INCORPORATING TRAJECTORY INFORMATION IN RANDOM MATRIX ELLIPTICAL EXTENDED TARGET TRACKING

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

BY

KURTULUŞ KEREM ŞAHİN

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

SEPTEMBER 2024

Approval of the thesis:

INCORPORATING TRAJECTORY INFORMATION IN RANDOM MATRIX ELLIPTICAL EXTENDED TARGET TRACKING

submitted by **KURTULUŞ KEREM ŞAHİN** 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. Naci Emre Altun
Dean, Graduate School of **Natural and Applied Sciences** _____

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

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

Examining Committee Members:

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

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

Assist. Prof. Dr. Sinan Yıldırım
Industrial Engineering, Sabancı University _____

Date:02.09.2024

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: Kurtuluş Kerem Şahin

Signature :

ABSTRACT

INCORPORATING TRAJECTORY INFORMATION IN RANDOM MATRIX ELLIPTICAL EXTENDED TARGET TRACKING

Şahin, Kurtuluş Kerem
M.S., Department of Electrical and Electronics Engineering
Supervisor: Assoc. Prof. Dr. Emre Özkan

September 2024, 71 pages

This thesis focuses on Extended Target Tracking (ETT) using Random Matrix Methods (RMM), which provide enhanced estimations of target size and movement in tracking systems. Traditional methods often miss crucial trajectory details, which, if considered, could improve tracking performance.

To address this issue, we have developed two new RMM-based models. The first, the trajectory-aligned model is designed for targets moving in a consistent direction, ensuring that the orientation aligns with the trajectory. The second, the drifting model is for targets whose orientation deviates from their heading direction. Utilizing the variational Bayes (VB) method, we obtain posterior densities by performing analytical and iterative steps for both models. This methodological choice ensures that our models not only deliver precise tracking results but also operate efficiently in real-time applications.

Extensive testing on both simulated and real-world data has proven that our models effectively outperform current methods in handling drifting and trajectory-aligned targets. These tests confirm the flexibility and efficiency of our models under diverse

conditions. The demonstrated success of our models in both simulated and real environments underscores their potential to significantly enhance current standards in extended target tracking.

Keywords: Extended Target Tracking, Random Matrix, Maneuvering Motion Models, Variational Bayes, Automotive Radars

ÖZ

RASLANTISAL MATRİS TABANLI ELİPTİK GENİŞLETİLMİŞ HEDEF TAKİBİ YÖNTEMLERİNE YÖRÜNGE BİLGİSİ EKLENMESİ

Şahin, Kurtuluş Kerem
Yüksek Lisans, Elektrik ve Elektronik Mühendisliği Bölümü
Tez Yöneticisi: Doç. Dr. Emre Özkan

Eylül 2024 , 71 sayfa

Bu tezde, Genişletilmiş Hedef Takibi (GHT) üzerine odaklanmaktadır ve hedef takip sistemlerinde hedef boyutu ve hareketinin gelişmiş tahminleri Raslantısal Matris Yöntemleri (RMY) kullanılarak sağlanmaktadır. Geleneksel yöntemler genellikle kritik yörünge detaylarını göz ardı ederken; bu detaylar dikkate alındığında, hedef takip performansı artırılabilir.

Bu sorunu ele almak için iki yeni RMY tabanlı model geliştirilmiştir. İlk olarak geliştirilen yörüngeye hizalı model, hedefler sürekli bir yönde hareket ettirildiğinde yörünge ile yönelim açısının uyumlu olmasını sağlamak üzere tasarlanmıştır. İkinci olarak geliştirilen yörüngeden sapma modeli, yönelimi başlangıç doğrultusundan sapma gösteren hedefler için uygundur. Varyasyonel Bayes (VB) yöntemi kullanılarak her iki model için analitik ve yinelemeli adımlar gerçekleştirilerek sonrasal olasılık yoğunlukları elde edilmektedir. Bu metodolojik seçim, modellerin sadece kesin izleme sonuçları sunmakla kalmayıp aynı zamanda gerçek zamanlı uygulamalarda etkin bir şekilde çalışmasını sağlamaktadır.

Hem benzetimli hem de gerçek dünya verileri üzerinde yapılan kapsamlı testlerle, modellerin mevcut yöntemleri sapma ve yörüngeye hizalı davranışlar gösteren hedeflerin takibinde etkili bir şekilde aştığı kanıtlanmıştır. Bu testler, modellerin çeşitli koşullar altında esneklik ve verimliliğini teyit etmektedir. Modellerin hem benzetimli hem de gerçek ortamlarda gösterdiği başarı, mevcut genişletilmiş hedef takip standartlarını önemli ölçüde geliştirme potansiyelini vurgulamaktadır.

Anahtar Kelimeler: Genişletilmiş Hedef Takibi, Raslantısal Matris, Manevralı Hareket Modelleri, Varyasyonel Bayes, Otomotiv Radarlar

To my loving family and friends

ACKNOWLEDGMENTS

I would like to express my gratitude to my supervisor, Assoc. Prof. Dr. Emre Özkan, for his invaluable guidance, support, and encouragement throughout the course of this research.

I am profoundly thankful to Ali Emre Balcı, not only for his enduring friendship and support but also for co-authoring the paper that forms the foundation of this study. His contributions were essential to the development of the work presented here.

Special thanks go to Mehmet Çetinkaya for the numerous constructive mathematical discussions that greatly enhanced the quality of this work. I am also grateful to the METU Sensor Fusion Lab team for providing a collaborative and intellectually stimulating environment. Their support and the constructive discussions we had played a significant role in the completion of this thesis.

Finally, this research was supported by the Scientific and Technological Research Council of Turkey (TÜBİTAK) Division of Supporting Scientists Programmes (BİDEB) under the 2211 - National Graduate Scholarship Programme.

TABLE OF CONTENTS

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

LIST OF ALGORITHMS

ALGORITHMS

LIST OF ABBREVIATIONS

ABBREVIATIONS

EKF	Extended Kalman Filter
ELBO	Evidence Lower Bound
ETT	Extended Target Tracking
i.i.d	Independent Identically Distributed
IG	Inverse-Gamma
IMM	Interacting Multiple Model
IW	Inverse-Wishart
KF	Kalman Filter
KL	Kullback-Liebler
RM	Random Matrix
UKF	Unscented Kalman Filter
VB	Variational Bayesian

LIST OF SYMBOLS

SYMBOLS

$\mathbb{S}^d_{++}$ Set of d dimensional real symmetric positive definite matrices.

$\mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ Multivariate Gaussian probability density with mean $\boldsymbol{\mu}$ and symmetric positive definite covariance matrix $\boldsymbol{\Sigma}$.

$$\mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma}) = |2\pi\boldsymbol{\Sigma}|^{-0.5} \exp(-0.5(\boldsymbol{x} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\cdot)).$$

$\boldsymbol{T}_k$ Rotation matrix in two dimensions depending on the respective orientation angle θ_k is given as

$$\boldsymbol{T}_k \triangleq \boldsymbol{T}(\theta_k) = \begin{bmatrix} \cos(\theta_k) & -\sin(\theta_k) \\ \sin(\theta_k) & \cos(\theta_k) \end{bmatrix}.$$

$\dot{\boldsymbol{T}}_k$ Derivative of the rotation matrix in two dimensions depending on the respective orientation angle θ_k is given as

$$\dot{\boldsymbol{T}}_k \triangleq \frac{\partial \boldsymbol{T}_k}{\partial \theta_k} = \dot{\boldsymbol{T}}(\theta_k) = \begin{bmatrix} -\sin(\theta_k) & -\cos(\theta_k) \\ \cos(\theta_k) & -\sin(\theta_k) \end{bmatrix}.$$

$\mathcal{IG}(\gamma; \alpha, \beta)$ Inverse-Gamma probability density with shape parameter α and scale parameter β,

$$\mathcal{IG}(\gamma; \alpha, \beta) = \frac{\beta^n}{\Gamma(\alpha)} \gamma^{-(1+\alpha)} \exp\left(-\beta/\gamma\right).$$

$\mathcal{Y}_k$ Set of measurements obtained at time k $\mathcal{Y}_k = \{\boldsymbol{y}_k^1, \boldsymbol{y}_k^2, \ldots, \boldsymbol{y}_k^{m_k}\}$.

$\mathcal{Y}_{i:j}$ Collection of all measurements sets gathered within the time interval $[i, j]$.

$KL(p \mid\mid q)$ Kullback-Leibler divergence between the densities $p(x)$ and $q(x)$,

$$KL(p \mid\mid q) = \int p(x) \log\left(\frac{p(x)}{q(x)}\right) \mathrm{d}x.$$

$(\boldsymbol{z}-\boldsymbol{\mu})^T(\mathcal{M})(\cdot)$	Shorthand notation for the quadratic form $(\boldsymbol{z}-\boldsymbol{\mu})^T(\mathcal{M})(\boldsymbol{z}-\boldsymbol{\mu})$
$(\boldsymbol{z})\pm(\cdot)^T$	Shorthand notation for the expression $(\boldsymbol{z})\pm(\boldsymbol{z})^T$
Tr	$\mathrm{Tr}(A)=\sum_{i=1}^{n}a_{ii}$ Trace of matrix $A=\{a\}_{i,j}$
$\mathbb{E}_q[\cdot]$	The expectation with respect to the density q
$\mathbb{E}_{\setminus q}[\cdot]$	Expectation that is taken over all the densities except q.
$\nabla f(\boldsymbol{x}_0)$	Gradient of function $f(\boldsymbol{x}):\mathbb{R}^n\rightarrow\mathbb{R}$ evaluated at the point $\boldsymbol{x}=\boldsymbol{x}_0$, the transpose of the Jacobian, $\nabla f(\boldsymbol{x}_0)\triangleq\left[\frac{\partial f}{\partial x_1}(\boldsymbol{x}_0)\quad\ldots\quad\frac{\partial f}{\partial x_n}(\boldsymbol{x}_0)\right]^T.$
$\boldsymbol{F}_{cv}$, $\boldsymbol{Q}_{cv}$	The near constant velocity (CV) model matrices parameterized by sampling time T $\boldsymbol{F}_{cv}\triangleq\begin{bmatrix}1&T\\0&1\end{bmatrix},\quad\boldsymbol{Q}_{cv}\triangleq\begin{bmatrix}T^3/3&T^2/2\\T^2/2&T\end{bmatrix}.$
blkdiag	Block diagonal operator defined as; $\mathrm{blkdiag}:\mathbb{R}^{n\times m}\times\mathbb{R}^{l\times t}\rightarrow\mathbb{R}^{(n+l)\times(m+t)}$ with the following output; $\mathrm{blkdiag}(\boldsymbol{A},\boldsymbol{B})=\begin{bmatrix}\boldsymbol{A}&\boldsymbol{0}\\\boldsymbol{0}&\boldsymbol{B}\end{bmatrix}$
diag	Diagonal operator defined as $\mathrm{diag}:\mathbb{R}^n\rightarrow\mathbb{R}^{n\times n}$ with the following output; $\mathrm{diag}(v)=A,\quad\{a\}_{i,i}=v_i$
$\otimes$	is the Kronecker product operator $\otimes:\mathbb{R}^{n\times m}\times\mathbb{R}^{l\times t}\rightarrow\mathbb{R}^{(n+l)\times(m+t)}$ with the following output; $\boldsymbol{A}\otimes\boldsymbol{B}=\begin{bmatrix}a_{11}\boldsymbol{B}&\cdots&a_{1n}\boldsymbol{B}\\\vdots&\ddots&\vdots\\a_{m1}\boldsymbol{B}&\cdots&a_{mn}\boldsymbol{B}\end{bmatrix}$

CHAPTER 1

INTRODUCTION

1.1 Motivation

Extended Target Tracking (ETT) has become increasingly critical in modern tracking applications due to advancements in sensor technologies that allow for the detection of complex object shapes and their dynamic behavior. Traditional tracking methods often assume that an object is represented as a single point. Still, this assumption falls short in scenarios where the object's spatial extent influences its movement and interaction with the environment. ETT addresses this by estimating not only the position and velocity of a target but also its size, shape, and orientation, providing a more comprehensive understanding of the target's behavior.

Despite the progress in ETT, a significant limitation in the current literature is treating the rotation information of the target's extent as uncorrelated from its dynamic behavior. This simplifying assumption neglects the potential insights gained from correlating the target's kinematic state, such as velocity and heading, with its rotational dynamics. If this correlation were accounted for, the tracker could potentially achieve more accurate predictions of the target's future state, especially in scenarios where the target exhibits continuous angular velocity dynamics. For instance, in cases where a target's heading direction is aligned with its extent, such as in vehicles or commercial aircraft, maintaining this correlation could enhance the tracker's ability to predict sharp maneuvers and changes in direction. This raises the question of whether incorporating the correlation between the target's heading and the orientation of its spatial extent could yield better tracking performance.

1.2 Target Tracking

In traditional tracking methods, known as single point target tracking, an object is assumed to generate at most one measurement per time frame or scan. This assumption holds in scenarios where the object is distant from the sensor, such as radar-based air surveillance, where the object is treated as a point due to sensor resolution limitations.

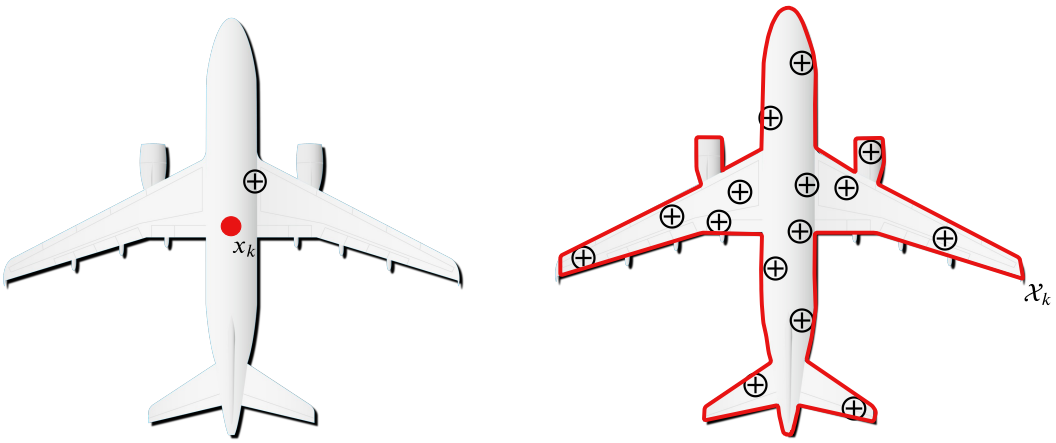
Various methods have been developed under this assumption. Single model-based methods include the Kalman Filter, Extended Kalman Filter (EKF) [1], and Unscented Kalman Filter (UKF) [2]. For scenarios involving multiple motion models, Generalized Pseudo-Bayesian Filters (GPB-I and GPB-II) [3] and Interacting Multiple Model (IMM) [3] filters are used. In more complex or nonlinear scenarios, Particle Filtering [4] and Markov Chain Monte Carlo (MCMC) techniques [5] are used. These methods effectively handle non-linearities and non-Gaussian processes, making them suitable for more challenging tracking environments.

However, as sensor technologies have advanced, the assumption that an object generates only a single measurement per scan has become increasingly limiting. In many modern applications, such as autonomous driving or robotics, objects often occupy several sensor resolution cells, generating multiple measurements. This has led to the development of Extended Target Tracking methods, where the goal is to estimate not only the object's position and velocity but also its size and shape, which may vary over time [6–8]. A comparison of two target tracking paradigms is illustrated in Figure 1.1.

1.3 Extended Target Tracking

Various approaches have been proposed thus far to address the ETT problem in learning the underlying shape of targets in simple mathematical forms. The complexity of shape modeling directly impacts the tracking method's accuracy and computational demands [7, 8].

Extended target tracking methods can be classified based on how they represent the object's shape. The most straightforward approach to shape representation assumes



(a) Point target tracking

(b) Extended target tracking

Figure 1.1: Comparison between point target tracking and extended target tracking. In (a), the aircraft is represented as a single point (x_k), with sensor measurements shown as crosshairs $(\oplus)$. In (b), the aircraft is treated as an extended object, with its outline represented in red $(\mathcal{X}_k)$. The distributed sensor measurements $(\oplus)$ across the target's surface capture the extended nature of the tracking scenario.

that the object can be modeled as a fundamental geometric shape, such as a circle, ellipse, or rectangle. These methods are computationally efficient and are often sufficient when tracking objects like vehicles or pedestrians, where the shape can be approximated with a simple geometric form.

- Ellipse Models: A common example is the Random Matrix Approach [9–25] where the object's extent is modeled as a covariance matrix or deterministic approaches [26, 27] which uses semi-axes lengths of an ellipse. This method is particularly effective in scenarios where the object's shape is roughly elliptical, such as pedestrian or marine vessel tracking.

- Multiellipsoidal Models: In scenarios where a single ellipse cannot adequately represent the target's shape, multiellipsoidal models are employed. These models represent the target extent using multiple ellipses, each corresponding to a different part of the target's surface. This approach provides a more detailed and accurate representation, especially in complex tracking environments where the target might have an irregular shape or consists of multiple distinct

parts [17, 20, 28].

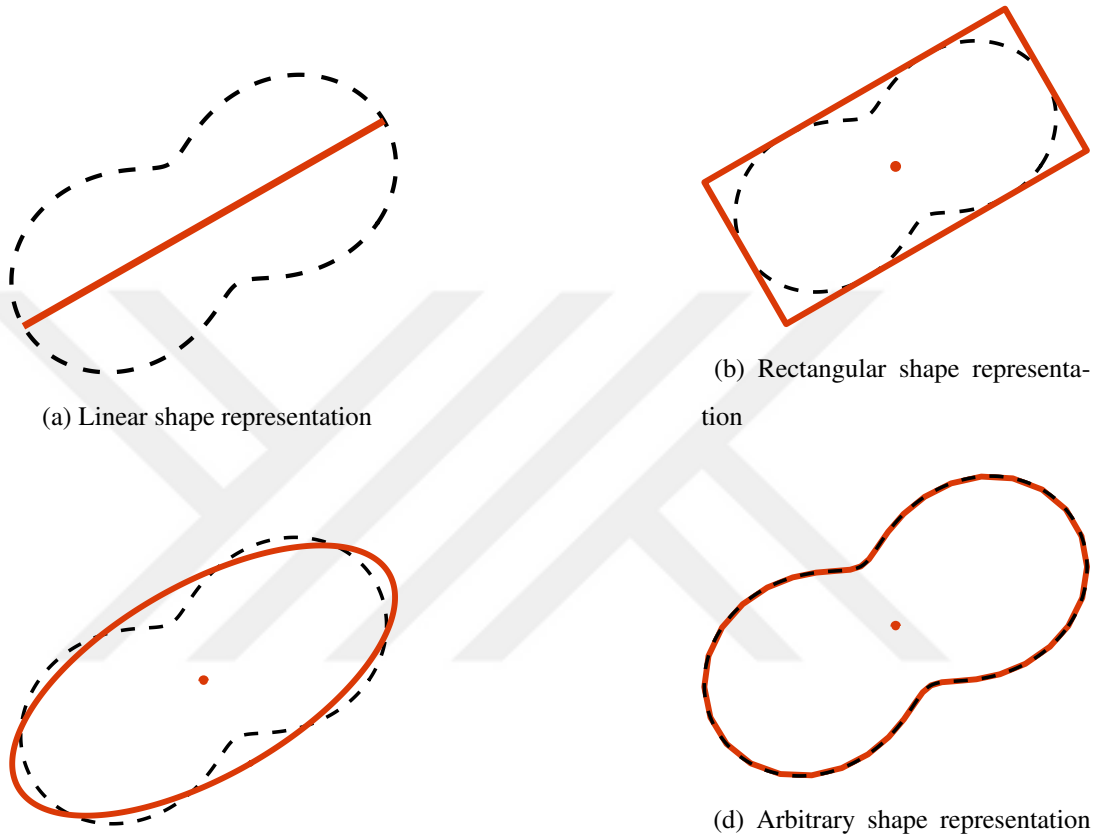
- Rectangle Models: When tracking vehicles, a rectangle may represent the object's extent. This is a good approximation for many practical applications like car tracking using LIDAR [29, 30].

More advanced shape models are used for more complex objects whose shapes cannot be accurately captured by simple geometric forms. These models allow for arbitrary shape representation and are often employed when the object's shape is irregular or requires high accuracy. Figure 1.2 presents some examples of extent representations.

- Random Hypersurface Models: These models represent the object's shape as a parametric curve, allowing for the tracking of objects with complex and varying boundaries [31,32]. These extents can also be modeled using Gaussian processes to handle the spatial correlation between different parts of the object's boundary. Symmetry constraints can be easily enforced wherever needed [33–35].
- Spline Models: Using spline representations in target tracking allows for modeling object contours with high precision and flexibility. This method has proven particularly useful in tracking elongated or deformable objects in aerial and ground-based sensor data [36–38].

1.4 RM-based ETT

RM-based methods assume an elliptical shape for the target extent. These methods represent an extended target with a Gaussian kinematic state vector and a symmetric positive definite extent matrix. The kinematic state vector mostly holds information about the target position and velocity, while the extent matrix describes the target shape as an ellipse. While earlier methods define an inverse-Wishart (IW) prior for the extent matrix [9–14], some recent studies use inverse-Gamma (IG) priors together with a rotation matrix [19]. The rich literature on RM-based ETT is initiated by the seminal work of Koch [9], where the author proposed the first solution for the RM-based ETT problem under zero additive sensor noise assumption. Later, Feldmann



(a) Linear shape representation

(b) Rectangular shape representation

(c) Elliptical shape representation

(d) Arbitrary shape representation (Gaussian Process)

Figure 1.2: Different styles of shape representation in ETT. The black dashed line represents the ground truth object, while the red solid lines with red dots depict the extent models and their center locations.

et al. [10] provided a solution that can take additive Gaussian sensor noise into account. To incorporate evolving extent dynamics into the additive Gaussian sensor noise model, [12, 13] proposed approximating the predicted densities using moment matching, Kullback-Leibler (KL) divergence minimization, and numerical optimization techniques with approximate transition densities. The authors of [15, 21] later proposed an analytical model for predicted densities with artificial transition models using non-central Inverse Wishart probability densities. Optimization-based methods proposed by [11, 19] suggest utilizing the Variational Bayesian (VB) optimization method to improve the accuracy of the correction step of the estimation process.

Even with these advancements in RM-based ETT, a significant gap in the existing research is the common practice of treating the rotation information of the target's extent as separate from its dynamic behavior. Building upon the foundation laid by these RM-based methods, our work addresses this overlooked aspect, specifically integrating the rotation information of the target's extent with its dynamic behavior. In the following section, we outline the key contributions of our approach and how it advances the current state of elliptical ETT.

1.5 Contributions

In this thesis, we propose two models that are frequently observed in tracking problems but have been largely overlooked in the literature to differentiate the dependence between the kinematic state and the extent orientation. The first model covers the targets that have their motion direction mostly parallel to their extent heading directions, such as cars, bikes, and trains [39–41]. We denote targets that satisfy this property as "trajectory-aligned" targets. The second model is tailored for targets that have mostly independent motion directions and extent headings, such as maritime vessels, unmanned aerial vehicles, insects, and microscopic life forms [19, 20, 42, 43]. These targets tend to exhibit extent rotations that are weakly coupled with their state dynamics. We categorize these targets as "drifting" targets. This notably apparent difference necessitates two separate models, each of which can exploit these dependencies and achieve higher accuracy by performing adequate updates. In our derivations, we perform variational inference and obtain approximate posteriors via com-

putationally efficient, analytical, iterative steps. Our derivations for each target type incorporate orientation dynamics and account for relevant correlation structures. This approach not only enhances the accuracy of inferring the proposed models but also poses greater challenges compared to previous variational methods that ignore this dependence [11, 19].

We also show that ignoring the correlation between the target trajectory and shape, as in [19], will result in poor performance, especially for fast-maneuvering targets. Our simulated and real data experiments involving video and radar tracking applications validate such performance losses and demonstrate significant performance gains of our methods over state of the art.

1.6 Organization of the Thesis

This thesis is organized in the following structure: Chapter 2 delves into the foundational concepts and mathematical frameworks essential to understanding extended target tracking. This includes discussions on Bayesian inference, state space models, Kalman filtering, and an introduction to variational Bayesian methods that form the basis of the proposed models. Chapter 3 presents a detailed mathematical derivation of two innovative algorithms designed to track trajectory-aligned and drifting targets accurately. This chapter covers each model's underlying assumptions, algorithmic steps, and theoretical justifications. Chapter 4 outlines the experimental setup, utilizing both simulated and real datasets for validation. It offers a critical analysis of the performance of the proposed models compared to existing methods, highlighting improvements in tracking accuracy through various metrics and scenarios. Chapter 5 emphasizes significant advancements in tracking technologies. It summarizes key results, discusses their implications for the field, and reflects on the strengths and limitations of the work presented. This chapter also explores potential ways to enhance further the methodologies introduced in this study.

CHAPTER 2

THEORETICAL BACKGROUND

2.1 Bayesian Inference

Bayesian inference is a statistical method that uses Bayes' rule to update the probability estimate for a hypothesis as more evidence becomes available [44]. This approach is particularly useful in dynamic systems, where we aim to estimate states that evolve in time. Bayesian inference provides a powerful framework for combining prior knowledge with new observations to make informed predictions of the systems' states.

Bayesian inference revolves around updating our beliefs based on new data [45]. The key formula at the heart of Bayesian inference is Bayes' rule, which is stated as follows

$$p(\boldsymbol{x}_k|\mathbf{y}_k) = \frac{p(\mathbf{y}_k|\boldsymbol{x}_k)p(\boldsymbol{x}_k)}{p(\mathbf{y}_k)}, \tag{2.1}$$

where

- $p(\mathbf{x}_k|\mathbf{y}_k)$ is the posterior probability density function (pdf),
- $p(\mathbf{y}_k|\boldsymbol{x}_k)$ is the likelihood of the observation $\mathbf{y}_k$ given the state $\boldsymbol{x}_k$,
- $p(\boldsymbol{x}_k)$ is the prior probability of the state, and
- $p(\mathbf{y}_k)$ is the marginal likelihood, a normalizing constant.

2.1.1 State-Space Models

A state-space model allows us to describe a system using latent state variables and observations. The model is typically composed of two equations.

- **State Transition Equation:** Describes the evolution of the state vector as

$$\boldsymbol{x}_{k+1} = f(\boldsymbol{x}_k, \boldsymbol{\omega}_k). \tag{2.2}$$

- **Measurement Equation:** Describes the relation of observations with the state variables as

$$\mathbf{y}_k = h(\boldsymbol{x}_k, \boldsymbol{\nu}_k), \tag{2.3}$$

  where $\boldsymbol{x}_k$ is the state at time k, $\boldsymbol{\omega}_k$ represents the process noise, $\mathbf{y}_k$ is the observation at time k and $\boldsymbol{\nu}_k$ denotes the measurement noise.

2.1.2 Recursive Bayesian Estimation

Recursive Bayesian estimation is a key technique used for state estimation in dynamic systems. It involves two steps:

1. The time update step involves using the state transition equation to predict the next state based on the current state estimate utilizing the Chapman-Kolmogorov equation as

$$p(\boldsymbol{x}_{k+1}|\mathbf{y}_{1:k}) = \int p(\boldsymbol{x}_{k+1}|\boldsymbol{x}_k) p(\boldsymbol{x}_k|\mathbf{y}_{1:k}) \, \mathrm{d}\boldsymbol{x}_k. \tag{2.4}$$

2. Measurement update step refines this prediction based on the new observation using Bayes' rule given by

$$p(\boldsymbol{x}_k|\mathbf{y}_{1:k}) = \frac{p(\mathbf{y}_k|\boldsymbol{x}_k) p(\boldsymbol{x}_k|\mathbf{y}_{1:k-1})}{p(\mathbf{y}_k|\mathbf{y}_{1:k-1})}. \tag{2.5}$$

These steps are repeated at each time step to provide a continuous estimate of the state as new observations are made.

Bayesian inference and state-space models form a framework for understanding and predicting the behavior of dynamic systems. Through recursive Bayesian estimation, we can continuously update our understanding of a system's state, making it indispensable in fields such as robotics, economics, and beyond.

2.2 Kalman Filtering

Kalman filtering is a specific case of Bayesian inference applied to linear state-space models with Gaussian noise. It provides optimal and efficient computational means for recursive Bayesian estimation under linearity and Gaussianity assumptions.

2.2.1 Linear State-Space Model

The following equations define the linear state-space model:

- **State Transition Equation:**

$$\boldsymbol{x}_{k+1} = \boldsymbol{F}\boldsymbol{x}_k + \boldsymbol{G}\boldsymbol{\omega}_k, \quad \boldsymbol{\omega}_k \overset{i.i.d}{\sim} \mathcal{N}(\mathbf{0}, \boldsymbol{Q}), \tag{2.6}$$

- **Measurement Equation:**

$$\mathbf{y}_k = \boldsymbol{H}\boldsymbol{x}_k + \boldsymbol{L}\boldsymbol{\nu}_k, \quad \boldsymbol{\nu}_k \overset{i.i.d}{\sim} \mathcal{N}(\mathbf{0}, \boldsymbol{R}), \tag{2.7}$$

- **Initial Probability Distribution:**

$$\boldsymbol{x}_0 \sim \mathcal{N}(\boldsymbol{x}_0; \hat{\boldsymbol{x}}_0, \boldsymbol{P}_0), \tag{2.8}$$

  where $x_k \in \mathbb{R}^n$ is the state at time k, $\mathbf{y}_k \in \mathbb{R}^m$ is the measurement at time k, $\boldsymbol{\omega}_k \in \mathbb{R}^s$ is the zero mean i.i.d process noise with multivariate normal distribution with covariance $\boldsymbol{Q} \in \mathbb{R}^{s\times s}$, $\boldsymbol{\nu}_k \in \mathbb{R}^t$ is the zero mean i.i.d measurement (sensor) noise with multivariate normal distribution with covariance $\boldsymbol{R} \in \mathbb{R}^{t\times t}$, $\boldsymbol{F} \in \mathbb{R}^{n\times n}$ is the state transition matrix, $\boldsymbol{G} \in \mathbb{R}^{s\times n}$ is the process noise matrix, $\boldsymbol{H} \in \mathbb{R}^{n\times m}$ is the measurement matrix, and $\boldsymbol{L} \in \mathbb{R}^{t\times m}$ is the measurement noise matrix.

2.2.2 Kalman Filter Algorithm

The Kalman filter performs the recursive estimation in two main steps: time update and measurement update.

2.2.2.1 Time Update Step

Given $p(\boldsymbol{x}_{k-1}|\mathbf{y}_{k-1}) = \mathcal{N}\left(\boldsymbol{x}_{k-1}; \hat{\boldsymbol{x}}_{k-1|k-1}, \boldsymbol{P}_{k-1|k-1}\right)$, the posterior density at time $k-1$, the prediction consists of calculating the sufficient statistics of the prior density at k as

$$\hat{\boldsymbol{x}}_{k|k-1} = \boldsymbol{F}\hat{\boldsymbol{x}}_{k-1|k-1}, \tag{2.9}$$

$$\boldsymbol{P}_{k|k-1} = \boldsymbol{F}\boldsymbol{P}_{k-1|k-1}\boldsymbol{F}^T + \boldsymbol{G}\boldsymbol{Q}\boldsymbol{G}^T, \tag{2.10}$$

where $\hat{\boldsymbol{x}}_{k|k-1}$ is the predicted state estimate, $\boldsymbol{P}_{k|k-1}$ is the predicted error covariance, and $\boldsymbol{Q}$ is the process noise covariance.

2.2.2.2 Measurement Update Step

In the measurement update step, the predicted state mean and covariance are corrected using the new observation $\mathbf{y}_k$ as

$$\hat{\boldsymbol{x}}_{k|k} = \hat{\boldsymbol{x}}_{k|k-1} + \boldsymbol{K}_k(\mathbf{y}_k - \boldsymbol{H}\hat{\boldsymbol{x}}_{k|k-1}), \tag{2.11}$$

$$\boldsymbol{P}_{k|k} = \boldsymbol{P}_{k|k-1} - \boldsymbol{K}_k\boldsymbol{S}_k\boldsymbol{K}_k^T, \tag{2.12}$$

where $\boldsymbol{S}_k$ is the innovation covariance and $\boldsymbol{K}_k$ is the Kalman gain, which are calculated as

$$\boldsymbol{S}_k = \boldsymbol{H}\boldsymbol{P}_{k|k-1}\boldsymbol{H}^T + \boldsymbol{L}\boldsymbol{R}\boldsymbol{L}^T, \text{ and,} \tag{2.13}$$

$$\boldsymbol{K}_k = \boldsymbol{P}_{k|k-1}\boldsymbol{H}^T{\boldsymbol{S}_k}^{-1}, \tag{2.14}$$

respectively. $\hat{\boldsymbol{x}}_{k|k}$ is the updated state estimate, $\boldsymbol{P}_{k|k}$ is the updated state covariance, and $\boldsymbol{R}$ is the observation noise covariance.

2.3 Extended Kalman Filtering

The extended Kalman filter (EKF) is an extension of the Kalman filter that handles nonlinear state-space models. In the EKF, the state transition and observation models are linearized using Taylor series expansions around the current state estimate.

2.3.1 Nonlinear State Space Model

The following equations define the nonlinear state-space model:

- **State Transition Equation:**

$$\boldsymbol{x}_{k+1} = f(\boldsymbol{x}_k, \boldsymbol{\omega}_k) \quad \boldsymbol{\omega}_k \overset{i.i.d}{\sim} \mathcal{N}(\mathbf{0}, \boldsymbol{Q}), \tag{2.15}$$

- **Measurement Equation:**

$$\mathbf{y}_k = \boldsymbol{h}(\boldsymbol{x}_k, \boldsymbol{\nu}_k) \quad \boldsymbol{\nu}_k \overset{i.i.d}{\sim} \mathcal{N}(\mathbf{0}, \boldsymbol{R}), \tag{2.16}$$

  where $\boldsymbol{f}$ is the nonlinear state transition function, $\boldsymbol{h}$ is the nonlinear measurement function.

2.3.2 Taylor Series Expansion

While using the EKF, we linearize the nonlinear functions $\boldsymbol{f}$and $\boldsymbol{h}$ using their first-order Taylor series expansions around the previous posterior mean $\hat{\boldsymbol{x}}_{k-1|k-1}$ for the prediction step and around the predicted mean $\hat{\boldsymbol{x}}_{k|k-1}$ for the update step.

2.3.2.1 State Transition Function Linearization

The state transition function $\boldsymbol{f}(\boldsymbol{x}_k)$ is expanded around $(\boldsymbol{x}_k, \boldsymbol{\omega}_k) = (\hat{\boldsymbol{x}}_{k|k}, \mathbf{0})$:

$$\boldsymbol{f}(\boldsymbol{x}_k, \boldsymbol{\omega}_k) \approx f(\hat{\boldsymbol{x}}_{k|k}, \mathbf{0}) + \boldsymbol{F}_k(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_{k|k}) + \boldsymbol{G}_k\boldsymbol{\omega}_k, \tag{2.17}$$

where $\boldsymbol{F}_k$ is the Jacobian matrix with respect to the state $\boldsymbol{x}_k$ of $\boldsymbol{f}$ evaluated at $\boldsymbol{x}_k = \hat{\boldsymbol{x}}_{k|k}$, $\boldsymbol{\omega}_k = \mathbf{0}$ and $\boldsymbol{G}_k$ is the Jacobian matrix with respect to the process noise $\boldsymbol{\omega}_k$ of $\boldsymbol{f}$

evaluated at $\boldsymbol{x}_k = \hat{\boldsymbol{x}}_{k|k}$, $\boldsymbol{\omega}_k = \mathbf{0}$, that are given as

$$\boldsymbol{F}_k = \left(\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}_k}\right)_{\substack{\boldsymbol{x}_k=\hat{x}_{k|k} \\ \boldsymbol{\omega}_k=\mathbf{0}}}, \quad \boldsymbol{G}_k = \left(\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{\omega}_k}\right)_{\substack{\boldsymbol{x}=\hat{x}_{k|k} \\ \boldsymbol{\omega}_k=\mathbf{0}}}. \tag{2.18}$$

2.3.2.2 Measurement Function Linearization

The measurement function $\boldsymbol{h}(\boldsymbol{x}_k)$ is expanded around $(\boldsymbol{x}_k, \boldsymbol{\nu}_k) = (\hat{\boldsymbol{x}}_{k|k-1}, \mathbf{0})$:

$$\boldsymbol{h}(\boldsymbol{x}_k, \boldsymbol{\nu}_k) \approx \boldsymbol{h}(\hat{\boldsymbol{x}}_{k|k-1}, \mathbf{0}) + \boldsymbol{H}_k(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_{k|k-1}) + \boldsymbol{L}_k\boldsymbol{\nu}_k, \tag{2.19}$$

where $\boldsymbol{H}_k$ is the Jacobian matrix of $\boldsymbol{h}$ with respect to the state $\boldsymbol{x}_k$ evaluated at $\boldsymbol{x}_k = \hat{\boldsymbol{x}}_{k|k-1}$, $\boldsymbol{\nu}_k = \mathbf{0}$ and $\boldsymbol{L}_k$ is the Jacobian matrix of $\boldsymbol{h}$ with respect to the measurement noise $\boldsymbol{\nu}_k$ evaluated at $\boldsymbol{x}_k = \hat{\boldsymbol{x}}_{k|k-1}$, $\boldsymbol{\nu}_k = \mathbf{0}$ that are given as

$$\boldsymbol{H}_k = \left(\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}_k}\right)_{\substack{\boldsymbol{x}_k=\hat{x}_{k|k-1} \\ \boldsymbol{\nu}_k=\mathbf{0}}}, \quad \boldsymbol{L}_k = \left(\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{\nu}_k}\right)_{\substack{\boldsymbol{x}=\hat{x}_{k|k-1} \\ \boldsymbol{\nu}_k=\mathbf{0}}}. \tag{2.20}$$

2.3.3 Extended Kalman Filter Algorithm

The EKF follows the same prediction and update steps as the Kalman filter but uses linearized models where necessary.

2.3.3.1 Time Update Step

The predicted mean and covariance are computed as

$$\hat{\boldsymbol{x}}_{k|k-1} = \boldsymbol{f}(\hat{\boldsymbol{x}}_{k-1|k-1}, \mathbf{0}), \tag{2.21}$$

$$\boldsymbol{P}_{k|k-1} = \boldsymbol{F}_{k-1}\boldsymbol{P}_{k-1|k-1}\boldsymbol{F}_{k-1}^T + \boldsymbol{G}_{k-1}\boldsymbol{Q}\boldsymbol{G}_{k-1}^T, \tag{2.22}$$

respectively.

2.3.3.2 Measurement Update Step

The predicted state is corrected using the new observation:

$$\hat{\boldsymbol{x}}_{k|k} = \hat{\boldsymbol{x}}_{k|k-1} + \boldsymbol{K}_k(\mathrm{y}_k - \boldsymbol{h}(\hat{\boldsymbol{x}}_{k|k-1}, \mathbf{0})), \tag{2.23}$$

$$\boldsymbol{P}_{k|k} = \boldsymbol{P}_{k|k-1} - \boldsymbol{K}_k \boldsymbol{S}_k \boldsymbol{K}_k^T, \tag{2.24}$$

where the innovation covariance and Kalman gain are

$$\boldsymbol{S}_k = \boldsymbol{H}_k \boldsymbol{P}_{k|k-1} \boldsymbol{H}_k^T + \boldsymbol{L}_k \boldsymbol{R} \boldsymbol{L}_k^T, \tag{2.25}$$

$$\boldsymbol{K}_k = \boldsymbol{P}_{k|k-1} \boldsymbol{H}_k^T \boldsymbol{S}_k{}^{-1}, \tag{2.26}$$

respectively.

2.4 Variational Inference

Variational Bayesian (VB) inference is a technique used to approximate complex posterior distributions in Bayesian statistics. The key idea behind VB is to transform the problem of computing a difficult-to-compute posterior distribution into an optimization problem [46, Chapter 10]. This is achieved by approximating the true posterior distribution $p(\mathbf{X}|\mathcal{Y})$ with a simpler variational distribution $q(\mathbf{X})$ and minimizing the Kullback-Leibler (KL) divergence between them.

The KL divergence between the true posterior $p(\mathbf{X}|\mathcal{Y})$ and the variational distribution $q(\mathbf{X})$ is given by

$$\mathrm{KL}(q(\mathbf{X}) \parallel p(\mathbf{X}|\mathcal{Y})) = \int q(\mathbf{X}) \log \frac{q(\mathbf{X})}{p(\mathbf{X}|\mathcal{Y})} \, d\mathbf{X}. \tag{2.27}$$

Minimizing this KL divergence is equivalent to maximizing the evidence lower bound (ELBO), which is defined as

$$\mathcal{L}(q) \triangleq \mathbb{E}_q \left[\log p(\mathcal{Y}, \mathbf{X})\right] - \mathbb{E}_q \left[\log q(\mathbf{X})\right]. \tag{2.28}$$

where $\mathcal{Y}$ are the observed data, $\mathbf{X}$ are the latent variables, $p(\mathcal{Y}, \mathbf{X})$ is the joint distribution of the data, and latent variables, and $q(\mathbf{X})$ is the variational distribution approximating the posterior.

This equivalence can be seen from the following manipulations

$$\mathrm{KL}(q(\mathbf{X}) \parallel p(\mathbf{X}|\mathcal{Y})) = \int q(\mathbf{X}) \log \left[\frac{q(\mathbf{X})}{p(\mathbf{X}|\mathcal{Y})}\right] \, \mathrm{d}\mathbf{X}, \tag{2.29a}$$

$$= \int q(\mathbf{X}) \left(\log \left[q(\mathbf{X})\right] - \log \left[p(\mathbf{X}|\mathcal{Y})\right]\right) \, \mathrm{d}\mathbf{X}, \tag{2.29b}$$

$$= \int q(\mathbf{X}) \left(\log \left[q(\mathbf{X}) \right] - \log \left[\frac{p(\mathbf{X}, \mathcal{Y})}{p(\mathcal{Y})} \right] \right) \, \mathrm{d}\mathbf{X}, \tag{2.29c}$$

$$= \int q(\mathbf{X}) \log \left[q(\mathbf{X}) \right] \, \mathrm{d}\mathbf{X} - \int q(\mathbf{X}) \log \left[p(\mathbf{X}, \mathcal{Y}) \right] \, \mathrm{d}\mathbf{X} + \int q(\mathbf{X}) \log \left[p(\mathcal{Y}) \right] \, \mathrm{d}\mathbf{X}, \tag{2.29d}$$

$$= \mathbb{E}_q \left[\log q(\mathbf{X}) \right] - \mathbb{E}_q \left[\log p(\mathcal{Y}, \mathbf{X}) \right] + \mathbb{E}_q \left[\log p(\mathcal{Y}) \right], \tag{2.29e}$$

$$= -\mathcal{L}(q) + \log p(\mathcal{Y}). \tag{2.29f}$$

The goal of VB inference is to maximize the ELBO with respect to the variational distribution $q(\mathbf{X})$, making $q(\mathbf{X})$ a good approximation to the true posterior $p(\mathbf{X}|\mathcal{Y})$.

2.4.1 Mean-Field Approximation

Factorized distributions, also known as mean-field approximations, are crucial in variational inference for simplifying the calculations of posterior distributions in Bayesian statistics.

A factorized distribution assumes that the variational distribution $q(\mathbf{X})$ decomposes into a product of independent distributions for disjoint subsets of variables:

$$q(\mathbf{X}) = \prod_{i=1}^{N} q_i(\boldsymbol{x}_i), \tag{2.30}$$

where $\mathbf{X} = (\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N)$ represents the set of random variables, and each $q_i(\boldsymbol{x}_i)$ is an independent distribution specific to the random variable $\boldsymbol{x}_i$.

By assuming independence among the components of $\mathbf{X}$, factorized distributions transform the complex task of estimating a joint distribution into several simpler marginal distributions. This approach reduces the computational complexity, avoiding the curse of dimensionality typical in high-dimensional spaces.

Furthermore, the factorization allows for analytical solutions in the update steps of the variational inference algorithms. For each factor $q_i(\boldsymbol{x}_i)$, the update rules can often be derived in closed forms, turning the problem of finding the density $q(\mathbf{X})$ into an iterative optimization process.

The independence assumption implies that each distribution $q_i(\boldsymbol{x}_i)$ can be optimized

independently of others. This property enhances the scalability of variational inference algorithms, making them applicable to high-dimensional (large-scale) Bayesian models.

2.4.2 Functional Derivative of ELBO and Gradient Ascend Method

After establishing the mean-field approximation to simplify the problem, we now focus on deriving the update rules for the variational distributions by considering the functional derivative of the ELBO. This approach allows us to iteratively refine our approximation by using the gradient ascent method.

To derive the update rule for a specific factor $q_j(\boldsymbol{x}_j)$, we need to maximize the ELBO for $q_j(\boldsymbol{x}_j)$ while holding the other factors fixed. First, we express the ELBO in a form that explicitly shows the dependence on $q_j(\boldsymbol{x}_j)$.

$$\mathcal{L}(q) = \mathbb{E}_{q_j}\left[\mathbb{E}_{q_{\setminus j}}\left[\log p(\mathcal{Y}, \mathbf{X})\right]\right] - \mathbb{E}_{q_j}\left[\log q_j(\boldsymbol{x}_j)\right] + \text{const}_j, \tag{2.31}$$

$$= \int q_j(\boldsymbol{x}_j)\left(\mathbb{E}_{q_{\setminus j}}\left[\log p(\mathcal{Y}, \mathbf{X})\right]\right)\,\mathrm{d}\boldsymbol{x}_j - \int q_j(\boldsymbol{x}_j)\log q_j(\boldsymbol{x}_j)\,\mathrm{d}\boldsymbol{x}_j + \text{const}_j. \tag{2.32}$$

Consider a small perturbation $\delta q_j(\boldsymbol{x}_j)$ in $q_j(\boldsymbol{x}_j)$

$$q_j(\boldsymbol{x}_j) \to q_j(\boldsymbol{x}_j) + \epsilon\delta q_j(\boldsymbol{x}_j), \tag{2.33}$$

which can be substituted into the ELBO

$$\mathcal{L}(q + \epsilon\delta q_j) = \int \left[q_j(\boldsymbol{x}_j) + \epsilon\delta q_j(\boldsymbol{x}_j)\right]\mathbb{E}_{q_{\setminus j}}\left[\log p(\mathcal{Y}, \mathbf{X})\right]\,\mathrm{d}\boldsymbol{x}_j, - \int \left[q_j(\boldsymbol{x}_j) + \epsilon\delta q_j(\boldsymbol{x}_j)\right]\log\left[q_j(\boldsymbol{x}_j) + \epsilon\delta q_j(\boldsymbol{x}_j)\right]\,\mathrm{d}\boldsymbol{x}_j + \text{const}_j. \tag{2.34}$$

To obtain the functional derivative of the ELBO, with the definition presented in Appendix A we need derivatives of the perturbed ELBO with respect to ϵ, eavaluated at $\epsilon = 0$.

The derivative of the first integral is

$$\frac{\partial}{\partial\epsilon}\int \left(q_j(\boldsymbol{x}_j) + \epsilon\delta q_j(\boldsymbol{x}_j)\right)\mathbb{E}_{q_{\setminus j}}\left[\log p(\mathcal{Y}, \mathbf{X})\right]\,\mathrm{d}\boldsymbol{x}_j\Big|_{\epsilon=0}, \tag{2.35a}$$

$$= \frac{\partial}{\partial \epsilon} \int q_j(\boldsymbol{x}_j) \mathbb{E}_{q_{\setminus j}} \left[\log p(\mathcal{Y}, \mathbf{X})\right] \, \mathrm{d}\boldsymbol{x}_j \Big|_{\epsilon=0}$$
$$+ \frac{\partial}{\partial \epsilon} \epsilon \int \delta q_j(\boldsymbol{x}_j) \mathbb{E}_{q_{\setminus j}} \left[\log p(\mathcal{Y}, \mathbf{X})\right] \, \mathrm{d}\boldsymbol{x}_j \Big|_{\epsilon=0}, \tag{2.35b}$$
$$= \int \delta q_j(\boldsymbol{x}_j) \mathbb{E}_{q_{\setminus j}} \left[\log p(\mathcal{Y}, \mathbf{X})\right] \, \mathrm{d}\boldsymbol{x}_j. \tag{2.35c}$$

Similarly, the second integral can be differentiated with respect to ϵ as

$$\frac{\partial}{\partial \epsilon} \int \left[q_j(\boldsymbol{x}_j) + \epsilon \delta q_j(\boldsymbol{x}_j)\right] \log \left[q_j(\boldsymbol{x}_j) + \epsilon \delta q_j(\boldsymbol{x}_j)\right] \, \mathrm{d}\boldsymbol{x}_j \Big|_{\epsilon=0},$$
$$= \frac{\partial}{\partial \epsilon} \int \left(q_j(\boldsymbol{x}_j) + \epsilon \delta q_j(\boldsymbol{x}_j)\right) \Bigg(\log \left[q_j(\boldsymbol{x}_j)\right]$$
$$+ \epsilon \frac{\delta q_j(\boldsymbol{x}_j)}{q_j(\boldsymbol{x}_j) + \epsilon \delta q_j(\boldsymbol{x}_j)} + \mathcal{O}(\epsilon^2) \Bigg) \, \mathrm{d}\boldsymbol{x}_j \Big|_{\epsilon=0}, \tag{2.36a}$$
$$= \frac{\partial}{\partial \epsilon} \int q_j(\boldsymbol{x}_j) \log \left[q_j(\boldsymbol{x}_j)\right] + \epsilon \delta q_j(\boldsymbol{x}_j) \log \left[q_j(\boldsymbol{x}_j)\right] \, \mathrm{d}\boldsymbol{x}_j \Big|_{\epsilon=0}$$
$$+ \frac{\partial}{\partial \epsilon} \int \frac{q_j(\boldsymbol{x}_j) \epsilon \delta q_j(\boldsymbol{x}_j)}{q_j(\boldsymbol{x}_j) + \epsilon \delta q_j(\boldsymbol{x}_j)} + \mathcal{O}(\epsilon^2) \, \mathrm{d}\boldsymbol{x}_j \Big|_{\epsilon=0}, \tag{2.36b}$$
$$= \frac{\partial}{\partial \epsilon} \int q_j(\boldsymbol{x}_j) \log \left[q_j(\boldsymbol{x}_j)\right] + \epsilon \delta q_j(\boldsymbol{x}_j) \log \left[q_j(\boldsymbol{x}_j)\right]$$
$$+ \frac{\epsilon \delta q_j(\boldsymbol{x}_j)(q_j(\boldsymbol{x}_j) + \epsilon \delta q_j(\boldsymbol{x}_j)) + \mathcal{O}(\epsilon^2)}{q_j(\boldsymbol{x}_j) + \epsilon \delta q_j(\boldsymbol{x}_j)} + \mathcal{O}(\epsilon^2) \, \mathrm{d}\boldsymbol{x}_j \Big|_{\epsilon=0}, \tag{2.36c}$$
$$= \frac{\partial}{\partial \epsilon} \int q_j(\boldsymbol{x}_j) \log \left[q_j(\boldsymbol{x}_j)\right] \, \mathrm{d}\boldsymbol{x}_j \Big|_{\epsilon=0}$$
$$+ \frac{\partial}{\partial \epsilon} \epsilon \int \delta q_j(\boldsymbol{x}_j) \left(\log \left[q_j(\boldsymbol{x}_j)\right] + 1\right) \, \mathrm{d}\boldsymbol{x}_j \Big|_{\epsilon=0} + \frac{\partial}{\partial \epsilon} \epsilon^2 \int \mathcal{O}(1) \, \mathrm{d}\boldsymbol{x}_j \Big|_{\epsilon=0}, \tag{2.36d}$$
$$= \int \delta q_j(\boldsymbol{x}_j) \left(\log \left[q_j(\boldsymbol{x}_j)\right] + 1\right) \, \mathrm{d}\boldsymbol{x}_j. \tag{2.36e}$$

Combining the terms, we get

$$\int \frac{\delta \mathcal{L}}{\delta q_j(\boldsymbol{x}_j)} \delta q_j(\boldsymbol{x}_j) \, dx_j$$
$$= \int \delta q_j(\boldsymbol{x}_j) \mathbb{E}_{q_{\setminus j}} \left[\log p(\mathcal{Y}, \mathbf{X})\right] \, \mathrm{d}\boldsymbol{x}_j - \int \delta q_j(\boldsymbol{x}_j) \left(\log \left[q_j(\boldsymbol{x}_j)\right] + 1\right) \, \mathrm{d}\boldsymbol{x}_j, \tag{2.37}$$
$$= \int \delta q_j(\boldsymbol{x}_j) \left(\mathbb{E}_{q_{\setminus j}} \left[\log p(\mathcal{Y}, \mathbf{X})\right] - \log \left[q_j(\boldsymbol{x}_j)\right] - 1 \right) \, \mathrm{d}\boldsymbol{x}_j. \tag{2.38}$$

The final result for the functional derivative is

$$\frac{\delta \mathcal{L}}{\delta q_j(\boldsymbol{x}_j)} = \mathbb{E}_{q_{\setminus j}} \left[\log p(\mathcal{Y}, \mathbf{X})\right] - \log q_j(\boldsymbol{x}_j) - 1. \tag{2.39}$$

By [47, Chapter 1, Section 3, Theorem 2], the distribution maximizing the ELBO has a vanishing functional derivative. We can set the functional derivative to zero to find the maximizing distribution.

$$\mathbb{E}_{q_{\setminus j}}\left[\log p(\mathcal{Y}, \mathbf{X})\right] - \log q_j(\boldsymbol{x}_j) - 1 = 0, \tag{2.40}$$

which we can solve for $q_j(\boldsymbol{x}_j)$ to obtain

$$\log q_j(\boldsymbol{x}_j) = \mathbb{E}_{q_{\setminus j}}\left[\log p(\mathcal{Y}, \mathbf{X})\right] - 1. \tag{2.41}$$

Exponentiating both sides and ensuring normalization gives the solution for $q_j(\boldsymbol{x}_j)$ as

$$q_j(\boldsymbol{x}_j) \propto \exp\left(\mathbb{E}_{q_{\setminus j}}\left[\log p(\mathcal{Y}, \mathbf{X})\right]\right), \tag{2.42}$$

$$q_j(\boldsymbol{x}_j) = \frac{\exp\left(\mathbb{E}_{q_{\setminus j}}\left[\log p(\mathcal{Y}, \mathbf{X})\right]\right)}{\int \exp\left(\mathbb{E}_{q_{\setminus j}}\left[\log p(\mathcal{Y}, \mathbf{X})\right]\right) \, \mathrm{d}\boldsymbol{x}_j}. \tag{2.43}$$

This rule is iteratively applied for each factor until convergence, updating $q_j(\boldsymbol{x}_j)$ while keeping the other factors fixed. This ensures that the ELBO is maximized over iterations, leading to a good approximation of the true posterior distribution.

CHAPTER 3

MODELS AND INFERENCE

3.1 Problem Formulation

Consider a target that generates multiple measurements within a single scan with measurements spread throughout the target's body. In RM-based elliptical ETT frameworks, the measurement likelihood can be expressed as,

$$p(\boldsymbol{y}_k^j \mid \boldsymbol{x}_k, \boldsymbol{X}_k) = \mathcal{N}(\boldsymbol{y}_k^j;\ \boldsymbol{H}\boldsymbol{x}_k,\ s\boldsymbol{X}_k + \boldsymbol{R}_k), \tag{3.1}$$

where,

- $\boldsymbol{y}_k^j \in \mathbb{R}^{n_y}$ denotes the j^{th} measurement obtained at time k
- $\boldsymbol{x}_k \in \mathbb{R}^{n_x}$ is the kinematic state vector
- $\boldsymbol{X}_k \in \mathbb{S}_{++}^d$ is the d dimensional geometric extent state
- $\boldsymbol{H} \in \mathbb{R}^{n_y \times n_x}$ is a matrix that selects the Cartesian position of the kinematic state
- $s \in \mathbb{R}_+$ is a scaling parameter determined by the measurement distribution on the target extent [10].

The representation of the extent state may vary across various approaches. We define the extent state as a pair $(\boldsymbol{\Gamma}_k, \boldsymbol{T}_k)$ which consists of a diagonal matrix $\boldsymbol{\Gamma}_k$, whose entries correspond to the axis lengths of the represented ellipsoid and an orthonormal matrix $\boldsymbol{T}_k$ representing the orientation of the underlying extent with fixed axis directions [19]. The extent matrix $\boldsymbol{X}_k$ can be defined as

$$\boldsymbol{X}_k \triangleq \boldsymbol{T}_k \boldsymbol{\Gamma}_k \boldsymbol{T}_k^T, \tag{3.2}$$

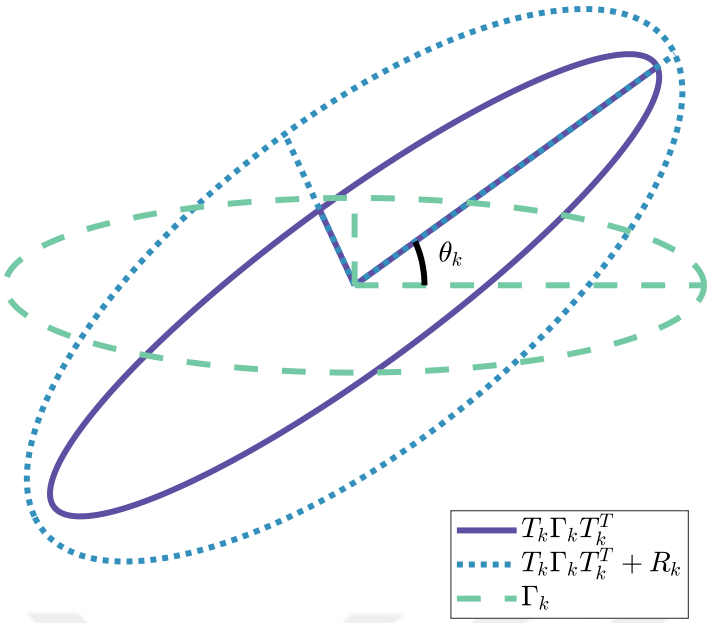


Figure 3.1: Spatial region in which the measurements are assumed to be generated in terms of rotation matrix $\boldsymbol{T}_k$, diagonal positive definite matrix $\boldsymbol{\Gamma}_k$ and measurement noise covariance $\boldsymbol{R}_k$

where $\boldsymbol{\Gamma}_k \triangleq \operatorname{diag}(\gamma_k^1, \gamma_k^2, \ldots, \gamma_k^d)$ where each element is a positive real number distributed with IG distribution, i.e., $\gamma_k^i \sim \mathcal{IG}\left(\alpha_k^i, \beta_k^i\right)$. This extent matrix definition is well-defined and conforms to the symmetric positive definite random matrix definition, as established by the principal axis theorem [48]. The spatial regions defined by this representation are illustrated in Figure 3.1.

We aim to estimate the kinematic and extent states using the measurements obtained until time k. We employ Bayesian filtering and state estimation frameworks to utilize the recursive solution of the Chapman-Kolmogorov and Bayes' equations for prediction and correction updates [7] respectively

$$p(\xi_k|\mathcal{Y}_{1:k-1}) = \int p(\xi_k|\xi_{k-1}) p(\xi_{k-1}|\mathcal{Y}_{1:k-1}) \, \mathrm{d}\xi_{k-1}, \tag{3.3a}$$

$$p(\xi_k|\mathcal{Y}_{1:k}) = \frac{p(\mathcal{Y}_k|\xi_k) \, p(\xi_k|\mathcal{Y}_{1:k-1})}{\int p(\mathcal{Y}_k|\xi_k) \, p(\xi_k|\mathcal{Y}_{1:k-1}) \, \mathrm{d}\xi_k}, \tag{3.3b}$$

where we define $\xi_k \triangleq (\boldsymbol{x}_k, \boldsymbol{X}_k)$ as the state variable.

For a set of measurements $\mathcal{Y}_k = \{\boldsymbol{y}_k^j\}_{j=1}^{m_k}$ obtained at time k, we define the measure-

ment likelihood as

$$p(\mathcal{Y}_k \mid \xi_k) = \prod_{j=1}^{m_k} p(\boldsymbol{y}_k^j \mid \xi_k), \tag{3.4}$$

and for each conditionally independent measurement $\boldsymbol{y}_k^j$ the likelihood is obtained by substituting (3.2) in (3.1)

$$p(\boldsymbol{y}_k^j \mid \xi_k) = \mathcal{N}(\boldsymbol{y}_k^j; \boldsymbol{H}\boldsymbol{x}_k,\, s\boldsymbol{T}_k\boldsymbol{\Gamma}_k\boldsymbol{T}_k^T + \boldsymbol{R}_k). \tag{3.5}$$

In the following, we introduce two distinct target models characterized by variations in their kinematic state definitions and the relationship between the target trajectory and orientation. We provide formulations for their measurement updates, followed by the presentation of the time update equations for our models.

3.2 Trajectory-Aligned Model

Objects that mostly move in the direction of their heading, such as cars, bicycles, or trains, exhibit tightly dependent kinematic and extent state motion characteristics. The trajectory-aligned model aims to exploit this dependence to improve tracking performance. Within this model, the dependence of the extent orientation and the kinematic state is maintained using the correlation structure of the joint state. In this section, we will derive the update equations for this model.

3.2.1 Measurement Update for the Trajectory-Aligned Model

Consider the prior joint density

$$p(\boldsymbol{x}_0, \boldsymbol{\Gamma}_0) = \mathcal{N}\left(\boldsymbol{x}_0; \hat{\boldsymbol{x}}_0, \boldsymbol{P}_0\right) \times \prod_{i=1}^{2} \mathcal{IG}(\gamma_0^i; \alpha_0^i, \beta_0^i), \tag{3.6}$$

where the Gaussian factor and the IG factors correspond to the kinematic state and the extent variables, respectively. Starting with the prior (3.6), our aim is to propagate the joint density in time, perform variational inference, and obtain approximate posteriors.

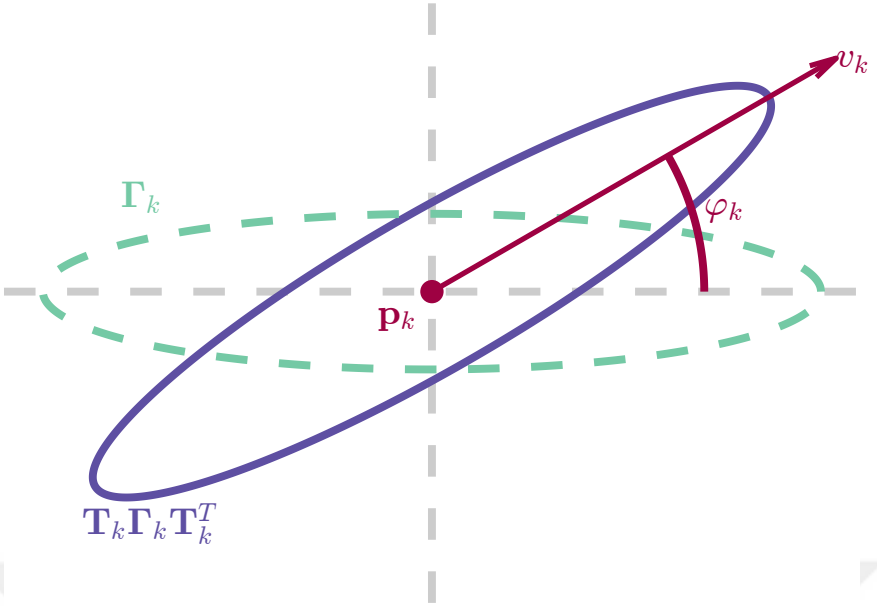


Figure 3.2: An illustration of the state variables used in trajectory-aligned model, where $\boldsymbol{p}_k = \begin{bmatrix} p_k^x & p_k^y \end{bmatrix}^T$ the cartesian position of the ellipse center, v_k is the speed, φ_k is the heading angle, $\boldsymbol{\Gamma}_k$ is the diagonal extent matrix and $\boldsymbol{T}_k = \boldsymbol{T}(\varphi_k)$ is the rotation matrix.

Assume that at time k, we have a set of measurements $\mathcal{Y}_k$ and the following predicted density

$$p(\boldsymbol{x}_k, \boldsymbol{\Gamma}_k | \mathcal{Y}_{1:k-1}) = \mathcal{N}\left(\boldsymbol{x}_k; \hat{\boldsymbol{x}}_{k|k-1}, \boldsymbol{P}_{k|k-1}\right) \times \prod_{i=1}^{2} \mathcal{IG}(\gamma_k^i; \alpha_{k|k-1}^i, \beta_{k|k-1}^i), \tag{3.7}$$

where $\hat{\boldsymbol{x}}_{k|k-1}$ and $\boldsymbol{P}_{k|k-1}$ are the mean and covariance of the predicted kinematic state, and for each diagonal element γ_k^i of the extent state $\boldsymbol{\Gamma}_k$, $\alpha_{k|k-1}^i$ is the shape and $\beta_{k|k-1}^i$ is the scale parameter of the IG distribution. The posterior density can be obtained in the measurement update by applying the Bayes' rule

$$p(\boldsymbol{x}_k, \boldsymbol{\Gamma}_k | \mathcal{Y}_{1:k}) = \frac{p(\mathcal{Y}_k | \boldsymbol{x}_k, \boldsymbol{\Gamma}_k) p(\boldsymbol{x}_k, \boldsymbol{\Gamma}_k | \mathcal{Y}_{1:k-1})}{p(\mathcal{Y}_k | \mathcal{Y}_{1:k-1})}. \tag{3.8}$$

The kinematic state $\boldsymbol{x}_k$ contains Cartesian position of the centroid and the heading angle of the motion. The measurement likelihood in (3.4) becomes

$$p(\mathcal{Y}_k | \boldsymbol{x}_k, \boldsymbol{\Gamma}_k) = \prod_{j=1}^{m_k} \mathcal{N}(\boldsymbol{y}_k^j; \boldsymbol{H}\boldsymbol{x}_k, s\boldsymbol{T}_k\boldsymbol{\Gamma}_k\boldsymbol{T}_k^T + \boldsymbol{R}_k), \tag{3.9}$$

where $\boldsymbol{H}$ and $\boldsymbol{H}_{\varphi}$ are the matrices selecting the Cartesian position and the heading angle from the kinematic state vector, respectively, and $\boldsymbol{T}_k = \boldsymbol{T}(\boldsymbol{H}_{\varphi}\boldsymbol{x}_k)$ is the rotation matrix.

With the measurement likelihood (3.9), an exact solution of (3.8) is unattainable [10]. Therefore, we aim to find an approximate solution using the variational Bayes (VB) method [46, Chapter 10]. Within the VB framework, we approximate the true posterior as factorized distributions

$$p(\boldsymbol{x}_k, \boldsymbol{\Gamma}_k, \mathcal{Z}_k|\mathcal{Y}_{1:k}) \approx q_{\boldsymbol{x}}(\boldsymbol{x}_k) q_{\boldsymbol{\Gamma}}(\boldsymbol{\Gamma}_k) q_{\mathcal{Z}}(\mathcal{Z}_k), \tag{3.10}$$

where $\mathcal{Z}_k$ denotes the set of *noise-free* measurements at time k. The noise-free measurements are introduced to address the absence of conjugacy caused by the measurement noise covariance $\boldsymbol{R}_k$ [11, 19]. The joint density of the noise-free measurement $\boldsymbol{z}_k^j$ and the measurement $\boldsymbol{y}_k^j$ is defined as

$$p(\boldsymbol{y}_k^j, \boldsymbol{z}_k^j|\boldsymbol{x}_k, \boldsymbol{\Gamma}_k) = \mathcal{N}(\boldsymbol{y}_k^j; \boldsymbol{z}_k^j, \boldsymbol{R}_k)\, \mathcal{N}(\boldsymbol{z}_k^j; \boldsymbol{H}\boldsymbol{x}_k, s\boldsymbol{T}_k\boldsymbol{\Gamma}_k\boldsymbol{T}_k^T). \tag{3.11}$$

Marginalizing this joint density yields the measurement likelihood (3.5),

$$p(\boldsymbol{y}_k^j|\boldsymbol{x}_k, \boldsymbol{\Gamma}_k) = \int p(\boldsymbol{y}_k^j|\boldsymbol{z}_k^j) p(\boldsymbol{z}_k^j|\boldsymbol{x}_k, \boldsymbol{\Gamma}_k)\, \mathrm{d}\boldsymbol{z}_k^j. \tag{3.12}$$

The approximate posterior (3.10) can be found by minimizing the cost function

$$\hat{q}_{\boldsymbol{x}}, \hat{q}_{\boldsymbol{\Gamma}}, \hat{q}_{\mathcal{Z}} \triangleq \underset{q_{\boldsymbol{x}}, q_{\boldsymbol{\Gamma}}, q_{\mathcal{Z}}}{\operatorname{argmin}} KL(q_{\boldsymbol{x}}(\boldsymbol{x}_k) q_{\boldsymbol{\Gamma}}(\boldsymbol{\Gamma}_k) q_{\mathcal{Z}}(\mathcal{Z}_k) || p(\boldsymbol{x}_k, \boldsymbol{\Gamma}_k, \mathcal{Z}_k|\mathcal{Y}_{1:k})). \tag{3.13}$$

Solution for each factor can be obtained after fixed point iterations [46, Chapter 10], in the form

$$\log \hat{q}_{\sigma}(\sigma_k) = \mathbb{E}_{\setminus\sigma}\big[\log p(\boldsymbol{x}_k, \boldsymbol{\Gamma}_k, \mathcal{Z}_k, \mathcal{Y}_k|\mathcal{Y}_{1:k-1})\big] + c_{\sigma}, \tag{3.14}$$

where notation $\setminus\sigma$ denotes exclusion of σ which is a placeholder for variables $\boldsymbol{x}_k$, $\boldsymbol{\Gamma}_k$, $\mathcal{Z}_k$, and c_{σ} denotes a constant expression with respect to σ. Approximate densities $\hat{q}_{\sigma}$ in (3.13) are calculated recursively by iterating (3.14) for each variable $\sigma \in \{\boldsymbol{x}_k, \boldsymbol{\Gamma}_k, \mathcal{Z}_k\}$ until certain stopping criteria are met. Alternatively, as is commonly practiced in the literature, a fixed number of iterations can be performed [46, Chapter 10], [11,19,20]. To provide clarity for calculations, the joint density in (3.14) is expressed explicitly in terms of (3.7) and (3.11) as follows

$$p(\boldsymbol{x}_k, \boldsymbol{\Gamma}_k, \mathcal{Z}_k, \mathcal{Y}_k|\mathcal{Y}_{1:k-1})$$

$$= p(\boldsymbol{x}_k, \boldsymbol{\Gamma}_k | \mathcal{Y}_{1:k-1})\, p(\mathcal{Y}_k | \mathcal{Z}_k)\, p(\mathcal{Z}_k | \boldsymbol{x}_k, \boldsymbol{\Gamma}_k), \tag{3.15a}$$

$$= \mathcal{N}(\boldsymbol{x}_k; \hat{\boldsymbol{x}}_{k|k-1}, \boldsymbol{P}_{k|k-1}) \prod_{i=1}^{2} \mathcal{IG}(\gamma_k^i; \alpha_{k|k-1}^i, \beta_{k|k-1}^i)$$

$$\times \prod_{j=1}^{m_k} \mathcal{N}(\boldsymbol{y}_k^j; \boldsymbol{z}_k^j, \boldsymbol{R}_k)\, \mathcal{N}(\boldsymbol{z}_k^j; \boldsymbol{H}\boldsymbol{x}_k, s\boldsymbol{T}_k \boldsymbol{\Gamma}_k \boldsymbol{T}_k^T). \tag{3.15b}$$

The iterations are performed by updating each factor in (3.10) one by one while keeping the others fixed. We start with the initial distributions with sufficient statistics;

$$\boldsymbol{P}_{k|k}^{(0)} = \boldsymbol{P}_{k|k-1}, \tag{3.16a} \qquad \hat{\boldsymbol{x}}_{k|k}^{(0)} = \hat{\boldsymbol{x}}_{k|k-1}, \tag{3.16b}$$

$$\alpha_{k|k}^{i,(0)} = \alpha_{k|k-1}^i, \tag{3.16c} \qquad \beta_{k|k}^{i,(0)} = \beta_{k|k-1}^i, \tag{3.16d}$$

$$\Sigma_k^{z,(0)} = \mathbb{E}_{q_{\boldsymbol{\Gamma}}^{(0)}}\left[s\boldsymbol{\Gamma}_k\right], \tag{3.16g} \qquad \hat{\boldsymbol{z}}_k^{j,(0)} = \boldsymbol{y}_k^j. \tag{3.16h}$$

where $\mathbb{E}_{q_{\boldsymbol{\Gamma}}^{(0)}}[(s\boldsymbol{\Gamma}_k)] = \operatorname{diag}\left(\frac{s\beta_{k|k-1}^1}{\alpha_{k|k-1}^1 - 1}, \frac{s\beta_{k|k-1}^2}{\alpha_{k|k-1}^2 - 1}\right)$ and proceed with fixed-point iterations. The derivations of update equations for each factorized distribution at an intermediate step ($(\ell+1)^{\text{st}}$ iteration) are detailed in the following.

3.2.1.1 Update of $\hat{q}_{\boldsymbol{x}}^{(\ell+1)}(\cdot)$

Using the joint density (3.15b) and applying the update rule in (3.14) the update rule for $\hat{q}_{\boldsymbol{x}}^{(\ell+1)}(\cdot)$ is obtained.

$$\log q_{\boldsymbol{x}}^{(\ell+1)}(\boldsymbol{x}_k) = \mathbb{E}_{\setminus \boldsymbol{x}_k}\left[\log p(\boldsymbol{x}_k, \boldsymbol{\Gamma}_k, \mathcal{Z}_k, \mathcal{Y}_k | \mathcal{Y}_{1:k-1})\right] + c_{\boldsymbol{x}_k}, \tag{3.17a}$$

$$= -\frac{1}{2} \sum_{j=1}^{m_k} \mathbb{E}_{\setminus \boldsymbol{x}_k}\left[(\boldsymbol{z}_k^j - \boldsymbol{H}\boldsymbol{x}_k)^T \boldsymbol{T}_k (s\boldsymbol{\Gamma}_k)^{-1} \boldsymbol{T}_k^T(\cdot)\right]$$

$$+ \log \mathcal{N}(\boldsymbol{x}_k; \hat{\boldsymbol{x}}_{k|k-1}, \boldsymbol{P}_{k|k-1}) + c_{\boldsymbol{x}_k}. \tag{3.17b}$$

The quadratic form in (3.17b) involves nonlinearities with respect to $\boldsymbol{x}_k$ due to the rotation matrix $\boldsymbol{T}_k$. To obtain an update resulting in a multivariate Gaussian density we use the following approximate quadratic form

$$\mathbb{E}_{\setminus \boldsymbol{x}_k}\left[(a_j - b_j \boldsymbol{x}_k)^T (s\boldsymbol{\Gamma}_k)^{-1} (a_j - b_j \boldsymbol{x}_k)\right]. \tag{3.18}$$

Thus, a first-order Taylor series approximation of $h_j(\boldsymbol{x}_k) \triangleq \boldsymbol{T}_k^T(\boldsymbol{z}_k^j - \boldsymbol{H}\boldsymbol{x}_k)$ at point $\hat{\boldsymbol{x}}_{k|k}^{(\ell)}$ in the form of $a_j - b_j \boldsymbol{x}_k$ will be used instead of the nonlinear function h_j

$$h_j(\boldsymbol{x}_k) \approx h_j(\hat{\boldsymbol{x}}_{k|k}^{(\ell)}) + \nabla h_j^T(\hat{\boldsymbol{x}}_{k|k}^{(\ell)})(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_{k|k}^{(\ell)}), \tag{3.19a}$$

$$a_j \triangleq h_j(\hat{\boldsymbol{x}}_{k|k}^{(\ell)}) - \nabla h_j^T(\hat{\boldsymbol{x}}_{k|k}^{(\ell)})\hat{\boldsymbol{x}}_{k|k}^{(\ell)}, \tag{3.19b}$$

$$b_j \triangleq -\nabla h_j^T(\hat{\boldsymbol{x}}_{k|k}^{(\ell)}). \tag{3.19c}$$

The first-order Taylor expansion of the function $h_j(\boldsymbol{x}_k) = \boldsymbol{T}_k^T(z_k^j - \boldsymbol{H}\boldsymbol{x}_k)$ around the point $\hat{\boldsymbol{x}}_{k|k}^{(\ell)}$ in the form of $a_j - b_j\boldsymbol{x}_k$ The gradient of $\nabla h_j(\hat{\boldsymbol{x}}_{k|k}^{(\ell)})$ is required.

$$\nabla h_j(\hat{\boldsymbol{x}}_{k|k}^{(\ell)}) = \boldsymbol{H}_\varphi^T(z_k^j - \boldsymbol{H}\hat{\boldsymbol{x}}_{k|k}^{(\ell)})^T\dot{\boldsymbol{T}}_k - \boldsymbol{H}^T\boldsymbol{T}_k. \tag{3.20}$$

Using the definitions of the constants a_j and b_j we get

$$a_j = \boldsymbol{T}_{k,\ell}^T(z_k^j - \boldsymbol{H}\hat{\boldsymbol{x}}_{k|k}^{(\ell)}) - \left(\dot{\boldsymbol{T}}_{k,\ell}^T(z_k^j - \boldsymbol{H}\hat{\boldsymbol{x}}_{k|k}^{(\ell)})\boldsymbol{H}_\varphi - \boldsymbol{T}_{k,\ell}^T\boldsymbol{H}\right)\hat{\boldsymbol{x}}_{k|k}^{(\ell)}, \tag{3.21a}$$

$$b_j = -\dot{\boldsymbol{T}}_{k,\ell}^T(z_k^j - \boldsymbol{H}\hat{\boldsymbol{x}}_{k|k}^{(\ell)})\boldsymbol{H}_\varphi + \boldsymbol{T}_{k,\ell}^T\boldsymbol{H}, \tag{3.21b}$$

where we define $\dot{\boldsymbol{T}}_{k,\ell} \triangleq \dot{\boldsymbol{T}}(\boldsymbol{H}_\varphi\hat{\boldsymbol{x}}_{k|k}^{(\ell)})$ as the derivative of the rotation matrix. Variational update of $\boldsymbol{x}_k$

$$q_{\boldsymbol{x}}^{(\ell+1)}(\boldsymbol{x}_k) = \mathcal{N}(\boldsymbol{x}_k; \hat{\boldsymbol{x}}_{k|k}^{(\ell+1)}, \boldsymbol{P}_{k|k}^{(\ell+1)}), \tag{3.22}$$

is obtained by the product rule of two multivariate Normal densities,

$$\boldsymbol{P}_{k|k}^{(\ell+1)} = (\boldsymbol{P}_{k|k-1}^{-1} + \Psi)^{-1}, \tag{3.23a}$$

$$\hat{\boldsymbol{x}}_{k|k}^{(\ell+1)} = \boldsymbol{P}_{k|k}^{(\ell+1)}(\boldsymbol{P}_{k|k-1}^{-1}\hat{\boldsymbol{x}}_{k|k-1} + \psi), \tag{3.23b}$$

$$\Psi \triangleq \sum_{j=1}^{m_k} \mathbb{E}_{q_{\mathcal{Z}}^{(\ell)}}\left[b_j^T\mathbb{E}_{q_{\boldsymbol{\Gamma}}^{(\ell)}}\left[(s\boldsymbol{\Gamma}_k)^{-1}\right]b_j\right], \tag{3.23c}$$

$$\psi \triangleq \sum_{j=1}^{m_k} \mathbb{E}_{q_{\mathcal{Z}}^{(\ell)}}\left[b_j^T\mathbb{E}_{q_{\boldsymbol{\Gamma}}^{(\ell)}}\left[(s\boldsymbol{\Gamma}_k)^{-1}\right]a_j\right]. \tag{3.23d}$$

Closed forms of the expectations in (3.23c) and (3.23d) are as followns;

$$\mathbb{E}_{q_{\mathcal{Z}}^{(\ell)}}\left[b_j^T\mathbb{E}_{q_{\boldsymbol{\Gamma}}^{(\ell)}}\left[(s\boldsymbol{\Gamma}_k)^{-1}\right]b_j\right] = \boldsymbol{H}^T\boldsymbol{T}_{k,\ell}\overline{(s\boldsymbol{\Gamma}_k)^{-1}}\boldsymbol{T}_{k,\ell}^T\boldsymbol{H} \tag{3.24}$$

$$+ \boldsymbol{H}_\varphi^T\,\mathrm{Tr}\left(\left(\Sigma_k^{z,\ell} + (\overline{z_k^j} - \boldsymbol{H}\hat{\boldsymbol{x}}_{k|k}^{(\ell)})(\cdot)^T\right)\dot{\boldsymbol{T}}_{k,\ell}\overline{(s\boldsymbol{\Gamma}_k)^{-1}}\dot{\boldsymbol{T}}_{k,\ell}^T\right)\boldsymbol{H}_\varphi$$

$$- \left(\boldsymbol{H}^T\boldsymbol{T}_{k,\ell}\overline{(s\boldsymbol{\Gamma}_k)^{-1}}\dot{\boldsymbol{T}}_{k,\ell}^T(\overline{z_k^j} - \boldsymbol{H}\hat{\boldsymbol{x}}_{k|k}^{(\ell)})\boldsymbol{H}_\varphi + (\cdot)^T\right). \tag{3.25}$$

$$\mathbb{E}_{q_{\mathcal{Z}}^{(\ell)}}\left[b_j^T\mathbb{E}_{q_{\boldsymbol{\Gamma}}^{(\ell)}}\left[(s\boldsymbol{\Gamma}_k)^{-1}\right]a_j\right] =$$

$$\boldsymbol{H}_\varphi^T\Big[\mathrm{Tr}\left(\left(\Sigma_k^{z,\ell} + (\overline{z_k^j} - \boldsymbol{H}\hat{\boldsymbol{x}}_{k|k}^{(\ell)})(\cdot)^T\right)\dot{\boldsymbol{T}}_{k,\ell}\overline{(s\boldsymbol{\Gamma}_k)^{-1}}\right.$$

$$\left.\times\left((\boldsymbol{H}_\varphi\hat{\boldsymbol{x}}_{k|k}^{(\ell)})\dot{\boldsymbol{T}}_{k,\ell}^T - \boldsymbol{T}_{k,\ell}^T\right)\right)$$

$$-(\overline{z_k^j} - \boldsymbol{H}\hat{\boldsymbol{x}}_{k|k}^{(\ell)})^T\dot{\boldsymbol{T}}_{k,\ell}\overline{(s\boldsymbol{\Gamma}_k)^{-1}}\boldsymbol{T}_{k,\ell}^T\boldsymbol{H}\hat{\boldsymbol{x}}_{k|k}^{(\ell)}\Big]$$

$$+ \boldsymbol{H}^T T_{k,\ell} \overline{(s\boldsymbol{\Gamma}_k)^{-1}} \Big(\boldsymbol{T}_{k,\ell}^T (\overline{z_k^j} - \boldsymbol{H}\hat{\boldsymbol{x}}_{k|k}^{(\ell)}) \\ -(\boldsymbol{H}_\varphi \hat{\boldsymbol{x}}_{k|k}^{(\ell)}) \dot{\boldsymbol{T}}_{k,\ell}^T (\overline{z_k^j} - \boldsymbol{H}\hat{\boldsymbol{x}}_{k|k}^{(\ell)}) + \boldsymbol{T}_{k,\ell}^T \boldsymbol{H} \hat{\boldsymbol{x}}_{k|k}^{(\ell)} \Big). \tag{3.26}$$

where,

$$\mathbb{E}_{q_{\mathcal{Z}}^{(\ell)}}[\mathbf{z}_k^j] = \overline{z_k^j} = \hat{\mathbf{z}}_{k|k}^{j,(\ell)}. \tag{3.27}$$

$$\mathbb{E}_{q_{\boldsymbol{\Gamma}}^{(\ell)}}[(s\boldsymbol{\Gamma}_k)^{-1}] = \overline{(s\boldsymbol{\Gamma}_k)^{-1}} = \operatorname{diag}\left(\frac{\alpha_{k|k}^{1(\ell)}}{s\beta_{k|k}^{1(\ell)}}, \frac{\alpha_{k|k}^{2(\ell)}}{s\beta_{k|k}^{2(\ell)}} \right). \tag{3.28}$$

3.2.1.2 Update of $\hat{q}_{\boldsymbol{\Gamma}}^{(\ell+1)}(\cdot)$

Using the joint density (3.15b) and applying the update rule in (3.14) the update rule for $\hat{q}_{\boldsymbol{\Gamma}}^{(\ell+1)}(\cdot)$ is obtained.

$$\log q_{\boldsymbol{\Gamma}}^{(\ell+1)}(\boldsymbol{\Gamma}_k) = \mathbb{E}_{\setminus \boldsymbol{\Gamma}_k}\big[\log p(\boldsymbol{x}_k, \boldsymbol{\Gamma}_k, \mathcal{Z}_k, \mathcal{Y}_k | \mathcal{Y}_{1:k-1})\big] + c_{\boldsymbol{\Gamma}_k} \tag{3.29a}$$

$$= \mathbb{E}_{\setminus \boldsymbol{\Gamma}_k}\left[\log p(\mathcal{Z}_k, |\boldsymbol{x}_k, \boldsymbol{\Gamma}_k)\right] + \sum_{i=1}^{2} \log \mathcal{IG}(\gamma_k^i; \alpha_{k|k-1}^i, \beta_{k|k-1}^i) + c_{\boldsymbol{\Gamma}_k} \tag{3.29b}$$

$$= -\frac{1}{2} \sum_{j=1}^{m_k} \mathbb{E}_{\setminus \boldsymbol{\Gamma}_k}\big[(\boldsymbol{z}_k^j - \boldsymbol{H}\boldsymbol{x}_k)^T \boldsymbol{T}_k (s\boldsymbol{\Gamma}_k)^{-1} \boldsymbol{T}_k^T (\boldsymbol{z}_k^j - \boldsymbol{H}\boldsymbol{x}_k)\big] + \sum_{i=1}^{2} \log \mathcal{IG}(\gamma_k^i; \alpha_{k|k-1}^i, \beta_{k|k-1}^i) + c_{\boldsymbol{\Gamma}_k} \tag{3.29c}$$

$$= -\frac{1}{2} \sum_{j=1}^{m_k} \operatorname{Tr}\big(\mathbb{E}_{\setminus \boldsymbol{\Gamma}_k}\big[\boldsymbol{T}_k^T (\boldsymbol{z}_k^j - \boldsymbol{H}\boldsymbol{x}_k)(\cdot)^T \boldsymbol{T}_k\big] (s\boldsymbol{\Gamma}_k)^{-1}\big) + \sum_{i=1}^{2} \log \mathcal{IG}(\gamma_k^i; \alpha_{k|k-1}^i, \beta_{k|k-1}^i) + c_{\boldsymbol{\Gamma}_k} \tag{3.29d}$$

The updated extent density

$$q_{\boldsymbol{\Gamma}}^{(\ell+1)}(\boldsymbol{\Gamma}_k) = \prod_{i=1}^{2} \mathcal{IG}(\gamma_k^i; \alpha_{k|k}^{i,(\ell+1)}, \beta_{k|k}^{i,(\ell+1)}), \tag{3.30}$$

is found after calculating (3.29d) with the inverse-Gamma density parameters,

$$\alpha_{k|k}^{i,(\ell+1)} = \alpha_{k|k-1}^i + \frac{m_k}{2}, \tag{3.31a}$$

$$\beta_{k|k}^{i,(\ell+1)} = \beta_{k|k-1}^{i} + \frac{1}{2s}\sum_{j=1}^{m_k} \mathbb{E}_{\backslash \mathbf{\Gamma}_k}\left[\boldsymbol{T}_k^T(\boldsymbol{z}_k^j - \boldsymbol{H}\boldsymbol{x}_k)(\cdot)^T\boldsymbol{T}_k\right]_{ii}. \tag{3.31b}$$

The analytic form of the expectation in (3.31b) can be obtained using the law of iterated expectations [49];

$$\mathbb{E}_{\backslash \mathbf{\Gamma}_k}\left[\boldsymbol{T}_k^T(\boldsymbol{z}_k^j - \boldsymbol{H}\boldsymbol{x}_k)(\cdot)^T\boldsymbol{T}_k\right] = \mathbb{E}_{q_\varphi^{(\ell)}}\left[\boldsymbol{T}_k^T\mathbb{E}_{q_{\tilde{\boldsymbol{x}}|\varphi}^{(\ell)}, q_{\mathcal{Z}}^{(\ell)}}\left[(\boldsymbol{z}_k^j - \boldsymbol{H}\boldsymbol{x}_k)(\cdot)^T\right]\boldsymbol{T}_k\right] \tag{3.32}$$

$$\begin{aligned}
&= \mathbb{E}_{q_\varphi^{(\ell)}}\left[\boldsymbol{T}_k^T(\Sigma_k^{z,\ell} + \overline{z_k^j}\,\overline{z_k^j}^T)\boldsymbol{T}_k\right] \\
&+ \mathbb{E}_{q_\varphi^{(\ell)}}\left[\boldsymbol{T}_k^T\tilde{\boldsymbol{H}}\boldsymbol{M}_1\tilde{\boldsymbol{H}}^T\boldsymbol{T}_k\right] + \mathbb{E}_{q_\varphi^{(\ell)}}\left[\varphi_k\boldsymbol{T}_k^T\tilde{\boldsymbol{H}}\boldsymbol{M}_2\tilde{\boldsymbol{H}}^T\boldsymbol{T}_k\right] \\
&+ \mathbb{E}_{q_\varphi^{(\ell)}}\left[\varphi_k^2\boldsymbol{T}_k^T\tilde{\boldsymbol{H}}\boldsymbol{M}_3\tilde{\boldsymbol{H}}^T\boldsymbol{T}_k\right] - (\mathbb{E}_{q_\varphi^{(\ell)}}\left[\boldsymbol{T}_k^T\tilde{\boldsymbol{H}}\boldsymbol{M}_4\boldsymbol{T}_k\right] + (\cdot)^T) \\
&- (\mathbb{E}_{q_\varphi^{(\ell)}}\left[\varphi\boldsymbol{T}_k^T\tilde{\boldsymbol{H}}\boldsymbol{M}_5\boldsymbol{T}_k\right] + (\cdot)^T).
\end{aligned} \tag{3.33}$$

where $\tilde{\boldsymbol{H}}$ is a matrix that selects the Cartesian position from the reduced state $\tilde{\boldsymbol{x}}$ which is the state without the heading angle, where we partition the density $q_{\boldsymbol{x}}$ as

$$\boldsymbol{x} = \begin{bmatrix}\tilde{\boldsymbol{x}} \\ \varphi\end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix}\tilde{\boldsymbol{x}} \\ \varphi\end{bmatrix}; \begin{bmatrix}\hat{\tilde{\boldsymbol{x}}} \\ \hat{\varphi}\end{bmatrix}, \begin{bmatrix}\boldsymbol{P}_{\tilde{\boldsymbol{x}}} & \boldsymbol{P}_{\tilde{\boldsymbol{x}},\varphi} \\ \boldsymbol{P}_{\varphi,\boldsymbol{x}} & P_\varphi\end{bmatrix}\right). \tag{3.34}$$

By exploiting the Gaussian conditioning rules, the matrices $\boldsymbol{M}_i$ can be obtained as;

$$\begin{aligned}
\boldsymbol{M}_1 =&((\hat{\varphi}_{(k|k)}^{(\ell)}/P_{\varphi(k|k)}^{(\ell)})^2 - 1/P_{\varphi(k|k)}^{(\ell)})((\boldsymbol{P}_{\tilde{\boldsymbol{x}},\varphi,(k|k)}^{(\ell)})(\cdot)^T) \\
&- (\hat{\varphi}_{(k|k)}^{(\ell)}/P_{\varphi(k|k)}^{(\ell)})([\hat{\tilde{\boldsymbol{x}}}_{(k|k)}^{(\ell)}(\boldsymbol{P}_{\tilde{\boldsymbol{x}},\varphi,(k|k)}^{(\ell)})^T] + [\cdot]^T) \\
&+ \boldsymbol{P}_{\tilde{\boldsymbol{x}},(k|k)}^{(\ell)} + (\hat{\tilde{\boldsymbol{x}}}_{(k|k)}^{(\ell)})(\cdot)^T,
\end{aligned} \tag{3.35a}$$

$$\begin{aligned}
\boldsymbol{M}_2 =&(P_{\varphi(k|k)}^{(\ell)})^{-1}([\hat{\tilde{\boldsymbol{x}}}_{(k|k)}^{(\ell)}(\boldsymbol{P}_{\tilde{\boldsymbol{x}},\varphi,(k|k)}^{(\ell)})^T] + [\cdot]^T) \\
&- 2\hat{\varphi}_{(k|k)}^{(\ell)}(P_{\varphi(k|k)}^{(\ell)})^{-2}((\boldsymbol{P}_{\tilde{\boldsymbol{x}},\varphi,(k|k)}^{(\ell)})(\cdot)^T),
\end{aligned} \tag{3.35b}$$

$$\boldsymbol{M}_3 = (P_{\varphi(k|k)}^{(\ell)})^{-2}\left((\boldsymbol{P}_{\tilde{\boldsymbol{x}},\varphi,(k|k)}^{(\ell)})(\cdot)^T\right), \tag{3.35c}$$

$$\boldsymbol{M}_4 = \left(\hat{\tilde{\boldsymbol{x}}}_{(k|k)}^{(\ell)} - (\hat{\varphi}_{(k|k)}^{(\ell)}/P_{\varphi(k|k)}^{(\ell)})\boldsymbol{P}_{\tilde{\boldsymbol{x}},\varphi,(k|k)}^{(\ell)}\right)\overline{z_k^j}^T, \tag{3.35d}$$

$$\boldsymbol{M}_5 = \left((\boldsymbol{P}_{\tilde{\boldsymbol{x}},\varphi,(k|k)}^{(\ell)})/P_{\varphi(k|k)}^{(\ell)}\right)\overline{z_k^j}^T. \tag{3.35e}$$

The exact expressions for the terms in (3.33) can be obtained using the Lemma 1 with [50];

$$\mathrm{vec}(\boldsymbol{T}_{k,\ell}^T\tilde{\boldsymbol{M}}\boldsymbol{T}_{k,\ell}) = \hat{\boldsymbol{M}}\boldsymbol{t}_{\hat{\varphi}_k^{(\ell)}} = \frac{1}{2}\begin{bmatrix} m_{11} & m_{22} & (m_{12}+m_{21}) \\ m_{21} & -m_{12} & (-m_{11}+m_{22}) \\ m_{12} & -m_{21} & (-m_{11}+m_{22}) \\ m_{22} & m_{11} & -(m_{12}+m_{21})\end{bmatrix}\boldsymbol{t}_{\hat{\varphi}_k^{(\ell)}}. \tag{3.36a}$$

Where $\tilde{\boldsymbol{M}} = \{m\}_{ij}$ is a 2×2 matrix which can be any of the $\boldsymbol{M}_i$ with proper multiplications with $\tilde{\boldsymbol{H}}$, as an example $\tilde{\boldsymbol{M}}_1 = \tilde{\boldsymbol{H}}\boldsymbol{M}_1\tilde{\boldsymbol{H}}^T$ as in (3.33);

$$\boldsymbol{t}_\varphi \triangleq \begin{bmatrix} 1+\cos(2\varphi) & 1-\cos(2\varphi) & \sin(2\varphi) \end{bmatrix}^T, \tag{3.37}$$

and $\mathrm{vec}(\mathbb{E}_{q_\varphi^{(\ell)}}\left[\boldsymbol{T}_k^T\tilde{\boldsymbol{M}}_i\boldsymbol{T}_k\right]) = \hat{\boldsymbol{M}}\mathbb{E}_{q_\varphi^{(\ell)}}[\boldsymbol{t}_{\hat{\varphi}_{(k|k)}^{(\ell)}}]$. Note that the expression gives the terms of the expectation as a vector, and they are reordered as a 2×2 matrix for their use. The terms containing additional φ or φ^2 terms can be similarly obtained.

Lemma 1. *Let $\varphi \sim \mathcal{N}(\varphi; \hat{\varphi}, P_\varphi)$ and define the vector*

$$\boldsymbol{t}_\varphi \triangleq \begin{bmatrix} 1+\cos(2\varphi) & 1-\cos(2\varphi) & \sin(2\varphi) \end{bmatrix}^T. \tag{3.38}$$

Then the expectations $\mathbb{E}[\boldsymbol{t}_\varphi]$, $\mathbb{E}[\varphi\boldsymbol{t}_\varphi]$ and $\mathbb{E}[\varphi^2\boldsymbol{t}_\varphi]$ are analytically calculated as

$$\mathbb{E}[\boldsymbol{t}_\varphi] = \begin{bmatrix} 1+e^{-2P_\varphi}\cos(2\hat{\varphi}) \\ 1-e^{-2P_\varphi}\cos(2\hat{\varphi}) \\ e^{-2P_\varphi}\sin(2\hat{\varphi}) \end{bmatrix}, \tag{3.39a}$$

$$\mathbb{E}[\varphi\boldsymbol{t}_\varphi] = \begin{bmatrix} \hat{\varphi}+e^{-2P_\varphi}\left(\hat{\varphi}\cos(2\hat{\varphi}) - 2P_\varphi\sin(2\hat{\varphi})\right) \\ \hat{\varphi}-e^{-2P_\varphi}\left(\hat{\varphi}\cos(2\hat{\varphi}) - 2P_\varphi\sin(2\hat{\varphi})\right) \\ e^{-2P_\varphi}\left(2P_\varphi\cos(2\hat{\varphi}) + \hat{\varphi}\sin(2\hat{\varphi})\right) \end{bmatrix}, \tag{3.39b}$$

$$\mathbb{E}\left[\varphi^2\boldsymbol{t}_\varphi\right] = \begin{bmatrix} \mathbb{E}[\varphi^2]+e^{-2P_\varphi}\left((\mathbb{E}[\varphi^2]-4P_\varphi^2)\cos(2\hat{\varphi}) - 4P_\varphi\hat{\varphi}\sin(2\hat{\varphi})\right) \\ \mathbb{E}[\varphi^2]-e^{-2P_\varphi}\left((\mathbb{E}[\varphi^2]-4P_\varphi^2)\cos(2\hat{\varphi}) - 4P_\varphi\hat{\varphi}\sin(2\hat{\varphi})\right) \\ e^{-2P_\varphi}\left((\mathbb{E}[\varphi^2]-4P_\varphi^2)\sin(2\hat{\varphi}) + 4P_\varphi\hat{\varphi}\cos(2\hat{\varphi})\right) \end{bmatrix}, \tag{3.39c}$$

where $\mathbb{E}[\varphi^2] = \hat{\varphi}^2 + P_\varphi$.

3.2.1.3 Update of $\hat{q}_{\mathcal{Z}}^{(\ell+1)}(\cdot)$

Using the joint density (3.15b) and applying the update rule in (3.14) the update rule for $\hat{q}_{\mathcal{Z}}^{(\ell+1)}(\cdot)$ is obtained.

$$\log q_{\mathcal{Z}}^{(\ell+1)}(\mathcal{Z}_k) = \mathbb{E}_{\setminus\mathcal{Z}_k}\left[\log p(\boldsymbol{x}_k, \boldsymbol{\Gamma}_k, \mathcal{Z}_k, \mathcal{Y}_k|\mathcal{Y}_{1:k-1})\right] + c_{\mathcal{Z}_k}, \tag{3.40a}$$

$$= \mathbb{E}_{\setminus\mathcal{Z}_k}\left[\log p(\mathcal{Z}_k, |\boldsymbol{x}_k, \boldsymbol{\Gamma}_k)\right] + \sum_{j=1}^{m_k}\log\mathcal{N}(\boldsymbol{y}_k^j; \boldsymbol{z}_k^j, \boldsymbol{R}_k) + c_{\mathcal{Z}_k}, \tag{3.40b}$$

$$= -\frac{1}{2}\sum_{j=1}^{m_k}\mathbb{E}_{\setminus\mathcal{Z}_k}\left[(\boldsymbol{z}_k^j - \boldsymbol{H}\boldsymbol{x}_k)^T\boldsymbol{T}_k(s\boldsymbol{\Gamma}_k)^{-1}\boldsymbol{T}_k^T(\boldsymbol{z}_k^j - \boldsymbol{H}\boldsymbol{x}_k)\right]$$

$$+\sum_{j=1}^{m_k}\log\mathcal{N}(\boldsymbol{y}_k^j;\boldsymbol{z}_k^j,\boldsymbol{R}_k)+c_{\mathcal{Z}_k}, \tag{3.40c}$$

$$\begin{aligned}=&-\frac{1}{2}\sum_{j=1}^{m_k}\Bigg((\boldsymbol{z}_k^j)^T\mathbb{E}_{q_{\boldsymbol{x}}^{(\ell)}}\left[\boldsymbol{T}_k\mathbb{E}_{q_{\boldsymbol{\Gamma}}^{(\ell)}}\left[(s\boldsymbol{\Gamma}_k)^{-1}\right]\boldsymbol{T}_k^T\right](\boldsymbol{z}_k^j)\\&-2(\boldsymbol{z}_k^j)^T\mathbb{E}_{q_{\boldsymbol{x}}^{(\ell)}}\left[\boldsymbol{T}_k\mathbb{E}_{q_{\boldsymbol{\Gamma}}^{(\ell)}}\left[(s\boldsymbol{\Gamma}_k)^{-1}\right]\boldsymbol{T}_k^T\boldsymbol{H}\boldsymbol{x}_k\right]\Bigg)\\&+\sum_{j=1}^{m_k}\log\mathcal{N}(\boldsymbol{y}_k^j;\boldsymbol{z}_k^j,\boldsymbol{R}_k)+c_{\mathcal{Z}_k}.\end{aligned} \tag{3.40d}$$

After calculating expectations in (3.40d) a quadratic form in terms of $\boldsymbol{z}_k^j$ is obtained by adding appropriate constants to $c_{\mathcal{Z}_k}$ term. The variational update for the final distribution of noise-free measurements

$$q_{\mathcal{Z}}^{(\ell+1)}=\prod_{j=1}^{m_k}\mathcal{N}(\boldsymbol{z}_k^j;\hat{\boldsymbol{z}}_k^{j,(\ell+1)},\Sigma_k^{z,(\ell+1)}), \tag{3.41}$$

is obtained by the following equations,

$$\Sigma_k^{z,(\ell+1)}=\left(\boldsymbol{R}_k^{-1}+\mathbb{E}_{q_{\boldsymbol{x}}^{(\ell)}}\left[\boldsymbol{T}\mathbb{E}_{q_{\boldsymbol{\Gamma}}^{(\ell)}}\left[(s\boldsymbol{\Gamma}_k)^{-1}\right]\boldsymbol{T}^T\right]\right)^{-1}, \tag{3.42a}$$

$$\hat{\boldsymbol{z}}_k^{j,(\ell+1)}=\Sigma_k^{z,(\ell+1)}\left(\boldsymbol{R}_k^{-1}\boldsymbol{y}_k^j+\mathbb{E}_{q_{\boldsymbol{x}}^{(\ell)}}\left[\boldsymbol{T}\mathbb{E}_{q_{\boldsymbol{\Gamma}}^{(\ell)}}\left[(s\boldsymbol{\Gamma}_k)^{-1}\right]\boldsymbol{T}^T\boldsymbol{H}\boldsymbol{x}_k\right]\right). \tag{3.42b}$$

The closed-form expressions of the expectations are as follows;

$$\mathbb{E}_{q_{\boldsymbol{x}}^{(\ell)}}\left[\boldsymbol{T}_k\mathbb{E}_{q_{\boldsymbol{\Gamma}}^{(\ell)}}\left[(s\boldsymbol{\Gamma}_k)^{-1}\right]\boldsymbol{T}_k^T\right]=\mathbb{E}_{q_{\varphi}^{(\ell)}}\left[\boldsymbol{T}_k\overline{(s\boldsymbol{\Gamma}_k)^{-1}}\boldsymbol{T}_k^T\right], \tag{3.43}$$

$$\begin{aligned}&\mathbb{E}_{q_{\boldsymbol{x}}^{(\ell)}}\left[\boldsymbol{T}_k\mathbb{E}_{q_{\boldsymbol{\Gamma}}^{(\ell)}}\left[(s\boldsymbol{\Gamma}_k)^{-1}\right]\boldsymbol{T}_k^T\boldsymbol{H}\boldsymbol{x}_k\right]=\\&\quad\mathbb{E}_{q_{\varphi}^{(\ell)}}\left[\boldsymbol{T}_k\overline{(s\boldsymbol{\Gamma}_k)^{-1}}\boldsymbol{T}_k^T\right]\tilde{\boldsymbol{H}}\left(\hat{\tilde{\boldsymbol{x}}}_{(k|k)}^{(\ell)}-(\hat{\varphi}_{(k|k)}^{(\ell)}/P_{\varphi(k|k)}^{(\ell)})\boldsymbol{P}_{\tilde{\boldsymbol{x}},\varphi,(k|k)}^{(\ell)}\right)\\&+\mathbb{E}_{q_{\varphi}^{(\ell)}}\left[\varphi_k\boldsymbol{T}_k\overline{(s\boldsymbol{\Gamma}_k)^{-1}}\boldsymbol{T}_k^T\right]\tilde{\boldsymbol{H}}\left(\left(P_{\varphi(k|k)}^{(\ell)}\right)^{-1}\boldsymbol{P}_{\tilde{\boldsymbol{x}},\varphi,(k|k)}^{(\ell)}\right).\end{aligned} \tag{3.44}$$

In (3.44) Lemma 1 can be used with $\varphi=-\varphi_{(k|k)}^{(\ell)}$ with the state partitioning given in (3.34).

A pseudocode of the trajectory-aligned model's measurement update is presented in Algorithm 1.

Algorithm 1: Measurement Update for Trajectory-Aligned Model

Input: I_{max}, $\mathcal{Y}_k$, and $p(\boldsymbol{x}_k, \boldsymbol{\Gamma}_k|\mathcal{Y}_{1:k-1})$ as in (3.7)

Initialization:

Set initial densities:

$$\boldsymbol{x}_k^{(0)} \sim q_{\boldsymbol{x}}^{(0)} = \mathcal{N}(\boldsymbol{x}_k; \hat{\boldsymbol{x}}_{k|k-1}, \boldsymbol{P}_{k|k-1}),$$
$$\boldsymbol{\Gamma}_k^{(0)} \sim q_{\boldsymbol{\Gamma}}^{(0)} = \prod_{i=1}^{2} \mathcal{IG}(\gamma_k^i; \alpha_{k|k-1}^i, \beta_{k|k-1}^i),$$
$$\mathcal{Z}_k^{(0)} \sim q_{\mathcal{Z}}^{(0)} = \prod_{j=1}^{m_k} \mathcal{N}(\boldsymbol{z}_k^j; \boldsymbol{y}_k^j, \mathbb{E}_{q_{\boldsymbol{\Gamma}}^{(0)}}[s\boldsymbol{\Gamma}_k])$$

Set $\ell = 0$

while $\ell < I_{max}$ **do**

- **Update state density** $q_{\boldsymbol{x}}^{(\ell+1)}$ **as in** (3.22):
- Update $\boldsymbol{P}_{k|k}^{(\ell+1)}$ (3.23a)
- Update $\hat{\boldsymbol{x}}_{k|k}^{(\ell+1)}$ (3.23b)
- **Update extent density** $q_{\boldsymbol{\Gamma}}^{(\ell+1)}$ **as in** (3.30):
- **for** *each dimension* i **do**
  - Update $\alpha_{k|k}^{i,(\ell+1)}$ (3.31a)
  - Update $\beta_{k|k}^{i,(\ell+1)}$ (3.31b)
- **end**
- **Update noise-free density** $q_{\mathcal{Z}}^{(\ell+1)}$ **as in** (3.41):
- Update $\Sigma_k^{z,(\ell+1)}$ (3.42a)
- **for** *each measurement* $\mathbf{y}_k^j$ **do**
  - Update $\hat{\boldsymbol{z}}_k^{j,(\ell+1)}$ (3.42b)
- **end**
- Increase ℓ

end

Output: $p(\boldsymbol{x}_k, \boldsymbol{\Gamma}_k|\mathcal{Y}_{1:k}) \approx q_{\boldsymbol{x}}^{(\ell)}(\boldsymbol{x}_k) q_{\boldsymbol{\Gamma}}^{(\ell)}(\boldsymbol{\Gamma}_k)$

3.3 Drift Model

Targets like maritime vessels, UAVs, insects, and microscopic creatures often exhibit motion characteristics where their movements are relatively independent of their extent orientations. Due to its constraints on the heading angle, the trajectory-aligned model cannot capture such motion characteristics. In contrast, the proposed drift model treats the extent orientation and kinematic state separately, resulting in better tracking accuracy for targets with holonomic motions.

3.3.1 Measurement Update for the Drift Model

Consider the prior density

$$p(\boldsymbol{x}_0, \boldsymbol{\Gamma}_0, \boldsymbol{\vartheta}_0) = \mathcal{N}\left(\boldsymbol{x}_0; \hat{\boldsymbol{x}}_0, \boldsymbol{P}_0\right) \times \prod_{i=1}^{2} \mathcal{IG}(\gamma_0^i; \alpha_0^i, \beta_0^i) \times \mathcal{N}(\boldsymbol{\vartheta}_0; \hat{\boldsymbol{\vartheta}}_0, \boldsymbol{\Theta}_0), \tag{3.45}$$

where the last factor represents the orientation state and its kinematics. In the following, we derive analytical update equations to perform variational inference and obtain approximate posteriors.

Assume that we have the prior density at time k as

$$p(\boldsymbol{x}_k, \boldsymbol{\Gamma}_k, \boldsymbol{\vartheta}_k | \mathcal{Y}_{1:k-1}) = \mathcal{N}(\boldsymbol{x}_k; \hat{\boldsymbol{x}}_{k|k-1}, \boldsymbol{P}_{k|k-1})\, \mathcal{N}(\boldsymbol{\vartheta}_k; \hat{\boldsymbol{\vartheta}}_{k|k-1}, \boldsymbol{\Theta}_{k|k-1}) \times \prod_{i=1}^{2} \mathcal{IG}(\gamma_k^i; \alpha_{k|k-1}^i, \beta_{k|k-1}^i). \tag{3.46}$$

The joint density of the measurements with the noise-free measurements z_k^j is defined as

$$p(\boldsymbol{y}_k^j, \boldsymbol{z}_k^j | \boldsymbol{x}_k, \boldsymbol{\Gamma}_k, \boldsymbol{\vartheta}_k) = \mathcal{N}(\boldsymbol{y}_k^j; \boldsymbol{z}_k^j, \boldsymbol{R}_k)\, \mathcal{N}(\boldsymbol{z}_k^j; \boldsymbol{H}\boldsymbol{x}_k, s\boldsymbol{T}_k \boldsymbol{\Gamma}_k \boldsymbol{T}_k), \tag{3.47}$$

where $\boldsymbol{T}_k$ denotes the rotation matrix $\boldsymbol{T}(\boldsymbol{H}_\theta \boldsymbol{\vartheta}_k)$, $\boldsymbol{H}$ and $\boldsymbol{H}_\theta$ are the matrices selecting the Cartesian position from the kinematic state and the heading angle from the orientation state, respectively. Unlike the trajectory-aligned model, the joint likelihood

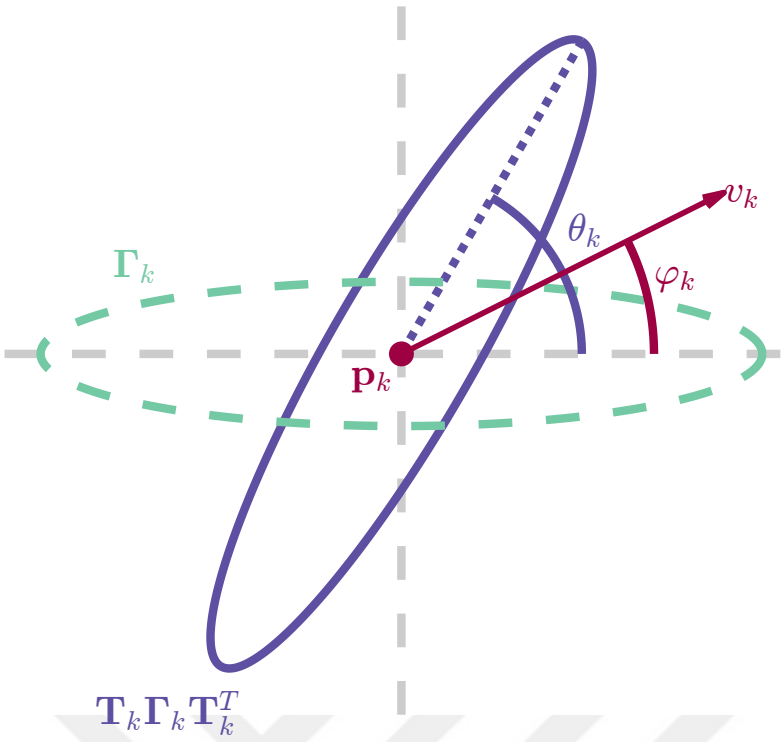


Figure 3.3: An illustration of the state variables used in drift model, where $\boldsymbol{p}_k = \begin{bmatrix} p_k^x & p_k^y \end{bmatrix}^T$ the cartesian position of the ellipse center, v_k is the speed, φ_k is the heading angle, θ_k is the extent orientation angle, $\mathbf{\Gamma}_k$ is the diagonal extent matrix and $\boldsymbol{T}_k = \boldsymbol{T}(\theta_k)$ is the rotation matrix.

depends on the independent orientation state $\boldsymbol{\vartheta}_k$. Within the VB framework, we seek the posterior density in a factorized form,

$$p(\boldsymbol{x}_k, \mathbf{\Gamma}_k, \boldsymbol{\vartheta}_k, \mathcal{Z}_k | \mathcal{Y}_{1:k}) \approx q_{\boldsymbol{x}}(\boldsymbol{x}_k) q_{\mathbf{\Gamma}}(\mathbf{\Gamma}_k) q_{\boldsymbol{\vartheta}}(\boldsymbol{\vartheta}_k) q_{\mathcal{Z}}(\mathcal{Z}_k). \tag{3.48}$$

The joint density $p(\boldsymbol{x}_k, \mathbf{\Gamma}_k, \boldsymbol{\vartheta}_k, \mathcal{Z}_k, \mathcal{Y}_k \mid \mathcal{Y}_{1:k-1})$ for the drift model is

$$\begin{aligned} p(\boldsymbol{x}_k, \mathbf{\Gamma}_k, \boldsymbol{\vartheta}_k, \mathcal{Z}_k, \mathcal{Y}_k \mid \mathcal{Y}_{1:k-1}) = & \prod_{j=1}^{m_k} \mathcal{N}(\boldsymbol{y}_k^j; \boldsymbol{z}_k^j, \boldsymbol{R}) \, \mathcal{N}(z_k^j; \boldsymbol{H}\boldsymbol{x}_k, s\boldsymbol{T}_k \mathbf{\Gamma}_k \boldsymbol{T}_k^T) \\ & \times \mathcal{N}(\boldsymbol{x}_k; \hat{\boldsymbol{x}}_{k|k-1}, \boldsymbol{P}_{k|k-1}) \prod_{i=1}^{2} \mathcal{IG}(\gamma_k^i; \alpha_{k|k-1}^i, \beta_{k|k-1}^i) \\ & \times \mathcal{N}(\boldsymbol{\vartheta}_k; \hat{\boldsymbol{\vartheta}}_{k|k-1}, \boldsymbol{\Theta}_{k|k-1}). \end{aligned} \tag{3.49}$$

Similar to the trajectory-aligned model, we perform fixed point iterations for each factorized density in (3.48) one at a time. We start with the initial distributions with

sufficient statistics;

$$\boldsymbol{P}_{k|k}^{(0)} = \boldsymbol{P}_{k|k-1}, \quad (3.50a) \qquad \hat{\boldsymbol{x}}_{k|k}^{(0)} = \hat{\boldsymbol{x}}_{k|k-1}, \quad (3.50b)$$

$$\alpha_{k|k}^{i,(0)} = \alpha_{k|k-1}^{i}, \quad (3.50c) \qquad \beta_{k|k}^{i,(0)} = \beta_{k|k-1}^{i}, \quad (3.50d)$$

$$\boldsymbol{\Theta}_{k|k}^{(0)} = \boldsymbol{\Theta}_{k|k-1}, \quad (3.50e) \qquad \hat{\boldsymbol{\vartheta}}_{k|k}^{(0)} = \hat{\boldsymbol{\vartheta}}_{k|k}, \quad (3.50f)$$

$$\Sigma_{k}^{z,(0)} = \mathbb{E}_{q_{\boldsymbol{\Gamma}}^{(0)}}\left[s\boldsymbol{\Gamma}_k\right], \quad (3.50g) \qquad \hat{\boldsymbol{z}}_{k}^{j,(0)} = \boldsymbol{y}_{k}^{j}. \quad (3.50h)$$

where $\mathbb{E}_{q_{\boldsymbol{\Gamma}}^{(0)}}[(s\boldsymbol{\Gamma}_k)] = \operatorname{diag}\left(\frac{s\beta_{k|k-1}^{1}}{\alpha_{k|k-1}^{1}-1}, \frac{s\beta_{k|k-1}^{2}}{\alpha_{k|k-1}^{2}-1}\right)$ and proceed with fixed-point iterations. The $(\ell+1)^{\text{st}}$ iterations for the densities are given in the following subsections.

3.3.1.1 Update of $\hat{q}_{\boldsymbol{x}}^{(\ell+1)}(\cdot)$

Using the joint density (3.49) and applying the update rule in (3.14) the update rule for $\hat{q}_{\boldsymbol{x}}^{(\ell+1)}(\cdot)$ is obtained.

$$\log q_{\boldsymbol{x}}^{(\ell+1)}(\boldsymbol{x}_k) = \mathbb{E}_{\setminus \boldsymbol{x}_k}\left[\log p(\boldsymbol{x}_k, \boldsymbol{\Gamma}_k, \boldsymbol{\vartheta}_k \mathcal{Z}_k, \mathcal{Y}_k | \mathcal{Y}_{1:k-1})\right] + c_{\boldsymbol{x}_k}, \quad (3.51a)$$

$$= -\frac{1}{2}\sum_{j=1}^{m_k} \operatorname{Tr}\left((\overline{\boldsymbol{z}_k^j} - \boldsymbol{H}\boldsymbol{x}_k)(\cdot)^T \mathbb{E}_{q_{\boldsymbol{\Gamma}}^{(\ell)}, q_{\boldsymbol{\vartheta}}^{(\ell)}}\left[\left(s\boldsymbol{T}_k\boldsymbol{\Gamma}_k\boldsymbol{T}_k^T\right)^{-1}\right]\right)$$

$$+ \log \mathcal{N}(\boldsymbol{x}_{k|k}; \hat{\boldsymbol{x}}_{k|k-1}, \boldsymbol{P}_{k|k-1}) + c_{\boldsymbol{x}_k}. \quad (3.51b)$$

By utilizing the product rule, iterations for the mean and covariance are obtained as

$$\boldsymbol{P}_{k|k}^{(\ell+1)} = \left(\boldsymbol{P}_{k|k-1}^{-1} + m_k \boldsymbol{H}^T \mathbb{E}_{q_{\boldsymbol{\Gamma}}^{(\ell)}, q_{\boldsymbol{\vartheta}}^{(\ell)}}\left[\left(s\boldsymbol{T}_k\boldsymbol{\Gamma}_k\boldsymbol{T}_k^T\right)^{-1}\right]\right)^{-1}, \quad (3.52a)$$

$$\hat{\boldsymbol{x}}_{k|k}^{(\ell+1)} = \boldsymbol{P}_{k|k}^{(\ell+1)}\left(\boldsymbol{P}_{k|k-1}^{-1}\hat{\boldsymbol{x}}_{k|k-1} + \boldsymbol{H}^T \mathbb{E}_{q_{\boldsymbol{\Gamma}}^{(\ell)}, q_{\boldsymbol{\vartheta}}^{(\ell)}}\left[\left(s\boldsymbol{T}_k\boldsymbol{\Gamma}_k\boldsymbol{T}_k^T\right)^{-1}\right]\left(\sum_{j=1}^{m_k}\overline{\boldsymbol{z}_k^j}\right)\right). \quad (3.52b)$$

Where,

$$\mathbb{E}_{q_{\boldsymbol{\Gamma}}^{(\ell)}, q_{\boldsymbol{\vartheta}}^{(\ell)}}\left[\left(s\boldsymbol{T}_k\boldsymbol{\Gamma}_k\boldsymbol{T}_k^T\right)^{-1}\right] = \mathbb{E}_{q_{\boldsymbol{\vartheta}}^{(\ell)}}\left[\boldsymbol{T}_k\overline{(s\boldsymbol{\Gamma}_k)^{-1}}\boldsymbol{T}_k^T\right]. \quad (3.53)$$

In (3.53), Lemma 1 can be used with $\varphi = -\boldsymbol{H}_{\boldsymbol{\vartheta}}\boldsymbol{\vartheta}_k^{(\ell)}$.

3.3.1.2 Update of $\hat{q}_{\boldsymbol{\Gamma}}^{(\ell+1)}(\cdot)$

Using the joint density (3.49) and applying the update rule in (3.14) the update rule for $\hat{q}_{\boldsymbol{\Gamma}}^{(\ell+1)}(\cdot)$ is obtained.

$$\log q_{\mathbf{\Gamma}}^{(\ell+1)}(\mathbf{\Gamma}_k) = \mathbb{E}_{\setminus \mathbf{\Gamma}_k}\left[\log p(\boldsymbol{x}_k, \mathbf{\Gamma}_k, \boldsymbol{\vartheta}_k, \mathcal{Z}_k, \mathcal{Y}_k | \mathcal{Y}_{1:k-1})\right] + c_{\mathbf{\Gamma}_k}, \tag{3.54a}$$

$$= -\frac{1}{2}\sum_{j=1}^{m_k} \mathbb{E}_{\setminus \mathbf{\Gamma}_k}\left[(\boldsymbol{z}_k^j - \boldsymbol{H}\boldsymbol{x}_k)^T \boldsymbol{T}_k (s\mathbf{\Gamma}_k)^{-1} \boldsymbol{T}_k^T (\boldsymbol{z}_k^j - \boldsymbol{H}\boldsymbol{x}_k)\right]$$

$$+ \sum_{i=1}^{2} \log \mathcal{IG}(\gamma_k^i; \alpha_{k|k-1}^i, \beta_{k|k-1}^i) + c_{\mathbf{\Gamma}_k}. \tag{3.54b}$$

The updated extent density

$$q_{\mathbf{\Gamma}}^{(\ell+1)}(\mathbf{\Gamma}_k) = \prod_{i=1}^{2} \mathcal{IG}(\gamma_k^i; \alpha_{k|k}^{i,(\ell+1)}, \beta_{k|k}^{i,(\ell+1)}), \tag{3.55}$$

is found after calculating (3.29d) with the inverse-Gamma density parameters

$$\alpha_{k|k}^{i,(\ell+1)} = \alpha_{k|k-1}^i + \frac{m_k}{2}, \tag{3.56a}$$

$$\beta_{k|k}^{i,(\ell+1)} = \beta_{k|k-1}^i + \sum_{j=1}^{m_k} \frac{\mathbb{E}_{q_{\boldsymbol{\vartheta}}}\left[\boldsymbol{T}_k^T \overline{(z_k^j - \boldsymbol{H}\boldsymbol{x}_k)(\cdot)^T} \boldsymbol{T}_k\right]_{ii}}{2s}. \tag{3.56b}$$

where,

$$\mathbb{E}_{q_{\mathcal{Z}}^{(\ell)}, q_{\mathbf{x}}^{(\ell)}}[(\mathbf{z}_k^j - \boldsymbol{H}\mathbf{x}_k)(.)^T] = \overline{(\mathbf{z}_k^j - \boldsymbol{H}\mathbf{x}_k)(.)^T} =$$

$$\boldsymbol{\Sigma}_k^{z,\ell} + \boldsymbol{H} P_{k|k}^{(\ell)} \boldsymbol{H}^T + \left(\hat{\mathbf{z}}_{k|k}^{j,(\ell)} - \boldsymbol{H}\hat{\mathbf{x}}_{k|k}^{(\ell)}\right)(\,\cdot\,)^T. \tag{3.57}$$

The rest of the expectations can be calculated as in (3.53). Lemma 1 can be used with $\varphi = \boldsymbol{H}_{\boldsymbol{\vartheta}} \boldsymbol{\vartheta}_k^{(\ell)}$.

3.3.1.3 Update of $\hat{q}_{\mathcal{Z}}^{(\ell+1)}(\cdot)$

Using the joint density (3.49) and applying the update rule in (3.14) the update rule for $\hat{q}_{\mathcal{Z}}^{(\ell+1)}(\cdot)$ is obtained.

$$\log q_{\mathcal{Z}}^{(\ell+1)}(\mathcal{Z}_k) = \mathbb{E}_{\setminus \mathcal{Z}_k}\left[\log p(\boldsymbol{x}_k, \mathbf{\Gamma}_k, \boldsymbol{\vartheta}_k, \mathcal{Z}_k, \mathcal{Y}_k | \mathcal{Y}_{1:k-1})\right] + c_{\mathbf{\Gamma}_k}, \tag{3.58a}$$

$$= -\frac{1}{2}\sum_{j=1}^{m_k} \mathrm{Tr}\left((\boldsymbol{z}_k^j - \boldsymbol{H}\overline{\boldsymbol{x}}_k)(\cdot)^T \mathbb{E}_{q_{\mathbf{\Gamma}}^{(\ell)}, q_{\boldsymbol{\vartheta}}^{(\ell)}}\left[(s\boldsymbol{T}_k \mathbf{\Gamma}_k \boldsymbol{T}_k^T)^{-1}\right]\right)$$

$$+ \sum_{j=1}^{m_k} \log \mathcal{N}(\boldsymbol{y}_k^j; \boldsymbol{z}_k^j, \boldsymbol{R}_k) + c_{\mathcal{Z}_k}. \tag{3.58b}$$

The variational update for the final distribution of noise-free measurements

$$q_{\mathcal{Z}}^{(\ell+1)} = \prod_{j=1}^{m_k} \mathcal{N}(\boldsymbol{z}_k^j; \hat{\boldsymbol{z}}_k^{j,(\ell+1)}, \Sigma_k^{z,(\ell+1)}), \tag{3.59}$$

is obtained by the equations;

$$\Sigma_k^{z,(\ell+1)} = \left(\boldsymbol{R}_k^{-1} + \mathbb{E}_{q_{\boldsymbol{\Gamma}}^{(\ell)}, q_{\boldsymbol{\vartheta}}^{(\ell)}} \left[(s\boldsymbol{T}_k \boldsymbol{\Gamma}_k \boldsymbol{T}_k^T)^{-1} \right] \right)^{-1}, \tag{3.60a}$$

$$\hat{\boldsymbol{z}}_k^{j,(\ell+1)} = \Sigma_k^{z,(\ell+1)} \left(\boldsymbol{R}_k^{-1} \boldsymbol{y}_k^j + \mathbb{E}_{q_{\boldsymbol{\Gamma}}^{(\ell)}, q_{\boldsymbol{\vartheta}}^{(\ell)}} \left[(s\boldsymbol{T}_k \boldsymbol{\Gamma}_k \boldsymbol{T}_k^T)^{-1} \right] \boldsymbol{H} \overline{\boldsymbol{x}}_k \right). \tag{3.60b}$$

The necessary expectations are as follows;

$$\mathbb{E}_{q_{\boldsymbol{x}}^{(\ell)}} \left[\boldsymbol{T}_k \mathbb{E}_{q_{\boldsymbol{\Gamma}}^{(\ell)}} \left[(s\boldsymbol{\Gamma}_k)^{-1} \right] \boldsymbol{T}_k^T \right] = \mathbb{E}_{q_{\varphi}^{(\ell)}} \left[\boldsymbol{T}_k \overline{(s\boldsymbol{\Gamma}_k)^{-1}} \boldsymbol{T}_k^T \right]. \tag{3.61}$$

$$\begin{aligned} &\mathbb{E}_{q_{\boldsymbol{x}}^{(\ell)}} \left[\boldsymbol{T}_k \mathbb{E}_{q_{\boldsymbol{\Gamma}}^{(\ell)}} \left[(s\boldsymbol{\Gamma}_k)^{-1} \right] \boldsymbol{T}_k^T \boldsymbol{H} \boldsymbol{x}_k \right] = \\ &\quad \mathbb{E}_{q_{\varphi}^{(\ell)}} \left[\boldsymbol{T}_k \overline{(s\boldsymbol{\Gamma}_k)^{-1}} \boldsymbol{T}_k^T \right] \tilde{\boldsymbol{H}} \left(\hat{\tilde{\boldsymbol{x}}}_{(k|k)}^{(\ell)} - (\hat{\varphi}_{(k|k)}^{(\ell)} / P_{\varphi(k|k)}^{(\ell)}) \boldsymbol{P}_{\tilde{\boldsymbol{x}},\varphi,(k|k)}^{(\ell)} \right) \\ &+ \mathbb{E}_{q_{\varphi}^{(\ell)}} \left[\varphi_k \boldsymbol{T}_k \overline{(s\boldsymbol{\Gamma}_k)^{-1}} \boldsymbol{T}_k^T \right] \tilde{\boldsymbol{H}} \left((P_{\varphi(k|k)}^{(\ell)})^{-1} \boldsymbol{P}_{\tilde{\boldsymbol{x}},\varphi,(k|k)}^{(\ell)} \right). \end{aligned} \tag{3.62}$$

In (3.62) Lemma 1 can be used with $\varphi = -\boldsymbol{H}_{\boldsymbol{\vartheta}} \boldsymbol{\vartheta}_k^{(\ell)}$.

3.3.1.4 Update of $\hat{q}_{\boldsymbol{\vartheta}}^{(\ell+1)}(\cdot)$

Using the joint density (3.49) and applying the update rule in (3.14) the update rule for $\hat{q}_{\boldsymbol{\vartheta}}^{(\ell+1)}(\cdot)$ is obtained.

$$\log q_{\boldsymbol{\vartheta}}^{(\ell+1)}(\boldsymbol{\vartheta}_k) = \mathbb{E}_{\setminus \boldsymbol{\vartheta}_k} \left[\log p(\boldsymbol{x}_k, \boldsymbol{\Gamma}_k, \boldsymbol{\vartheta}_k, \mathcal{Z}_k, \mathcal{Y}_k | \mathcal{Y}_{1:k-1}) \right] + c_{\boldsymbol{\vartheta}_k}, \tag{3.63a}$$

$$= -\frac{1}{2} \sum_{j=1}^{m_k} \mathbb{E}_{\setminus \boldsymbol{\vartheta}_k} \left[\operatorname{Tr} \left((\boldsymbol{z}_k^j - \boldsymbol{H} \overline{\boldsymbol{x}}_k)(\cdot)^T (s\boldsymbol{T}_k \boldsymbol{\Gamma}_k \boldsymbol{T}_k^T)^{-1} \right) \right]$$

$$+ \log \mathcal{N}(\boldsymbol{\vartheta}_{k|k}; \hat{\boldsymbol{\vartheta}}_{k|k-1}, \boldsymbol{\Theta}_{k|k-1}) + c_{\boldsymbol{\vartheta}_k}. \tag{3.63b}$$

To obtain a quadratic form in terms of $\boldsymbol{\vartheta}_k$ similar to the trajectory-aligned model a first-order Taylor Series approximation is used for the function $g_j(\boldsymbol{\vartheta}_k) \triangleq \boldsymbol{T}^T(\boldsymbol{z}_k^j - \boldsymbol{H}\boldsymbol{x}_k)$ around $\hat{\boldsymbol{\vartheta}}_{k|k}^{(\ell)}$.

$$g_j(\boldsymbol{\vartheta}_k) \approx g_j(\hat{\boldsymbol{\vartheta}}_{k|k}^{(\ell)}) + \nabla g_j^T(\hat{\boldsymbol{\vartheta}}_{k|k}^{(\ell)})(\boldsymbol{\vartheta}_k - \hat{\boldsymbol{\vartheta}}_{k|k}^{(\ell)}), \tag{3.64a}$$

$$c_j \triangleq g(\hat{\boldsymbol{\vartheta}}_{k|k}^{(\ell)}) - \nabla g_j^T(\hat{\boldsymbol{\vartheta}}_{k|k}^{(\ell)}) \hat{\boldsymbol{\vartheta}}_{k|k}^{(\ell)}, \tag{3.64b}$$

$$d_j \triangleq -\nabla g_j^T(\hat{\boldsymbol{\vartheta}}_{k|k}^{(\ell)}). \tag{3.64c}$$

The expectation in (3.63b) can be approximated as

$$\mathbb{E}_{\backslash \boldsymbol{\vartheta}_k}\left[(c_j - d_j\boldsymbol{\vartheta}_k)^T(s\boldsymbol{\Gamma}_k)^{-1}(c_j - d_j\boldsymbol{\vartheta}_k)\right]. \tag{3.65}$$

$$c_j = \dot{\boldsymbol{T}}_{k,\ell}^T\left(\boldsymbol{z}_k^j - \boldsymbol{H}\boldsymbol{x}_k\right)\boldsymbol{H}_{\boldsymbol{\vartheta}}\hat{\boldsymbol{\vartheta}}_{(k|k)}^{(\ell)} + \boldsymbol{T}_{k,\ell}^T\left(\boldsymbol{z}_k^j - \boldsymbol{H}\boldsymbol{x}_k\right), \tag{3.66}$$

$$d_j = -\dot{\boldsymbol{T}}_{k,\ell}^T\left(\boldsymbol{z}_k^j - \boldsymbol{H}\boldsymbol{x}_k\right), \tag{3.67}$$

where $\boldsymbol{T}_{k,\ell} = \boldsymbol{T}(\boldsymbol{H}_{\boldsymbol{\vartheta}}\hat{\boldsymbol{\vartheta}}_{(k|k)}^{(\ell)})$ with the definition as in and $\dot{\boldsymbol{T}}_{k,\ell} = \dot{\boldsymbol{T}}(\boldsymbol{H}_{\boldsymbol{\vartheta}}\hat{\boldsymbol{\vartheta}}_{(k|k)}^{(\ell)})$. The approximate update for the orientation state is obtained as

$$q_{\boldsymbol{\vartheta}}^{(\ell+1)}(\boldsymbol{\vartheta}_k) = \mathcal{N}(\boldsymbol{\vartheta}_k; \hat{\boldsymbol{\vartheta}}_{k|k}^{(\ell+1)}, \boldsymbol{\Theta}_{k|k}^{(\ell+1)}), \tag{3.68}$$

where

$$\boldsymbol{\Theta}_{k|k}^{(\ell+1)} = \left(\boldsymbol{\Theta}_{k|k-1}^{-1} + \boldsymbol{\Delta}\right)^{-1}, \tag{3.69a}$$

$$\hat{\boldsymbol{\vartheta}}_{k|k}^{(\ell+1)} = \boldsymbol{\Theta}_{k|k}^{(\ell+1)}\left(\boldsymbol{\Theta}_{k|k-1}^{-1}\hat{\boldsymbol{\vartheta}}_{k|k-1} + \boldsymbol{\delta}\right), \tag{3.69b}$$

$$\boldsymbol{\Delta} \triangleq \sum_{j=1}^{m_k} \mathbb{E}_{q_{\boldsymbol{x}}^{(\ell)}, q_{\mathcal{Z}}^{(\ell)}}\left[d_j^T \mathbb{E}_{q_{\boldsymbol{\Gamma}}^{(\ell)}}\left[(s\boldsymbol{\Gamma}_k)^{-1}\right] d_j\right], \tag{3.69c}$$

$$\boldsymbol{\delta} \triangleq \sum_{j=1}^{m_k} \mathbb{E}_{q_{\boldsymbol{x}}^{(\ell)}, q_{\mathcal{Z}}^{(\ell)}}\left[d_j^T \mathbb{E}_{q_{\boldsymbol{\Gamma}}^{(\ell)}}\left[(s\boldsymbol{\Gamma}_k)^{-1}\right] c_j\right]. \tag{3.69d}$$

Where the expectations can be obtained as follows;

$$\begin{aligned}
&\mathbb{E}_{q_{\boldsymbol{x}}^{(\ell)}, q_{\mathcal{Z}}^{(\ell)}}\left[d_j^T \mathbb{E}_{q_{\boldsymbol{\Gamma}}^{(\ell)}}\left[(s\boldsymbol{\Gamma}_k)^{-1}\right] d_j\right] = \\
&\qquad \boldsymbol{H}_{\boldsymbol{\vartheta}}^T \operatorname{Tr}\left(\overline{(s\boldsymbol{\Gamma}_k)^{-1}}\dot{\boldsymbol{T}}_{k,\ell}^T\overline{(\mathbf{z}_k^j - \boldsymbol{H}\mathbf{x}_k)(.)^T}\dot{\boldsymbol{T}}_{k,\ell}\right)\boldsymbol{H}_{\boldsymbol{\vartheta}},
\end{aligned} \tag{3.70}$$

$$\begin{aligned}
&\mathbb{E}_{q_{\boldsymbol{x}}^{(\ell)}, q_{\mathcal{Z}}^{(\ell)}}\left[d_j^T \mathbb{E}_{q_{\boldsymbol{\Gamma}}^{(\ell)}}\left[(s\boldsymbol{\Gamma}_k)^{-1}\right] c_j\right] = \\
&\qquad \boldsymbol{H}_{\boldsymbol{\vartheta}}^T \operatorname{Tr}\left(\overline{(s\boldsymbol{\Gamma}_k)^{-1}}\dot{\boldsymbol{T}}_{k,\ell}^T\overline{(\mathbf{z}_k^j - \boldsymbol{H}\mathbf{x}_k)(.)^T}\dot{\boldsymbol{T}}_{k,\ell}\right)\boldsymbol{H}_{\boldsymbol{\vartheta}}\hat{\boldsymbol{\vartheta}}_{(k|k)}^{(\ell)} \\
&\qquad - \boldsymbol{H}_{\boldsymbol{\vartheta}}^T \operatorname{Tr}\left(\overline{(s\boldsymbol{\Gamma}_k)^{-1}}\boldsymbol{T}_{k,\ell}^T\overline{(\mathbf{z}_k^j - \boldsymbol{H}\mathbf{x}_k)(.)^T}\dot{\boldsymbol{T}}_{k,\ell}\right).
\end{aligned} \tag{3.71}$$

A pseudocode of the drift model's measurement update is presented in Algorithm 2.

Algorithm 2: Measurement Update for Drift Model

Input: I_{max}, $\mathcal{Y}_k$ and $p(\boldsymbol{x}_k, \boldsymbol{\Gamma}_k, \boldsymbol{\vartheta}_k|\mathcal{Y}_{1:k-1})$ as in (3.46)

Initialization:

Set initial densities:

$$\begin{aligned}
\boldsymbol{x}_k^{(0)} &\sim q_{\boldsymbol{x}}^{(0)} = \mathcal{N}(\boldsymbol{x}_k; \hat{\boldsymbol{x}}_{k|k-1}, \boldsymbol{P}_{k|k-1}), \\
\boldsymbol{\Gamma}_k^{(0)} &\sim q_{\boldsymbol{\Gamma}}^{(0)} = \prod_{i=1}^{2} \mathcal{IG}(\gamma_k^i; \alpha_{k|k-1}^i, \beta_{k|k-1}^i), \\
\boldsymbol{\vartheta}_k^{(0)} &\sim q_{\boldsymbol{\vartheta}}^{(0)} = \mathcal{N}(\boldsymbol{\vartheta}_k; \hat{\boldsymbol{\vartheta}}_{k|k-1}, \boldsymbol{\Theta}_{k|k-1}), \\
\mathcal{Z}_k^{(0)} &\sim q_{\mathcal{Z}}^{(0)} = \prod_{j=1}^{m_k} \mathcal{N}(\boldsymbol{z}_k^j; \boldsymbol{y}_k^j, \mathbb{E}_{q_{\boldsymbol{\Gamma}}^{(0)}}[s\boldsymbol{\Gamma}_k])
\end{aligned}$$

Set $\ell = 0$

while $\ell < I_{max}$ **do**

  Update state density $q_{\boldsymbol{x}}^{(\ell+1)}$:

  Update $\boldsymbol{P}_{k|k}^{(\ell+1)}$ (3.52a)

  Update $\hat{\boldsymbol{x}}_{k|k}^{(\ell+1)}$ (3.52b)

  Update extent density $q_{\boldsymbol{\Gamma}}^{(\ell+1)}$ **as in** (3.55):

  for *each dimension* i **do**

    Update $\alpha_{k|k}^{i,(\ell+1)}$ (3.56a)

    Update $\beta_{k|k}^{i,(\ell+1)}$ (3.56b)

  end

  Update orientation state density $q_{\boldsymbol{\vartheta}}^{(\ell+1)}$:

  Update $\boldsymbol{\Theta}_{k|k}^{(\ell+1)}$ (3.69a)

  Update $\hat{\boldsymbol{\vartheta}}_{k|k}^{(\ell+1)}$ (3.69b)

  Update noise-free density $q_{\mathcal{Z}}^{(\ell+1)}$:

  Update $\Sigma_k^{z,(\ell+1)}$ (3.60a)

  for *each measurement* $\mathbf{y}_k^j$ **do**

    Update $\hat{\boldsymbol{z}}_k^{j,(\ell+1)}$ (3.60b)

  end

  Increase ℓ

end

Output: $p(\boldsymbol{x}_k, \Gamma_k, \boldsymbol{\vartheta}_k|\mathcal{Y}_{1:k}) \approx q_{\boldsymbol{x}}^{(\ell)}(\boldsymbol{x}_k) q_{\boldsymbol{\Gamma}}^{(\ell)}(\boldsymbol{\Gamma}_k) q_{\boldsymbol{\vartheta}}^{(\ell)}(\boldsymbol{\vartheta}_k)$

3.4 Time Update

We present a common time update framework for the proposed models involving the calculation of (3.3a) for the state and extent variables at time k,

$$p(\boldsymbol{x}_k, \boldsymbol{X}_k \mid \mathcal{Y}_{1:k-1}) = \int p(\boldsymbol{x}_k, \boldsymbol{X}_k \mid \boldsymbol{x}_{k-1}, \boldsymbol{X}_{k-1}) \times p(\boldsymbol{x}_{k-1}, \boldsymbol{X}_{k-1} \mid \mathcal{Y}_{1:k-1}) \, \mathrm{d}\boldsymbol{x}_{k-1} \, \mathrm{d}\boldsymbol{X}_{k-1}. \tag{3.72}$$

To calculate (3.72) analytically, we make use of the assumptions that are commonly used in the ETT literature [7],

$$p(\boldsymbol{x}_k, \boldsymbol{X}_k \mid \boldsymbol{x}_{k-1}, \boldsymbol{X}_{k-1}) = p(\boldsymbol{x}_k \mid \boldsymbol{x}_{k-1}) p(\boldsymbol{X}_k \mid \boldsymbol{X}_{k-1}), \tag{3.73}$$

$$p(\boldsymbol{x}_k, \boldsymbol{X}_k \mid \mathcal{Y}_{1:k-1}) = p(\boldsymbol{x}_k \mid \mathcal{Y}_{1:k-1}) p(\boldsymbol{X}_k \mid \mathcal{Y}_{1:k-1}). \tag{3.74}$$

With these assumptions (3.72) is written as a product of two distributions,

$$p(\boldsymbol{x}_k, \boldsymbol{X}_k \mid \mathcal{Y}_{1:k-1}) = p(\boldsymbol{x}_k \mid \mathcal{Y}_{1:k-1}) p(\boldsymbol{X}_k \mid \mathcal{Y}_{1:k-1}) \tag{3.75}$$

$$= \int p(\boldsymbol{x}_k \mid \boldsymbol{x}_{k-1}) p(\boldsymbol{x}_{k-1} \mid \mathcal{Y}_{1:k-1}) \, \mathrm{d}\boldsymbol{x}_{k-1} \times \int p(\boldsymbol{X}_k \mid \boldsymbol{X}_{k-1}) p(\boldsymbol{X}_{k-1} \mid \mathcal{Y}_{1:k-1}) \, \mathrm{d}\boldsymbol{X}_{k-1}. \tag{3.76}$$

Assume we have the following kinematic state transition model

$$\boldsymbol{x}_{k+1} = f(\boldsymbol{x}_k) + \boldsymbol{\nu}_k, \quad \boldsymbol{\nu}_k \sim \mathcal{N}(\mathbf{0}, \boldsymbol{Q}_{\boldsymbol{x}}), \tag{3.77}$$

A Gaussian approximation $\mathcal{N}(\hat{\boldsymbol{x}}_{k|k-1}, \boldsymbol{P}_{k|k-1})$ to the predicted density for the kinematic state $p(\boldsymbol{x}_k \mid \mathcal{Y}_{1:k-1})$ can be obtained by linearization. The resulting sufficient statistics of the prediction density are

$$\hat{\boldsymbol{x}}_{k|k-1} = f(\hat{\boldsymbol{x}}_{k-1|k-1}) \tag{3.78}$$

$$\boldsymbol{P}_{k|k-1} = \boldsymbol{F}_{\boldsymbol{x}} \boldsymbol{P}_{k-1|k-1} \boldsymbol{F}_{\boldsymbol{x}}^T + \boldsymbol{Q}_{\boldsymbol{x}}, \tag{3.79}$$

where $\boldsymbol{F}_{\boldsymbol{x}} = \nabla f^T(\hat{\boldsymbol{x}}_{k-1|k-1})$.

To calculate the predicted density of the extent state $p(\boldsymbol{X}_k \mid \mathcal{Y}_{1:k-1})$, we employ a maximum entropy update which is suitable for unknown dynamics [51, Theorem 1].

The sufficient statistics for the predicted extent parameters $\gamma^i_{k|k-1} \sim \mathcal{IG}(\alpha^i_{k|k-1}, \beta^i_{k|k-1})$ are calculated as,

$$\alpha^i_{k|k-1} = \tau\alpha^i_{k-1|k-1}, \quad \beta^i_{k|k-1} = \tau\beta^i_{k-1|k-1}, \tag{3.80}$$

for $i = 1, 2$ where $0 < \tau < 1$ is a forgetting factor.

The time update of the drift model is conducted similarly to the trajectory-aligned model, with differences caused by the additional orientation state $\boldsymbol{\vartheta}_k$. Thus, the time update equations can be given as follows for the sufficient statistics of the extended state;

$$\hat{\boldsymbol{x}}_{k|k-1} = f(\hat{\boldsymbol{x}}_{k-1|k-1}), \tag{3.81a}$$

$$\boldsymbol{P}_{k|k-1} = \boldsymbol{F}_{\boldsymbol{x}}\boldsymbol{P}_{k-1|k-1}\boldsymbol{F}^T_{\boldsymbol{x}} + \boldsymbol{Q}_{\boldsymbol{x}}, \tag{3.81b}$$

$$\hat{\boldsymbol{\vartheta}}_{k|k-1} = \boldsymbol{f}_{\boldsymbol{\vartheta}}(\hat{\boldsymbol{\vartheta}}_{k-1|k-1}), \tag{3.81c}$$

$$\boldsymbol{\Theta}_{k|k-1} = \boldsymbol{F}_{\boldsymbol{\vartheta}}\boldsymbol{\Theta}_{k-1|k-1}\boldsymbol{F}^T_{\boldsymbol{\vartheta}} + \boldsymbol{Q}_{\boldsymbol{\vartheta}}, \tag{3.81d}$$

$$\alpha^i_{k|k-1} = \tau\alpha^i_{k-1|k-1}, \tag{3.81e}$$

$$\beta^i_{k|k-1} = \tau\beta^i_{k-1|k-1}, \tag{3.81f}$$

where

$$\boldsymbol{x}_{k+1} = f(\boldsymbol{x}_k) + \boldsymbol{\nu}_k, \qquad \boldsymbol{\nu}_k \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{Q}_{\boldsymbol{x}}), \tag{3.82a}$$

$$\boldsymbol{\vartheta}_{k+1} = \boldsymbol{f}_{\boldsymbol{\vartheta}}(\boldsymbol{\vartheta}_k) + \boldsymbol{u}_k, \quad \boldsymbol{u}_k \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{Q}_{\boldsymbol{\vartheta}}), \tag{3.82b}$$

and $\boldsymbol{F}_{\boldsymbol{x}} = \nabla f^T(\hat{\boldsymbol{x}}_{k-1|k-1})$, $\boldsymbol{F}_{\boldsymbol{\vartheta}} = \nabla f^T_{\boldsymbol{\vartheta}}(\hat{\boldsymbol{\vartheta}}_{k-1|k-1})$.

CHAPTER 4

EXPERIMENTAL RESULTS

In this section, we evaluate the performances of the proposed methods in both simulated scenarios and real-world tracking tasks involving various sensor modalities. We present two simulated experiments and two experiments with real data in Sections 4.2 and 4.3, respectively. Before presenting the experimental results, we define the evaluation metrics in Section 4.1.

4.1 Performance Metrics

For performance evaluation, we consider the commonly used Gaussian-Wasserstein (GW) distance metric [52] for extent estimates and root-mean-square error (RMSE) for orientation angle estimates. We perform multiple Monte Carlo (MC) runs and report the average errors for each time instant.

The squared GW distance between a pair of estimated and true ellipses is defined as

$$GW^2(\hat{\boldsymbol{c}}, \boldsymbol{c}, \hat{\boldsymbol{X}}, \boldsymbol{X}) \triangleq ||\boldsymbol{c} - \hat{\boldsymbol{c}}||_2^2 + \mathrm{Tr}(\hat{\boldsymbol{X}} + \boldsymbol{X} - 2(\hat{\boldsymbol{X}}^{\frac{1}{2}}\boldsymbol{X}\hat{\boldsymbol{X}}^{\frac{1}{2}})^{\frac{1}{2}}), \tag{4.1}$$

where $(\hat{\boldsymbol{c}}, \hat{\boldsymbol{X}})$ and $(\boldsymbol{c}, \boldsymbol{X})$ represent estimated and true center-extent pairs, respectively. The GW distance for k^{th} time instant averaged over N MC runs is

$$GW_k \triangleq \frac{1}{N}\sum_{i=1}^{N}\sqrt{GW^2(\hat{\boldsymbol{c}}_k^i, \boldsymbol{c}_k^i, \hat{\boldsymbol{X}}_k^i, \boldsymbol{X}_k^i)}, \tag{4.2}$$

and the RMSE of the estimated orientation angle for k^{th} time instant over N MC runs is defined as

$$RMSE_k^{\theta} \triangleq \left(\frac{1}{N}\sum_{i=1}^{N}(\theta_{\text{true},k}^i - \theta_k^i)^2\right)^{1/2}, \tag{4.3}$$

where the superscript i denotes the index of the MC run, $\theta^i_{true,k}$ denotes the ground truth orientation angle, and θ^i_k denotes the estimated orientation angle.

4.2 Simulations

Two simulated examples involving trajectory-aligned and drifting targets are studied in this section. We denote the Aligned and Drift models as P1 and P2, respectively. The alternative methods [19], [22], [21], [15] and [27] are denoted as A1, A2, A3, A4, and A5 respectively. In all experiments, including simulations and real data, we selected a fixed number of iterations $I_{max} = 10$ in the variational updates.

4.2.1 Trajectory-aligned Scenario

This scenario draws inspiration from a ground vehicle navigating through an urban road layout characterized by various bends and twists where we consider the trajectory followed by the elliptical target in Figure 4.1. The trajectory consists of 60 time instants, during which the target alternates between constant velocity and constant turn motions. The target is initially at the origin, and its velocity vector is aligned with its major axis throughout the scenario. For this illustrative example, P1 (Trajectory-Aligned Model), A1 [19], A2 [22], A3 [21] and A4 [15] are compared. For P1 and A1, the forgetting factor parameter is set as $\tau = 0.95$, and a maximum of 10 variational iterations are performed. For P1, the coordinated turn model with polar velocity as in Appendix B is chosen, with process noise covariance $\boldsymbol{Q} = \text{diag}(5, 10^{-2})$. For A1, the constant velocity model in [19, (44a)] with process covariance matrix of $\boldsymbol{Q} = \text{blkdiag}(5 \times \boldsymbol{Q}_{cv} \otimes \mathbb{I}_2, 0.5)$ is selected. The constant velocity model given in [22] is chosen for A2, with process noise covariance matrices for kinematics and shape vectors $\boldsymbol{C}^{\omega}_r = 5 \times \boldsymbol{Q}_{cv} \otimes \mathbb{I}_2$ and $\boldsymbol{C}^{\omega}_p = \text{diag}(0.25, 0.01, 0.01)$, respectively. For A3, the coordinated turn model with Cartesian velocity model [53] is used with process noise covariance $\boldsymbol{Q} = \text{blkdiag}(5 \times \mathbb{I}_2, 0.1)$. For A4, the constant velocity model with process noise covariance matrix $\boldsymbol{Q} = 5 \times \boldsymbol{Q}_{cv} \otimes \mathbb{I}_2$ is selected and the rotation matrix is set as $\boldsymbol{M}_k = \boldsymbol{T}(\omega_k)$, where ω_k is the ground truth yaw rate of the ground truth object. The extent parameters for A3 and A4, $\boldsymbol{Q}_{k+1} = \frac{1}{3}\boldsymbol{V}^{-1}_{k|k}$ and v_{k+1}

are chosen according to [21, Eq. 23].

Time-averaged GW and RMSE values are reported in Table 4.1, and errors throughout the scenario are shown in Figure 4.2. P1 achieves the lowest GW distance values throughout the scenario, illustrating its robustness to sharp turn maneuvers. P1 also demonstrates the ability to accurately capture the ground truth orientation, compared to the alternatives, including A1, which fails to accurately estimate both the orientation and extent of the target in this trajectory-aligned scenario. Note that since A3 and A4 do not explicitly estimate the extent orientation angle, we only compare orientation angle estimations of P1, A1, and A2.

Table 4.1: Time-averaged errors for the aligned scenario

	P1	A1 [19]	A2 [22]	A3 [21]	A4 [15]
GW_{av} (m)	**0.88**	1.17	1.11	1.14	1.0
$RMSE^{\theta}_{av}$ (degrees)	**4.4**	105.9	106.5	-	-

4.2.2 Drifting Scenario

This illustrative example is motivated by rapid maneuvering behavior exhibited by marine vessels such as speedboats, where we consider the trajectory followed by the elliptical target in Figure 4.3. The trajectory consists of 60 time instants, during which the target exhibits a drifting behavior. In this example P2 (Drift Model), A1 [19], A2 [22], A3 [21] and A4 [15] are compared. For P2, constant velocity models of the kinematic state and orientation angle vectors with state transition matrices, $\boldsymbol{F_x} = \boldsymbol{F}_{cv} \otimes \mathbb{I}_2$ and $\boldsymbol{F_\theta} = \boldsymbol{F}_{cv}$, are chosen respectively. The process noise covariance matrices for state and orientation angle transition models are selected as $\boldsymbol{Q_x} = 10 \times \boldsymbol{Q}_{cv} \otimes \mathbb{I}_2$ and $\boldsymbol{Q_\theta} = 0.03 \times \boldsymbol{Q}_{cv}$ respectively. For A1, the constant velocity model as in Section 4.2.1 with the process noise covariance matrix $\boldsymbol{Q} = \text{blkdiag}(10 \times \boldsymbol{Q}_{cv} \otimes \mathbb{I}_2, 3 \times 10^{-2})$ is used. For A2, the state-space model as in Section 4.2.1 with noise covariance matrices $\boldsymbol{C}^{\omega}_{r} = 10 \times \boldsymbol{Q}_{cv} \otimes \mathbb{I}_2$ and $\boldsymbol{C}^{\omega}_{p} = \text{diag}(3 \times 10^{-2}, 0.01, 0.01)$ is employed. For A3, the coordinated turn model with Cartesian velocity model [53] is used with process noise covariance $\boldsymbol{Q} = \text{blkdiag}(10 \times \mathbb{I}_2, 3 \times 10^{-2})$. For A4, the constant velocity model with

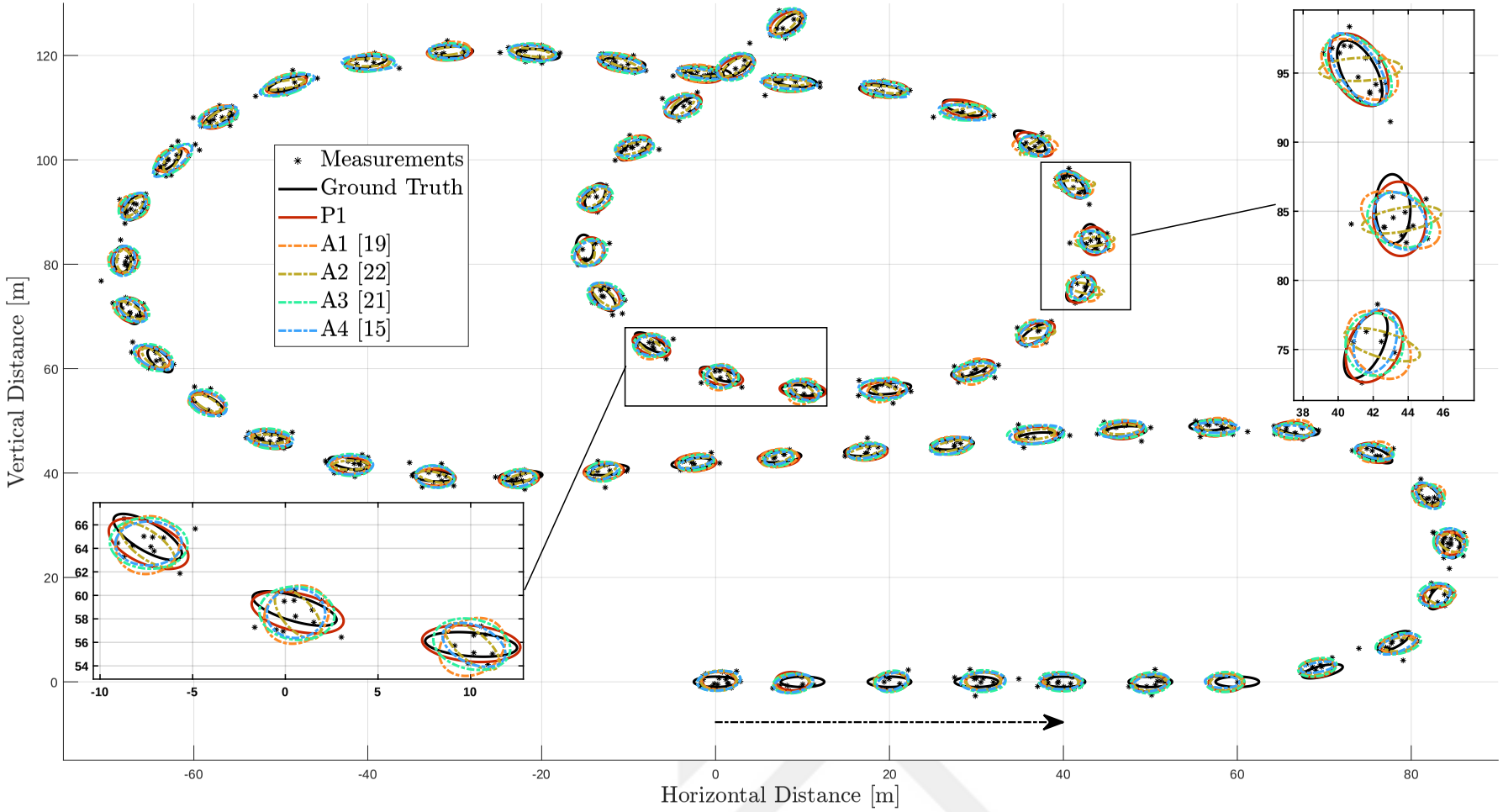
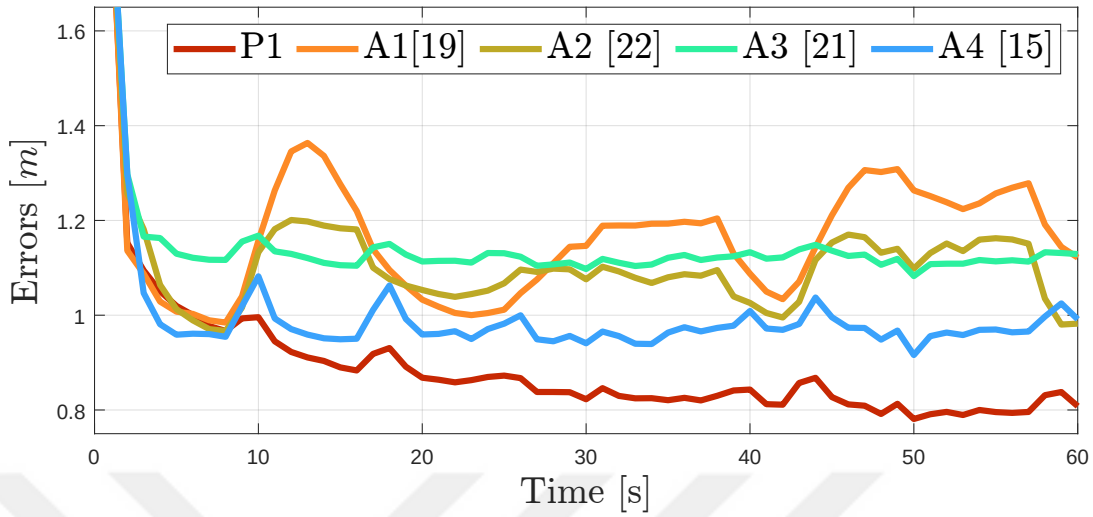
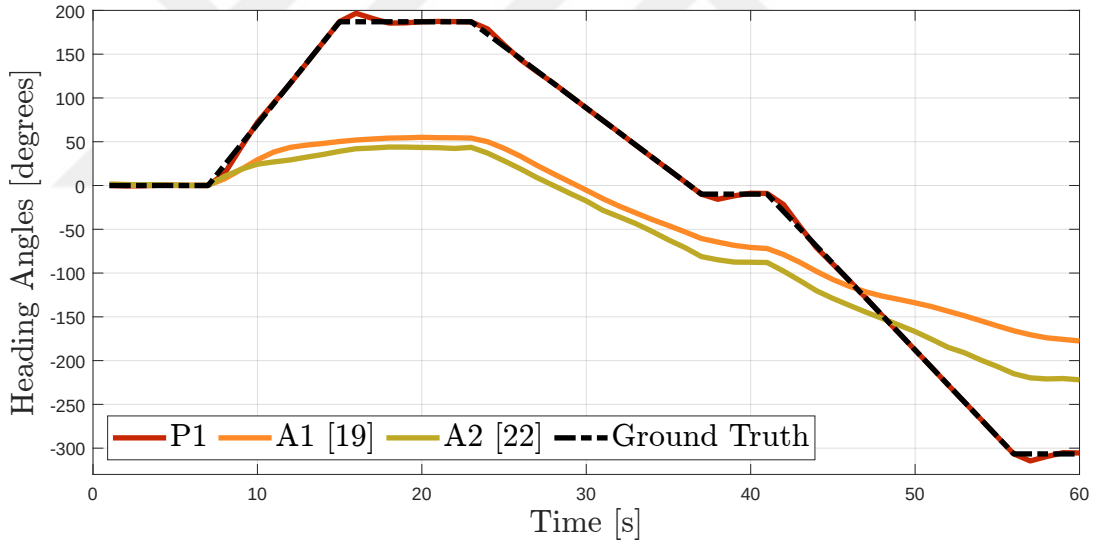


Figure 4.1: The trajectory described in Section 4.2.1. The object is initially at the origin, and the arrow direction indicates the initial direction of motion.



(a) Comparison of GW distances of P1, A1, A2, A3, A4 to the ground truth state in the trajectory given in Figure 4.1.



(b) Comparison of orientation angle estimations of P1, A1, and A2 to the ground truth orientation for the trajectory given in Figure 4.1.

Figure 4.2: GW distance errors (a) and orientation estimates (b) averaged over 10^3 MC runs for the simulated experiment in Section 4.2.1.

Table 4.2: Time-averaged errors for the drift scenario

	P2	A1 [19]	A2 [22]	A3 [21]	A4 [15]
GW_{av} (m)	$\mathbf{1.35}$	1.57	1.57	1.82	1.56
$RMSE^{\theta}_{av}$ (degrees)	$\mathbf{8.5}$	14.1	12.3	-	-

process noise covariance matrix $\boldsymbol{Q} = 10 \times \boldsymbol{Q}_{cv} \otimes \mathbb{I}_2$ is selected. The extent transition parameters $\boldsymbol{Q}_{k+1} = \frac{1}{3}\boldsymbol{V}_{k|k}^{-1}$, v_{k+1} are chosen according to [15, Eq. 37] for both A3 and A4, and the rotation matrix is set as $\boldsymbol{M}_k = \boldsymbol{T}(\omega_k)$ for A4, where ω_k is the ground truth yaw rate of the ground truth object.

Time-averaged GW and RMSE values are reported in Table 4.2, and GW values throughout the scenario are plotted in Figure 4.4. P2 achieves the minimum average RMSE and GW results compared to A1, A2, and A4. Note that GW values tend to increase at the beginning of turn maneuvers for all methods; P2 can estimate the target turn rate relatively quickly and recover its performance. We only compare orientation angle estimations of P2, A1, and A2 since A3 and A4 do not explicitly estimate the extent of orientation angle.

4.2.3 Effect of Measurement Noise Covariance and Number of Iterations

In this section, we investigate how the performance of the proposed methods depends on the measurement noise covariance and the maximum number of iterations allowed in the variational updates. For this purpose, we repeated the experiment in 4.2.1 while varying the measurement noise covariance as $R = \sigma_r^2 \times \mathbb{I}_2$, where $\sigma_r^2 \in \{0.01, 1, 5, 10, 20, 30, 50\}$. For each of these cases, we run the algorithm P1 (Trajectory-Aligned Model) with the number of variational iterations $I_{max} \in \{10, 25, 50\}$. The remaining parameters are kept the same. No additional tuning is performed to improve the performance. 100 MC runs are simulated for each case.

The GW and RMSE performance metrics are presented in Tables 4.3 and 4.4, respectively. As expected, the algorithm's performance degrades with increasing noise levels. When the sensor noise standard deviation exceeds 4 times the minor axis length

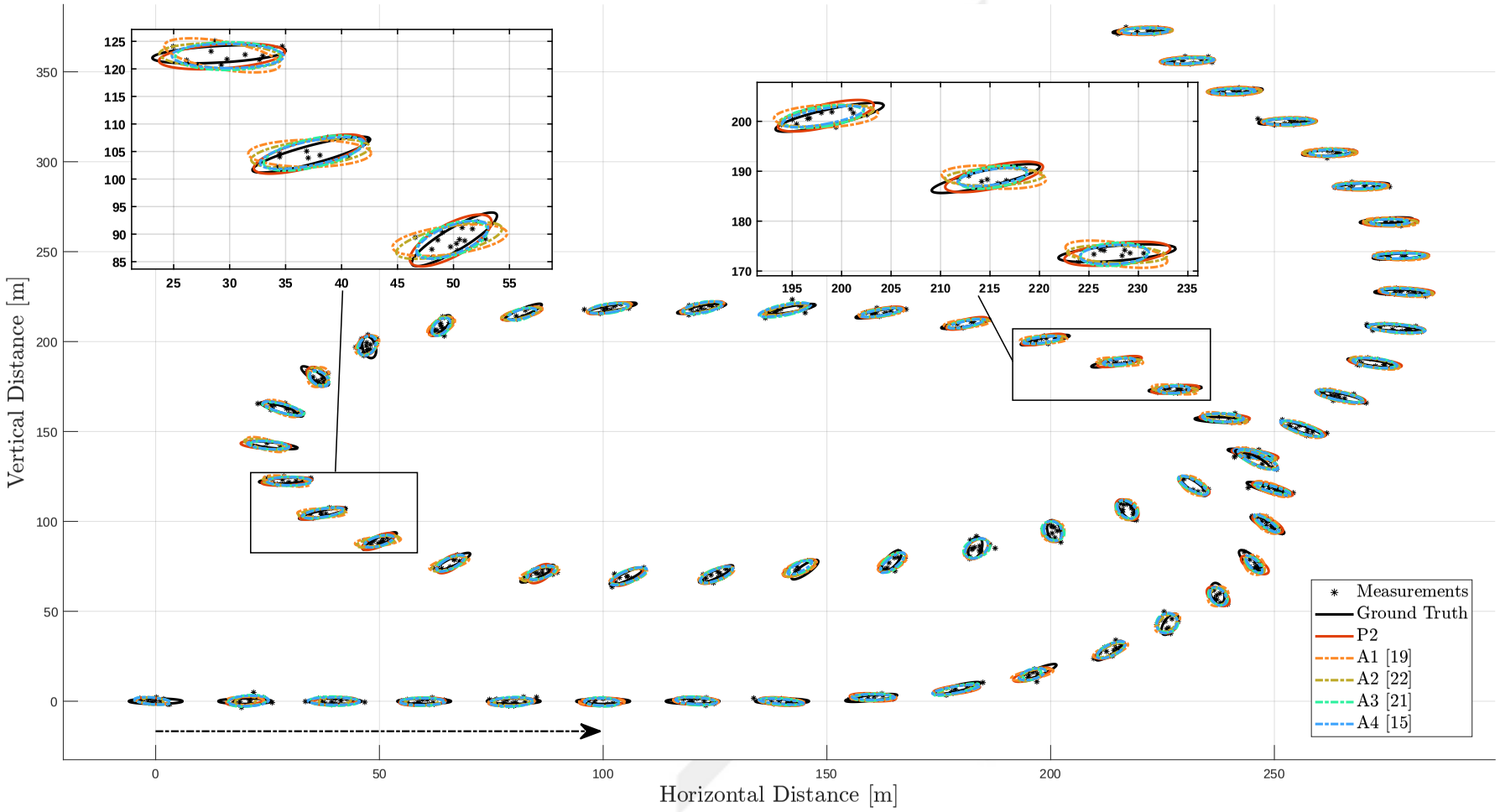


Figure 4.3: The trajectory described in Section 4.2.2. The object is initially at the origin, and the arrow direction indicates the initial direction of motion.

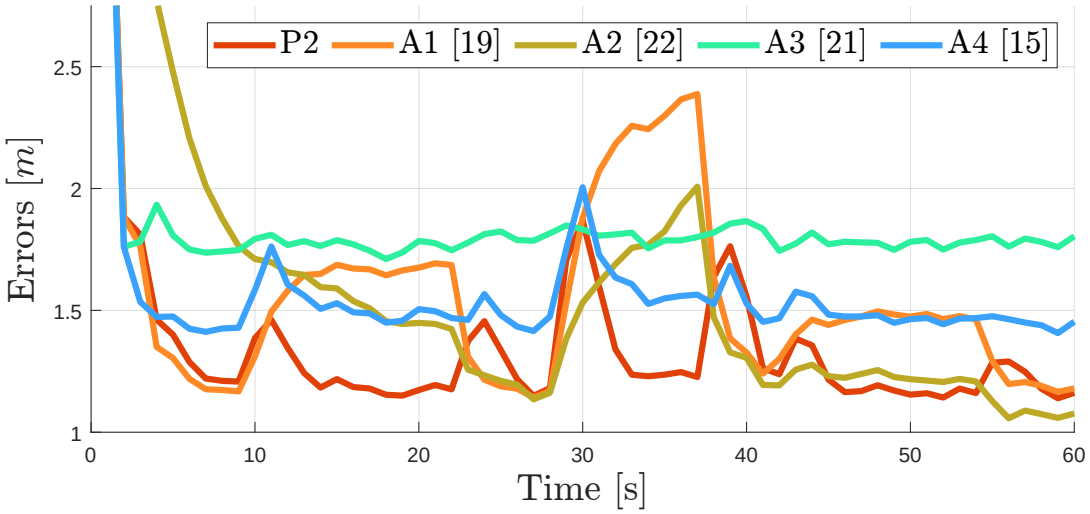


Figure 4.4: Comparison of GW distances of P2, A1, A2, A3, A4 to the ground truth for the trajectory given in Figure 4.3 averaged over 10^3 MC runs.

Table 4.3: Time-averaged GW (m) distances under different sensor noise levels

$I_{max} \setminus R$	$0.01 \times \mathbb{I}_2$	$1 \times \mathbb{I}_2$	$5 \times \mathbb{I}_2$	$10 \times \mathbb{I}_2$	$20 \times \mathbb{I}_2$	$30 \times \mathbb{I}_2$	$50 \times \mathbb{I}_2$
10	0.412	0.885	1.601	2.493	6.227	12.42	21.88
25	0.412	0.885	1.528	1.944	3.038	4.723	9.639
50	0.412	0.885	1.514	1.884	2.796	3.787	7.742

Table 4.4: Time-averaged $RMSE_\theta$ (degrees) under different sensor noise levels

$I_{max} \setminus R$	$0.01 \times \mathbb{I}_2$	$1 \times \mathbb{I}_2$	$5 \times \mathbb{I}_2$	$10 \times \mathbb{I}_2$	$20 \times \mathbb{I}_2$	$30 \times \mathbb{I}_2$	$50 \times \mathbb{I}_2$
10	2.8	4.5	7.8	12.9	26.8	73.2	92.6
25	2.8	4.3	7.0	9.7	14.8	19.8	46.8
50	2.8	4.3	6.8	9.0	14.3	17.1	34.3

of the extent, the orientation performance of P1 becomes unsatisfactory. Such a case is illustrated in Figure 4.5, where the measurements carry much less information regarding the target's orientation due to high noise covariance. Further increases in noise levels would drift the simulations away from realistic sensor models commonly encountered in ETT applications, such as automotive radar and lidar sensors.

Increasing the number of variational iterations proved effective in maintaining the performance up to a certain noise level. This is illustrated in Figure 4.6, where it can be observed that allowing the algorithm to perform 50 iterations provided better orientation angle tracking performance.

The drift model exhibits similar behavior. However, unlike the trajectory-aligned model, the drift model cannot extract additional information from the trajectory. As a result, performance degradation on the orientation angle tracking begins at lower noise levels. Again, increasing the number of iterations can help mitigate this problem.

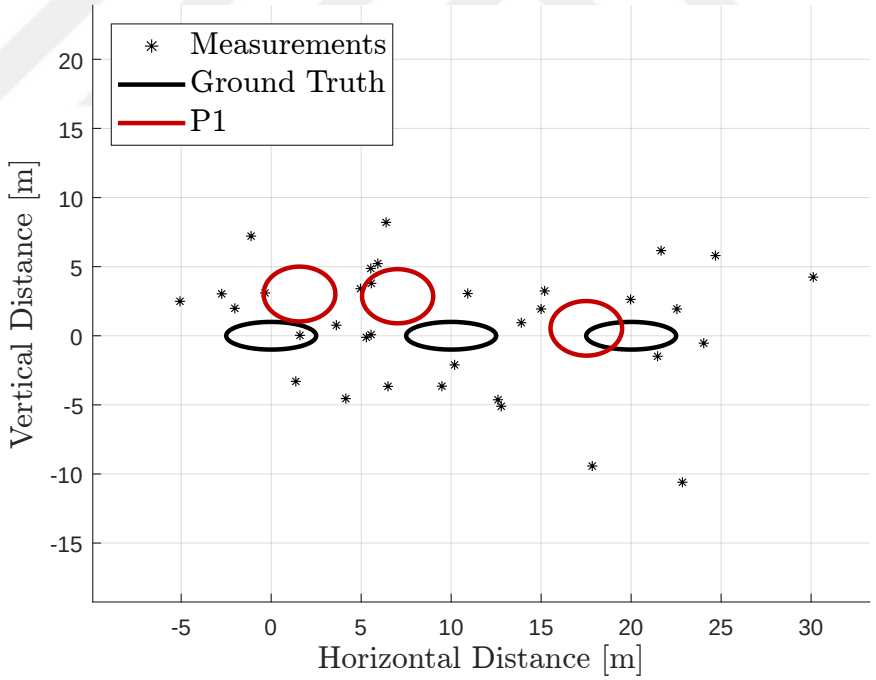


Figure 4.5: The measurement realizations for the first three frames of the high measurement noise covariance scenario, where $R = 20 \times \mathbb{I}_2$.

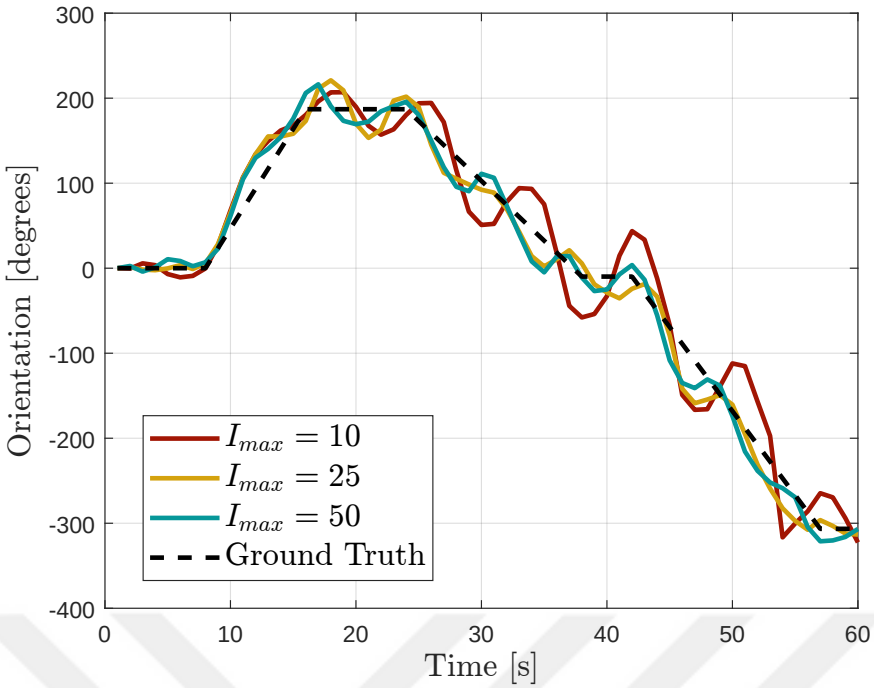


Figure 4.6: Comparison of orientation angle estimations of P1 to the ground truth orientation for different numbers of variational iterations.

4.3 Real Data Experiments

This section presents two real data experiments involving an automotive radar dataset [54] and a video tracking example [55]. We report the time-averaged GW and RMSE values of these experiments in Table 4.5 and 4.6 respectively.

4.3.1 Automotive Radar Data Experiment

This experiment considers automotive radar data and ground truth bounding boxes of a vehicle in the nuScenes dataset [54]. The vehicle takes a 90° turn at a junction in 33 time frames with 0.5 seconds sampling time. P1 (Trajectory-Aligned Model), A1 [19], A2 [22], A3 [21], and A5 [27] are compared in this real-data experiment. Parameters for this experiment are the same as in Section 4.2.1 except for the process noise covariance matrices. Process noise for P1 is selected as $\boldsymbol{Q} = \text{diag}(5 \times 10^{-2}, 10^{-2})$. Process noise covariance for A1 is chosen as $\boldsymbol{Q} = \text{blkdiag}(5 \times 10^{-2} \times \boldsymbol{Q}_{cv} \otimes \mathbb{I}_2, 0.5)$. Process noise covariance matrices of A2 are

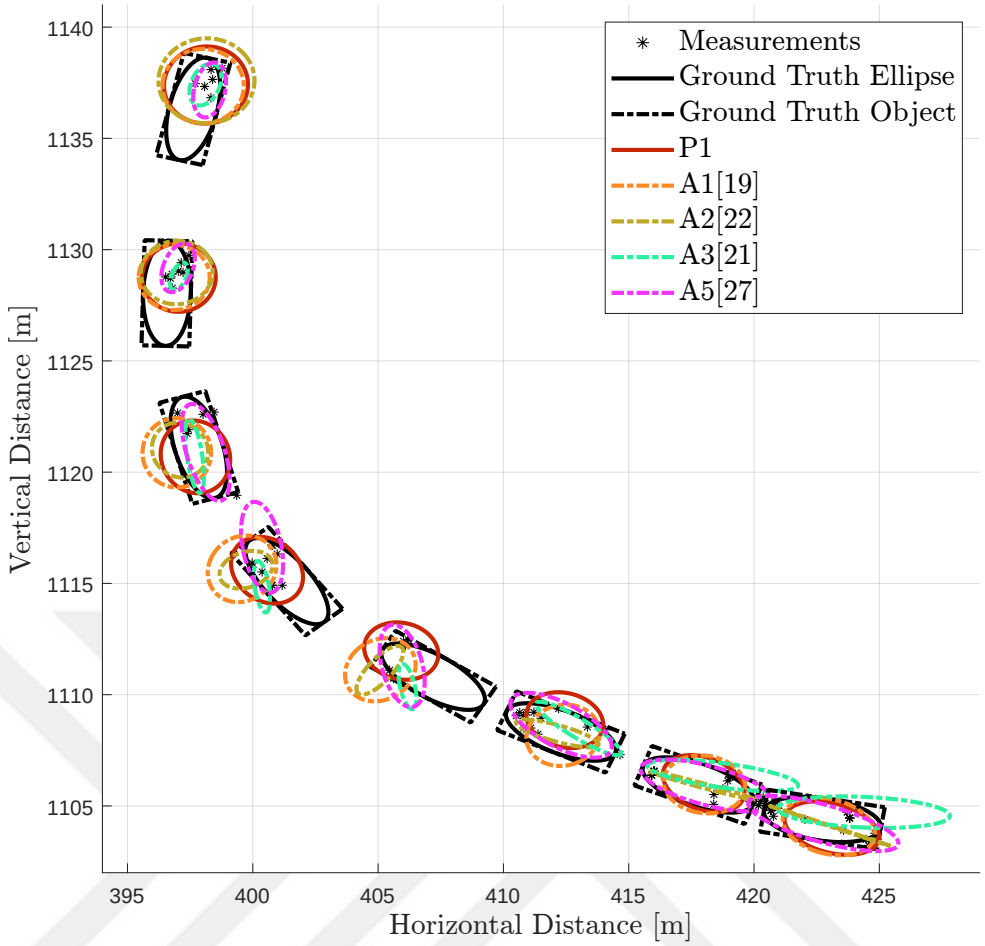


Figure 4.7: Eight frames of algorithm results for the Automotive Radar Experiment. The dashed arrow indicates the direction of motion.

$\boldsymbol{C}_r^\omega = 5 \times 10^{-2} \times \boldsymbol{Q}_{cv} \otimes \mathbb{I}_2$ and $\boldsymbol{C}_p^\omega = \text{diag}(0.25, 10^{-3}, 10^{-3})$ for kinematics and shape vectors, respectively. For A3, $\boldsymbol{Q} = \text{blkdiag}(5 \times 10^{-4} \times \mathbb{I}_2, 10^{-3})$ are chosen as the process noise covariance of the state dynamics. Finally parameters for A5 following the notation in [27] are set to $\delta_Q = 10^{-6}$, $j_{max} = 10$, $\gamma_k = 0.05$, $\boldsymbol{Q}_c = 5 \times \boldsymbol{Q}_{cv} \otimes \mathbb{I}_2$ and $\boldsymbol{Q}_s = \text{diag}(0.1^2, 0.1^2)$.

Figure 4.7 depicts the resulting algorithm estimations with 8 equally spaced frames for visual clarity and readability. P1 exhibits a comparatively high accuracy in estimating the orientation, leading to its superior tracking performance. This can be attributed to the aligned model's better utilization of the correlation between the extent orientation and the trajectory of the maneuvering vehicle.

Table 4.5: Time-averaged errors for the Automotive Radar Experiment

	P1	A1 [19]	A2 [22]	A3 [21]	A5 [27]
GW_{av} (m)	**1.56**	1.58	2.09	2.38	1.66
$RMSE^{\theta}_{av}$ (degrees)	**6.8**	103.2	99.7	-	18.2

4.3.2 Drifting Dinghy Experiment

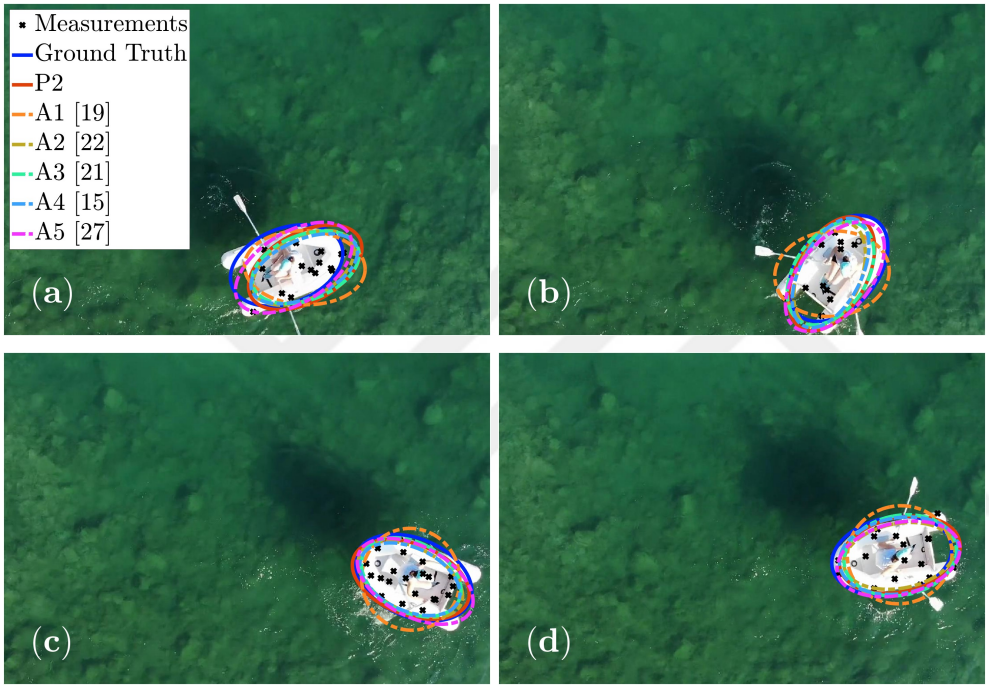


Figure 4.8: Four sample frames of algorithm results for the drifting dinghy experiment. Frames 24, 29, 41, and 45 are shown in the (a), (b), (c), and (d) order.

In this experiment, we consider the aerial view video data of a drifting dinghy [55]. The dinghy takes about a 180° turn in 50 frames with a sampling time of 2.0 seconds. We generated measurements by taking random samples from the white pixels of each frame where the number of measurements generated from the target is set as a Poisson distributed random variable with rate $\lambda = 15$. Ground truth ellipses and orientation angles of this scenario are extracted manually. P2 (Drift Model), A1 [19], A2 [22], A3 [21], A4 [15] and A5 [27] are compared in this experiment. Parameters for this experiment are the same as in Section 4.2.2 except for the process noise covariance

Table 4.6: Time-averaged errors for the Drifting Dinghy Experiment

	P2	A1 [19]	A2 [22]	A3 [21]	A4 [15]	A5 [27]
GW_{av} (m)	**0.97**	1.52	2.1	1.21	1.68	1.03
$RMSE^{\theta}_{av}$ (degrees)	**9.6**	87.6	14.2	-	-	10.8

matrices. Process noise covariances for P2 are selected as $\boldsymbol{Q_x} = 2.5 \times \boldsymbol{Q}_{cv} \otimes \mathbb{I}_2$ and $\boldsymbol{Q_\theta} = 10^{-3} \times \boldsymbol{Q}_{cv}$. Process noise covariance for A1 is chosen as $\boldsymbol{Q} = \text{blkdiag}(0.25 \times \boldsymbol{Q}_{cv} \otimes \mathbb{I}_2, 10^{-3})$. Process noise covariance matrices of A2 are $\boldsymbol{C}^{\omega}_{r} = 0.25 \times \boldsymbol{Q}_{cv} \otimes \mathbb{I}_2$ and $\boldsymbol{C}^{\omega}_{p} = \text{diag}(10^{-3}, 10^{-2}, 10^{-2})$ for kinematics and shape vectors, respectively. $\boldsymbol{Q} = \text{blkdiag}(0.25 \times \boldsymbol{Q}_{cv} \otimes \mathbb{I}_2, 10^{-3})$, for A3 $\boldsymbol{Q} = \text{blkdiag}(0.25 \times \boldsymbol{Q}_{cv} \otimes \mathbb{I}_2, 10^{-3})$ for A4, $\boldsymbol{Q} = 0.25 \times \boldsymbol{Q}_{cv} \otimes \mathbb{I}_2$ is chosen as the process noise covariance of the state dynamics. Finally parameters for A5 are set to $\delta_Q = 10^{-6}$, $j_{max} = 10$, $\gamma_k = 0.05$, $\boldsymbol{Q}_c = 0.05 \times \boldsymbol{Q}_{cv} \otimes \mathbb{I}_2$ and $\boldsymbol{Q}_s = \text{diag}(0.05, 0.05)$.

Estimates of the algorithms are depicted in Figure 4.8. We show 4 frames out of 50 in the figure for visual clarity. It can be seen that P2 represents the drifting behavior of the dinghy relatively well while capturing both orientation and trajectory information simultaneously.

CHAPTER 5

CONCLUSION AND FUTURE WORK

In this thesis, we have successfully developed and implemented novel random matrix-based methodologies for elliptical extended target tracking, specifically addressing the unique motion patterns of trajectory-aligned and drifting targets. Our approach, which distinguishes between the correlation of kinematic states and extent orientation, provides a critical improvement over traditional tracking methods that often neglect these distinctions.

Extended target tracking (ETT) has seen substantial advancements in recent years, primarily focusing on improving the accuracy and efficiency of tracking systems in diverse applications. Traditional approaches often assume simple, static models for target extent and kinematics, which can lead to significant limitations, especially in dynamic environments. Prior research has explored various methods, and while they are robust in many scenarios, they frequently overlook the crucial dynamics between the target's kinematic state and its orientation. Our research builds upon this foundation by introducing novel random matrix-based models that explicitly account for the correlation between kinematic states and extent orientation. This differentiation is crucial for dealing with real-world tracking scenarios where targets such as ground vehicles and UAVs exhibit complex motion patterns. By addressing these previously neglected dynamics, our models enhance the predictive capabilities of ETT systems, offering a significant contribution to the literature and extending the applicability of tracking technologies in practical settings.

For trajectory-aligned targets, such as cars, trains, and bicycles, our model leverages the strong correlation between kinematic state and extent orientation, resulting in significantly enhanced tracking accuracy. This model is particularly effective when tar-

gets exhibit predictable, aligned movement patterns. Conversely, our drifting model is optimized for maritime vessels and UAVs, where kinematic state and orientation are less correlated. By adapting to the independent nature of these parameters, this model offers improved accuracy for tracking targets with more unpredictable and autonomous movements.

Through rigorous testing on both simulated and real-world datasets, our models demonstrated superior performance compared to existing state-of-the-art methods. The experiments highlighted our models' ability to achieve lower estimation errors in orientation and extent states, particularly in scenarios characterized by high maneuverability and substantial environmental noise. These findings underscore the importance of our novel approach in enhancing the capabilities of extended target tracking systems, paving the way for more accurate and reliable tracking in complex scenarios.

This thesis confirms the efficacy of distinguishing between different types of target motions in tracking applications. It enhances the overall understanding and development of tracking technologies in automotive radars and maritime surveillance systems.

5.1 Future Work

Looking forward, several enhancements can be made to refine the RM-based ETT models introduced in this thesis further. First, the variational Bayesian (VB) iteration steps could be adapted based on convergence rates to optimize the computational efficiency and accuracy of the models. Implementing a dynamic adjustment mechanism for VB iterations while setting a maximum iteration limit would allow the algorithms to adapt to different scenarios and potentially lead to faster and more accurate convergence.

Additionally, extending the measurement model to include other types of measurements, such as Doppler and polar measurements, could significantly enhance the applicability and versatility of the proposed methods. Incorporating these measurement sources would allow the models to be applied in a broader range of environments and increase their utility in practical scenarios.

Moreover, developing a 3D extension of the proposed models would address the needs of applications where targets occupy three-dimensional space, such as aerial tracking or space surveillance.

Also, extending the models to handle multi-target scenarios would be a significant advancement. This extension would involve adapting the existing algorithms to track multiple distinct targets simultaneously, improving the models' capability in complex environments with numerous targets.

For scenarios where targets exhibit combined behaviors, such as a vehicle displaying both trajectory-aligned and drifting behaviors, a multimodel approach, such as the Interacting Multiple Model (IMM) algorithm, could be employed. This approach would allow for switching between different motion models, making the system adaptable to varying target dynamics.

Lastly, creating multiellipsoidal versions of the proposed models would enable tracking multiple overlapping targets more effectively. This development would be particularly beneficial in group tracking scenarios or objects with shapes better represented by unions of multiple elliptical regions.

REFERENCES

[1] Y. Bar-Shalom, X.-R. Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation: Theory Algorithms and Software*. John Wiley & Sons, Inc, 2001.

[2] E. Wan and R. Van Der Merwe, "The unscented Kalman filter for nonlinear estimation," in *Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium (Cat. No.00EX373)*, 2000, pp. 153–158.

[3] S. Blackman and R. Popoli, *Design and Analysis of Modern Tracking Systems*. Norwood, MA: Artech House, 1999.

[4] A. Doucet, N. de Freitas, and N. Gordon, "Sequential Monte Carlo methods in practice," *Statistics for Engineering and Information Science*, 2001.

[5] C. Andrieu, A. Doucet, and R. Holenstein, "Particle Markov chain Monte Carlo methods," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 72, no. 3, pp. 269–342, 2010.

[6] L. Mihaylova, A. Y. Carmi, F. Septier, A. Gning, S. K. Pang, and S. Godsill, "Overview of Bayesian sequential Monte Carlo methods for group and extended object tracking," *Digital Signal Processing*, vol. 25, pp. 1–16, 2014.

[7] K. Granstrom, M. Baum, and S. Reuter, "Extended object tracking: Introduction, overview, and applications," *Journal of Advances in Information Fusion (JAIF)*, vol. 12, no. 2, pp. 139–174, Dec 2017.

[8] K. Granström and M. Baum, "A tutorial on multiple extended object tracking," Feb. 2022. [Online]. Available: http://dx.doi.org/10.36227/techrxiv.19115858.v1

[9] J. W. Koch, "Bayesian approach to extended object and cluster tracking using

random matrices," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 44, no. 3, pp. 1042–1059, July 2008.

[10] M. Feldmann, D. Franken, and W. Koch, "Tracking of extended objects and group targets using random matrices," *IEEE Transactions on Signal Processing*, vol. 59, no. 4, pp. 1409–1420, April 2011.

[11] U. Orguner, "A variational measurement update for extended target tracking with random matrices," *IEEE Transactions on Signal Processing*, vol. 60, no. 7, pp. 3827–3834, July 2012.

[12] K. Granström and U. Orguner, "New prediction for extended targets with random matrices," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 50, no. 2, pp. 1577–1589, 2014.

[13] J. Lan and X. R. Li, "Tracking of maneuvering non-ellipsoidal extended object or target group using random matrix," *IEEE Transactions on Signal Processing*, vol. 62, no. 9, pp. 2450–2463, 2014.

[14] J. Lan and X. R. Li, "Tracking of extended object or target group using random matrix — Part I: New model and approach," in *2012 15th International Conference on Information Fusion*, 2012, pp. 2177–2184.

[15] N. J. Bartlett, C. Renton, and A. G. Wills, "A closed-form prediction update for extended target tracking using random matrices," *IEEE Transactions on Signal Processing*, vol. 68, pp. 2404–2418, 2020.

[16] K. Granström, P. Willett, and Y. Bar-Shalom, "An extended target tracking model with multiple random matrices and unified kinematics," in *2015 18th International Conference on Information Fusion (Fusion)*, 2015, pp. 1007–1014.

[17] S. F. Kara and E. Özkan, "Multi-ellipsoidal extended target tracking using sequential Monte Carlo," in *2018 21st International Conference on Information Fusion (FUSION)*, 2018, pp. 1–8.

[18] Q. Hu, H. Ji, and Y. Zhang, "Tracking of maneuvering non-ellipsoidal extended target with varying number of sub-objects," *Mechanical Systems and Signal Processing*, vol. 99, pp. 262–284, 2018.

[19] B. Tuncer and E. Özkan, "Random matrix based extended target tracking with orientation: A new model and inference," *IEEE Transactions on Signal Processing*, vol. 69, pp. 1910–1923, 2021.

[20] B. Tuncer, U. Orguner, and E. Özkan, "Multi-ellipsoidal extended target tracking with variational Bayes inference," *IEEE Transactions on Signal Processing*, vol. 70, pp. 3921–3934, 2022.

[21] N. J. Bartlett and A. G. Wills, "A robust random matrix prediction model for extended object rotations," *Proceedings of 2021 IEEE 24th International Conference on Information Fusion, FUSION 2021*, 2021.

[22] S. Yang and M. Baum, "Tracking the orientation and axes lengths of an elliptical extended object," *IEEE Transactions on Signal Processing*, vol. 67, no. 18, pp. 4720–4729, 2019.

[23] J. Lan and X. R. Li, "Extended-object or group-target tracking using random matrix with nonlinear measurements," *IEEE Transactions on Signal Processing*, vol. 67, no. 19, pp. 5130–5142, 2019.

[24] Z. Li, J. Zhang, J. Wang, and Q. Zhou, "Recursive noise adaptive extended object tracking by variational Bayesian approximation," *IEEE Access*, vol. 7, pp. 151 168–151 179, 2019.

[25] Z. Li, J. Zhang, J. Wang, and Q. Zhou, "Extended object tracking using random matrix with converted measurements," *IET Radar, Sonar & Navigation*, vol. 14, no. 7, pp. 981–991, 2020.

[26] S. Liu, Y. Liang, L. Xu, T. Li, and X. Hao, "EM-based extended object tracking without a priori extension evolution model," *Signal Processing*, vol. 188, Nov. 2021.

[27] M. Li, J. Lan, and X. R. Li, "Tracking of elliptical object with unknown but fixed lengths of axes," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 59, no. 5, pp. 6518–6533, 2023.

[28] K. Granström, P. Willett, and Y. Bar-Shalom, "An extended target tracking model with multiple random matrices and unified kinematics," in *2015 18th International Conference on Information Fusion (Fusion)*, 2015, pp. 1007–1014.

[29] K. Granström, C. Lundquist, and U. Orguner, "Tracking rectangular and elliptical extended targets using laser measurements," in *14th International Conference on Information Fusion*, 2011, pp. 1–8.

[30] K. Granström, S. Reuter, D. Meissner, and A. Scheel, "A multiple model PHD approach to tracking of cars under an assumed rectangular shape," in *17th International Conference on Information Fusion (FUSION)*, 2014, pp. 1–8.

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

[32] M. Baum and U. D. Hanebeck, "Shape tracking of extended objects and group targets with star-convex RHMs," in *14th International Conference on Information Fusion*, 2011, pp. 1–8.

[33] N. Wahlström and E. Özkan, "Extended target tracking using Gaussian processes," *IEEE Transactions on Signal Processing*, vol. 63, no. 16, pp. 4165–4178, Aug 2015.

[34] E. Özkan, N. Wahlström, and S. J. Godsill, "Rao-Blackwellised particle filter for star-convex extended target tracking models," in *2016 19th International Conference on Information Fusion (FUSION)*, 2016, pp. 1193–1199.

[35] M. Kumru and E. Özkan, "3D extended object tracking using recursive Gaussian processes," in *2018 21st International Conference on Information Fusion (FUSION)*, 2018, pp. 1–8.

[36] J. L. Yang, P. Li, and H. W. Ge, "Extended target shape estimation by fitting B-spline curve," *Journal of Applied Mathematics*, vol. 2014, 2014.

[37] H. Kaulbersch, J. Honer, and M. Baum, "A cartesian B-spline vehicle model for extended object tracking," *2018 21st International Conference on Information Fusion, FUSION 2018*, pp. 1771–1778, 9 2018.

[38] K.-M. Dahlén, C. Lindberg, M. Yoneda, and T. Ogawa, "An improved B-spline extended object tracking model using the iterative closest point method," in *2022 25th International Conference on Information Fusion (FUSION)*, 2022, pp. 1–8.

[39] D. H. Nguyen, J. H. Kay, B. J. Orchard, and R. H. Whiting, "Classification and tracking of moving ground vehicles," *Lincoln Laboratory Journal*, vol. 13, no. 2, pp. 275–308, 2002.

[40] K. Granström, S. Renter, M. Fatemi, and L. Svensson, "Pedestrian tracking using Velodyne data — Stochastic optimization for extended object tracking," in *2017 IEEE Intelligent Vehicles Symposium (IV)*, 2017, pp. 39–46.

[41] S. Scheidegger, J. Benjaminsson, E. Rosenberg, A. Krishnan, and K. Granström, "Mono-camera 3D multi-object tracking using deep learning detections and PMBM filtering," in *2018 IEEE Intelligent Vehicles Symposium (IV)*, 2018, pp. 433–440.

[42] J. S. Fowdur, M. Baum, and F. Heymann, "Tracking targets with known spatial extent using experimental marine radar data," *FUSION 2019 - 22nd International Conference on Information Fusion*, 7 2019.

[43] P. Christiansen, K. A. Steen, R. N. Jørgensen, and H. Karstoft, "Automated detection and recognition of wildlife using thermal cameras," *Sensors*, vol. 14, no. 8, pp. 13 778–13 793, 2014.

[44] S. Särkkä, *Bayesian Filtering and Smoothing*. Cambridge Univ. Press, 2013.

[45] F. Gustafsson, *Statistical Sensor Fusion*. Studentlitteratur, 2018.

[46] C. M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2006.

[47] I. M. Gelfand and S. V. Fomin, *Calculus of Variations*. Prentice-Hall, 1963.

[48] G. Strang, *Linear Algebra and its Applications*. Belmont, CA: Thomson, Brooks/Cole, 2006.

[49] R. G. Gallager, *Stochastic Processes: Theory for Applications*. Cambridge University Press, 2017.

[50] K. B. Petersen and M. S. Pedersen, "The matrix cookbook," Oct. 2008, version 20081110. [Online]. Available: http://www2.imm.dtu.dk/pubdb/p.php?3274

[51] E. Özkan, V. Šmídl, S. Saha, C. Lundquist, and F. Gustafsson, "Marginalized adaptive particle filtering for nonlinear models with unknown time-varying noise parameters," *Automatica*, vol. 49, no. 6, pp. 1566–1575, 2013.

[52] S. Yang, M. Baum, and K. Granström, "Metrics for performance evaluation of elliptic extended object tracking methods," in *2016 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*, 2016, pp. 523–528.

[53] M. Roth, G. Hendeby, and F. Gustafsson, "EKF/UKF maneuvering target tracking using coordinated turn models with polar/Cartesian velocity," in *17th International Conference on Information Fusion (FUSION)*, 2014, pp. 1–8.

[54] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom, "nuScenes: A multimodal dataset for autonomous driving," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[55] M. Kumru, H. Köksal, and E. Özkan, "Variational measurement update for extended object tracking using Gaussian processes," *IEEE Signal Processing Letters*, vol. 28, pp. 538–542, 2021.

APPENDICES

A Definition of Functional Derivatives

Given a functional $\mathcal{F}[f]$ that depends on a function $f(x)$, the functional derivative of $\mathcal{F}[f]$ with respect to $f(x)$ is defined such that:

$$\delta\mathcal{F}[f] = \int \frac{\delta\mathcal{F}[f]}{\delta f(x)}\delta f(x)\, dx. \tag{A.1}$$

Here, $\dfrac{\delta\mathcal{F}[f]}{\delta f(x)}$ is the functional derivative of $\mathcal{F}$ with respect to $f(x)$ [47, Chapter 1]. This expression indicates how small changes in the function $f(x)$ affect the value of the functional $\mathcal{F}[f]$.

For small ϵ and $\delta f(x)$;

$$\begin{aligned}
\delta\mathcal{F}[f] =& \lim_{\epsilon\to 0}\frac{1}{\epsilon}\left(\mathcal{F}[f+\epsilon\delta f]-\mathcal{F}[f]\right), && \text{(A.2)}\\
=& \lim_{\epsilon\to 0}\frac{1}{\epsilon}\left(\mathcal{F}[f+\epsilon\delta f]+\epsilon\frac{\partial}{\partial\epsilon}\mathcal{F}\left[f+\epsilon\delta f\right]\Big|_{\epsilon=0}+\mathcal{O}(\epsilon^2)\right)-\mathcal{F}[f], && \text{(A.3)}\\
=& \left(\frac{\partial}{\partial\epsilon}\mathcal{F}\left[f+\epsilon\delta f\right]\right)_{\epsilon=0}. && \text{(A.4)}
\end{aligned}$$

So a way to compute the derivative of a functional is given as;

$$\left(\frac{\partial}{\partial\epsilon}\mathcal{F}\left[f+\epsilon\delta f\right]\right)_{\epsilon=0} = \int \frac{\delta\mathcal{F}[f]}{\delta f(x)}\delta f(x)\ \mathrm{d}x. \tag{A.5}$$

B Coordinated Turn with Polar Velocity Model

In our experiments with the Aligned Model, we employ the discretized state space equations presented in [53] to propagate the kinematic state to the subsequent time step. These equations are given as

$$\boldsymbol{x}_{k+1} = f(\boldsymbol{x}_k) + \boldsymbol{\nu}_k, \quad \boldsymbol{\nu}_k \sim \mathcal{N}(\mathbf{0}, \boldsymbol{Q}_{\boldsymbol{x}}). \tag{B.1}$$

Here, the state vector $\boldsymbol{x}_k = [p_k^x \; p_k^y \; v_k \; \varphi_k \; \dot{\varphi}_k]^T$ encompasses the Cartesian positions, magnitude of the Cartesian velocity, heading angle, and turn rate. The state transition function $f(.)$ is expressed explicitly as,

$$f(\boldsymbol{x}_{k+1}) = \begin{bmatrix} p_k^x + \frac{2v_k}{\dot{\varphi}_k} sin(\frac{\dot{\varphi}_k T}{2}) \cos(\varphi_k + \frac{\dot{\varphi}_k T}{2}) \\ p_k^y + \frac{2v_k}{\dot{\varphi}_k} sin(\frac{\dot{\varphi}_k T}{2}) \sin(\varphi_k + \frac{\dot{\varphi}_k T}{2}) \\ v_k \\ \varphi_k + \dot{\varphi}_k T \\ \dot{\varphi}_k \end{bmatrix}. \tag{B.2}$$

Here, T is the sampling time. The process noise $\boldsymbol{\nu}_k$, is modeled with a state dependent function $g(\boldsymbol{x})$,

$$\boldsymbol{\nu}_k = g(\boldsymbol{x}_k)\boldsymbol{w}_k, \quad \boldsymbol{w}_k \overset{i.i.d}{\sim} \mathcal{N}(\boldsymbol{0}, \boldsymbol{Q}), \quad \boldsymbol{Q} \in \mathbb{S}_{++}^2, \tag{B.3}$$

$$g(\boldsymbol{x}_k) = \begin{bmatrix} \frac{T^2}{2}\cos(\varphi_k) & \frac{T^2}{2}\sin(\varphi_k) & T & 0 & 0 \\ 0 & 0 & 0 & \frac{T^2}{2} & T \end{bmatrix}^T. \tag{B.4}$$

C Proof of Lemma 1

Proof. Given $\varphi \sim \mathcal{N}(\hat{\varphi}, P_\varphi)$, the characteristic function of φ is [49]:

$$\phi_\varphi(t) = \mathbb{E}_\varphi\left[e^{jt\varphi}\right] = \exp\left(j\hat{\varphi}t - \frac{1}{2}P_\varphi t^2\right) \tag{C.1}$$

Expectation of $\sin(2\varphi)$

$$\mathbb{E}_\varphi\left[\sin(2\varphi)\right] = \Im\left(\phi_\varphi(2)\right) \tag{C.2}$$

$$\phi_\varphi(2) = \exp\left(j2\hat{\varphi} - 2P_\varphi\right) \tag{C.3}$$

$$\mathbb{E}_\varphi\left[\sin(2\varphi)\right] = \sin(2\hat{\varphi})\exp(-2P_\varphi) \tag{C.4}$$

Expectation of $\cos(2\varphi)$

$$\mathbb{E}_\varphi\left[\cos(2\varphi)\right] = \Re\left(\phi_\varphi(2)\right) \tag{C.5}$$

$$\phi_\varphi(2) = \exp\left(j2\hat{\varphi} - 2P_\varphi\right) \tag{C.6}$$

$$\mathbb{E}_\varphi\left[\cos(2\varphi)\right] = \cos(2\hat{\varphi})\exp(-2P_\varphi) \tag{C.7}$$

Expectation of $\varphi \sin(2\varphi)$

$$\phi'_{\varphi}(t) = \frac{\mathrm{d}}{\mathrm{d}t}\left(\exp\left(j\hat{\varphi}t - \frac{1}{2}P_{\varphi}t^2\right)\right) \tag{C.8}$$

$$= \exp\left(j\hat{\varphi}t - \frac{1}{2}P_{\varphi}t^2\right)(j\hat{\varphi} - P_{\varphi}t) \tag{C.9}$$

$$\phi'_{\varphi}(2) = \exp\left(j2\hat{\varphi} - 2P_{\varphi}\right)(j\hat{\varphi} - 2P_{\varphi}) \tag{C.10}$$

$$\mathbb{E}_{\varphi}\left[\varphi \sin(2\varphi)\right] = \Im\left(j\phi'_{\varphi}(2)\right) \tag{C.11}$$

$$= \Im\left(\exp\left(j2\hat{\varphi} - 2P_{\varphi}\right)(\hat{\varphi} - 2jP_{\varphi})\right) \tag{C.12}$$

$$= (\hat{\varphi}\sin(2\hat{\varphi}) + 2P_{\varphi}\cos(2\hat{\varphi}))\exp(-2P_{\varphi}) \tag{C.13}$$

Expectation of $\varphi \cos(2\varphi)$

$$\mathbb{E}_{\varphi}\left[\varphi \cos(2\varphi)\right] = \Re\left(j\phi'_{\varphi}(2)\right) \tag{C.14}$$

$$= \Re\left(\exp\left(j2\hat{\varphi} - 2P_{\varphi}\right)(\hat{\varphi} - 2jP_{\varphi})\right) \tag{C.15}$$

$$= (\hat{\varphi}\cos(2\hat{\varphi}) - 2P_{\varphi}\sin(2\hat{\varphi}))\exp(-2P_{\varphi}) \tag{C.16}$$

Expectation of $\varphi^2 \sin(2\varphi)$

$$\phi''_{\varphi}(t) = \frac{\mathrm{d}^2}{\mathrm{d}t^2}\left(\exp\left(j\hat{\varphi}t - \frac{1}{2}P_{\varphi}t^2\right)\right) \tag{C.17}$$

$$= \exp\left(j\hat{\varphi}t - \frac{1}{2}P_{\varphi}t^2\right)\left((j\hat{\varphi} - P_{\varphi}t)^2 - P_{\varphi}\right) \tag{C.18}$$

$$\phi''_{\varphi}(2) = \exp\left(j2\hat{\varphi} - 2P_{\varphi}\right)\left((j\hat{\varphi} - 2P_{\varphi})^2 - P_{\varphi}\right) \tag{C.19}$$

$$\mathbb{E}_{\varphi}\left[\varphi^2 \sin(2\varphi)\right] = \Im\left(-\phi''_{\varphi}(2)\right) \tag{C.20}$$

$$= \Im\left(-\exp\left(j2\hat{\varphi} - 2P_{\varphi}\right)\left(-\hat{\varphi}^2 - 4j\hat{\varphi}P_{\varphi} - 4P_{\varphi}^2 - P_{\varphi}\right)\right) \tag{C.21}$$

$$= \hat{\varphi}^2\sin(2\hat{\varphi})\exp(-2P_{\varphi}) + 4\hat{\varphi}P_{\varphi}\cos(2\hat{\varphi})\exp(-2P_{\varphi})$$
$$- 4P_{\varphi}^2\sin(2\hat{\varphi})\exp(-2P_{\varphi}) + P_{\varphi}\sin(2\hat{\varphi})\exp(-2P_{\varphi}) \tag{C.22}$$

So,

$$\mathbb{E}_{\varphi}\left[\varphi^2 \sin(2\varphi)\right] = \left((\hat{\varphi}^2 - 4P_{\varphi}^2 + P_{\varphi})\sin(2\hat{\varphi}) + 4\hat{\varphi}P_{\varphi}\cos(2\hat{\varphi})\right)\exp(-2P_{\varphi}) \tag{C.23}$$

Expectation of $\varphi^2 \cos(2\varphi)$

$$\mathbb{E}_{\varphi}\left[\varphi^2 \cos(2\varphi)\right] = \Re\left(-\phi''_{\varphi}(2)\right) \tag{C.24}$$

$$= \Re\left(-\exp\left(j2\hat{\varphi} - 2P_\varphi\right)\left(-\hat{\varphi}^2 - 4j\hat{\varphi}P_\varphi - 4P_\varphi^2 - P_\varphi\right)\right) \quad \text{(C.25)}$$

$$= \hat{\varphi}^2\cos(2\hat{\varphi})\exp(-2P_\varphi) - 4\hat{\varphi}P_\varphi\sin(2\hat{\varphi})\exp(-2P_\varphi)$$
$$- 4P_\varphi^2\cos(2\hat{\varphi})\exp(-2P_\varphi) + P_\varphi\cos(2\hat{\varphi})\exp(-2P_\varphi) \quad \text{(C.26)}$$

So,

$$\mathbb{E}_\varphi\left[\varphi^2\cos(2\varphi)\right] = \left((\hat{\varphi}^2 - 4P_\varphi^2 + P_\varphi)\cos(2\hat{\varphi}) - 4\hat{\varphi}P_\varphi\sin(2\hat{\varphi})\right)\exp(-2P_\varphi) \quad \text{(C.27)}$$

We define the vector $\boldsymbol{t}_\varphi \triangleq \begin{bmatrix} 1+\cos(2\varphi) \\ 1-\cos(2\varphi) \\ \sin(2\varphi) \end{bmatrix}$.

The expectations $\mathbb{E}\left[\boldsymbol{t}_\varphi\right]$, $\mathbb{E}\left[\varphi\boldsymbol{t}_\varphi\right]$, and $\mathbb{E}\left[\varphi^2\boldsymbol{t}_\varphi\right]$ can be analytically calculated as follows:

Expectation of $\boldsymbol{t}_\varphi$:

$$\mathbb{E}\left[\boldsymbol{t}_\varphi\right] = \mathbb{E}\left[\begin{bmatrix} 1+\cos(2\varphi) \\ 1-\cos(2\varphi) \\ \sin(2\varphi) \end{bmatrix}\right] = \begin{bmatrix} 1+\mathbb{E}_\varphi\left[\cos(2\varphi)\right] \\ 1-\mathbb{E}_\varphi\left[\cos(2\varphi)\right] \\ \mathbb{E}_\varphi\left[\sin(2\varphi)\right] \end{bmatrix} = \begin{bmatrix} 1+e^{-2P_\varphi}\cos 2\hat{\varphi} \\ 1-e^{-2P_\varphi}\cos 2\hat{\varphi} \\ e^{-2P_\varphi}\sin 2\hat{\varphi} \end{bmatrix} \quad \text{(C.28)}$$

Expectation of $\varphi\boldsymbol{t}_\varphi$:

$$\mathbb{E}\left[\varphi\boldsymbol{t}_\varphi\right] = \mathbb{E}\left[\varphi\begin{bmatrix} 1+\cos(2\varphi) \\ 1-\cos(2\varphi) \\ \sin(2\varphi) \end{bmatrix}\right] = \begin{bmatrix} \mathbb{E}_\varphi\left[\varphi(1+\cos(2\varphi))\right] \\ \mathbb{E}_\varphi\left[\varphi(1-\cos(2\varphi))\right] \\ \mathbb{E}_\varphi\left[\varphi\sin(2\varphi)\right] \end{bmatrix} \quad \text{(C.29)}$$

$$= \begin{bmatrix} \mathbb{E}_\varphi\left[\varphi\right] + \mathbb{E}_\varphi\left[\varphi\cos(2\varphi)\right] \\ \mathbb{E}_\varphi\left[\varphi\right] - \mathbb{E}_\varphi\left[\varphi\cos(2\varphi)\right] \\ \mathbb{E}_\varphi\left[\varphi\sin(2\varphi)\right] \end{bmatrix} \quad \text{(C.30)}$$

$$= \begin{bmatrix} \hat{\varphi} + \left(\hat{\varphi}\cos(2\hat{\varphi}) - 2P_\varphi\sin(2\hat{\varphi})\right)e^{-2P_\varphi} \\ \hat{\varphi} - \left(\hat{\varphi}\cos(2\hat{\varphi}) - 2P_\varphi\sin(2\hat{\varphi})\right)e^{-2P_\varphi} \\ \left(2P_\varphi\cos(2\hat{\varphi}) + \hat{\varphi}\sin(2\hat{\varphi})\right)e^{-2P_\varphi} \end{bmatrix} \quad \text{(C.31)}$$

Expectation of $\varphi^2 \boldsymbol{t}_\varphi$:

$$\mathbb{E}\left[\varphi^2 \boldsymbol{t}_\varphi\right] = \mathbb{E}\left[\varphi^2 \begin{bmatrix} 1+\cos(2\varphi) \\ 1-\cos(2\varphi) \\ \sin(2\varphi) \end{bmatrix}\right] = \begin{bmatrix} \mathbb{E}_\varphi\left[\varphi^2(1+\cos(2\varphi))\right] \\ \mathbb{E}_\varphi\left[\varphi^2(1-\cos(2\varphi))\right] \\ \mathbb{E}_\varphi\left[\varphi^2\sin(2\varphi)\right] \end{bmatrix} \tag{C.32}$$

$$= \begin{bmatrix} \mathbb{E}_\varphi\left[\varphi^2\right] + \mathbb{E}_\varphi\left[\varphi^2\cos(2\varphi)\right] \\ \mathbb{E}_\varphi\left[\varphi^2\right] - \mathbb{E}_\varphi\left[\varphi^2\cos(2\varphi)\right] \\ \mathbb{E}_\varphi\left[\varphi^2\sin(2\varphi)\right] \end{bmatrix} \tag{C.33}$$

$$= \begin{bmatrix} \hat{\varphi}^2 + P_\varphi + \left((\hat{\varphi}^2 + P_\varphi - 4P_\varphi^2)\cos(2\hat{\varphi}) - 4\hat{\varphi}P_\varphi\sin(2\hat{\varphi})\right)e^{-2P_\varphi} \\ \hat{\varphi}^2 + P_\varphi - \left((\hat{\varphi}^2 + P_\varphi - 4P_\varphi^2)\cos(2\hat{\varphi}) - 4\hat{\varphi}P_\varphi\sin(2\hat{\varphi})\right)e^{-2P_\varphi} \\ \left((\hat{\varphi}^2 + P_\varphi - 4P_\varphi^2)\sin(2\hat{\varphi}) + 4\hat{\varphi}P_\varphi\cos(2\hat{\varphi})\right)e^{-2P_\varphi} \end{bmatrix} \tag{C.34}$$

Thus, the expectations are:

$$\mathbb{E}\left[\boldsymbol{t}_\varphi\right] = \begin{bmatrix} 1+e^{-2P_\varphi}\cos 2\hat{\varphi} \\ 1-e^{-2P_\varphi}\cos 2\hat{\varphi} \\ e^{-2P_\varphi}\sin 2\hat{\varphi} \end{bmatrix}, \tag{C.35a}$$

$$\mathbb{E}\left[\varphi\boldsymbol{t}_\varphi\right] = \begin{bmatrix} \hat{\varphi} + e^{-2P_\varphi}\left(\hat{\varphi}\cos 2\hat{\varphi} - 2P_\varphi\sin 2\hat{\varphi}\right) \\ \hat{\varphi} - e^{-2P_\varphi}\left(\hat{\varphi}\cos 2\hat{\varphi} - 2P_\varphi\sin 2\hat{\varphi}\right) \\ e^{-2P_\varphi}\left(2P_\varphi\cos 2\hat{\varphi} + \hat{\varphi}\sin 2\hat{\varphi}\right) \end{bmatrix}, \tag{C.35b}$$

$$\mathbb{E}\left[\varphi^2\boldsymbol{t}_\varphi\right] = \begin{bmatrix} \mathbb{E}[\varphi^2] + e^{-2P_\varphi}\left((\mathbb{E}[\varphi^2] - 4P_\varphi^2)\cos 2\hat{\varphi} - 4P_\varphi\hat{\varphi}\sin 2\hat{\varphi}\right) \\ \mathbb{E}[\varphi^2] - e^{-2P_\varphi}\left((\mathbb{E}[\varphi^2] - 4P_\varphi^2)\cos 2\hat{\varphi} - 4P_\varphi\hat{\varphi}\sin 2\hat{\varphi}\right) \\ e^{-2P_\varphi}\left((\mathbb{E}[\varphi^2] - 4P_\varphi^2)\sin 2\hat{\varphi} + 4P_\varphi\hat{\varphi}\cos 2\hat{\varphi}\right) \end{bmatrix}, \tag{C.35c}$$

where $\mathbb{E}[\varphi^2] = \hat{\varphi}^2 + P_\varphi$.

□