre 2002] Lefebvre, T., Bruyninckx, H., Schutter, J. De.: 'Kalman filters for non-linear systems: a comparison of performance', International journal of Control, 2004, **77**, (7), pp. 639-53

[Li2010] Li, W., Ge, Q.: 'UKF-STF tracking with correlated noises for the nonlinear system', Proc. 8th World Congress on Intelligent Control and Automation, IEEE, 2010, pp. 3466-3471

[Lippiello2007] Lippiello, V., Siciliano, B., Villani, L.: 'Adaptive extended Kalman filtering for visual motion estimation of 3D objects', Control Engineering Practice, 2007, **15**, (1), pp. 123-134

[Liqiang2015] Liqiang, Z., Jianlin, W., Tao, Y., Huan, J., Tangjiang, L.: 'Design of adaptive robust square-root cubature Kalman filter with noise statistic estimator', Applied Mathematics and Computation, 2015, **256**, pp. 352-67

[Liu2009] Liu, J., Lu, M.: 'An adaptive UKF filtering algorithm for GPS position estimation', Proc. 5th International Conference on Wireless Communications, Networking and Mobile Computing, IEEE, 2009, pp. 1-4

[Liu2011] Liu, G., Worgotter, F., Markelic, I.: 'Nonlinear estimation using central difference information filter', Proc. IEEE Workshop on Statistical Signal Processing (SSP), pp. 593-596, 2011

[Liu2012] Liu, G.: 'Bayes Filters with Improved Measurements for Visual Object Tracking', Doctoral dissertation, Göttingen State and University, 2012

[Maybeck1981] Maybeck, P. S., Jensen, R. L., Harnly, D. A.: 'An adaptive extended Kalman filter for target image tracking', 1981, IEEE Trans Aerospace and Electronic Systems, 2, pp. 173-180

[Maybeck1982] Maybeck, P.S.: 'Stochastic models, estimation, and control', in Bellman, R. (Ed.):'Mathematics in science and engineering' (Academic Press, New York, 1982, Vol. 141(2), 1st edn.)

[Mehra1970] Mehra, R. K.: 'On the identification of variances and adaptive Kalman filtering', IEEE Trans Automatic Control, 1970, **15**, (2), pp. 175-84

[Mehra1971] Mehra, R. K.: 'On-line identification of linear dynamic systems with applications to Kalman filtering', IEEE Trans Automatic Control, 1971, **16**, (1), pp.12-21

[Mehra1972] Mehra, R. K.: 'Approaches to adaptive filtering', IEEE Trans Automatic Control, 1972, **17**, (5), pp. 693-698

[Meng2000] Meng, Q. H., Sun, Y.C., Cao, Z. L.: 'Adaptive extended Kalman filter (AEKF)-based mobile robot localization using sonar', Robotica, 2000, **18**, (5), pp. 459-473

[Merwe2001] Merwe, Van der R., Wan, E. A.: 'The Square-Root Unscented Kalman Filter for State and Parameter-Estimation', Proc. International conference on Acoustics, Speech, and Signal Processing (ICASSP), Salt Lake City, Utah, 2001

[Merwe2004] Van Der Merwe R.: 'Sigma-point Kalman filters for probabilistic inference in dynamic state-space models', Doctoral dissertation, Oregon Health & Science University, 2004

[Mohamed1999] Mohamed, A. H.: 'Optimizing the Estimation Procedure in INS/GPS Integration for Kinematic Applications', Doctoral dissertation, University of Calgary, 1999

[MohanM.2015] Mohan M, S., Naik, N., Gemson, R. M. O., Ananthasayanam, M. R.: 'Introduction to the Kalman filter and tuning its statistics for near optimal estimates and Cramer Rao bound', Technical Report, TR/EE2015/401, Deptt. of Electrical Engineering, Indian Institute of Technology Kanpur, February 2015

[Myers1976] Myers, K., Tapley, B. D.: 'Adaptive sequential estimation with unknown noise statistics', 1976, IEEE Trans. Automatic Control, **21**, (4), pp. 520-523

[Narasimhappa2012] Narasimhappa, M., Rangababu, P., Sabat, S. L., Nayak, J.: 'A modified Sage-Husa adaptive Kalman Filter for denoising fiber optic gyroscope signal', Proc. IEEE India Conference (INDICON), 2012, pp. 1266-1271

[Neethling1974] Neethling, C., Young. P.: 'Comments on "Identification of optimum filter steady state gain for systems with unknown noise covariances"', IEEE Trans. Automatic Control, 1974, **19**, (5), pp. 623-625

[Nørgaard2000] Nørgaard, M., Poulsen, N. K., Ravn, O.: 'New developments in state estimation for nonlinear systems', Automatica, 2000, **36**, (11), pp. 1627-1638

[Odelson2006] Odelson, B. J., Lutz, A., Rawlings, J. B.: 'The autocovariance-least squares method for estimating covariances: application to model-based control of chemical reactors', IEEE Trans. on Control System Technology, 2006, **14**, (3), pp. 532-540

[Oussalah2000] Oussalah, M., Schutter, J.: 'Adaptive Kalman Filter for Noise Identification', Proc. International Conference on Noise and Vibration Engineering, ISMA25, 2000, pp. 1225-1232

[Ragi2013] Ragi, S., Chong, E. K.: 'Decentralized control of unmanned aerial vehicles for multitarget tracking', Proc. IEEE International Conference on Unmanned Aircraft Systems, 2013, pp. 260-268

[Ristic2003] Ristic, B., Farina, A., Benvenuti, D., Arulampalam, M. S.: 'Performance bounds and comparison of nonlinear filters for tracking a ballistic object on re-entry', IEE Proceedings on Radar, Sonar and Navigation, IET, 2003, **150**, (2), pp. 65-70

[Ristic2004] Ristic, B., Arulampalam, Gordon, S., N.: 'Beyond the Kalman Filter', (Aryech House, Boston, London, 2004, 1st edn).

[Sadhu2007] Sadhu, S., Srinivasan, M., Bhaumik, S., Ghoshal, T. K.: 'Central difference formulation of risk-sensitive filter', IEEE Letters on Signal Processing, 2007, **14**, (6), pp. 421-424

[Sadhu2006] Sadhu, S., Mondal, S., Srinivasan, M., Ghoshal, T., K.: 'Sigma point Kalman filter for bearing only tracking', Signal processing, 2006, **86**, (12), pp. 3769-3777

[Sage1969] Sage, A. P., Husa, G. W.: 'Algorithms for sequential adaptive estimation of prior statistics', In Proc. IEEE Symposium on Adaptive Processes (8th) Decision and Control, 1969, Vol. 8, pp. 6a1-6a10

[Sarkka2009] Sarkka, S., Nummenmaa, A.: 'Recursive noise adaptive Kalman filtering by variational Bayesian approximations', 2009, IEEE Trans. Automatic Control, **54**, (3), pp. 596-600

[Sarkka2013a] Sarkka, Simo.: 'Bayesian filtering and smoothing' (Vol. 3. IMS Textbooks series, ISBN 9781107619289, Cambridge University Press, 2013, 1st edn.)

[Sarkka2013b] Sarkka, S., Hartikainen, J: 'Non-linear noise adaptive Kalman filtering via variational Bayes', Proc. IEEE International Workshop on Machine Learning for Signal Processing, 2013, pp. 1-6

[Sarmavuori2012] Sarmavuori, J., Sarkka, S.: 'Fourier-Hermite Kalman filter', IEEE Trans. Automatic Control, 2012, **57**, pp. 1511–1515

[Schei1997] T. S. Schei, 'A finite-difference method for linearization in nonlinear estimation algorithms', Automatica, 1997, **33**, pp. 2053-2058

[Shi2008] Shi, Y., Han, C., Lian, F.: 'The iterated Divided Difference filter', Proc., IEEE International Conference on Automation and Logistics, 2008. pp. 1799-1802

[Shumway2000] Shumway, R. H., Stoffer, D. S.: 'Time Series Analysis and its Applications' (Springer, Verlag, New York, 2000)

[Simandl2009] Simandl, M., Dunik, J.: 'Derivative-free estimation methods: New results and performance analysis', Automatica, 2009, **45**, (7), pp. 1749-1757

[Simon2006] Simon D.: 'Optimal State Estimation: Kalman, H Infinity, and Nonlinear Approaches'(John Wilely & Sons, New Jersey, 2006, 1st edn.)

[Singh2012] Singh, N.K: 'Tracking of ballistic target using Gauss-Hermite and Ensemble Kalman filter', Master's Thesis, Jadavpur University, India, 2012

[Singh2013] Singh, A. K., Bhaumik, S.: 'Nonlinear estimation using transformed Gauss-Hermite quadrature points', Proc. IEEE International Conference on Signal Processing, Computing and Control (ISPCC), 2013, pp. 1-4

[Singh2015] Singh, A. K., Bhaumik, S.: 'Higher degree cubature quadrature kalman filter', International Journal of Control, Automation and Systems, 2015, **13**, (5), pp. 1097-105

[Soken2009] Soken, H. E., Hajiyev, C.: 'Adaptive unscented Kalman filter with multiple fading factors for pico satellite attitude estimation', Proc. International Conference on Recent Advances in Space Technologies, IEEE, 2009, pp. 541-546

[Soken2011] Soken, H. E., Hajiyev, C.: 'A novel adaptive unscented Kalman filter for pico satellite attitude estimation', Proc. 5th International Conference on Physics and Control (PHYSCON), Leon, Spain, September, 2011

[Soken2012] Soken, H. E., Hajiyev, C.: 'UKF-based reconfigurable attitude parameters estimation and magnetometer calibration', IEEE Trans. Aerospace and Electronic Systems, 2012, **48**, (3), pp. 2614-2627

[Soken2014] Soken, H. E., Sakai, S. I.: 'Adaptive Tuning of the UKF for Satellite Attitude Estimation', Journal of Aerospace Engineering, 2014, **28**, (3), 04014088, DOI: 10.1061/(ASCE)AS.1943-5525.0000412

[Subrahmanya2009] Subrahmanya, N., Shin, Y. C.: 'Adaptive divided difference filtering for simultaneous state and parameter estimation', Automatica, 2009, **45**, (7), pp. 1686-1693

[Sun2012] Sun, J., Zhou, J., Gu, X.: 'Variational Bayesian two-stage Kalman filter for systems with unknown inputs', Procedia Engineering, 2012, **29**, pp.2265-2273

[Tang2012] Tang, X., Wei, J., Chen, K.: 'Square-root adaptive cubature Kalman filter with application to spacecraft attitude estimation', Proc. 15th International Conference on Information Fusion (FUSION), IEEE, 2012, pp. 1406-1412

[Tao2014] Tao, W., Xian-Feng, T., Quan-Bo, G.: 'Multi-sensor fusion based on unscented strong tracking information filter', Proc. 26th Control and Decision Conference, IEEE, 2014, pp. 370-374

[Vercauteren2005] Vercauteren, T, Wang, X.: ‘Decentralized sigma-point information filters for target tracking in collaborative sensor networks’, IEEE Trans. Signal Processing, 2005, **53**, (8), pp. 2997-3009

[Wan2000] Wan, E. A., R., Merwe, V., D., 2000: ‘The unscented Kalman filter for nonlinear estimation’, Proc. IEEE Symposium (AS-SPCC), 2000, Lake Louise, Alberta, Canada, pp. 153-158.

[Wang2012] Wang, X., Liang, Y., Pan, Q., Yang, F.: ‘A Gaussian approximation recursive filter for nonlinear systems with correlated noises’, Automatica, 2012, **48**, (9), pp. 2290-2297

[Wang2015] Wang, X., Wu, L., Yi, X., Zhang, Y., Zhang, L., Zhao, H., Gao, F.: ‘Performance enhancement of pharmacokinetic diffuse fluorescence tomography by use of adaptive extended Kalman filtering’, Computational and Mathematical methods in Medicine, 2015, Article ID 739459, DOI: http://dx.doi.org/10.1155/2015/739459

[Wanxin2011] Wanxin, S.: ‘Application of improved UKF algorithm in initial alignment of SINS’, Proc. 2nd International Conference on Artificial Intelligence, Management Science and Electronic Commerce (AIMSEC), IEEE, 2011, pp. 187-190

[Whyte2008] Durrant-Whyte, H., Henderson, T. C.: ‘Multisensor data fusion’, ‘Springer Handbook of Robotics’ (Springer Berlin Heidelberg, 2008) , pp. 585-610

[Wu2006] Wu, Y., Hu, D., Wu, M., Hu, X.: ‘A numerical-integration perspective on Gaussian filters’, IEEE Trans. on Signal Processing, 2006, **54**, (8), pp. 2910-2921

[Xia2014] Xia, C., Fu, J., Xu, Y., Chen, Z.: ‘Machine tool selected point temperature rise identification based on operational thermal modal analysis’, International Journal of Advanced Manufacturing Technology, 2014, **70**, (4), pp. 19-31

[Xia2015] Xia, B., Wang, H., Tian, Y., Wang, M., Sun, W., Xu, Z.: ‘State of charge estimation of lithium-ion batteries using an adaptive cubature Kalman filter’, Energies, 2015, **8**, (6), pp. 5916-5936

[Xie1999] Xie, X. Q., Zhou, D. H., Jin, Y. H.: ‘Strong tracking filter based adaptive generic model control’, Journal of Process Control, 1999, **9**, (4), pp. 337-350

[Yang2001a] Yang, Y., He, H., Xu, G.: ‘Adaptively robust filtering for kinematic geodetic positioning’, Journal of Geodesy, 2001, **75**, (2), pp. 109–116

[Yang2001b] Yang, Y., Xu, T.H., He, H.: ‘On adaptively kinematic filtering’,. Selected Papers for English of Acta Geodetica et Cartographica Sinica, 2001, pp. 25-32

[Yang2003] Yang, Y., Xu, T.: 'An adaptive Kalman filter based on Sage windowing weights and variance components', The Journal of Navigation, 2003, **56**, (2), pp. 231-40

[Yang2005] Yang, Y., Gao, W.: 'Comparison of adaptive factors in Kalman filters on navigation results', Journal of Navigation, 2005, **58**, (3), pp. 471-478

[Yu2014] Yu, F., Sun, Q., Lv, C., Ben Y, Fu, Y.: 'A SLAM Algorithm Based on Adaptive Cubature Kalman Filter', Mathematical Problems in Engineering, 2014, Article ID 171958, DOI: http://dx.doi.org/10.1155/2014/171

[Zagrobelny2014] Zagrobelny, M., A., Rawlings, J. B.: 'Identification of Disturbance Covariances Using Maximum Likelihood Estimation,' Technical Report, TR No. 2014-02, Department of Chemical and Biological Engineering, University of Wisconsin-Madison, 2014.

[Zarchan2000] Zarchan, P., Musoff, H.: 'Fundamentals of Kalman Filtering' (American Institute of Aeronautics and Astronautics, Reston,VA, 2000)

[Zarei2014] Zarei, J., Shokri, E.: 'Convergence analysis of non-linear filtering based on cubature Kalman filter', IET Science, Measurement & Technology, 2014, **9**, (3), pp. 294-305

[Zeng2012] Zeng, N., Wang, Z., Li, Y., Du, M., Liu, X.: 'A hybrid EKF and switching PSO algorithm for joint state and parameter estimation of lateral flow immunoassay models', IEEE/ACM Trans Computational Biology and Bioinformatics, 2012, **9**, (2), pp. 321-329

[Zhang2014] Zhang, X. C., Teng, Y. L.: 'A new derivation of the Cubature Kalman filters', Asian Journal of Control, 2014, **16**, (5), pp.1501-1510

[Zhao2009] Zhao, L., Wang, X.: 'An adaptive UKF with noise statistic estimator', Proc. 4th IEEE Conference on Industrial Electronics and Applications, 2009, pp. 614-618

[Zhou1996] Zhou, D.H., Frank, P. H.: 'Strong tracking Kalman filtering of nonlinear time-varying stochastic systems with coloured noise: application to parameter estimation and empirical robustness analysis,' International Journal of Control, 1996, **65**, (2), pp. 295-307