Sensor Tasking for Satellite Tracking Utilizing Observability Measures

by

MITCHEL THOMAS MCDONALD

Presented to the Faculty of the Graduate School of
The University of Texas at Arlington in Partial Fulfillment
of the Requirements for the Degree of

MASTER OF SCIENCE IN AEROSPACE ENGINEERING

THE UNIVERSITY OF TEXAS AT ARLINGTON

December 2018

To Dad- this would have not happened without you.

ACKNOWLEDGEMENTS

I would like to thank my thesis supervisor, Dr. Kamesh Subbarao, for giving me the opportunity to work on space systems as part of the Aerospace Systems Lab. Thank you for providing me an excellent mentorship while I performed my research.

Thank you to Dr. Animesh Chakravorty and Dr. Shuo Wang for taking time to serve on my committee.

Thank you to my friends and colleagues at the Aerospace Systems Laboratory, Abel, Ameya, Baris, Diganta, Karan, Kashish, Kati, Paul, Prabhjeet, Rahul, Rajnish, and Roopak for their company and assistance.

Thanks (and Gig 'em!) to Dr. Helen Reed of Texas A&M University and the members and alumni of AggieSat Lab.

I would like to acknowledge a memory of Lilly, who was my feline companion from childhood to graduate school. You are greatly missed.

A heartfelt thanks to Britt Plotsky for her love and support while I completed my degree.

Lastly, a most sincere thank you to Mom, Dad, Meridith and the rest of my family for supporting my dreams and making this possible.

November 13, 2018

ABSTRACT

Sensor Tasking for Satellite Tracking Utilizing Observability Measures

MITCHEL THOMAS MCDONALD, M.S

The University of Texas at Arlington, 2018

Supervising Professor: Kamesh Subbarao

In this thesis, novel observability-measure based sensor tasking methods are studied for satellite tracking applications. The tasking is performed by first computing the Hellinger Distance between ground/space based sensors and space objects, and then the measure is utilized for selecting the sensors that maximize observability. Several other measures such as the Fisher Information Gain, Largest Lyapunov Exponent, and Shannon Information Gain have also been utilized and the performance of these measures are computed against each other. The object's state estimates are obtained using nonlinear estimation techniques. The Extended Kalman Filter, Unscented Kalman Filter, and Bootstrap Particle Filter are compared within in this framework. Representative numerical simulations are performed to evaluate the efficacy of the new tasking approach. The proposed tasking approach is also compared with some baseline approaches.

TABLE OF CONTENTS

This thesis was typeset using the AIAA Journal of Guidance, Control, and Dynamics guidelines.

LIST OF ILLUSTRATIONS

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 Background and Motivation

Space object tracking has been of strategic interest to the United States since the launch of Sputnik. Since then, the U.S. Space Surveillance Network has evolved to handle a variety of objectives [1]. This includes working with commercial space operations to mitigate orbital collisions, as well as monitor developments in anti-satellite systems [2],[3].

A recent estimate found that the U.S. Space Surveillance Network tracks up to $20,000$ objects [2]. This is done at a sensor-to-object ratio that is relatively small [1]. Given this situation, an issue may arise when multiple objects may be within view of a sensor, but only a few can be chosen since they contain the most information of the objects' positions in the measurements. This requires 'sensor tasking', a method to prioritize tracked objects in a manner that leads to an enhancement in the quality of the estimate of the space object based on the obtained measurements.

This thesis presents a new sensor tasking method for satellite tracking via a nonlinear estimation algorithm. This is performed utilizing stochastic observability measures. Several measures are identified and compared for their utility to the problem. Certain measures are then applied to space object tracking problem and compared to established methodologies.

1.2 Related Work

This thesis builds the basic framework using the problem definition established by Erwin [4]. In this work the authors use an Extended Kalman Filter (EKF) to estimate the state of the satellite, assuming the sensors know their position and can communicate with other sensors. The specific sensor tasking solution is based on two approaches, i.e. Fisher Information based, and a modified approach that accounts for the size of the covariance estimate. The modified approach uses a pareto variable that trades between a Fisher information gain solution and a pure covariance forecast approach. An additional scaling gain is utilized to normalize the relative sizes of the covariance matrices present in the fusion equation.

Williams [5],[6] builds directly off the developments of Erwin. Ref. [5] identifies the myopic nature of a Fisher Information based approach, and proposes a tasking solution based on Lyapunov Exponents as an alternative. Additionally, an Unscented Kalman Filter (UKF) is introduced due to its robust handling of nonlinear systems [21]. This is compared to the prior EKF implementation. In Ref. [6], Williams introduces Shannon Mutual Information as a measure information gain relative to an object's state. This is described in contrast to Fisher Information, which found to be just an absolute gain in information. Additionally, an AEGIS Kalman Filter is also implemented. This is based on a UKF estimator, with additional features for nonlinear detection and handling. Notably, Ref. [6] demonstrates that tasking using Shannon Mutual Information outperforms Fisher Information and Lyanpunov Exponent tasking in the numerical simulation case presented in Ref. [5].

More recent approaches to sensor tasking applied to Space Situational Awareness (SSA) includes the use of Deep Reinforcement Learning (DRL) that solves the Sensor Management Problem (SMP) by translating the SSA goals to sensors and resource allocation [7]. The primary reason for the choice of the algorithms in Ref. [7]

is to be able to work with large scale space object catalogs. Since large numbers of objects need to be continuously tracked with limited number of sensors, the associated challenge is to optimally manage the resources (Refs. [8–10] which is further aggravated due to the nonlinearity present in the object dynamics as well as the measurements. The use of nonlinear estimation approaches for tracking the objects makes the entire problem computationally very intensive [11]. DRL in this case is proposed as an alternative especially with the use of methods such as the Asynchronous Advantage Actor-Critic (A3C) method [12].

Within the context of the tasking solutions mentioned earlier, it is to be noted that the measures used (such as Shannon Mutual Information) are not true metrics and do not lead to well defined objective functions in an optimization or task prioritization framework. To address this issue, several observability measures are proposed but focus on one that is a bounded metric. Following along the lines as in Ref. [13–15], a Squared Hellinger Distance based tasking gain is computed and compared with the measures mentioned earlier such as the Fisher Information, largest Lyapunov Exponent, and Shannon Mutual Information. The tasking solutions are implemented in a high fidelity simulation with parameters similar to Ref. [5] and Ref. [6].

1.3 Problem Description

This thesis seeks to develop a estimator/tasking method with the lowest possible position error for each space object. To enable analysis of these methods, models of spacecraft/sensor dynamics and sensor detection are employed from Ref. [4]. These have been simplified to 2D motion. This is considered reasonable, since the implementation of a 3D model is trivial and does not contribute significantly to the analysis of estimator/tasking methods [4],[5]. The models are implemented in a software sim-

ulation, which generates the needed information for an estimator/tasking method to run.

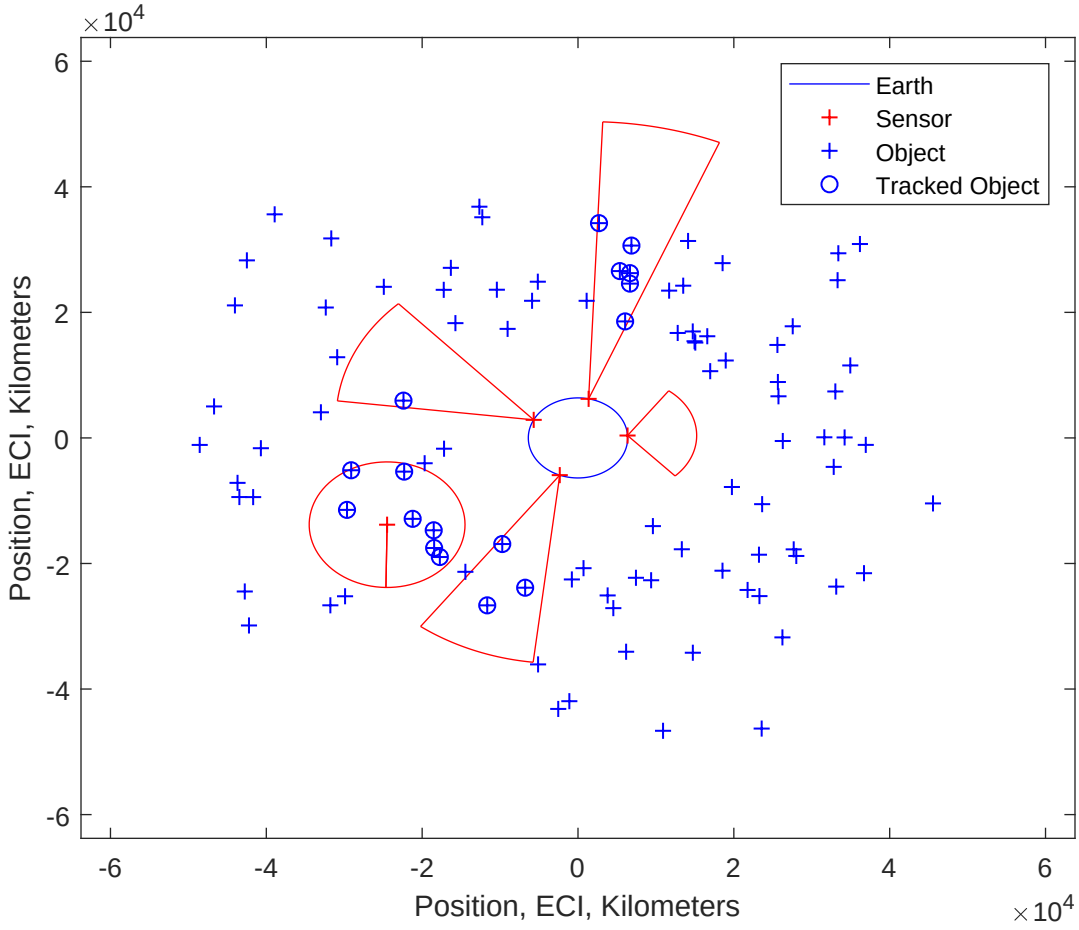

Fig. 1.1. Illustration of a sensor/object tracking simulation. In this case, 5 sensors and 100 space objects are modeled. Each sensor field of regard is outlined in red.

1.4 Thesis Outline

The rest of this thesis is organized as follows. In Chapter 2, the governing equations of motion of the space objects, and ground/space sensors, and the measurement

model are presented. In Chapter 3, the nonlinear estimation methods used in the analysis are described. This includes EKF, UKF, and Bootstrap Filter implementations. In Chapter 4, tasking methods are discussed. Observability measure based tasking is discussed in detail, and a summary of the various observability measures is presented. A new method of deriving a observability measure for a Bootstrap Filter is presented and verified in this section. Chapter 5 presents the simulation analysis of the methods described. Lastly, Chapter 6 provides the concluding remarks for this work.

CHAPTER 2

SYSTEM DYNAMICS

2.1 Space Object Dynamics

Consider a system of N space objects in random Geocentric orbits. These are observed by M sensors, which can be located on either the Earth's surface or in orbit. The semi-major axis of the i^{th} space object, a_i, is generated by a uniform distribution bounded by a_{lower} and a_{upper}. Eccentricities are generated by utilizing the following function:

$$e_i = \frac{(a_i - a_{lower})(a_{upper} - a_i)}{(a_{upper} - a_{lower})^2} \tag{2.1}$$

This function maps the eccentricity to a parabolic function such that the resulting value is constrained between between 0 and 0.25. This also ensures that orbits significantly closer to LEO (a_{lower}) and GEO (a_{upper}) are more circularized. These orbital elements are then converted to a state vector for initialization in the simulation. This system is modeled in a plane (Inertial Reference: $\{O_I,\ X_I,\ Y_I\}$, where O_I is the origin of the frame) in discrete time, where measurements are only available at $t = k\Delta t$, where k is the time interval [5]. The state of the i^{th} object is given as:

$$\boldsymbol{X}_{i,k} = \begin{bmatrix} x_{i,k} \\ y_{i,k} \\ \dot{x}_{i,k} \\ \dot{y}_{i,k} \end{bmatrix} \tag{2.2}$$

Where an object's equation of motion is modeled:

$$\dot{\boldsymbol{X}}_{i,k} = \begin{bmatrix} \dot{x}_{i,k} \\ \dot{y}_{i,k} \\ \ddot{x}_{i,k} \\ \ddot{y}_{i,k} \end{bmatrix} + \boldsymbol{G}\boldsymbol{w}_i = \begin{bmatrix} \dot{x}_{i,k} \\ \dot{y}_{i,k} \\ \frac{-\mu_e x_{i,k}}{[(x_{i,k})^2+(y_{i,k})^2]^{3/2}} \\ \frac{-\mu_e y_{i,k}}{[(x_{i,k})^2+(y_{i,k})^2]^{3/2}} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \boldsymbol{w}_i \tag{2.3}$$

Where μ_e is Earth's gravitational parameter. The lumped system uncertainty, $\boldsymbol{w}_i$, typically accounts for any un-modeled orbital pertubations such as drag and J_2 effects. $\boldsymbol{G} \in \mathbb{R}^{4\times 2}$ is the input distribution matrix. For this work, these disturbances are modeled as purely additive zero-mean process noise $\boldsymbol{w}_i \sim \mathcal{N}(0, \mathbf{Q})$ with covariance $Q \in \mathbb{R}^{2\times 2}$.

2.2 Sensor Dynamics

The governing equations of motion for the sensors are summarized in this section. Similar to the state of the objects to be tracked, the state of a sensor is modeled as:

$$\boldsymbol{s}_{j,k} = \begin{bmatrix} x^s_{j,k} \\ y^s_{j,k} \\ \dot{x}^s_{j,k} \\ \dot{y}^s_{j,k} \end{bmatrix} \tag{2.4}$$

Where a ground sensor's equation of motion is modeled:

$$\dot{\boldsymbol{s}}_{j,k} = \begin{bmatrix} \dot{x}^s_{j,k} \\ \dot{y}^s_{j,k} \\ \ddot{x}^s_{j,k} \\ \ddot{y}^s_{j,k} \end{bmatrix} = \begin{bmatrix} \dot{x}^s_{j,k} \\ \dot{y}^s_{j,k} \\ -\omega_e^2 x^s_{j,k} \\ -\omega_e^2 y^s_{j,k} \end{bmatrix} \tag{2.5}$$

Where ω_e is the average rotational speed of the Earth. A space based sensor's equations of motion are:

$$\dot{\boldsymbol{s}}_{j,k} = \begin{bmatrix} \dot{x}^s_{j,k} \\ \dot{y}^s_{j,k} \\ \ddot{x}^s_{j,k} \\ \ddot{y}^s_{j,k} \end{bmatrix} = \begin{bmatrix} \dot{x}^s_{j,k} \\ \dot{y}^s_{j,k} \\ \frac{-\mu_e x^s_{j,k}}{[(x^s_{j,k})^2+(y^s_{j,k})^2]^{3/2}} \\ \frac{-\mu_e y^s_{j,k}}{[(x^s_{j,k})^2+(y^s_{j,k})^2]^{3/2}} \end{bmatrix} \tag{2.6}$$

Which matches (2.3) without the inclusion of $\boldsymbol{w}_i$. It is assumed that sensor objects are perfectly known, with no uncertainty.

2.3 Measurement Model

A sensor's field-of-regard Γ_j has a length ray Δ_j and half-angle Ψ_j. The ray and angle measurements for an object/sensor pair are given by the nonlinear model [4],[5]:

$$\boldsymbol{y}_{(i,j),k} = \begin{bmatrix} \rho_{ij} \\ \psi_{ij} \end{bmatrix} + \mathbf{v}_{j,k} = \begin{bmatrix} \sqrt{(x_{i,k} - x^s_{j,k})^2 + (y_{i,k} - y^s_{j,k})^2} \\ \tan^{-1}\left(\frac{y_{i,k} - y^s_{j,k}}{x_{i,k} - x^s_{j,k}}\right) - \tan^{-1}\left(\frac{y^s_{j,k}}{x^s_{j,k}}\right) \end{bmatrix} + \begin{bmatrix} \nu^j_\rho \\ \nu^j_\psi \end{bmatrix} \tag{2.7}$$

Where $\mathbf{v_{j,k}}$ is the sensor noise vector represented as $\mathcal{N} \sim (0, R)$ with covariance $R = \begin{bmatrix} \nu^{j2}_\rho & 0 \\ 0 & \nu^{j2}_\psi \end{bmatrix}$ [4],[5]. An illustration of the sensor model is shown in Figure 2.1.

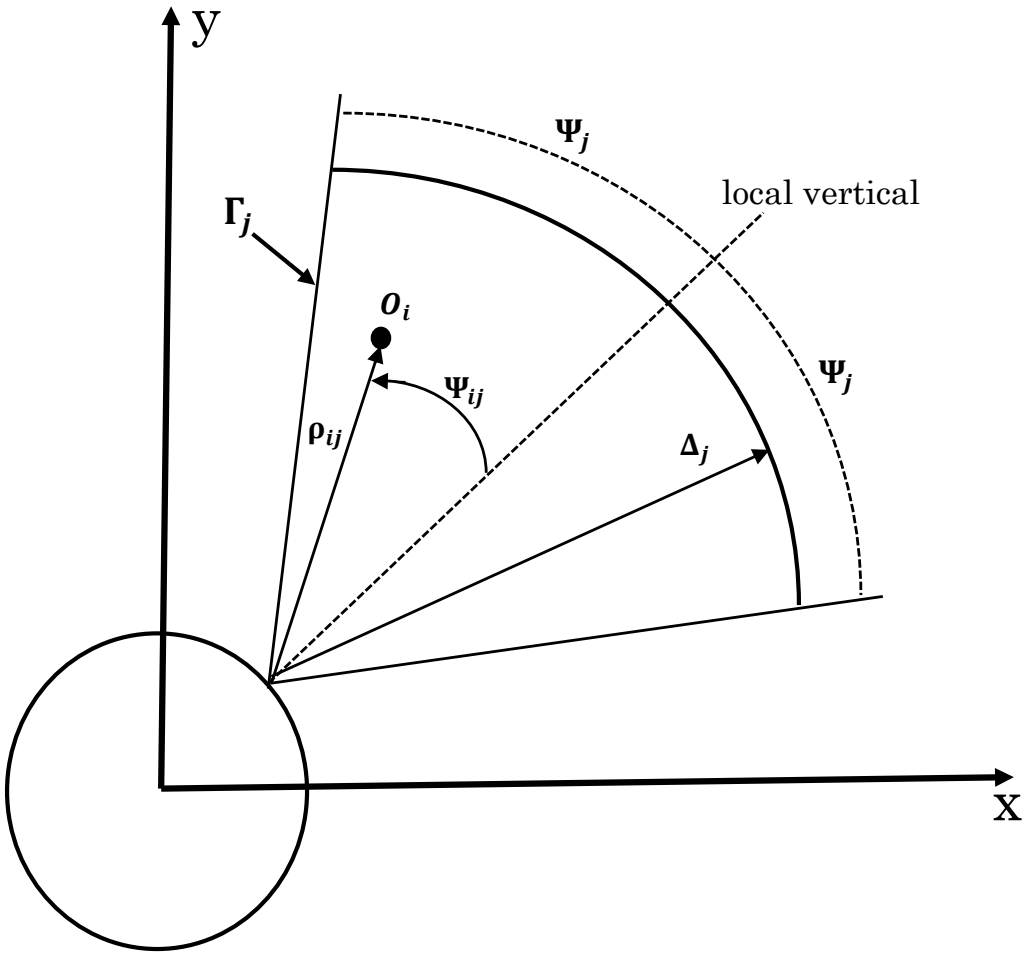

Fig. 2.1. Illustration of ground-based sensor model.

CHAPTER 3

ESTIMATION METHODS

The two most commonly employed nonlinear estimators, the Extended Kalman Filter (EKF) and the Unscented Kalman Filter (UKF) are implemented for this problem. A Bootstrap Particle Filter (BPF) is also implemented for a comparative analysis. Each estimator determines an optimal estimate for the state $\boldsymbol{X}_{i,k}^{+}$ and covariance $\boldsymbol{P}_{k}^{i+}$.

3.1 Extended Kalman Filter

The Extended Kalman Filter (EKF) estimates the state of a nonlinear system using the available state dynamics and measurement models, as well as the actual measurements. This is performed by linearizing the dynamics of the system and applying the result to the traditional Kalman Filter [16]. The state update after a measurement is processed is given by:

$$\hat{\boldsymbol{X}}_{i,k}^{+} = \begin{bmatrix} \hat{x}_{i,k}^{+} \\ \hat{y}_{i,k}^{+} \\ \hat{\dot{x}}_{i,k}^{+} \\ \hat{\dot{y}}_{i,k}^{+} \end{bmatrix} \tag{3.1}$$

and the updated covariance is given by:

$$\boldsymbol{P}_{k}^{i+} = \mathrm{E}[\boldsymbol{X}_{i,k} - \hat{\boldsymbol{X}}_{i,k}^{+}][\boldsymbol{X}_{i,k} - \hat{\boldsymbol{X}}_{i,k}^{+}]^{T} \tag{3.2}$$

The EKF, implemented from Ref. [4] and Ref. [5], is shown in Figure 3.1:

Prediction	$\dot{\hat{\boldsymbol{X}}}_{i,t} = \boldsymbol{f}(\hat{\boldsymbol{X}}_{i,t}), \qquad \hat{\boldsymbol{X}}_{i,t}(0) = \hat{\boldsymbol{X}}_i(0)$ $\dot{\boldsymbol{P}}_t^i = \boldsymbol{F}_t^i \boldsymbol{P}_t^i + \boldsymbol{P}_t^i \boldsymbol{F}_t^{iT} + \boldsymbol{Q}_t^i, \qquad \boldsymbol{P}_t^i(0) = \boldsymbol{P}^i(0)$
Kalman Gain	$\boldsymbol{K}_{k+1}^i = \boldsymbol{P}_{k+1}^{xy} \left[\boldsymbol{P}_{k+1}^{yy} + \boldsymbol{R}_{k+1}^i\right]^{-1}$ Where: $\boldsymbol{P}_{k+1}^{yy} = \boldsymbol{H}_{k+1}^i \boldsymbol{P}_{k+1}^{i-} [\boldsymbol{H}_{k+1}^i]^T$ $\boldsymbol{P}_{k+1}^{xy} = \boldsymbol{P}_{k+1}^{i-} [\boldsymbol{H}_{k+1}^i]^T$
Update	$\hat{\boldsymbol{X}}_{i,k+1}^+ = \hat{\boldsymbol{X}}_{i,k+1}^- + \boldsymbol{K}_{k+1}^i \left(\boldsymbol{y}_{(i,j),k+1} - \hat{\boldsymbol{y}}_{(i,j),k+1}\right)$ $\boldsymbol{P}_{k+1}^{i+} = \boldsymbol{P}_{k+1}^{i-} - \boldsymbol{K}_{k+1}^i [\boldsymbol{P}_{k+1}^{yy} + \boldsymbol{R}_{k+1}^i] [\boldsymbol{K}_{k+1}^i]^T$

Fig. 3.1. **Graphic of the EKF nonlinear estimator.**

In the prediction step, $\dot{\hat{\boldsymbol{X}}}_{i,t}$ and $\dot{\boldsymbol{P}}_t^i$ are integrated from $t \in [t_k, t_{k+1}]$ from the state $\hat{\boldsymbol{X}}_{i,k}^+$ and covariance $\boldsymbol{P}_k^{i+}$. The resulting values, $\hat{\boldsymbol{X}}_{i,t+1}$ and $\boldsymbol{P}_{t+1}^i$ are set as the predicted state $\hat{\boldsymbol{X}}_{i,k+1}^-$ and covariance $\boldsymbol{P}_{k+1}^{i-}$. $\boldsymbol{F}_t^i$ and $\boldsymbol{H}_{k+1}^i$ are Jacobians of the form:

$$\boldsymbol{F}_t^i = \left(\frac{\partial \boldsymbol{f}}{\partial \mathbf{X}}\right)_{\boldsymbol{X}=\hat{\boldsymbol{X}}_{i,t}}, \quad \boldsymbol{H}_{k+1}^i = \begin{bmatrix} \frac{\partial \rho_{ij}}{\partial \mathbf{X}} \\ \frac{\partial \psi_{ij}}{\partial \mathbf{X}} \end{bmatrix}_{\boldsymbol{X}=\hat{\boldsymbol{X}}_{i,k+1}^-}$$

3.2 Unscented Kalman Filter

The Unscented Kalman Filter (UKF) is a newer approach to nonlinear estimation using a Kalman Filter. This method works by modeling the Gaussian distribution of the state [21]. This distribution is modeled by generating a set of sigma points $\boldsymbol{\chi}_{i,t}$. These are then integrated $t \in [t_k, t_{k+1}]$ from the initial condition state $\boldsymbol{\chi}_{i,k}^+$ and covariance $\boldsymbol{P}_k^{i+}$. The resulting sigma points can then be used to construct the predicted state, the predicted covariance, the cross-covariance of the state and measurement, and the innovation covariance of the measurements. These then can be used to perform a Kalman Filter update. The UKF is an attractive alternative to the EKF, as it can be shown to have a lower estimated error for a nonlinear system [16]. However, this comes at an additional computational cost compared to the EKF [16].

The UKF utilizes a set of tuning parameters. Parameters α, β, γ, Λ, and κ are set to the following [16],[5]:

$$10^{-4} \leq \alpha \leq 1$$

$$\beta = 2$$

$$\gamma = \sqrt{L + \Lambda}$$

$$\Lambda = \alpha^2(L + \kappa) - L$$

$$\kappa = 3 - L$$

For this implementation, both noise and uncertainty are assumed to be additive zero-mean Guassian noise. This allows $\boldsymbol{\chi}_{i,t}$ to only need propagate the state and sets $n = L = 4$ [16]. This UKF, implemented from Ref. [5], is shown in Figure 3.2:

Prediction Step	$\hat{\boldsymbol{P}}^i_{CH} = \text{chol}(\boldsymbol{P}^{i+}_k)$ $\boldsymbol{\chi}_{i,t} = [\hat{\boldsymbol{X}}^-_{i,0}]_{1\times(2n+1)} + \sqrt{L+\lambda}\begin{bmatrix} 0_{n\times 1} & \hat{\boldsymbol{P}}^i_{CH} & -\hat{\boldsymbol{P}}^i_{CH}\end{bmatrix}$ $\dot{\boldsymbol{\chi}}_{i,t} = \boldsymbol{f}(\boldsymbol{\chi}_{i,t})$ $W^{mean}_\gamma = \begin{cases} \frac{\Lambda}{(n+\Lambda)}, & \gamma = 0 \\ \frac{1}{2(n+\Lambda)}, & \gamma = 1,\ldots,2n \end{cases}$ $\hat{\boldsymbol{X}}^-_{i,k+1} = \sum_{\gamma=0}^{2n} W^{mean}_\gamma \boldsymbol{\chi}_{(\gamma,i,k+1)}$ $W^{cov}_\gamma = \begin{cases} \frac{\Lambda}{(n+\Lambda)} + (1-\alpha^2+\beta), & \gamma = 0 \\ \frac{1}{2(n+\Lambda)}, & \gamma = 1,...,2n \end{cases}$ $\boldsymbol{P}^{i-}_{k+1} = \sum_{\gamma=0}^{2n} W^{cov}_\gamma \left[\boldsymbol{\chi}_{(\gamma,i,k+1)} - \hat{\boldsymbol{X}}^-_{i,k+1}\right]\left[\boldsymbol{\chi}_{(\gamma,i,k+1)} - \hat{\boldsymbol{X}}^-_{i,k+1}\right]^T + \boldsymbol{Q}^i_k$
Kalman Gain	$\boldsymbol{K}^i_{k+1} = \boldsymbol{P}^{xy}_{k+1}\left[\boldsymbol{P}^{yy}_{k+1} + \boldsymbol{R}^i_{k+1}\right]^{-1}$ Where: $\boldsymbol{P}^{xy}_{k+1} = \sum_{\gamma=0}^{2n} W^{cov}_\gamma \left[\boldsymbol{\zeta}_{(\gamma,i,k+1)} - \hat{\boldsymbol{y}}^-_{i,k+1}\right]\left[\boldsymbol{\chi}_{(\gamma,i,k+1)} - \hat{\boldsymbol{X}}^-_{i,k+1}\right]^T$ $\boldsymbol{P}^{yy}_{k+1} = \sum_{\gamma=0}^{2n} W^{cov}_\gamma \left[\boldsymbol{\zeta}_{(\gamma,i,k+1)} - \hat{\boldsymbol{y}}^-_{i,k+1}\right]\left[\boldsymbol{\zeta}_{(\gamma,i,k+1)} - \hat{\boldsymbol{y}}^-_{i,k+1}\right]^T$ $\boldsymbol{\zeta}_{i,k+1} = \begin{bmatrix}\rho_{ij}(\chi_{i,k+1})\\ \psi_{ij}(\chi_{i,k+1})\end{bmatrix}$ $\hat{\boldsymbol{y}}_{i,k+1} = \sum_{\gamma=0}^{2n} W^{mean}_\gamma \boldsymbol{\zeta}_{(\gamma,i,k+1)}$
Update Step	$\hat{\boldsymbol{X}}^+_{i,k+1} = \hat{\boldsymbol{X}}^-_{i,k+1} + \boldsymbol{K}^i_{k+1}\left(\boldsymbol{y}_{i,k+1} - \hat{\boldsymbol{y}}_{i,k+1}\right)$ $\boldsymbol{P}^{i+}_{k+1} = \boldsymbol{P}^{i-}_{k+1} - \boldsymbol{K}^i_{k+1}\left[\boldsymbol{P}^{yy}_{k+1} + \boldsymbol{R}^i_{k+1}\right]\left[\boldsymbol{K}^i_{k+1}\right]^T$

Fig. 3.2. Graphic of the UKF nonlinear estimator.

3.3 Bootstrap Particle Filter

The Bootstrap Particle Filter (BPF) from Ref. [16] is a implementation of a particle filter. This filter performs estimation through the direct simulation of the state and measurement distributions through a set of particles. These are used to directly model the posterior distribution $p(\tilde{\mathbf{y}}_{(i,j),k+1}|\mathbf{X}_{i,k}^{(p)})$, which is used in a function to weight the importance of a given particle. This estimator can then be used to determine the optimal estimate $\mathbf{X}_{i,k}^{+}$ and covariance $\mathbf{P}_{k}^{i+}$.

The direct modeling of the state probability distribution provides the BPF an attractive feature over the EKF or UKF. This makes no assumption on the distribution modeled, which may allow for the estimator to capture information about the state otherwise lost to an EKF or UKF. However, the accuracy of this distribution is dependent on the number of particles modeled. Since each particle requires numerical integration, the computational cost of this estimator may be significant compared to the prior estimators presented.

The implemented BPF is shown in the Figure below:

Prediction Step	$\dot{\mathbf{X}}_{i,k}^{(p)} = f(\mathbf{X}_{i,k}^{(p)}, \mathbf{w}_{i,k}^{(p)})$ $\mathbf{y}_{(i,j),k+1}{}^{(p)} = h(\hat{\mathbf{X}}_{i,k+1}^{(p)})$
Update Step	$e_{i,k+1}^{(p)} = \tilde{\mathbf{y}}_{(i,j),k+1} - \mathbf{y}_{(i,j),k+1}{}^{(p)}$ $P_{i,k+1}^{e(p)} = \mathrm{E}\{e_{i,k+1}^{(p)}(e_{i,k+1}^{(p)})^T\}$ $p(\tilde{\mathbf{y}}_{(i,j),k+1}\|\mathbf{X}_{i,k}^{(p)}) = \det(2\pi P_{k+1}^{e(p)})^{-\frac{1}{2}} \exp\left(-\frac{1}{2}(e_{i,k+1}^{(p)})^T (P_{i,k+1}^{e(p)})^{-1} e_{k+1}^{(p)}\right)$ $\mathbf{w}_{i,k+1} = \mathbf{w}_{i,k} p(\tilde{\mathbf{y}}_{(i,j),k+1}^{(p)}\|\mathbf{X}_{i,k}^{(p)})$ $\mathbf{w}_{i,k+1}^{(p)} = \frac{\mathbf{w}_{i,k+1}^{(p)}}{\sum_{p=1}^{Np} \mathbf{w}_{i,k+1}^{(p)}}$
Estimate Construction	$\hat{\mathbf{X}}_{i,k} \approx \sum_{p=1}^{Np} \mathbf{w}_{i,k+1} \mathbf{X}_{i,k}^{(p)}$
Covariance Construction	$\mathbf{P}_{k}^{i+} \approx \sum_{p=1}^{Np} \mathbf{w}_{i,k+1} (\mathbf{X}_{i,k}^{(p)} - \hat{\mathbf{X}}_{i,k})(\mathbf{X}_{i,k}^{(p)} - \hat{\mathbf{X}}_{i,k})^T$

Fig. 3.3. Graphic of the Bootstrap Filter nonlinear estimator.

A known issue with particle filter estimation is that a repeated sequence of importance sampling may continuously reduce the number of particles with non-negligible weighting until one particle remains [16]. A solution to this is monitoring the "Effective Sample Size" $\mathbf{N}_{eff}$ and to resample the particles to return the set to equal weighting.

An effective sample size can be determined by the following relationship:

$$\mathbf{N}_{eff} \approx \frac{1}{\sum_{j=1}^{N} \mathbf{w}_{k+1}^{(j)}} \tag{3.3}$$

Multiple resampling methods are avilable, with Multinomial Resampling being the method of choice [16]. This is initialized by taking a cumulative summation of the particle set:

$$\mathbf{z}_{i,k}^{(p_e)} = \sum_{p=1}^{(p_e)} \mathbf{w}_{(i,k)}^{(p)}$$

A single uniform sample $u^{(p)}$ is then drawn from N_p uniform samples on the interval of $(0, 1]$ for $p = 1, 2, ..., N_p$. The the index of the new particle set is then iterated:

while $\mathbf{z}_{i,k}^{(p_e)} < u^{(p)}$ **do**
| $p_e \leftarrow p_e + 1$

end

Where $\leftarrow$ indicates replacement. This index is then used to replace the particle set $\mathbf{X}_{i,k}^{(p)}$ with $\mathbf{X}_{i,k}^{(p_e)}$ and all values in the importance weights set are reset to $\frac{1}{N_p}$. Notably, system observability is a critical aspect of the methodologies being implemented. As a result, resampling must be judiciously applied to avoid observability losses [16]. For this instance, resampling only occurs if $\mathbf{N}_{eff}$ is less than half the size of the particle set.

Using the particle set ${\mathbf{y}_{(i,j),k+1}}^{(p)}$, the density $p(\hat{\mathbf{y}}_{(i,j),k+1}|\mathbf{X}_{i,k}^{(p)})$ can be generated from a Gaussian Kernel function. To determine the appropriate bandwidth for a given vector element, Silverman's 'Rule of Thumb' can be employed [17]:

$$b_L = \sigma_L \left(\frac{4}{d+2} N_p \right)^{\frac{1}{d+4}}, \quad L = 1, 2, ..., d \tag{3.4}$$

Where b_L and σ_L are the bandwidth and standard deviation of the L^{th} random variable of the particle set ${\mathbf{y}_{(i,j),k+1}}^{(p)}$ of dimension d. From this, N_p probability values are generated for each particle set. This is an important development, as it will be used to determine sensor tasking measures in subsequent sections.

CHAPTER 4

TASKING METHODS

Since the objective of this thesis is to compare alternative observability measures to a previously published tasking solution, we adopt the same framework for the sensor tasking of space objects as in Ref. [4] . If the i^{th} object is being considered by the j^{th} sensor, the sensor tasking solution is solved in the form of the following linear programming problem:

$$\max \left(\sum_{i=1}^{N} \sum_{j=1}^{M} \mu^{ij} \xi^{ij} \right), \qquad \xi^{ij} \in \{0,\ 1\} \tag{4.1}$$

Where μ_{ij} is the sensor gain for an object-sensor pair and ξ^{ij} is the tasking solution. The tasking solution is subject to the constraint [4], [5]:

$$\sum_{i=1}^{N} \xi^{ij} \leq 1, \qquad j = 1, \ldots, M \tag{4.2}$$

The solution to the above will ensure that a single sensor observes only one object at any instant and the selected sensor-object pair is the one with the largest μ^{ij} value.

4.1 Previous Tasking Methods

There have been some established methods to determine an appropriate value of μ^{ij}, which include the use of the trace of the Fisher Information for an object-sensor pair from the Kalman Filter (EKF or UKF) covariance update step [4], [5]:

$$\mu_F^{ij} = \text{Tr}(\mathbf{\Omega}_k^{ij}) \tag{4.3}$$

Where $\mathbf{\Omega}_k^{ij}$ is the Fisher-Information for an object-sensor pair at a sample-instant k. The resulting gain μ_F^{ij} is referred to as the Fisher Information Gain or FIG.

While the FIG tasking method can be shown to manage uncertainty, it does so at the cost of values with a low FIG measure. This could result in estimates not receiving an adequate amount of updates [5]. Another method introduced in Ref. [5] is intended to work around this and utilizes an approximation of the Largest Lyanpunov Exponent (LLE):

$$\mu_L^{ij} = \frac{1}{t_k} \ln_2 \sqrt{\frac{\mathrm{Tr}(\boldsymbol{P}_k^{i+})}{\mathrm{Tr}(\boldsymbol{P}_0^i)}} \tag{4.4}$$

Where $\boldsymbol{P}_0^i$ is the estimation error covariance at initialization, and $\boldsymbol{P}_k^{i+}$ is the updated covariance of the estimate. This method essentially measures the stability of the system, where a higher value indicates a greater difficulty predicting the estimate [5]. This was shown to be an improvement over FIG sensor tasking [5].

4.2 Observability Measure based Tasking

From the tasking methods presented, it is seen that some form of prioritization scheme is used on objects that are visible to the sensor set. Since a stochastic system is modeled, a reasonable extension of the sensor tasking framework would be to utilize a measure of stochastic observability. In this manner, the tasking measure also provides the most amount of information about the system to the estimator at each time step. Ref. [15] identifies Mutual Information, Bhattacharya Distance, and the Squared Hellinger Distance as candidates for observability measures.

An implementation of Mutual Information as a sensor tasking measure has been previously implemented by Ref. [6]. This was referred to as the Shannon Information Gain or SIG [6]. This is performed by deriving the Mutual Information (MI) between two continuous Gaussian random variables, and applying the assumptions of a Kalman Filter for a nonlinear system [6],[15]:

$$\mu_S^{ij} = \mathrm{MI}(\boldsymbol{x}_k^i, \tilde{\boldsymbol{y}}_k^{ij}) = \frac{1}{2} \ln \left(\frac{\det\left(\boldsymbol{P}_k^{i-}\right)}{\det\left(\boldsymbol{P}_k^{i+}\right)} \right) \tag{4.5}$$

Like LLE tasking, SIG tasking has been shown to be an improvement over FIG tasking [6]. Notably, both SIG and FIG measures are directly related to the Kullback-Leibler (KL) divergence. For a pair of continuous random variables, Mutual Information is solved as a special case of the Kullback-Leibler divergence [13]. Whereas, Fisher Information is the curvature of the KL divergence with respect to the random variables. Given that MI is an established means to measure the distance between distributions, it is intuitive that the curvature of a statistical distance provides less information [15]. It is also seen that SIG tasking can perform better than LLE tasking under certain sensor noise conditions [6].

Consider that the tasking solution in Eq. (4.1) evaluates a set of μ^{ij} values, where each value is assumed to be an objective measure of prioritization. To correctly synthesize μ^{ij}, it is imperative that a stochastic observability measure is consistent with the definition of a distance function or metric. Consider that the properties of the distance function $d(x, y)$ are [18]:

1. It is nonnegative: $d(x, y) > 0$, when $x \neq y$ and $d(x, x) = 0$
2. Satisifies Triangle Inequality: $d(x, z) \leq d(x, y) + d(y, z)$
3. Satisfies Symmetry: $d(x, y) = d(y, x)$

However, to compare stochastic observability measures such as the 'distance' between two pdfs is not trivial. While KL divergence is typically employed to judge the closeness of two pdfs, it is not a metric. It doesn't satisfy the triangle inequality and is not symmetric either. This is clearly seen from the definition of the KL divergence for two multivariate Gaussian distributions $p_1(\boldsymbol{x}, \boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$ $p_2(\boldsymbol{x}, \boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$ i.e. $D_{KL}(p_1, p_2) = \int p_1 \ln \left(\dfrac{p_1}{p_2} \right) d\boldsymbol{x}$. While there are other statistical distance measures, two are focused on, namely the Bhattacharya Distance (BD), and the Squared Hellinger Distance (HD). Ref. [15] also identifies that BD is a bounded metric, but does not satisfy the Triangle Inequality, but HD does and is a bounded metric. The

following section derives the Bhattacharya Distance and Squared Hellinger Distance observability measures based on the same assumptions that were applied to the Mutual Information to derive the SIG. A simplified analysis is performed using scalar random values to compare the MI, BD, and HD.

4.2.1 Mutual Information (MI), Bhattacharya Distance (BD) and the Squared Hellinger Distance (HD) Observability Measures:

Given the joint density $h(\boldsymbol{x}, \boldsymbol{y})$ of two continuous random variables, the $MI(\boldsymbol{x}, \boldsymbol{y})$ is computed as [13]:

$$MI(\boldsymbol{x}, \boldsymbol{y}) = \int \int h(\boldsymbol{x}, \boldsymbol{y}) \ln \left(\frac{h(\boldsymbol{x}, \boldsymbol{y})}{h(\boldsymbol{x}) h(\boldsymbol{y})} \right) d\boldsymbol{x} d\boldsymbol{y} \tag{4.6}$$

Note, the above is a special case of the KL divergence defined in the previous section. Now, consider the definition of Bhattacharya Distance (BD), which measures the distance between the densities $f(\boldsymbol{x})$ and $g(\boldsymbol{x})$ [14],[19]:

$$BD(f(\boldsymbol{x}), g(\boldsymbol{x})) = -\ln \left(BC \left(f(\boldsymbol{x}), g(\boldsymbol{x}) \right) \right) \qquad 0 \leq BD \leq \infty \tag{4.7}$$

where, BC is the Bhattacharya Coefficient and is defined as,

$$BC(f(\boldsymbol{x}), g(\boldsymbol{x})) = \int \sqrt{f(\boldsymbol{x}) g(\boldsymbol{x})} d\boldsymbol{x} \qquad 0 \leq BC \leq 1 \tag{4.8}$$

Consider the densities $f(\boldsymbol{x})$ and $g(\boldsymbol{x})$ to be a set of multivariate Gaussian distributions, $f(\boldsymbol{x}) \sim \mathcal{N}(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$ and $g(\boldsymbol{x}) \sim \mathcal{N}(\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2)$. It can be shown that the Bhattacharya Distance between the two distributions is [19]:

$$BD(f(\boldsymbol{x}), g(\boldsymbol{x})) = \frac{1}{8} (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^T \left(\frac{\boldsymbol{\Sigma}_1 + \boldsymbol{\Sigma}_2}{2} \right)^{-1} (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2) + \frac{1}{2} \ln \frac{\det \left(\frac{\boldsymbol{\Sigma}_1 + \boldsymbol{\Sigma}_2}{2} \right)}{\sqrt{\det \left(\boldsymbol{\Sigma}_1 \right) \det \left(\boldsymbol{\Sigma}_2 \right)}} \tag{4.9}$$

Note: The BD is symmetric as opposed to the KL divergence, but does not satisfy the triangle inequality.

The Squared Hellinger Distance for the two densities mentioned above is defined as:

$$HD^2(f(\boldsymbol{x}), g(\boldsymbol{x})) = \frac{1}{2}\int \left(\sqrt{f(\boldsymbol{x})} - \sqrt{g(\boldsymbol{x})}\right)^2 d\boldsymbol{x} \qquad 0 \leq HD \leq 1 \tag{4.10}$$

It has been shown in Ref. [14] that HD obeys the triangle inequality, and thus is the only measure to satisfy the properties of the distance function.

4.2.2 Application of the MI, BD, and HD to a dynamical system

Consider an ensemble of N agents where each agent is governed by nonlinear dynamics as in Eq. (4.11). We also assume there are M sensors where each sensor (j) observes one agent (i) at any instant (k) and the corresponding measurement $\tilde{\boldsymbol{y}}_k^{ij}$ is given by Eq. (4.12).

$$\boldsymbol{x}_k^i = f(\boldsymbol{x}_{k-1}^i) + \boldsymbol{w}_k \qquad \boldsymbol{w}_k \sim \mathcal{N}(0, \sigma_q^2) \tag{4.11}$$

$$\tilde{\boldsymbol{y}}_k^{ij} = h(\boldsymbol{x}_k^i) + \boldsymbol{\nu}_k \qquad \boldsymbol{\nu}_k \sim \mathcal{N}(0, \sigma_r^2) \tag{4.12}$$

The error between the estimate and actual state is $\boldsymbol{e}_x = \hat{\boldsymbol{x}}_k^i - \boldsymbol{x}_k^i$ and the residual between the measurement and the predicted measurement is $\boldsymbol{e}_y = \tilde{\boldsymbol{y}}_k^{ij} - \hat{\boldsymbol{y}}_k^{ij}$. The covariance of the estimation errors is given by $\boldsymbol{P}_k^{e_x e_x} = E\{\boldsymbol{e}_x \boldsymbol{e}_x{}^T\}$ and the innovations covariance is given by $\boldsymbol{P}_k^{e_y e_y} = E\{\boldsymbol{e}_y \boldsymbol{e}_y{}^T\}$. Additionally, the cross-covariance between the predicted state and the measurement is defined as, $\boldsymbol{P}_k^{e_y e_x} = E\{\boldsymbol{e}_y \boldsymbol{e}_x{}^T\}$. Consider the joint distribution of $\boldsymbol{x}_k^i$ and $\tilde{\boldsymbol{y}}_k^{ij}$, which is denoted as $p(\boldsymbol{x}_k^i, \tilde{\boldsymbol{y}}_k^{ij})$. Consider that covariance of the joint distribution is of the form:

$$\boldsymbol{P}^{Jxy} = \begin{bmatrix} \boldsymbol{P}_k^{e_x e_x} & P_{n\times m}^J \\ P_{n\times m}^J{}^T & \boldsymbol{P}_k^{e_y e_y} \end{bmatrix}$$

Where $P_{n\times m}^J$ describes how $\boldsymbol{x}_k^i$ and $\tilde{\boldsymbol{y}}_k^{ij}$ correlate with each other. However, if $\boldsymbol{x}_k^i$ and $\tilde{\boldsymbol{y}}_k^{ij}$ were independent from one another, it is known that:

$$p(\boldsymbol{x}_k^i, \tilde{\boldsymbol{y}}_k^{ij}) = p(\boldsymbol{x}_k^i)p(\tilde{\boldsymbol{y}}_k^{ij}) \tag{4.13}$$

If $\boldsymbol{x}_k^i \in \mathbb{R}^n$ and $\tilde{\boldsymbol{y}}_k^{ij} \in \mathbb{R}^m$, the covariance of this distribution would be of the form:

$$\boldsymbol{P}^{Ixy} = \begin{bmatrix} \boldsymbol{P}_k^{e_x e_x} & 0_{n\times m} \\ 0_{m\times n} & \boldsymbol{P}_k^{e_y e_y} \end{bmatrix}$$

Since there is no correlation between the random variables, there is nothing the measurement $\tilde{\boldsymbol{y}}_k^{ij}$ could add to the estimate of $\boldsymbol{x}_k^i$. In effect, this joint distribution represents an un-observable system at a given time k.

If some distance measure was obtained between the "true" joint density and the "independent" joint density, it would describe how far the former was from the completely un-observable independent density.

Given this, it is already known the Mutual Information is a special form of the Kullback-Leibler divergence that measures the distance between the joint and independent densities of two variables [13]:

$$MI(\boldsymbol{x}_k^i, \tilde{\boldsymbol{y}}_k^{ij}) = \int\int p(\boldsymbol{x}_k^i, \tilde{\boldsymbol{y}}_k^{ij}) \ln\left(\frac{p(\boldsymbol{x}_k^i, \tilde{\boldsymbol{y}}_k^{ij})}{p(\boldsymbol{x}_k^i)p(\tilde{\boldsymbol{y}}_k^{ij})}\right) d\boldsymbol{x}_k^i d\tilde{\boldsymbol{y}}_k^{ij} \tag{4.14}$$

Bhattacharya and Hellinger distances are generalized probability distribution distance measurements. The distance between the joint and independent densities for these distances are of the form:

$$BD(p(\boldsymbol{x}_k^i, \tilde{\boldsymbol{y}}_k^{ij}), p(\boldsymbol{x}_k^i)p(\tilde{\boldsymbol{y}}_k^{ij})) = -\ln\left(BC\left(p(\boldsymbol{x}_k^i, \tilde{\boldsymbol{y}}_k^{ij}), p(\boldsymbol{x}_k^i)p(\tilde{\boldsymbol{y}}_k^{ij})\right)\right) \qquad 0 \leq BD \leq \infty \tag{4.15}$$

Where:

$$BC(p(\boldsymbol{x}_k^i, \tilde{\boldsymbol{y}}_k^{ij}), p(\boldsymbol{x}_k^i)p(\tilde{\boldsymbol{y}}_k^{ij})) = \int\int \sqrt{p(\boldsymbol{x}_k^i, \tilde{\boldsymbol{y}}_k^{ij})p(\boldsymbol{x}_k^i)p(\tilde{\boldsymbol{y}}_k^{ij})} d\boldsymbol{x}_k^i d\tilde{\boldsymbol{y}}_k^{ij} \qquad 0 \leq BC \leq 1 \tag{4.16}$$

$$HD^2(p(\boldsymbol{x}_k^i, \tilde{\boldsymbol{y}}_k^{ij}), p(\boldsymbol{x}_k^i)p(\tilde{\boldsymbol{y}}_k^{ij})) = \frac{1}{2}\int\int \left(\sqrt{p(\boldsymbol{x}_k^i, \tilde{\boldsymbol{y}}_k^{ij})} - \sqrt{p(\boldsymbol{x}_k^i)p(\tilde{\boldsymbol{y}}_k^{ij})}\right)^2 d\boldsymbol{x}_k^i d\tilde{\boldsymbol{y}}_k^{ij} \qquad 0 \leq HD \leq 1 \tag{4.17}$$

Or alternatively:

$$HD^2(p(\boldsymbol{x}_k^i, \tilde{\boldsymbol{y}}_k^{ij}), p(\boldsymbol{x}_k^i)p(\tilde{\boldsymbol{y}}_k^{ij})) = 1 - BC(p(\boldsymbol{x}_k^i, \tilde{\boldsymbol{y}}_k^{ij}), p(\boldsymbol{x}_k^i)p(\tilde{\boldsymbol{y}}_k^{ij})) \qquad 0 \leq HD \leq 1 \tag{4.18}$$

Which can be utilized as measures of stochastic observability.

4.2.3 Application of Particle Filter Assumptions

A particle filter estimates a system through a set of particles $\mathbf{X}_{i,k}^{(p)}$ representing the state and its uncertainty. From this particle set, the densities of the state and measurement can be directly constructed, even if the distribution is non-Gaussian. The measurement estimate for a given particle can be modeled as:

$$\mathbf{y}_{(i,j),k}^{(p)} = h(\mathbf{X}_{i,k}^{(p)})$$

The particle sets $\mathbf{X}_{i,k}^{(p)}$ and $\mathbf{y}_{(i,j),k}^{(p)}$ can then be used to generate approximations of the densities $p(\boldsymbol{x}_k^i)$ and $p(\tilde{\boldsymbol{y}}_{k+1}^{ij}|\boldsymbol{x}_k^i)$ respectively. From these, the relationships for $p(\tilde{\boldsymbol{y}}_k^{ij}, \boldsymbol{x}_k^i)$ and $p(\tilde{\boldsymbol{y}}_k^{ij})$ can be found:

$$p(\tilde{\boldsymbol{y}}_k^{ij}, \boldsymbol{x}_k^i) = p(\tilde{\boldsymbol{y}}_k^{ij}|\boldsymbol{x}_k^i)p(\boldsymbol{x}_k^i)$$

$$p(\tilde{\boldsymbol{y}}_k^{ij}) = \int p(\tilde{\boldsymbol{y}}_k^{ij}|\boldsymbol{x}_k^i)p(\boldsymbol{x}_k^i)d\boldsymbol{x}_k^i$$

Equations (4.14)-(4.18) can then be directly employed using a quadrature technique. However, this would require a quadrature of order 6 to evaluate for the present problem. This may significantly affect the computation time due to the curse of dimensionality. However, a particle filter generates an importance function $\mathbf{w}_k^{(p)}$ that can be used to approximate the integral:

$$\sum_{p=1}^{Np} \mathbf{w}_{i,k}^{(p)} f(\mathbf{x}_{i,k}^{(p)}) \approx \int p(\boldsymbol{x}_k^i) f(\boldsymbol{x}_k^i) d\boldsymbol{x}_k^i$$

Which can be used to reduce the order of integration for a quadrature technique. From Ref. [20] it is known that the mutual information can be modified to be of the form:

$$\begin{aligned} MI(\boldsymbol{x}_k^i, \tilde{\boldsymbol{y}}_k^{ij}) = & \int \sum_{p=1}^{Np} \{\mathbf{w}_k^{(p)} p(\tilde{\boldsymbol{y}}_k^{ij} | \boldsymbol{x}_k^i = \mathbf{x}_{i,k}^{(p)}) \ln p(\tilde{\boldsymbol{y}}_k^{ij} | \boldsymbol{x}_k^i = \mathbf{x}_{i,k}^{(p)}))\} d\tilde{\boldsymbol{y}}_k^{ij} \\ & - \int \{\sum_{p=1}^{Np} (\mathbf{w}_k^{(p)} p(\tilde{\boldsymbol{y}}_k^{ij} | \boldsymbol{x}_k^i = \mathbf{x}_{i,k}^{(p)}) \ln (\sum_{p=1}^{Np} \mathbf{w}_k^{(p)} p(\tilde{\boldsymbol{y}}_k^{ij} | \boldsymbol{x}_k^i = \mathbf{x}_{i,k}^{(p)}))\} d\tilde{\boldsymbol{y}}_k^{ij} \end{aligned} \tag{4.19}$$

Which reduces the quadrature technique needed to order 2 for the problem presented. Similarly, the Bhattacharya Coefficient can be re-arranged:

$$BC(p(\boldsymbol{x}_k^i, \tilde{\boldsymbol{y}}_k^{ij}), p(\boldsymbol{x}_k^i) p(\tilde{\boldsymbol{y}}_k^{ij})) = \int \sqrt{p(\tilde{\boldsymbol{y}}_k^{ij})} \int p(\boldsymbol{x}_k^i) \sqrt{p(\tilde{\boldsymbol{y}}_k^{ij} | \boldsymbol{x}_k^i)} d\boldsymbol{x}_k^i d\tilde{\boldsymbol{y}}_k^{ij}$$

Which can be modified to the form:

$$BC(p(\boldsymbol{x}_k^i, \tilde{\boldsymbol{y}}_k^{ij}), p(\boldsymbol{x}_k^i) p(\tilde{\boldsymbol{y}}_k^{ij})) = \int \sqrt{\sum_{p=1}^{Np} (\mathbf{w}_k^{(p)} p(\tilde{\boldsymbol{y}}_k^{ij} | \boldsymbol{x}_k^i = \mathbf{x}_{i,k}^{(p)}) \sum_{p=1}^{Np} \{\mathbf{w}_k^{(p)} \sqrt{p(\tilde{\boldsymbol{y}}_k^{ij} | \boldsymbol{x}_k^i = \mathbf{x}_{i,k}^{(p)})}\} d\tilde{\boldsymbol{y}}_k^{ij}} \tag{4.20}$$

Similarly, this also will only require a quadrature technique of order 2 to evaluate. This can then be directly applied to equations (4.16) and (4.18) to determine the Bhattacharya and Squared Hellinger distances.

4.2.4 Application of Gaussian and Kalman Filter assumptions

Under the assumption that the state, measurement, and joint estimate can be modeled as Gaussian distributions, we obtain:

$$p(\boldsymbol{x}_k^i) \sim \mathcal{N}(\hat{\boldsymbol{x}}_k^{i-}, \boldsymbol{P}_k^{e_x e_x}) \tag{4.21}$$

$$p(\tilde{\boldsymbol{y}}_k^{ij}) \sim \mathcal{N}(\hat{\boldsymbol{y}}_k^{ij}, \boldsymbol{P}_k^{e_y e_y}) \tag{4.22}$$

Where the "true" joint distribution was a density where some correlation may occur i.e.

$$p(\boldsymbol{x}_k^i, \tilde{\boldsymbol{y}}_k^{ij}) \sim \mathcal{N}\left(\begin{bmatrix} \hat{\boldsymbol{x}}_k^{i-} \\ \tilde{\boldsymbol{y}}_k^{ij} \end{bmatrix}_J, \boldsymbol{P}^{Jxy} \right) \tag{4.23}$$

Eq. (4.13) and Eq. (4.23) are thus utilized to compute the various observability measures. Substituting the state estimates and covariances into Eq. (4.9), the BD measure is computed as:

$$\begin{aligned} \mathrm{BD}(p(\boldsymbol{x}_k^i, \tilde{\boldsymbol{y}}_k^{ij}), p(\boldsymbol{x}_k^i)p(\tilde{\boldsymbol{y}}_k^{ij})) &= \frac{1}{8}\left(\begin{bmatrix} \hat{\boldsymbol{x}}_k^{i-} \\ \tilde{\boldsymbol{y}}_k^{ij} \end{bmatrix}_J - \begin{bmatrix} \hat{\boldsymbol{x}}_k^{i-} \\ \tilde{\boldsymbol{y}}_k^{ij} \end{bmatrix}_I \right)^T \left[\frac{\boldsymbol{P}^{Jxy} + \boldsymbol{P}^{Ixy}}{2} \right]^{-1} \left(\begin{bmatrix} \hat{\boldsymbol{x}}_k^{i-} \\ \tilde{\boldsymbol{y}}_k^{ij} \end{bmatrix}_J - \begin{bmatrix} \hat{\boldsymbol{x}}_k^{i-} \\ \tilde{\boldsymbol{y}}_k^{ij} \end{bmatrix}_I \right) \\ &\quad + \frac{1}{2} \ln \left(\frac{\det\left[\frac{\boldsymbol{P}^{Jxy} + \boldsymbol{P}^{Ixy}}{2} \right]}{\sqrt{\det\left[\boldsymbol{P}^{Jxy}\right] \det\left[\boldsymbol{P}^{Ixy}\right]}} \right) \end{aligned} \tag{4.24}$$

Under the assumptions of a Kalman Filter, it is known that the estimate of the joint density and the independent density at a given time k are the same:

$$\begin{bmatrix} \hat{\boldsymbol{x}}_k^{i-} \\ \tilde{\boldsymbol{y}}_k^{ij} \end{bmatrix}_J = \begin{bmatrix} \hat{\boldsymbol{x}}_k^{i-} \\ \tilde{\boldsymbol{y}}_k^{ij} \end{bmatrix}_I$$

This reduces BD in (4.24) to:

$$\mathrm{BD}(p(\boldsymbol{x}_k^i, \tilde{\boldsymbol{y}}_k^{ij}), p(\boldsymbol{x}_k^i)p(\tilde{\boldsymbol{y}}_k^{ij})) = \frac{1}{2} \ln \left(\frac{\det\left[\frac{\boldsymbol{P}^{Jxy} + \boldsymbol{P}^{Ixy}}{2} \right]}{\sqrt{\det\left[\boldsymbol{P}^{Jxy}\right] \det\left[\boldsymbol{P}^{Ixy}\right]}} \right) \tag{4.25}$$

The assumption of a Kalman Filter allows the usage of a Gaussian Joint Distribution, with the covariance $\boldsymbol{P}^{Jxy}$ of the form:

$$\left[\boldsymbol{P}^{Jxy}\right] = \begin{bmatrix} \boldsymbol{P}_k^{e_x e_x} & \boldsymbol{P}_k^{e_x e_y} \\ \left(\boldsymbol{P}_k^{e_x e_y}\right)^T & \boldsymbol{P}_k^{e_y e_y} \end{bmatrix}$$

From Eq. (4.25) we see:

$$\left[\frac{\boldsymbol{P}^{Jxy} + \boldsymbol{P}^{Ixy}}{2}\right] = \begin{bmatrix} \boldsymbol{P}_k^{e_x e_x} & \frac{1}{2}\boldsymbol{P}_k^{e_x e_y} \\ \frac{1}{2}\left(\boldsymbol{P}_k^{e_x e_y}\right)^T & \boldsymbol{P}_k^{e_y e_y} \end{bmatrix}$$

Where the determinant of this relationship can be reduced to:

$$\det\left[\frac{\boldsymbol{P}^{Jxy} + \boldsymbol{P}^{Ixy}}{2}\right] = \det\left[\boldsymbol{P}_k^{e_y e_y}\right] \det\left[\boldsymbol{P}_k^{e_x e_x} - \frac{1}{4}\boldsymbol{P}_k^{e_x e_y}\left[\boldsymbol{P}^{e_y e_y}\right]_k^{-1}\left[\boldsymbol{P}_k^{e_x e_y}\right]^T\right] \quad (4.26)$$

Similarly, it can be seen that the determinant of $\boldsymbol{P}^{Jxy}$ is of the form:

$$\det\left[\boldsymbol{P}^{Jxy}\right] = \det\left[\boldsymbol{P}_k^{e_y e_y}\right] \det\left[\boldsymbol{P}_k^{e_x e_x} - \boldsymbol{P}_k^{e_x e_y}\left[\boldsymbol{P}^{e_y e_y}\right]_k^{-1}\left[\boldsymbol{P}_k^{e_x e_y}\right]^T\right] \quad (4.27)$$

And the determinant of $\boldsymbol{P}^{Ixy}$ is:

$$\det\left[\boldsymbol{P}^{Ixy}\right] = \det\left[\boldsymbol{P}_k^{e_x e_x}\right] \det\left[\boldsymbol{P}_k^{e_y e_y}\right] \quad (4.28)$$

Subsituting (4.26), (4.27) and (4.28) into (4.25) and reducing terms, it is found:

$$\mathrm{BD}(p(\boldsymbol{x}_k^i, \tilde{\boldsymbol{y}}_k^{ij}), p(\boldsymbol{x}_k^i)p(\tilde{\boldsymbol{y}}_k^{ij})) = \frac{1}{2}\ln\left(\frac{\det\left[\boldsymbol{P}_k^{e_x e_x} - \frac{1}{4}\boldsymbol{P}_k^{e_x e_y}\left[\boldsymbol{P}^{e_y e_y}\right]_k^{-1}\left[\boldsymbol{P}_k^{e_x e_y}\right]^T\right]}{\sqrt{\det\left[\boldsymbol{P}_k^{e_x e_x} - \boldsymbol{P}_k^{e_x e_y}\left[\boldsymbol{P}^{e_y e_y}\right]_k^{-1}\left[\boldsymbol{P}_k^{e_x e_y}\right]^T\right]\det\left[\boldsymbol{P}_k^{e_x e_x}\right]}}\right) \quad (4.29)$$

Alternately,

$$\mathrm{BD}(p(\boldsymbol{x}_k^i, \tilde{\boldsymbol{y}}_k^{ij}), p(\boldsymbol{x}_k^i)p(\tilde{\boldsymbol{y}}_k^{ij})) = \frac{1}{2}\ln\left(\frac{\det\left[\frac{3}{4}\boldsymbol{P}_k^{e_x e_x} + \frac{1}{4}\left(\boldsymbol{P}_k^{e_x e_x} - \boldsymbol{P}_k^{e_x e_y}\left[\boldsymbol{P}^{e_y e_y}\right]_k^{-1}\left[\boldsymbol{P}_k^{e_x e_y}\right]^T\right]\right)}{\sqrt{\det\left[\boldsymbol{P}_k^{e_x e_x} - \boldsymbol{P}_k^{e_x e_y}\left[\boldsymbol{P}^{e_y e_y}\right]_k^{-1}\left[\boldsymbol{P}_k^{e_x e_y}\right]^T\right]\det\left[\boldsymbol{P}_k^{e_x e_x}\right]}}\right) \quad (4.30)$$

Again, applying the Kalman Filter assumptions, we recognize that the predicted error covariance is:

$$\boldsymbol{P}_k^{i-} = \boldsymbol{P}_k^{e_x e_x}$$

and the updated error covariance is:

$$\boldsymbol{P}_k^{i+} = \boldsymbol{P}_k^{e_x e_x} - \boldsymbol{P}_k^{e_x e_y} \left[\boldsymbol{P}^{e_y e_y}\right]_k^{-1} \left[\boldsymbol{P}_k^{e_x e_y}\right]^T \tag{4.31}$$

Substituting these relationships into (4.30), we obtain:

$$\mathrm{BD}(p(\boldsymbol{x}_k^i, \tilde{\boldsymbol{y}}_k^{ij}), p(\boldsymbol{x}_k^i)p(\tilde{\boldsymbol{y}}_k^{ij})) = \frac{1}{2} \ln \left(\frac{\det\left[\frac{3}{4}\boldsymbol{P}_k^{i-} + \frac{1}{4}\boldsymbol{P}_k^{i+}\right]}{\sqrt{\det\left[\boldsymbol{P}_k^{i+}\right] \det\left[\boldsymbol{P}_k^{i-}\right]}} \right) \tag{4.32}$$

which is the same as the one found in Ref. [15]. Since HD can be determined directly from the BD, we obtain [15]:

$$\mathrm{HD}(f(\boldsymbol{x}), g(\boldsymbol{x})) = \sqrt{1 - \exp\left(-\mathrm{BD}(f(\boldsymbol{x}), g(\boldsymbol{x}))\right)} \tag{4.33}$$

The Hellinger Distance observability measure is simply:

$$\mathrm{HD}(p(\boldsymbol{x}_k^i, \tilde{\boldsymbol{y}}_k^{ij}), p(\boldsymbol{x}_k^i)p(\tilde{\boldsymbol{y}}_k^{ij})) = \sqrt{1 - \frac{\sqrt[4]{\det\left[\boldsymbol{P}_k^{i+}\right] \det\left[\boldsymbol{P}_k^{i-}\right]}}{\sqrt{\det\left[\frac{3}{4}\boldsymbol{P}_k^{i-} + \frac{1}{4}\boldsymbol{P}_k^{i+}\right]}}} \tag{4.34}$$

which is again the same as the one obtained in Ref. [15].

Under the same Kalman Filter assumptions used to derive BD and HD, we can derive the MI measure as [6],[15]:

$$\mathrm{MI}(\boldsymbol{x}_k^i, \tilde{\boldsymbol{y}}_k^{ij}) = \frac{1}{2} \ln \left(\frac{\det\left[\boldsymbol{P}_k^{i-}\right]}{\det\left[\boldsymbol{P}_k^{i+}\right]} \right) \tag{4.35}$$

4.3 Analysis and Selection of Observability Measures

To analyze the properties of each stochastic observability measure, we consider the simple scalar case under Kalman Filter assumptions where the covariances $\boldsymbol{P}_k^{i-}$

and $\boldsymbol{P}_k^{i+}$ are reduced to the scalar variances $(\boldsymbol{\sigma}_k^{i-})^2$ and $(\boldsymbol{\sigma}_k^{i+})^2$. Then for Mutual Information, eq. (4.5) can be reduced to:

$$\mathrm{MI}(\boldsymbol{x}_k^i, \tilde{\boldsymbol{y}}_k^{ij}) = \frac{1}{2}\ln\left(\frac{\boldsymbol{\sigma}_k^{i-}}{\boldsymbol{\sigma}_k^{i+}}\right)^2 = \ln\left(\frac{\boldsymbol{\sigma}_k^{i-}}{\boldsymbol{\sigma}_k^{i+}}\right) = \ln(r) \tag{4.36}$$

Notably, this allows the ratio $r = \frac{\boldsymbol{\sigma}_k^{i-}}{\boldsymbol{\sigma}_k^{i+}}$ to represent the increase or reduction in uncertainty between the prediction and update of an estimate. If the estimator works correctly, the updated variance is lower than the predicted, resulting in a ratio that is greater than one.

Similarly, the Bhattacharya Distance in Eq. (4.32) can be reduced to:

$$\mathrm{BD}(p(\boldsymbol{x}_k^i, \tilde{\boldsymbol{y}}_k^{ij}), p(\boldsymbol{x}_k^i)p(\tilde{\boldsymbol{y}}_k^{ij})) = \frac{1}{2}\ln\left(\frac{\frac{3}{4}(\boldsymbol{\sigma}_k^{i-})^2 + \frac{1}{4}(\boldsymbol{\sigma}_k^{i+})^2}{\sqrt{(\boldsymbol{\sigma}_k^{i+})^2(\boldsymbol{\sigma}_k^{i-})^2}}\right) \tag{4.37}$$

Eq. (4.37) can then be re-arranged into the following form to be a function of the ratio $r = \frac{\boldsymbol{\sigma}_k^{i-}}{\boldsymbol{\sigma}_k^{i+}}$:

$$\mathrm{BD}(p(\boldsymbol{x}_k^i, \tilde{\boldsymbol{y}}_k^{ij}), p(\boldsymbol{x}_k^i)p(\tilde{\boldsymbol{y}}_k^{ij})) = \frac{1}{2}\ln\left(\frac{3r}{4} + \frac{1}{4r}\right) \tag{4.38}$$

Note: Both measures, MI and BD are unbounded for the highly idealized perfect estimation process, $r \to \infty$, i.e. $\sigma_k^{i+} \to 0$

Substituting Eq. (4.38) into Eq. (4.33), a reduced form of the Hellinger Distance can be similarly found:

$$\mathrm{HD}(p(\boldsymbol{x}_k^i, \tilde{\boldsymbol{y}}_k^{ij}), p(\boldsymbol{x}_k^i)p(\tilde{\boldsymbol{y}}_k^{ij})) = \sqrt{1 - \frac{1}{\sqrt{\frac{3r}{4} + \frac{1}{4r}}}} \tag{4.39}$$

Note: HD is bounded for the highly idealized perfect estimation process when $r \to \infty$, i.e. i.e. $\sigma_k^{i+} \to 0$ Figure 4.1 shows the bounded nature of the HD in comparison with MI and BD.

It has been shown that HD is the only observability measure presented that satisfies the properties of a distance function. The fact that HD is bounded also

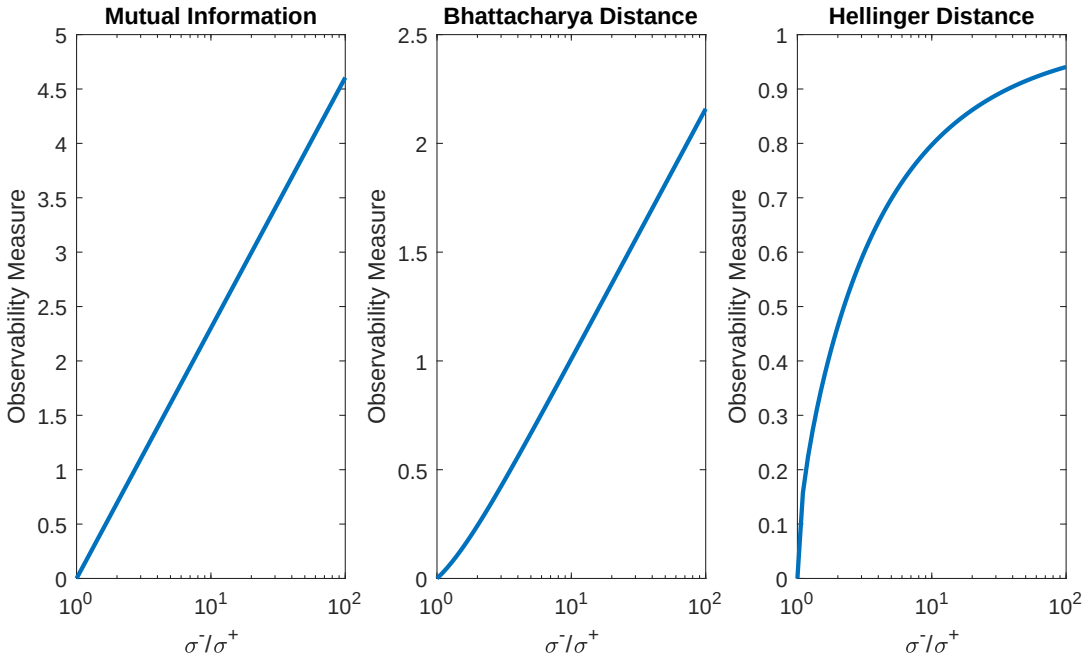

Fig. 4.1. Plot of scalar observability measures.

makes it convenient to implement this measure within a sensor tasking and fusion algorithm without having to check for limits. It can be seen that a tasking solution based on Hellinger Distance has significant benefit over other observability measures. Thus, this thesis implements the HD metric as follows:

$$\mu_H^{ij} = \sqrt{1 - \frac{\sqrt[4]{\det\left[\boldsymbol{P}_k^{i+}\right]\det\left[\boldsymbol{P}_k^{i-}\right]}}{\sqrt{\det\left[\frac{3}{4}\boldsymbol{P}_k^{i-} + \frac{1}{4}\boldsymbol{P}_k^{i+}\right]}}} \tag{4.40}$$

This tasking gain is referred to as the Hellinger Distance Gain (HDG). An estimator/tasking method with HDG is implemented and evaluated in the subsequent sections.

4.4 Verification and Analysis of Observability Measures for a Particle Filter

Section 4.2.2 derives stochastic observability measures for different estimators. While the derivations for Mutual Information, Bhattacharya, and Hellinger distance

are well-established under Kalman Filter assumptions, those derived for particle filters are not.

To identify relationships between different stochastic observability models for different estimators, two models were implemented. The first of these, a Linear-Time-Invariant (LTI) spring-mass-damper system, was used to verify that a given stochastic observability measure for a particle filter was correctly derived and implemented. The second was a simple nonlinear tracking model. This is used to evaluate a measure's ability to observe a nonlinear system. For both models, this was performed by comparing the results of EKF and UKF estimators to a set of Bootstrap Particle Filters. Each estimation algorithm was then used to determine a time history of the Mutual Information (MI) and Hellinger Distance (HD) estimates.

4.4.1 Verification of Observability Measures

It has been shown that the stochastic observability measures derived under Gaussian and Kalman Filter assumptions are exact if applied to an LTI system [15]. Notably, these measures are a function only of the predicted and posterior covariances. Given their exactness, one can use these results to verify the correctness of stochastic observability measures derived for a particle filter.

4.4.1.1 System Dynamics of LTI Spring-Mass-Damper

A spring-mass-damper with stiffness K and a dampling ratio of B was implemented. The motion of the object is defined by the differential equation:

$$\ddot{x}_k + Kx_k + B\dot{x}_k = 0$$

Both the stiffness and damping ratios were given a value of 5. This system was then converted to a discrete time system with a $\Delta t = \frac{1}{5}$ seconds:

$$\mathbf{X}_{k+1} = \Phi_k \mathbf{X}_k = \begin{bmatrix} 0.9903 & 0.1897 \\ -0.0948 & 0.8955 \end{bmatrix} \begin{bmatrix} x_k \\ \dot{x}_k \end{bmatrix}$$

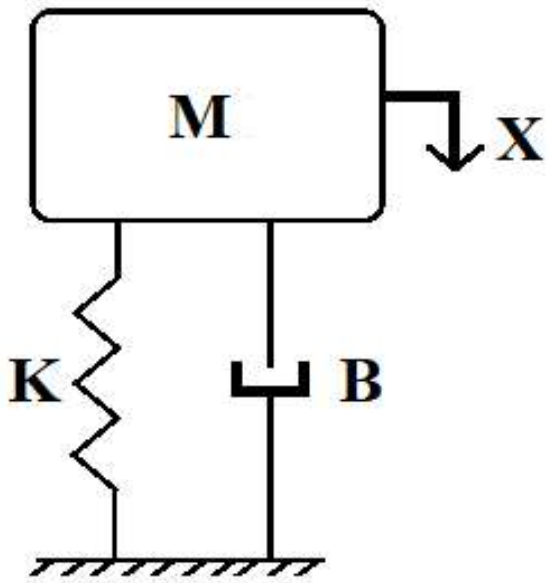

Fig. 4.2. FBD of the linear system.

No process noise is assumed for the system. A measurement given by the sensor is of the form:

$$\mathbf{Y}_k = \mathbf{H}_k \mathbf{X}_k + \nu_k = \begin{bmatrix} 1 & 0 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} x_k \\ \dot{x}_k \end{bmatrix} + \begin{bmatrix} \nu_{S1} \\ \nu_{S2} \end{bmatrix}$$

Where ν_k is the sensor noise vector. In this instance, noise is represented as additive zero-mean Gaussian noise with a standard deviation of 0.25.

4.4.1.2 System Analysis of LTI Spring-Mass-Damper

To evaluate the observability measures, the model presented was estimated by a EKF, UKF and BF for 10 seconds at a timestep of $\frac{1}{5}$th of a second. The EKF, UKF, and BF implemented follow Figures 3.1-3.3 in Chapter 3. For the EKF, $\mathbf{F}_t$ is a Jacobian of the form:

$$\mathbf{F}_t = \Phi_k \tag{4.41}$$

For the UKF, the noise is both Gaussian and additive. This allows $n = L = 2$. The following tuning parameters are used for the implementation:

$$\alpha = 0.001$$
$$\beta = 2$$
$$\gamma = \sqrt{L + \Lambda}$$
$$\Lambda = \alpha^2(L + \kappa) - L$$
$$\kappa = 3 - L$$

The Bootstrap Filter is ran three times, with 500, 1000, 5000, and 10000 particles. At each timestep, the Mutual Information and Hellinger Distance between the object and sensor set is determined. These results are shown in Figures 4.3 and 4.4. From these, it can be seen that the EKF and UKF observability measures are identical. Since the EKF and UKF are extensions of the Kalman Filter to handle nonlinear systems, it is known that these results are exact, such that they can be used as a benchmark for the particle filter results. It can also be seen that the particle filter observability measures, while of varying quantity, retain the same geometry as the Kalman Filter results. All measures see an increase in observability after the system is initially excited, with this observability decreasing as the movement of the system is damped out. While the selected particle filter results show significantly

lower observability measures, these results appear to more closely align with the exact EKF/UKF results with increasing particle amounts. This stands to reason, as a particle set can be seen as a 'container' used to approximate the distributions of the state and sensor. The larger this container is, the larger amount of information can be stored about these distributions. It can be assumed that with a sufficient amount of particles, the measures created by the EKF and UKF could be well approximated. However, this comes with increased computational cost, as seen in Table 4.1.

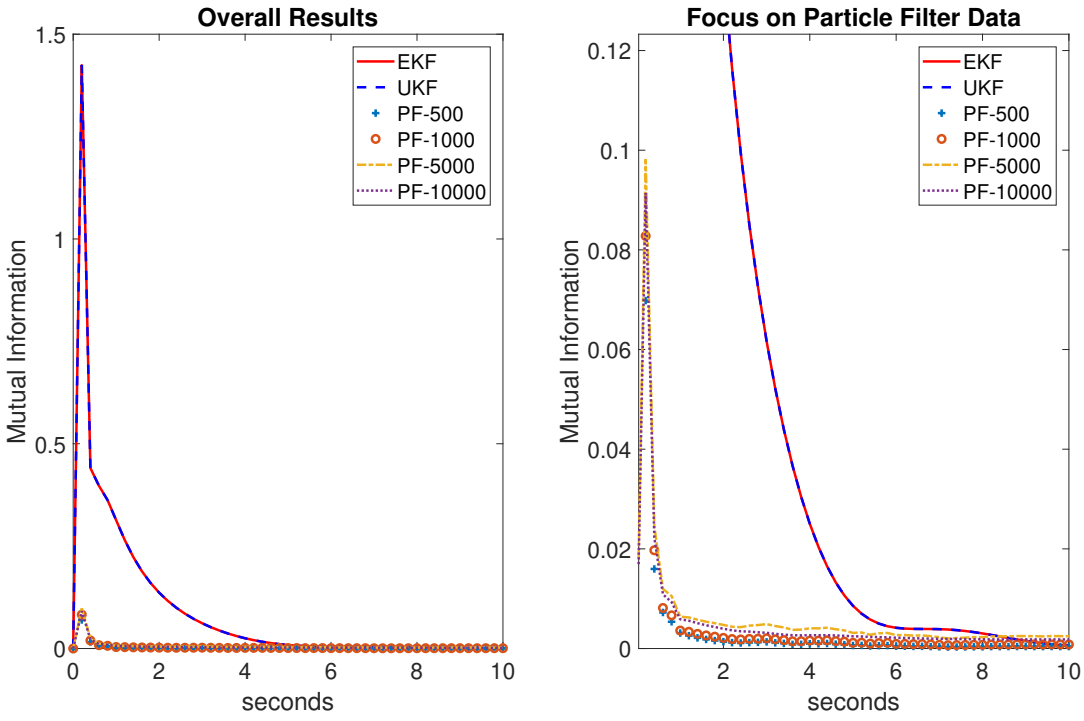

Fig. 4.3. Time history of LTI Mutual Information measures.

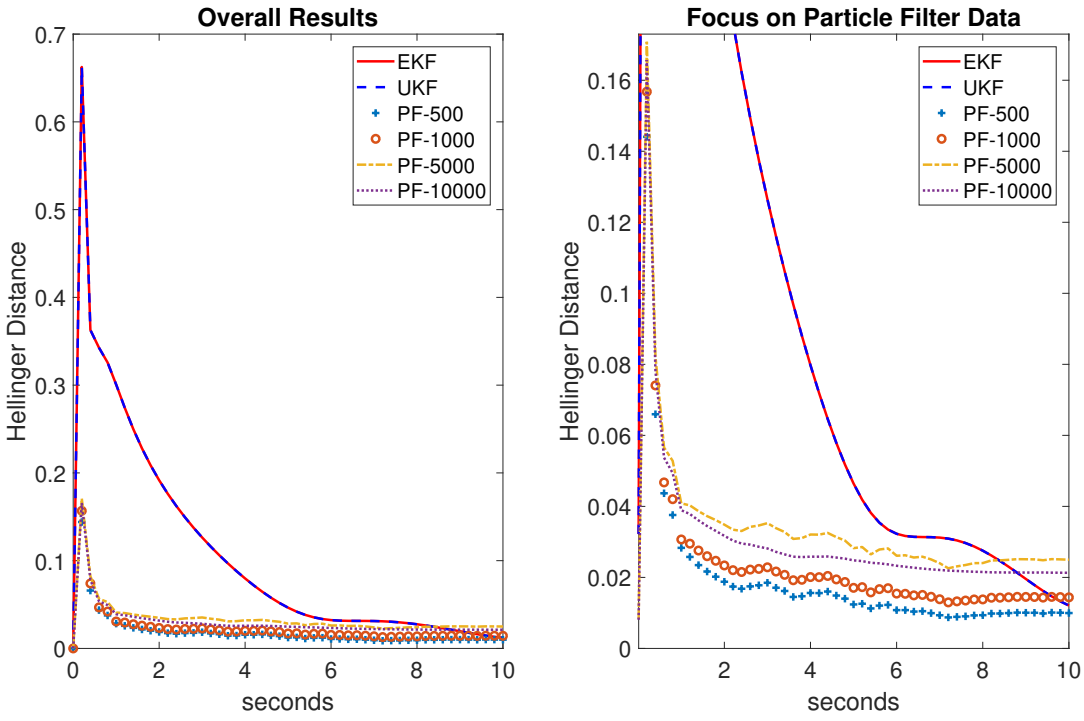

Fig. 4.4. Time history of LTI Hellinger Distance measures.

Table 4.1. Computation time for LTI System (Intel Core i5-4690K 3.50 GHz processor)

Estimator	Time, Seconds
EKF	2.73
UKF	1.40
BF, 500 Particles	30.22
BF, 1,000 Particles	59.10
BF, 5,000 Particles	347.85
BF, 10,000 Particles	839.19

4.4.2 Performance Analysis of Observability Measures

While section 4.4.1 verifies the stochastic observability derivations for a particle filter, this only evaluates the measures with a linear Guassian system. A key feature

of the particle filter derivation is that no assumption is made about the distribution or the linearity of the system. This section provides a performance analysis of the observability measures for a particle filter when applied to a simple nonlinear, non-gaussian system.

4.4.2.1 System Dynamics of Simple Nonlinear Tracking Model

Consider an object O flying in a circle about an inertial z axis. The object's path is defined by a radius A relative to the inertial z axis, and flies at a constant altitude. The object is tracked by $S1$ and $S2$, which are both range sensors.

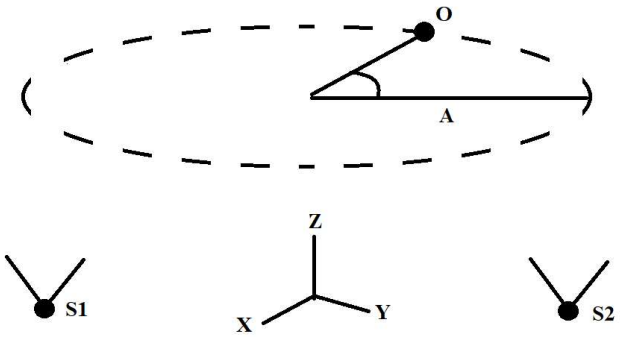

Fig. 4.5. FBD of the system modeled.

The motion of the object is defined by the equation:

$$\mathbf{X}_k = \begin{bmatrix} x_o \\ y_o \\ z_o \end{bmatrix} = \begin{bmatrix} A\cos(\omega t) \\ A\sin(\omega t) \\ 50 \end{bmatrix} \tag{4.42}$$

Where $A = 30$ and $\omega = 1.5$. No process noise is assumed for this system.

A measurement given by the sensor pair is of the form:

$$\mathbf{Y}_k = \begin{bmatrix} \rho_{S1,k} \\ \rho_{S2,k} \end{bmatrix} + \nu_k = \begin{bmatrix} \sqrt{(x_o - x_{S1})^2 + (y_o - y_{S1})^2 + (z_o - z_{S1})^2} \\ \sqrt{(x_o - x_{S2})^2 + (y_o - y_{S2})^2 + (z_o - z_{S2})^2} \end{bmatrix} + \begin{bmatrix} \nu_{S1} \\ \nu_{S2} \end{bmatrix} \tag{4.43}$$

Where ν_k is the sensor noise vector. In this instance, noise is represented as additive zero-mean Gaussian noise with variance 0.25.

4.4.2.2 System Analysis of Simple Nonlinear Tracking Model

To evaluate stochastic observability performance, the model presented was estimated by a EKF, UKF and BF for 7 seconds at a timestep of $\frac{1}{5}$th of a second. Sensors were placed at [A 0 0] and [-A 0 0]. The EKF, UKF, and BF implemented follow Figures 3.1-3.3 in Chapter 3. For the EKF, $\mathbf{F}_t$ and $\mathbf{H}_k$ are Jacobians of the form:

$$\mathbf{F}_t = \begin{bmatrix} 0 & -\omega & 0 \\ \omega & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{4.44}$$

$$\mathbf{H}_k = \begin{bmatrix} \frac{(x_o - x_{S1})}{\rho_{S1,k}} & \frac{(y_o - y_{S1})}{\rho_{S1,k}} & \frac{(z_o - z_{S1})}{\rho_{S1,k}} \\ \frac{(x_o - x_{S2})}{\rho_{S2,k}} & \frac{(y_o - y_{S2})}{\rho_{S2,k}} & \frac{(z_o - z_{S2})}{\rho_{S2,k}} \end{bmatrix} \tag{4.45}$$

For the UKF, the noise is both Gaussian and additive. This allows $n = L = 3$. The following tuning parameters are used for the implementation:

$$\alpha = 0.001$$

$$\beta = 2$$

$$\gamma = \sqrt{L + \Lambda}$$

$$\Lambda = \alpha^2(L + \kappa) - L$$

$$\kappa = 3 - L$$

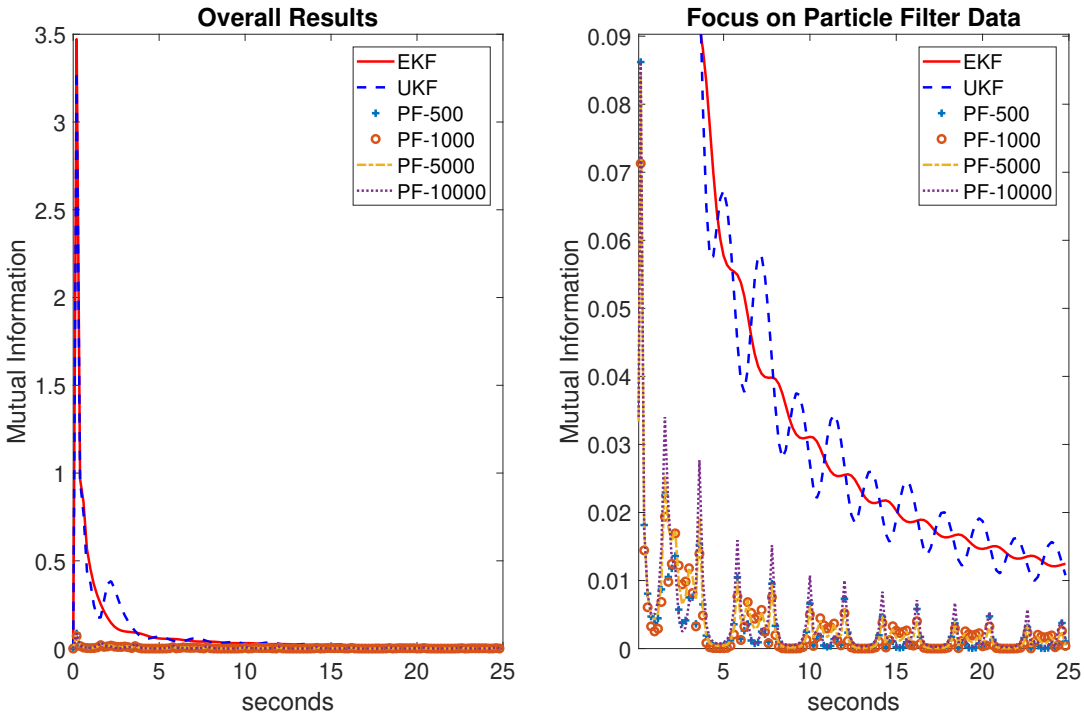

Fig. 4.6. Time history of simple nonlinear Mutual Information measures.

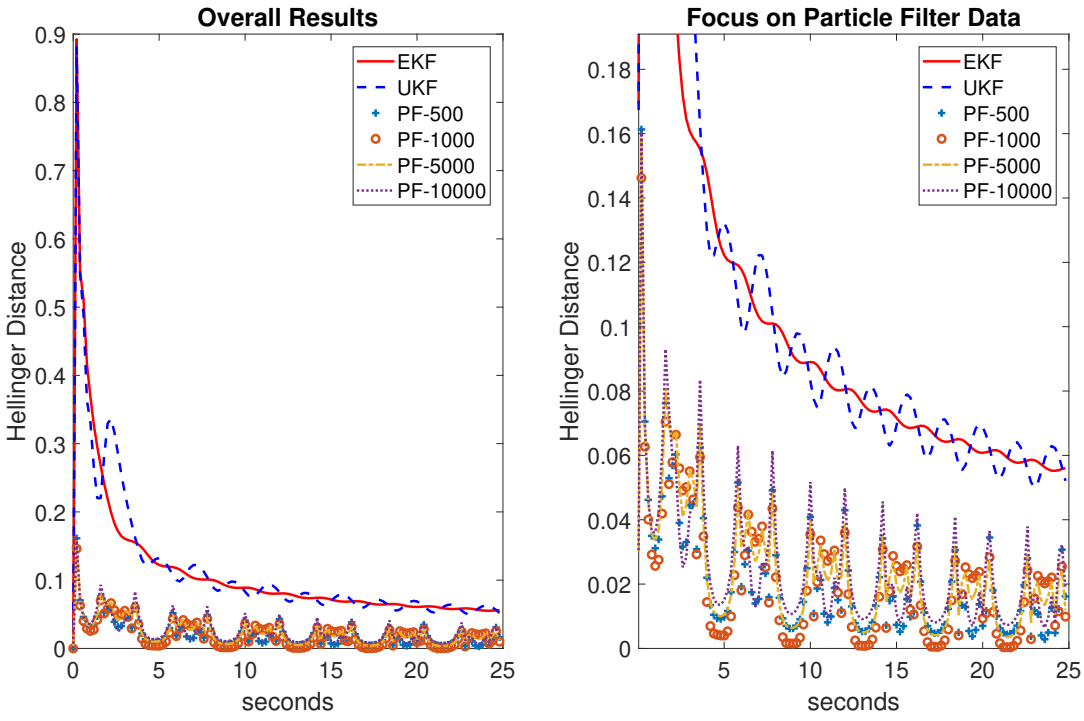

Fig. 4.7. Time History of simple nonlinear Hellinger Distance measures.

The Bootstrap Filter is ran three times, with 500, 1000, 5000, and 10,000 particles. At each timestep, the Mutual Information and Hellinger Distance between the object and sensor set is determined. These results are shown in Figures 4.6 and 4.7. From these figures, it can be seen that all but the EKF observability measures demonstrate some kind of significant sinusoidal activity. To provide insight to the behavior shown, it should be considered how a filter's error handling impacts observability measures. Given that the state can be represented in the form of the random variable $\mathbf{x}_k^i = \hat{\mathbf{x}}_k^i + \tilde{\mathbf{x}}_k^i$, where $\hat{\mathbf{x}}_k^i$ is the state mean and $\tilde{\mathbf{x}}_k^i$ is a zero-mean white-noise sequence [16], the random variable $\mathbf{y}_k^{ij}$ can be represented as $\mathbf{y}_k^{ij} = f(\hat{\mathbf{x}}_k^i + \tilde{\mathbf{x}}_k^i)$. This can be represented by the Taylor series:

$$\mathbf{y}_k^{ij} = f(\hat{\mathbf{x}}_k^i + \tilde{\mathbf{x}}_k^i) = f(\hat{\mathbf{x}}_k^i) + \nabla f \tilde{\mathbf{x}}_k^i + \nabla^2 f \frac{\tilde{\mathbf{x}}_k^{i2}}{2!} + \nabla^3 f \frac{\tilde{\mathbf{x}}_k^{i3}}{3!} + \nabla^4 f \frac{\tilde{\mathbf{x}}_k^{i4}}{4!} + ...$$

Where any sensor noise is assumed to be contained within the function of $\mathbf{y}_k^{ij}$. From this, the expectation and residual of $\mathbf{y}_k^{ij}$ can be written as:

$$\hat{\mathbf{y}}_k^{ij} = f(\hat{\mathbf{x}}_k^i) + E[\nabla f \tilde{\mathbf{x}}_k^i + \nabla^2 f \frac{\tilde{\mathbf{x}}_k^{i2}}{2!} + \nabla^3 f \frac{\tilde{\mathbf{x}}_k^{i3}}{3!} + \nabla^4 f \frac{\tilde{\mathbf{x}}_k^{i4}}{4!} + ...]$$

$$\begin{aligned} \mathbf{y}_k^{ij} - \hat{\mathbf{y}}_k^{ij} = \nabla f \tilde{\mathbf{x}}_k^i + \nabla^2 f \frac{\tilde{\mathbf{x}}_k^{i2}}{2!} + \nabla^3 f \frac{\tilde{\mathbf{x}}_k^{i3}}{3!} + \nabla^4 f \frac{\tilde{\mathbf{x}}_k^{i4}}{4!} + ... \\ -E[\nabla f \tilde{\mathbf{x}}_k^i + \nabla^2 f \frac{\tilde{\mathbf{x}}_k^{i2}}{2!} + \nabla^3 f \frac{\tilde{\mathbf{x}}_k^{i3}}{3!} + \nabla^4 f \frac{\tilde{\mathbf{x}}_k^{i4}}{4!} + ...] \end{aligned}$$

Consider that a particle filter does not characterize the distribution of $\tilde{\mathbf{x}}_k^i$, nor does its derivation explicitly truncate or eliminate terms from the Taylor series of $\mathbf{y}_k^{ij}$ or its expectation. This allows for the particle filter to account for any nonlinearities in $\mathbf{y}_k^{ij}$. This leaves the accuracy of observability measures derived in section 4.2.3 largely dependent on the number of particles represented. In this instance, the sinusoidal behavior produced by the particle filter observability measures matches ω, the frequency of the object moving around the circular path. This shows a relationship between the observability measure and the position of the object relative to the sensors.

Under the assumptions of a Kalman Filter, $\tilde{\mathbf{x}}_k^i$ is a Gaussian zero-mean white-noise sequence. Since this random variable is symmetric, the odd numbered terms in the expectation evaluate to zero:

$$\hat{\mathbf{y}}_k^{ij} = f(\hat{\mathbf{x}}_k^i) + E[\nabla^2 f \frac{\tilde{\mathbf{x}}_k^{i2}}{2!} + \nabla^4 f \frac{\tilde{\mathbf{x}}_k^{i4}}{4!} + ...]$$

$$\mathbf{y}_k^{ij} - \hat{\mathbf{y}}_k^{ij} = \nabla f \tilde{\mathbf{x}}_k^i + \nabla^2 f \frac{\tilde{\mathbf{x}}_k^{i2}}{2!} + \nabla^3 f \frac{\tilde{\mathbf{x}}_k^{i3}}{3!} + \nabla^4 f \frac{\tilde{\mathbf{x}}_k^{i4}}{4!} + ... \\ -E[\nabla^2 f \frac{\tilde{\mathbf{x}}_k^{i2}}{2!} + \nabla^4 f \frac{\tilde{\mathbf{x}}_k^{i4}}{4!} + ...]$$

Applying the Taylor series residual for a Kalman Filter, consider that $\mathbf{P}_k^{e_y e_y}$ is of the form [21]:

$$\mathbf{P}_k^{e_y e_y} = \nabla f \mathbf{P}_k^{e_x e_x} (\nabla f)^T + \frac{1}{2 \times 4!} \nabla^2 f (E[\tilde{\mathbf{x}}_k^{i4}] - E[\tilde{\mathbf{x}}_k^{i2} \mathbf{P}_k^{e_y e_y}] - E[\mathbf{P}_k^{e_y e_y} \tilde{\mathbf{x}}_k^{i2}] + \mathbf{P}_k^{e_y e_y 2})(\nabla^2 f)^T + \\ \frac{1}{3!} \nabla^3 f E[\tilde{\mathbf{x}}_k^{i4}] (\nabla f)^T + ... \tag{4.46}$$

An Unscented Kalman Filter (UKF) is derived to correctly determine the first and second moments of $\mathbf{y}_k^{ij}$, allowing it to be accurate up to the fourth order terms for the assumptions given. From equation (4.46), it can be seen that some of the nonlinearities of the system are captured through the inclusion of $\tilde{\mathbf{x}}_k^i$ in the higher order terms. An Extended Kalman Filter is derived to only approximate the first terms of equation (4.46), which aligns with the weak sinusoidal results shown in Figures 4.6 and 4.7.

Additionally, a number of trends are shown that match the behavior of the previous section. The particle filter observability measures shown are significantly lower in value than the EKF/UKF results. However, these results appear to grow with an increasing amount of particles used. Increasing observability fidelity also requires an increased computational cost, as shown in Table 4.2.

Table 4.2. Computation time for a simple nonlinear tracking System (Intel Core i5-4690K 3.50 GHz processor)

Estimator	Time, Seconds
EKF	0.78
UKF	2.45
BF, 500 Particles	51.54
BF, 1,000 Particles	103.35
BF, 5,000 Particles	680.69
BF, 10,000 Particles	1587.27

CHAPTER 5

SIMULATION ANALYSIS

The following section presents the simulation analysis used to evaluate the estimator/tasking methods presented. First, the performance metrics used to compare tasking methods are summarized. Two analysis cases are then presented. The first of these provides an overall analysis of all filter-tasker methods presented, utilizing the Bootstrap Filter set as a baseline case. The second analysis provides a closer analysis of the EKF/UKF results for better performance discrimination.

5.1 Performance Metrics

To quantify the performance of a given sensor tasking algorithm, several methods to be used are presented. To evaluate the average overall position error at a specific time instant k, the following metric is used [5]:

$$E_k^r = \frac{1}{N} \sum_{i=1}^{N} \Delta r_{i,k} \tag{5.1}$$

where

$$\Delta r_{i,k} = \sqrt{(x_{i,k} - \hat{x}_{i,k}^*)^2 + (y_{i,k} - \hat{y}_{i,k}^*)^2} \tag{5.2}$$

Using Eq. (5.1), average position error for the entire simulation can be found [5]:

$$E^r = \frac{1}{(t_f/\Delta t)} \sum_{k=0}^{t_f/\Delta t} E_k^r \tag{5.3}$$

This can be utilized as a measure of overall sensor tasking algorithm performance. Another measure is the average estimated error ellipse area for the simulation [5]:

$$\hat{A}_{err} = \frac{1}{N(t_f/\Delta t)} \sum_{i=1}^{N} \sum_{k=0}^{t_f/\Delta t} \pi \sqrt{\hat{\zeta}_{i,k}^1 \hat{\zeta}_{i,k}^2} \tag{5.4}$$

Where $\hat{\zeta}^1_{i,k}$ and $\hat{\zeta}^2_{i,k}$ are the eigenvalues of the upper 2×2 matrix of $\boldsymbol{P}^{i+}_k$ [5]. This metric measures data association. The larger the reported error area, the more uncertain an estimation algorithm is of its object set.

5.2 Analysis of Overall Performance

Using the performance metrics established in the previous section, a simulation analysis is performed to compare different tasking measure to Hellinger Distance for a EKF, UKF, and a 500 particle BPF. To perform this, simulation parameters from Ref. [5] are utilized.

Table 5.1. Sensor initial states and constraints in space object simulation

Sensor j	$x^s_{j,0}$, km	$y^s_{j,0}$, km	$\dot{x}^s_{j,0}$ km/s	$\dot{y}^s_{j,0}$ km/s	Δ_j km	Ψ_j deg
1	3.9138×10^3	6.8712×10^3	-8.2702	4.0600	10^4	180.0
2	4.1925×10^3	4.7972×10^3	-0.3498	0.3057	4.4157×10^4	10.0
3	-3.5957×10^3	5.2594×10^3	-0.3835	-0.2622	2.5371×10^4	20.0
4	5.7576×10^3	-2.7277×10^3	0.1989	0.4198	8.871×10^3	50.0

Table 5.2. Constants in space object simulation

Constant	Value
μ_e	$3.986\times10^5\ km^3/s^2$
ω_e	$7.292\times10^{-5}\ rad/sec$
$\hat{\sigma}^i_{pos,0}$	$1\ km$
$\hat{\sigma}^i_{vel,0}$	$10^{-3}\ km$
ω_{accel}	$10^{-6}\ km/s^2$
t_f	$172,328\ s$
$t_f/\Delta t$	500
(ν^{j2}_{ρ})	$0.5\ km^2$
(ν^{j2}_{ψ})	$0.1\ deg^2$

Where the initial estimates and covariance of each object are given by:

$$\hat{\boldsymbol{X}}_0^i = \boldsymbol{X}_0^i + \begin{bmatrix} N(0, \hat{\sigma}^i_{pos,0}), \\ N(0, \hat{\sigma}^i_{pos,0}) \\ N(0, \hat{\sigma}^i_{vel,0}) \\ N(0, \hat{\sigma}^i_{vel,0}) \end{bmatrix}$$

$$\left[\boldsymbol{P}_0^i\right] = \begin{bmatrix} \hat{\sigma}^2 i_{pos,0} & 0 & 0 & 0 \\ 0 & \hat{\sigma}^2 i_{pos,0} & 0 & 0 \\ 0 & 0 & \hat{\sigma}^2 i_{vel,0} & 0 \\ 0 & 0 & 0 & \hat{\sigma}^2 i_{vel,0} \end{bmatrix}$$

The continuous process uncertainty covariance $\boldsymbol{Q}$ is given by:

$$\left[Q\right] = \begin{bmatrix} \omega^2_{accel} & 0 \\ 0 & \omega^2_{accel} \end{bmatrix}$$

The sensor noise covariance $\boldsymbol{R}$ is given by:

$$\left[R\right] = \begin{bmatrix} \nu^{j2}_{\rho} & 0 \\ 0 & \nu^{j2}_{\psi} \end{bmatrix}$$

For each simulation, $N = 100$ objects are generated with a semi-major axis between $20,000 - 45,0000$ km, and eccentricities varying between $0 - 0.25$. Four ground sensors are generated, with the initial states and constraints presented in Table 5.1. In addition to the tasking measures presented, each estimator was ran with a "No Tasking" (NT) case. This case performed no tasking, and accepted all measurements within a sensor's field-of-regard. The NT cases were then used as a baseline case to establish an expected lower error bound. The Unscented Kalman Filter was tuned to provide the lowest possible error, with α set to a value of 1.

With the simulation parameters stated, a run was performed to characterize the performance of each tasking measure under a given estimation method. A typical run can be seen in Figure 5.1 and Table A.1.

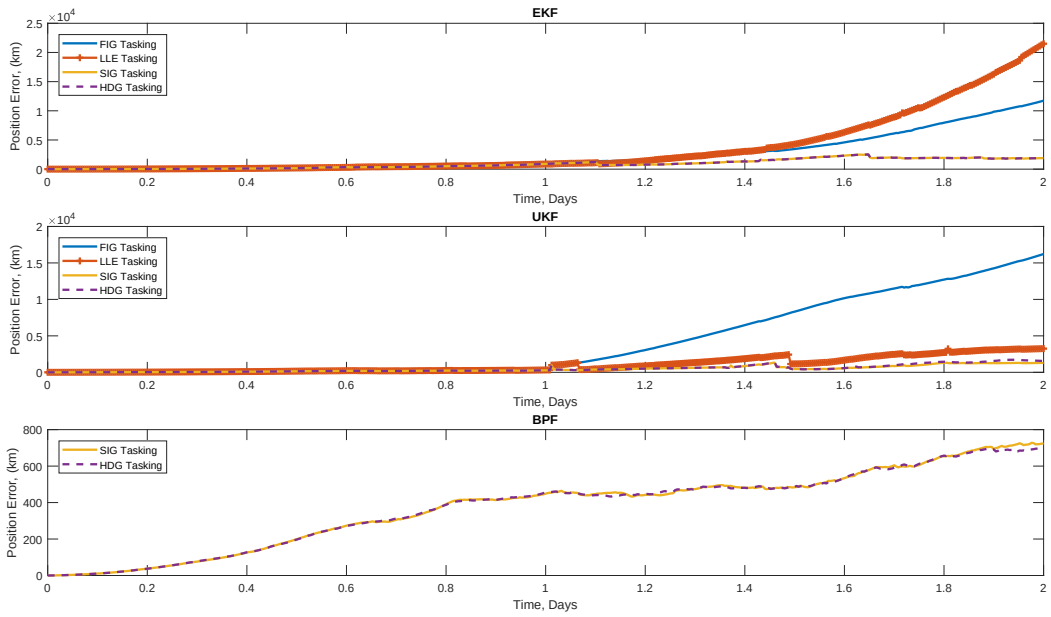

Fig. 5.1. Plot of E_k^r for all tasking methods.

From Table A.1, it can be seen that the Unscented Kalman Filter typically outperforms the Extended Kalman Filter. This aligns with the error analysis described in section 4.4.2.1 and the observations made in Refs. [5, 6]. With the inclusion of the Bootsrap Particle Filter, this estimator produces the lowest actual error. This stands to reason, as using this estimator with the measures derived in section 4.2.3 allows for a sensor tasking algorithm to account for information otherwise lost to the EKF/UKF. However, this method is also the most computationally expensive,

Table 5.3. Overall simulation results for a single run

Tasking	Filter	E^r, (km)	$\hat{A}_{\text{err}}$, (km^2)	$\frac{(E^r)_{\text{Tasking}}}{(E^r)_{\text{No Tasking}}}$	$\frac{(A_{\text{err}})_{\text{Tasking}}}{(A_{\text{err}})_{\text{No Tasking}}}$
All Data	EKF	704.006	866.314	-	-
FIG	EKF	2439.743	3357.538	3.466	3.876
LLE	EKF	3486.892	2195.665	4.953	2.534
SIG	EKF	896.324	1030.769	1.273	1.190
HDG	EKF	896.324	1030.769	1.273	1.190
All Data	UKF	523.329	53137.053	-	-
FIG	UKF	4129.279	88649.729	7.890	1.668
LLE	UKF	987.288	101451.393	1.887	1.909
SIG	UKF	450.863	71025.170	0.862	1.337
HDG	UKF	491.101	70900.292	0.938	1.334
All Data	BPF-500	239.625	59652.381	-	-
SIG	BPF-500	373.096	151024.346	1.557	2.532
HDG	BPF-500	371.700	144159.449	1.551	2.417

Table 5.4. Computation time table for NT estimator set (Intel Core i5-4690K 3.50 GHz processor)

Estimator	Time, Seconds
EKF	156.98
UKF	464.94
BF, 500 Particles	19,232.14

as shown in Table 5.4. This table shows the completion times of each "No Tasking" estimator.

As previously mentioned, a set of "No Tasking" (NT) estimators were implemented to act as a lower error bound to performance. However, it can be seen in Table A.1 that SIG/HDG tasking for the UKF case achieves a lower error than their NT counterpart. This suggests that a strategic selection of information based on an information theoretic approach can improve estimates than utilizing all available data.

For previously established tasking measures for the EKF/UKF, the results presented here largely match the observations made in Ref. [5] and Ref. [6]. When

comparing tasking measures for a given estimator, it is seen that FIG/LLE tasking are the worst performers and SIG tasking is the best. However, the run presented deviates from the prior results in that FIG tasking manages to outperform LLE tasking for the EKF. Additional instances of prior tasking measures deviating from Refs. [5, 6] were found. These are presented in Appendix A. Table A.1 also shows a performance discrepancy between the EKF and UKF for FIG tasking. Results in Appendix A shows similar occasional deviations in performance for a given filter-tasking method. With the inclusion of the HDG results, it is shown that HDG tasking outperforms SIG tasking for the BPF and is tied with SIG tasking as the best performer for the EKF, but does not outperform SIG tasking for the UKF. However, the SIG/HDG results are quite comparative across all cases. The competitiveness of SIG/HDG tasking was replicated over several runs presented in Appendix A. From these results, it is seen that HDG tasking would outperform SIG on some runs, and SIG would outperform HDG on others. Considering the deviations in behavior that occur across all tasking methods and estimators, these are attributable to the myopic aspects of the tasking methodology. The behavior of the tasking methodology is the result of Equations 4.1 and 4.2 and the tasking gain μ_{ij} only considering information gained during a given timestep. A partial exception to this is Largest Lyanpunov Exponent (LLE) based tasking, which considers prior information, but does not consider future information [5]. This results in a potential "lost opportunity" in gaining information about a system and resulting in the performance discrepancies seen here and in Appendix A. Overcoming the issue in discriminating performance is the focus of the following section.

5.3 Characterization of SIG and HDG Performance for EKF and UKF Estimators

As identified in the previous section, occasional deviations in performance for tasking measures make it difficult to identify the holistic performance of a given filter-tasking method over a single run. This is especially the case for SIG and HDG performances, which are typically comparative regardless of estimator. To better characterize performance, a Monte Carlo method is used to capture results over 50 EKF/UKF runs. A single run then observed to match the arguments presented in Chapter 4. For this analysis, the BPF is neglected due to its computational cost.

To perform the analysis described, simulation parameters from Ref. [5] are utilized:

Table 5.5. Sensor initial states and constraints in space object simulation

Sensor j	$x^s_{j,0}$, km	$y^s_{j,0}$, km	$\dot{x}^s_{j,0}$ km/s	$\dot{y}^s_{j,0}$ km/s	Δ_j km	Ψ_j deg
1	3.9138×10^3	6.8712×10^3	-8.2702	4.0600	10^4	180.0
2	4.1925×10^3	4.7972×10^3	-0.3498	0.3057	4.4157×10^4	10.0
3	-3.5957×10^3	5.2594×10^3	-0.3835	-0.2622	2.5371×10^4	20.0
4	5.7576×10^3	-2.7277×10^3	0.1989	0.4198	8.871×10^3	50.0
5	-4.9047×10^3	-4.0662×10^3	0.2965	-0.3577	3.0×10^4	15.0

Table 5.6. Constants in space object simulation

Constant	Value
μ_e	3.986×10^5 km^3/s^2
ω_e	7.292×10^{-5} rad/sec
$\hat{\sigma}^i_{pos,0}$	1 km
$\hat{\sigma}^i_{vel,0}$	10^{-3} km
ω_{accel}	10^{-6} km/s
t_f	$172,328$ s
$t_f/\Delta t$	500
(ν^{j2}_{ρ})	0.5 km^2
(ν^{j2}_{ψ})	0.1 deg^2

Where the initial estimates and covariance of each object are given by:

$$\hat{\boldsymbol{X}}_0^i = \boldsymbol{X}_0^i + \begin{bmatrix} N(0, \hat{\sigma}^i_{pos,0}), \\ N(0, \hat{\sigma}^i_{pos,0}) \\ N(0, \hat{\sigma}^i_{vel,0}) \\ N(0, \hat{\sigma}^i_{vel,0}) \end{bmatrix}$$

$$\left[\boldsymbol{P}_0^i\right] = \begin{bmatrix} \hat{\sigma}^2 i_{pos,0} & 0 & 0 & 0 \\ 0 & \hat{\sigma}^2 i_{pos,0} & 0 & 0 \\ 0 & 0 & \hat{\sigma}^2 i_{vel,0} & 0 \\ 0 & 0 & 0 & \hat{\sigma}^2 i_{vel,0} \end{bmatrix}$$

For this analysis, process uncertainty covariance $\boldsymbol{Q}_k$ was set as a discrete value given by:

$$\left[Q_k\right] = \begin{bmatrix} \omega^2_{accel} & 0 \\ 0 & \omega^2_{accel} \end{bmatrix}$$

The sensor noise covariance $\boldsymbol{R}$ is given by:

$$\left[R\right] = \begin{bmatrix} \nu^{j2}_{\rho} & 0 \\ 0 & \nu^{j2}_{\psi} \end{bmatrix}$$

For each simulation, $N = 100$ objects are generated with a semi-major axis between $20,000 - 45,0000$ km, and eccentricities varying between $0 - 0.25$. Four ground sensors and one space sensor are generated, with the initial states and constraints presented in Table 5.5. In addition to the tasking measures presented, each estimator was ran with a "No Tasking" (NT) case. This case performed no tasking, and accepted all measurements within a sensor's field-of-regard. The NT cases were then used as a baseline case to establish an expected lower error bound. The Unscented Kalman Filter was tuned to provide the lowest possible error, with α set to a value of 0.001.

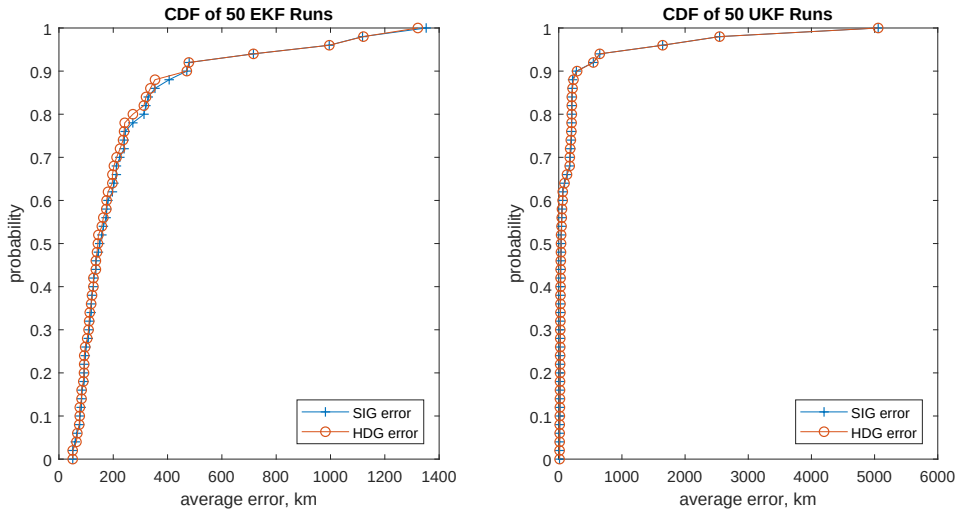

Fig. 5.2. CDF of E_r values for 50 runs.

To characterize performance, a Monte Carlo Method was used. 50 runs of the simulation were performed, with the average error and data association area collected for each run. Since the comparison of SIG and HDG tasking was of interest, a set of cumulative distribution function plots were made from the average error for these tasking measures.

From Figure 5.2, both SIG and HDG tasking show close performance regardless of error size. This is especially seen in the UKF case, where the curves shown are indistinguishable. For most of the EKF case, both curves show a similar trend. However, HDG tasking is shown to achieve a higher percentage of lower error in the $300 - 500$ km error region for the EKF case.

To determine the overall performance of each filter and tasking set, a performance metric table was generated for all 50 runs based on the tasking measures introduced in Ref. [5]. To characterize the dataset, the medians of the average error and estimated error ellipse for each tasking measure were taken. This method was

largely informed by Figure 5.2, which showed that the average of the data could be significantly skewed by a limited number of runs.

From Table 5.7, it can be seen HDG tasking provides the lowest E^r value amongst tasking measures regardless for either filter. This is done with a marginal difference in $\hat{A}_{err}$ compared to SIG tasking. Notably, these results also align with observations made about established different tasking measures in Ref. [5] and Ref .[6]. To investigate this behavior, the differences between SIG and HDG tasking are observed in a single run case. Figure 5.3 shows the tasking measures of Object 12, the object that had the most tasking differences between SIG and HDG tasking in the EKF single run case. For Object 12, there were 6 instances in which HDG tasking performed differently than SIG tasking. These were either observations shifted in time, or observations that were lost between tasking methods. In this instance, the SIG measure tasked one more time than the HDG measure.

From Figure 5.3, it can be seen that differences in tasking occurred as the measure approached the mean for all objects. This indicates that HDG tasking impacted performance most when gains were close in value. This illustrates the prioritization

Table 5.7. Median of simulation results for 50 simulation runs

Tasking	Filter	E^r, (km)	$\hat{A}_{\text{err}}$, (km^2)	$\frac{(E^r)_{\text{Tasking}}}{(E^r)_{\text{No Tasking}}}$	$\frac{(A_{\text{err}})_{\text{Tasking}}}{(A_{\text{err}})_{\text{No Tasking}}}$
No Tasking	EKF	149.454	821.979	-	-
FIG	EKF	1808.890	2252.948	12.103	2.741
LLE	EKF	1097.118	1136.046	7.341	1.382
SIG	EKF	154.043	892.324	1.031	1.086
HDG	EKF	144.438	892.321	0.966	1.086
No Tasking	UKF	32.974	664.763	-	-
FIG	UKF	720.705	2315.974	21.857	3.484
LLE	UKF	112.344	871.724	3.407	1.311
SIG	UKF	38.489	726.549	1.167	1.093
HDG	UKF	36.829	726.653	1.117	1.093

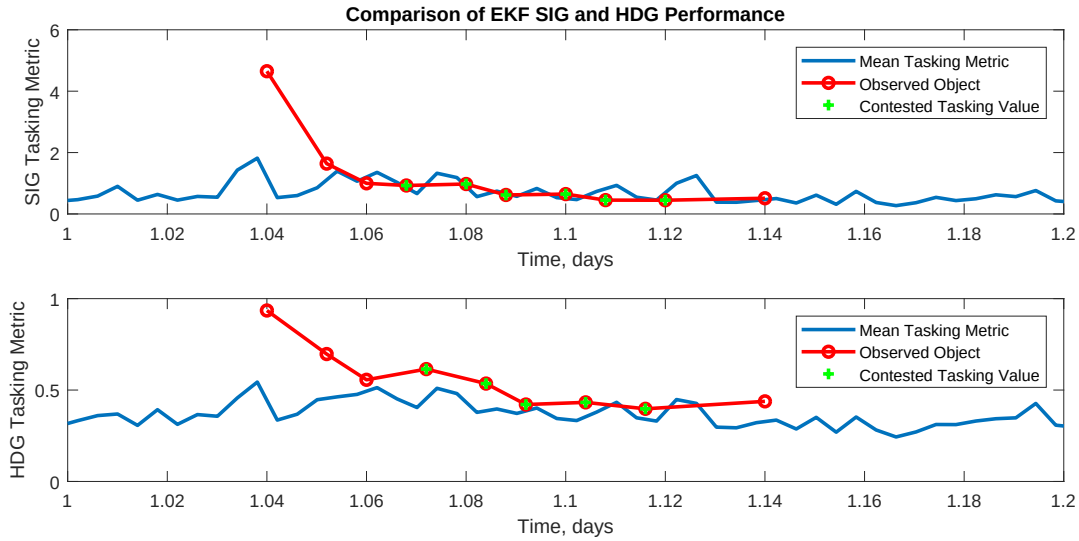

Fig. 5.3. Time history of the EKF SIG and HDG tasking measures for object 12.

scheme of HDG tasking; as gains become closer in value, HDG tasking can more accurately measure the distance between distributions and task accordingly. Notably, Object 12 achieved a E_i^r of 157.121 kilometers under HDG tasking, where it only achieved 196.414 kilometers under SIG tasking. With this performance gain being achieved with one fewer measurement than SIG tasking, this allowed HDG tasking to prioritize additional objects.

Figure 5.4 illustrates the tasking measures of Object 74, the object that had the most tasking differences between SIG and HDG tasking in the UKF case. For Object 74, there were 5 instances where tasking differences occurred. All tasking differences were the result of shifts in time, with no loss of observations for either measure.

The behavior of HDG tasking in Figure 5.4 matches what is seen in Figure 5.3; HDG tasking differed from SIG tasking as object tasking occurred near the mean tasking measure. Object 74 achieved a E_i^r of 4.752 kilometers under HDG tasking, and 4.806 kilometers under SIG tasking. While HDG tasking experiences a performance

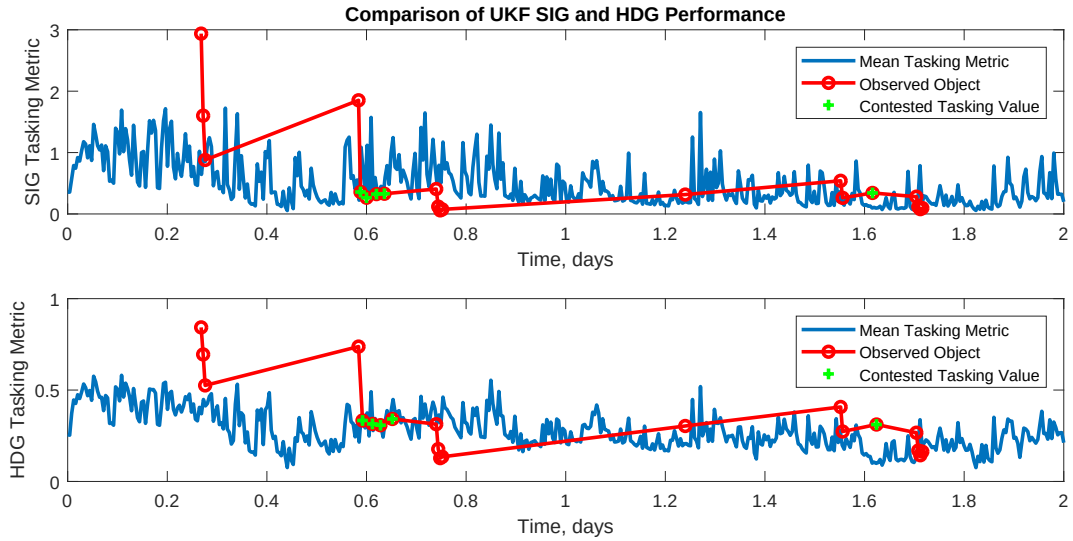

Fig. 5.4. Time history of the UKF SIG and HDG tasking measures for object 74.

increase, it is marginal compared to the EKF case. This result, along with no losses in observation between tasking measures is an indication that SIG tasking retains more of its competitiveness when a UKF is used.

CHAPTER 6

CONCLUDING REMARKS AND FUTURE WORK

This thesis presents a new sensor tasking measure for space object tracking. This measure tasks sensors by measuring the observability measure between a given sensor and object using the Hellinger Distance. This measure was chosen for satisfying the properties of a distance function and to address the unbounded aspects of the Shannon Information Gain measure in sensor tasking problems.

The Hellinger Distance Gain (HDG) tasking was evaluated against a set of previously established measures and applied to a EKF, UKF, and BPF. Additionally, a new derivation was made and verified to evaluate the HDG measure for a Particle Filter.

Notably, it was observed that HDG tasking has a close performance to SIG tasking. This close performance and the myopic nature of the tasking solutions impacted the ability to characterize these results. As a result, the EKF and UKF were evaluated with a Monte Carlo analysis with the median results taken. It was found that HDG tasking outperformed all other tasking measures for both filters, with a lower median error than all other measures. This was done with a estimated error area comparable to the SIG tasking case.

During the analysis of the EKF/UKF Monte Carlo data sets, the largest divergence of performance between tasking methods occurred in the EKF case. This was observed in both the median error, as well as differences in object tasking. UKF SIG tasking was much more competitive with UKF HDG tasking. This was observed through a lower discrepancy in median error, as well as fewer differences in object

tasking. Since it is known that a UKF is less susceptible to error loss than an EKF, it appears that SIG tasking retains its competitiveness in an environment with lower error. It is possible that a greater performance divergence can be seen for either filter if higher sensor error was experienced.

The close performance between SIG and HDG tasking was also used to illustrate the differences between the tasking solutions. By investigating the objects that had the greatest tasking differences between the two measures, it was shown that most tasking differences occurred as tasking gains became more competitive. This analysis, along with the lower error shown, illustrated the intended behavior of HDG tasking. It showed that HDG tasking provided a more accurate means of comparing object observability than other established measures for the simulation conditions given.

6.1 Future Work

This thesis presents work that shows significant promise in utilizing Hellinger Distance as a method to compare system observability in the context of space object tracking. Some future work includes:

- Providing a proof that the linear programming problem presented in Equations (4.1)-(4.2) is well-defined, as this is not provided by Ref. [4].
- Evaluating and Improving the computational efficiency of a given filter-tasker algorithm. This includes:
  - Evaluating the time complexity of the problem.
  - Investigating parallel computing techniques.
- Analysis of information theoretic values in a non-myopic sensor tasking method.
- Applying the problem to a higher fidelity space object tracking environment. This includes identifying the appropriate process noise size based on orbital regime and coupling the sensor tasking algorithm with data association tech-

niques. Shortening the timestep may also provide greater fidelity with orbits at lower altitudes, however this will come with an additional computational cost. If made short enough, relativity effects will also need to be accounted for.

- Applying information theoretic based sensor tasking to additional problems involving object tracking. Autonomous vehicle navigation may be a potential avenue.

APPENDIX A

Three Simulation Runs for Overall Performance Analysis

A.1 Run 1 Results

NOTE: This is the same run presented in section 5.2.

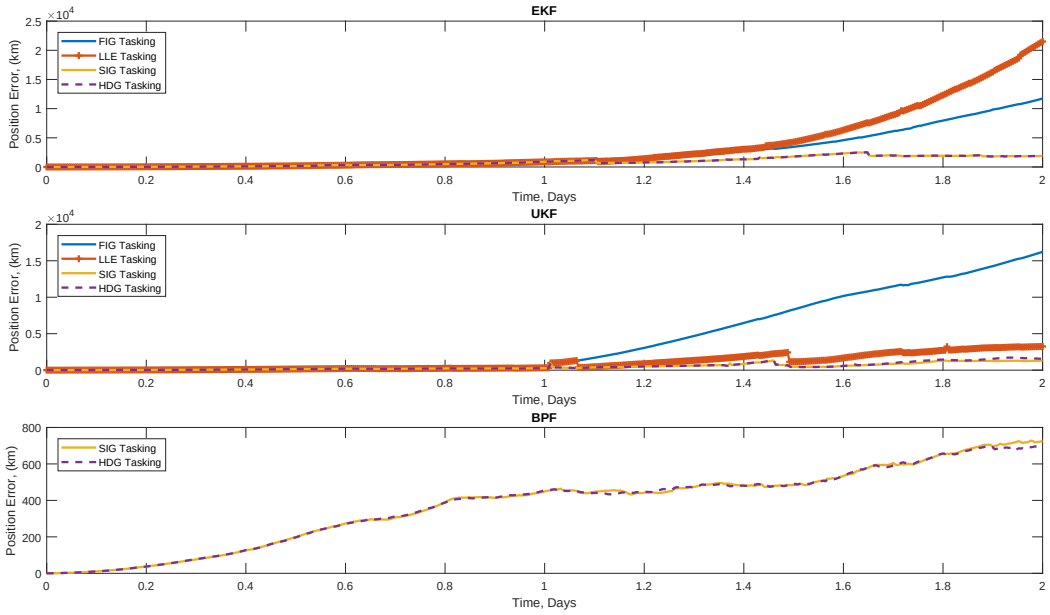

Fig. A.1. Run 1 plot of E_k^r for all tasking methods.

Table A.1. Overall simulation results for a single run

Tasking	Filter	E^r, (km)	$\hat{A}_{\text{err}}$, (km^2)	$\frac{(E^r)_{\text{Tasking}}}{(E^r)_{\text{No Tasking}}}$	$\frac{(A_{\text{err}})_{\text{Tasking}}}{(A_{\text{err}})_{\text{No Tasking}}}$
All Data	EKF	704.006	866.314	-	-
FIG	EKF	2439.743	3357.538	3.466	3.876
LLE	EKF	3486.892	2195.665	4.953	2.534
SIG	EKF	896.324	1030.769	1.273	1.190
HDG	EKF	896.324	1030.769	1.273	1.190
All Data	UKF	523.329	53137.053	-	-
FIG	UKF	4129.279	88649.729	7.890	1.668
LLE	UKF	987.288	101451.393	1.887	1.909
SIG	UKF	450.863	71025.170	0.862	1.337
HDG	UKF	491.101	70900.292	0.938	1.334
All Data	BPF-500	239.625	59652.381	-	-
SIG	BPF-500	373.096	151024.346	1.557	2.532
HDG	BPF-500	371.700	144159.449	1.551	2.417

A.2 Run 2 Results

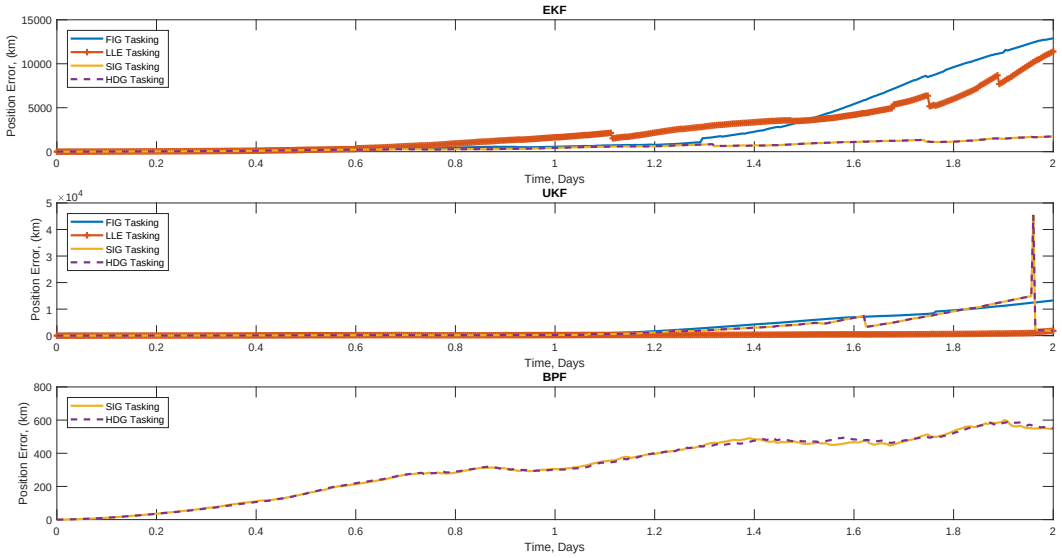

Fig. A.2. Run 2 plot of E_k^r for all tasking methods.

Table A.2. Overall simulation results for run two

Tasking	Filter	E^r, (km)	$\hat{A}_{\text{err}}$, (km^2)	$\frac{(E^r)_{\text{Tasking}}}{(E^r)_{\text{No Tasking}}}$	$\frac{(A_{\text{err}})_{\text{Tasking}}}{(A_{\text{err}})_{\text{No Tasking}}}$
All Data	EKF	533.780	795.425	-	-
FIG	EKF	2594.792	2656.067	4.861	3.339
LLE	EKF	2439.376	1261.589	4.570	1.586
SIG	EKF	569.282	874.922	1.067	1.100
HDG	EKF	569.282	874.922	1.067	1.100
All Data	UKF	295.569	140126.850	-	-
FIG	UKF	2932.389	191378.428	9.921	1.366
LLE	UKF	301.762	276658.305	1.021	1.974
SIG	UKF	2463.214	190174.865	8.334	1.357
HDG	UKF	2465.346	193051.928	8.341	1.378
All Data	BPF-500	190.616	55466.491	-	-
SIG	BPF-500	310.927	130323.030	1.631	2.350
HDG	BPF-500	311.741	130946.963	1.635	2.361

A.3 Run 3 Results

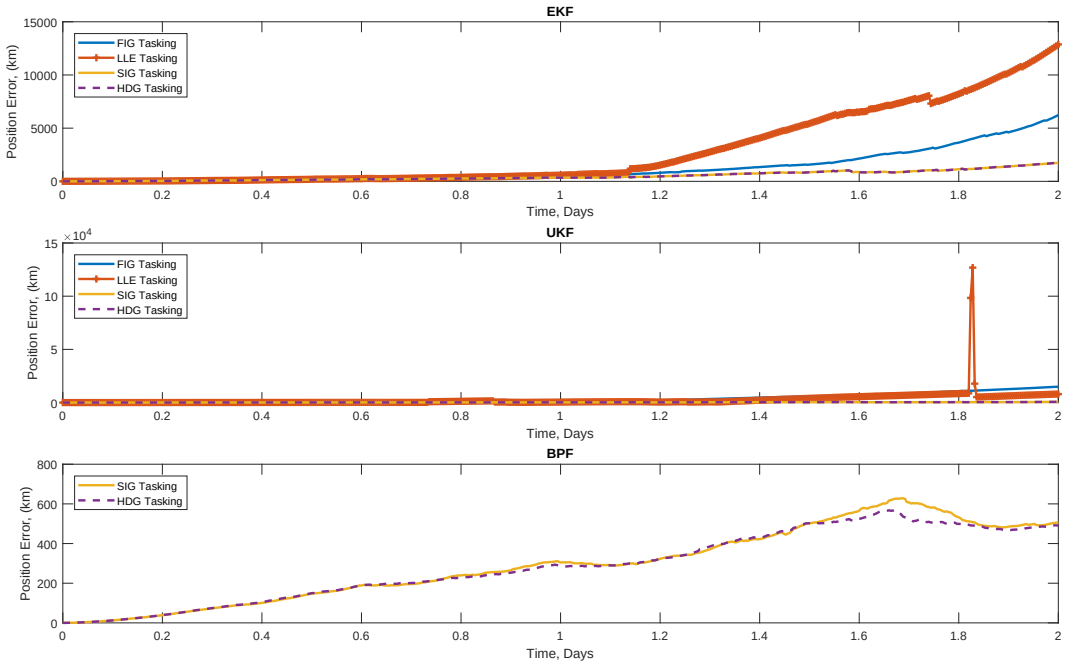

Fig. A.3. Run 3 plot of E_k^r for all tasking methods.

Table A.3. Overall simulation results for run three

Tasking	Filter	E^r, (km)	$\hat{A}_{\text{err}}$, (km^2)	$\frac{(E^r)_{\text{Tasking}}}{(E^r)_{\text{No Tasking}}}$	$\frac{(A_{\text{err}})_{\text{Tasking}}}{(A_{\text{err}})_{\text{No Tasking}}}$
All Data	EKF	460.443	830.652	-	-
FIG	EKF	1200.923	2082.009	2.608	2.506
LLE	EKF	2795.367	1337.062	6.071	1.610
SIG	EKF	500.908	921.069	1.088	1.109
HDG	EKF	500.908	921.069	1.088	1.109
All Data	UKF	905.944	973387.780	-	-
FIG	UKF	3216.668	91383.727	3.551	0.094
LLE	UKF	2521.739	13336941117.396	2.784	13701.570
SIG	UKF	226.601	73131.103	0.250	0.075
HDG	UKF	229.560	73172.873	0.253	0.075
All Data	BF-500	179.383	57457.141	-	-
SIG	BF-500	295.986	129777.439	1.650	2.259
HDG	BF-500	286.920	129871.827	1.599	2.260

REFERENCES

[1] R. Sridharan and A. F. Pensa, "U.s. space surveillance network capabilities," *Proceedings of SPIE*, vol. 3434, pp. 88–100, 1998.

[2] F. Hoots, "Keeping track: Space surveillance for operational support," *Crosslink, Fall 2015*, vol. 16, no. 1, 2015.

[3] S. Kan, "China's anti-satellite weapon test," *CRS Report for Congress*, vol. RS22652, 2007.

[4] R. S. Erwin, P. Albuquerque, S. K. Jayaweera, and I. Hussein, "Dynamic sensor tasking for space situational awareness," *2010 American Control Conference*, pp. 1153–1158, 2010.

[5] P. S. Williams, D. B. Spencer, and R. S. Erwin, "Coupling of estimation and sensor tasking applied to satellite tracking," *Journal of Guidance, Navigation, and Control*, vol. 36, no. 4, pp. 993–1007, 2013.

[6] P. S. Williams, "Coupling between nonlinear esimation and dynamic sensor tasking applied to satellite tracking," *The Pennsylvania State University, The Graduate School*, 2012.

[7] R. Linares and R. Furfaro, "An autonomous sensor tasking approach for large scale space object cataloging," in *Advanded Maui Optical and Space Surveillance Technologies Conference (AMOS)*, pp. 1–17, 2017.

[8] C. Cai and S. Ferrari, "Information-driven sensor path planning by approximate cell decomposition," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 39.

[9] M. Chu, H. Haussecker, and F. Zhao, "Scalable information-driven sensor querying and routing for ad hoc heterogeneous sensor networks," *International Journal of High Performance Computing Applications*, vol. 16, no. 3, pp. 293—313, 2002.

[10] F. Zhao, J. Shin, and J. Reich, "Information-driven dynamic sensor collaboration," *IEEE Signal processing magazine*, vol. 19, no. 2, pp. 61—72, 2002.

[11] K. Vishwajeet, P. Singla, and M. Jah, "Nonlinear uncertainty propagation for perturbed two-body orbits," *AIAA Journal of Guidance, Control, and Dynamics*, vol. 37, no. 5, p. 1415–1425, 2014.

[12] V. Mnih, A. P. Badia, M. Mirza, A. Graves, T. Lillicrap, T. Harley, D. Silver, and K. Kavukcuoglu, "Asynchronous methods for deep reinforcement learning," in *International Conference on Machine Learning*, pp. 1928–1937, 2016.

[13] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. New York, NY: Wiley-Interscience, 1st ed., 1991.

[14] M. Basseville, "Distance measures for signal processing and pattern recognition," *Signal Processing*, vol. 18, pp. 349–369, 1989.

[15] Y. Subasi and M. Demirkler, "Quantitative measure of observability for stochastic systems," *Proceedings of the 18th IFAC World Congress*, pp. 4244–4249, September, 2011.

[16] J. Crassidis and J. Junkins, *Optimal Estimation of Dynamic Systems*. 6000 Broken Sound Parkway NW, Suite 300, Boca Raton, Florida: CRC Press, 2nd ed., 2012.

[17] B. M. Silverman, *Density Estimation for Statistics and Data Analysis*. London, New York: Chapman-Hall/CRC, 1986.

[18] R. Ash, *Real Variables with Metric Space Topology*. 445 Hoes Lane, PO Box 1331, Piscataway, NJ 08855-1331: IEEE Press, 1993.

[19] K. Fukunaga, *Introduction to Statistical Pattern Recognition*. Academic Press, 2nd ed., 1990.

[20] G. M. Hoffman and C. J. Tomlin, "Mobile sensor network control using mutual information methods and particle filters," *IEEE Transactions on Automatic Control*, vol. 55, no. 1, pp. 32–47, 2010.

[21] S. Julier, J. Uhlmann, and H. F. Durrant-Whyte, "A new method for the nonlinear transformation of means and covariances in filters and estimators," *IEEE Transactions on Automatic Control*, vol. 43, no. 3, pp. 477–482, 2000.

[22] E. Ertin, J. Fisher, and L. Potter, "Maximum mutual information principle for dynamic sensor query problems," in *In Proc. IPSN'03*, April 2003.

[23] J. Liu, J. Reich, and F. Zhao, "Collaborative in-network processing for target tracking," *EURASIP JASP: Special Issues on Sensor Networks*, vol. 4, pp. 378–391, 2003.

[24] B. O. Teixeira, M. A. Santillo, R. S. Erwin, and D. S. Bernstein, "Spacecraft tracking using sampled-data kalman filters," *IEEE Control Systems Magazine*, vol. 28, no. 4, pp. 78–84, 2008.

[25] B. O. S. Teixeira, M. A. Santillo, R. S. Erwin, and D. S. Bernstein, "Spacecraft tracking using sampled-data kalman filters," *IEEE Control Systems Magazine*, vol. 28, no. 4, pp. 78–94, 2008.

[26] C. Shannon, "A mathematical theory of communication," *Bell System Technical Journal*, vol. 27, pp. 379–423, 1948.

[27] B. Upadhyhya and H. W. Sorenson, "Bayesian discriminant approach to input signal selection in parameter estimation problems," *Information Sciences*, vol. 12, 1977.

[28] F. Harrou, M. Madakyaru, and Y. Sun, "Nonlinear partial least squares with hellinger distance for nonlinear process monitoring," *2016 IEEE Symposium Series on Computational Intelligence (SSCI)*, 2016.

BIOGRAPHICAL STATEMENT

Mitchel T. McDonald was born in Arlington, Texas, in 1992. He received his B.S. degree from Texas A&M University in 2016 in Aerospace Engineering. While pursuing his B.S., he participated in AggieSat Lab as the Attitude, Determination, and Control Systems (ADCS) Lead for the AggieSat 4 mission.

In 2017, he joined The University of Texas at Arlington to pursue a M.S. in Aerospace Engineering. His interests include spacecraft and missile Guidance, Navigation, and Control.