VARIATIONAL SMOOTHING FOR EXTENDED TARGET TRACKING WITH RANDOM MATRICES

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

BY

SAVAŞ ERDEM KARTAL

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

APRIL 2022

Approval of the thesis:

VARIATIONAL SMOOTHING FOR EXTENDED TARGET TRACKING WITH RANDOM MATRICES

submitted by **SAVAŞ ERDEM KARTAL** in partial fulfillment of the requirements for the degree of **Master of Science in Electrical and Electronics Engineering Department, Middle East Technical University** by,

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

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

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

Examining Committee Members:

Prof. Dr. Kemal Leblebicioğlu
Electrical and Electronics Engineering, METU _____

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

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

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

Assist. Prof. Dr. Gökhan Soysal
Electrical and Electronics Engineering, Ankara University _____

Date: 05.04.2022

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

Name, Surname: Savaş Erdem Kartal

Signature :

ABSTRACT

VARIATIONAL SMOOTHING FOR EXTENDED TARGET TRACKING WITH RANDOM MATRICES

Kartal, Savaş Erdem
M.S., Department of Electrical and Electronics Engineering
Supervisor: Prof. Dr. Umut Orguner

April 2022, 83 pages

In this thesis, two Bayesian smoothers are proposed for random matrix based extended target tracking (ETT). The proposed smoothers are based on the variational Bayes techniques and they are derived for an extended target model without and with orientation. The random matrix models of Feldman *et al.* and Tuncer and Özkan are used as the extended target models without and with orientation, respectively. The performance of both smoothers is evaluated using simulation results on two different scenarios. It is seen that the variational smoothers derived for both models outperform the previous smoother recently given in the literature on the scenario with a non-maneuvering target. On the other hand, it is seen that the performance of the smoother for the model without orientation is reduced significantly below expectations on the scenario with maneuvers. Nevertheless, the smoother for the model with orientation is shown to have little performance degradation in the maneuvering scenario. Overall the results obtained in this thesis show that:

- the variational approach results in better smoothers than the existing smoother in the literature,

- the explicit modeling of orientation is beneficial in smoothing as well as filtering for tracking maneuvering extended targets.

Keywords: Extended target tracking, smoother, variational Bayes, random matrices, target extension, target orientation

ÖZ

RASTLANTISAL MATRİSLER İLE BÜYÜK HEDEF TAKİBİ İÇİN VARYASYONEL DÜZGÜNLEŞTİRME

Kartal, Savaş Erdem
Yüksek Lisans, Elektrik ve Elektronik Mühendisliği Bölümü
Tez Yöneticisi: Prof. Dr. Umut Orguner

Nisan 2022 , 83 sayfa

Bu tezde, rastlantısal matris tabanlı büyük hedef takibi (BHT) için iki Bayesyen düzgünleştirici sunulmuştur. Sunulan düzgünleştiriciler varyasyonel Bayes tekniği tabanlıdır ve oryantasyonlu ve oryantasyonsuz büyük hedef takip modelleri için türetilmiştir. Feldmann *vd.*'nin ve Tuncer ve Özkan'ın rastlantısal matris modelleri sırasıyla oryantasyonsuz ve oryantasyonlu büyük hedef modelleri olarak kullanılmıştır. Her düzgünleştiricinin performansı iki ayrı senaryo üzerinde yapılan simülasyonlar ile değerlendirilmiştir. Her iki model için türetilen varyasyonel düzgünleştiricilerin, manevra yapmayan hedef senaryosunda literatürde yakın zamanda verilen önceki düzgünleştiriciden daha iyi performans gösterdiği görülmektedir. Öte yandan oryantasyonsuz modeli kullanan düzgünleştiricinin performansının manevralı senaryoda beklentilerin oldukça altında kaldığı gözlemlenmiştir. Bununla birlikte, oryantasyonlu modeli kullanan düzgünleştiricinin performansı manevralı senaryoda çok az düşüş göstermiştir. Genel olarak tezde elde edilen sonuçlar göstermektedir ki:

- varyasyonel yaklaşım kullanılarak türetilen düzgünleştirici literatürdeki düzgünleştiriciden daha iyi sonuç vermektedir,

- belirgin oryantasyon modellemesi manevra yapan büyük hedefleri izlemek için filtreleme kadar düzgünleştirmede de faydalıdır.

Anahtar Kelimeler: Büyük hedef takibi, düzgünleştirici, varyasyonel Bayes, rastlantısal matris, hedef uzantısı, hedef oryantasyonu

To my family and beloved friends...

ACKNOWLEDGMENTS

I would like to express my sincere gratitude to my supervisor Prof. Umut Orguner for his expert guidance throughout my thesis study. This study could not have been possible without his support and I am grateful to be able to work with him.

I owe a debt of thanks to my girlfriend Melisa İnan for all her love and support in the difficult times I have been through during this study. I must also express my special acknowledgments to my family and friends who have been with me at every moment I need them.

TABLE OF CONTENTS

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

LIST OF ALGORITHMS

ALGORITHMS

LIST OF ABBREVIATIONS

ETT	Extended Target Tracking
RM	Random Matrix
IW	Inverse Wishart
IG	Inverse Gamma
RTS	Rauch–Tung–Striebel
RHS	Right Hand Side
KL	Kullback-Leibler
MC	Monte Carlo
2D	2 Dimensional
GW	Gaussian Wasserstein
RMSE	Root Mean Square Error
SPD	Symmetric Positive Definite
VS	Variational ETT smoother without orientation proposed in this work
VSO	Variational ETT smoother with orientation proposed in this work
FFK	Feldmann *et al.*'s ETT filter [1]
GB	Granström and Bramstång's ETT smoother [2]
TO	Tuncer and Özkan's variational ETT filter [3]

Notations

x_k	Kinematic state vector
$\hat{x}_{k\|k}$	Estimated kinematic state mean
$\hat{x}_{k\|k-1}$	Predicted kinematic state mean

$\hat{x}_{k\|K}$	Smoothed kinematic state mean
$P_{k\|k}$	Estimated kinematic state covariance matrix
$P_{k\|k-1}$	Predicted kinematic state covariance matrix
$P_{k\|K}$	Smoothed kinematic state covariance matrix
x_x	Position in x direction
x_y	Position in y direction
v_x	Velocity in x direction
v_y	Velocity in y direction
X_k	Extent matrix
$X_{k\|k}$	Estimated extent
$X_{k\|k-1}$	Predicted extent
$X_{k\|K}$	Smoothed extent
$\nu_{k\|k}$	Estimated degrees of freedom of extent
$\nu_{k\|k-1}$	Predicted degrees of freedom of extent
$\nu_{k\|K}$	Smoothed degrees of freedom of extent
$V_{k\|k}$	Estimated scale matrix of extent
$V_{k\|k-1}$	Predicted scale matrix of extent
$V_{k\|K}$	Smoothed scale matrix of extent
θ_k	Orientation angle
$\hat{\theta}_{k\|k}$	Estimated orientation angle mean
$\hat{\theta}_{k\|k-1}$	Predicted orientation angle mean
$\hat{\theta}_{k\|K}$	Smoothed orientation angle mean
$\Theta_{k\|k}$	Estimated orientation angle covariance
$\Theta_{k\|k-1}$	Predicted orientation angle covariance
$\Theta_{k\|K}$	Smoothed orientation angle covariance
T_{θ_k}	Rotation matrix
Λ_k	Axis length matrix

λ_k	Axis length
$\alpha_{k\|k}$	Estimated shape parameter of axis length
$\alpha_{k\|k-1}$	Predicted shape parameter of axis length
$\alpha_{k\|K}$	Smoothed shape parameter of axis length
$\beta_{k\|k}$	Estimated scale parameter of axis length
$\beta_{k\|k-1}$	Predicted scale parameter of axis length
$\beta_{k\|K}$	Smoothed scale parameter of axis length
GW_x	Position Gaussian Wasserstein distance
GW_Λ	Axis Length Gaussian Wasserstein distance
GW_X	Extent Gaussian Wasserstein distance
GW_1	Overall Gaussian Wasserstein distance with axis length
GW_2	Overall Gaussian Wasserstein distance with extent
RMSE_θ	Orientation angle root-mean-square-error
y_k	Measurement vector
$\mathcal{Y}_k$	Set of measurements
$\overline{y}_k$	Mean of measurements
$\overline{Y}_k$	Spread of measurements
m_k	Number of measurements
w_k	Measurement noise
z_k	Noise-free measurement vector
$\mathcal{Z}_k$	Noise-free set of measurements
$K_{k\|k-1}$	Kalman gain
$S_{k\|k-1}$	Innovation variance
$N_{k\|k-1}$	Innovation matrix
τ	Time constant
Δt_k	Time interval
t_k	Time step

T	Sample time
d	Object dimension
r	Kinematic state model dimension
A	State transition matrix
B	Process noise gain matrix
G	Smoother gain matrix
H	Measurement matrix
R	Measurement noise covariance
Q	Process noise covariance
M	Object extension temporal evolution matrix
v	Process noise
s	Scale factor
ψ	Forgetting factor

CHAPTER 1

INTRODUCTION

The conventional target tracking applications usually consider the objects to be observed as point sources. However, due to the recent increase in sensor resolutions, the objects might give rise to multiple distinct detections in varying numbers. For example, short-range applications such as surveillance, autonomous weapons, autonomous driving, or robotics often consider the object of interest as extended and such applications require extended target tracking (ETT). With the integration of high-resolution sensors into an increasing number of application areas, the ETT has become an important and emerging research field recently.

In extended target tracking applications, in addition to the kinematic state, the object extension should also be considered as an "internal degree of freedom" characterizing an extended object. Therefore, the object extension needs to be included in the object state and has to be estimated jointly with the kinematic state. The combined state leads to a nonlinear estimation problem whose solution requires some approximations and nonlinear estimation techniques. Studies in the literature can be classified by different models used to represent the shape of the object extension, measurements, and object dynamics.

The object extension can be modeled in different shapes. A rectangular shape representing the object extension is used in [4] and [5]. Modeling the extension by an ellipsoidal shape is another approach that is widely used because ellipse shape is usually sufficient for the measurement modeling, see, e.g., [6–8]. There are also more complex arbitrary shape models for the scenarios where the object extension can not be modeled by a geometric shape, see, e.g., [9], [10]. However, in the scope of this thesis, the ellipsoidal object extension model will be adopted.

The measurement models differ in their assumptions about the number of detections and the position on the object on which these detections are generated. Early examples assume a measurement model which describes the detections as reflections from fixed sources on the object [11], [12]. This approach is also used in [13–15] for the modeling of the reflection points on cars. The accuracy of this approach decreases as the uncertainty of the number and location of the detections increases. Gilholm *et al.* proposed another approach in [16] and [17], that models the measurements spatially distributed over the target. In this model, the number of measurements is Poisson distributed.

The nonlinear nature of the extended target tracking requires nonlinear estimation techniques. The Extended Kalman Filter [18] can be used where nonlinearities are weak but the performance of this algorithm is reduced significantly as the nonlinearities get higher. An alternative method based on particle filtering, also known as the Monte Carlo method, is proposed in [19] where posterior distribution of the target state is represented by a set of particles; each of which has an importance weight assigned to it. Instead of propagating densities with prediction and update equations, these particles and weights are propagated to obtain an approximation to the target's posterior distribution. Other examples utilizing this approach can be found in [20] and [21].

One of the popular approaches of extended target tracking is the random hypersurface model developed by Baum and Hanebeck in [9] and [22], that models the object extensions in star-convex shapes. The algorithm they developed is capable of tracking arbitrary shaped extended targets by using a parametric representation of the shape contour.

A more recent model is proposed by Wahlström and Özkan in [23], where they use Gaussian processes to learn the shape of the object. Their model is capable of estimating a variety of object shapes. This model is also used later in [24] and [25].

Another well-known approach to extent modeling and estimation is the random matrix (RM) model. This model, first proposed by Koch in [6], is an example of spatially distributed measurement models. It uses symmetric positive definite random matrices to model the object extension and thus assumes that the object has an ellipsoidal

shape. The measurements are assumed to be spatially distributed over the object with a Gaussian distribution whose covariance is the object extension. Koch's model is then improved by Feldmann *et al.* in [1, 26, 27], where the sensor error is also considered as a parameter affecting measurement distribution. In [2], Granström and Bramstång present Bayesian smoothing algorithms for both Koch's and Feldmann *et al.*'s models. The approximations of Feldmann *et al.*'s measurement update generate the questions of optimality. In [28], Orguner derives an analytical measurement update using variational inference to solve this. The variational Bayes technique is later used in [3] to track the target's heading angle explicitly.

In the context of this thesis, the random matrix model will be used. Therefore, the rest of the document will be limited to the scope of this model. For a more detailed overview of current research in extended target tracking, including different aspects of modeling approaches, [29] can be studied. Moreover, an elaborate comparison of random matrix and random hypersurface models is given in [30] by showing the different assumptions and properties of the two methods.

The previous studies show that the smoothing algorithms show better performance than the filtering algorithms. Moreover, the variational Bayes technique also improves the estimation performance. None of the studies in the literature combines the power of smoothers with the variational Bayes technique and this lead us to study on such an algorithm. In substance, this thesis study proposes two variational smoothing algorithms for the random matrix based extended target tracking applications under measurement noise. Using the analytical techniques of variational Bayes inference, this work presents approximate smoothing algorithms for the extended target tracking models with and without considering the orientation angle of the target. The performance of the proposed algorithms is validated with two different simulation studies.

1.1 Organization of the Thesis

This thesis is organized as follows. In Chapter 2, we first provide an overview of the previous research on RM based ETT by focusing on the filtering and smoothing algorithms in the literature and elaborating on some studies which form the basis

of this study. Then, the variational inference technique is explained and the brief information about the studies using this technique in the literature is given. In Chapter 3, we propose our first smoothing algorithm which is based on an ETT model without orientation. The derivation steps are provided in detail and the pseudo-code for the implementation of the algorithm is given. At the end of this chapter, we explain why we need another algorithm that treats the orientation as a separate random variable by providing some reasoning on the initial simulation results. In Chapter 4, we propose another algorithm that is based on an ETT model with orientation. The derivation steps and pseudo-code of this algorithm are given in this chapter. In Chapter 5, the performance of the proposed algorithms is compared with the other filtering and smoothing algorithms in the literature on two different ETT simulation scenarios. The comments on the results and a comparison table for the computation time of the algorithms is provided at the end of the chapter. Finally, in Chapter 6, the thesis is concluded with a summary of our study and potential future work.

CHAPTER 2

BACKGROUND

The objective of this chapter is to provide a review of the studies in the literature on random matrix based extended target tracking. First, filtering and smoothing algorithms based on the random matrix model will be investigated. Then, necessary information about the variational inference will be given and some of the studies which use this method for extended target tracking will be reviewed.

2.1 Extended Target Tracking using Random Matrices

The random matrix model, which is realized within the Bayesian Framework, is a well-known approach in extended target tracking. The approach is based on augmenting the kinematic state vector x_k at each time t_k, which represents the target's position, velocity, and in some cases acceleration, by the additional state matrix X_k representing target's extension.

The rest of this section is organised to introduce different filtering and smoothing algorithms based on the random matrix model. The studies in [1–3], [6], [28], which form the basis of our study, will be explained in detail.

2.1.1 Filtering Algorithms

The random matrix model, which is proposed originally by Koch in [6] and later developed by Feldmann *et al.* in [1], is highly promising and thus has been a pioneer for many other filtering algorithms. In this subsection, we will provide information about the filtering algorithms that have been leveraged for the derivation of the smoothers

proposed in our study.

2.1.1.1 Koch's Model

The method proposed by Koch in [6] estimates both the kinematic and the extent states of the target recursively. The target extent is modeled as an ellipsoidal object which is described by a symmetric positive definite (SPD) random matrix X_k to be estimated from the sensor measurements. The target extent matrix is distributed with an inverse Wishart density (see [31]).

Within the Bayesian Framework, the extended object tracking algorithm is a recursive estimation scheme for the conditional probability density $p(x_k, X_k|\mathcal{Y}_{0:k})$ of the object state (x_k, X_k) at each time t_k using the accumulated measurements $\mathcal{Y}_{0:k}$ up to and including time t_k, dynamic model describing the temporal evolution of the objects, and sensor model parameters.

The dimension of the kinematic state vector x_k is rd, where d is the object dimension and $r-1$ defines up to which derivative the kinematic state is modeled. For example, the kinematic state x_k containing position and velocity can be shown as

$$x_k = [x_x, x_y, v_x, v_y]^\mathsf{T}, \tag{2.1}$$

in two spatial dimensions.
We assume that there are m_k conditionally independent position measurements at scan k belonging to the target given as

$$y_k^j = Hx_k + w_k^j \quad \text{for} \quad j = 0, ..., m_k. \tag{2.2}$$

In (2.2), H represents the measurement matrix which extracts the position information from the kinematic vector x_k. It can be expressed as

$$H = [I_d, 0_d] = \tilde{H} \otimes I_d, \tag{2.3}$$

where $\tilde{H} = [1, 0]$, d is the object dimension; i.e., $d = 2$ for 2D position tracking, and $\otimes$ is the Kronecker product. The measurement noise w_k^j is normally distributed with mean of zero and covariance of X_k. These individual measurements form the set of measurements $\mathcal{Y}_k = \{y_k^j\}_{j=1}^{m_k}$ at each scan k, where m_k is the number of measurements. The accumulated measurement sequence up to time k are denoted as $\mathcal{Y}_{0:k}$.

Since this approach assumes the statistical sensor error to be negligible, extension becomes the only factor affecting the spread of the measurements. Thus, the measurement noise can be written as a zero mean normally distributed random vector with covariance X_k. The likelihood of the measurement set $\mathcal{Y}_k$ given the number of measurements, kinematic state and extension becomes

$$p(\mathcal{Y}_k|m_k, x_k, X_k) = \prod_{j=1}^{m_k} \mathcal{N}(y_k^j; Hx_k, X_k), \tag{2.4}$$

where $\mathcal{N}(x; \mu, \Sigma)$ denotes a normal density with mean μ and variance Σ. The mean and spread of the measurements can be written as

$$\overline{y}_k = \frac{1}{m_k} \sum_{j=1}^{m_k} y_k^j \quad \text{and} \quad \overline{Y}_k = \sum_{j=1}^{m_k} (y_k^j - \overline{y}_k)(y_k^j - \overline{y}_k)^\intercal, \tag{2.5}$$

respectively. The joint density in (2.4) can be factorized as

$$p(\mathcal{Y}_k|m_k, x_k, X_k) \propto \mathcal{N}\left(\overline{y}_k; Hx_k, \frac{X_k}{m_k}\right) \mathcal{W}(\overline{Y}_k; m_k - 1, X_k), \tag{2.6}$$

where $\mathcal{W}(X; \nu, V)$ denotes the Wishart density of a $d \times d$ SPD random matrix X, with a scalar degrees of freedom ν, a $d \times d$ SPD scale matrix V, and a normalizing constant Z, which is defined as follows [31].

$$\mathcal{W}(X; \nu, V) = \frac{1}{Z} |V|^{-(1/2)\nu} |X|^{(\nu-d-1)/2} \text{etr}\left[\frac{1}{2} V^{-1} X\right], \quad \nu \geqslant d \tag{2.7}$$

where etr(.) is used for $\exp(\text{tr}(\cdot))$. The expected value of X is given as

$$\mathbb{E}[X] = \nu V. \tag{2.8}$$

The Bayes update equation for the joint density (x_k, X_k) is given as

$$p(x_k, X_k|\mathcal{Y}_{0:k}) = \frac{p(\mathcal{Y}_k|m_k, x_k, X_k) p(x_{k-1}, X_{k-1}|\mathcal{Y}_{0:k-1})}{\int p(\mathcal{Y}_k|m_k, x_k, X_k) p(x_{k-1}, X_{k-1}|\mathcal{Y}_{0:k-1}) dx_k dX_k}. \tag{2.9}$$

By exploiting the Bayes formula in (2.9), we can obtain that the posterior density $p(x_k, X_k|\mathcal{Y}_{0:k})$ is proportional to the product of the likelihood $p(\mathcal{Y}_k|m_k, x_k, X_k)$ and the prior density $p(x_{k-1}, X_{k-1}|\mathcal{Y}_{0:k-1})$. The prior density can be factored as

$$p(x_{k-1}, X_{k-1}|\mathcal{Y}_{0:k-1}) = p(x_{k-1}|X_{k-1}, \mathcal{Y}_{0:k-1}) p(X_{k-1}|\mathcal{Y}_{0:k-1}), \tag{2.10}$$

where

$$p(x_{k-1}|X_{k-1}, \mathcal{Y}_{0:k-1}) = \mathcal{N}(x_k, \hat{x}_{k|k-1}, P_{k|k-1} \otimes X_k), \tag{2.11}$$

$$p(X_{k-1}|\mathcal{Y}_{0:k-1}) = \mathcal{IW}(X_k; \nu_{k|k-1}, V_{k|k-1}). \tag{2.12}$$

Here $\hat{x}_{k|k-1}$ and $P_{k|k-1}$ are the kinematic state mean vector and covariance matrix, respectively. $\mathcal{IW}(X;\nu,V)$ denotes the inverse Wishart density of a $d\times d$ SPD random matrix X, with a scalar degrees of freedom ν, a $d \times d$ SPD scale matrix V, and a normalizing constant Z, which is defined as [31]

$$\mathcal{IW}(X;\nu,V) = \frac{1}{Z}|V|^{(1/2)(\nu-d-1)}|X|^{-(1/2)\nu}\text{etr}\left[-\frac{1}{2}VX^{-1}\right]. \tag{2.13}$$

The expected value of $X \sim \mathcal{IW}(X;\nu,V)$ is given as

$$\mathbb{E}[X] = \frac{V}{\nu-2d-2}, \qquad \nu-2d-2>0. \tag{2.14}$$

We can write the product of the measurement likelihood and the prior density as

$$\begin{aligned} p(\mathcal{Y}_k|m_k,x_k,X_k)p(x_k,X_k,\mathcal{Y}_{0:k-1}) \propto \mathcal{N}&\left(\overline{y}_k; Hx_k, \frac{X_k}{m_k}\right) \\ &\times \mathcal{N}(x_k;\hat{x}_{k|k-1},\tilde{P}_{k|k-1}\otimes X_k) \\ &\times \mathcal{W}(\overline{Y}_k; m_k-1, X_k) \\ &\times \mathcal{IW}(X_k;\nu_{k|k-1},V_{k|k-1}). \end{aligned} \tag{2.15}$$

The kinematical part $\mathcal{N}(x_k,\hat{x}_{k|k},\tilde{P}_{k|k}\otimes X_k)$ can be extracted from the product of Gaussians on the RHS of (2.15) as

$$\begin{aligned} \mathcal{N}\left(\overline{y}_k; Hx_k, \frac{X_k}{m_k}\right)\mathcal{N}(x_k;\hat{x}_{k|k-1},\tilde{P}_{k|k-1}\otimes X_k) =& \mathcal{N}(x_k,\hat{x}_{k|k},\tilde{P}_{k|k}\otimes X_k) \\ &\times \mathcal{N}(\overline{y}_k; H\hat{x}_{k|k-1}, \tilde{S}_{k|k-1}X_k). \end{aligned} \tag{2.16}$$

The parameters $\hat{x}_{k|k}$ and $\tilde{P}_{k|k}$ are given as

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + (\tilde{K}_{k|k-1}\otimes I_d)(\overline{y}_k - H\hat{x}_{k|k-1}), \tag{2.17a}$$

$$\tilde{P}_{k|k} = \tilde{P}_{k|k-1} - \tilde{K}_{k|k-1}\tilde{S}_{k|k-1}\tilde{K}_{k|k-1}^{\mathsf{T}}, \tag{2.17b}$$

where $\tilde{S}_{k|k-1}$ is the scalar innovation variance

$$\tilde{S}_{k|k-1} = \tilde{H}\tilde{P}_{k|k-1}\tilde{H}^{\mathsf{T}} + \frac{1}{m_k}, \tag{2.18}$$

and $\tilde{K}_{k|k-1}$ is the gain

$$\tilde{K}_{k|k-1} = \tilde{P}_{k|k-1}\tilde{H}^{\mathsf{T}}\tilde{S}_{k|k-1}^{-1}. \tag{2.19}$$

In order to obtain the update equation for the extent, we need to calculate the product of factors depending on the extent X_k on the RHS of (2.15). First, using the form of the Gaussian distribution, the second term on the RHS in (2.16) can be written as

$$\mathcal{N}(\overline{y}_k; H\hat{x}_{k|k-1}, \tilde{S}_{k|k-1}X_k) \propto |X_k|^{-1/2}\text{etr}\left[-\frac{1}{2}N_{k|k-1}X^{-1}\right], \tag{2.20}$$

where $N_{k|k-1}$ is being an innovation matrix defined as

$$N_{k|k-1} = (\overline{y}_k - H\hat{x}_{k|k-1})(\overline{y}_k - H\hat{x}_{k|k-1})^{\mathsf{T}}. \tag{2.21}$$

Then, combining (2.20) with the product of the Wishart and inverse Wishart distribution in (2.15) we get

$$\begin{aligned}\mathcal{IW}(X_k; \nu_{k|k}, V_{k|k}) \propto &\mathcal{W}(\overline{Y}_k; m_k - 1, X_k)\mathcal{IW}(X_k; \nu_{k|k-1}, V_{k|k-1}) \\ &\times |X_k|^{-1/2}\text{etr}\left[-\frac{1}{2}N_{k|k-1}X^{-1}\right],\end{aligned} \tag{2.22}$$

with the parameters $\nu_{k|k}$ and $V_{k|k}$ that can be written as

$$V_{k|k} = V_{k|k-1} + \tilde{S}^{-1}_{k|k-1}N_{k|k-1} + \overline{Y}_k, \tag{2.23a}$$

$$\nu_{k|k} = \nu_{k|k-1} + m_k. \tag{2.23b}$$

As a result of the measurement update, we get the posterior density of the joint state (x_k, X_k) as

$$p(x_k, X_k|\mathcal{Y}_{0:k}) = p(x_k|X_k, \mathcal{Y}_{0:k})p(X_k|\mathcal{Y}_{0:k}). \tag{2.24}$$

The next step of the recursive Bayesian estimation, called prediction, requires the joint probability density $p(x_k, X_k|\mathcal{Y}_{0:k})$ to be updated in time using the underlying model of the system.

$$p(x_{k-1}, X_{k-1}|\mathcal{Y}_{0:k-1}) \xrightarrow[\text{models}]{\text{evolution}} p(x_k, X_k|\mathcal{Y}_{0:k-1}). \tag{2.25}$$

By using Markov-type assumptions, the underlying evolution model of the system, represented by the joint transition density $p(x_k, X_k|x_{k-1}, X_{k-1}, \mathcal{Y}_{0:k-1})$ can be written as

$$\begin{aligned}p(x_k, X_k|x_{k-1}, X_{k-1}, \mathcal{Y}_{0:k-1}) &= p(x_k, X_k|x_{k-1}, X_{k-1}) \\ &= p(x_k|X_k, x_{k-1})p(X_k|X_{k-1}).\end{aligned} \tag{2.26}$$

In (2.26), it is also assumed that the kinematic state has no effect on the time update of the object extension. The temporal evolution of the kinematic state can be modeled as

$$x_k = Ax_{k-1} + v_k, \tag{2.27}$$

where v is the process noise with $p(v_k) = \mathcal{N}(v_k; 0, \tilde{Q}_k \otimes X_k)$, and $A = \tilde{A} \otimes I_d$ is the state transition matrix.

When we integrate the product of this joint transition density and the posterior distribution of the previous measurement update step, we get the Chapman-Kolmogorov equation

$$\begin{aligned} p(x_k, X_k|\mathcal{Y}_{0:k-1}) &= \int p(x_k, X_k|x_{k-1}, X_{k-1}, \mathcal{Y}_{0:k-1}) \\ &\quad \times p(x_{k-1}, X_{k-1}|\mathcal{Y}_{0:k-1}) dx_{k-1} dX_{k-1}, \\ &= \int p(x_k|X_k, x_{k-1}) p(X_k|X_{k-1}) \\ &\quad \times p(x_{k-1}|X_{k-1}, \mathcal{Y}_{0:k-1}) p(X_{k-1}|\mathcal{Y}_{0:k-1}) dx_{k-1} dX_{k-1}. \end{aligned} \tag{2.28}$$

Assuming that the kinematical part of the filtered distribution at time t_{k-1} is a Gaussian and the temporal evolution of the object extension has no effect on the prediction of this part, we get

$$p(x_{k-1}|X_{k-1}, \mathcal{Y}_{0:k-1}) \approx p(x_{k-1}|X_k, \mathcal{Y}_{0:k-1}), \tag{2.29a}$$

$$p(x_{k-1}|X_k, \mathcal{Y}_{0:k-1}) = \mathcal{N}(x_{k-1}; \hat{x}_{k-1|k-1}, \tilde{P}_{k-1|k-1} \otimes X_k). \tag{2.29b}$$

Substituting (2.29b) into (2.28), the predicted density of the kinematical part becomes

$$\begin{aligned} p(x_k|X_k, \mathcal{Y}_{0:k-1}) &= \int \mathcal{N}(x_k; Ax_{k-1}, \tilde{Q} \otimes X_k) \\ &\quad \times \mathcal{N}(x_{k-1}; \hat{x}_{k-1|k-1}, \tilde{P}_{k-1|k-1} \otimes X_k) dx_{k-1}, \qquad (2.30a) \\ &= \mathcal{N}(x_k; \hat{x}_{k|k-1}, \tilde{P}_{k|k-1} \otimes X_k), \qquad (2.30b) \end{aligned}$$

where the parameters $\hat{x}_{k|k-1}$ and $\tilde{P}_{k|k-1}$ are given as

$$\hat{x}_{k|k-1} = A\hat{x}_{k-1|k-1}, \tag{2.31a}$$

$$\tilde{P}_{k|k-1} = \tilde{A}\tilde{P}_{k-1|k-1}\tilde{A}^{\mathsf{T}} + \tilde{Q}. \tag{2.31b}$$

For the object extension part, let us assume that the filtered distribution at time t_{k-1} is inverse Wishart and it is given by

$$p(X_{k-1}|\mathcal{Y}_{0:k-1}) = \mathcal{IW}(X_{k-1}; \nu_{k-1|k-1}, V_{k-1|k-1}), \tag{2.32a}$$

$$\propto |X_{k-1}|^{-(1/2)\nu_{k-1|k-1}} \text{etr}\left[-\frac{1}{2}V_{k-1|k-1}X_{k-1}^{-1}\right]. \tag{2.32b}$$

Assuming that the expectation of the previous filtered density is equal to the expectation of the filtered density, the following equality holds.

$$\frac{V_{k|k-1}}{\nu_{k|k-1} - 2d - 2} = \frac{V_{k-1|k-1}}{\nu_{k-1|k-1} - 2d - 2}. \tag{2.33}$$

The degrees of freedom ν determines the covariance of the inverse Wishart distribution, so we can expect it to decrease with increasing sampling periods $\Delta t_k = t_k - t_{k-1}$. The following prediction update equations consider this effect and also keep the expectation the same.

$$\nu_{k|k-1} = \exp\left(-\frac{\Delta t_k}{\tau}\right)\nu_{k-1|k-1}, \tag{2.34a}$$

$$V_{k|k-1} = \frac{\nu_{k|k-1} - 2d - 2}{\nu_{k-1|k-1} - 2d - 2}V_{k-1|k-1}. \tag{2.34b}$$

2.1.1.2 Feldmann *et al.*'s Model

The Bayesian algorithm derived in [6] and described in the previous subsection is capable of estimating both the kinematic and extension states of an object. However, it assumes that the object extension dominates the spread of the measurements and thus ignores the effect of measurement noise. In presence of sensor error, this approach would estimate extension and measurement noise together and thus the estimated target extent becomes biased by the measurement noise covariance. In cases where the sensor error is comparable with the object extension, the estimation performance of this approach will be degraded.

In order to compensate for this effect, the measurement likelihood function is adjusted in [1]. The updated likelihood function taking the possible sensor errors into account can be written as

$$p(\mathcal{Y}_k|m_k, x_k, X_k) = \prod_{j=1}^{m_k} \mathcal{N}(y_k^j; Hx_k, sX_k + R), \tag{2.35}$$

where R is the sensor error covariance matrix and s is the scaling factor of the object extension. Including the effect of the scale factor enables us to model the spread of the measurements in a better way.

For the likelihood in (2.35), no analytical solution can be found for the Bayes update equation because of the additional R. Therefore, approximations are needed.

In [1], the predicted object extension $\hat{x}_{k|k-1}$ is assumed to be significantly close to true extent X_k. Also, the kinematic state density is approximated such that it is no longer conditioned on the object extension. Thus, we can write the joint state density as

$$p(x_k, X_k|\mathcal{Y}_{0:k}) \approx p(x_k|\mathcal{Y}_{0:k})p(X_k|\mathcal{Y}_{0:k}). \tag{2.36}$$

Assuming that the prior distribution of the kinematic state is normal, i.e.,

$$p(x_k|\mathcal{Y}_{0:k-1}) = \mathcal{N}(x_k; \hat{x}_{k|k-1}, P_{k|k-1}), \tag{2.37}$$

we can use standard Kalman filter update equations to obtain the posterior density

$$p(x_k|\mathcal{Y}_{0:k}) = \mathcal{N}(x_k; \hat{x}_{k|k}, P_{k|k}). \tag{2.38}$$

The update equations for the parameters $\hat{x}_{k|k}$ and $P_{k|k}$ of the posterior are given as

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_{k|k-1}(\overline{y}_k - H\hat{x}_{k|k-1}), \tag{2.39a}$$

$$P_{k|k} = P_{k|k-1} - K_{k|k-1}S_{k|k-1}K_{k|k-1}^{\mathsf{T}}, \tag{2.39b}$$

where $S_{k|k-1}$ is an approximation of the innovation covariance

$$S_{k|k-1} = HP_{k|k-1}H^{\mathsf{T}} + \frac{sX_{k|k-1} + R}{m_k}, \tag{2.40}$$

and $K_{k|k-1}$ is the gain

$$K_{k|k-1} = P_{k|k-1}H^{\mathsf{T}}S_{k|k-1}^{-1}. \tag{2.41}$$

The sensor error becomes an effective parameter of the kinematic state measurement update equations due to the modified version of the innovation covariance in (2.40).

In the measurement update of the object extension part in [1], the extent $X_{k|k}$, which is the expected value of the inverse Wishart posterior distribution, is estimated instead of the scale matrix $V_{k|k}$. Also, $\alpha_{k|k} = \nu_{k|k} - 2d - 2$ is updated instead of the degrees

of freedom $\nu_{k|k}$ for easier calculation. The resulting measurement update equations of the object extension are given as

$$X_{k|k} = \frac{1}{\alpha_{k|k}}(\alpha_{k|k-1}X_{k|k-1} + \hat{N}_{k|k-1} + \hat{Y}_{k|k-1}), \tag{2.42a}$$

$$\alpha_{k|k} = \alpha_{k|k-1} + m_k, \tag{2.42b}$$

where the parameters $\hat{N}_{k|k-1}$ and $\hat{Y}_{k|k-1}$ are given by

$$\hat{N}_{k|k-1} = X_{k|k-1}^{1/2} S_{k|k-1}^{-1/2} N_{k|k-1} (S_{k|k-1}^{-1/2})^{\mathsf{T}} (X_{k|k-1}^{1/2})^{\mathsf{T}}, \tag{2.43a}$$

$$\hat{Y}_{k|k-1} = X_{k|k-1}^{1/2} (sX_{k|k-1} + R)^{-1/2} \overline{Y}_{k|k-1} ((sX_{k|k-1} + R)^{-1/2})^{\mathsf{T}} (X_{k|k-1}^{1/2})^{\mathsf{T}}. \tag{2.43b}$$

The resulting posterior density is in the form of an inverse Wishart distribution, i.e.,

$$p(X_k|\mathcal{Y}_{0:k}) \approx \mathcal{IW}(X_k; \nu_{k|k}, \alpha_{k|k}X_{k|k}). \tag{2.44}$$

Similar to [6], Kalman filter prediction equations are used to update the kinematic state in time.

$$\hat{x}_{k|k-1} = A\hat{x}_{k-1|k-1}, \tag{2.45a}$$

$$P_{k|k-1} = AP_{k-1|k-1}A^{\mathsf{T}} + Q. \tag{2.45b}$$

Notice that the dimensions of the matrices used in (2.31b) and (2.45b) are different. In (2.31b), $r \times r$ state covariance $\tilde{P}_{k|k-1}$, process noise variance $\tilde{Q}_{k|k-1}$ and the state transition model $\tilde{A}$ matrices are used. However, in (2.45b) these matrices are used in full dimension $n_x \times n_x$.

The expected value of the object extension is assumed to remain constant during the time update.

$$X_{k|k-1} = X_{k-1|k-1}. \tag{2.46}$$

Similar to (2.34a), the time update equation of the degrees of freedom parameter is exponentially decreasing over time, i.e.,

$$\alpha_{k|k-1} = 2 + \exp\left(-\frac{\Delta t_k}{\tau}\right)(\alpha_{k-1|k-1} - 2). \tag{2.47}$$

2.1.1.3 Variational Bayes Approach

Feldmann *et al.*'s approach requires some assumptions to derive a Bayesian solution. In [28], Orguner proposed a variational Bayes approach for the measurement update

of the random matrix model. This technique is also used in [3] to derive the variational measurement update for extended target model with the orientation angle as an additional target state. A more detailed description of this approach is going to be given in Section 2.2.

2.1.1.4 Other Approaches

Other studies on the random matrix model are available in literature. For example, a prediction update for extended targets has been proposed in [32], where the object extension transformation is considered to be dependent on the kinematic state of the object. As shown in (2.26), previous work assumes that the time update of the object extension is independent of the kinematic state. However, this assumption would lose its validity during a constant or variable turn-rate maneuver. As a solution, a generalized prediction update for the extension based on the minimization of the Kullback-Leibler divergence has been derived.

Another research on the random matrix filtering is given in [33], where the time variation and distortion of the object extension is also considered. This study accounts for the changes in the object shape and orientation in addition to the size. Also, it proposes a solution for the distortion of the object extension problem that is caused by the sensor-to-target geometry.

Unlike the previous approaches where measurement data is considered to be received from a single sensor, a network of multiple sensors is considered in [34]. The study deals with the difficulty of varying object extension due to different sensor perspectives and compares four different measurement updates for the extended target tracking random matrix model.

2.1.2 Smoothing Algorithms

Koch's and Feldmann *et al.*'s approaches provides similar but different filtering (forward) algorithms for the recursive Bayesian estimation of the extended target random matrix model. Although Koch discusses the backward recursion briefly in [6], the de-

tailed information is not given. In [2], the smoothing (backward) expressions for both algorithms are derived to refine filtering estimates. In the view of this thesis, we will only use the model in [1] and therefore will only mention the smoothing expressions related to this model.

2.1.2.1 Granström and Bramstång's Approach

The Bayesian smoothing for an extended object starts with the filtered density at the final time step K. The well-known RTS-smoother is used in [35] to derive the backward recursion equations.

$$\begin{aligned} p(x_k, X_k|\mathcal{Y}_{0:K}) &= p(x_k, X_k|\mathcal{Y}_{0:k}) \\ &\quad \times \int \frac{p(x_{k+1}, X_{k+1}|x_k, X_k)p(x_{k+1}, X_{k+1}|\mathcal{Y}_{0:K})}{p(x_{k+1}, X_{k+1}|\mathcal{Y}_{0:k})} dx_k dX_k. \end{aligned} \tag{2.48}$$

The backward recursion starts with the final posterior density. Suppose that an intermediate smoothed density is Gaussian inverse Wishart distribution, i.e.,

$$\begin{aligned} p(x_{k+1}, X_{k+1}|\mathcal{Y}_{0:K}) &= p(x_{k+1}|\mathcal{Y}_{0:K})p(X_{k+1}|\mathcal{Y}_{0:K}) \\ &= \mathcal{N}(x_{k+1}; x_{k+1|K}, P_{k+1|K})\mathcal{IW}(X_k; \nu_{k+1|K}, V_{k+1|K}). \end{aligned} \tag{2.49}$$

Assuming that the transition density of the kinematic state does not depend on the extent state, we get

$$\begin{aligned} p(x_k, X_k|x_{k-1}, X_{k-1}) &= p(x_k|x_{k-1})p(X_k|X_{k-1}) \\ &= \mathcal{N}(x_k; Ax_{k-1}, Q)\mathcal{W}\left(X_k; n_k, \frac{MX_{k-1}M^\intercal}{n_k}\right), \end{aligned} \tag{2.50}$$

where M is a $d \times d$ invertible matrix describing the change of the object extension over time, the smoothed density $p(x_{k-1}, X_{k-1}|\mathcal{Y}_{0:K})$ is also in the same Gaussian inverse Wishart form, i.e.,

$$p(x_k, X_k|\mathcal{Y}_{0:K}) = \mathcal{N}(x_k; \hat{x}_{k|K}, P_{k|K})\mathcal{IW}(X_k; \nu_{k|K}, V_{k|K}). \tag{2.51}$$

The parameters of the smoothed density can be calculated by using the following smoothing expressions.

$$\hat{x}_{k|K} = \hat{x}_{k|k} + G(x_{k+1|K} - x_{k+1|k}), \tag{2.52a}$$

$$P_{k|K} = P_{k|k} + G(P_{k+1|K} - P_{k+1|k})G^{\mathsf{T}}, \tag{2.52b}$$

$$\nu_{k|K} = \nu_{k|k} + \eta^{-1}\left(\nu_{k+1|K} - \nu_{k+1|k} - \frac{2(d+1)^2}{n}\right), \tag{2.52c}$$

$$V_{k|K} = V_{k|k} + \eta^{-1}M^{-1}(V_{k+1|K} - V_{k+1|k})(M^{-1})^{\mathsf{T}}, \tag{2.52d}$$

$$G = P_{k|k}A^{\mathsf{T}}P_{k+1|k}^{-1}, \tag{2.52e}$$

$$\eta = 1 + \frac{\nu_{k+1|K} - \nu_{k+1|k} - 3(d+1)}{n}. \tag{2.52f}$$

2.2 Variational Bayes Inference

The measurement update equations proposed in [1] make the assumption that the predicted object extension is almost the same as the true extent. Considering the cases where this assumption is not applicable, we need to think of a more structured Bayesian approximation scheme. For this purpose, the well-known variational inference method (see [36, Ch. 10] and [37]) can be used. The implementations of the variational method to derive the measurement update equations of different target tracking scenarios can be found in [3], [28], and [38] .

Before explaining the previous work that uses variational method in extended target tracking applications, we need to understand the details of the variational inference. Consider a system where Y denotes the set of observed variables and X denotes the set of all latent variables and parameters. The log-likelihood function can be decomposed as

$$\ln p(Y) = \mathcal{L}(q) + \mathrm{KL}(q||p), \tag{2.53}$$

where

$$\mathcal{L}(q) = \int q(X) \ln\left\{\frac{p(Y,X)}{q(X)}\right\} dZ, \tag{2.54}$$

$$\mathrm{KL}(q||p) = -\int q(X) \ln\left\{\frac{p(X|Y)}{q(X)}\right\} dZ. \tag{2.55}$$

$\mathrm{KL}(q||p)$ is the Kullback-Leibler divergence [39] (also known as relative entropy) between $p(X|Y)$ and $q(X)$. Noting that $\mathrm{KL}(q||p)$ is always non-negative, we can write that $\ln p(Y) \geqslant \mathcal{L}(q)$. This result shows that $\mathcal{L}(q)$ is a lower bound of the log-likelihood. It is equal to $\ln p(Y)$ only if $\mathrm{KL}(q||p)$ is equal to zero, which occurs when $q(X)$ is equal to $p(X|Y)$. The optimization problem can be solved by maximizing the lower bound $\mathcal{L}(q)$ with respect to the distribution $q(X)$, which consequently maximizes the log-likelihood.

Consider the restricted family of distributions where $q(X)$ can be partitioned into number of disjoint groups. In this restricted family, q distributions approximated as the factorization of these groups, such that

$$q(X) = \prod_{i=1}^{M} q_i(X_i). \tag{2.56}$$

In order to maximize the lower bound, we need to optimize $\mathcal{L}(q)$ with respect to each factor of $q(X)$. By substituting (2.56) into (2.54), we can write $\mathcal{L}(q)$ as

$$\mathcal{L}(q) = \int \prod_i q_i \left\{ \ln p(Y, X) - \sum_i \ln q_i \right\} dX. \tag{2.57}$$

We can modify (2.57) to investigate the dependence on one of the factors $q_j(X_j)$, which is denoted simply by q_j, as follows.

$$\mathcal{L}(q) = \int q_j \left\{ \int \ln p(Y, X) \prod_{i \neq j} q_i dX_i \right\} dX_j - \int q_j \ln q_j dX_j + \text{const.} \tag{2.58}$$

Here introducing the expectation with respect to distributions q_i for $i \neq j$

$$\mathbb{E}_{i \neq j}[\ln p(Y, X)] = \int \ln p(Y, X) \prod_{i \neq j} q_i dX_i, \tag{2.59}$$

we can modify $\mathcal{L}(q)$ as

$$\mathcal{L}(q) = \int q_j \ln \tilde{p}(Y, X_j) dX_j - \int q_j \ln dX_j + \text{const}, \tag{2.60}$$

where

$$\ln \tilde{p}(Y, X_j) = \mathbb{E}_{i \neq j}[\ln p(Y, X)] + \text{const.} \tag{2.61}$$

The form of the (2.60) is a negative Kullback-Leibler divergence between q_j and $\tilde{p}(Y, X_j)$. Therefore, in order to maximize $\mathcal{L}(q)$, we need to have $q_j = \tilde{p}(Y, X_j)$. The optimal solution $q_j^*(X_j)$ is given by

$$\ln q_j^*(X_j) = \mathbb{E}_{i \neq j}[\ln p(X, X)] + \text{const.} \tag{2.62}$$

The optimal solution given in (2.62) forms the basis of the variational method. It shows that the logarithm of optimal solution for the factor q_j can be obtained by taking the expectation of the joint distribution's logartihm with respect to all the other factors $q_{i\neq j}$.

2.2.1 Orguner's Approach

Based on the result (2.62), Orguner proposed a variational measurement update for extended target tracking random matrix model in [28]. In order to be able to generate an analytical solution, the noise-free measurements are defined as additional latent variables. The measurement likelihood can be written as

$$\mathcal{N}(y_k^j; Hx_k, sX_k + R) = \int \mathcal{N}(y_k^j; z_k^j, R)\mathcal{N}(z_k^j; Hx_k, sX_k)dz_k^j, \tag{2.63}$$

where z_k^j represents the noise-free measurements of the noisy measurements y_k^j. Including $\mathcal{Z}_k$ into the list of quantities to be estimated, the joint posterior density $p(x_k, X_k, \mathcal{Z}_k|\mathcal{Y}_{0:k})$ needs to be computed. This joint smoothing posterior density can be approximated by using the following variational approximation:

$$p(x_k, X_k, \mathcal{Z}_k|\mathcal{Y}_{0:k}) \approx q(x_k, X_k, \mathcal{Z}_k) = q_x(x_k)q_X(X_k)q_{\mathcal{Z}}(\mathcal{Z}_k), \tag{2.64}$$

where $q_x(.), q_X(.), q_{\mathcal{Z}}(.)$ are the approximate posterior densities of x_k, X_k, and $\mathcal{Z}_k$, respectively. The analytical solutions of the estimates $\hat{q}_x(.), \hat{q}_X(.)$, and $\hat{q}_{\mathcal{Z}}(.)$, can be obtained using (2.62) as follows.

$$\log \hat{q}_x(x_k) = \mathbb{E}_{\hat{q}_X, \hat{q}_{\mathcal{Z}}}[\log p(x_k, X_k, \mathcal{Z}_k, \mathcal{Y}_k|\mathcal{Y}_{0:k-1})] + c_x, \tag{2.65a}$$

$$\log \hat{q}_X(X_k) = \mathbb{E}_{\hat{q}_x, \hat{q}_{\mathcal{Z}}}[\log p(x_k, X_k, \mathcal{Z}_k, \mathcal{Y}_k|\mathcal{Y}_{0:k-1})] + c_X, \tag{2.65b}$$

$$\log \hat{q}_{\mathcal{Z}}(\mathcal{Z}_k) = \mathbb{E}_{\hat{q}_x, \hat{q}_X}[\log p(x_k, X_k, \mathcal{Z}_k, \mathcal{Y}_k|\mathcal{Y}_{0:k-1})] + c_{\mathcal{Z}}, \tag{2.65c}$$

where c_x, c_x, and $c_{\mathcal{Z}}$ are constant with respect to x_k, X_k, and $\mathcal{Z}_k$, respectively.

The equations for the $(i+1)$th iterates of the posteriors are obtained using (2.65). These equations form the convergent recursive algorithm to obtain approximate pos-

teriors. The posterior densities are given as

$$q_x^{(i+1)}(x_k) = \mathcal{N}(x_k; \hat{x}_{k|k}^{(i+1)}, P_{k|k}^{(i+1)}), \tag{2.66a}$$

$$q_X^{(i+1)}(X_k) = \mathcal{IW}(X_k; \nu_{k|k}, V_{k|k}^{(i+1)}), \tag{2.66b}$$

$$q_{\mathcal{Z}}^{(i+1)}(\mathcal{Z}_k) = \prod_{i=1}^{m_k} \mathcal{N}(z_k^j; \hat{z}_k^{j.(i+1)}, \Sigma_k^{z,(i+1)}). \tag{2.66c}$$

The update equations for the parameters of the kinematic state are given as follows.

$$\hat{x}_{k|k}^{(i+1)} = P_{k|k}^{(i+1)}(P_{k|k-1}^{-1}\hat{x}_{k|k-1}^{(i+1)} + m_k H^{\mathsf{T}}\overline{(sX_k)^{-1}}\overline{z}_k), \tag{2.67}$$

$$P_{k|k}^{(i+1)} = (P_{k|k-1}^{-1} + m_k H^{\mathsf{T}}\overline{(sX_k)^{-1}}H)^{-1}, \tag{2.68}$$

where

$$\overline{z}_k \triangleq \frac{1}{m_k}\sum_{j=1}^{m_k} \mathbb{E}_{q_{\mathcal{Z}}^{(i)}}[z_k^j],$$

$$\overline{(sX_k)^{-1}} \triangleq \mathbb{E}_{q_X^{(i)}}[(sX_k)^{-1}].$$

The update equations for the extent state are given as follows.

$$\nu_{k|k} = \nu_{k|k-1} + m_k, \tag{2.69}$$

$$V_{k|k}^{(i+1)} = V_{k|k-1} + \frac{1}{s}\sum_{j=1}^{m_k} \overline{(z_k^j - Hx_k)(z_k^j - Hx_k)^{\mathsf{T}}}, \tag{2.70}$$

where

$$\overline{(z_k^j - Hx_k)(z_k^j - Hx_k)^{\mathsf{T}}} \triangleq \mathbb{E}_{q_x^{(i)}, q_{\mathcal{Z}}^{(i)}}\left[(z_k^j - Hx_k)(z_k^j - Hx_k)^{\mathsf{T}}\right].$$

Finally, the update equations for the noise-free measurements are given as follows.

$$\hat{z}_k^{j,(i+1)} = \Sigma_k^{z,(i+1)}(\overline{(sX_k)^{-1}}H\overline{x_k} + R^{-1}y_k^j), \tag{2.71}$$

$$\Sigma_k^{z,(i+1)} = (\overline{(sX_k)^{-1}} + R^{-1})^{-1}, \tag{2.72}$$

where $\overline{x_k} \triangleq \mathbb{E}_{q_x^{(i)}}(x_k)$.

2.2.2 Tuncer and Özkan's Approach

The variational Bayes technique proposed by Orguner in [28] is used to estimate approximate posteriors but it relies only on the forgetting factor for the changes in the

object orientation. Such an approach would not be sufficient in cases where the object orientation changes significantly over time. In [3], Tuncer and Özkan proposed another solution that provides the ability to track the orientation of the target separately from its extent.

In this study, the extent state is composed of the orientation angle $\theta_k \in \mathbb{R}$, and diagonal positive definite axis length matrix $\Lambda_k \in \mathbb{R}^{d\times d}$, Λ_k =diag$(\lambda_k^1, \lambda_k^2, ..., \lambda_k^d)$. The measurement likelihood becomes

$$p(y_k^j|x_k, \Lambda_k, \theta_k) \sim \mathcal{N}(y_k^j; Hx_k, sT_{\theta_k}\Lambda_k T_{\theta_k}^{\mathsf{T}} + R), \tag{2.73}$$

where $T_{\theta_k} \in \mathbb{R}^{d\times d}$ is the rotation matrix.

$$T_{\theta_k} = \begin{bmatrix} \cos(\theta_k) & -\sin(\theta_k) \\ \sin(\theta_k) & \cos(\theta_k) \end{bmatrix}. \tag{2.74}$$

In order to derive the variational smoother algorithm required in the context of this thesis study, we have benefited from the results of [28] and [3]. Therefore, the detailed derivations of both studies are given in Chapter 3 to derive the related smoothing expressions.

CHAPTER 3

VARIATIONAL SMOOTHER FOR THE MODEL WITHOUT ORIENTATION

The extended target tracking algorithm proposed by Koch was very promising because it was capable of estimating both kinematic and extension states of a target. This algorithm was later improved by Feldmann *et al.* by including the effect of measurement error in the problem. However, due to the additional measurement error parameter, the problem had no analytical solution and thus required heuristic assumptions and approximations. Later, Orguner proposed an iterative analytical measurement update based on well-known variational inference. Moreover, Granström and Bramstång derived the Bayesian smoothing algorithm for both Koch's and Feldmann *et al.*'s model to observe the benefits of smoothing [35] on extended target tracking applications.

In this chapter, the variational smoothing algorithm, which can be considered as a combination of Orguner's algorithm with Granström and Bramstång's, will be explained. We will first describe the solution of variational smoothing for the extended target tracking. Then, the derivation of the variational smoother for the extended target random matrix model will be provided. Later, the psuedo-code for the algorithm will be given. Lastly, brief comments about the algorithm performance will be provided.

3.1 System Description and Problem Formulation

For an extended target of interest, one can assume that the target state consists of the kinematic state $x_k \in \mathbb{R}^{n_x}$ and the symmetric and positive definite extent ma-

trix $X_k \in \mathbb{R}^{n_y \times n_y}$ where n_x and n_y are the dimensions of the kinematic state and the measurements respectively. The measurements at time k can be represented by $\mathcal{Y}_k \triangleq \{y_k^j \in \mathbb{R}^{n_y}\}_{j=1}^{m_k}$, where m_k is the number of measurements at time k. The individual measurements y_k^j at time k are assumed to be independent and normally distributed given the kinematic and extension states. Each measurement is independent and identically distributed as

$$y_k^j \sim \mathcal{N}(y_k^j; Hx_k, sX_k + R), \tag{3.1}$$

where $H \in \mathbb{R}^{n_y \times n_x}$ is the measurement matrix, s is the scaling factor, and R is the covariance of the measurement noise.

Bayesian filtering of extended objects is a recursive estimation cycle formed by the Chapman-Kolmogorov prediction (2.28) and Bayes update (2.9) steps. The extended smoother is the backward recursion of the filtered estimates. The recursion starts at the final time step K and given all the measurements, it aims to refine the filtered estimates. The smoothing equation of an extended target is given in (2.48).

The aim of our variational smoothing algorithm is to obtain an analytical approximation for the posterior density of the kinematic state $x_{0:K}$ and the object extension $X_{0:K}$. Since the exact analytical solution is not available due to covariance addition in the measurement distribution (3.1), a new variable z_k^j representing the noise-free measurements is introduced as in [28] (see (2.63) for the resulting measurement likelihood).

The joint density of the noisy measurements y_k^j and noise-free measurements z_k^j can be written as

$$p(y_k^j, z_k^j | x_k, X_k) = \mathcal{N}(y_k^j; z_k^j, R)\mathcal{N}(z_k^j; Hx_k, sX_k). \tag{3.2}$$

Including $\mathcal{Z}_k \triangleq \{z_k^j\}_{j=1}^{m_k}$ into the list of quantities to be estimated, the posterior density to be calculated becomes $p(x_{0:K}, X_{0:K}, \mathcal{Z}_{0:K} | \mathcal{Y}_{0:K})$. In order to generate an approximate analytical solution for this posterior, following variational approximation is used.

$$p(x_{0:K}, X_{0:K}, \mathcal{Z}_{0:K} | \mathcal{Y}_{0:K}) \approx q(x_{0:K}, X_{0:K}, \mathcal{Z}_{0:K}) \triangleq q_x(x_{0:K}) q_X(X_{0:K}) q_{\mathcal{Z}}(\mathcal{Z}_{0:K}), \tag{3.3}$$

where $q_x(.), q_X(.)$, and $q_Z(.)$ are the approximate posterior densities of $x_{0:K}, X_{0:K}$, and $\mathcal{Z}_{0:K}$, respectively. The well-known variational inference technique [36, Ch. 10] can be used to calculate the estimates $\hat{q}_x, \hat{q}_X$, and $\hat{q}_{\mathcal{Z}}$ of the approximate posterior densities as

$$\hat{q}_x, \hat{q}_X, \hat{q}_{\mathcal{Z}} = \underset{q_x, q_X, q_Z}{\arg\min}\, \mathrm{KL}(q(x_{0:K}, X_{0:K}, \mathcal{Z}_{0:K}) || p(x_{0:K}, X_{0:K}, \mathcal{Z}_{0:K} | \mathcal{Y}_{0:K})), \tag{3.4}$$

where $\mathrm{KL}(q(x)||p(x)) \triangleq \int q \log(\frac{q}{p}) dx$ is the Kullback-Lieber divergence [39]. As explained previously in Section 2.2, the optimal solution of this problem satisfies the following equations:

$$\log \hat{q}_x(x_{0:K}) = \mathbb{E}_{\hat{q}_X, \hat{q}_{\mathcal{Z}}}[\log p(x_{0:K}, X_{0:K}, \mathcal{Z}_{0:K}, \mathcal{Y}_{0:K})] + c_x, \tag{3.5a}$$

$$\log \hat{q}_X(X_{0:K}) = \mathbb{E}_{\hat{q}_x, \hat{q}_{\mathcal{Z}}}[\log p(x_{0:K}, X_{0:K}, \mathcal{Z}_{0:K}, \mathcal{Y}_{0:K})] + c_X, \tag{3.5b}$$

$$\log \hat{q}_{\mathcal{Z}}(\mathcal{Z}_{0:K}) = \mathbb{E}_{\hat{q}_x, \hat{q}_X}[\log p(x_{0:K}, X_{0:K}, \mathcal{Z}_{0:K}, \mathcal{Y}_{0:K})] + c_{\mathcal{Z}}, \tag{3.5c}$$

where c_x, c_X, and $c_{\mathcal{Z}}$ are constants with respect to variables $x_{0:K}, X_{0:K}$, and $\mathcal{Z}_{0:K}$, respectively. In order to obtain a solution for (3.5), fixed-point iteration can be used where only one factor is updated and the expected values on the right-hand sides are calculated using the last estimated values of other factors. The iteration is guaranteed to converge to a local optima of (3.4).

The joint smoothing posterior density to be approximated can be written as

$$\begin{aligned} p(x_{0:K}, X_{0:K}, \mathcal{Z}_{0:K} | \mathcal{Y}_{0:K}) &\propto p(x_{0:K}, X_{0:K}, \mathcal{Z}_{0:K}, \mathcal{Y}_{0:K}) \\ &= p(\mathcal{Y}_0 | \mathcal{Z}_0) p(\mathcal{Z}_0 | x_0, X_0) p(x_0, X_0) \\ &\quad \times \prod_{k=1}^{K} p(\mathcal{Y}_k | \mathcal{Z}_k) p(\mathcal{Z}_k | x_k, X_k) p(x_k | x_{k-1}) p(X_k | X_{k-1}), \end{aligned} \tag{3.6}$$

In order to obtain the state transition densities we need to examine the dynamical model of the system. The kinematical dynamics of the object can be expressed by the following state space model.

$$x_k = A x_{k-1} + w_k, \qquad w_k \sim \mathcal{N}(0, Q), \tag{3.7}$$

where A is the state (temporal evolution) matrix and w_k is the Gaussian distributed process noise with zero mean and covariance Q. This dynamic model results in the following state transition density for the kinematic state.

$$p(x_k | x_{k-1}) = \mathcal{N}(x_k; A x_{k-1}, Q). \tag{3.8}$$

The dynamical model for the object extension is adopted from [40] where the Beta-Bartlett stochastic evolution model was proposed. In this model, for an inverse Wishart distributed matrix $X_k \in \mathbb{R}^{d\times d}$ parametrized as

$$p(X_{k-1}) = \mathcal{IW}(X_{k-1}; \nu_{k-1|k-1}, V_{k-1|k-1}), \tag{3.9}$$

the transition density $p(X_k|X_{k-1})$ is given such that the predicted density also becomes inverse Wishart distributed with

$$p(X_k) = \mathcal{IW}(X_k; \nu_{k|k-1}, V_{k|k-1}). \tag{3.10}$$

The parameters of the predicted density are updated as follows.

$$\nu_{k|k-1} = \psi\nu_{k-1|k-1} + (1-\psi)(2d+2), \tag{3.11a}$$

$$V_{k|k-1} = \psi V_{k-1|k-1}, \tag{3.11b}$$

where $0 \ll \psi \leq 1$ is the forgetting (covariance discount) factor.
The backwards smoothing recursion equations are given as follows.

$$\nu_{k|K} = (1-\psi)\nu_{k|k} + \psi\nu_{k+1|K}, \tag{3.12a}$$

$$V_{k|K} = \left((1-\psi)V_{k|k}^{-1} + \psi V_{k+1|K}^{-1}\right)^{-1}. \tag{3.12b}$$

3.2 Derivation of the Variational Smoothing Equations

In this section, starting from the ith iterates of $q_x(.), q_X(.)$, and $q_{\mathcal{Z}}(.)$, the equations for the $(i+1)$th iterates, denoted as $q_x^{(i+1)}(.), q_X^{(i+1)}(.)$, and $q_{\mathcal{Z}}^{(i+1)}(.)$ are derived. For simplicity all the constant terms are denoted as c throughout the derivation.

The joint smoothing posterior density in (3.6) can be expressed as

$$\begin{aligned} &p(x_{0:K}, X_{0:K}, \mathcal{Z}_{0:K}, \mathcal{Y}_{0:K}) \\ &\quad= \left[\prod_{j=1}^{m_0} \mathcal{N}(y_0^j; z_0^j, R)\mathcal{N}(z_0^j; Hx_0, sX_0)\right] \mathcal{N}(x_0; \hat{x}_0, P_0)\mathcal{IW}(X_0; \nu_0, V_0) \\ &\qquad\times \prod_{k=1}^{K}\left[\prod_{j=1}^{m_k} \mathcal{N}(y_k^j; z_k^j, R)\mathcal{N}(z_k^j; Hx_k, sX_k)\right] p(x_k|x_{k-1})p(X_k|X_{k-1}), \end{aligned} \tag{3.13}$$

where $\hat{x}_0, P_0$ and ν_0, V_0 are the initial values for the kinematic and extent states, respectively.

In order to obtain the optimal solution, the expected value of the posterior density in (3.13) with respect to the last estimates of the fixed variables needs to be calculated. Some expected values are shown with the lines over the variables for simplicity.

3.2.1 Derivation for the approximate posterior $q_x^{(i+1)}(.)$

Using (3.5a), we can write

$$\log q_x^{(i+1)}(x_{0:K}) = \log \mathcal{N}(x_0; \hat{x}_0, P_0) + \sum_{k=1}^{K} \log \mathcal{N}(x_k; Ax_{k-1}, Q) + \sum_{k=0}^{K} \sum_{j=1}^{m_k} -0.5 \operatorname{tr}\left[((\overline{z_k^j} - Hx_k)(\overline{z_k^j} - Hx_k)^{\mathsf{T}} \overline{(sX_k)^{-1}} \right] + c, \tag{3.14a}$$

$$= \log \mathcal{N}(x_0; \hat{x}_0, P_0) + \sum_{k=1}^{K} \log \mathcal{N}(x_k; Ax_{k-1}, Q) + \sum_{k=0}^{K} -0.5 \operatorname{tr}\left[m_k (\overline{z}_k - Hx_k)(\overline{z}_k - Hx_k)^{\mathsf{T}} \overline{(sX_k)^{-1}} \right] + c, \tag{3.14b}$$

$$= \log \mathcal{N}(x_0; \hat{x}_0, P_0) + \sum_{k=1}^{K} \log \mathcal{N}(x_k; Ax_{k-1}, Q) + \sum_{k=0}^{K} \log \mathcal{N}\left(\overline{z}_k; Hx_k, \frac{\overline{(sX_k)^{-1}}^{-1}}{m_k} \right) + c, \tag{3.14c}$$

where

$$\overline{z_k^j} \triangleq \mathbb{E}_{q_{\mathcal{Z}}^{(i)}}[z_k^j], \tag{3.15a}$$

$$\overline{z}_k \triangleq \frac{1}{m_k} \sum_{j=1}^{m_k} \overline{z_k^j}, \tag{3.15b}$$

$$\overline{(sX_k)^{-1}} \triangleq \mathbb{E}_{q_X^{(i)}}[(sX_k)^{-1}]. \tag{3.15c}$$

Observing that (3.14c) has the form as the logarithm of the joint posterior distribution of the kinematic state in a linear Gaussian state-space model with the process

noise covariance Q and the measurement noise covariance $\left(\mathbb{E}_{q_X^{(i)}}[(sX_k)^{-1}]\right)^{-1}$, the well-known RTS smoother can be used to compute the approximate posterior density $q_x^{(i+1)}(x_{0:K})$.

3.2.1.1 Forward Recursion

It is necessary for the smoother to preserve the functional form of the state density. Thus, initializing the kinematic state with a Gaussian distributed prior $q_{x,0|-1}^{(i+1)}(x_0) = \mathcal{N}(x_0; \hat{x}_0, P_0)$, the filtered posterior can be written as

$$q_{x,k|k}^{(i+1)}(x_k) = \mathcal{N}(x_k; \hat{x}_{k|k}^{(i+1)}, P_{k|k}^{(i+1)}), \tag{3.16}$$

whose parameters are updated with the following measurement update equations.

$$\hat{x}_{k|k}^{(i+1)} = P_{k|k}^{(i+1)}((P_{k|k-1}^{(i+1)})^{-1}\hat{x}_{k|k-1}^{(i+1)} + m_k H^\mathsf{T}\overline{(sX_k)^{-1}}\overline{z}_k)^{-1}, \tag{3.17a}$$

$$P_{k|k}^{(i+1)} = ((P_{k|k-1}^{(i+1)})^{-1} + m_k H^\mathsf{T}\overline{(sX_k)^{-1}}H)^{-1}. \tag{3.17b}$$

Different than Feldmann's measurement update equation in (2.39a), the above measurement update uses the average of noise-free measurements instead of noisy measurements.

The prediction step requires the solution to the following Chapman-Kolmogorov equation:

$$\begin{aligned} p(x_k, X_k, |\mathcal{Y}_{0:k-1}) = \int & p(x_k, X_k|x_{k-1}, X_{k-1}) \\ & \times p(x_{k-1}, X_{k-1}|\mathcal{Y}_{0:k-1})dx_{k-1}dX_{k-1}. \end{aligned} \tag{3.18}$$

Assuming that the dynamical models of the kinematic and extent states are independent, the time update of these states can be separated. Using Kalman filter prediction equations, the predicted density $q_{x,k+1|k}^{(i+1)}(x_{k+1})$ can be obtained in the form of a Gaussian as

$$q_{x,k+1|k}^{(i+1)}(x_{k+1}) = \mathcal{N}(x_{k+1}; \hat{x}_{k+1|k}^{(i+1)}, P_{k+1|k}^{(i+1)}), \tag{3.19}$$

whose parameters are updated with the following time update equations.

$$\hat{x}_{k+1|k}^{(i+1)} = A\hat{x}_{k|k}^{(i+1)}, \tag{3.20a}$$

$$P_{k+1|k}^{(i+1)} = AP_{k|k}^{(i+1)}A^\mathsf{T} + Q. \tag{3.20b}$$

3.2.1.2 Backward Recursion

Using the RTS Smoother equations, the smoothed density $q_{x,k|K}^{(i+1)}(x_k)$ is also in the form of a Gaussian as

$$q_{x,k|K}^{(i+1)}(x_k) = \mathcal{N}(x_k; \hat{x}_{k|K}^{(i+1)}, P_{k|K}^{(i+1)}), \tag{3.21}$$

with the following backward recursion equations

$$G_k = P_{k|k}^{(i+1)} A^{\mathsf{T}} (P_{k+1|k}^{(i+1)})^{-1}, \tag{3.22a}$$

$$\hat{x}_{k|K}^{(i+1)} = \hat{x}_{k|k}^{(i+1)} + G_k(\hat{x}_{k+1|K}^{(i+1)} - A\hat{x}_{k|k}^{(i+1)}), \tag{3.22b}$$

$$P_{k|K}^{(i+1)} = P_{k|k}^{(i+1)} + G_k(P_{k+1|K}^{(i+1)} - P_{k+1|k}^{(i+1)})G_k^{\mathsf{T}}. \tag{3.22c}$$

3.2.2 Derivation for the approximate posterior $q_X^{(i+1)}(.)$

Using (3.5b), we can write

$$\begin{aligned}
\log q_X^{(i+1)}(X_{0:K}) &= \log \mathcal{IW}(X_0; \nu_0, V_0) + \sum_{k=1}^{K} \log p(X_k|X_{k-1}) \\
&\quad + \sum_{k=0}^{K} \mathbb{E}_{q_x^{(i)}, q_{\mathcal{Z}}^{(i)}}[\log p(\mathcal{Z}|x_k, X_k)] + c, \qquad (3.23a) \\
&= \log \mathcal{IW}(X_0; \nu_0, V_0) + \sum_{k=1}^{K} \log p(X_k|X_{k-1}) \\
&\quad + \sum_{k=0}^{K} -0.5 \operatorname{tr}\left[\sum_{j=1}^{m_k} \overline{(z_k^j - Hx_k)(z_k^j - Hx_k)^{\mathsf{T}}} (sX_k)^{-1}\right] \\
&\quad + \sum_{k=0}^{K} -0.5 m_k \log |X_k| + c, \qquad (3.23b)
\end{aligned}$$

where

$$\overline{(z_k^j - Hx_k)(z_k^j - Hx_k)^{\mathsf{T}}} \triangleq \mathbb{E}_{q_x^{(i)}, q_{\mathcal{Z}}^{(i)}}\left[(z_k^j - Hx_k)(z_k^j - Hx_k)^{\mathsf{T}}\right]. \tag{3.24}$$

Taking the exponential of both sides we can write

$$q_X^{(i+1)}(X_{0:K}) \propto \mathcal{IW}(X_0; \nu_0, V_0) \prod_{k=1}^{K} p(X_k|X_{k-1}) \prod_{k=0}^{K} L_{X,k}^{(i+1)}(X_k), \tag{3.25}$$

where

$$L_{X,k}^{i+1}(X_k) \triangleq |X_k|^{-\frac{1}{2}m_k} \exp\left(-\frac{1}{2}\sum_{j=1}^{m_k} \mathrm{tr}\,\overline{(z_k^j - Hx_k)(z_k^j - Hx_k)^{\mathsf{T}}}(sX_k)^{-1}\right) \quad (3.26)$$

for $k = 0, ..., K$. The posterior density in (3.25) corresponds to a standard smoothing problem for the following Markov model.

$$X_0 \sim \mathcal{IW}(X_0, \nu_0, V_0), \quad (3.27a)$$

$$X_k|X_{k-1} \sim p(X_k|X_{k-1}), \quad k = 1, ..., K, \quad (3.27b)$$

$$Y_{X,k}^{(i+1)} \sim p(Y_{X,k}^{(i+1)}|X_k) \triangleq L_{X,k}^{(i+1)}(X_k), \quad k = 0, ..., K, \quad (3.27c)$$

with some pseudo-measurements $Y_{X,k}^{(i+1)}$ and their likelihood $L_{X,k}^{(i+1)}(.)$. The solution to this problem can be obtained using a forward and backward recursion.

3.2.2.1 Forward Recursion

Suppose that the prior density is inverse Wishart distribution, i.e.,

$$q_{X,k|k-1}^{(i+1)} = \mathcal{IW}(X_k, \nu_{k|k-1}, V_{k|k-1}). \quad (3.28)$$

In the Bayesian update equations, the posterior density is proportional to the product of the prior with the measurement likelihood. Using the pseudo-likelihood, the posterior can be obtained as

$$q_{X,k|k}^{(i+1)}(X_k) \propto L_{X,k}^{(i+1)}(X_k) q_{X,k|k-1}^{(i+1)}(X_k). \quad (3.29)$$

Notice that the form of the pseudo-likelihood function given in (3.26) have resemblance to the form of an inverse Wishart distribution. Therefore, the multiplication in (3.29) results in an inverse Wishart posterior as

$$q_{X,k|k}^{(i+1)}(X_k) = \mathcal{IW}(X_k, \nu_{k|k}, V_{k|k}^{(i+1)}), \quad (3.30)$$

whose parameters are updated as follows.

$$\nu_{k|k} = \nu_{k|k-1} + m_k, \quad (3.31a)$$

$$V_{k|k}^{(i+1)} = V_{k|k-1}^{(i+1)} + \frac{1}{s}\sum_{j=1}^{m_k} \overline{(z_k^j - Hx_k)(z_k^j - Hx_k)^{\mathsf{T}}}. \quad (3.31b)$$

Note that the update equation of the degrees of freedom parameter $\nu_{k|k}$ does not depend on any expected value calculated at the estimates of the previous iteration. Hence, it does not change after the first iteration.

The prediction update formula can be expressed as

$$q^{(i+1)}_{X,k+1|k}(X_{k+1}) = \int p(X_{k+1}|X_k) q^{(i+1)}_{X,k|k}(X_k) dX_k. \tag{3.32}$$

When the posterior (3.30) is substituted into the prediction update formula (3.32) and using the Beta-Bartlett transition density prediction update formulae given in (3.11), an inverse Wishart predicted density $q^{(i+1)}_{X,k+1|k}(X_{k+1})$ is obtained.

$$q^{(i+1)}_{X,k+1|k}(X_{k+1}) = \mathcal{IW}(X_{k+1}, \nu_{k+1|k}, V^{(i+1)}_{k+1|k}), \tag{3.33}$$

where its parameters can be updated as

$$\nu_{k+1|k} = \psi \nu_{k|k} + (1-\psi)(2n_y + 2), \tag{3.34a}$$

$$V^{(i+1)}_{k+1|k} = \psi V^{(i+1)}_{k|k}. \tag{3.34b}$$

Note that the forward recursion preserves the inverse Wishart form of the predicted and posterior densities.

3.2.2.2 Backward Recursion

The backward recursion starts with the filtered density at final time step K. Suppose that an intermediate smoothed density is inverse Wishart as given below.

$$q^{(i+1)}_{X,k+1|K}(X_{k+1}) = \mathcal{IW}(X_{k+1}, \nu_{k+1|K}, V^{(i+1)}_{k+1|K}). \tag{3.35}$$

The backward update formula can be expressed as

$$q^{(i+1)}_{X,k|K}(X_k) = \int \frac{p(X_{k+1}|X_k) q^{(i+1)}_{X,k|k}(X_k)}{q^{(i+1)}_{X,k+1|k}(X_{k+1})} q^{(i+1)}_{X,k|K}(X_{k+1}) dX_{k+1}. \tag{3.36}$$

When the smoothed density (3.35) is substituted into the backward update formula (3.36) and using the Beta-Bartlet transition density backward smoothing update formulae given in (3.12), an inverse Wishart smoothed density $q^{(i+1)}_{X,k|K}(X_k)$ is obtained.

$$q^{(i+1)}_{X,k|K}(X_k) = \mathcal{IW}(X_k, \nu_{k|K}, V^{(i+1)}_{k|K}), \tag{3.37}$$

where

$$\nu_{k|K} = (1-\psi)\nu_{k|k} + \psi\nu_{k+1|K}, \tag{3.38a}$$

$$V_{k|K}^{(i+1)} = \left((1-\psi)\left(V_{k|k}^{(i+1)}\right)^{-1} + \psi\left(V_{k+1|K}^{(i+1)}\right)^{-1}\right)^{-1}. \tag{3.38b}$$

Note here again that the backward recursion also preserves the inverse Wishart form of the filtered density.

3.2.3 Derivation for the approximate posterior $q_{\mathcal{Z}}^{(i+1)}(.)$

Using (3.5c), we can write

$$\begin{aligned}\log q_{\mathcal{Z}}^{(i+1)}(\mathcal{Z}_{0:K}) &= \sum_{k=0}^{K}\sum_{j=0}^{m_k}\log\mathcal{N}(y_k^j; z_k^j, R) \\ &\quad + \sum_{k=0}^{K}\sum_{j=0}^{m_k} -0.5\,\mathrm{tr}\left[(z_k^j - H\overline{x_k})(z_k^j - H\overline{x_k})^{\mathsf{T}}\overline{(sX_k)^{-1}}\right] + c,\end{aligned} \tag{3.39a}$$

$$\begin{aligned}&= \sum_{k=0}^{K}\sum_{j=0}^{m_k}\log\mathcal{N}(y_k^j; z_k^j, R) \\ &\quad + \sum_{k=0}^{K}\sum_{j=0}^{m_k}\log\mathcal{N}(z_k^j; H\overline{x_k}, \overline{(sX_k)^{-1}}^{-1}) + c,\end{aligned} \tag{3.39b}$$

where $\overline{x_k} \triangleq \mathbb{E}_{q_x^{(i)}}(x_k)$. Observing Gaussian multiplication and using Kalman Filter measurement update formulas the approximate posterior can be expressed as

$$q_{\mathcal{Z}}^{(i+1)}(\mathcal{Z}_k) = \prod_{j=1}^{m_k}\mathcal{N}(z_k^j; \hat{z}_k^{j,(i+1)}, \Sigma_k^{z,(i+1)}), \tag{3.40}$$

whose update formulas are given as follows.

$$\hat{z}_k^{j,(i+1)} = \Sigma_k^{z,(i+1)}\left(\overline{(sX_k)^{-1}}H\overline{x_k} + R^{-1}y_k^j\right), \tag{3.41a}$$

$$\Sigma_k^{z,(i+1)} = \left(\overline{(sX_k)^{-1}} + R^{-1}\right)^{-1}. \tag{3.41b}$$

3.2.4 Calculation of the Expected Values

The equations derived in subsections 3.2.1-3.2.3 can be used to set up a variational smoothing algorithm to obtain approximate smoothed estimates sequentially. Thanks

to the factorization, the noise-free measurement set $\mathcal{Z}_{0:K}$ can easily be marginalized out from the joint density to obtain approximate smoothed density $p(x_{0:K}, X_{0:K}|\mathcal{Y}_{0:K})$ $\approx q_{x,k|K}(x_k)q_{X,k|K}(X_k)$.

The expected values given in the smoother equations can be calculated as

$$\overline{x_k} = \hat{x}^{(i)}_{k|K}, \tag{3.42a}$$

$$\overline{z^j_k} = \hat{z}^{j,(i)}_k, \tag{3.42b}$$

$$\overline{(sX_k)^{-1}} = \nu_{k|K}(sV^{(i)}_{k|K})^{-1}, \tag{3.42c}$$

$$\begin{aligned}\overline{(z^j_k - Hx_k)(z^j_k - Hx_k)^{\mathsf{T}}} &= (\hat{z}^{j,(i)}_k - H\hat{x}^{(i)}_{k|K})(\hat{z}^{j,(i)}_k - H\hat{x}^{(i)}_{k|K})^T \\ &\quad + HP^{(i)}_{k|K}H^{\mathsf{T}} + \Sigma^{z,(i)}_k.\end{aligned} \tag{3.42d}$$

Note that for the inverse Wishart distributed random matrix $X \sim \mathcal{IW}(X;\nu,V)$, its inverse is Wishart distributed with $X^{-1} \sim W(X^{-1};\nu,V^{-1})$ [31]. We can obtain (3.42c) by using this fact and the expected value equation of the Wishart distribution given in (2.8).

3.3 Variational Smoothing Algorithm for the model without Orientation

The psuedo-code for the proposed variational smoothing algorithm is given in Algorithm 1.

Algorithm 1: Variational Smoother for the Model without Orientation

Inputs: $A, H, \hat{x}_0, P_0, v_0, V_0, \psi \; and \; \mathcal{Y}_{0:K}$

Initialization:

$\hat{x}_{k|K}^{(0)} \leftarrow \hat{x}_0, \quad P_{k|K}^{(0)} \leftarrow P_0, \quad V_{k|K}^{(0)} \leftarrow V_0, \quad \nu_{k|K}^{(0)} \leftarrow \nu_0$

$\{z_k^{j,(0)}\}_{j=1}^{m_k} \leftarrow \{y_k^j\}_{j=1}^{m_k}, \quad \Sigma_k^{z,(0)} \leftarrow s\frac{V_0}{\nu_0 - 2n_y - 2} \qquad \text{for} \quad k = 0, ..., K$

for $i = 0$ to $i_{max} - 1$ **do**

  Calculate the expectations in (3.42)

  Initialize the predicted estimates

  $\hat{x}_{0|-1} \leftarrow \hat{x}_0, \quad P_{0|-1} \leftarrow P_0, \quad \nu_{0|-1} \leftarrow \nu_0, \quad V_{0|-1} \leftarrow V_0$

  for $k = 0$ to K **do**

    Measurement Update

      Update $\hat{x}_{k|k}^{(i+1)}$ and $P_{k|k}^{(i+1)}$ using (3.17)

      Update $\nu_{k|k}$ and $V_{k|k}^{(i+1)}$ using (3.31)

      Update $\hat{z}_k^{j,(i+1)}$ and $\Sigma_k^{z,(i+1)}$ using (3.41) for $j = 1, ..., m_k$

    Prediction

      Update $\hat{x}_{k+1|k}^{(i+1)}$ and $P_{k+1|k}^{(i+1)}$ using (3.20)

      Update $\nu_{k+1|k}$ and $V_{k+1|k}^{(i+1)}$ using (3.34)

  end for

  for $k = K - 1$ down to 0 **do**

    Smoothing

      Update $\hat{x}_{k|K}^{(i+1)}$ and $P_{k|K}^{(i+1)}$ using (3.22)

      Update $\nu_{k|K}$ and $V_{k|K}^{(i+1)}$ using (3.38)

  end for

end for

Set final smoothed estimates:

$\hat{x}_{k|K} \leftarrow \hat{x}_{k|K}^{(i_{max})}, \; P_{k|K} \leftarrow P_{k|K}^{(i_{max})}, \; V_{k|K} \leftarrow V_{k|K}^{(i_{max})}, \; \nu_{k|K} \leftarrow \nu_{k|K}, \;$ and

$X_{k|K} = V_{k|K}/(\nu_{k|K} - 2n_y - 2) \qquad \text{for } k = 0, ..., K.$

Leaving a proper margin in the selection of the iteration number i_{max} so that the recur-

sion converges, we can use the resulting expected values of the smoothed posteriors for x_k and X_k as the estimates for the object's kinematic and extent states.

3.3.1 Comments on the Algorithm Performance

The variational smoother proposed in this chapter is an iterative analytical algorithm that can be used to estimate the kinematic and extent states of an extended target. In order to see the effect of variational method on the smoothing algorithm, we can compare its performance with the Bayesian Smoother proposed by Granström and Bramstång in [2].

In the context of this thesis study, we studied two different extended target tracking scenarios. The first scenario simulates a target that moves along an almost straight path with constant velocity (see Figure 3.1). The second scenario, which is a popular simulation trajectory in extended target tracking studies and used in several papers such as [1], [3], [33], simulates a maneuvering target on a trajectory composed of one 45°and two 90°turns (see Figure 3.2). The simulation results and estimation errors of both scenarios are investigated in detail. It was observed that the derived variational smoother shows satisfactory performance in the first scenario where target does not have sharp turns. However, when we run this algorithm in the second scenario, its estimation accuracy degrades due to poor performance during the turns.

These results have led us to search for a more developed algorithm that can improve the estimation performance in maneuvering tracking scenarios. In order to overcome this problem, we propose a modified version of the variational smoother that also estimates the orientation of the target. The modified algorithm is elaborated in Chapter 4 with the derivation steps. The detailed simulation results and estimation performance metrics of all the algorithms mentioned are given in Chapter 5.

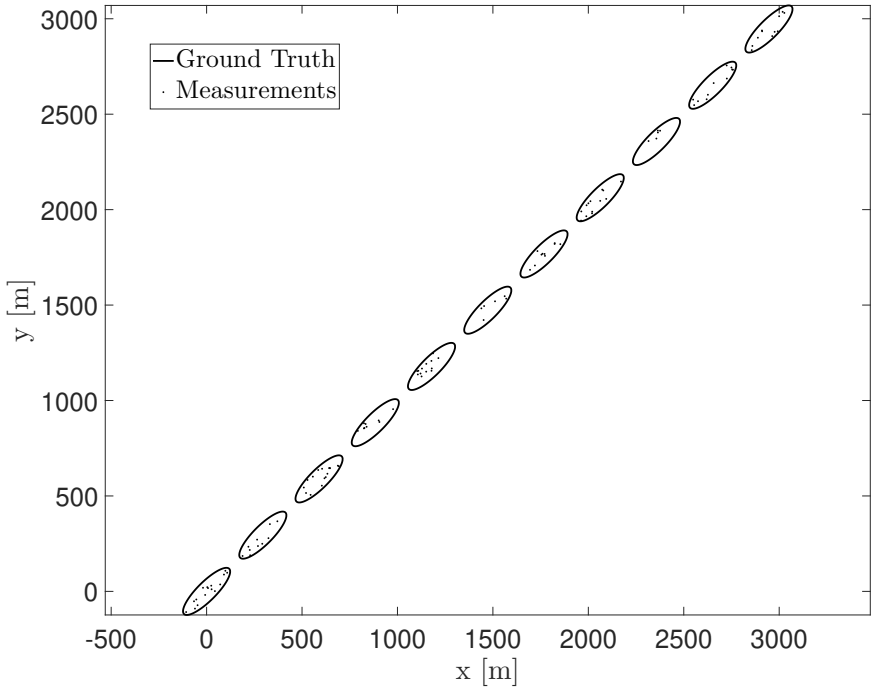

Figure 3.1: Scenario 1 - Target moving along an almost straight path with nearly constant velocity

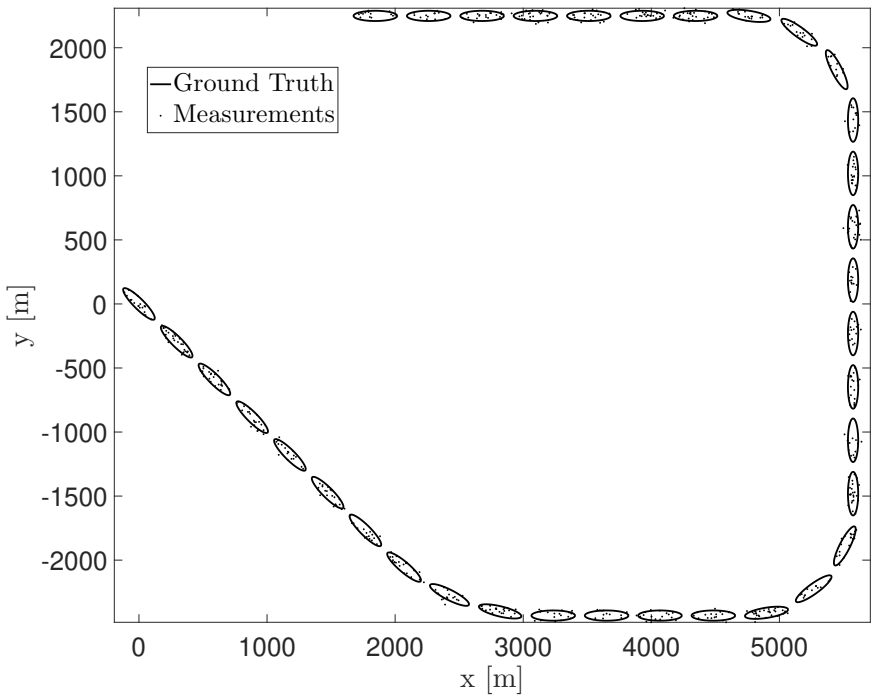

Figure 3.2: Scenario 2 - Maneuvering target with constant velocity

CHAPTER 4

VARIATIONAL SMOOTHER FOR THE MODEL WITH ORIENTATION

The variational smoothing algorithm proposed in Chapter 3 is capable of estimating the extent matrix but it relies only on the forgetting factor ψ for all of the changes in the object orientation. Such an approach would not be sufficient in cases where object orientation changes significantly over time. This expectation is also supported by the simulation results given in Chapter 5.

In [3], another solution that provides the ability to track both the orientation and the extent of a target is proposed. The purpose of this chapter is to provide necessary information about this model and the derivation steps of the smoothing algorithm based on such a model. After the derivation details, the psuedo-code of the algorithm will be provided at the end of the chapter.

4.1 System Description and Problem Formulation

When the orientation of the object is considered as another variable to be estimated, the extent state is composed of the orientation angle $\theta_k \in \mathbb{R}$, and diagonal positive definite axis length matrix $\Lambda_k \in \mathbb{R}^{n_y \times n_y}$, $\Lambda_k \triangleq \text{diag}(\lambda_k^1, \lambda_k^2, ..., \lambda_k^{n_y})$. The measurement likelihood can be written as

$$p(y_k^j | x_k, \Lambda_k, \theta_k) \sim \mathcal{N}(y_k^j; Hx_k, sT_{\theta_k}\Lambda_k T_{\theta_k}^{\mathsf{T}} + R), \tag{4.1}$$

where $T_{\theta_k} \in \mathbb{R}^{n_y \times n_y}$ is the rotation matrix. In two spatial dimensions it is defined as

$$T_{\theta_k} \triangleq \begin{bmatrix} \cos(\theta_k) & -\sin(\theta_k) \\ \sin(\theta_k) & \cos(\theta_k) \end{bmatrix}, \tag{4.2}$$

and satisfies the following properties: $T_{\theta_k}^{-1} = T_{\theta_k}^{\mathsf{T}}$, and $\det(T_{\theta_k}) = 1$.

Separating the orientation from the extent matrix adds another variable to be estimated to the Bayesian framework. Therefore, the variables to be estimated become the kinematic state vector x_k, axis length matrix Λ_k, and orientation angle θ_k. Note that different than the extent matrix specified in the previous chapter, the axis length matrix represents only the size of the ellipsoidal shape of the target. Therefore, it is a diagonal matrix whose diagonal elements specify the axis lengths of the ellipse.

Assume that the joint prior distribution of the kinematic state, axis length matrix, and orientation angle has the following form:

$$p(x_0, \Lambda_0, \theta_0) = \mathcal{N}(x_0; \hat{x}_0, P_0) \left[\prod_{\ell=1}^{n_y} \mathcal{IG}(\lambda_0^\ell; \alpha_0^\ell, \beta_0^\ell) \right] \mathcal{N}(\theta_0; \hat{\theta}_0, \Theta_0), \tag{4.3}$$

where $\mathcal{IG}(\lambda; \alpha, \beta)$ denotes the inverse Gamma distribution of the scalar variable λ with the scalar shape parameter α and scalar scale parameter β, defined as

$$\mathcal{IG}(\lambda; \alpha, \beta) = \frac{\beta^\alpha}{\Gamma(\alpha)} (\lambda)^{-\alpha-1} \exp\left(-\frac{\beta}{\lambda}\right), \tag{4.4}$$

where $\Gamma(.)$ denotes the gamma function. The mean of the distribution is given as

$$\mathbb{E}[\lambda] = \frac{\beta}{\alpha - 1} \quad \text{for} \quad \alpha > 1. \tag{4.5}$$

The parameters $\hat{x}_0$, P_0, $\{\alpha_0^l\}_{l=1}^{n_y}$, $\{\beta_0^l\}_{l=1}^{n_y}$, and $\hat{\theta}_0, \Theta_0$ are the initial values for the kinematic, axis length, and orientation states, respectively.

Applying the modification described above, additional to the densities of the kinematic state vector $x_{0:K}$ and the axis length matrix $\Lambda_{0:K}$, we also need to calculate the posterior density of the orientation angle $\theta_{0:K}$. Therefore, the joint posterior density can be written as

$$\begin{aligned} p(x_{0:K}, \Lambda_{0:K}, \theta_{0:K}, \mathcal{Z}_{0:K} | \mathcal{Y}_{0:K}) &\propto p(x_{0:K}, \Lambda_{0:K}, \theta_{0:K}, \mathcal{Z}_{0:K}, \mathcal{Y}_{0:K}) \\ &= p(\mathcal{Y}_0 | \mathcal{Z}_0) p(\mathcal{Z}_0 | x_0, \Lambda_0, \theta_0) p(x_0, \Lambda_0, \theta_0) \\ &\quad \times \prod_{k=1}^{K} p(\mathcal{Y}_k | \mathcal{Z}_k) p(\mathcal{Z}_k | x_k, \Lambda_k, \theta_k) p(x_k | x_{k-1}) p(\Lambda_k | \Lambda_{k-1}) p(\theta_k | \theta_{k-1}). \end{aligned} \tag{4.6}$$

Assuming that the dynamical models of the kinematic, axis length, and orientation states are independent, the time update of these states can be separated. Since the prior distributions of the kinematic and orientation states are both Gaussian, their

dynamical models can be expressed in a similar way such as

$$x_k = A_x x_{k-1} + w_{k,x} \qquad w_{k,x} \sim \mathcal{N}(0, Q_x), \tag{4.7a}$$

$$\theta_k = A_\theta \theta_{k-1} + w_{k,\theta} \qquad w_{k,\theta} \sim \mathcal{N}(0, Q_\theta). \tag{4.7b}$$

Hence, their state transition densities can be written as

$$p(x_k|x_{k-1}) = \mathcal{N}(x_k; A_x x_{k-1}, Q_x), \tag{4.8a}$$

$$p(\theta_k|\theta_{k-1}) = \mathcal{N}(\theta_k; A_\theta \theta_{k-1}, Q_\theta), \tag{4.8b}$$

where A is the state (temporal evolution) parameter and Q is the covariance of the process noise. The subscripts x and θ represent that these parameters are related to the kinematic or orientation state part of the system dynamics, respectively.

For the dynamical model of the axis length matrix, we need to modify the Beta-Bartlett stochastic evolution model given in (3.11) and (3.12) for an inverse Gamma distributed variable. Using Appendix A, we can set $\alpha = \frac{\nu}{2} - 1$, $\beta = \frac{\psi}{2}$, and $d = 1$. Hence, the prediction update equations become

$$\alpha_{k|k-1} = \psi \alpha_{k-1|k-1} + (1 - \psi), \tag{4.9a}$$

$$\beta_{k|k-1} = \psi \beta_{k-1|k-1}, \tag{4.9b}$$

where $0 \ll \psi \leq 1$ is the forgetting (covariance discount) factor.
This parameter change does not affect the backwards smoothing equations. However, in order to show the Beta-Bartlett updates with the same notation, we can re-write them as

$$\alpha_{k|K} = (1 - \psi)\alpha_{k|k} + \psi \alpha_{k+1|K}, \tag{4.10a}$$

$$\beta_{k|K} = \left((1 - \psi)\beta_{k|k}^{-1} + \psi \beta_{k+1|K}^{-1}\right)^{-1}. \tag{4.10b}$$

4.2 Derivation of the Variational Smoothing Equations

The joint posterior density in (4.3) can be written explicitly as

$$p(x_{0:K}, \Lambda_{0:K}, \theta_{0:K}, Z_{0:K}, \mathcal{Y}_{0:K})$$
$$= \left[\prod_{j=1}^{m_0} \mathcal{N}(y_0^j; z_0^j, R)\mathcal{N}(z_0^j; Hx_0, sT_{\theta_0}\Lambda_0 T_{\theta_0}^\mathsf{T})\right]$$

$$\times \mathcal{N}(x_0; \hat{x}_0, P_0) \left[\prod_{\ell=1}^{n_y} \mathcal{IG}(\lambda_0^\ell; \alpha_0^\ell, \beta_0^\ell)\right] \mathcal{N}(\theta_0; \hat{\theta}_0, \Theta_0)$$
$$\times \prod_{k=1}^{K} \left[\prod_{j=1}^{m_k} \mathcal{N}(y_k^j; z_k^j, R)\mathcal{N}(z_k^j; Hx_k, sT_{\theta_k}\Lambda_k T_{\theta_k}^{\mathsf{T}})\right]$$
$$\times \mathcal{N}(x_k; Ax_{k-1}, Q)p(\Lambda_k|\Lambda_{k-1})p(\theta_k|\theta_{k-1}). \tag{4.11}$$

We will use the same method of variational approximation to obtain an analytical solution for this posterior.

$$p(x_{0:K}, \Lambda_{0:K}, \theta_{0:K}, \mathcal{Z}_{0:K}, \mathcal{Y}_{0:K}) \approx q(x_{0:K}, \Lambda_{0:K}, \theta_{0:K}, \mathcal{Z}_{0:K})$$
$$\triangleq q_x(x_{0:K})q_\Lambda(\Lambda_{0:K})q_\theta(\theta_{0:K})q_\mathcal{Z}(\mathcal{Z}_{0:K}), \tag{4.12}$$

where $q_x(.), q_\Lambda(.)$, $q_\theta(.)$, and $q_Z(.)$ are the approximate posterior densities of $x_{0:K}$, $\Lambda_{0:K}$, $\theta_{0:K}$, and $\mathcal{Z}_{0:K}$, respectively. The optimal solution of this problem satisfies the following equations:

$$\log \hat{q}_x(x_{0:K}) = \mathbb{E}_{\hat{q}_\Lambda, \hat{q}_\theta, \hat{q}_\mathcal{Z}}[\log p(x_{0:K}, \Lambda_{0:K}, \theta_{0:K}, \mathcal{Z}_{0:K}, \mathcal{Y}_{0:K})] + c_x, \tag{4.13a}$$
$$\log \hat{q}_\Lambda(\Lambda_{0:K}) = \mathbb{E}_{\hat{q}_x, \hat{q}_\theta, \hat{q}_\mathcal{Z}}[\log p(x_{0:K}, \Lambda_{0:K}, \theta_{0:K}, \mathcal{Z}_{0:K}, \mathcal{Y}_{0:K})] + c_\Lambda, \tag{4.13b}$$
$$\log \hat{q}_\theta(\theta_{0:K}) = \mathbb{E}_{\hat{q}_x, \hat{q}_\Lambda, \hat{q}_\mathcal{Z}}[\log p(x_{0:K}, \Lambda_{0:K}, \theta_{0:K}, \mathcal{Z}_{0:K}, \mathcal{Y}_{0:K})] + c_\theta, \tag{4.13c}$$
$$\log \hat{q}_\mathcal{Z}(\mathcal{Z}_{0:K}) = \mathbb{E}_{\hat{q}_x, \hat{q}_\Lambda, \hat{q}_\theta}[\log p(x_{0:K}, \Lambda_{0:K}, \theta_{0:K}, \mathcal{Z}_{0:K}, \mathcal{Y}_{0:K})] + c_\mathcal{Z}, \tag{4.13d}$$

where c_x, c_Λ, c_θ, and $c_\mathcal{Z}$ are constants with respect to variables $x_{0:K}, \Lambda_{0:K}, \theta_{0:K}$, and $\mathcal{Z}_{0:K}$, respectively.

4.2.1 Derivation for the approximate posterior $q_x^{(i+1)}(.)$

Substituting (4.9) into (4.13a), we can write the approximate posterior of the kinematic state as

$$\log q_x^{(i+1)}(x_{0:K}) = \log \mathcal{N}(x_0; \hat{x}_0, P_0) + \sum_{k=1}^{K} \log \mathcal{N}(x_k; Ax_{k-1}, Q)$$
$$+ \sum_{k=0}^{K}\sum_{j=1}^{m_k} -0.5 \operatorname{tr}\left[((\overline{z_k^j} - Hx_k)(\overline{z_k^j} - Hx_k)^{\mathsf{T}}\overline{(sT_{\theta_k}\Lambda_k T_{\theta_k}^{\mathsf{T}})^{-1}}\right] + c, \tag{4.14a}$$
$$= \log \mathcal{N}(x_0; \hat{x}_0, P_0) + \sum_{k=1}^{K} \log \mathcal{N}(x_k; A\hat{x}_{k-1}, Q)$$

$$+\sum_{k=0}^{K} -0.5\,\mathrm{tr}\left[m_k(\overline{z_k}-Hx_k)(\overline{z_k}-Hx_k)^{\mathsf{T}}\overline{(sT_{\theta_k}\Lambda_k T_{\theta_k}^{\mathsf{T}})^{-1}}\right]+c, \tag{4.14b}$$

$$= \log\mathcal{N}(x_0;\hat{x}_0,P_0)+\sum_{k=1}^{K}\log\mathcal{N}(x_k;A\hat{x}_{k-1},Q)$$
$$+\sum_{k=0}^{K}\log\mathcal{N}\left(\overline{z_k};Hx_k,\frac{\overline{(sT_{\theta_k}\Lambda_k T_{\theta_k}^{\mathsf{T}})^{-1}}^{-1}}{m_k}\right)+c, \tag{4.14c}$$

where $\overline{z_k^j}$ is given in (3.15a), $\overline{z_k}$ is given in (3.15b), and

$$\overline{(sT_{\theta_k}\Lambda_k T_{\theta_k}^{\mathsf{T}})^{-1}} \triangleq \mathbb{E}_{q_\Lambda^{(i)},q_\theta^{(i)}}[(sT_{\theta_k}\Lambda_k T_{\theta_k}^{\mathsf{T}})^{-1}). \tag{4.15}$$

Taking the exponential of both sides in (4.14c), we can observe that it has the form of the joint posterior distribution for the kinematic state in a linear Gaussian state-space model with the process noise covariance Q and the measurement noise covariance $\left(\mathbb{E}_{q_\Lambda^{(i)},q_\theta^{(i)}}[(sT_{\theta_k}\Lambda_k T_{\theta_k}^{\mathsf{T}})^{-1})\right)^{-1}$. Hence, we can use RTS smoother to compute the approximate posterior density $q_x^{(i+1)}(x_{0:K})$.

4.2.1.1 Forward Recursion

Initializing the forward recursion with a Gaussian prior $q_{x,0|-1}^{(i+1)}(x_0)=\mathcal{N}(x_0;\hat{x}_0,P_0)$ and using the Kalman Filter measurement update equations, the Gaussian posterior $q_{x,k|k}^{(i+1)}(x_k)$ can be obtained as a

$$q_{x,k|k}^{(i+1)}(x_k)=\mathcal{N}(x_k;\hat{x}_{k|k}^{(i+1)},P_{k|k}^{(i+1)}), \tag{4.16}$$

where its parameters are updated by using the following equations.

$$\hat{x}_{k|k}^{(i+1)}=P_{k|k}^{(i+1)}((P_{k|k-1}^{(i+1)})^{-1}\hat{x}_{k|k-1}^{(i+1)}+m_k H^{\mathsf{T}}\overline{(sT_{\theta_k}\Lambda_k T_{\theta_k}^{\mathsf{T}})^{-1}}\overline{z}_k), \tag{4.17a}$$
$$P_{k|k}^{(i+1)}=((P_{k|k-1}^{(i+1)})^{-1}+m_k H^{\mathsf{T}}\overline{(sT_{\theta_k}\Lambda_k T_{\theta_k}^{\mathsf{T}})^{-1}}H)^{-1} \tag{4.17b}$$

Using the Kalman filter time update equations, the predicted density $q_{x_{k+1|k}}^{(i+1)}(x_{k+1})$ is also Gaussian

$$q_{x_{k+1|k}}^{(i+1)}(x_{k+1})=\mathcal{N}(x_{k+1};\hat{x}_{k+1|k}^{(i+1)},P_{k+1|k}^{(i+1)}), \tag{4.18}$$

where we can write the update equations for its parameters as

$$\hat{x}_{k+1|k}^{(i+1)}=A_x\hat{x}_{k|k}^{(i+1)}, \tag{4.19a}$$
$$P_{k+1|k}^{a,(i+1)}=A_x P_{k|k}^{a,(i+1)}A_x^{\mathsf{T}}+Q_x. \tag{4.19b}$$

4.2.1.2 Backward Recursion

Using the RTS smoother equations, the smoothed density $q^{(i+1)}_{x_{k|K}}(x_k)$ can be written in the Gaussian form as

$$q^{(i+1)}_{x_{k|K}}(x_k) = \mathcal{N}(x_k; \hat{x}^{(i+1)}_{k|K}, P^{(i+1)}_{k|K}), \tag{4.20}$$

with the backward update equations are given as

$$G_k = P^{(i+1)}_{k|k} A_x^{\mathsf{T}} (P^{(i+1)}_{k+1|k})^{-1}, \tag{4.21a}$$

$$\hat{x}^{(i+1)}_{k|K} = \hat{x}^{(i+1)}_{k|k} + G_k(\hat{x}^{(i+1)}_{k+1|K} - A_x \hat{x}^{(i+1)}_{k|k}), \tag{4.21b}$$

$$P^{(i+1)}_{k|K} = P^{(i+1)}_{k|k} + G_k(P^{(i+1)}_{k+1|K} - P^{(i+1)}_{k+1|k}) G_k^{\mathsf{T}}. \tag{4.21c}$$

4.2.2 Derivation for the approximate posterior $q^{(i+1)}_{\Lambda}(.)$

Substituting (4.9) into (4.13b) we get

$$\begin{aligned}
\log q^{(i+1)}_{\Lambda}(\Lambda_{0:K}) &= \sum_{\ell=1}^{n_y} \log \mathcal{IG}(\lambda_0^\ell; \alpha_0^\ell, \beta_0^\ell) + \sum_{k=1}^{K} \log p(\Lambda_k|\Lambda_{k-1}) \\
&\quad + \sum_{k=0}^{K} \mathbb{E}_{q^{(i)}_x, q^{(i)}_\theta, q^{(i)}_{\mathcal{Z}}}[\log p(\mathcal{Z}_k|x_k, \Lambda_k, \theta_k)] + c, \qquad (4.22a) \\
&= \sum_{\ell=1}^{n_y} \log \mathcal{IG}(\lambda_0^\ell; \alpha_0^\ell, \beta_0^\ell) + \sum_{k=1}^{K} \log p(\Lambda_k|\Lambda_{k-1}) \\
&\quad + \sum_{k=0}^{K} -0.5 \operatorname{tr} \sum_{j=1}^{m_k} \overline{(z_k^j - Hx_k)(z_k^j - Hx_k)^{\mathsf{T}} (sT_{\theta_k} \Lambda_k T_{\theta_k}^{\mathsf{T}})^{-1}} \\
&\quad + \sum_{k=0}^{K} -0.5 m_k \log |\Lambda_k| + c, \qquad (4.22b)
\end{aligned}$$

where

$$\begin{aligned}
&\overline{(z_k^j - Hx_k)(z_k^j - Hx_k)^{\mathsf{T}} (sT_{\theta_k} \Lambda_k T_{\theta_k}^{\mathsf{T}})^{-1}} \\
&\qquad \triangleq \mathbb{E}_{q^{(i)}_x, q^{(i)}_\theta, q^{(i)}_{\mathcal{Z}}} \left[(z_k^j - Hx_k)(z_k^j - Hx_k)^{\mathsf{T}} (sT_{\theta_k} \Lambda_k T_{\theta_k}^{\mathsf{T}})^{-1} \right]. \qquad (4.23)
\end{aligned}$$

Taking the exponential of both sides, we get the posterior density

$$q^{(i+1)}_{\Lambda}(\Lambda_{0:K}) \propto \left[\prod_{\ell=1}^{n_y} \mathcal{IG}(\lambda_0^\ell; \alpha_0^\ell, \beta_0^\ell) \right] \prod_{k=1}^{K} p(\Lambda_k|\Lambda_{k-1}) \prod_{k=0}^{K} L^{(i+1)}_{\Lambda,k}(\Lambda_k), \tag{4.24}$$

where

$$L_{\Lambda,k}^{(i+1)}(\Lambda_k) \triangleq |\Lambda_k|^{-\frac{1}{2}m_k} \times \exp\left(-\frac{1}{2}\operatorname{tr}\left((s\Lambda_k)^{-1}\sum_{j=1}^{m_k}\overline{T_{\theta_k}^{\mathsf{T}}(z_k^j - Hx_k)(z_k^j - Hx_k)^{\mathsf{T}}T_{\theta_k}}\right)\right) \tag{4.25}$$

for $k = 0, ..., K$. The over-lined part on the right hand side of (4.25) represents the following intermediate expected value

$$\overline{T_{\theta_k}^{\mathsf{T}}(z_k^j - Hx_k)(z_k^j - Hx_k)^{\mathsf{T}}T_{\theta_k}} \triangleq \mathbb{E}_{q_x^{(i)}, q_\theta^{(i)}, q_{\mathcal{Z}}^{(i)}}\left[T_{\theta_k}^{\mathsf{T}}(z_k^j - Hx_k)(z_k^j - Hx_k)^{\mathsf{T}}T_{\theta_k})\right]. \tag{4.26}$$

The posterior density in (4.24) corresponds to a standard smoothing problem for the following Markov model

$$\Lambda_0 \sim \prod_{\ell=1}^{n_y} \mathcal{IG}(\lambda_0^\ell; \alpha_0^\ell, \beta_0^\ell), \tag{4.27a}$$

$$\Lambda_k|\Lambda_{k-1} \sim p(\Lambda_k|\Lambda_{k-1}), \quad k = 1, ..., K, \tag{4.27b}$$

$$Y_{\Lambda,k}^{(i+1)} \sim p(Y_{\Lambda,k}^{(i+1)}|\Lambda_k) \triangleq L_{Q,k}^{i+1}(\Lambda_k), \quad k = 0, ..., K, \tag{4.27c}$$

with some pseudo-measurements $Y_{\Lambda,k}^{(i+1)}$ and their likelihood $L_{Q,k}^{(i+1)}(.)$. Thus, we can solve it using forward and backward recursion.

Define

$$M_k \triangleq \sum_{j=1}^{m_k}\overline{T_{\theta_k}^{\mathsf{T}}(z_k^j - Hx_k)(z_k^j - Hx_k)^{\mathsf{T}}T_{\theta_k}}, \tag{4.28}$$

so that we can write the likelihood as

$$L_{\Lambda,k}^{(i+1)}(\Lambda_k) \triangleq |\Lambda_k|^{-\frac{1}{2}m_k}\exp\left(-\frac{1}{2}\operatorname{tr}\left((s\Lambda_k)^{-1}M_k\right)\right). \tag{4.29}$$

Since $\Lambda_k = \operatorname{diag}(\lambda_k^1, ..., \lambda_k^{n_y})$ is a diagonal matrix, we can write its determinant and inverse as

$$|\Lambda_k| = \prod_{\ell=1}^{n_y}\lambda_k^\ell, \qquad \Lambda_k^{-1} = \operatorname{diag}\left(\frac{1}{\lambda_k^1}, ..., \frac{1}{\lambda_k^{n_y}}\right).$$

Hence, the exponential term on the right hand side of (4.29) can be written as

$$\operatorname{tr}\left((s\Lambda_k)^{-1}M_k\right) = \frac{1}{s}\sum_{j=1}^{n_y}\frac{[M_k]_{ll}}{\lambda_k^\ell}, \tag{4.30a}$$

$$\exp\left(-\frac{1}{2}\operatorname{tr}\left((s\Lambda_k)^{-1}M_k\right)\right) = \prod_{\ell=1}^{n_y}\exp\left(-\frac{1}{2s}\frac{[M_k]_{ll}}{\lambda_k^\ell}\right). \tag{4.30b}$$

Substituting (4.30b) into (4.29), the likelihood of each diagonal element of Λ_k is given as

$$L_{\Lambda,k}^{(i+1),\ell}(\lambda_k^\ell) = (\lambda_k^\ell)^{-\frac{1}{2}m_k} \exp\left(-\frac{1}{2s}\frac{[M_k]_{ll}}{\lambda_k^\ell}\right), \tag{4.31}$$

resulting in a likelihood of

$$L_{\Lambda,k}^{(i+1)}(\Lambda_k) = \prod_{\ell=1}^{n_y}\left[L_{\Lambda,k}^{(i+1),\ell}(\lambda_k^\ell)\right]. \tag{4.32}$$

Assume that the transition density of the axis length matrix is given as

$$p(\Lambda_k|\Lambda_{k-1}) = \prod_{\ell=1}^{n_y} p(\lambda_k^\ell|\lambda_{k-1}^\ell), \tag{4.33}$$

where $p(\lambda_k^\ell|\lambda_{k-1}^\ell)$ is the Beta-Bartlett transition density for the scalar λ_k^ℓ.
By combining these results, the approximate posterior of each diagonal element of the axis length matrix can be expressed as

$$q_\lambda^{(i+1)}(\lambda_{0:K}^\ell) \propto \mathcal{IG}(\lambda_0^\ell; \alpha_0^\ell, \beta_0^\ell)\prod_{k=1}^{K} p(\lambda_k^\ell|\lambda_{k-1}^\ell)\prod_{k=0}^{K} L_{\Lambda,k}^{(i+1),\ell}(\lambda_k^\ell), \tag{4.34}$$

and the approximate posterior of the axis length matrix is given as

$$q_\Lambda^{(i+1)}(\Lambda_{0:K}) \propto \prod_{\ell=1}^{n_y}\left[\mathcal{IG}(\lambda_0^\ell; \alpha_0^\ell, \beta_0^\ell)\prod_{k=1}^{K} p(\lambda_k^\ell|\lambda_{k-1}^\ell)\prod_{k=0}^{K} L_{\Lambda,k}^{(i+1)}(\lambda_k^\ell)\right], \tag{4.35a}$$

$$\propto \prod_{\ell=1}^{n_y} q_\lambda^{(i+1)}(\lambda_{0:K}^\ell). \tag{4.35b}$$

4.2.2.1 Forward Recursion

Suppose that the prior density is inverse Gamma

$$q_{\Lambda,k|k-1}^{(i+1)}(\Lambda_k) = \prod_{\ell=1}^{n_y}\mathcal{IG}(\lambda_k^{\ell,(i+1)}; \alpha_{k|k-1}^{\ell,(i+1)}, \beta_{k|k-1}^{\ell,(i+1)}). \tag{4.36}$$

Using the following measurement update formula

$$q_{\Lambda,k|k}^{(i+1)}(\Lambda_k) \propto L_{\Lambda,k}^{(i+1)}(\Lambda_k) q_{\Lambda,k|k-1}^{(i+1)}(\Lambda_k), \tag{4.37}$$

we get an inverse Gamma posterior $q_{\Lambda,k|k}^{(i+1)}(\Lambda_k)$ thanks to the form of the diagonal axis length matrix and pseudo-likelihood function given in (4.32).

$$q_{\Lambda,k|k}^{(i+1)}(\Lambda_k) = \prod_{\ell=1}^{n_y}\mathcal{IG}(\lambda_k^{\ell,(i+1)}; \alpha_{k|k}^{\ell,(i+1)}, \beta_{k|k}^{\ell,(i+1)}), \tag{4.38}$$

whose parameters can be updated with the following equations

$$\alpha_{k|k}^{\ell,(i+1)} = \alpha_{k|k-1}^{\ell,(i+1)} + \frac{1}{2}m_k, \tag{4.39a}$$

$$\beta_{k|k}^{\ell,(i+1)} = \beta_{k|k-1}^{\ell,(i+1)} + \frac{1}{2s}\sum_{j=1}^{m_k}\left[\overline{T_{\theta_k}^{\mathsf{T}}(z_k^j - Hx_k)(z_k^j - Hx_k)^{\mathsf{T}}T_{\theta_k}}\right]_{ll}. \tag{4.39b}$$

When the posterior in (4.38) is substituted into the prediction update formula given in (3.32) and using the Beta-Bartlett transition density prediction update formulae given in (4.9), an inverse Gamma predicted density $q_{\Lambda,k+1|k}^{(i+1)}(\Lambda_{k+1})$ is obtained.

$$q_{\Lambda,k+1|k}^{(i+1)}(\Lambda_{k+1}) = \prod_{\ell=1}^{n_y}\mathcal{IG}(\lambda_{k+1}^{\ell,(i+1)}; \alpha_{k+1|k}^{\ell,(i+1)}, \beta_{k+1|k}^{\ell,(i+1)}), \tag{4.40}$$

where we can write the update equations for its parameters as

$$\alpha_{k+1|k}^{\ell} = \psi\alpha_{k|k}^{\ell} + (1-\psi), \tag{4.41a}$$

$$\beta_{k|k-1}^{\ell,(i+1)} = \psi\beta_{k-1|k-1}^{\ell,(i+1)}, \tag{4.41b}$$

for $\ell = 1, ..., n_y$. This results show that all filtered densities are in the form of an inverse Gamma distribution.

4.2.2.2 Backward Recursion

Suppose that an intermediate smoothed density is inverse Gamma as given below.

$$q_{\Lambda,k+1|K}^{(i+1)}(\Lambda_{k+1}) = \prod_{\ell=1}^{n_y}\mathcal{IG}(\lambda_{k+1}^{\ell,(i+1)}; \alpha_{k+1|K}^{\ell,(i+1)}, \beta_{k+1|K}^{\ell,(i+1)}) \tag{4.42}$$

When the smoothed density (4.42) is substituted into the backward update formula (3.36) and using the backwards smoothing update formulae for Beta-Bartlett transition density given in (4.10), an inverse Gamma smoothed density $q_{\Lambda,k|K}^{(i+1)}(\Lambda_k)$ is obtained.

$$q_{\Lambda,k|K}^{(i+1)}(\Lambda_k) = \prod_{\ell=1}^{n_y}\mathcal{IG}(\lambda_k^{\ell,(i+1)}; \alpha_{k|K}^{\ell,(i+1)}, \beta_{k|K}^{(i+1)}) \tag{4.43}$$

whose parameters are updated as follows.

$$\alpha_{k|K}^{\ell} = (1-\psi)\alpha_{k|k}^{\ell} + \psi\alpha_{k+1|K}^{\ell}, \tag{4.44a}$$

$$\beta_{k|K}^{\ell,(i+1)} = \left((1-\psi)\left(\beta_{k|k}^{\ell,(i+1)}\right)^{-1} + \psi\left(\beta_{k+1|K}^{\ell,(i+1)}\right)^{-1}\right)^{-1}, \tag{4.44b}$$

for $\ell = 1, ..., n_y$. Notice that the backward update equations also preserve the inverse Gamma form of the posterior.

4.2.3 Derivation for the approximate posterior $q_\theta^{(i+1)}(.)$

Substituting (4.9) into (4.13c) we get

$$\log q_\theta^{(i+1)}(\theta_{0:K}) = \log \mathcal{N}(\theta_0; \hat{\theta}_0, \Theta_0) + \sum_{k=1}^{K} \log p(\theta_k|\theta_{k-1}) + \mathbb{E}_{q_x^{(i)}, q_\Lambda^{(i)}, q_\mathcal{Z}^{(i)}} \left[\log p(\mathcal{Z}_k|x_k, \Lambda_k, \theta_k)\right] + c, \tag{4.45a}$$

$$= \log \mathcal{N}(\theta_0; \hat{\theta}_0, \Theta_0) + \sum_{k=1}^{K} \log p(\theta_k|\theta_{k-1}) + \sum_{k=0}^{K} \sum_{j=1}^{m_k} -0.5 \operatorname{tr}\left[\overline{(z_k^j - Hx_k)(z_k^j - Hx_k)^\mathsf{T}(sT_{\theta_k}\Lambda_k T_{\theta_k}^\mathsf{T})^{-1}}\right] + c, \tag{4.45b}$$

where

$$\overline{(z_k^j - Hx_k)(z_k^j - Hx_k)^\mathsf{T}(sT_{\theta_k}\Lambda_k T_{\theta_k}^\mathsf{T})^{-1}} = \mathbb{E}_{q_x^{(i)}, q_\Lambda^{(i)}, q_\mathcal{Z}^{(i)}} \left[(z_k^j - Hx_k)(z_k^j - Hx_k)^\mathsf{T}(sT_{\theta_k}\Lambda_k T_{\theta_k}^\mathsf{T})^{-1}\right]. \tag{4.46}$$

Since the rotation matrix T_{θ_k} is a non-linear function of the orientation angle θ_k, the exact solution for $q_\theta^{(i+1)}(\theta_{0:K})$ does not exist. Therefore, a first order approximation will be used for the non-linear function $f(\theta_k) \triangleq T_{\theta_k}^\mathsf{T}(z_k^j - Hx_k)$ using its Taylor series expansion around $\hat{\theta}_{k|k}^{(i)}$,

$$f(\theta_k) = f(\hat{\theta}_{k|k}^{(i)}) + \nabla f(\hat{\theta}_{k|k}^{(i)})(\theta_k - \hat{\theta}_{k|k}^{(i)}) + H.O.T., \tag{4.47}$$

$$\text{where} \quad \nabla f(\hat{\theta}_{k|k}^{(i)}) \triangleq \left.\frac{\partial f}{\partial \theta_k}\right|_{\theta_k = \hat{\theta}_{k|k}^{(i)}}.$$

When the first order approximation of $f(\theta_k)$ is substituted into (4.45), we can write the expectation term as

$$\mathbb{E}_{q_x^{(i)}, q_\Lambda^{(i)}, q_\mathcal{Z}^{(i)}} \left[(a - b\theta_k)^\mathsf{T}(s\Lambda)^{-1}(a - b\theta_k)\right], \tag{4.48}$$

where

$$a \triangleq \left[f(\hat{\theta}_{k|k}^{(i)}) - \nabla f(\hat{\theta}_{k|k}^{(i)})\theta_{k|k}^{(i)}\right], \tag{4.49a}$$

$$b \triangleq -\nabla f((\hat{\theta}_{k|k}^{(i)}). \tag{4.49b}$$

Observing that this expectation term is the Gaussian likelihood of θ_k, the posterior $q_\theta^{(i+1)}(\theta_{0:K})$ density can be computed by using RTS smoother.

4.2.3.1 Forward Recursion

Forward recursion start with the Gaussian prior density $q^{(i+1)}_{\theta,0|-1}(\theta_0)$. The approximate posterior density can be expressed as Gaussian with mean $\hat{\theta}^{(i+1)}_{k|k}$ and covariance $\Theta^{(i+1)}_{k|k}$,

$$q^{(i+1)}_{\theta_{k|k}}(\theta_k) = \mathcal{N}(\theta_k; \hat{\theta}^{(i+1)}_{k|k}, \Theta^{(i+1)}_{k|k}), \tag{4.50}$$

where the mean and covariance of the posterior density can be computed by using Kalman filter measurement update equations as follows.

$$\hat{\theta}^{(i+1)}_{k|k} = \Theta^{(i+1)}_{k|k}\left((\Theta^{(i+1)}_{k|k-1})^{-1}\hat{\theta}^{(i+1)}_{k|k-1} + \delta\right), \tag{4.51a}$$

$$\Theta^{(i+1)}_{k|k} = \left((\Theta^{(i+1)}_{k|k-1})^{-1} + \Delta\right)^{-1}, \tag{4.51b}$$

$$\begin{aligned}\delta = \sum_{j=1}^{m_k} \operatorname{tr}\Big[&\overline{(s\Lambda_k)^{-1}}(T'_{\hat{\theta}^{(i)}_{k|k}})^{\mathsf{T}}\overline{(z^j_k - Hx_k)(.)^{\mathsf{T}}}(T'_{\hat{\theta}^{(i)}_{k|k}})\hat{\theta}^{(i)}_{k|k}\Big] \\ &- \operatorname{tr}\left[\overline{(s\Lambda_k)^{-1}}(T_{\hat{\theta}^{(i)}_{k|k}})^{\mathsf{T}}\overline{(z^j_k - Hx_k)(.)^{\mathsf{T}}}(T'_{\hat{\theta}^{(i)}_{k|k}})\right],\end{aligned} \tag{4.51c}$$

$$\Delta = \sum_{j=1}^{m_k} \operatorname{tr}\left[\overline{(s\Lambda_k)^{-1}}(T'_{\hat{\theta}^{(i)}_{k|k}})^{\mathsf{T}}\overline{(z^j_k - Hx_k)(.)^{\mathsf{T}}}(T'_{\hat{\theta}^{(i)}_{k|k}})\right], \tag{4.51d}$$

where

$$\overline{(s\Lambda_k)^{-1}} \triangleq \mathbb{E}_{q^{(i)}_{\Lambda}}[(s\Lambda_k)^{-1}], \tag{4.52a}$$

$$\overline{(z^j_k - Hx_k)(.)^{\mathsf{T}}} \triangleq \mathbb{E}_{q^{(i)}_x, q^{(i)}_{\mathcal{Z}}}\left[(z^j_k - Hx_k)(z^j_k - Hx_k)^{\mathsf{T}}\right], \tag{4.52b}$$

$$T'_{\hat{\theta}^{(i)}_{k|k}} \triangleq \left.\frac{\partial T_{\theta_k}}{\partial \theta_k}\right|_{\theta_k = \hat{\theta}^{(i)}_{k|k}}. \tag{4.52c}$$

Kalman filter prediction equations can be used in the time update step of the orientation state. The dynamics of the orientation state can be expressed by the following state space model.

$$\theta_k = A_\theta \theta_{k-1} + w_k, \qquad w_k \sim \mathcal{N}(0, Q_\theta). \tag{4.53}$$

Using the system dynamics above, the predicted density $q^{(i+1)}_{\theta_{k+1|k}}(\theta_{k+1})$ can be obtained in the form of a Gaussian as

$$q^{(i+1)}_{\theta_{k+1|k}}(\theta_{k+1}) = \mathcal{N}(\theta_k; \hat{\theta}^{(i+1)}_{k+1|k}, \Theta^{(i+1)}_{k+1|k}), \tag{4.54}$$

where

$$\hat{\theta}^{(i+1)}_{k+1|k} = A_\theta \hat{\theta}^{(i+1)}_{k|k}, \tag{4.55a}$$

$$\Theta^{(i+1)}_{k+1|k} = A_\theta \Theta^{(i+1)}_{k|k} A_\theta^{\mathsf{T}} + Q_\theta. \tag{4.55b}$$

4.2.3.2 Backward Recursion

Using the well known RTS smoother formulas, the smoothed density $q^{(i+1)}_{\theta_{k|K}}(\theta_k)$ becomes also in the form of a Gaussian as

$$q^{(i+1)}_{\theta_{k|K}}(\theta_k) = \mathcal{N}(\theta_k; \hat{\theta}^{(i+1)}_{k|K}, \Theta^{(i+1)}_{k|K}), \tag{4.56}$$

where

$$G_k = \Theta^{(i+1)}_{k|k} A_\theta^{\mathsf{T}} (\Theta^{(i+1)}_{k+1|k})^{-1}, \tag{4.57a}$$

$$\hat{\theta}^{(i+1)}_{k|K} = \hat{\theta}^{(i+1)}_{k|k} + G_k(\hat{\theta}^{(i+1)}_{k+1|K} - A_\theta \hat{\theta}^{(i+1)}_{k|k}), \tag{4.57b}$$

$$\Theta^{(i+1)}_{k|K} = \Theta^{(i+1)}_{k|k} + G_k(\Theta^{(i+1)}_{k+1|K} - \Theta^{(i+1)}_{k+1|k}) G_k^{\mathsf{T}}. \tag{4.57c}$$

4.2.4 Derivation for the approximate posterior $q^{(i+1)}_{\mathcal{Z}}(.)$

Substituting (4.9) into (4.13c) we get

$$\log q^{(i+1)}_{\mathcal{Z}}(\mathcal{Z}_{0:K}) = \sum_{k=0}^{K} \log p(\mathcal{Y}_k|\mathcal{Z}_k) + \sum_{k=0}^{K} \mathbb{E}_{q_x^{(i)}, q_\Lambda^{(i)}, q_\theta^{(i)}} \left[\log p(\mathcal{Z}_k|x_k, \Lambda_k, \theta_k)\right] + c, \tag{4.58a}$$

$$= \sum_{k=0}^{K} \sum_{j=0}^{m_k} \log \mathcal{N}(y_k^j; z_k^j, R) + \sum_{k=0}^{K} \sum_{j=0}^{m_k} -0.5 \operatorname{tr}\left[(z_k^j - H\overline{x_k})(z_k^j - H\overline{x_k})^{\mathsf{T}} \overline{(sT_{\theta_k} \Lambda_k T_{\theta_k}^{\mathsf{T}})^{-1}}\right] + c, \tag{4.58b}$$

$$= \sum_{k=0}^{K} \sum_{j=0}^{m_k} \log \mathcal{N}(y_k^j; z_k^j, R) + \sum_{k=0}^{K} \sum_{j=0}^{m_k} \log \mathcal{N}\left(z_k^j; H\overline{x_k}, \overline{(sT_{\theta_k} \Lambda_k T_{\theta_k}^{\mathsf{T}})^{-1}}^{-1}\right) + c, \tag{4.58c}$$

where $\overline{x_k} \triangleq \mathbb{E}_{q_x^{(i)}}(x_k)$. Observing Gaussian multiplication and using Kalman Filter measurement update formulas, the approximate posterior of the noise-free measurements can be written as follows.

$$q_{\mathcal{Z}}^{(i+1)}(\mathcal{Z}_k) = \prod_{j=1}^{m_k} \mathcal{N}(z_k^j; \hat{z}_k^{j,(i+1)}, \Sigma_k^{z,(i+1)}), \tag{4.59}$$

where its parameters are updated as follows

$$\hat{z}_k^{j,(i+1)} = \Sigma_k^{z,(i+1)} \left(\overline{(sT_{\theta_k}\Lambda_k T_{\theta_k}^{\mathsf{T}})^{-1}} H\overline{x_k} + R^{-1} y_k^j \right), \tag{4.60a}$$

$$\Sigma_k^{z,(i+1)} = \left(\overline{(sT_{\theta_k}\Lambda_k T_{\theta_k}^{\mathsf{T}})^{-1}} + R^{-1} \right)^{-1}. \tag{4.60b}$$

4.2.5 Calculation of the Expected Values

$$\overline{x_k} = \hat{x}_{k|K}^{(i)}, \tag{4.61a}$$

$$\overline{z_k} = \frac{1}{m_k} \sum_{j=1}^{m_k} \hat{z}_k^{j,(i)}, \tag{4.61b}$$

$$\overline{(s\Lambda_k)^{-1}} = \text{diag}\left(\frac{\alpha_{k|K}^{1,(i)}}{s\beta_{k|K}^{1,(i)}}, \frac{\alpha_{k|K}^{2,(i)}}{s\beta_{k|K}^{2,(i)}}, ..., \frac{\alpha_{k|K}^{n_y,(i)}}{s\beta_{k|K}^{n_y,(i)}} \right), \tag{4.61c}$$

$$\begin{aligned} \overline{(z_k^j - Hx_k)(.)^{\mathsf{T}}} &= HP_{k|K}^{(i)}H^{\mathsf{T}} + \Sigma_k^{z,(i)} \\ &\quad + \left(\hat{z}_k^{j,(i)} - Hx_{k|K}^{(i)} \right) \left(\hat{z}_k^{j,(i)} - Hx_{k|K}^{(i)} \right)^{\mathsf{T}}, \end{aligned} \tag{4.61d}$$

$$\begin{aligned} \overline{T_{\theta_k}^{\mathsf{T}}(z_k^j - Hx_k)(z_k^j - Hx_k)^{\mathsf{T}} T_{\theta_k}} &= \mathbb{E}_{q_\theta^{(i)}} \Bigg[T_{\theta_k}^{\mathsf{T}} \Bigg(\left(z_k^{j,(i)} - Hx_{k|K}^{(i)} \right) \left(z_k^{j,(i)} - Hx_{k|K}^{(i)} \right)^{\mathsf{T}} \\ &\quad + HP_{k|K}^{(i)}H^{\mathsf{T}} + \Sigma_k^{z,(i)} \Bigg) T_{\theta_k} \Bigg], \end{aligned} \tag{4.61e}$$

The expectation in (4.61e) can be calculated using the property $T_{\theta_k}^{\mathsf{T}} = T_{-\theta_k}$ of the rotation matrix and Lemma 1.

4.2.5.1 Calculation of $\mathbb{E}_{q_\Lambda^{(i)}, q_\theta^{(i)}}[(sT_{\theta_k}\Lambda_k T_{\theta_k}^{\mathsf{T}})^{-1}]$:

The derivation steps given in Lemma 1 are taken from [28] and repeated here only for the sake of completeness.

Lemma 1. (Taken from [28]) Given

$$M^{-1} = \begin{bmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{bmatrix}, \tag{4.62}$$

and $q_\theta^{(i)}(\theta_{k|K}) = \mathcal{N}(\theta_k; \hat{\theta}_{k|K}^{(i+1)}, \Theta_{k|K}^{(i+1)})$, the entries of the matrix $\mathbb{E}_{q_\theta^{(i)}}[(T_{\theta_k} M T_{\theta_k}^\mathsf{T})^{-1}]$ can be computed as

$$\begin{aligned}
\mathbb{E}_{q_\theta^{(i)}}[(T_{\theta_k} M T_{\theta_k}^\mathsf{T})^{-1}]_{11} &= \begin{bmatrix} m_{11} & m_{22} & -(m_{12}+m_{21}) \end{bmatrix} K\left(\hat{\theta}_{k|K}^{(i)}, \Theta_{k|K}^{(i)}\right), \\
\mathbb{E}_{q_\theta^{(i)}}[(T_{\theta_k} M T_{\theta_k}^\mathsf{T})^{-1}]_{12} &= \begin{bmatrix} m_{12} & -m_{21} & (m_{11}-m_{22}) \end{bmatrix} K\left(\hat{\theta}_{k|K}^{(i)}, \Theta_{k|K}^{(i)}\right), \\
\mathbb{E}_{q_\theta^{(i)}}[(T_{\theta_k} M T_{\theta_k}^\mathsf{T})^{-1}]_{21} &= \begin{bmatrix} m_{21} & -m_{12} & (m_{11}-m_{22}) \end{bmatrix} K\left(\hat{\theta}_{k|K}^{(i)}, \Theta_{k|K}^{(i)}\right), \\
\mathbb{E}_{q_\theta^{(i)}}[(T_{\theta_k} M T_{\theta_k}^\mathsf{T})^{-1}]_{22} &= \begin{bmatrix} m_{22} & m_{11} & (m_{12}+m_{21}) \end{bmatrix} K\left(\hat{\theta}_{k|K}^{(i)}, \Theta_{k|K}^{(i)}\right),
\end{aligned} \tag{4.63}$$

where

$$K\left(\hat{\theta}_{k|K}^{(i)}, \Theta_{k|K}^{(i)}\right) \triangleq \frac{1}{2} \begin{bmatrix} 1+\cos(2\hat{\theta}_{k|K}) \exp(-2\Theta_{k|K}^{(i)}) \\ 1-\cos(2\hat{\theta}_{k|K}) \exp(-2\Theta_{k|K}^{(i)}) \\ \sin(2\hat{\theta}_{k|K}) \exp(-2\Theta_{k|K}^{(i)}) \end{bmatrix}. \tag{4.64}$$

Since Λ_k is diagonal by definition, we can write

$$\begin{aligned}
\mathbb{E}_{q_\Lambda^{(i)}, q_\theta^{(i)}}[(s T_{\theta_k} \Lambda_k T_{\theta_k}^\mathsf{T})^{-1}] &= (1-\exp(-2\Theta_{k|K}^{(i)})) \frac{\mathrm{tr}(\overline{(s\Lambda_k)^{-1}})}{2} \mathbb{I}_2 \\
&\quad + \exp(-2\Theta_{k|K}^{(i)}) \left(T_{\hat{\theta}_{k|K}^{(i)}} \overline{(s\Lambda_k)^{-1}} T_{\hat{\theta}_{k|K}^{(i)}}^\mathsf{T} \right).
\end{aligned} \tag{4.65}$$

Let's define

$$\overline{(z_k^j - Hx_k)(.)^\mathsf{T}} \triangleq N = \begin{bmatrix} n_{11} & n_{12} \\ n_{21} & n_{22}, \end{bmatrix}. \tag{4.66}$$

We need to calculate $\mathbb{E}[T_{\theta_k}^\mathsf{T} N T_{\theta_k}]$. In order to be able to do this, we are going to put this expected value in the form of Lemma 1.

$$T_{\theta_k}^\mathsf{T} N T_{\theta_k} = (T_{\theta_k}^\mathsf{T} N^{-1} T_{\theta_k})^{-1} = (T_{-\theta_k} N^{-1} T_{-\theta_k}^\mathsf{T})^{-1}, \tag{4.67}$$

$$\mathbb{E}[T_{\theta_k}^\mathsf{T} N T_{\theta_k}] = \mathbb{E}\left[(T_{-\theta_k} N^{-1} T_{-\theta_k}^\mathsf{T})^{-1} \right], \tag{4.68}$$

$$\begin{aligned}
\mathbb{E}_{q_\theta^{(i)}}[T_{\theta_k}^\mathsf{T} N T_{\theta_k}]_{11} &= \begin{bmatrix} n_{11} & n_{22} & +(n_{12}+n_{21}) \end{bmatrix} K\left(\hat{\theta}_{k|K}^{(i)}, \Theta_{k|K}^{(i)}\right), \\
\mathbb{E}_{q_\theta^{(i)}}[T_{\theta_k}^\mathsf{T} N T_{\theta_k}^\mathsf{T}]_{12} &= \begin{bmatrix} n_{12} & -n_{21} & (n_{22}-n_{11}) \end{bmatrix} K\left(\hat{\theta}_{k|K}^{(i)}, \Theta_{k|K}^{(i)}\right), \\
\mathbb{E}_{q_\theta^{(i)}}[T_{\theta_k}^\mathsf{T} N T_{\theta_k}^\mathsf{T}]_{21} &= \begin{bmatrix} n_{21} & -n_{12} & (n_{22}-n_{11}) \end{bmatrix} K\left(\hat{\theta}_{k|K}^{(i)}, \Theta_{k|K}^{(i)}\right), \\
\mathbb{E}_{q_\theta^{(i)}}[T_{\theta_k}^\mathsf{T} N T_{\theta_k}^\mathsf{T}]_{22} &= \begin{bmatrix} n_{22} & n_{11} & -(n_{12}+n_{21}) \end{bmatrix} K\left(\hat{\theta}_{k|K}^{(i)}, \Theta_{k|K}^{(i)}\right),
\end{aligned} \tag{4.69}$$

with the $K\left(\hat{\theta}_{k|K}^{(i)}, \Theta_{k|K}^{(i)}\right)$ defined in (4.64).

4.2.5.2 Calculation of $\mathbb{E}_{q_\Lambda^{(i)}}[(s\Lambda_k)]$

This expectation will be used to initialize the covariance of the noise-free measurements Σ_k^z and to calculate the resulting expected value of the axis length matrix. It can be calculated by using the mean of an inverse Gamma distribution given in (4.5).

$$\mathbb{E}_{q_\Lambda^{(i)}}[(s\Lambda_k)] = \overline{(s\Lambda_k)} = \text{diag}\left(\frac{s\beta_{k|K}^{1,(i)}}{\alpha_{k|K}^{1,(i)}-1}, \frac{s\beta_{k|K}^{2,(i)}}{\alpha_{k|K}^{2,(i)}-1}, ..., \frac{s\beta_{k|K}^{n_y,(i)}}{\alpha_{k|K}^{n_y,(i)}-1}\right), \tag{4.70}$$

4.3 Variational Smoothing Algorithm for the Model with Orientation

The psuedo-code for the proposed variational smoothing algorithm is given in Algorithm 2.

Algorithm 2: Variational Smoother for the Model with Orientation

Inputs: $A, H, \hat{x}_0, P_0, \{\alpha_0^\ell, \beta_0^\ell\}_{\ell=1}^{n_y}, \hat{\theta}_0, \Theta_0, \psi$ *and* $\mathcal{Y}_{0:K}$

Initialization:

$\hat{x}_{k|K}^{(0)} \leftarrow \hat{x}_0, \quad P_{k|K}^{(0)} \leftarrow P_0, \quad \hat{\theta}_{k|K}^{(0)} \leftarrow \hat{\theta}_0, \quad \Theta_{k|K}^{(0)} \leftarrow \Theta_0,$

$\{\alpha_{k|K}^{\ell,(0)}\}_{\ell=1}^{n_y} \leftarrow \{\alpha_0^\ell\}_{\ell=1}^{n_y}, \quad \{\beta_{k|K}^{\ell,(0)}\}\}_{\ell=1}^{n_y} \leftarrow \{\beta_0^\ell\}_{\ell=1}^{n_y}, \quad \{z_k^{j,(0)}\}_{j=1}^{m_k} \leftarrow \{y_k^j\}_{j=1}^{m_k},$

$\Sigma_k^{z,(0)} \leftarrow \mathbb{E}_{q_\Lambda^{(0)}}[(s\Lambda_k)]$ using (4.70), for $k = 0, ..., K$

for $i = 0$ to $i_{max} - 1$ **do**

  Calculate the expectations in (4.61)

  Initialize the predicted estimates

  $\hat{x}_{0|-1} \leftarrow \hat{x}_0, \quad P_{0|-1} \leftarrow P_0,$

  $\{\alpha_{0|-1}^\ell\}_{\ell=1}^{n_y} \leftarrow \{\alpha_0^\ell\}_{\ell=1}^{n_y}, \quad \{\beta_{0|-1}^\ell\}_{\ell=1}^{n_y} \leftarrow \{\beta_0^\ell\}_{\ell=1}^{n_y},$

  $\hat{\theta}_{0|-1} \leftarrow \hat{\theta}_0, \quad \Theta_{0|-1} \leftarrow \Theta_0,$

  for $k = 0$ to K **do**

    Measurement Update

      Update $\hat{x}_{k|k}^{(i+1)}$ and $P_{k|k}^{(i+1)}$ using (4.17)

      Update $\alpha_{k|k}^{\ell,(i+1)}$ and $\beta_{k|k}^{\ell,(i+1)}$ using (4.39) for $\ell = 1, ..., n_y$

      Update $\hat{\theta}_{k|k}^{(i+1)}$ and $\Theta_{k|k}^{(i+1)}$ using (4.51)

      Update $\hat{z}_k^{j,(i+1)}$ and $\Sigma_k^{z,(i+1)}$ using (4.58) for $j = 1, ..., m_k$

    Prediction

      Update $\hat{x}_{k+1|k}^{(i+1)}$ and $P_{k+1|k}^{(i+1)}$ using (4.19)

      Update $\alpha_{k+1|k}^{\ell,(i+1)}$ and $\beta_{k+1|k}^{\ell,(i+1)}$ using (4.41) for $\ell = 1, ..., n_y$

      Update $\hat{\theta}_{k+1|k}^{(i+1)}$ and $\Theta_{k+1|k}^{(i+1)}$ using (4.55)

  end for

  for $k = K - 1$ down to 0 **do**

    Smoothing

      Update $\hat{x}_{k|K}^{(i+1)}$ and $P_{k|K}^{(i+1)}$ using (4.21)

      Update $\alpha_{k|K}^{\ell,(i+1)}$ and $\beta_{k|K}^{\ell,(i+1)}$ using (4.44) for $\ell = 1, ..., n_y$

      Update $\hat{\theta}_{k|K}^{(i+1)}$ and $\Theta_{k|K}^{(i+1)}$ using (4.57)

  end for

end for

Set final smoothed estimates:

$\hat{x}_{k|K} \leftarrow \hat{x}_{k|K}^{(i_{max})}, \quad P_{k|K} \leftarrow P_{k|K}^{(i_{max})}, \quad \hat{\theta}_{k|K} \leftarrow \hat{\theta}_{k|K}^{(i_{max})}, \quad \Theta_{k|K} \leftarrow \Theta_{k|K}^{(i_{max})},$

$\{\alpha_{k|K}^\ell\}_{\ell=1}^{n_y} \leftarrow \{\alpha_{k|K}^{\ell,(i_{max})}\}_{\ell=1}^{n_y}, \quad \{\beta_{k|K}^\ell\}_{\ell=1}^{n_y} \leftarrow \{\beta_{k|K}^{\ell,(i_{max})}\}_{\ell=1}^{n_y},$

$\Lambda_{k|K} = \text{diag}\left(\frac{\beta_{k|K}^1}{(\alpha_{k|K}^1 - 1)}, \frac{\beta_{k|K}^2}{(\alpha_{k|K}^2 - 1)}, ..., \frac{\beta_{k|K}^{n_y}}{(\alpha_{k|K}^{n_y} - 1)}\right),$

$X_{k|K} = \left(1 - \exp(-2\Theta_{k|K})\right) \frac{\text{tr}(\Lambda_{k|K})}{2} \mathbb{I}_2 + \exp(-2\Theta_{k|K})(T_{\hat{\theta}_{k|K}} \Lambda_{k|K} T_{\hat{\theta}_{k|K}}^\mathsf{T}),$

for $k = 0, ..., K$

CHAPTER 5

SIMULATION RESULTS

In this chapter, we will present the simulation results of the algorithms proposed in Chapter 3 and Chapter 4. For simpler representation, we will denote these algorithms as VS and VSO, which stands for variational smoother and variational smoother with orientation, respectively. In order to evaluate the performances of the proposed algorithms, their results will be compared with Feldmann *et al.*'s filter, denoted as FFK, proposed in [1], Granström and Bramstång's smoother, denoted as GB, proposed in [2], and Tuncer and Özkan's variational filter for the model with orientation, denoted as TO, proposed in [28].

Throughout the simulation study, two different extended target tracking scenarios were utilized. In the first scenario, a target moving on a straight trajectory with nearly constant velocity is simulated. In the second scenario, a maneuvering target with constant velocity is used. The simulations were performed 100 times and the error metrics are given as the average of these 100 MC runs. The details of these scenarios, the model and initial parameters, and the error metrics used to evaluate the algorithm performances will be provided in the following sections.

5.1 Scenario 1: Non-maneuvering Target Case

In this simulation scenario, an object which moves with nearly constant velocity on an almost straight path is used. In the following subsections, we provide the parameters and equations used to generate measurements for this scenario. Then, the algorithm results are given with the related error metrics.

5.1.1 Measurement Generation

At each MC run, the scenario is generated randomly by using the constant velocity state space model for the kinematic state given as

$$x_{k+1} = A_x x_k + B w_{k,x}, \quad w_{k,x} \sim \mathcal{N}(w_{k,x}; 0, Q_x), \tag{5.1}$$

where $x_k = [p_k^x, p_k^y, v_k^x, v_k^y]^\mathsf{T}$ and $n_x = 4$, in two spatial dimensions. $w_{k,x}$ is the zero mean and $Q_x = \sigma^2 I_2$ covariance process noise of the kinematic state with $\sigma = 1\,\mathrm{m/s^2}$ and I_2 is the 2×2 identity matrix. The true parameters of the state space model are given as

$$A_x = \begin{bmatrix} I_2 & TI_2 \\ 0_2 & I_2 \end{bmatrix}, \quad B = \begin{bmatrix} T^2 I_2 \\ TI_2 \end{bmatrix}, \tag{5.2}$$

where 0_2 is 2×2 zero matrix and $T = 10\,\mathrm{s}$ is the sampling period. Initializing a Gaussian prior for the kinematic state $x_0 \sim \mathcal{N}(x_0; \hat{x}_0, P_0)$ with mean and covariance

$$\hat{x}_0 = \begin{bmatrix} 0\,\mathrm{m} & 0\,\mathrm{m} & 10\,\mathrm{m/s} & 10\,\mathrm{m/s} \end{bmatrix}^\mathsf{T}, \tag{5.3a}$$

$$P_0 = \mathrm{diag}\left(\begin{bmatrix} 1\,\mathrm{m^2} & 1\,\mathrm{m^2} & 1\,\mathrm{m^2/s^2} & 1\,\mathrm{m^2/s^2} \end{bmatrix} \right), \tag{5.3b}$$

we generate the true kinematic state for $k = 0, ..., 70$. We consider the orientation angle as the inverse tangent of the ratio of the velocities in the x and y direction to calculate the rotation matrix.

$$\theta_k = \tan^{-1}(v_k^x / v_k^y), \quad T_{\theta_k} \triangleq \begin{bmatrix} \cos(\theta_k) & -\sin(\theta_k) \\ \sin(\theta_k) & \cos(\theta_k) \end{bmatrix}. \tag{5.4}$$

The lengths of the major and minor axes are set to $\lambda_k^1 = 170^2\,\mathrm{m^2}$ and $\lambda_k^2 = 40^2\,\mathrm{m^2}$, respectively. The axis length matrix is $\Lambda_k = \mathrm{diag}(\lambda_k^1, \lambda_k^2)$. The true extent can be obtained as follows.

$$X_k = T_{\theta_k} \Lambda_k T_{\theta_k}^\mathsf{T}. \tag{5.5}$$

The measurement set $\mathcal{Y}_k = \{y_k^j\}_{j=1}^{m_k}$ is generated as follows.

$$m_k = \max\left(2, m \sim \text{Poisson}(10)\right), \tag{5.6a}$$

$$y_k^j \sim \mathcal{N}(y; Hx_k, sX_k + R), \tag{5.6b}$$

where $s = 0.25$ is the scaling factor, $R = 5^2 I_2 \mathrm{m^2}$ is the sensor noise, $H = [I_2\ 0_{2\times 2}]$ is the measurement matrix, and Poisson(M) represents the Poisson density with expected value M.

5.1.2 Model and Initial Density Parameters

The predicted kinematic state estimates and covariance are initialized with the initial kinematic state mean and covariance such that

$$x_{0|-1} = \hat{x}_0, \tag{5.7a}$$

$$P_{0|-1} = P_0. \tag{5.7b}$$

For the models with orientation, the shape and scale estimates of the axis length matrix are initialized as $\alpha^{1,2}_{0|-1} = [2\ 2]^\mathsf{T}$ and $\beta^{1,2}_{0|-1} = [100^2\ 100^2]^\mathsf{T}$. The predicted mean of the orientation is initialized as $\theta_{0|-1} = \theta_0$ using the true orientation angle and the predicted covariance is set to $\Theta_{0|-1} = 0.01\,\mathrm{rad}^2$. The state transition parameter of the orientation is given as $A_\theta = 1$, and the process noise covariance of the orientation is given as $Q_\theta = 0.001\,\mathrm{rad}^2$. In order to initialize the algorithms with equivalent extent estimates, the expected value of their extents are equalized. The initial extent estimate can be obtained by using Lemma 1 and (4.70).

$$\begin{aligned} X_{0|-1} &= \mathbb{E}_{q_\Lambda, q_\theta}[T_{\theta_0}\Lambda_0 T_{\theta_0}^\mathsf{T}] = \mathbb{E}_{q_\theta}[T_{\theta_0}\,\mathbb{E}_{q_\Lambda}[\Lambda_0]T_{\theta_0}^\mathsf{T}] \\ &= \mathbb{E}_{q_\theta}[(T_{\theta_0}(\mathbb{E}_{q_\Lambda}[\Lambda_0])^{-1}T_{\theta_0}^\mathsf{T})^{-1}] \\ &= (1-\exp(-2\Theta_{0|-1}))\frac{\mathrm{tr}(\overline{\Lambda_0})}{2}\mathbb{I}_2 \\ &\quad + \exp(-2\Theta_{0|-1})\left(T_{\theta_0}\overline{(\Lambda_0)}T_{\theta_0}^\mathsf{T}\right). \end{aligned} \tag{5.8}$$

For the models without orientation, the initial degrees of freedom is set to $\nu_{0|-1} = 7$ and the scale matrix is calculated as

$$V_{0|-1} = X_{0|-1}(\nu_{0|-1} - 2n_y - 2). \tag{5.9}$$

The forgetting factor is set to $\psi = 0.85$ for VS, VSO, and TO algorithms. The equivalent time constant parameter τ of FFK, and the degrees of freedom parameter n_k of GB can be obtained using the forgetting factor as described in Appendix B. The resulting equivalences are given as

$$\tau = -\frac{T}{\ln\psi}, \tag{5.10a}$$

$$n_k = \frac{\nu_{k|k} - 2n_y - 2}{\frac{1}{\psi} - 1}. \tag{5.10b}$$

5.1.3 Error Metrics

The estimation performance of the algorithms are compared using the error metrics of Gaussian Wasserstein (GW) distance [41] and root-mean-square-error(RMSE).

5.1.3.1 Gaussian Wasserstein Distance

The error metrics of the estimated kinematic state vector and axis length matrix are calculated using their Gaussian Wasserstein distances with respect to the ground truth of the target. The GW distance can be obtained as follows.

$$\begin{aligned}&\text{GW1}(p_{k,gt}, \Lambda_{k,gt}, p_{k|K}, \Lambda_{k|K})^2 \\ &\quad \triangleq ||p_{k,gt} - p_{k|K}||_2^2 + \text{tr}\left[\Lambda_{k,gt} + \Lambda_{k|K} - 2(\Lambda_{k,gt}^{1/2}\Lambda_{k|K}\Lambda_{k,gt}^{1/2})^{1/2}\right], \end{aligned} \tag{5.11}$$

where $p_{k,gt}$ and $\Lambda_{k,gt}$ represent the ground truth of the target center and axes lengths, while $p_{k|K}$ and $\Lambda_{k|K}$ represent the smoothed estimates of the center and axes lengths. The GW1 distance in (5.11) is divided into two parts representing the position and axis length error separately. These terms are given as follows.

$$\text{GW}_p = ||p_{gt} - p_{k|K}||_2^2, \tag{5.12a}$$

$$\text{GW}_\Lambda = \text{tr}\left[\Lambda_{gt} + \Lambda_{k|K} - 2(\Lambda_{gt}^{1/2}\Lambda_{k|K}\Lambda_{gt}^{1/2})^{1/2}\right]. \tag{5.12b}$$

In order to include the effect of the orientation error in GW distance, we also calculate the GW distance using the extent matrix.

$$\begin{aligned}&\text{GW2}(p_{k,gt}, X_{k,gt}, p_{k|K}, X_{k|K})^2 \\ &\quad \triangleq ||p_{k,gt} - p_{k|K}||_2^2 + \text{tr}\left[X_{k,gt} + X_{k|K} - 2(X_{k,gt}^{1/2}X_{k|K}X_{k,gt}^{1/2})^{1/2}\right], \end{aligned} \tag{5.13}$$

where $X_{k,gt}$ represents the ground truth of the target extent and $X_{k|K}$ represents the smoothed estimates of the extent. The individual extent error can be written as follows.

$$\text{GW}_X = \text{tr}\left[X_{gt} + X_{k|K} - 2(X_{gt}^{1/2}X_{k|K}X_{gt}^{1/2})^{1/2}\right]. \tag{5.14}$$

5.1.3.2 Root-mean-square-error

In order to obtain the error of the orientation, we use the RMSE metric which can be calculated as follows.

$$\text{RMSE}(\theta_{gt}, \theta_{k|K}) = \sqrt{\frac{1}{K}\sum_{k=1}^{K}(\theta_{k,gt} - \theta_{k|K})^2}, \tag{5.15}$$

where K is the final time step, θ_{gt} and $\theta_{k|K}$ are the orientation of the ground truth and smoothed estimate, respectively.

5.1.4 Simulation Results

In order to be able to compare the performance of the algorithms, the average error metrics are given for each time step in Figure 5.2-Figure 5.7. Moreover, an example trajectory of the target with the algorithm estimates is given in Figure 5.1. In all plots, filtering algorithms are shown in dashed lines, while smoothing algorithms are given in solid lines.

As it can be seen on the trajectory in Figure 5.1, the positional error of the filtering algorithms increase when the number of measurements is low at a particular scan. On the other hand, smoothing algorithms are able to overcome this issue thanks to the refining nature of the backward recursion. Since the target orientation does not tend to change over time, there is no significant difference in the extent and orientation estimates of the algorithms.

The average GW distance of the kinematic state is given in Figure 5.2. We can observe that the average GW_x of the smoothing algorithms VS, VSO, and GB are lower throughout the trajectory. Since the update equations of the kinematic state are similar to each other, the average GW_x distance of the filtering algorithms are close to each other at each time. Likewise, no significant difference is observed on the average GW_x distance of the smoothing algorithms.

When we look at the average GW distances of the axis length matrices, which are given in Figure 5.3, we can observe that the proposed algorithms VS and VSO show the best performance throughout the trajectory. The other smoothing algorithm GB

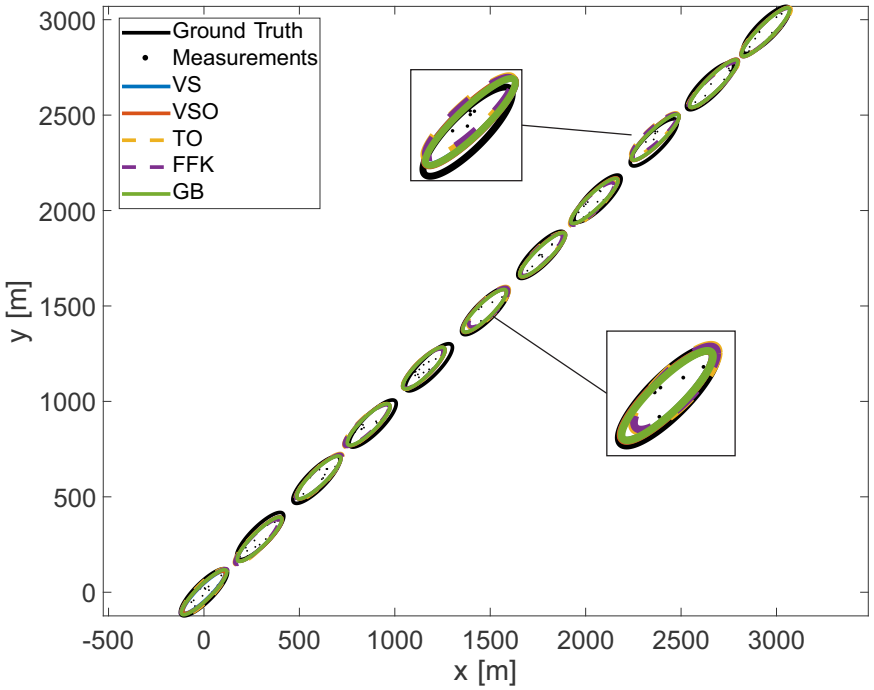

Figure 5.1: An example target trajectory of scenario 1. The results of the filtering algorithms are shown with dashed lines while those of the smoothing algorithms are shown with solid lines.

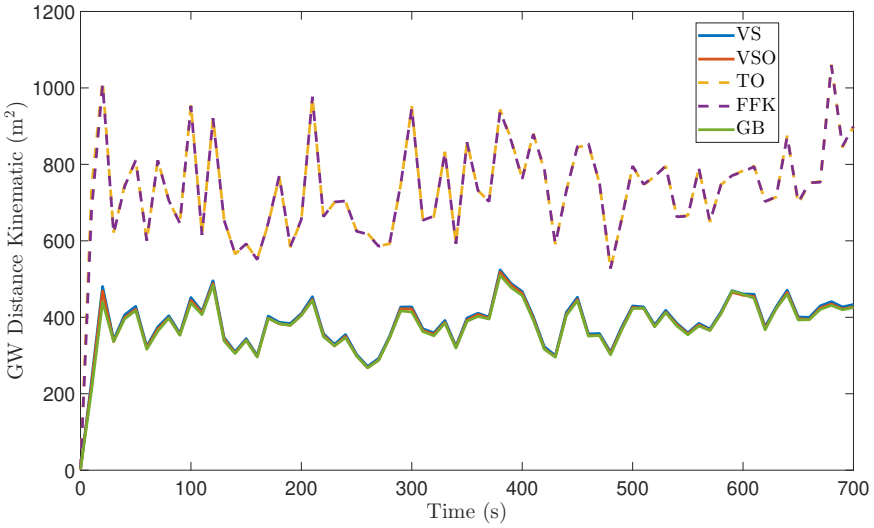

Figure 5.2: Average GW_x distance for scenario 1. The results of the filtering algorithms are shown with dashed lines while those of the smoothing algorithms are shown with solid lines.

also shows better performance than the filtering algorithms but its performance is lower than the proposed variational smoothers. This result is expected because of the iterative structure of the variational algorithms. The same difference can be observed in the average GW_{Λ} distances of TO and FFK algorithms. The variational nature of the TO algorithm provides better estimation results.

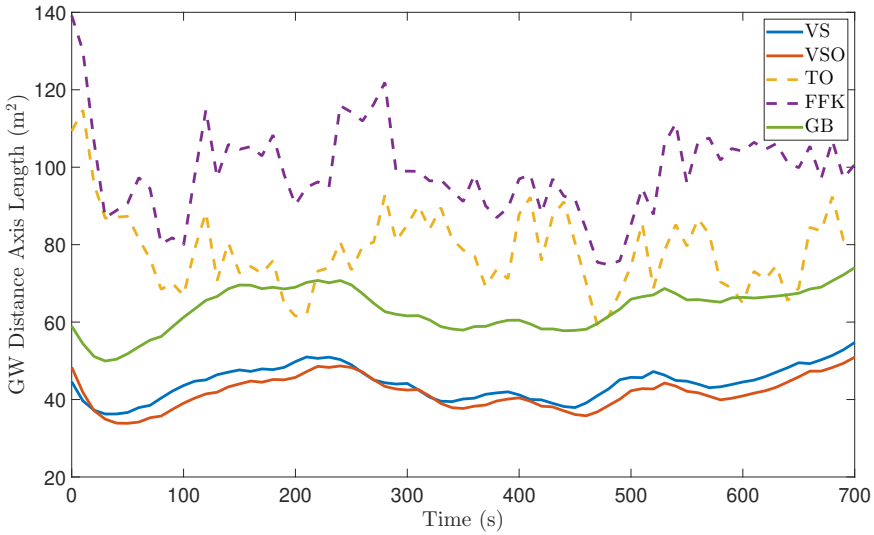

Figure 5.3: Average GW_{Λ} distance for scenario 1. The results of the filtering algorithms are shown with dashed lines while those of the smoothing algorithms are shown with solid lines.

In Figure 5.4 the average GW1 distance, which is a combination of GW_x and GW_{Λ} distances, is given. Since kinematic state error dominates the overall error, the shape of the GW1 is similar to GW_x. However, it can be clearly seen that the performance of the variational algorithms are better within their respective group (smoother or filter).

The GW1 distance provides us the information about the algorithm's performance on estimating the kinematic state and axis length matrix but it doesn't contain information about the estimation performance of the orientation. In order to evaluate this, the average RMSE of the orientation is given in Figure 5.5. When we examine the average RMSE(θ), it shows that the orientation error of the VSO is lower than other smoothing algorithms and the orientation error of TO is lower than FFK. As expected, these results show that estimating the orientation separate from the extent provides lower orientation error.

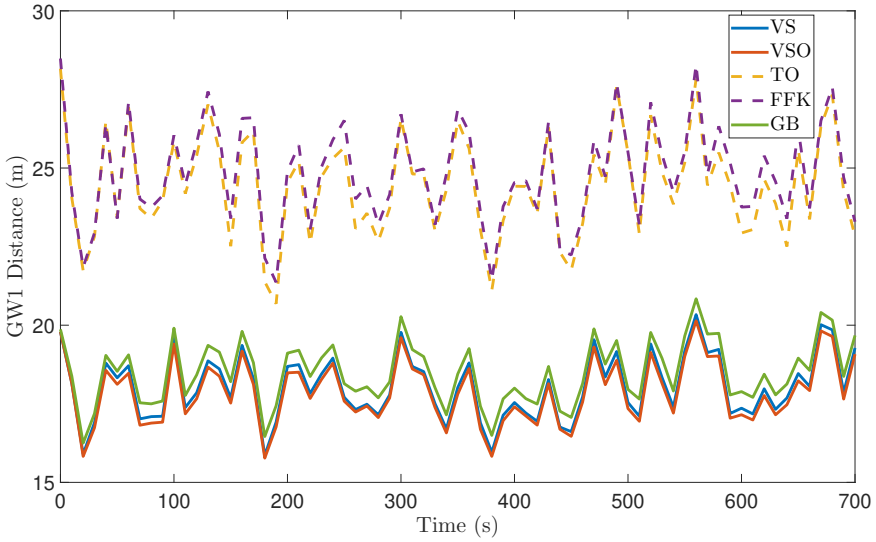

Figure 5.4: Average GW1 distance for scenario 1. The results of the filtering algorithms are shown with dashed lines while those of the smoothing algorithms are shown with solid lines.

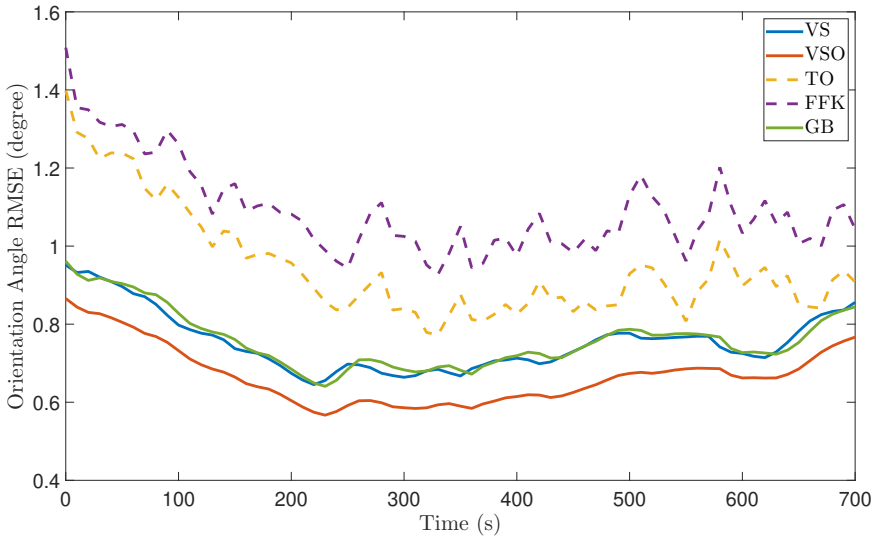

Figure 5.5: Average RMSE(θ) for scenario 1. The results of the filtering algorithms are shown with dashed lines while those of the smoothing algorithms are shown with solid lines.

Combining the RMSE of the orientation with the GW distance of the axis length matrix, we can obtain the GW distance of the extent matrix in Figure 5.6. Compared to the Figure 5.3, the performance differences between the algorithms for the model with and without orientation becomes more evident. The effect of the orientation esti-

mation can also been seen by looking at the TO and GB. Although GB is a smoothing algorithm, its GW_X is so close to the filtering algorithm TO and even exceeds TO's error in some regions.

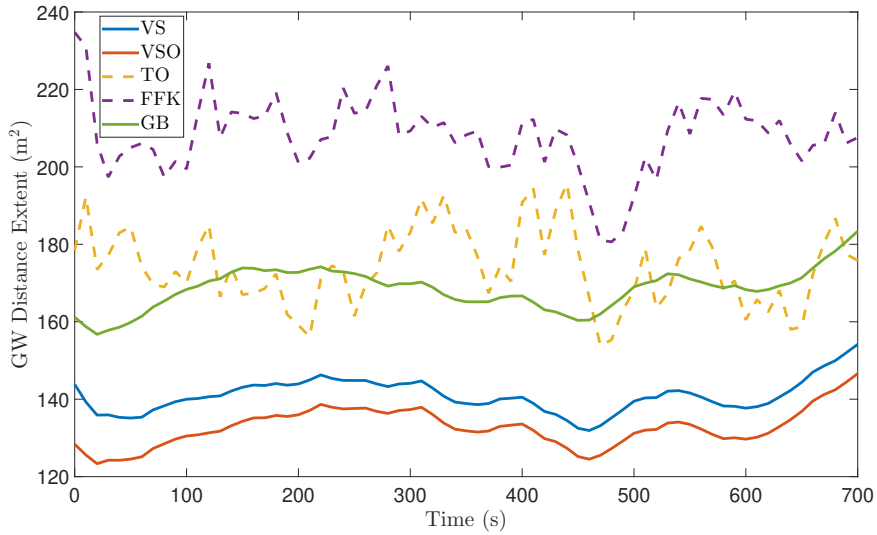

Figure 5.6: Average GW_X distance for scenario 1. The results of the filtering algorithms are shown with dashed lines while those of the smoothing algorithms are shown with solid lines.

The average GW2 distance, which shows the overall error of the kinematic state and extent matrix errors, is given in Figure 5.7. Similar to GW1, the smoothing algorithms show better performance compared to the filtering algorithms. Also, the variational algorithms have less error within their respective group.

The overall performance of the algorithms is reported as the average GW distance and orientation RMSE in Table 5.1. The proposed smoothers VS and VSO achieves the best results followed by the 3$^{\mathrm{rd}}$ smoother GB. This result demonstrate the superiority of the smoothing algorithms over filtering algorithms. In addition, the advantage of the variational technique can be observed by looking at the results of TO and FFK or VS and GB. TO outperforms FFK while VS and VSO outperform GB thanks to their iterative structure. Estimating the orientation separately does not provide an exceptional performance differentiation due to the target's nearly constant velocity behavior in this scenario.

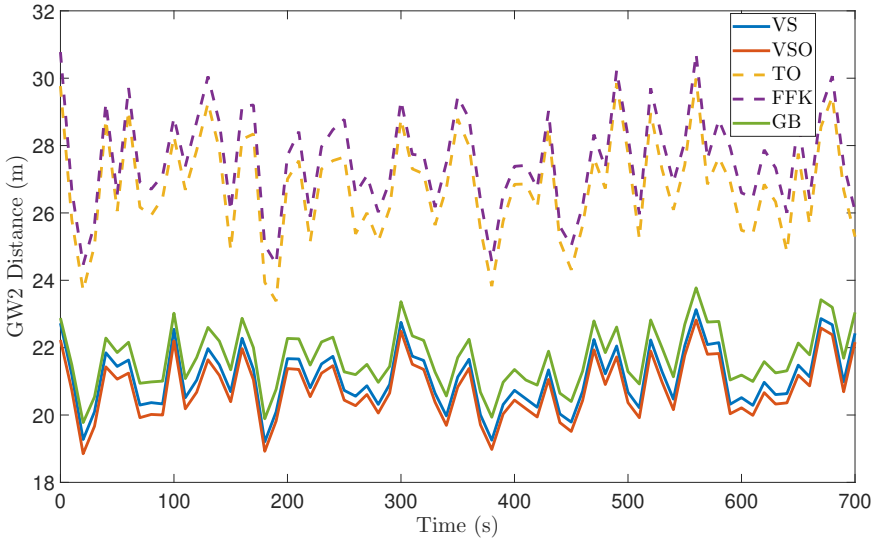

Figure 5.7: Average GW2 distance for scenario 1. The results of the filtering algorithms are shown with dashed lines while those of the smoothing algorithms are shown with solid lines.

Table 5.1: Average GW distance and RMSE results of Scenario 1.

	GW_x [m^2]	GW_Λ [m^2]	GW1 [m]	RMSE(θ) [°]	GW_X [m^2]	GW2 [m]
VS	338.12	38.78	15.81	0.67	123.52	18.50
VSO	334.14	36.63	15.66	0.60	116.28	18.23
TO	639.27	68.84	21.42	0.85	152.61	23.48
FFK	639.49	86.57	21.77	0.98	182.35	24.16
GB	331.81	55.54	16.21	0.68	147.68	19.03

5.1.5 Effects of Low Number of Measurements

In order to see the effect of number of measurements on algorithm performance, the simulations are repeated using an average of 3 measurements at each scan. The typical estimates of the algorithms are illustrated in Figure 5.8 and the overall error metric results are given in Table 5.2.

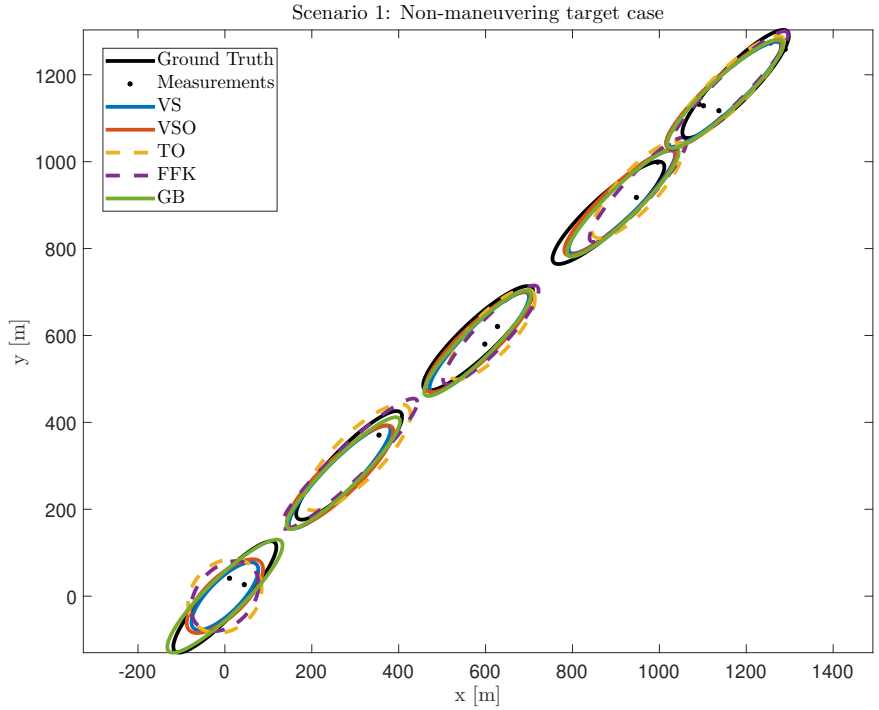

Figure 5.8: An example target trajectory of scenario 1 with an average of 3 measurements at each scan. The results of the filtering algorithms are shown with dashed lines while those of the smoothing algorithms are shown with solid lines.

The same conclusions as the previous subsection can be made based on the results except that the performance of all algorithms degrades due to the low number of measurements. It is clear that the performance, especially the position estimation performance, of the filtering algorithms is reduced more than the smoothing algorithms.

Table 5.2: Average GW distance and RMSE results of Scenario 1 with an average of 3 measurements at each scan.

	GW_x [m^2]	GW_Λ [m^2]	GW1 [m]	RMSE(θ) [°]	GW_X [m^2]	GW2 [m]
VS	964.30	257.22	28.97	2.13	296.77	29.69
VSO	920.27	195.03	27.51	1.88	226.19	28.12
TO	2110.16	258.11	39.14	1.88	292.72	39.65
FFK	2123.29	447.95	41.37	2.34	494.32	41.98
GB	915.55	282.46	28.87	2.09	320.32	29.56

5.2 Scenario 2: Maneuvering Target Case

The maneuvering target scenario, which is a widely used target trajectory in extended target tracking simulations, contains two 90° and one 45° turns. The target moves on straight paths in between these turns and it has a constant velocity of $13.88\,\mathrm{m/s}$ throughout the trajectory. The major and minor axis lengths of the target are equal to $\lambda_1 = 170\,\mathrm{m}$ and $\lambda_2 = 40\,\mathrm{m}$, respectively.

The true target position starts at the origin with an orientation angle of $\theta_0 = \pi/4$. It moves on a straight path for 20 samples and then starts a 45° turn in counter clockwise direction. This turn takes 11 samples. After moving straight on $+x$ direction for 10 samples again, the target starts its second turn. The target completes this 90° turn in 11 samples. Then, it moves in $+y$ direction for 20 samples and starts its last turn of 90° which also lasts 11 samples. The target moves on $-x$ direction for 20 samples and completes the trajectory of 103 samples in total. The sampling time is $T = 10\,\mathrm{s}$ throughout the trajectory (see Figure 3.2).

5.2.1 Measurement Generation

At each MC run, the measurements $\mathcal{Y}_k = \{y_k^j\}_{j=1}^{m_k}$ are generated with a Gaussian distribution around the true target. Setting the measurement parameters as

$$m_k = \max(2, m \sim \text{Poisson}(20)), \tag{5.16a}$$

$$R = 50^2 I_2 \mathrm{m}^2, \tag{5.16b}$$

$$s = 0.25, \tag{5.16c}$$

$$H = [I_2\ 0_{2\times 2}], \tag{5.16d}$$

each measurement can be generated as

$$y_k^j \sim \mathcal{N}(y_k^j; Hx_k, sX_k + R). \tag{5.17}$$

5.2.2 Model and Initial Density Parameters

The kinematic state mean and covariance are initialized as

$$x_{0|-1} = \begin{bmatrix} 0\,\mathrm{m} & 0\,\mathrm{m} & 10\,\mathrm{m/s} & -10\,\mathrm{m/s} \end{bmatrix}^{\mathsf{T}}, \tag{5.18a}$$

$$P_{0|-1} = \mathrm{diag}\left(\begin{bmatrix} 900\,\mathrm{m}^2 & 900\,\mathrm{m}^2 & 16\,\mathrm{m}^2/\mathrm{s}^2 & 16\,\mathrm{m}^2/\mathrm{s}^2 \end{bmatrix} \right). \tag{5.18b}$$

The state transition matrix and gain matrix of the process noise can be written as

$$A_x = \begin{bmatrix} I_2 & TI_2 \\ 0_2 & I_2 \end{bmatrix}, \quad B = \begin{bmatrix} T^2 I_2 \\ TI_2 \end{bmatrix}, \tag{5.19}$$

where $Q_x = \sigma^2 I_2$ is the process noise covariance with $\sigma = \sqrt{10}\ \mathrm{m/s}^2$.

For the models with orientation, the system model parameters are set to $A_\theta = 1$ and $Q_\theta = 0.01\,\mathrm{rad}^2$. The scale and shape estimates are initialized as

$$\alpha^{1,2}_{0|-1} = [2\ 2]^{\mathsf{T}}, \tag{5.20a}$$

$$\beta^{1,2}_{0|-1} = [100^2\ 100^2]^{\mathsf{T}}. \tag{5.20b}$$

The mean and covariance of the orientation are initialized as

$$\theta_{0|-1} = -\pi/6\ \mathrm{rad}, \tag{5.21a}$$

$$\Theta_{0|-1} = 1\,\mathrm{rad}^2. \tag{5.21b}$$

In order to initialize all of the algorithms with equivalent extent estimates, the expected value of the initial extent X_0 is calculated using (5.8). The degrees of freedom and scale matrix parameters of the extent state for the model without orientation are initialized as

$$\nu_{0|-1} = 7, \tag{5.22a}$$

$$V_{0|-1} = X_0(\nu_0 - 2n_y - 2). \tag{5.22b}$$

The forgetting factor is set to $\psi = 0.75$ for the VS, VSO, TO algorithms. The equivalent time constant parameter τ of FFK, and the degrees of freedom parameter n_k of GB can be obtained by using (5.10).

5.2.3 Error Metrics

The same error metrics given in Subsection 5.1.3 are also used in scenario 2.

5.2.4 Simulation Results

Similar to Subsection 5.1.4, the average error metrics are provided for each time step in Figure 5.10-Figure 5.15. Furthermore, an example trajectory of the target with the algorithm estimates is given in Figure 5.9. In all plots, filtering algorithms are given in dashed lines, while smoothing algorithms are shown in solid lines.

When we analyze the estimation results of the algorithms in Figure 5.9, we can clearly observe that the estimation performance of the algorithms without orientation degrades during the turns. However, because TO and VSO algorithms treat the orientation as a separate variable to be estimated, these algorithms can maintain their good estimation performance during the turns as well.

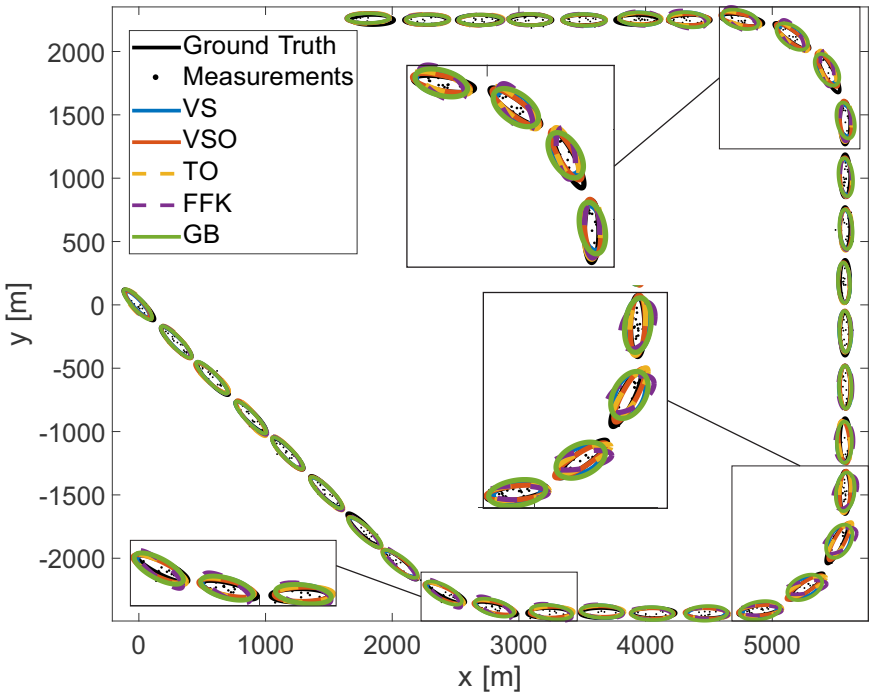

Figure 5.9: An example trajectory of scenario 2. The results of the filtering algorithms are shown with dashed lines while those of the smoothing algorithms are shown with solid lines.

The average positional error of the algorithms can be seen in Figure 5.10. As expected, the form of the average GW_x is similar to the one in scenario 1 because the performance of the positional estimation is not affected substantially from the maneuvering behavior of the target. As before, the smoothing algorithms VS, VSO, and

GB performs better than the filtering algorithms TO and FFK.

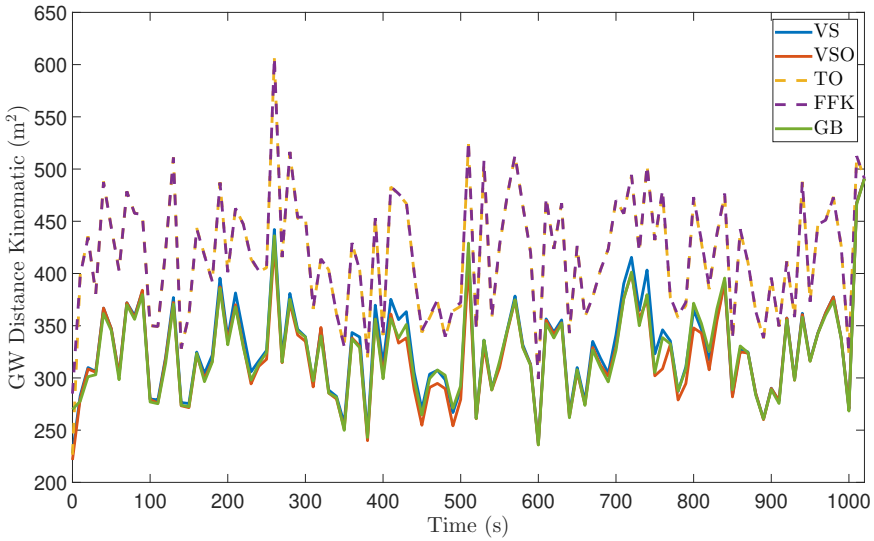

Figure 5.10: Average GW_x distance for scenario 2. The results of the filtering algorithms are shown with dashed lines while those of the smoothing algorithms are shown with solid lines.

In Figure 5.11, the average error of the axis length matrices are given. The average GW_Λ of the VS, GB, and FFK algorithms rises dramatically during the turns since these algorithms do not treat the orientation as a separate random variable. The proposed VS algorithm shows the poorest performance among all algorithms. When we investigate the reason for this behavior, we found out that the Beta-Bartlett backward recursion is not the best selection for the scenarios where target shows rapid orientation changes. Since the Beta-Bartlett smoother makes a backward update which is similar to the covariance update of the covariance intersection algorithm [42], it is not capable of reacting fast enough to the orientation variations. Nevertheless, the proposed VSO algorithm achieves the best performance among all algorithms thanks to its model with orientation.

The average GW1 distance of the algorithms is given in Figure 5.12. Since the VS, FFK, and GB show significant performance degradation in axis length estimation, this behavior dominates the overall GW1 distance as well. Different than the axis length estimation errors, the errors of VS and GB are closer to the error of FFK because of their lower kinematic estimation error. The performance of the proposed algorithm VSO is better than other algorithms throughout the trajectory.

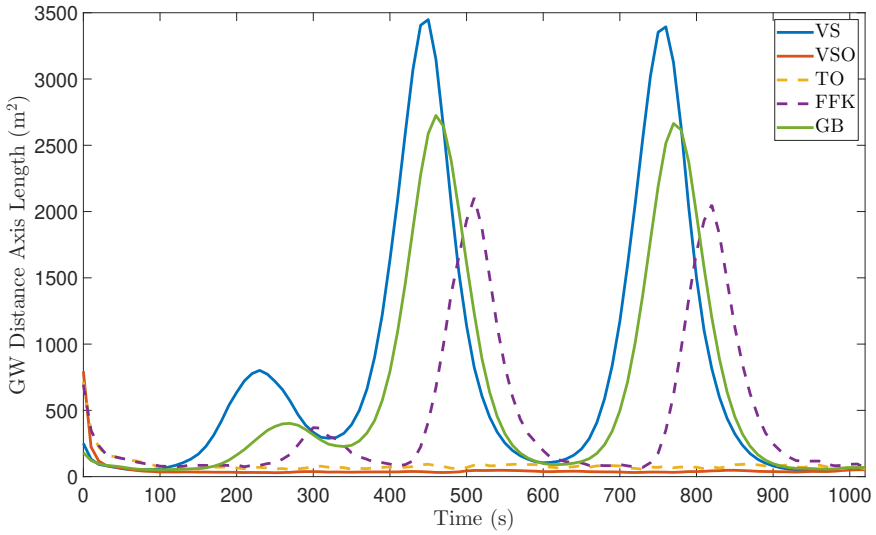

Figure 5.11: Average GW_{Λ} distance for scenario 2. The results of the filtering algorithms are shown with dashed lines while those of the smoothing algorithms are shown with solid lines.

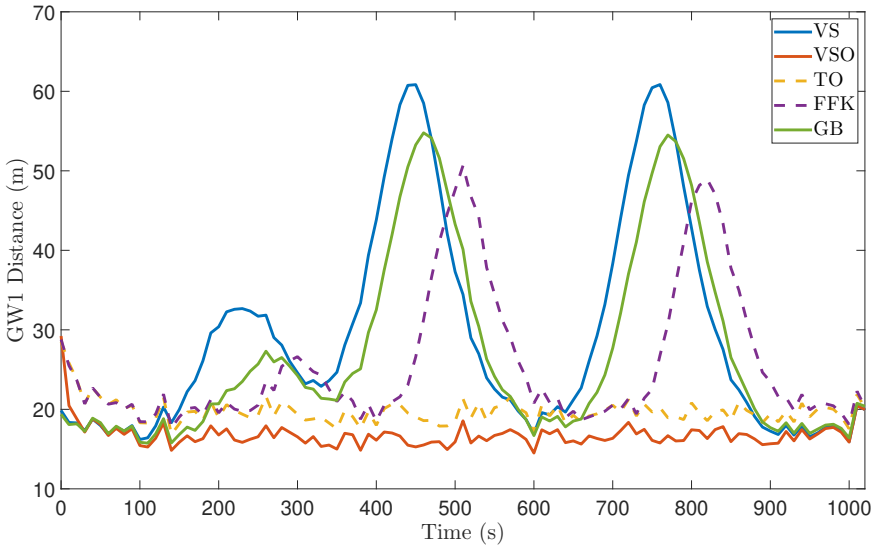

Figure 5.12: Average GW1 distance for scenario 2. The results of the filtering algorithms are shown with dashed lines while those of the smoothing algorithms are shown with solid lines.

As it can be seen in Figure 5.13, the average RMSE of the orientation angle exhibit considerable increase during the turns. The orientation error does not change much during the turns for VSO and TO as expected. FFK shows the worst orientation angle estimation performance among all algorithms for the model without orientation. This

behavior can be explained by the improvement of the smoothing algorithms.

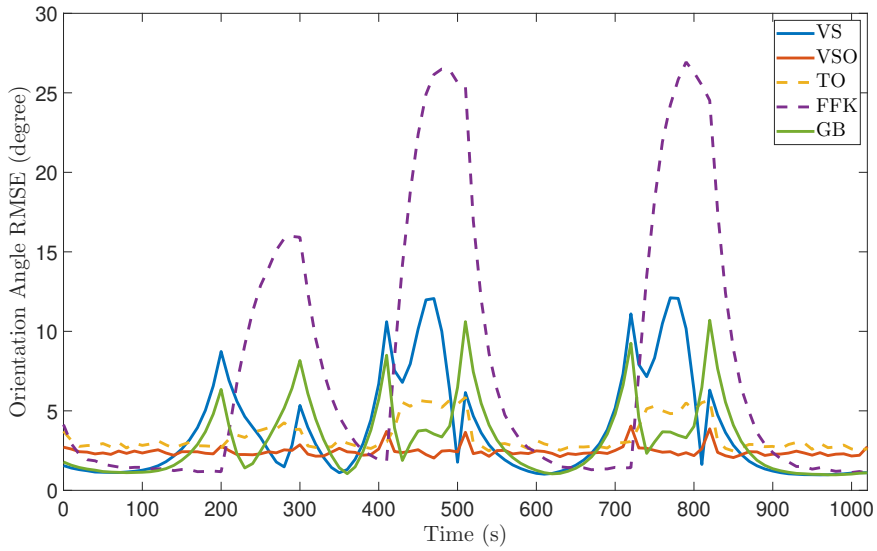

Figure 5.13: Average RMSE(θ) for scenario 2. The results of the filtering algorithms are shown with dashed lines while those of the smoothing algorithms are shown with solid lines.

The angle estimation errors also affect the average extent estimation performance of the algorithms. Looking into Figure 5.14, we can see that the GW_X distance of FFK is the most among all algorithms which is different than the result of GW_Λ. Although axis length errors of the VS and GB are higher than FFK, their performance are better in estimating the extent because the extent matrix also holds orientation angle information.

Integrating the effect of angle errors into the extent and consequently into the GW2 distance, we can see the resulting performance of the algorithms in Figure 5.15. The GW2 distance results show that VSO performs best in scenario 2 without being affected by the turns of the target. Conversely, FFK shows the worst performance as expected because it does not estimate the orientation separately and does not conduct a backward recursion.

The overall GW distance and RMSE error metric results of scenario 2 is given in Table 5.3. The proposed algorithm VSO outperforms the alternative algorithms in the literature for all of the error parameters. This result demonstrates the advantage of estimating the orientation angle as another random variable for maneuvering target

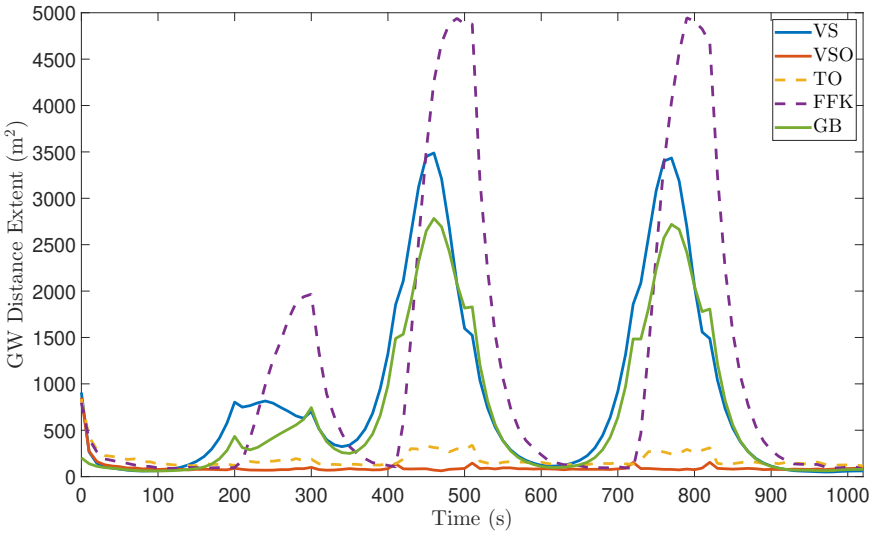

Figure 5.14: Average GW_X distance for scenario 2. The results of the filtering algorithms are shown with dashed lines while those of the smoothing algorithms are shown with solid lines.

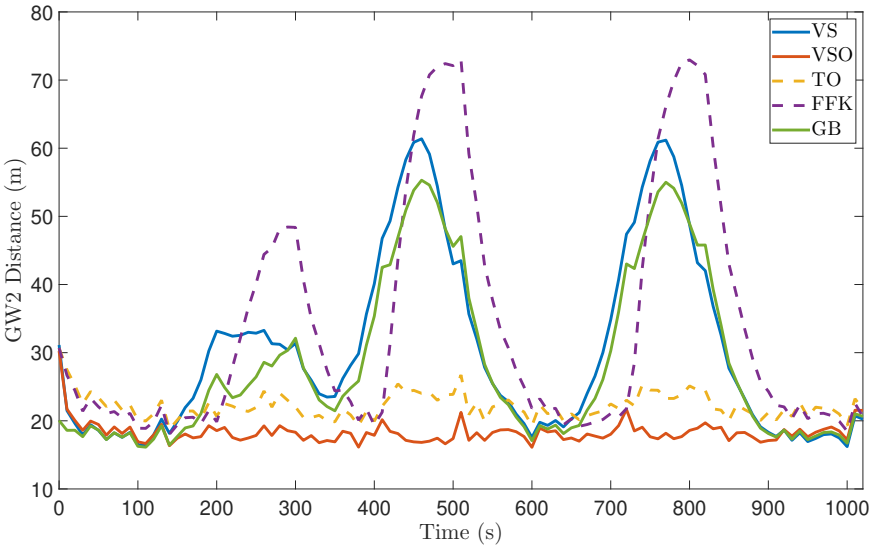

Figure 5.15: Average GW2 distance for scenario 2. The results of the filtering algorithms are shown with dashed lines while those of the smoothing algorithms are shown with solid lines.

scenarios. If we group the algorithms for the model with or without orientation, it is also obvious that the smoothing algorithms are superior to the filtering algorithms within their respective groups.

Table 5.3: Average GW distance and RMSE results of scenario 2.

	GW_x [m^2]	GW_Λ [m^2]	GW1 [m]	RMSE(θ) [°]	GW_X [m^2]	GW2 [m]
VS	333.74	757.33	29.19	4.81	822.56	30.31
VSO	327.67	46.85	16.90	2.44	92.02	18.43
TO	420.61	86.35	19.89	3.51	181.09	22.38
FFK	421.76	407.75	25.68	11.56	1136.04	34.12
GB	330.46	588.35	26.83	3.77	670.53	28.27

5.3 Computational Load

The average run times of the algorithms (per time step) when number of iterations is 1, 5, and 20, are given in Table 5.4. The superior performance of VSO comes at a cost of computation time due to its iterative nature and additional smoothing algorithm computations. It is worth noting that the proposed algorithms VS and VSO converge in an average of 5 iterations but the simulations were performed in 20 iterations to make sure of the convergence. Therefore, the run time can be decreased by optimizing the number of iterations depending on the application.

Table 5.4: Average algorithm run times (per time step) with different number of iterations.

	VS	VSO	TO	FFK	GB
Run time with a single iteration (ms)	1.32	2.37	2.13	0.69	1.13
Run time with 5 iterations (ms)	4.61	7.74	5.93	0.69	1.13
Run time with 20 iterations (ms)	17.07	29.14	19.37	0.69	1.13

CHAPTER 6

CONCLUSION AND FUTURE WORK

In this thesis study, we proposed two Bayesian smoothers for extended targets whose extents are modeled with random matrices. By leveraging the previous work in the literature and using the variational approximation technique, we derived the smoothing algorithms for the extended target models without and with the orientation. The performances of both algorithms were presented using the simulation results on two different ETT scenarios. In order to make a proper evaluation, we included the algorithms of Feldmann *et al.* (FFK), Tuncer and Özkan (TO), and Granström and Bramstång (GB), into both simulation scenarios and compared their results with the proposed algorithms.

The results of the first scenario with a non-maneuvering extended target show that under no model mismatch between the target motion and the filter model, the proposed variational smoothers outperform the previous algorithms in the literature. The results obtained in this scenario also provide information about the estimation accuracy of the algorithms with different methodologies. In general, the smoothing algorithms perform better than the filtering algorithms. When we consider the algorithms separately according to their subgroups of smoother or filter, it can be declared that the overall results of the algorithms for the model with orientation are slightly better than the results of the algorithms for the model without orientation. This difference becomes more evident for the results of orientation and consequently extent estimation. Moreover, the algorithms which utilize the variational method outperform the other algorithms. Although all algorithms show satisfactory results due to the non-maneuvering extended target scenario, the results validate the capability of the proposed algorithms and their methodologies.

In the second scenario with maneuvering extended target, the algorithms which do not treat the orientation as a separate random variable show performance degradation during the turns. However, the proposed VSO algorithm is able to retain its performance throughout the trajectory and achieves the best performance among all algorithms. In a more general sense, the results of this simulation scenario show that the explicit modelling of the orientation becomes much more effective for tracking maneuvering targets. The remaining comparison results of the second scenario are in analogy with the results of the first scenario.

Overall, VSO algorithm proposed in this thesis study shows superior performance in both non-maneuvering and maneuvering target scenarios and it can be used in target tracking applications with batch data processing where estimation accuracy is more important than the computational resources. On the other hand, if the application is not accuracy driven or the target is expected to have a relatively constant orientation, the proposed VS algorithm can be used in order to allocate less computational resources to smoothing.

In closing, the future work may perform experiments using real data collected from aerial vehicles with image sensors, radars or lidars. Moreover, the performance of the VS algorithm can be improved further by introducing different transition models than the matrix Beta-Bartlett stochastic evolution model. Furthermore, adaptive estimation of forgetting factor used in filtering and smoothing might be considered for maneuvering ETT. Similar smoothing algorithms can be derived for other ETT models, such as Gaussian process ETT model [23] or random hypersurface model [9], [22], and also for other measurement models involving measurements which are nonlinear functions of the kinematic state, e.g., range, angle measurements [43]. The scope of the proposed smoothers can also be expanded for multiple target tracking [44], [45] applications.

REFERENCES

[1] M. Feldmann, D. Fränken, and W. Koch, "Tracking of extended objects and group targets using random matrices," *IEEE Trans. on Signal Processing*, vol. 59, pp. 1409–1420, Apr 2011.

[2] K. Granström and J. Bramstång, "Bayesian smoothing for the extended object random matrix model," *IEEE Trans. on Signal Processing*, vol. 67, pp. 3732–3742, Jul 2019.

[3] B. Tuncer and E. Özkan, "Random matrix based extended target tracking with orientation: A new model and inference," *IEEE Trans. on Signal Processing*, vol. 69, pp. 1910–1922, Oct 2020.

[4] K. Granström, S. Reuter, D. Meissner, and A. Scheel, "A multiple model PHD approach to tracking of cars under an assumed rectangular shape," in *Proc. Int. Conf. Inf. Fusion*, (Salamanca,Spain), Jul 2014.

[5] C. Knill, A. Scheel, and K. Dietmayer, "A direct scattering model for tracking vehicles with high-resolution radars," *IEEE Intelligent Vehicles Symposium*, pp. 298–303, Jun 2016.

[6] J. W. Koch, "Bayesian approach to extended object and cluster tracking using random matrices," *IEEE Trans. on Aerospace and Electronic Systems*, vol. 44, no. 3, pp. 1042–1059, 2008.

[7] D. J. Salmond and M. C. Parr, "Track maintenance using measurements of target extent," *IEEE Radar, Sonar, Navig.*, vol. 150, no. 6, pp. 385–395, 2003.

[8] B. Ristic and D. Salmond, "A study of a nonlinear filtering problem for tracking an extended target," in *Proc. 7th Int. Conf. Inf. Fusion*, pp. 503–509, 2004.

[9] M. Baum and U. Hanebeck, "Shape tracking of extended objects and group targets with star-convex rhms," in *Proc. Int. Conf. Inf. Fusion*, (Chiago, IL, USA), pp. 338–345, Jul 2011.

[10] X. Cao, J. Lan, and X. R. Li, "Extension-deformation approach to extended object tracking," in *Proc. Int. Conf. Inf. Fusion*, pp. 1185–1192, Jul 2016.

[11] D. Salmond and N. Gordon, "Group and extended object tracking," *IEE Colloquium on Target Tracking: Algorithms and Applications*, pp. 16/1–16/4, 1999.

[12] J. C. Dezert, "Tracking maneuvering and bending extended target in cluttered environment," in *Signal and Data Processing of Small Targets*, vol. 3373, (Orlando, FL, USA), pp. 283–294, Apr 1998.

[13] M. Bühren and B. Yang, "Simulation of automotive radar target lists using a novel approach of object representation," in *IEEE Intelligent Vehicles Symposium*, (Tokyo, Japan), pp. 314–319, June 2006.

[14] J. Gunnarsson, L. Svensson, L. Danielsson, and F. Bengtsson, "Tracking vehicles using radar detections," in *IEEE Intelligent Vehicles Symposium*, pp. 296–302, Jun 2007.

[15] L. Hammarstrand, L. Svensson, F. Sandblom, and J. Sörstedt, "Extended object tracking using a radar resolution model," *IEEE Trans. on Aerospace and Electronic Systems*, vol. 48, pp. 2371–2386, Jul 2012.

[16] K. Gilholm, S. Godsill, S. Maskell, and D. Salmond, "Poisson models for extended target and group tracking," in *Proc. of Signal and Data Processing of Small Targets*, vol. 5913, (San Diego, CA, USA), pp. 230–241, SPIE, Aug 2005.

[17] K. Gilholm and D. Salmond, "Spatial distribution model for tracking extended objects," *IEE Proc. of Radar, Sonar and Navigation*, vol. 152, pp. 364–371, Oct 2005.

[18] B. D. O. Anderson and J. B. Moore, *Optimal Filtering*. Englewood Cliffs: Prentice-Hall, 1979.

[19] J. Vermaak, N. Ikoma, and S. Godsill, "Sequential Monte Carlo framework for extended object tracking," *IEE Proc. of Radar, Sonar, Navigation*, vol. 152, pp. 353–363, Oct 2005.

[20] D. Angelova and L. Mihaylova, "Extended object tracking using Monte Carlo methods," *IEEE Trans. on Signal Processing*, vol. 56, pp. 825–832, Feb 2008.

[21] D. Angelova, L. Mihaylova, N. Petrov, and A. Gning, "A convolution particle filtering approach for tracking elliptical extended objects," pp. 1542–1549, Jul 2013.

[22] M. Baum and U. Hanebeck, "Random hypersurface models for extended object tracking," *IEEE Int. Symposium on Signal Processing and Information Technology (ISSPIT)*, pp. 178–183, Dec 2009.

[23] N. Wahlström and E. Özkan, "Extended target tracking using Gaussian processes," *IEEE Trans. on Signal Processing*, vol. 63, pp. 4165–4178, Aug 2015.

[24] E. Özkan, N. Wahlström, and S. J. Godsill, "Rao-Blackwellised particle filter for star-convex extended target tracking models," in *Proc. of the 19th Int. Conf. Inf. Fusion*, pp. 1193–1199, Jul 2016.

[25] M.Kumru and E. Özkan, "3D extended object tracking using recursive Gaussian processes," in *Proc. of the 21st Int. Conf. Inf. Fusion*, pp. 1–8, 2018.

[26] M. Feldmann and D. Fränken, "Tracking of extended objects and group targets using random matrices - A new approach," in *Proc. Int. Conf. Inf. Fusion*, (Cologne,Germany), Jul 2008.

[27] M. Feldmann and D. Fränken, "Advances on tracking of extended objects and group targets using random matrices," in *Proc. Int. Conf. Inf. Fusion*, (Seattle,USA), pp. 1029–1036, Jul 2009.

[28] U. Orguner, "A variational measurement update for extended target tracking with random matrices," *IEEE Trans. on Signal Processing*, vol. 60, pp. 3827–3834, Jul 2012.

[29] K. Granström, M. Baum, and S. Reuter, "Extended object tracking: Introduction, overview and applications," *Journal of Advances in Information Fusion*, vol. 12, pp. 139–174, December 2016.

[30] M. Baum, M. Feldmann, D. Fränken, D. Hanebeck, and W. Koch, "Extended object and group tracking: A comparison of random matrices and random hypersurface models," *IEEE ISIF Workshop on Sensor Data Fusion: Trends, Solut., App.,*, Oct 2010.

[31] A. K. Gupta and D. Nagar, *Matrix Variate Distributions*. Boca Raton, FL: Chapman & Hall/CRC Press, 1999.

[32] K. Granström and U. Orguner, "New prediction for extended targets with random matrices," *IEEE Trans. on Aerospace and Electronic Systems*, vol. 50, pp. 1577–1589, Jul 2014.

[33] J. Lan and X. R. Li, "Tracking of extended object or target group using random matrix: New model and approach," *IEEE Trans. on Aerospace and Electronic Systems*, vol. 52, Dec 2016.

[34] G. Vivone, K. Granström, P. Braca, and P. Willett, "Multiple sensor measurement updates for the extended target tracking random matrix model," *IEEE Trans. on Aerospace and Electronic Systems*, vol. 63, pp. 2544–2558, Oct 2017.

[35] H. E. Rauch, C. T. Striebel, and F. Tung, "Maximum likelihood estimates of linear dynamic systems," *Journal of the American Institute of Aeronautics and Astronautics*, vol. 3, pp. 1445–1450, Aug 1965.

[36] C. M. Bishop, *Pattern Recognition and Machine Learning*. New York: Springer, 2007.

[37] D. Tzikas, A. Likas, and N. Galatsanos, "The variational approximation for Bayesian inference," *IEEE Trans. on Signal Processing*, vol. 25, pp. 131–146, Nov 2008.

[38] T. Ardeshiri, E. Özkan, U. Orguner, and F. Gustafsson, "Approximate Bayesian smoothing with unknown process and measurement noise covariances," *IEEE Trans. on Signal Processing*, vol. 22, pp. 2450–245, Dec 2015.

[39] T. M. Cover and J. Thomas, *Elements of Information Theory*. Hoboken, NJ: Wiley, 2006.

[40] C. M. Carvalho and M. West, "Dynamic matrix-variate graphical models," *Bayesian Anal*, vol. 2, pp. 69–98, 2007.

[41] S. Yang, K. Granström, and M. Baum, "Metrics for performance evaluation of elliptic extended object tracking methods," in *IEEE Int. Conf. Multisensor Fusion Integration Intell. Syst.*, pp. 523–528, Sept 2016.

[42] S. Julier and J. Uhlmann, "A non-divergent estimation algorithm in the presence of unknown correlations," in *Proc. of the 1997 American Control Conference (Cat. No.97CH36041)*, vol. 4, pp. 2369–2373 vol.4, 1997.

[43] K. Granström, A. Natale, P. Braca, G. Ludeno, and F. Serafino, "Gamma Gaussian inverse Wishart probability hypothesis density for extended target tracking using X-band marine radar data," *IEEE Trans. on Geoscience and Remote Sensing*, vol. 53, no. 12, pp. 6617–6631, 2015.

[44] Y. Bar-Shalom, "Extension of the probabilistic data association filter to multitarget tracking," in *Proc. of the 5th Symposium on Nonlinear Estimation*, (San Diego, CA, USA), Sep 1974.

[45] R. Mahler, *Statistical Multisource-Multitarget Information Fusion*. Norwood, MA, USA: Artech House, 2007.

APPENDIX A

PARAMETER MAPPING BETWEEN INVERSE WISHART AND INVERSE GAMMA DISTRIBUTIONS

Inverse Wishart Distribution

Consider $d \times d$ SPD matrix X_k describing the current ellipsodial object extension

$$\mathcal{IW}(X;\nu,V) \propto |X|^{-\frac{1}{2}\nu}\text{etr}(-\frac{1}{2}X^{-1}V) \tag{A.1}$$

If $d = 1$, we get

$$\mathcal{IW}(X;\nu,V) \propto X^{-\frac{1}{2}\nu}\text{etr}(-\frac{1}{2}\frac{V}{X}) \tag{A.2}$$

Inverse Gamma Distribution

$$\mathcal{IG}(X;\alpha,\beta) \propto X^{-\alpha-1}\exp -\frac{\beta}{X} \tag{A.3}$$

By equating (A.2) and (A.3) we get

$$\alpha \equiv \frac{1}{2}\nu - 1 \tag{A.4}$$

$$\beta \equiv \frac{V}{2} \tag{A.5}$$

Expected value of the inverse Wishart density is

$$\mathbb{E}[X] = \frac{V}{\nu - 2d - 2} \tag{A.6}$$

When $d = 1$ we have

$$\mathbb{E}[X] = \frac{V}{\nu - 4} \tag{A.7}$$

Expected value of the inverse Gamma density is

$$\mathbb{E}[X] = \frac{\beta}{\alpha - 1} = \frac{\frac{\nu}{2}}{\frac{1}{2}\nu - 2} = \frac{V}{\nu - 4} \tag{A.8}$$

equal to the expected value found in (A.7), which confirms the (A.4) and (A.5).

APPENDIX B

EQUIVALENT PREDICTION PARAMETERS IN DIFFERENT EXTENDED TARGET TRACKING MODELS

The algorithms covered in this study use different prediction parameters to govern the time evolution of the extent. The variational smoother with and without orientation algorithms proposed in this thesis along with Özkan and Tuncer's variational filter with orientation algorithm [28] use the forgetting factor ψ; Feldmann et al.'s filter [28] uses the time constant τ; and Granström's smoother [2] uses the degrees of freedom parameter n_k. In order to be able to compare the performances of these algorithms properly, we need to find the condition when these parameters are equivalent.

The equivalent time constant parameter can be obtained by equating $\exp(-T/\tau) = \psi$. This equality gives

$$\tau = -\frac{T}{\ln \psi}. \tag{B.1}$$

In [2], a Wishart extension transition density is used, i.e.,

$$p(X_{k+1}|X_k) = \mathcal{W}\left(X_{k+1}; n_k, \frac{X_k}{n_k}\right). \tag{B.2}$$

When we consider the Beta-Bartlett transition we have

$$V_{k+1|k} = \psi V_{k|k}, \tag{B.3a}$$

$$\nu_{k+1|k} - 2n_y - 2 = \psi(\nu_{k|k} - 2n_y - 2). \tag{B.3b}$$

Given the assumption that "the extension does not tend to change over time", we have

$$X_{k+1|k} = \frac{V_{k+1|k}}{\nu_{k+1|k} - 2n_y - 2} = \frac{\psi V_{k|k}}{\psi(\nu_{k|k} - 2n_y - 2)} = X_{k|k}. \tag{B.4}$$

Let us now compare the variances:

$$\sigma_{k|k}^{ij} \triangleq \text{var}\left([X_k]_{ij}|\mathcal{Y}_{0:K}\right), \tag{B.5a}$$

$$\sigma_{k+1|k}^{ij} \triangleq \text{var}\left([X_k+1]_{ij}|\mathcal{Y}_{0:K}\right). \tag{B.5b}$$

Using the inverse Wishart covariance formulas given in [31] we can write

$$\begin{aligned}\text{var}\left([X]_{ij}\right) &= \frac{2(\nu-2n_y-2)^{-1}[V]_{ij}^2+[V]_{ii}[V]_{jj}+[V]_{ij}^2}{(\nu-2n_y-1)(\nu-2n_y-2)(\nu-2n_y-4)},\\ &= \frac{\left(\frac{2}{\nu-2n_y-2}+1\right)[V]_{ij}^2+[V]_{ii}[V]_{jj}}{(\nu-2n_y-1)(\nu-2n_y-2)(\nu-2n_y-4)},\\ &= \frac{(\nu-2n_y)[V]_{ij}^2+(\nu-2n_y-2)[V]_{ii}[V]_{jj}}{(\nu-2n_y-1)(\nu-2n_y-2)(\nu-2n_y-4)}.\end{aligned} \tag{B.6a}$$

This gives

$$\begin{aligned}var\left([X]_{ii}\right) &= \frac{2(\nu-2n_y-1)[V]_{ii}^2}{(\nu-2n_y-1)(\nu-2n_y-2)^2(\nu-2n_y-4)},\\ &= \frac{2[V]_{ii}^2}{(\nu-2n_y-2)^2(\nu-2n_y-4)}.\end{aligned} \tag{B.7a}$$

Assuming that $\nu-2n_y-2 \gg 2$, we can write

$$var\left([X]_{ii}\right) \approx \frac{2[V]_{ii}^2}{(\nu-2n_y-2)^3}. \tag{B.8}$$

We can now write $\sigma_{k|k}$ and $\sigma_{k+1|k}$ as

$$\sigma_{k|k}^{ii} = \frac{2[V_{k|k}]_{ii}^2}{(\nu_{k|k}-2n_y-2)^3}, \tag{B.9a}$$

$$\begin{aligned}\sigma_{k+1|k}^{ii} &= \frac{2[V_{k+1|k}]_{ii}^2}{(\nu_{k+1|k}-2n_y-2)^3}\\ &= \frac{2\psi^2[V_{k|k}]_{ii}^2}{\psi^3(\nu_{k|k}-2n_y-2)^3} = \frac{\sigma_{k|k}^{ii}}{\psi}\end{aligned} \tag{B.9b}$$

As a result, Beta-Bartlett prediction keeps the mean the same while increasing the variance by a factor of $1/\psi$.

In [32], it is given that with the Wishart transition distribution

$$p(X_{k+1}|X_k) = \mathcal{W}\left(X_{k+1}; n_k, \frac{X_k}{n_k}\right), \tag{B.10}$$

we have

$$\sigma_{k+1|k}^{ii} = \left(1+\frac{\nu_{k|k}-2n_y-2}{n_k}\right)\sigma_{k|_k}. \tag{B.11}$$

Therefore, we can find the equivalent n_k as

$$\left(1 + \frac{\nu_{k|k} - 2n_y - 2}{n_k}\right) = \frac{1}{\psi}, \tag{B.12}$$

which gives

$$n_k = \frac{\nu_{k|k} - 2n_y - 2}{\frac{1}{\psi} - 1}. \tag{B.13}$$